ABSTRACT

Title of Dissertation ENTROPY APPROACH TO META-MODELING, MULTI-OBJECTIVE GENETIC ALGORITHM, AND QUALITY ASSESSMENT OF SOLUTION SETS FOR DESIGN OPTIMIZATION

Ali Farhangmehr, Doctor of Philosophy, 2003

Dissertation directed by: Professor Shapour Azarm
Department of Mechanical Engineering

A new entropy-based approach to meta-modeling and multi-objective optimization of engineering design problems is presented. The approach consists of four main components, as follows:

1. Meta-Modeling: Engineering design optimization problems often involve computationally costly simulation models. Multi-objective optimization of such models usually involves many function evaluations that prohibit a direct application of most available techniques. In this dissertation, a new sequential meta-modeling technique -- referred to as Sequential MAXimum Entropy Design, or SMAXED -- is presented that aims at finding a good meta-model with minimum computational burden.

2. Multi-Objective Genetic Algorithm (MOGA): We introduce a new multi-objective genetic algorithm that aims at obtaining the most

diverse (i.e., highest entropy) solution set. The new MOGA – referred to as Thermodynamical MOGA or T-MOGA -- is based on simulating Maxwellian system (of mono-atomic gas molecules in a container).

3. Minimality of Quality Indexes: Once a Pareto solution set to a multi-objective optimization problem is obtained via a multi-objective optimization algorithm, it is usually of great interest to know how 'good' the observed solution set represents the Pareto frontier. This can be done either visually, or objectively via quality indexes. In this part of research, a new theoretical framework is presented for selection of a handful of these indexes such that all desired aspects of quality are addressed with minimum or no redundancy.

4. Entropy Index: Finally, to assure the quality of solution sets in terms of diversity, a new quality index is presented. The new index -- referred to as entropy index -- is based on the notion of entropy.

In situations where a direct application of most optimization techniques is computationally intractable, the proposed SMAXED approach can be employed to construct a global approximation to the simulation model, followed by T-MOGA to obtain a diverse solution set. Using a carefully selected set of quality indexes assures an objective performance assessment and comparison of the proposed methodology.

ACKNOWLEDGMENTS

First and foremost I would like to thank my advisor, Professor Shapour Azarm, for his expertise, support, and guidance in the development of this research. I would also like to thank my thesis committee members, Professor Linda Schmidt, Professor Peter Sandborn, Professor Arjang Assad, and Professor Amr Baz for their invaluable insight and inputs to improve the thesis. I would like to acknowledge my fellow graduate students in the Design Decision Support Laboratory, who played an important role in the development of this research. The work presented here was supported in part by a National Science Foundation Grant DMI-0112767, and in part by a U.S. Office of Naval Research contract N000149810842. These supports are gratefully acknowledged.

I am grateful to Shayan and Okhtay for their valuable inputs and many hours that we spent discussing various aspects of science. I am indebted to Micha Kemp for her taking the time to proofread parts of my thesis. I would also like to thank Mr. and Mrs. Iman and Haleh Nikmaram for their help and advice in numerous occasions in the past two years. Finally, I am indebted to my parents for their dedication and support throughout my education.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

NOMENCLATURE

DOE	Design Of Experiments
GA	Genetic Algorithm
H	Shannon Entropy/Boltzmann Entropy/Entropy Index
MOGA	Multi-Objective Genetic Algorithm
MOO	Multi-Objective Optimization
NDS	Non-Dominated solution Set
NSGA	Nondominated Sorting Genetic Algorithm
O(.)	Order of computation complexity
RMSE	Root Mean Square Error
SMAXED	Sequential MAXimum Entropy Design
T-MOGA	Thermodynmical Multi-Objective Genetic Algorithm
VEGA	Vector Evaluated Genetic Algorithm

APPROXIMATION (CHAPTER 2)

B_m^i	i-th block of m experiments
D	A designed set of experiments
$\overline{D}$	Complement of D in U
$D_{k\times m}$	Set of Experiments after k blocks of m experiments each
H	Shannon Entropy
I_0	Initial information (prior)

$I(y_D)$	Posterior information
$L(\mathbf{x}_i)$ or L_i	Characteristic Certainty Width at $\mathbf{x}_i$
$R(\|\mathbf{x}_i-\mathbf{x}_j\|)$	Correlation function of two points $\mathbf{x}_i$ and $\mathbf{x}_j$
U	Set of all possible design variable vectors
$\mathbf{V}_{DD}$	Prior covariance matrix (of points in D)
$\mathbf{V}_{\overline{D}\overline{D}\|D}$	Posterior covariance matrix (of points in $\overline{D}$)
$\mathbf{x}$	Design variable vector
$y(\mathbf{x})$	Response to design vector $\mathbf{x}$
$\hat{\mathbf{y}}_{\overline{D}}$	Estimated response values (of points in $\overline{D}$)
Y_i or $Y(\mathbf{x}_i)$	Prior process at $\mathbf{x}_i$
$Y_{i\|D}$	Posterior process at $\mathbf{x}_i$
μ_i	Mean of Y_i
σ_{ii}	Variance of Y_i
σ_{ij}	Covariance of Y_i and Y_j (prior)
$\sigma_{ii\|D}$	Covariance of Y_i and Y_j (posterior)
σ'_{ij}	Updated covariance of Y_i and Y_j
θ	Correlation Parameter

MULTI-OBJECTIVE GENETIC ALGORITHMS (CHAPTER 3)

$\boldsymbol{A_i}$	Offset Vector at point $\mathbf{x}_i$

$\boldsymbol{C_i}$	Progress Vector at point $\mathbf{x}_i$
$f(\mathbf{x},\mathbf{v},t)$	Density of particles at point (**x, v**) in state-space
H	Boltzmann's Entropy
n	Dimension of the **x** space (3-dimensional for a gas)
N	Number of particles (solution points)
N_j	Number of solution points with rank j
$P_1, P_2, \ldots, P_{N1}$	Solution points with rank 1
$P_{N1+1}, P_{N1+2}, \ldots, P_{N1+N2}$	Solution points with rank 2
$\mathbf{x}$	Position of a particle
$\mathbf{v}$	Speed of a particle

MINIMALITY OF QUALITY INDEXES (CHAPTER 4)

C	C metric
$InfI$	Inforiority index
OS_k	Overall Spread with respect to k-th objective
$Q(A,B)$	Quality Index comparing sets A and B
$Q \sim R$	Quality Index Q is compatible with excellence relation R
Q_c	Coverage index
R	Excellence relation (general)
R_O	Outperformance excellence relation
R_C	Coverage excellence relation
R'_C	Modified coverage relation

S	*Hypercubic Size of Non-dominated Space*
$x \succeq y$	Point x dominates point y
$\mathbf{\Phi}$	Set of excellence relations
Λ_R	Partially ordered domain of excellence relation R
$\mathbf{\Gamma}$	Set of quality indexes
ρ	Correlation
$<>$	Mean

ENTROPY INDEX (CHAPTER 5)

B_0	Bad (Nadir) point before normalization
B	Normalized bad point, i.e. (1,…,1)
D	Density function
f_k	k-th objective (before normalization and transformation)
f_i'	k-th objective (after normalization and transformation)
$\mathbf{f}'$	Vector of normalized objectives
F^m	Feasible design variable domain ($F^m \subseteq R^m$)
$G_0 = (g_1,\ldots, g_i,\ldots, g_m)$	Good (Ideal) point before normalization
G	Normalized good point, i.e. (0,…,0)
H	Entropy Index
N	Number of solution points
$\mathbf{u}_k$	Cartesian unit vectors along f_k'
$\mathbf{v}_k$	k-th unit vector after Gram-Schmidt orthogonalization

(X, Y)	Cartesian coordinates on the projected hyperplane (2-dimensiona)
Ω_i	Influence function at point $\mathbf{x}_i$
ρ_{ij}	Normalized density at cell (i,j)

CRASHWORTHINESS DESIGN (CHAPTER 7)

f_1	Maximum force (during crash event)
f_2	Maximum displacement (during crash event)
x_1	Collapse strength of connector material (MPa)
x_2	Sheet metal thickness of rail forward (mm)
$\hat{y}_{f_1}$	Meta-model for f_1 (maximum force)
$\hat{y}_{f_2}$	Meta-model for f_2 (maximum displacement)
z_1	Normalized x_1
z_2	Normalized x_2
τ	Standard deviation of the correlation function
θ	Correlation function parameter

CHAPTER 1

INTRODUCTION

This dissertation presents an entropy-based approach to meta-modeling and multi-objective optimization of engineering design problems. In particular, a new sequential entropy-based approach is presented that adaptively constructs an interpolative surrogate model to approximate computationally expensive objective and/or constraint functions. Also, a new Multi-Objective Genetic Algorithm (MOGA) is developed that aims at obtaining a non-dominated solution set with the 'best' possible distribution over the Pareto frontier (See Appendix I for terminology and definitions, and Appendix II for a brief description of Genetic Algorithms and MOGA). The dissertation also includes an entropy-based quality index that quantifies the goodness of a Pareto solution set in terms of its distribution quality. Finally, an engineering case study and several test cases are presented to demonstrate the merits of the proposed approach.

1.1 BACKGROUND AND MOTIVATION

Engineering design optimization problems often involve single or multiple design objectives with variables that are mixed discrete and continuous. Several stochastic and heuristic optimization techniques have been applied to these types of problems, such as ant colony optimization (Colorni et al. 1992), particle swarm optimization (Kennedy and Eberhart, 1995), tabu search (Pham and Karaboga, 2000), simulated annealing (Kirkpatric et al. 1983) and genetic algorithms (Goldberg, 1989).

However, when there are multiple conflicting objectives, multi-objective evolutionary algorithms such as MOGAs have been receiving significant attention. MOGAs have proved to be successfully applicable to a wide variety of engineering applications with discrete, integer and/or continuous variables (see Van Veldhuizen and Lamont 2000; Deb 2001; Coello Coello et al. 2002; for a comprehensive literature survey). The main idea behind a MOGA is to simulate biological evolution wherein the principles of natural selection and survival of the fittest are the driving force for improvements in a population of design alternatives (see Appendix II for further details).

Although MOGAs are capable of generating a non-dominated solution set that approximates the Pareto frontier in a single run, there are some major drawbacks in constructing a widely applicable multi-objective optimization tool based on these algorithms. According to Deb (1998), there are two issues that a good multi-objective optimization tool should address while obtaining a non-dominated solution set: 1) keeping the computational effort as low as possible while generating sufficient number of non-dominated solutions using the available resources -- e.g., time, computational power; and 2) maintaining diversity among these solutions. In reality, however, the solutions obtained via MOGA tend to group into clusters, degrading the quality of the solution set in terms of diversity of solutions (Zitzler 1999; Farhang-Mehr et al. 2001). Moreover, a MOGA usually requires a large number of function calls that may become computationally prohibitive specifically in complex real-world applications. Therefore, *the overall objective of this dissertation is to develop an approximation-assisted multi-objective optimization framework for engineering design problems such that the best*

representation of the optimal design space is obtained with minimum computational effort.

1.2 DISSERTATION'S RESEARCH COMPOENETS

The dissertation has four main research components. These components include the development of: 1) a new sequential meta-modeling technique; 2) a new MOGA based on concepts from statistical thermodynamics; 3) a new theoretical framework for selection of a minimal set of quality indexes; and 4) a new entropy index. An overall framework is developed that can be used to integrate these four research components.

The motivation and research objectives for each of these components are presented next.

1.2.1 Research Component 1: Meta-Modeling

One main issue that limits the application of MOGAs to real-world engineering design optimization problems is that MOGAs may require many (objective and/or constraint) function evaluations during the optimization process. These functions – also referred to as responses -- can be computationally expensive. One approach to reduce the computational effort or number of function evaluations is to approximate the responses by less expensive surrogate models. There are many different surrogate or (meta-) modeling techniques in the literature (see Barthelemy and Haftka 1993; Simpson et al. 1998a; Koch et al. 1999). Among these techniques, the most common approach is to choose a sample of design points via a classical Design Of Experiment (DOE) technique (defined later in Chapter 2) and fit a polynomial to the response values. These techniques, however, are mostly appropriate for physical models that have inherent randomness, and thus, are not capable of providing sufficiently accurate

surrogates for deterministic computer simulations. This raises a need for the development of more advanced approaches that are capable of dealing with deterministic computer simulations with limited computational burden. Research Objective 1 addresses this issue.

Research Objective 1: *To develop a new sequential entropy-based design of experiments approach that aims at obtaining the best possible representation of the design space for the purpose of building a good approximation model of the objective and/or constraint functions.*

This DOE approach -- referred to as Sequential MAXimum Entropy Design (SMAXED) -- is designed to retrieve maximum possible information about the behavior of the function being approximated while maintaining the computational expenses as low as possible (see Chapter 2 for details).

1.2.2 Research Component 2: Thermodynamical MOGA

Solution sets obtained from MOGAs usually tend to cluster in certain sub-regions of the Pareto frontier -- leaving the rest of the Pareto frontier empty or sparsely populated (Deb, 1998). The challenge in this part of the research is to develop a MOGA that maximizes diversity of solutions as much as possible in order to better represent the entire Pareto frontier. Research Objective 2 addresses this issue.

Research Objective 2: *To develop a new MOGA, one that aims at obtaining a Pareto solution set with maximum possible diversity along the Pareto frontier.*

The new MOGA -- referred to as Thermodynamical MOGA (or T-MOGA) -- is based on simulating a Maxwellian system (of mono-atomic gas molecules in a container). It is shown that such a system always tends to re-distribute itself to achieve a steady state of maximum entropy (see Chapter 3 for details).

1.2.3 Research Component 3: Minimality of Quality Indexes

Once a Pareto solution set is obtained by a MOGA, it is usually of great interest to know how 'good' the observed solution set represents the exact Pareto frontier. Note that relying on visual assessment of a solution set is no longer sufficient for a comparison of different evolutionary multi-objective optimization techniques. Indeed, visual and intuitive quality assessment can be very misleading or even impossible when the number of design objectives is more than three, yet it is the prevailing comparison tool in the literature (Van Veldhuizen and Lamont 1998). Recently, there has been a growing attention to the development of 'quality indexes' that measure different aspects of solution set quality (e.g. Zitzler et al. 2002; Knowles 2002). Many of the reported indexes, however, are coupled in the sense that they address common aspects of quality. Taking insufficient care in selecting quality indexes for comparing MOGAs can introduce redundancy and inconsistency in the comparison conclusions. On the other hand, selecting too few of these quality indexes does not guarantee an exhaustive comparison with respect to all aspects of quality -- posing a new question to explore: which index or collection of indexes must be used for an exhaustive (but not redundant) comparison of different MOGAs? Research Objective 3 addresses this issue.

Research Objective 3: *To investigate correlation and redundancy issues among quality indexes. To derive theoretical restrictions (necessary conditions) for selecting a minimal collection of quality indexes that address all aspects of quality (set forth by a decision-maker) exhaustively with minimum redundancy.*

Due to the subjective nature of quality indexes, and quality of a set of solutions in general, the above-mentioned properties may not be noticeable from the formulation of an index. Therefore, a theoretical framework is developed to classify and select a minimal set of quality indexes for a given problem based on inputs from a decision-maker (see Chapter 4 for details).

1.2.4 Research Component 4: Entropy Index

There are two major classes of quality indexes (Zitzler 1999; Deb 2001). Some indexes determine how close a solution set is to the exact Pareto frontier, which in turn indicates the Pareto optimality of the solution set. In contrast, there are other indexes that address the distribution quality of the solutions over the Pareto frontier. This latter class of indexes is specifically beneficial in comparison studies of different MOGAs (as well as other multi-objective optimization techniques) in terms of their ability to produce and maintain diversity among the solution points. In the development of some of these indexes, however, it is usually assumed that the exact Pareto frontier is known a priori, which rarely holds in engineering optimization problems. Moreover, these indexes are generally based on the distances of every two solution points in the population, and thus have a complexity of $O(N^2)$ that may become computationally

prohibitive for a large number of solution points (Deb 2001). Research Objective 4 addresses this issue.

Research Objective 4: T*o formulate a new quality index -- referred to as entropy index -- to address diversity of solutions based on the concept of entropy.*

The proposed entropy index has several advantages over similar indexes reported in the literature (see Chapter 5 for details).

1.2.5 Overall Framework: Approximation-Assisted Optimization

In Chapters 6, the above-mentioned approaches are applied to several engineering and numerical test examples. In Chapter 7, an overall approximation-assisted optimization framework is presented in the context of a case study: crashworthiness design of the front-end of a pickup truck. This overall framework starts with sequential approximation of computationally expensive objective/constraint functions using SMAXED approach. T-MOGA is then used to find a non-dominated solution set to the optimization problem. In the mean time, the entropy quality index is used throughout the optimization process to monitor the quality of the obtained solutions.

1.3 ORGANIZATION OF DISSERTATION

The organization of the rest of this dissertation is as follows. Chapter 2 presents a sequential maximum-entropy design approach (Research Component 1). Chapter 3 is devoted to an approach to multi-objective optimization, i.e., T-MOGA (Research

Component 2). In Chapter 4, a theoretical framework is developed for selection of a minimal set of quality indexes (Research Component 3), while Chapter 5 is focused on the development of a quality index, namely entropy index (Research Component 4). Chapter 6 applies the proposed methodologies to several engineering and numerical test examples. Chapter 7 combines the above mentioned research components into an overall framework and demonstrates a case study: crashworthiness design of front-end of a pickup truck. Finally, Chapter 8 gives the conclusions, contributions and recommended future research directions for the dissertation.

Figure 1.1 depicts the flow of information in this dissertation.

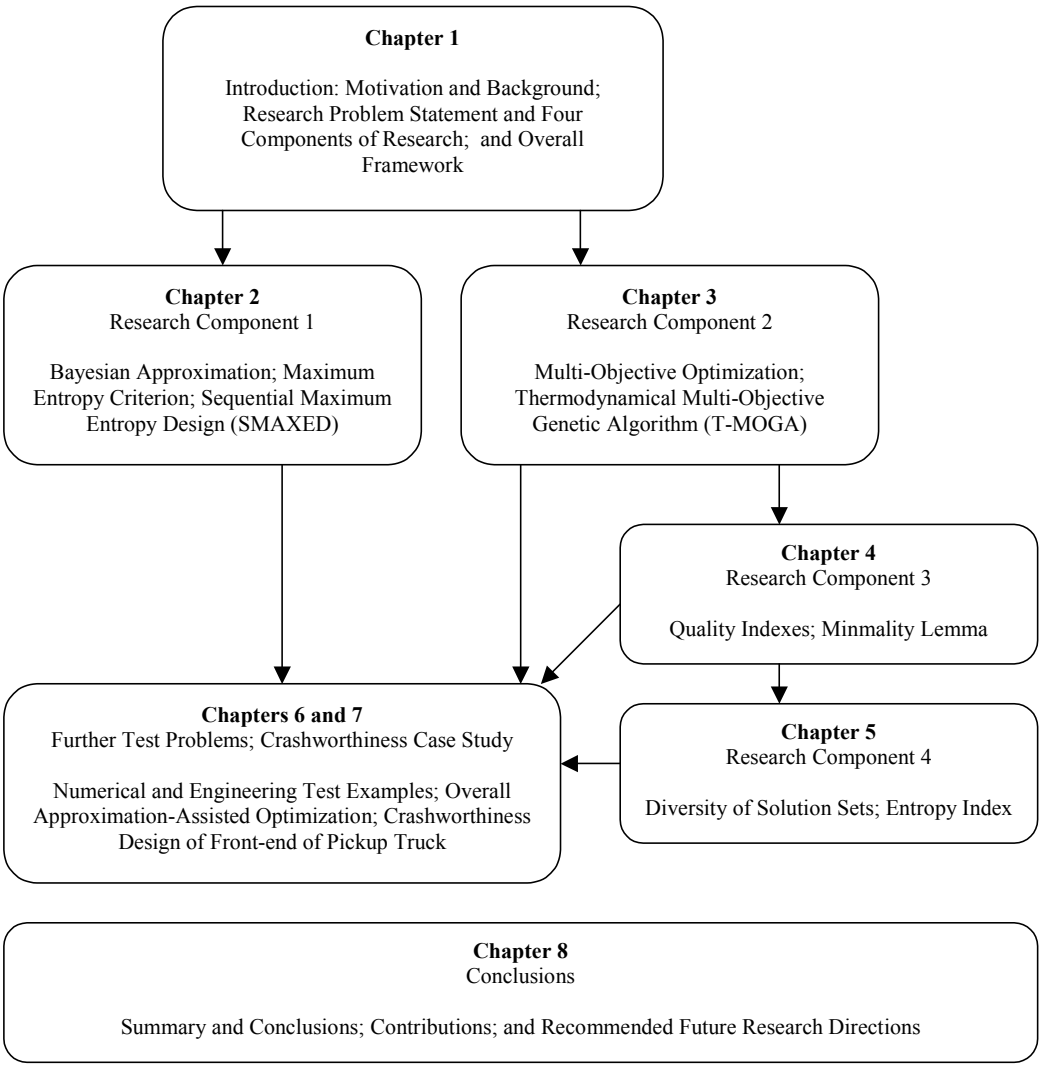


Figure 1.1: Organization of dissertation

CHAPTER 2

SEQUENTIAL MAXIMUM ENTROPY APPROXIMATION

As complex computer simulation models are opening a new frontier in analysis and optimization of engineering problems, designers are often faced with situations where the expense of running many simulation runs remains, for the most part, an unresolved research problem. Multi-objective optimization of such problems, in particular, often requires numerous evaluations of objective and/or constraint functions that can rapidly become computationally prohibitive. According to Koch et al. (1999) there are two issues in a direct-function-evaluation optimization approach that makes it almost inapplicable to real-world problems: 1) number of variables; and 2) computational expenses. Indeed, comprehensive parametric analyses are becoming extremely time-intensive as the number of variables involved in the design process increases. In response to this growing problem, many researchers have tried to decouple complex systems into several concurrent sub-systems (disciplines) wherein each discipline is optimized separately with the least possible inter-disciplinary information flow. Despite this reduction in the number of variables within a discipline, the function evaluations can still remain computationally expensive or perhaps even grow as a whole for the system. This issue becomes more critical in evolutionary algorithms wherein a significant number of function evaluations is needed to simultaneously evolve a population of solutions throughout the process -- raising the

need for development of more efficient approximation techniques to reduce the computational burden.

In this chapter, we first briefly review different approximation techniques reported in the literature, and especially Bayesian meta-modeling and maximum entropy design strategy. In Section 2.2, a new sequential approximation approach is presented, hereafter referred to as SMAXED, which will be applied to and demonstrated with a simple test case in Section 2.3. Finally, the concluding remarks of this chapter are given in Section 2.4.

2.1 BACKGROUND AND PREVIOUS WORK

There are two main steps in approximation of a computationally costly response function: 1) Design of Experiments (DOE), where a set of vector of design variables (inputs) are selected and the corresponding response values are computed; 2) Meta-modeling, where a meta-model (or surrogate) is constructed based on the response values.

In the following, we briefly review some of the most common approximation methodologies that are reported in the literature.

2.1.1 Design of Experiments (DOE) Techniques

No matter which approximation technique is used to construct a meta-model, it is always necessary to design a good sample of points that best represents the design space prior to creating a meta-model.

Perhaps, full factorial design is the most trivial and widely used DOE approach. In this approach, the allowable range of variables is divided into equally-spaced levels. Then a response surface is built by an exhaustive function evaluation at each level.

However, the number of function evaluations in this simple approach grows rapidly as the number of response variables increases. As such many other techniques have been developed that promise lower computational effort. Simpson et al. (2001) classify the existing DOE techniques into two major groups.

1) *Classical DOE*: This class of techniques accounts for inherent randomness in the behavior of the model. Central Composite Design (Myers and Montgomery, 1995) and Box-Behnken (Box, 1987) are among the most widely used classical DOE techniques. These techniques are primarily aimed at physical experimentations in which -- because of the inherent measurement errors -- fixing the input variables and repeating the same experiment could result in different outputs (i.e., different response function values). One common approach for handling this randomness in the response is to design several replicates of the same experiment with the same input values. These classical techniques are generally well established and relatively easy to apply. Therefore, as computer simulation of physical models became prevalent in virtually every area of science and technology, there was a tendency among researchers to adopt the classical notion of DOE for this rapidly growing deterministic application. Sacks et al. (1989), Simpson et al.(2001) as well as other researchers discuss that while the classical notion of DOE is developed to accommodate the existence of random experimental errors, the modern simulation codes generally produce deterministic results. Hence, design of several replicates for an experiment is no longer relevant and in fact results in unnecessary computational effort.

2) *Space Filling DOE*: A second class of techniques was developed in which experiments were dispersed to 'fill the design space'. These techniques are especially

appropriate for design and analysis of computer experiment in which a limited number of experiments are conducted to probe the response of an expensive simulation code (e.g., a complex finite element structural model) at certain points in the design space and construct a much less expensive yet reasonably accurate approximation model. This approximation model can then replace the original expensive simulation for applications in which a large number of function calls may be required (e.g., design optimization). Examples of this second class of techniques are Random Latin Hypercube, Orthogonal Array, Integrated Mean-Squared Error, MaxiMin, MiniMax and Maximum Entropy experimental designs. (For a review of experimental design and analysis of deterministic computer experiments, see, for instance, Kohler and Owen 1996, as well as Simpson et al. 1997, 1998b.)

2.1.2 Meta-Modeling Techniques

Barthelemy and Haftka (1993) group the existing meta-modeling techniques into three major categories: local, global, and mid-range techniques. In the following, brief descriptions of local and global techniques are given. Mid-range approximation is essentially a combination of both local- and long-range approaches.

1) *Local Approximation*: In this class of approximation techniques, the approximated function is only valid in a small neighborhood of interest. This can be particularly of great interest in fine-tuning of the optimal design around a previously-known good design. Perhaps the most popular approach in this category involves polynomial models, typically linear or quadratic functions, usually created by performing a least square fit to a set of points (Giunta, 1998). These polynomial models are known as

response surface models (see, Myers and Montgomery, 1995, for a detailed description). These methods are popular because they are very simple and computationally inexpensive to create. Moreover, they provide a compact and explicit functional relationship between the response and the independent variables. The main shortcoming of these approximations is that they are valid only in a small neighborhood.

2) *Global Approximation*: This class of approximation techniques is valid in the entire design space and thus is favorable in global optimization of complex design problems. Since first- and second-order local response surface techniques are not capable of global modeling of complex functions with arbitrary shapes, their global implementation is mostly limited to continuous, unimodal objective/constraint functions, which is unlikely to be the case in many engineering design problems. As such, global modeling of the optimization problem generally relies heavily on selecting a good sample of design experiments and constructing the best representative global model based on these experiments.

Among different DOE and meta-modeling techniques, the Bayesian approach has received significant attention in the past two decades. Especially for deterministic computer experiments, Bayesian techniques provide interpolative meta-models that go through all experiment points (as expected from meta-modeling of a deterministic response function). In the following section, a brief description of Bayesian meta-modeling is given and our notation is introduced. This approach and notation is used in the rest of this chapter to introduce single-stage entropy design, and eventually SMAXED.

2.1.3 Interpolative Approximation: A Bayesian Viewpoint

The main idea behind the Bayesian approach is simple: the prior and posterior (before and after conducting experiments) knowledge about the response function is modeled by random processes (see for instance, Currin et al. 1993; Chaloner and Verdinelli 1995; Koehler and Own 1996; and Pacheco et al. 2001). Bayes rule can then be applied to update the prior distributions and obtain posterior distributions. The following is a brief description of Bayesian meta-modeling (Koehler and Owen 1996). This concept will be also used in Sections 2.1.4-6 to explain maximum entropy design.

Consider a deterministic response function, $y(\mathbf{x})$, with a vector of p input variables, denoted as $\mathbf{x}$. Here we assume that each element of this vector is bounded between 0 and 1. (If this is not the case, a linear transformation can be used to map the lower and upper bounds of $\mathbf{x}$ to 0 and 1, respectively.) So, $\mathbf{x} \in [0,1]^p$. A grid is constructed in this space -- each point (or node) of the grid representing one possible input vector for the response function. The grid size can be chosen such that it satisfies the required accuracy for the input variables. This essentially reduces the design space into a finite set of p-tuples in $[0,1]^p$, denoted by U. Note that such discrete representation is always possible for computer simulation codes, since the input variables are always discrete (digital), regardless of their representation format. The unknown (or actual) response function, y, is fully explored if and only if $y(\mathbf{x})$ is evaluated for all $\mathbf{x} \in U$. As mentioned earlier, an exhaustive exploration of the design space is impossible for most of the real-world computer simulation codes because of the computational cost. Therefore, only a handful of these input vectors must be chosen to represent the entire set.

Definition 2.1.3-1: A set of experiments, denoted by D, is defined as a subset of U, at each point of which an experiment is run and the corresponding response function (or response) is evaluated. Specifically, an n-design, denoted by D_n, is a design of size n with the set of responses represented by:

$$y_D = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_i, y_i), \ldots, (\mathbf{x}_n, y_n)\} \tag{2.1}$$

where

$$y_i = y(\mathbf{x}_i) \tag{2.2}$$

Equation 2.2 is deterministic and contains no inherent randomness. Put another way, for a certain input vector $\mathbf{x}_i$, the corresponding response value y_i is unique no matter how many times the computer simulation is run. Naturally, $\overline{D}$ is defined as the complement of D in U:

$$\overline{D} = \{\mathbf{x} : \mathbf{x} \in U; \mathbf{x} \notin D\} \tag{2.3}$$

Since we do not conduct experiments at the points in $\overline{D}$, their response values remain unknown. Therefore, the main problem is to gain knowledge about responses in $\overline{D}$ by conducting experiments in D and observing y_D. This can be done using a Bayesian approach, as discussed below.

Suppose the prior knowledge about a deterministic function y_i, at a point $\mathbf{x}_i$, is represented by a normal process, namely Y_i. (The normality assumption is chosen for the formulation simplicity.) This prior distribution has the expected value and variance of:

$$E(Y_i)=\mu_i \quad \text{and} \quad Var(Y_i)=\sigma_{ii} \tag{2.4}$$

However, Y_i and Y_j (i.e., two prior distributions at $\mathbf{x}_i$ and $\mathbf{x}_j$, respectively) are not statistically independent and their covariance is:

$$\mathrm{Cov}(Y_i, Y_j)=\sigma_{ij} \tag{2.5}$$

Therefore, the prior knowledge about the response for a given design, D_n, is represented by a multivariate normal process, Y_D. Put another way, y_D (which is deterministic) is a realization of the multivariate normal process, Y_D. Y_D is a vector whose elements are the prior distributions, i.e., Y_i, $\forall\, \mathbf{x}_i \in D_n$. The expected value of this vector is: $E[Y_D] = \boldsymbol{\mu}_D = [\mu_i]$, where μ_i is defined in Equation 2.4. The elements of Y_D are statistically dependent and the nxn covariance matrix is:

$$\mathbf{V}_{DD}=\mathrm{Cov}[Y_D, Y_D] = [\sigma_{ij}]_{n\mathrm{x}n} \; ; \quad \mathbf{x}_i, \mathbf{x}_j \in D_n \tag{2.6}$$

After performing n experiments in D_n and observing y_D, the posterior distribution of $\overline{D}$ is represented by $Y_{\overline{D}|D}$ which is also multivariate normal. The values of y at these points are not evaluated directly, so there is still uncertainty in the response values in $\overline{D}$. The posterior covariance matrix for the posterior process, i.e. $\mathbf{V}_{\overline{D}\overline{D}|D}$, is then:

$$\mathbf{V}_{\overline{D}\overline{D}|D} = \mathrm{Cov}[Y_{\overline{D}}, Y_{\overline{D}} \mid y_D] = [\sigma_{ij|D}]; \quad \mathbf{x}_i, \mathbf{x}_j \in \overline{D}_n \tag{2.7}$$

where $\sigma_{ij|D}$ is the covariance of the two posterior distributions at $\mathbf{x}_i$ and $\mathbf{x}_j$; $\mathbf{x}_i, \mathbf{x}_j \in \overline{D}_n$. For Gaussian distributions, the posterior covariance matrix of $\overline{D}$ can be obtained from prior distributions as follows:

$$\mathbf{V}_{\overline{D}\overline{D}|D} = \mathbf{V}_{\overline{D}\overline{D}} - \mathbf{V}_{\overline{D}D}\mathbf{V}_{DD}^{-1}\mathbf{V}_{\overline{D}D}^{T} \tag{2.8}$$

where: $\mathbf{V}_{\overline{D}D} = \mathrm{Cov}[\mathrm{Y}_{\overline{D}}, \mathrm{Y}_D]$. The posterior process is especially important in a Bayesian prediction of the response function. It is well-known that assuming a quadratic loss function, a Bayesian estimate for y in $\overline{D}$ after observing y_D is the mean of the posterior distribution (which minimizes the loss, see Casella and Berger 1990). In other words:

$$\hat{\mathbf{y}}_{\overline{D}} = \boldsymbol{\mu}_{\overline{D}|D} = \boldsymbol{\mu}_{\overline{D}} + \mathbf{V}_{\overline{D}D}\mathbf{V}_{DD}^{-1}(\mathbf{y}_D - \boldsymbol{\mu}_D) \tag{2.9}$$

where $\mathbf{y}_D$ is the vector of responses, i.e., $\mathbf{y}_D=[y_1, \ldots y_n]^T$. $\boldsymbol{\mu}_{\overline{D}|D}$ is a vector whose elements are the mean of the posterior processes of $\overline{D}$, and $\hat{\mathbf{y}}_{\overline{D}}$ estimates the response values for points in $\overline{D}$. This equation is the essence of Gaussian interpolation or kriging, the purpose of which is to construct an approximation model over U by observing y_D and assuming Gaussian distributions (Koehler and Owen 1996; Giunta et al. 1998). While the focus of this chapter is on design of experiments, nevertheless, we use the above relation to create an approximation model every time that a set of experiments is designed and evaluated.

Although the above interpolative surrogate modeling is formulated in a compact form (equation 2.9), it is not of any use in practice without knowing the covariance matrix of the prior processes. In the following section, a stationary (non-informative) condition is assumed to resolve this issue.

2.1.4 Stationary Assumption

The stationary assumption states that prior processes are stationary in the sense that their means and variances are the same everywhere in the design space (non-informative assumption). Moreover, the covariance of two points is only a function of

their distance and does not depend on their position in the design space. This is, in some sense, a non-informational and non-discriminatory assumption in that the prior process is taken to be identical everywhere in the design space. This assumption is stated formally next.

Stationary Assumption: The covariance matrix is stationary provided that $\sigma_{ij} = \sigma^2 R(\|\mathbf{x}_i - \mathbf{x}_j\|)$, where $R(.)$ is the correlation function which is monotonically decreasing with the distance between two points. Moreover, $R(.)$ satisfies $R(0)=1$, thus $\sigma_{ii}=\sigma^2$. The mean of the prior processes are also assumed to be identical: $\mu_i=\mu;\ \forall \mathbf{x}_i \in U$.

Based on the stationary assumption, for a correlation function that depends only on the Euclidian distance of two points, $\mathbf{V}_{DD}$ is invariant to any isometric transformation (e.g., translation, rotation) in the position of experiments in U. Indeed, the only thing that matters is the relative distance of the points while their absolute placement in the domain is ignored, i.e.,

$$\mathbf{R}_{DD}=\frac{1}{\sigma^2}\mathbf{V}_{DD}=[R_{ij}]_{n\times n};\ R_{ij}=R(\|\mathbf{x}_i-\mathbf{x}_j\|);\ \mathbf{x}_i,\mathbf{x}_j\in D \tag{2.10}$$

For the rest of this thesis, we assume a Gaussian correlation function as in the following. (See Koehler and Owen 1996, for other forms of the correlation function and their merits.)

$$R(d) = e^{-\theta d^2} \tag{2.11}$$

where d is the Euclidian distance between two points, and θ is a problem dependent constant which is not usually known a priori. Indeed all families of correlation functions contain control parameters that adjust the range of correlation. Appropriate

values for θ, μ and σ are set according to the designer's experience or knowledge of the response function smoothness, or by comparison to similar response functions. A less subjective approach to estimate these parameters is to randomly select a small training sample of experiments, namely *S*, of size n_s and use maximum likelihood estimators, as in the following (see also, Mardia and Marshall 1984; Koehler and Own 1996)

$$\hat{\mu} = (\mathbf{J}^T \mathbf{R}_{SS}^{-1} \mathbf{y}_S) \Big/ (\mathbf{J}^T \mathbf{R}_{SS}^{-1} \mathbf{J}) \tag{2.12}$$

and

$$\hat{\sigma} = (1/n_s)(\mathbf{y}_D - \hat{\mu}\mathbf{J})^T \mathbf{R}_{SS}^{-1} (\mathbf{y}_D - \hat{\mu}\mathbf{J}) \tag{2.13}$$

where $\mathbf{J}=[1,\ldots, 1]^T$ is a vector of degree n_s. To calculate $\hat{\mu}$ and $\hat{\sigma}$ from the above equations, the value of θ is needed. The quantity θ can also be estimated by a maximum likelihood approach (Mardia and Marshall 1984; Koehler and Own 1996). In the test example of this chapter (see Section 2.3), we will assume a value of $\theta = 10$, which implies a rapidly decaying correlation with distance.

The above Bayesian approach is also used to formulate a DOE technique known as maximum entropy design (described later in Section 2.1.6). The following section provides a brief description of information entropy as a measure for the information provided by an experiment.

2.1.5 Entropy: Worth of an Experiment

The term entropy was first coined by Boltzmann. Later, Shannon (1948) introduced the abstract notion of information entropy that has since found many

applications in different fields. Lindley (1956) interpreted this new concept as the amount of information obtained by a Bayesian observation of a dependent parameter. In the following, we will give a brief discussion that relates information entropy to the worth of an experiment (in the context of approximation), following the notation introduced in the previous sections. (Similar discussions can be found in Shewry and Wynn, 1987; and Koehler and Owen 1996, among others.)

As mentioned in Section 2.1.3, the prior and posterior knowledge about the response function at points in $\overline{D}$ is represented by prior and posterior distribution: $\pi_{\overline{D}}$ and $\pi_{\overline{D} \mid D}$, respectively. We define the information content of the prior process in the following (e.g. see Lindley 1956):

$$I_0 = \int \pi_{\overline{D}}(y) \log(\pi_{\overline{D}}(y)) dy \tag{2.14}$$

or:

$$I_0 = E_{\pi_{\overline{D}}}[\log \pi_{\overline{D}}] \tag{2.15}$$

where $E_{\pi_{\overline{D}}}$ denotes the expected value operator with respect to $\pi_{\overline{D}}$ (prior process). The above definition was first given by Shannon (1948) and is an immediate result of "additivity postulate" as described next.

Suppose that the value of a response at $\overline{D}$ is observed in two stages. I_0 represents how much information is missing at the beginning (before any experimentation). In the first stage, we determine whether $y_{\overline{D}}$ belongs to an arbitrarily chosen subspace, say *A*, with the probability of *P*. This provides a certain amount of

information, say I_1. In the second stage, we determine the exact value of $y_{\overline{D}}$ which provides a certain amount of information, denoted by either I_2 or I_3: If $y_{\overline{D}}$ was determined to belong to A in the first stage the obtained amount of information in the second stage will be denoted by I_2. Otherwise, it is referred to as I_3. This means an expected information of $PI_2+(1-P)I_3$ in the second stage. The additivity postulate states that these information values must be additive, i.e.,

$$I_0=I_1+ PI_2+(1-P)I_3 \tag{2.16}$$

The significance of the Shannon's work comes from the observation that the *only* function that possesses the above-mentioned additivity property in general -- regardless of probability distributions -- is the function of equation 2.15 (multiplied by an arbitrary constant). In other words, the only measure of information by an experiment that satisfies additivity requirement is:

$$I(y_D)= E_{\pi_{\overline{D}\mid D}}[\log \pi_{\overline{D}|D}] \tag{2.17}$$

Therefore, the worth of an experiment that observes y_D is $I(y_D)$-I_0. This clearly depends on the observed value of y_D, so the average amount of retrieved information will be: $E_{\pi_D}[I(y_D)-I_0]$. The quantity $E_{\pi_D}[I(y_D)]$ is in fact Shannon's posterior entropy H with a negative sign, as follows:

$$H = -E_{\pi_D}[I(y_D)] = -E_{\pi_D}\{E_{\pi_{\overline{D}\mid D}}[\log \pi_{\overline{D}|D}]\} \tag{2.18}$$

or in short:

$$H = - E_U [\log \pi \ ] \tag{2.19}$$

where $U = D \cup \overline{D}$. Therefore (see also Koehler and Owen 1996; for another approach with slightly different assumptions to derive the following result):

Corollary 2.1.5-1: Informational worth of *D* can be measured by reduction of entropy (from prior process to posterior process), i.e., $H(\pi_{\overline{D}}) - H(\pi_{\overline{D}|D})$.

In other words, a design that minimizes the entropy of the posterior distributions, i.e., $H(\pi_{\overline{D}|D})$, or equivalently maximizes the entropy of prior process, i.e., $H(\pi_{\overline{D}})$, is the most informative one.

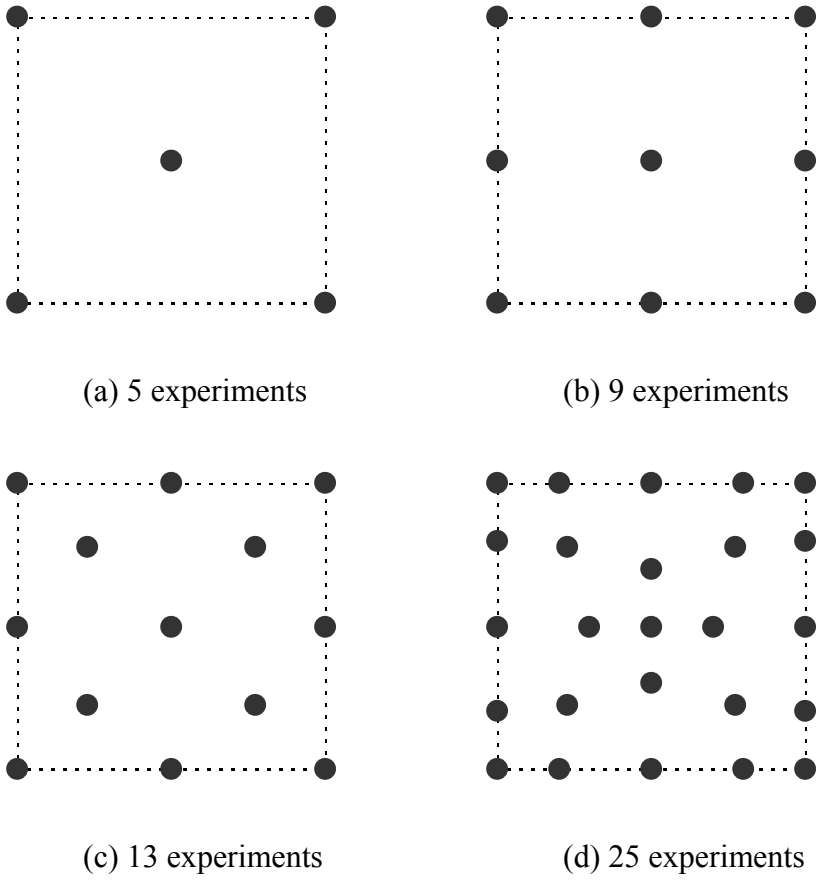
2.1.6 Maximum Entropy Design

Corollary 2.1.5-1 can be used directly to derive the most informative design of experiments. Under Gaussian assumption, however, it is easy to show that maximization of entropy is equivalent to maximization of the determinant of covariance matrix. That is, for a normal prior process, the maximum entropy is obtained if and only if the determinant, or '*det*', of the prior covariance matrix is maximized. (see Koehler and Owen, 1996 for a proof). This interesting observation constitutes an optimal DOE strategy.

Entropy Lemma: Maximum Entropy *n*-design is a subset of *U* that maximizes the determinant of the prior covariance matrix:

$$\underset{D_n}{\text{Maximize}} \ det\ (\mathbf{V}_{DD}) \tag{2.20}$$

After estimation of $\hat{\mu}$ and $\hat{\sigma}$, the maximum entropy design can be determined as follows: a subset of U of size n is selected such that the determinant of $\mathbf{R}_{DD}$ (equations 2.10 and 2.11) is maximum. According to the Entropy Lemma, this subset is the maximum entropy D_n and has the highest expected value of information retrieval from the unknown response function. All experiments in D_n are run, y_D is observed, and an approximation model is built accordingly. Figure 2.1 depicts maximum entropy designs in a two-dimensional design space, with different number of experiments. Note that maximum entropy design slightly emphasizes the boundaries. (The issue of the boundary effect will be fully addressed in Chapter 5.)



(a) 5 experiments (b) 9 experiments

(c) 13 experiments (d) 25 experiments

Figure 2.1: Single-stage maximum entropy design of experiments (2-dimensional)

2.1.7 Computational Complexity, and Multiple Hiker Algorithm

In order to find the entropy optimal design, as defined by Entropy Lemma, one can exhaustively calculate the determinant of all possible subsets of size n in U and compare these values -- an optimal D_n is a subset that maximizes determinant of the covariance matrix. However, this brute force implementation may become computationally infeasible in the sense that the number of possible subsets of U grows rapidly with the cardinality of U. Assume U is a grid of N nodes in the design space and a design of size n is desired. The number of possible subsets of size n in U is therefore $\binom{N}{n}$. Thus, unless the grid is coarse or the dimension of the input vector is low, an exhaustive search becomes very time consuming. Note that there is no need for calculating the actual (perhaps computationally expensive) response function values during this process, nevertheless, $\binom{N}{n}$ grows so rapidly that renders computing even the simplest surrogate models computationally infeasible. Indeed, the determinant of the covariance matrix can be calculated in only a few CPU cycles; however, the number of subsets grows so fast that the exhaustive search becomes computationally very costly or even infeasible. As such, Currin et al. (1991) suggest an algorithm that successively augments new experiments to the existing design. In that algorithm, they took advantage of the Shewry and Wynn's result (1987) for a one-point augmentation to an existing n-design, as discussed next.

Shewry and Wynn's (1987) Augmentation: If one desires to augment one more experiment to an existing set of experiments, the new experiment must be conducted at

a point, namely $\mathbf{x}_i \in \overline{D}_n$, with the largest variance of the posterior distribution. In other words, the best $\mathbf{x}_i$ to conduct a new experiment is the one at which $\sigma_{i|D_n}$ is maximum.

In the algorithm suggested by Currin et al. (1991), experiments are augmented one-by-one to the current set. A multiple-search is conducted over U to identify $\mathbf{x}_i \in \overline{D}_n$ with maximum $\sigma_{i|D_n}$, as follows: Consider n ascending 'hikers' that start from the current design, D_n, where the posterior variances are zero and move in the grid, one step at a time, in order to maximize the posterior variance. (This is very similar to a hill-climbing algorithm with multiple starting points.) Each of these hikers is allowed to move in $2p$ directions in the grid (2 directions along each edge of the grid), where p is the dimension of the input vector. In each step, we investigate all these possible movement directions for the hikers (at most $2pn$ evaluations). Each hiker then moves to the next node in the grid that has the highest variance. If two hikers meet at a node, they merge and continue as a single hiker. The algorithm continues until all hikers are at local maxima. Among them, the one with the maximum posterior variance is augmented to the current set of experiments. Obviously, this algorithm does not guarantee obtaining the global maximum, however, it dramatically reduces the computational burden of finding the optimum experiment. Also, it is likely that starting from multiple points helps explore the design space better and obtain the global optimum.

Note that although the above-mentioned algorithm is iterative, in the sense that experiments are augmented one-by-one to the design set, it does not take advantage of the response values (i.e., the response behavior) to update the prior distributions. That

is why we referred to this approach as a single-stage maximum entropy design approach, i.e., the response values to the entire set of experiments are evaluated once at the end of the process. In contrast, our new sequential approach (as will be introduced in the next section) evaluates the response function for all previous experiments throughout the process and according to this information updates the prior distributions for the next stage of design.

2.2 SEQUENTIAL MAXIMUM ENTROPY DESIGN (SMAXED)

A new sequential approach for DOE is introduced in this section. This approach is based on the observation that while the actual response function is not known a priori, as more experiments are designed and performed, more information is gained about the behavior of the response function. This information is then used to identify 'irregular' regions of the design space. Note that a good approximation model to the response function cannot be achieved without iteratively using this information towards the design of next set of experiments. Obviously, the stationary assumption is no longer applicable since the correlation of two points is not only a function of the relative distance but also dependent upon the characteristics of the regions in which the experiments are conducted and how well the function behaves in those regions. This information should be updated sequentially based on the data gathered from previous experiments, as explained in the following subsections.

2.2.1 Sequential Design of Experiments

As mentioned before, in a single-stage maximum entropy design, responses from a set of experiments are not used to enhance the design of next set of experiments. In fact, the maximum entropy design does not depend on the response y

at all. A maximum entropy design in a given design space will be always the same regardless of the behavior of the function which is being approximated. (For example, the single-stage design of Figure 2.1 does not depend on the shape of the response function.) Figure 2.2, for example, demonstrates a deterministic response function with a one-dimensional input in the interval [0,1]. (See the formulation of this response function in Section 2.3.)

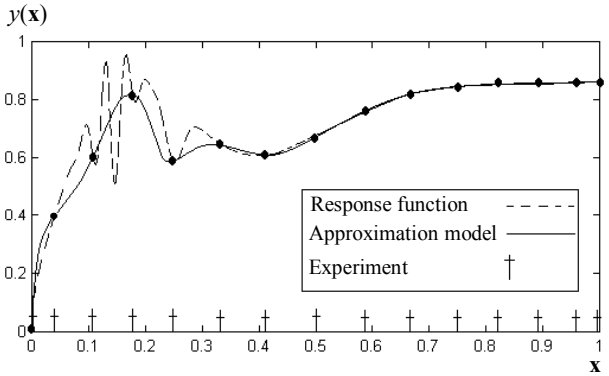


Figure 2.2: A design of 15 experiments for a deterministic response function and the resulting approximation model

In Figure 2.2, the daggers on the x-axis represent a maximum-entropy 15-design. After evaluating y_D, which is shown with bullets on the response curve, the interpolating surrogate model of Equation 2.9 is constructed (solid line in Figure 2.2). As shown in this figure, the experiments of D_n are distributed symmetrically along the design interval with an emphasis on the corners. A single-stage maximum entropy design is not problem specific and does not depend on the response function itself. Therefore, there are exactly 7 experiments in the interval [0,0.5) as well as the interval (0.5,1]. However, by looking at the actual response function, one notices that y(x) is not behaving the same everywhere in the design space. The left half interval [0,0.5) consists of many local optima -- located tightly together (i.e., has an irregular region), while a large portion of the domain in the right half of the interval, i.e., (0.5,1], is

monotonic. Thus, to approximate the behavior of the response function in the irregular region of the domain, more experiments must be conducted in the left half of the interval. In contrast, not that many experiments are needed in the less irregular region (i.e., the right half) of the domain to obtain a desired accuracy. A single-stage (i.e., not sequential) maximization of the entropy completely ignores this fact. As such, the approximation model deviates significantly from the actual response function in the irregular region of the domain, while there are redundant experiments in the less irregular portion.

In the rest of this section, we present the new sequential (multi-stage) maximum entropy design approach that takes advantage of the response values at each stage to design the next set of experiments.

2.2.2 Assumptions

We assume that at the beginning of the experimentation process there is essentially no specific information about the unknown response function. Therefore, the stationary assumption of Section 2.1.4 holds. We define an m-block, denoted as B_m, as a set of m experiments whose elements (i.e., the individual experiments) are designed and performed in a single stage. In other words, all of the m experiments in the m-block are designed by maximization of entropy at the beginning, but once the block is designed, all of its experiments are conducted one after another -- without using the response value of one experiment to update the prior information of the next experiments. Accordingly, a single-stage maximum entropy design of m experiments is basically a single m-block of experiments. As mentioned before, a single block of experiments may not be the best design since it does not take advantage of the

information gained in the process to place the next experiments in the regions where they are more informative. Basically, the geometry of a single block of experiments is totally independent of the response function that is being investigated. Given the design space, number of experiments and prior distributions, a maximum entropy block can be determined regardless of the response function. In this section, however, we present an algorithm that sequentially constructs several blocks of experiments for a given response function, y, defined on U. After each block is designed, the corresponding experiments are run, the values of the response function are assessed and used to determine those regions of the design space in which new experiments are expected to be more informative.

From Equation 2.10, the covariance of prior distributions at two points $\mathbf{x}_i$ and $\mathbf{x}_j$ is a decreasing function of the distance between two points;

$$\mathrm{Cov}(Y_i, Y_j) = \sigma_{ij} = \sigma^2 R(\|\mathbf{x}_i - \mathbf{x}_j\|) \quad \mathbf{x}_i, \mathbf{x}_j \in D \tag{2.21}$$

and

$$\mathrm{Var}(Y_i) = \sigma_{ii} = \sigma^2 \tag{2.22}$$

Equation 2.22 can be interpreted as follows: conducting an experiment at $\mathbf{x}_i$ transmits some information about the response at $\mathbf{x}_j$. For example, if the expected value of the prior distribution at $\mathbf{x}_i$ turns out to be an underestimation after observing y_i, i.e., $y_i > E(Y_i)$, because of the positive correlation it is more likely that $E(Y_j)$ is also an underestimation for the response function at an adjacent point $\mathbf{x}_j$ (i.e., based only on the information provided by the experiment conducted at $\mathbf{x}_i$). This can be thought of as the influence of conducting an experiment at $\mathbf{x}_i$ on its neighboring points. However, this influence decreases with the distance from $\mathbf{x}_i$ due to the decreasing correlation

function. Points that are located in the vicinity of $\mathbf{x}_i$ are highly correlated with Y_i, and therefore observing y_i has a significant impact on the their posterior distributions. On the other hand, points that are located far from $\mathbf{x}_i$ are uncorrelated or loosely correlated and thus not influenced by $\mathbf{x}_i$. This is because $R(d)$ approaches zero as d approaches infinity.

Now consider two experiments A and B, as shown in Figure 2.3.

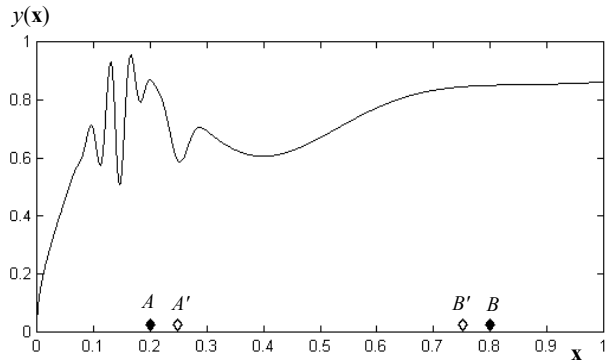


Figure 2.3: The uncertainty grows faster with distance where the response function is irregular

Both of these experiments influence their neighborhoods. If the response function is known a priori, one can say that experiment A is located in a neighborhood where $y(x)$ is multi-modal (or irregular) with many close local optima. The influence of the experiment A on point A' is therefore relatively weak. In other words, conducting an experiment at A and observing y_A does not say much about $y_{A'}$. On the other hand, experiment B conveys much more information about B', since B is located in a less irregular region of the response function. So, the correlation decays slower with distance (or equivalently uncertainty grows slower) in this region. By looking at Figure 2.3, one could intuitively say that more experiments must be conducted in the multi-modal region to enable a more accurate modeling of the response function. In contrast,

not that many experiments are needed in the less irregular region. Clearly, irregularity should be quantified and formally incorporated into the design criterion, as discussed later in our approach. However, we first present a new theorem that is used as a basis in the SMAXED approach.

2.2.3 SMAXED Theorem

Suppose a set of *n* experiments has already been designed according to Entropy Lemma, and D_n is obtained. In the following theorem, we prove that addition of a new experiment that has low correlation with previous experiments is in fact very informative (i.e., yields greater reduction in posterior entropy).

SMAXED Theorem: Suppose D_n has already been designed and evaluated. Assuming Gaussian priors, conducting a new experiment at $\mathbf{x}_i \in \overline{D}_n$ is more informative than conducting another one at $\mathbf{x}_j \in \overline{D}_n$, if $\sigma_{ik} < \sigma_{jk}$; $\forall \mathbf{x}_k \in D_n$ and $\sigma_{ii}=\sigma_{jj}=\sigma^2$.

Proof: From Equation 2.8, one can easily verify that for Gaussian priors:

$$\sigma_{ii|D} = \sigma_{ii} - \mathbf{V}_{iD}^T \mathbf{V}_{DD}^{-1} \mathbf{V}_{iD} \tag{2.23}$$

and

$$\sigma_{jj|D} = \sigma_{jj} - \mathbf{V}_{jD}^T \mathbf{V}_{DD}^{-1} \mathbf{V}_{jD} \tag{2.24}$$

where $\mathbf{V}_{iD}=[\sigma_{i1}, \sigma_{i2},\ldots, \sigma_{in}]^T$ is the covariance vector of $\mathbf{x}_i$ and D_n. Moreover, $\sigma_{ik} < \sigma_{jk}$; $\forall \mathbf{x}_k \in D_n$, and $\sigma_{ii}=\sigma_{jj}=\sigma^2$. Since σ_{ij}'s are all positive numbers, from Equations 2.23 and 2.24 we obtain: $\sigma_{ii|D_n} > \sigma_{jj|D_n}$. Therefore, using Shewry and Wynn's

augmentation result (recall Section 2.1.7), we conclude that conducting an experiment at $\mathbf{x}_i$ yields a greater reduction in posterior entropy than $\mathbf{x}_j$ and the theorem follows. □

An immediate result from the above theorem is that addition of a new experiment that is located far from already existing experiments is very informative (i.e., due to small covariance between the new experiment and already existing ones). In fact, the maximum entropy criterion tends to maximize the distances among experiments and place new experiments in remote regions where the correlation with the existing experiments is minimum (and in some sense, the uncertainty is maximum).

SMAXED Theorem also provides a way to account for irregularities in the response function. This is in fact the driving force behind the SMAXED approach. According to this theorem, assigning a weaker correlation in the irregular regions of the design space increases the informational worth of an experiment conducted in those regions. A maximum entropy design is basically a design that maximizes the information content of a set of experiments and thus places more experiments in the regions where the correlation decays faster. This follows our intuitive observation that there must be more experiments conducted in the irregular portion of the domain in Figure 2.2, while not that many are needed in the less irregular part. In reality, however, we do not know the behavior of the actual response function upfront. Indeed, there is no information about the response function at the beginning of the process. This is the main motive for introducing the new sequential algorithm.

The SMAXED approach, to be presented next, recursively designs and performs blocks of experiments in such a way that the response values obtained from each block are used to update the covariance matrix before the next block is designed.

2.2.4 SMAXED Approach

Suppose there are enough time and computational resources to conduct only n experiments. In SMAXED, we design and perform a total of n experiments in several blocks (for instance in k blocks of size m.) The overall n-design then will be:

$$D_n = B_m^1 \cup ... \cup B_m^i \cup ... \cup B_m^k ; \qquad n = km \tag{2.25}$$

where B_m^i is the i-th m-block, designed and performed after all previous (i-1) blocks are designed and performed.

Suppose that prior to the design of the first m-block, i.e., B_m^1, no information is available about the response function and as reasoned before the stationary assumption is applicable for the prior distribution. (μ and σ can be estimated using the maximum likelihood estimators, Equations 2.12 and 2.13.) After performing all experiments in B_m^1, an intermediate surrogate model is built based on these responses, y_m^1. The intermediate surrogate model provides a preliminary insight into the behavior of the response function.

Now that an initial understanding of the behavior of the unknown response function is obtained, one would like to take advantage of this knowledge in the design of the next set of experiments. One way to incorporate this knowledge is to use SMAXED Theorem (Section 2.2.5) and penalize (i.e., decrease) the correlations in those regions of the design space where the function is irregular. In other words, our goal is to identify those **x**'s in U for which the correlation decays faster with distance because of the presence of irregularity in the response function. The definition of function irregularity and how to update the covariance matrix accordingly is therefore

subjective in nature. Note that the correlation function itself is subjectively defined and the best choice is generally never known before solving the problem. In the following, a strategy is presented that is based on the multi-modality of the initial (or other intermediate) surrogate model. There might be other aspects of irregularity as well, such as discontinuity, singularity, etc. The proposed approach focuses only on the multi-modality of the response function. Note that, as demonstrated in the example of Figure 2.2, multi-modality in certain regions of the domain is a very important property that prompts for more experiments in those regions. As will become clear later in this chapter, if the function contains no local optima in U (excluding the boundaries of the domain), the proposed sequential approach results in the same design as a single-stage maximum entropy augmentation of experiments.

In SMAXED, prior to the design of a block of experiments, we obtain all (or as many as possible) local optima of the previous intermediate surrogate model (which is constructed based on the previous blocks of experiments). The computational complexity of obtaining these optima depends on the chosen surrogate model (e.g., interpolation, polynomial, etc.). Note that the approximation model is always considerably less expensive to compute compared to the original response function, and thus the optimization techniques that require many function calls can be computationally feasible to apply. The implementation and computational complexity issues are discussed in Section 2.2.6. That section also includes an algorithm to find as many local optima as possible without considerable computational effort.

We refer to a node that is not located on the boundary of the grid and its approximated response function value is strictly higher/lower than all immediate

neighboring nodes as an 'interior local optimum'. We also define a 'flat optimum set', denoted by $S \subset U$, as a connected set of two or more nodes with the same approximated response function value, strictly greater (or less) than all other immediate neighbors that are not included in *S*. (The term 'connected set' refers to a set, each of whose elements is an immediate neighbor of at least one other element in the set.) Clearly, the nodes of a flat optimum set are weakly optimum, because there exists at least one immediate neighbor in the flat optimum set, with the same approximated response function value. In general, one node is selected randomly from a flat optimum set as a representative of that set. Finally, set *P* is defined as the union of all interior local optima, including the representatives from all flat optimum sets.

In the following, we use *P* to quantify the irregularity in the neighborhood of a point, namely $\mathbf{x}_i \in U$.

Definition 2.2.4-1: We define the Characteristic Certainty Width (CCW) of a point $\mathbf{x}_i \in U$, denoted by $L(\mathbf{x}_i)$ or L_i, as the length of the diagonal of the smallest hyper-rectangle in the design space that encloses $\mathbf{x}_i$ and whose two opposite vertices are any two local optima in *P*. Moreover, we define L_0 as the length of the diagonal of the design space, which is a hyper-rectangle itself. It is then assumed that: $L(\mathbf{x}_i) = L_0$, if no such rectangle can be found that encloses $\mathbf{x}_i$.

The intuitive interpretation of this definition is shown in Figure 2.4. The set of optima, *P*, is marked by bullets in a 2-dimensional design space. Clearly, $L(\mathbf{A})=L(\mathbf{E})=L_0$; $L(\mathbf{B})=L_1$; $L(\mathbf{C})=L_2$; $L(\mathbf{D})=L_3$. For a one-dimensional design space, CCW of a point is basically the distance of the two optima that bracket that point.

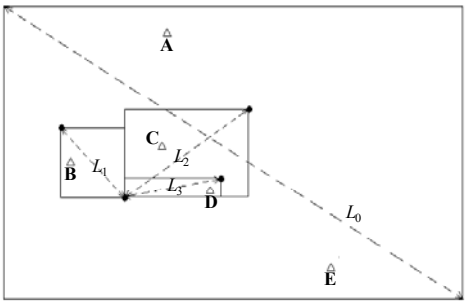


Figure 2.4 Characteristic certainty width

We use L as a measure of regularity in the behavior of the function. A large L implies that there is a wide unimodal region. As mentioned earlier, fewer experiments are needed for such a region in the design space, as opposed to an irregular region with many tightly located local optima. In other words, the correlation tapers off slower with distance in a region with a larger L. This corresponds to a larger covariance. Therefore, in the SMAXED approach we use (L_i/L_0) as a correcting factor that incorporates irregularities of the design space and updates the covariance of two points accordingly, as follows: (Compare to Equation 2.10.)

$$\begin{aligned}\sigma'_{ij} &= \sigma_{ij}\left(\frac{L_i}{L_0}\right)\left(\frac{L_j}{L_0}\right) \\ &= \left[\sigma\left(\frac{L_i}{L_0}\right)\right]\left[\sigma\left(\frac{L_j}{L_0}\right)\right] R\left(\left\|\mathbf{x}_i - \mathbf{x}_j\right\|\right) \qquad ; i \neq j \end{aligned} \tag{2.26}$$

where $R(.)$ is obtained from Equation 2.11. Comparing Equations 2.10 and 2.26, one notices that the covariance of two points is corrected by a factor of $[L(\mathbf{x}_i)/L_0][L(\mathbf{x}_j)/L_0]$. Correlation of those experiments that are located in irregular regions (i.e., small L) is reduced by a factor proportional to their irregularities. This in turn imposes a faster rate of uncertainty growth with distance in those regions (i.e., covariance of two points

tapers off faster). Hence, according to the SMAXED Theorem, updating the covariance matrix using the above equations and maximization of entropy automatically places more experiments in the irregular regions as the next block of experiments, because of their higher informational worth (greater entropy reduction). In contrast, in less irregular regions where the local optima are located far from each other, the covariance decays slowly with distance (or uncertainty grows slowly). The quantity L/L_0 is larger in those regions and thus covariances are not considerably reduced, if reduced at all (recall $L_i=L_0$ if $\mathbf{x}_i$ is not bracketed between two optima in P). Note that if we find none or only one local optimum inside the design space we have $L=L_0$, and therefore this strategy basically results in the same design as obtained from single-stage maximum entropy approach (i.e., augmenting new experiments without using intermediate approximation models). Indeed, a uni-modal response function with one or no interior local optimum is considered regular everywhere in the design space and therefore the covariances are not reduced at all; i.e., $(L/L_0)=1$.

2.2.5 SMAXED: Step-by-Step Description

A step-by-step description of SMAXED follows.

Step 1 – Since there is no initial information about the response function, the first block of experiments, B_m^1, is designed with the stationary assumption and according to Entropy Lemma. The initial $m \times m$ covariance matrix is constructed using Equations 2.21 and 2.22. The standard deviation, σ, is estimated via the maximum likelihood estimator of Equations 2.12 and 2.13.

Step 2 – In the *k*-th iteration we have: $D_{k\times m} = B_m^1 \cup ... \cup B_m^k$. All new experiments are observed, i.e., $y_{k\times m}$.

Step 3 – An intermediate approximation model is constructed based on the responses $y_{k\times m}$ from all previous experiments (using kriging).

Step 4 – The approximation model is optimized and all (or as many as possible) local optima are obtained to form set *P*.

Step 5 – The characteristic certainty width, $L(\mathbf{x}_i)$, is calculated for all $\mathbf{x}_i \in \overline{D}_{k\times m}$ (i.e., complement of $D_{k\times m}$.)

Step 6 – A *q*x*q* covariance matrix is constructed where $q=m(k+1)$. The first *mk*x*mk* rows and columns of the matrix correspond to the experiments that were already designed and performed in the previous iterations, i.e., $D_{k\times m}$. These entries are updated according to Equation 2.26 using the new CCW's. There are *m* rows and *m* columns remaining in this matrix that have to be determined, as discussed next.

Step 7 – The remaining *m*x*m* rows and columns of the matrix correspond to the next *m*-block, B_m^{k+1}. The elements of this block are selected from $\overline{D}_{k\times m}$ such that the determinant of the *q*x*q* matrix is maximized. Again, these new entries follow Equations 2.22 and 2.26. The result of this maximization is a set of *m* new points that marks *m* new experiments, i.e., B_m^{k+1}.

Step 8 – Go to step 2 and continue until a total of $n = k \times m$ experiments are designed and performed. The final approximation model is constructed based on these *n* experiments. Should one decides to conduct more experiments (i.e., by assigning more

computational resources), SMAXED can be run again to design new experiments, starting from the existing approximation model.

2.2.6 Revised Multiple Hiker Algorithm

There are basically two main sources of computational burden in SMAXED (i.e., other than the evaluations of the potentially expensive response function):

(i) Step 1 and Step 7; in which a block of experiments are chosen such that it maximizes the determinant of the covariance matrix. In the test example of the next section, this is done exhaustively by searching the set of all subsets of size m in U. However, as mentioned in Section 2.1.7, exhaustive search is not possible for large U's. Therefore, the multiple-hiker algorithm of Section 2.1.7 can be used to determine the optimum block of experiments in Steps 1 and 7.

(ii) Step 4; in which the local optima of the approximation model is obtained. The straightforward approach is to exhaustively search the grid of U for those nodes that are local minima (and/or maxima). Suppose U is p-dimensional and contains N points, thus each point in U has at most $2p$ neighbors (points on the boundary have fewer neighbors). To find all local optima, one should examine all points in U and compare the values of the approximation model. Although an approximation model is not costly to compute, for high-dimensional design spaces with fine grids, an exhaustive search may become very time-consuming or even computationally prohibitive. Therefore, to reduce the computational effort, we suggest an algorithm very similar to the one suggested by Currin et al. (1991). In our suggested algorithm, we deploy n ascending-hikers to locate local maxima as well as n descending-hikers to locate local minima

(i.e., a total of 2*n* hikers). The hikers start from the current set of *n* experiments. For instance, an ascending-hiker continuously compares the approximated response function values of immediate neighboring nodes with that of the current node and moves to a neighboring node that provides the higher elevation. If the current node has the highest elevation among all its neighboring nodes the hiker stops and the current node is declared as a strict local maximum and is added to *P*. If one of the neighboring nodes has the same elevation and the rest are strictly lower than the current node, there is a possibility that the hiker has reached a flat optimum. The hiker marks that node and continues the search by going to the next node with the same elevation. The hiker stops if no further improvement is possible and a representative from the current flat optimum set is added to *P*. The algorithm stops when all ascending- and descending-hikers stop. Note that although this algorithm significantly reduces the computational cost of the process, it may not necessarily obtain all optima, skipping some elements of *P*. As mentioned before, a full set of optima is not guaranteed in general without an exhaustive search of *U*, which is not usually practical. However, obtaining as many local optima as possible provides a better understanding of the irregularities in the response. If fewer than two local optima are located, the application of SMAXED results in the same design as that of a single-stage maximum entropy approach. As more and more local optima are detected, SMAXED identifies the irregular regions better, which in turn increases the accuracy of the approximation. Depending on the size of the design space, one can randomly add more starting points for the hikers to improve the success of the algorithm. (Again, unlike the actual response function, the

approximation model is not costly to compute; hence addition of new hikers does not significantly increase the overall run time of the algorithm.)

A simple numerical test example is used in the next section to demonstrate the performance of SMAXED. In Chapters 6 and 7, complicated engineering examples are used to demonstrate and compare the accuracy of the surrogate models obtained from SMAXED.

2.3 DEMONSTRATION EXAMPLE

The formulation of the response function of Figure 2.1.1-1 is given below:

$$\begin{aligned} y(\mathrm{x}) &= (1-e^{-2\sqrt{\mathrm{x}}}) + 6\mathrm{x}e^{-7\mathrm{x}}\sin(10\mathrm{x}) - 0.2\mathrm{e}^{-2000(\mathrm{x}-0.25)^2} \\ &+ 60\min(0, |\mathrm{x}-0.14| - 0.08)^2[\ln(\mathrm{x}+0.2) + 1.5\sin^2(85\mathrm{x})] \end{aligned} \tag{2.27}$$

In a single-stage maximum entropy design, a set of n experiments (in U) is selected such that the determinant of $\mathbf{V}_{DD}$ becomes maximum. Assuming that there is just sufficient computational power (or time) to perform 15 response function evaluations, we can exhaustively search the design space to identify the maximum entropy 15-design, as shown in Figure 2.2. These experiments are dispersed symmetrically in the design space regardless of the behavior of the response function in different regions. After observation of the response values, an interpolating surrogate model is constructed, as shown in Figure 1.

To demonstrate the application of SMAXED, we sequentially design a total of 15 experiments as 3 blocks of size 5 experiments. This is described next:

***Step 1* (*1st iteration*)** – Similar to a single-stage design, the first block of 5 experiments is chosen exhaustively from all possible subsets of size 5 in U such that the

determinant of the covariance matrix is maximum. The small arrows on the x-axis in Figure 2.5 illustrate B_5^1 for this response function. Note that one could alternatively use the multiple-hiker algorithm for one-point augmentation of experiments, and obtain an optimum 5-design. This latter approach results in a slightly different, yet good enough design while it significantly reduces the computational burden.

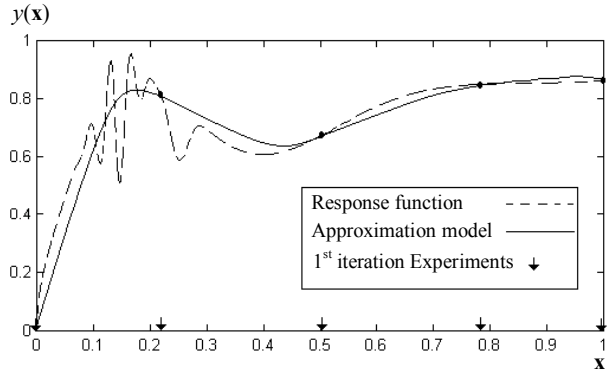


Figure 2.5 The initial block of 5 experiments and the resulting intermediate approximation model

Steps 2, 3, and 4 (1st iteration) – The surrogate model of the current design is shown in Figure 2.5. The intermediate surrogate model has three local optima inside the design space: P_1={0.17; 0.44; 0.96}.

Step 5 (1st iteration) – L(x) is computed to be:

$$L(\mathrm{x}) = \begin{cases} 0.44 - 0.17 = 0.27 & 0.17 < \mathrm{x} < 0.44 \\ 0.96 - 0.44 = 0.52 & 0.44 < \mathrm{x} < 0.96 \\ 1 & \text{otherwise} \end{cases} \tag{2.28}$$

Step 6 (1st iteration) – A 10x10 covariance matrix is constructed. The first 5 rows and columns correspond to the covariance of 5 experiments designed in the first block (small arrows in Figure 2.5).

Step 7 (1st iteration) – The second 5-block of experiments is designed such that the determinant of the 10x10 matrix is maximized. Note that the entries of this matrix are obtained from Equation 2.26, and using the CCW of Equation 2.28. The optimum second block is found by searching the set of subsets of size 5 in $\overline{B}_5^1$ exhaustively, and comparing their determinants. (As in Step 1, one could alternatively use a one-point augmentation approach to reduce the computational burden.) This yields B_5^2 which is a 5-block that results in the highest determinant of the 10x10 covariance matrix. (The first 5 rows and columns are already known, i.e., B_5^1.) Figure 2.6 shows the first and second 5-block of experiments (a total of 10 experiments). This figure clearly shows that more experiments are designed in the irregular region (between the first two optima in P_1 where L is small).

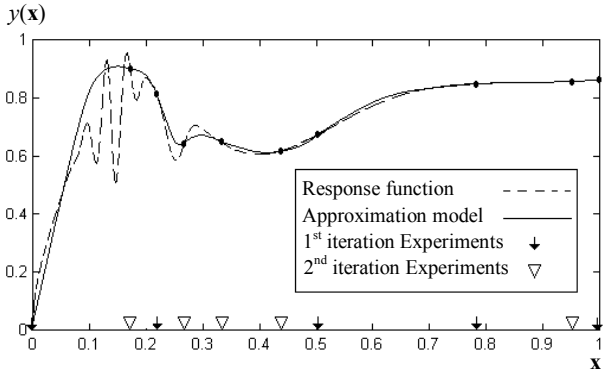


Figure 2.6: The first iteration of SMAXED and the corresponding approximation model

Step 8 (1st iteration) – There are a total of 10 experiments. So we repeat Steps 2 through 7 to design 5 more experiments, as in the following.

Steps 2 and 3 (2nd iteration) – As before, the new experiments, i.e., B_5^2, are evaluated and an interpolating surrogate model is constructed (Figure 2.6). The new surrogate model is more accurate in the irregular region.

Steps 4 and 5 (2nd iteration) – Optimization of this surrogate model yields: P_2={0.15; 0.26; 0.30; 0.41}. Hence, CCW(x) is:

$$L(\mathrm{x}) = \begin{cases} 0.11 & 0.15 < \mathrm{x} < 0.26 \\ 0.04 & 0.26 < \mathrm{x} < 0.30 \\ 0.11 & 0.30 < \mathrm{x} < 0.41 \\ 1 & \text{otherwise} \end{cases} \tag{2.29}$$

Steps 6 and 7 (2nd iteration) – As before, the new 5-block is determined such that it maximizes the determinant of a 15x15 covariance matrix whose first 10 rows and columns correspond to the experiments designed in the first iteration. The entries of the matrix are evaluated from Equation 2.26 using the CCW of Equation 2.29. Figure 2.7 demonstrates the new 5-block, i.e., B_5^3, along with the previous blocks, B_5^1 and B_5^2, and the new approximation model.

Step 8 (2nd iteration) – we have obtained a total of 15 experiments. The algorithm stops.

Note that both designs in Figures 2.2 and 2.7 contain 15 experiments. However, the approximation model from SMAXED shows a dramatic improvement (see Figure 2.7) as compared to that of a single-stage maximum entropy design (see Figure 2.2). This is because of the tendency of SMAXED approach to distribute experiments in the design space with an emphasis on the irregular regions of the domain. The accuracy of the approximation can be improved even further should one decides to continue SMAXED. As more experiments are conducted, the approximation model approaches

the actual response function, revealing the irregular regions in which because of the faster growth of uncertainty more experiments are needed to provide a more accurate approximation model.

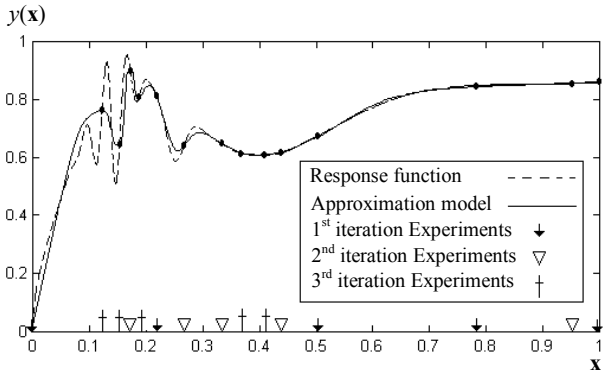


Figure 2.7: The second iteration of SMAXED and the corresponding approximation model (15 experiments total)

2.4 CONCLUDING REMARKS

In this chapter, a new sequential maximum-entropy design strategy is introduced. The new approach, referred to as SMAXED, takes advantage of the information gained during previous experiments to recursively update prior distributions of next sets of experiments. In the numerical example of this chapter, SMAXED performed comparatively well, identifying the irregular regions of the design space and spreading new experiments accordingly. Indeed, SMAXED placed a larger fraction of the available experiments in the irregular region of the design space where more experiments are needed to provide a better representation of the actual response function. Note that although the SMAXED approach can help reduce the number of response function evaluations, it may utilize CPU time for internal computations within the algorithm (i.e., optimization of the intermediate approximation model in Step 4, as well as maximization of entropy in Steps 1 and 7.)

Overall, if the response function is very time-consuming to compute, this extra effort to optimally place expensive experiments in the design space should provide a dramatic improvement in the accuracy of the approximation model.

Once computationally expensive response functions are replaced with surrogate models, MOGAs can be used to find optimal solutions. In the next chapter, Chapter 3, a new MOGA is developed that aims at obtaining a Pareto solution set with maximum possible diversity along the Pareto frontier. The new MOGA is based on simulating Maxwellian systems and is shown to have a tendency to increase diversity of solutions.

CHAPTER 3

MULTI-OBJECTIVE GENETIC ALGORITHM WITH CONCEPTS FROM STATISTICAL THERMODYNAMICS

Multi-Objective Optimization (MOO) algorithms are essentially search techniques that aim at achieving a certain set of goals (see Haimes et al. 1971; Geoffrion et al. 1972; and Steuer 1986, for a comprehensive literature survey of MOO techniques). Among different MOO techniques, Multi-Objective Genetic Algorithms (MOGAs) have received significant attention recently. MOGAs are based on the principle of natural selection and survival of the fittest. However, because of the very nature of MOGAs, the obtained solution sets from such algorithms tend to cluster in some regions of the Pareto frontier -- leaving the rest of the Pareto underrepresented. Therefore in this chapter, a new MOGA is introduced that is based on formalisms from statistical thermodynamics and aims at maximizing H (i.e. Boltzmann's entropy) in a MOGA population.

This chapter is organized as follows: Section 3.1 provides a brief introduction to multi-objective genetic algorithms, current improvement techniques in the literature, and the analogy between GAs and statistical model of ideal gas. Section 3.2 presents a new algorithm, i.e., Thermodynamical MOGA or T-MOGA, that takes advantage of this analogy to improve the quality of solutions. Finally, Section 3.3 presents a simple

test case to demonstrate the performance of the proposed approach. A case study and more examples can be found in Chapters 6 and 7.

3.1 BACKGROUND AND PREVIOUS WORK

The concept of GA was first introduced by Holland (1975) and soon received significant attention in various domains of search and optimization research (see Appendix II for a detailed description). GAs have several advantages over other classic optimization techniques. For example, GAs can handle mixed discrete-continuous optimization problems in a simple manner. Moreover, the concept of GA can be easily extended to address multi-objective optimization problems (MOGAs). Indeed, many researchers (e.g. Valenzuela-Rendon and Uresti-Charre 1997) suggest that multi-objective optimization is a problem area where GAs perform much better than blind-search techniques. Although this statement is subjective and many conditions may have to hold in order to guarantee a fast rate of convergence (Wolpert and Macready 1997), nevertheless, GAs are becoming quite popular in real-world applications. (For a more detailed description of GA and MOGA refer to Appendix II.)

3.1.1 Multi-Objective Genetic Algorithms (MOGAs)

In general, multi-objective evolutionary algorithms can be categorized into three main approaches: weighting approach, population-based non-Pareto approach and population-based Pareto approach (Wu, 2001). In the first approach, all objectives are combined into one using a set of weights and then a single-objective GA is applied to solve the problem. Despite the simplicity of this approach, its applications are limited since the appropriate objective-weights are not usually known prior to the

optimization process. Moreover, the weighted approach does not necessarily yield the entire Pareto-optimal solutions for non-convex problems. In the second approach, a population of solutions is evolved simultaneously throughout the process to create a set of Pareto optimal solutions. Vector Evaluated GA (VEGA) developed by Schaffer (1985) is the most well-known technique within this category. A VEGA disjoints the population into a set of sub-populations via objective-proportional selection. The sub-populations are then shuffled and the GA operators are applied. The main drawback of this approach as reported by Srinivas and Deb (1994) is the bias of the solutions to some sub-regions of the Pareto frontier. Moreover, since the fitness function in a VEGA is a linear combination of the objectives, few Pareto-optimal solutions can be obtained in the case of non-convex search spaces (Coello Coello, 1999). Goldberg (1989) suggested another approach: assigning a Pareto fitness to the individuals in a MOGA according to their *dominant number,* i.e., the number of other solution points that are dominated by a particular solution (see Appendix II). This approach soon became widely accepted and many MOGAs were developed based on this idea (Fonseca and Fleming 1993, Narayanan and Azarm 1999). In fact, Fonseca and Fleming (1993) implemented one of the first MOGAs that utilized Pareto-based fitness assignment along with a fitness sharing technique to prevent the tendency of the population to cluster in some sub-regions of the Pareto front. Another implementation of a MOGA using Pareto-based fitness was presented by Srinivas and Deb (1994) who created the Non-dominated Sorting Genetic Algorithm (NSGA). NSGA calculates the fitness of the solutions by determining the current non-dominated set, assigning them the best fitness, virtually removing them from the solution set and determining the next

subset of non-dominated solutions. Srinivas and Deb (1994) also introduced a fitness-sharing approach, using a pre-defined sharing parameter and penalizing the solutions within this sharing parameter. Since then many other successful implementations of MOGAs are reported in the literature (Ishibuchi and Murata 1996; Cunha, Oliviera and Covas 1997; Valenzuela-rendon and Uresti-Charre 1997; Fonseca and Fleming 1998). For a comprehensive literature survey of MOGA methodology and applications see Foneseca and Fleming (1995); Tamaki et al. (1996); Horn (1997); Coello Coello (1999); Zitzler (1999); Deb (2001) and Van Veldhuizen and Lamont (2000).

Many researchers have attempted to develop MOGAs that yield better solution sets in one way or another. The next section is devoted to a brief review of some of the most popular MOGA improvement techniques.

3.1.2 MOGA Improvement Techniques

Most of the recently proposed MOGAs claim to perform better in one aspect or another. For example, Veldhuizen and Lamont (1998) and Rudolph (1998) have addressed the issue of fast convergence to the optimal Pareto frontier. In addition, some variations of MOGA are developed to increase the diversity of the solutions in the population, via niche punishment of clustered individuals (Obayashi et al. 1998). In fact, most of the techniques in the literature use the concept of fitness sharing and niche punishment to dissolve clusters of solution points. Niche punishment spreads the solutions more uniformly over the Pareto frontier, which in turn increases the diversity of the solution set. (This is usually very desirable for the decision maker, since a diverse solution set provides a wide range of alternatives among which one can be selected.) One such algorithm -- introduced by Horn et al. (1994) -- is a Pareto-based

approach, referred to as the Niched Pareto Genetic Algorithm (NPGA). As in previous algorithms, the fitness sharing of this approach is based on the niche count. However, instead of using non-dominated sorting, a *Pareto domination tournament* strategy is introduced. The method basically compares ten (or more) solution points (instead of two) simultaneously, and rank orders them and selects individuals accordingly. This technique imposes some noise in the solution set during the process, yet this noise diminishes as a larger population size is selected.

In addition to the above mentioned algorithms, there are some other variations of MOGA in the literature that use a combination of niche-count and some other techniques to maximize the diversity of the solution set as much as possible (Narayanan and Azarm, 1999).

Other than fitness-based approaches, several recent techniques have been developed to improve the coverage and uniformity of the solutions via heuristics (e.g., Camponogara and Talukdar, 1997; Reynolds, 2000). In particular, these algorithms attempt to generate additional solutions to artificially fill in underrepresented areas in a Pareto solution set (i.e., along a Pareto frontier). The basic concept in these techniques is to randomly select and project some candidate points near the edge of a gap to fill in the interior or exterior voids in a solution set. However,

- Heuristic projection of solution points to artificially fill the gaps does not guarantee an improvement in the solution set quality (because for instance it may create a gap in another region).
- Even if a uniform distribution is obtained at a certain time during MOGA, it is not guaranteed to last. In other words, this uniformity may be transitional, and

if MOGA continues to project solutions and move them in random directions, the population may start to cluster again.

As such, in this chapter we present a new MOGA -- referred to as Thermodynamical MOGA or T-MOGA. T-MOGA moves solution points according to a well-known pattern known as Maxwellian distribution. As discussed in Section 3.1.3 and 3.1.4, Maxwellian systems are guaranteed to achieve an equilibrium state, that is: 1) uniform density; and 2) steady state (not transitional). Moreover, Maxwellian systems are the *only* systems that are guaranteed to have this property.

In the next section, we discuss the analogies between MOGA and Maxwellian systems. This analogy will be used later in Section 3.2 to develop T-MOGA. Because of the above mentioned-property of Maxwellian systems, T-MOGA (unlike previous MOGAs) should evolve to achieve a *steady-state* and *uniform* density of solutions everywhere in the solution space.

3.1.3 Analogy of MOGA with Statistical Model of Ideal Gas

Maxwell and Boltzmann developed the statistical theory of gases in the nineteenth century. The problem of statistical thermodynamics begins with a system about which only some macroscopic information is known. For example, consider the case of a monoatomic gas inside a container. In classical mechanics, the state of the gas is fully understood only if the position and momentum of each molecule is known. In statistical mechanics, however, such details are beyond our knowledge, or impossible to measure with certainty. Therefore, one only needs to know the general properties of the system (and not individual molecules). It can be shown that -- regardless of the

position and momentum of each individual molecule -- the overall behavior of a Maxwellian system of molecules (described in the next section) can be described as a function of only macroscopic properties of the system and time.

In this chapter, we use several properties and similarities between a Maxwellian system and MOGA to produce a greater diversity of solutions in the solution set, as in the following:

- A Maxwellian system of N particles always re-distributes itself such that it reaches an equilibrium state, i.e.,
    1. It becomes homogenously distributed in the vessel (local fluctuations in the density of molecules exist, but the average is the same everywhere in the vessel). The uniformity of solutions is a very favorable, and yet hard to reach, quality in MOGAs.
    2. The distribution becomes time-independent (steady-state). This corresponds to convergence of population in MOGA to a state of maximum diversity and remaining in that state.
- As discussed in the next section, entropy of a Maxwellian system (that quantifies disorder in the system) never decreases whether the system is in equilibrium or not. Indeed, it can be proved that Maxwellian system is the *only* system that has this property (See for instance Andrews 1963). In terms of MOGA, we always want to increase diversity in the population during the optimization process, and a Maxwellian system is the only system that is guaranteed to do so. (T-MOGA is only an approximation of a Maxwellian system, however, as discussed later and shown by many

examples in Chapters 6 and 7, it demonstrates this property with minor fluctuations.)

Note that the term 'entropy' in this chapter refers to Boltzmann's definition of entropy, which is very similar to Shannon's information entropy (see Chapter 2, Equation 2.19), but based on totally different assumptions. A formalism recently introduced by Prugel-Bennett and Shapiro (1997, 1999) also observed the usefulness of Boltzmann's entropy to analyze the dynamics of GAs. Their study was focused on single objective GA and was performed at a genotypic level (i.e., manipulates binary bits in a chromosome). The approach in this dissertation however is multi-objective and phenotypic in nature (i.e., manipulates the diversity of population in the objective space).

In the next section, we briefly present Maxwellian system of particles (ideal monoatomic gas) and its properties. This is used later in Section 3.2 to develop T-MOGA.

3.1.4 Maxwellian Systems and Boltzmann's Entropy

Consider the simplest case of a Maxwellian equilibrium system: a large number of particles, say N, enclosed in a vessel of volume **V.** (A more detailed discussion can be found in -- among others -- Andrews 1963, and Fay 1965.) It is assumed that these particles move freely in the vessel unless they hit the walls of the vessel, which will in fact reflect them back and keep the gas inside the vessel. The state-space is defined as a $2n$-dimensional domain whose coordinates are $x_1, ..., x_n$, and $v_1, ..., v_n$, where n is the dimension of the space, and $v_1, ..., v_n$ represent the velocity of particles with respect to $x_1, ..., x_n$, respectively. Therefore, to know the exact microscopic state of this system

we need $2n$xN quantities. If we are interested only in the macroscopic behavior of the system, however, we can use statistical methods and neglect microscopic fluctuations (fluctuations become totally negligible for large numbers of particles). The density function, $f(\mathbf{x},\mathbf{v},t)$, is defined as the density of points in this state-space. So

$$f(\mathbf{x},\mathbf{v},t)\, d\mathbf{x}\, d\mathbf{v} \tag{3.1}$$

is the number of particles in d**x** the velocity of which lie in the interval d**v**. Equilibrium assumption states that density, f, is (i) statistically constant everywhere in the vessel and thus, does not depend on **x**, (ii) at steady state and therefore is not a function of time.

From these assumptions and using energy as a macroscopic invariant (constant of the system), Maxwell showed that the density function has the following distribution (known as Maxwellian distribution):

$$f(\mathbf{v}) = n\,(\lambda/\pi)^{3/2} \exp(-\lambda v^2) \tag{3.2}$$

where λ is a positive constant that depends on energy (or equivalently, temperature) of the system. Boltzmann's entropy of a system of particles is defined as follows (compare to Equation 2.19):

$$H = -k\, E(\log f) = -k \int f . \log f . d\mathbf{v} \tag{3.3}$$

where the integration is performed over state-space and therefore H is a function of time only. The above definition is given for all distributions (whether Maxwellian or not) and quantifies disorder in the system. Boltzmann proved that H has a very important property, known as Boltzmann's H-Theorem (see for instance Haar, 1964).

Boltzmann's *H*-Theorem: In a Maxwellian system (with the Maxwellian distribution of Equation 3.2), we have $\mathrm{d}H/\mathrm{d}t \geq 0$. In other words, *H* never decreases with time.

The definition of *H* in Equation 3.3 is unique in the sense that it is the only macroscopic function of the system (multiplied by a constant) that never decreases with time. If such a system deviates from an equilibrium state, it is guaranteed to return to equilibrium with maximum *H*. The relaxation time (time to go back to equilibrium) depends on many parameters, and is very small (of the order of 10^{-9} for standard monoatomic gas) for large systems of particles. More importantly, the Boltzmann's *H*-Theorem not only is a sufficient condition for attaining a stable equilibrium state, but also provides a necessary condition, i.e. the Maxwellian distribution is the only distribution that guarantees attaining steady equilibrium state (with maximum *H*; See for instance Haar 1964.)

The similarity between Boltzmann's entropy and Shannon's information entropy is strikingly interesting, and it has been the subject of considerable debate and confusion in the literature. In this chapter, when we talk about 'entropy' it is interpreted in the context of statistical mechanics. The 'entropy' of Chapters 2, and 5 however -- as mentioned clearly in those chapters -- is interpreted in the context of information theory as explained in Section 2.1.5 (see Brillouin 1962 for differences in the interpretation of entropy in thermodynamics and information theory).

In the rest of this chapter, we take advantage of the above-mentioned property of Maxwellian systems to improve the distribution quality of solutions in MOGA. In short: *The proposed T-MOGA assigns a velocity to each solution point and expands the*

population accordingly. The resulting system is therefore bound to maximize H, and achieve a final equilibrium state, which: 1) *has statistically uniform density of solutions everywhere in the design space (with the feasibility constraints that will be treated as the walls of a vessel); and* 2) *is time independent, and therefore, becomes steady once equilibrium is achieved.* Because of the practicality issues of implementing T-MOGA, the simulation is only partial (Section 3.2.4), however, it is shown through examples (mainly in Chapter 6) that apart from minor fluctuations, T-MOGA still tends to closely follow *H*-Theorem.

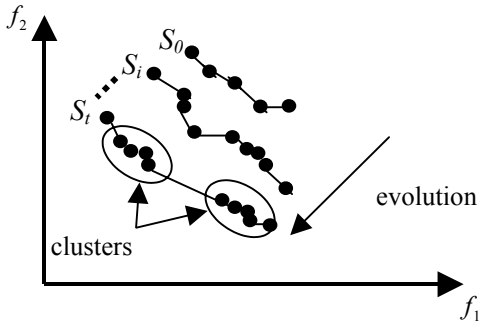
3.2 THERMODYNAMICAL MOGA (T-MOGA)

In this section, T-MOGA is developed based on an analogy with equilibrium statistical model of ideal gas. The new algorithm aims at obtaining the best possible (statistically uniform) description of the Pareto frontier.

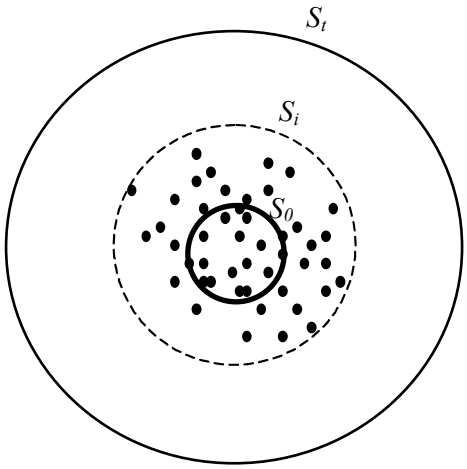
3.2.1 Approach

Consider the operations of a two-objective MOGA, as shown in Figure 3.1(a). We designate the set of non-dominated points after t iterations as S_t. As the population evolves from the non-dominated points for the initial population, S_0, to the non-dominated points after t iterations, S_t, the Pareto solution set or frontier is formed. The 'iteration number' in T-MOGA, t, corresponds to the 'time' in statistical thermodynamics. Due to evolutionary nature of MOGAs, the Pareto solution set usually consists of several clusters of points that may not cover the entire range of Pareto frontier (See Figure 3.1(a)). Clustering is a well-known and undesirable

phenomenon in MOGAs. According to Deb (1998), an ideal MOGA should spread the solution points evenly, as much as possible, along the Pareto frontier.



(a)



(b)

Figure 3.1: An analogy of (a) evolution of individuals in a MOGA, with (b) expansion of molecules of an ideal gas in an enclosure

As shown in Figure 3.1(b), a direct analogy can be observed between the operation of a MOGA and an ideal gas undergoing an expansion in an enclosure. According to the definition of an ideal gas, there is no interaction between the molecules of an ideal gas in an enclosure. That is, the molecules move along random directions until they collide with the boundary of the enclosure. As a result of the collision with the boundary, the molecules are reflected back into the enclosure again

with the same velocity but along a different direction following the mirror's law. As we mentioned before, if the velocity distribution of such a system follows the Maxwellian distribution, the set of particles will automatically evolve to maximize the entropy and consequently result in the uniformity and the coverage of the molecules in the enclosure. To an outside observer, the gas expands gradually until it fills in the inside of the enclosure and eventually reaches to a uniform, homogenous and time-invariant state.

The objective of this part of research is to modify a MOGA so that it enables an expansion of solutions along lateral directions. (A lateral direction is defined later in this chapter. In short, it is a random direction normal to the evolution direction shown in Figure 3.1(a).) To achieve this objective, we proceed with some modifications to a baseline MOGA, as described in the following sections. (The baseline MOGA, hereafter called MOGA-NA, was recently developed by Narayanan and Azarm, 1999.)

3.2.2 Velocity Assignment

A velocity value is assigned to every individual of the population in T-MOGA, according to an ideal gas model. At the beginning of the process, each individual is assigned with a velocity according to the Maxwellian velocity distribution in Equation 3.2. This velocity will remain constant during the entire optimization process. This is similar to the case of the ideal gas, where the magnitude of velocity of a molecule does not change but its direction changes as it collides with the boundary of the enclosure.

In addition to velocity values, for each solution point a lateral expansion direction is assumed normal to the progress vector (i.e. general direction of evolution). This is described next.

3.2.3 Transverse Expansion Hypersurface

MOGA operations mainly consist of evolving the population via the GA operators of crossover and mutation, as it approaches the Pareto frontier. To extend the MOGA operations so that it also simulates an expanding ideal gas in an enclosure, at the beginning of each iteration the population is expanded transversely, normal to the *progress vectors*. Progress vectors are defined as the vectors along which the individuals in the population are expected to evolve. The progress vector is different for each point in the population and should be estimated individually.

To estimate this vector, we propose an algorithm that is based on the relative position and fitness of points in the objective space while all operations are performed in the variable space.

Consider the two-objective optimization problem shown in Figure 3.2, wherein there are N_1 non-dominated solution points in the population at its current stage of evolution. We label these points as P_1, P_2, …, P_{N1}. If we eliminate these points from the population, we obtain N_2 non-dominated points, denote these as P_{N1+1}, P_{N1+2}, ..., P_{N1+N2}. As we repeat this process, we divide the population into several sets of points. We assign a ranking of one to the first set (i.e., N_1 points, the fittest points in the current population), a ranking of 2 to the second set of points, and so on. Each set will form a curve or a hyper-surface for the higher dimensions of the objective space, hereafter referred to as a *transverse-expansion hyper-surface*.

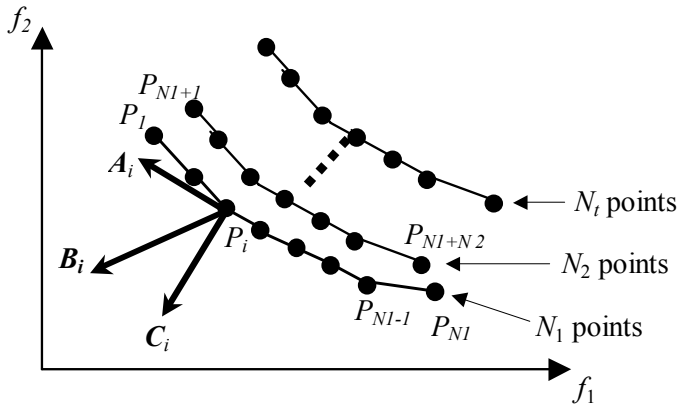


Figure 3.2: Estimating the progress vector in a two-dimensional objective space

The lateral expansion of the solutions is done in a direction tangent to this hyper-surface. As the solutions are evolved, these transverse hyper-surfaces gradually converge to the Pareto frontier. This hyper-surface will be (m-1)-dimensional in an m-dimensional objective space. For example as shown in Figure 3.3, the transverse hyper-surface of a three-objective problem will be a two-dimensional surface on which the population is expanded laterally.

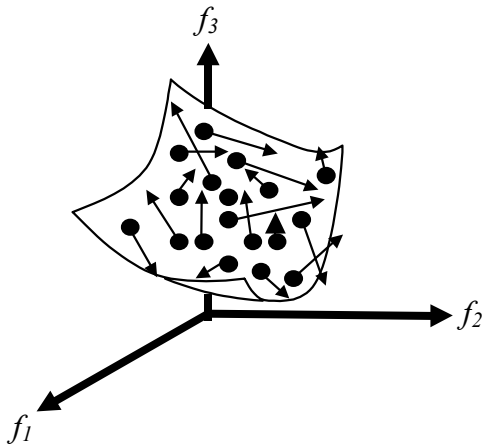


Figure 3.3: Transverse-expansion of individuals in T-MOGA

Consider the point P_i on the first non-dominated set ($1 \le i \le N_1$), see Figure 3.2. We denote the vector connecting P_k to P_i as $\boldsymbol{Z}_{ki}$. We define an offset vector $\boldsymbol{A}_i$ as:

$$\boldsymbol{A}_i = \frac{(\sum_{k=1}^{N_1} \boldsymbol{Z}_{ki})}{N_1} \quad (1 \le i \le N_1) \tag{3.4}$$

wherein $\boldsymbol{A}_i$ is an offset vector, since it is almost tangent to the transverse hyper-surface and points to the nearest edge of the non-dominated set. In addition, the magnitude of this vector is larger for a point near the edge of the set while in the middle of the set its magnitude is small. Similarly, the vector $\boldsymbol{B}_i$ is defined:

$$\boldsymbol{B}_i = \frac{(\sum_{k=N_1+1}^{N_1+N_2} \boldsymbol{Z}_{ki})}{N_2} \qquad (1 \le i \le N_1) \tag{3.5}$$

As can be seen in Figure 2, since the points in N_1 are evolved (in terms of all objectives) as compared to the points in N_2, $\boldsymbol{B}_i$ consists of a normal-progress component in addition to the offset vector. If the points in the first and second non-dominated sets, N_1 and N_2 points in Figure 3.2, respectively, are distributed in the same fashion, then the offset vectors due to both of these first and second non-dominated sets are approximately equal. Hence, one can subtract the offset vector, $\boldsymbol{A}_i$, from $\boldsymbol{B}_i$ to obtain a progress vector.

$$\boldsymbol{C}_i = \boldsymbol{B}_i - \boldsymbol{A}_i = \frac{(\sum_{k=N_1+1}^{N_1+N_2} \boldsymbol{Z}_{ki})}{N_2} - \frac{(\sum_{k=1}^{N_1} \boldsymbol{Z}_{ki})}{N_1} \qquad (1 \le i \le N_1) \tag{3.6}$$

where $\boldsymbol{C}_i$ estimates the progress vector of the point P_i (Figure 3.2). So far, we have assumed that P_i is in the first non-dominated set, i.e., the N_1 points in the population. For a point P_i, in general, in the m-th non-dominated set, we have:

$$C_i = \frac{\left(\sum_{k=N_1+N_2+\ldots+N_m+1}^{N_1+N_2+\ldots+N_m+N_{m+1}} Z_{ki}\right)}{N_{m+1}} - \frac{\left(\sum_{k=N_1+N_2+\ldots+N_{m-1}+1}^{N_1+N_2+\ldots+N_m} Z_{ki}\right)}{N_m} \qquad (3.7)$$

where: $(N_1+N_2+\ldots+N_{m-1}+1) \leq i \leq N_1+N_2+\ldots+N_m)$.

It is clear that the above mentioned estimates becomes less accurate for very inferior or poor ranking points and their corresponding hyper-surfaces since they may not consist of enough points to give an acceptable estimation of offset and progress vectors. However, as mentioned in the next section, we are mainly interested in the good-ranking hyper-surfaces since they provide a better chance of reproduction.

Now that we have obtained $\boldsymbol{C}_i$, a *lateral movement vector* is chosen randomly in a direction normal to the progress vector of each individual, as shown in Figure 3.4. Each point moves along its lateral movement vector until it attempts to violate a constraint -- just like a molecule in an enclosure that moves along a straight line until it collides with a boundary of the enclosure (This will be discussed in Section 3.2.5). In the next section, expansion operators are introduced in T-MOGA to expand the population of solutions in a lateral direction according to the Maxwellian distribution of equation 3.2.

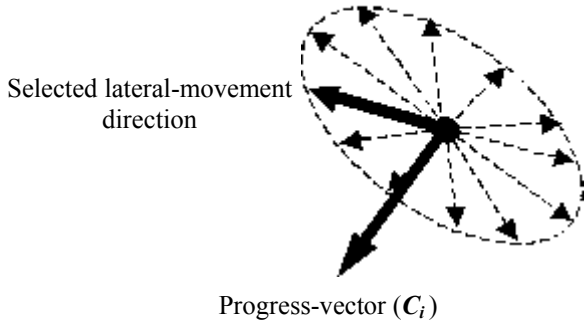


Figure 3.4: A lateral movement vector chosen randomly and normal to the progress vector of each individual

3.2.4 Expansion Operator

The expansion operator is applied at every iteration. This operator enhances each individual (or a certain percentage of the population) as follows.

a. A new chromosome (child) is generated by moving the chosen individual (single-parent) along its assigned lateral-movement vector in the variable-space.
b. The magnitude of the movement is proportional to its assigned velocity (as described in Section 3.2.2.

Since this operator is applied every iteration, it simulates the gradual movement and expansion of gas molecules with different velocity magnitudes and directions. However, there are two parameters that should be set in T-MOGA, the expansion percentage and expansion start, as described in the following.

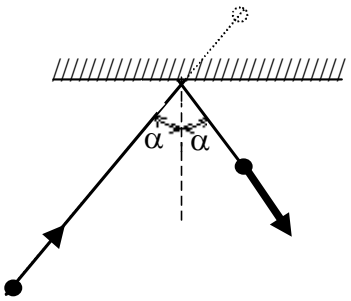
Expansion Percentage: To keep the number of function calls as low as possible, we apply the expansion operator only to a certain percentage of the fittest individuals in

the population. As these points are expanded to the new regions, they act like seeds for the new offspring in those regions.

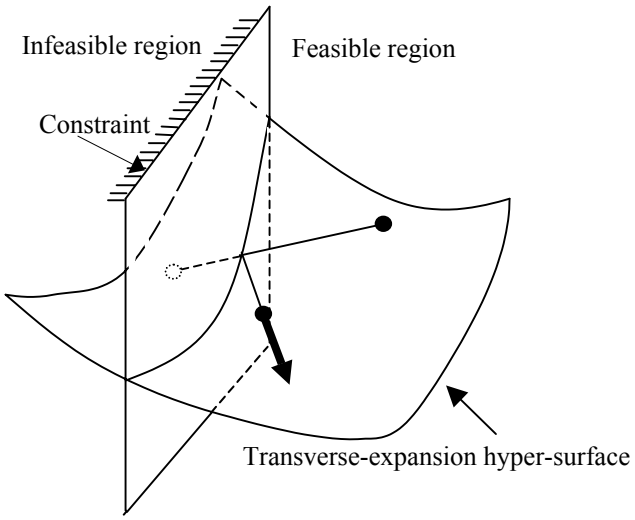
Expansion Start: As described in Section 3.2.3, the estimation of the progress vector is based on the relative position and fitness of the points. Specifically as the number of the first and second ranked non-dominated points (i.e., N_1 and N_2 points in Figure 3.2) increases, the corresponding progress and lateral-movement vectors can be estimated more accurately. Therefore, if we let the population evolve for a few iterations before we start applying the lateral expansion operator, we will have more points in these sets resulting in a more accurate estimation of the progress and lateral-movement vectors. In the examples discussed at the end of this chapter, for instance, the expansion starts at the fifth generations. Thereafter, the expansion operator is applied to all subsequent generations.

3.2.5 Constraint Handling

When a gas molecule collides with the boundary of an enclosure, it is reflected back into the enclosure without a change in its speed but the movement direction will be different and follows the mirror's reflection law (Figure 3.5(a)). Similarly, a constraint for T-MOGA is a hyper-surface in the variable space and can be treated as a wall or a boundary (Figure 3.5(b)).



(a)



(b)

Figure 3.5: (a) The direction of movement changes as a molecule hits a constraint, (b) the individual is reflected back into the feasible region

Assume that all individuals in the initial population are feasible. Every time that the population undergoes an expansion (i.e., an expansion operator is applied), there might be several points that attempt to violate one or more constraints to enter the infeasible region. We reflect these points back into the feasible region according to the mirror's reflection law and assign to them a new lateral-movement vector. This constraint handling aspect of T-MOGA makes it a "feasible-direction" type method.

This is because if all individuals in the initial population satisfy the constraints, all subsequent generations will remain within the feasible domain. To ensure that the initial population is feasible, T-MOGA checks for infeasible points before the first generation undergoes expansion and replaces, via a random number generator, all infeasible individuals with feasible ones. This is continued until all individuals in the initial population are feasible. A step-by-step description of T-MOGA is given next.

3.2.6 T-MOGA: Step-by-Step

A detailed flowchart for the proposed T-MOGA is shown in Figure 3.6. Below, the algorithm is given step-by-step.

Step 1: The initial population is generated randomly until all individuals are feasible.

Step 2: A velocity is assigned to each individual in the population according to the Maxwellian probability distribution function. This velocity remains constant throughout the process. (See Section 3.2.2)

Step 3: The individuals in the population is rank-ordered and the hyper-surfaces are created (i.e., sets N_1, N_2, ..., as described in Section 3.2.2)

Step 4: The progress vectors of the individuals are estimated from 3.7.

Step 5: A lateral movement vector is assigned to each individual randomly, normal to the progress vector obtained in Step 4.

Step 6: The expansion operator is applied to a pre-specified percentage of the population. (See Section 3.2.4)

Step 7: After the expansion is applied to the individuals, they are checked for a constraint violation.

Step 8: If an individual violates a constraint, it is reflected back from the corresponding constraint boundary and a new movement direction is assigned according to the mirror reflection law (Section 3.2.5). Go back to Step 7.

Step 9: The expanded population is evaluated and a fitness value is assigned to the individuals.

Step 10: A percentage of the individuals are selected for reproduction. The chance of being selected is higher for the fitter individuals (Narayanan and Azarm, 1999).

Step 11: The selected parents undergo crossover and mutation. The offspring are added to the current population. Then the entire population is rank-ordered and the worst individuals are discarded.

Step 12: If the stopping criterion is not met, go to Step 3 and continue. Otherwise, stop.

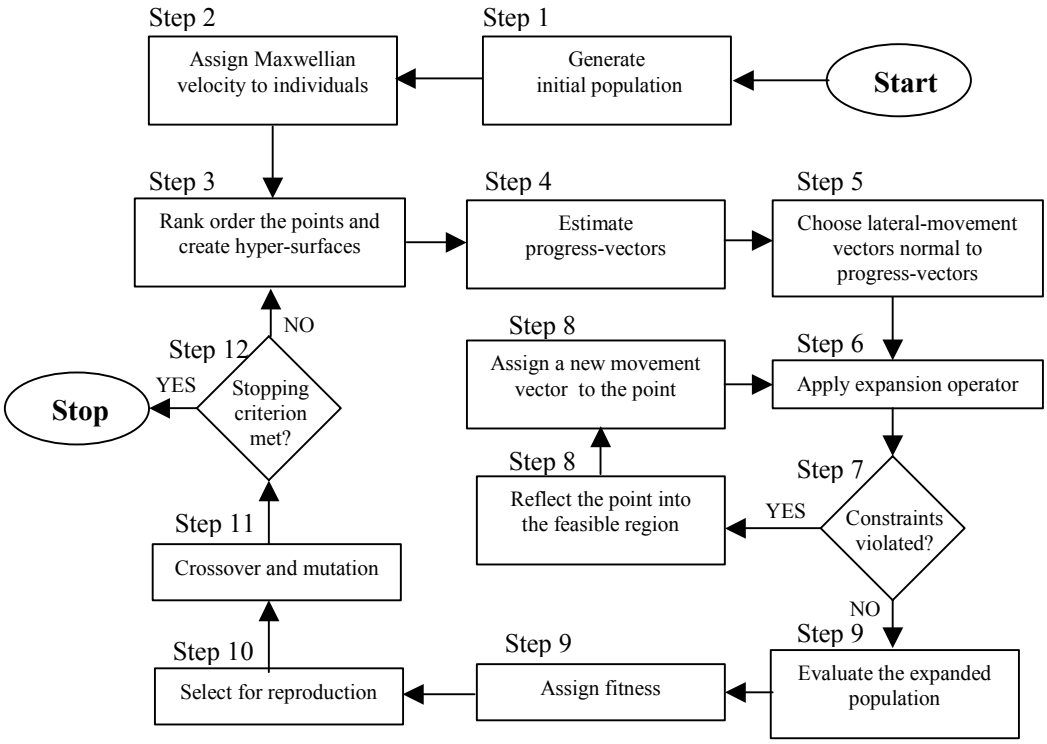


Figure 3.6: Flowchart of T-MOGA

As stated before for the ideal gas expansion, the population of chromosomes in T-MOGA is expected to gradually achieve the maximum-entropy macrostate that indicates a statistically uniform and steady-state (independent of time and position) density. During the process, as the individuals are expanded, the entropy of the population increases gradually.

In the next section, the performance of T-MOGA is demonstrated with a simple example. More test examples and a case study can be found in Chapters 6 and 7, respectively.

3.3 DEMONSTRATION EXAMPLE

The performance of T-MOGA is tested using a simple engineering design example: optimal design of a two-bar truss. This example was originally formulated as a single-objective problem by Kirsch (1981). The problem was reformulated as a two-objective problem by Narayanan and Azarm (1999) to demonstrate the application of a MOGA. In this example, as illustrated in Figure 3.7, the vertical position of point C and the cross-sectional areas of links AC and BC are to be selected and thus the design variables, which are all continuous, are x_1, x_2 and y. The objective of this design is to minimize the total volume (and consequently the weight) of the structure as well as the tensile stress in link AC. The constraints are the maximum allowable stresses in AC and BC that should not exceed 100,000 kPa and the total volume of the material used in the structure that should be held less than 0.1m^3.

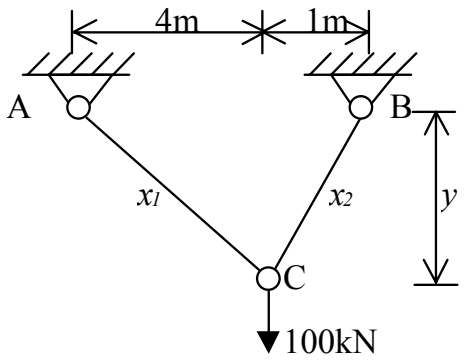


Figure 3.7: Two-bar truss

Constraints are imposed on the design objectives (i.e., objective constraints), as shown in the formulation of the problem. This is because the Pareto set is asymptotic and extends from -∞ to ∞. As x_1 and x_2 go to zero, f_{volume} goes to zero and $f_{stress,AC}$ and $f_{stress,BC}$ go to infinity. As x_1 and x_2 go to infinity, f_{volume} goes to infinity and $f_{stress,AC}$ and $f_{stress,BC}$ go to zero. Hence, in order to generate Pareto optimal solutions in a reasonable range, objective constraints are imposed. The problem formulation is given below.

$$\text{Minimize } f_{volume} = x_1(16+y^2)^{0.5} + x_2(1+y^2)^{0.5}$$

$$\text{Minimize } f_{stress,AC} = \frac{20(16+y^2)^{0.5}}{yx_1} \tag{3.8}$$

$$\text{Subject to : } f_{volume} \le 0.1$$
$$f_{stress,AC} \le 100000$$
$$f_{stress,BC} \le 100000$$
$$1 \le y \le 3$$
$$x_1, x_2 > 0$$

where

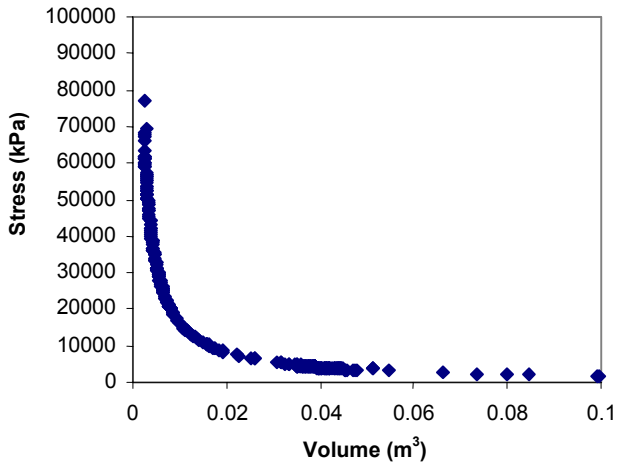
$$f_{stress,BC} = \frac{80(1+y^2)^{0.5}}{yx_2}$$

The problem has been solved with both a baseline MOGA (Narayanan and Azarm, 1999) and the T-MOGA. The GA's parametric values used to solve this problem are listed in Table 3.1.

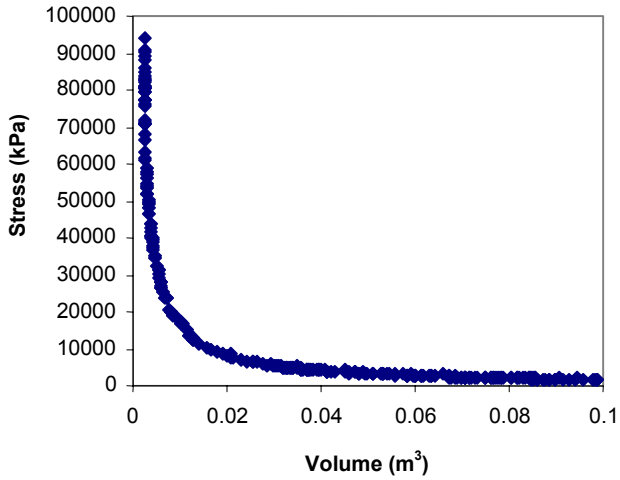
	MOGA	T-MOGA
Population size	50	50
Replacement per generation	10	10
Function calls	550	550
Crossover type	2-point	2-point
Crossover probability	0.8	0.8
Mutation probability	0.05	0.05
Bits per variable	10	10
Selection type	Stochastic universal selection (Levine, 1996)	Stochastic universal selection (Levine, 1996)
Expansion percentage	NA	10%
Expansion start	NA	5th
Expansion finish	NA	45th

Table 3.1: MOGA parameters

The results from both techniques are shown in Figures 3.8(a) and (b). As shown in these figures, a significant improvement is observed in the results obtained by T-MOGA when compared with those by the baseline MOGA. In Figure 3.8(a), the majority of solution points are clustered in two regions and the rest of the Pareto frontier is left empty or very sparingly populated. In contrast, T-MOGA results show a more evenly distributed set of solution points along the Pareto frontier without any noticeable clustering of the points. Moreover, comparing the range of the optimal solution set obtained from each technique, one can notice that T-MOGA covers a larger portion of the Pareto frontier. Based on this example, it can be concluded that the optimal set generated by the T-MOGA is much better than that generated by the baseline MOGA in terms of uniformity and coverage of the Pareto frontier (For a case study and more examples see Chapters 6 and 7.)



(a)



(b)

Figure 3.8: Pareto solution set for the two-bar truss example obtained by (a) baseline MOGA, and (b) T-MOGA

3.4 CONCLUDING REMARKS

In this chapter, we presented a new multi-objective genetic algorithm -- Thermodynamical MOGA (T-MOGA) -- based on an analogy with the thermodynamic behavior of ideal gas undergoing expansion in an enclosure. T-MOGA expands a sample of population while evolving to reach the Pareto frontier, and therefore

achieves better uniformity and coverage of the solutions. The constraint handling technique developed in T-MOGA is in fact a feasible direction approach, that is, if individuals in the initial population satisfy the constraints, all of their subsequent generations will remain feasible.

For demonstration, both T-MOGA and baseline MOGA were applied to a two-bar truss design problem. The results showed a significant improvement in the solutions obtained by T-MOGA in terms of uniformity and coverage of solutions over those obtained from the baseline MOGA. In particular, the Pareto solutions obtained from the baseline MOGA contained obvious gaps and clusters while those from the T-MOGA were almost free of them. In addition, the solutions obtained from the T-MOGA covered a significantly wider range of the Pareto frontier compared to the baseline MOGA.

In the test example of this chapter, we used a visual demonstration of solution sets obtained from two MOGAs to compare their performance (Figure 3.3-2). Visual comparison, however, is not practical for problems with 3 or more objectives. Therefore, researchers have developed quantitative measures for quality of non-dominated solution sets, referred to as *quality indexes* in this thesis. The next chapter discussed such indexes and their properties. More importantly, we develop a theoretical platform for selection of a comprehensive yet non-overlapping set of such indexes to compare the quality of MOGAs.

CHAPTER 4

MINIMALITY OF QUALITY INDEXES

Once a Pareto solution set to a multi-objective optimization problem is obtained, it is usually of great interest to know how 'good' the observed solution set represents the Pareto frontier. This can be done visually, by relying on the designer's intuitive judgment, or quantitatively, via quality-capturing indexes. However, visual quality assessment of solution sets is not always reliable in comparison study of different evolutionary multi-objective optimization techniques, raising the need for quantitative quality indexes.

In this chapter, we propose a conceptual framework for selection of a handful of these indexes such that all desired aspects of quality are addressed with minimum or no redundancy. Indeed, we prove that such sets of indexes -- referred to as 'minimal sets' -- must be constructed based on a one-to-one correspondence with quality aspects that are set forth by a decision-maker.

This chapter is organized as follows: Section 4.1 provides a brief background on quality indexes, and their properties. Section 4.2 presents a theoretical platform for selection of quality indexes based on inputs from a decision-maker. More importantly, set minimality of quality indexes is formally defined, and Minimality Theorem is proved. Finally, Section 4.3 presents a simple test case to investigate the applicability of the proposed framework.

4.1 BACKGROUND AND PREVIOUS WORK

The following subsections provide a brief literature review of the quality indexes and their properties and shortcomings.

4.1.1 Quality Indexes

Researchers have developed a myriad of techniques over the last few years to improve the quality of solution sets in one way or another. Naturally, performance assessment and comparison study of multi-objective genetic algorithms have also gained much attention. One obvious way to compare MOGAs is to simply visualize the final sets of solutions and rely on intuitive and visual judgments to decide the superiority of one technique to another. However, as discussed by Van Valdhuizen and Lamont (2000), visual assessment is not a reliable tool for comparison of different multi-objective optimization techniques. Especially, for problems with more than three-dimensions, visual judgment is either impossible or quite misleading, yet it is the prevailing approach used by many researchers.

More recently, there has been an emerging theme in the literature to quantitatively assess and compare the quality of non-dominated solution sets (NDS; refer to Appendix-I, for definition) via *quality indexes*. These quality indexes generally assign an absolute or relative value to an NDS to determine whether it is a 'good' representation of the Pareto frontier. For instance, Zitzler and Thiele (1998) performed a comparative study of several multi-objective optimization methods using two indexes: "size of the dominated space" and "fraction of solutions dominated by the other set". Van Veldhuizen (1999) introduced several quality indexes, such as: 'error ratio', 'generational distance', 'maximum Pareto frontier error' and 'overall non-

dominated vector generation ratio' to assess different aspects of a solution set quality. Sayin (2000) also defined indexes for coverage, uniformity and cardinality to determine how 'good' a set of discrete solution points represents the Pareto frontier. However, application of most of these indexes to real world problems may encounter difficulties because these indexes usually need some knowledge of the Pareto set or at least a reference set (i.e., an approximation of the Pareto frontier) which is not always available. Moreover, these indexes may be conflicting in the sense that while a MOGA may outperform with respect to another MOGA in terms of one index, at the same time, it may perform poorly with respect to (w.r.t.) another index. To resolve the former issue, Wu and Azarm (2001) suggested a new set of quality indexes (a few of which were adapted and modified from the literature). These indexes address the quality of a solution set without any a priori knowledge of the Pareto frontier. However, similar to the Van Veldhuizen's indexes, Wu and Azarm's indexes could be conflicting, introducing tradeoffs among different aspects of quality. (For a recent review of the quality indexes and their shortcomings, see Knowles 2002). In fact, as discussed in the following section, one must select a set of quality indexes such that all aspects of quality are addressed without unnecessary correlations among them.

4.1.2 Minimality: Encapsulation vs. Exhaustiveness

According to Deb (1998) there are two main tasks that MOGAs should address: 1) *fast convergence to the Pareto frontier*, and 2) *maintaining diversity*. Bosman and Thierens (2003) state that there is a tradeoff between these two goals in most cases, although MOGAs that perform better with respect to both are feasible. The common trend in the literature is to 'decompose' each of the above two tasks into several

subtasks and develop numerous quality indexes accordingly. For example, there are several indexes in the literature that are claimed to assess 'diversity of solutions' in one way or another, including: 'spacing index' (Schott, 1995); 'overall non-dominated vector generation', 'overall non-dominated vector generation ratio' (Van Veldhuizen 1999); 'coverage', 'uniformity', 'cardinality' (Sayin 2000); 'number of distinct choices', 'Pareto spread', and 'cluster' (Wu and Azarm 2001). In a similar fashion, researchers developed numerous indexes to assess the closeness of solution sets to the Pareto frontier (see Knowles 2002, for examples of these indexes).

Obviously, many of indexes that address a common aspect of quality are correlated in one way or another, introducing redundancy in a comparison study of MOGAs. Therefore, selecting too many of these indexes is not only confusing but also transforms the quality assessment of MOGAs into unnecessary and complex tradeoffs among different indexes. In fact, it is virtually impossible to find a situation in which a multiobjective optimization algorithm outperforms other algorithms in terms of all existing quality indexes. That is, one algorithm for example may produce more distinct solutions while the other distributes them more uniformly and a third one performs better in terms of having no gaps among its solution points.

Having the correlation among different quality indexes in mind, one may decide to 'encapsulate' (rather than decompose) different aspects of quality into one or a few scalar indexes. For instance, one might consider only two scalar indexes that encapsulate all of the aforementioned indexes: one representing the convergence to the Pareto frontier, and the other representing the diversity of solution points. However, as shown later in this chapter, selecting too few quality indexes does not guarantee an

'exhaustive' comparison with respect to all aspects of quality. This in turn poses a new question to researchers: which index or collection of indexes must be used for an exhaustive (but not redundant) comparison study of different MOGAs?

Later in this chapter, it is shown that a desirable collection of quality indexes must be *minimal* in the sense that: 1) There exists at least one index for every aspect of the solution set quality to guarantee an exhaustive performance assessment; 2) There exists minimum (or no) correlation among quality indexes to avoid redundancies (see Section 4.2.5 for a formal definition). Due to the subjective nature of the concept of 'quality', it is shown that minimality can be defined only based on a decision-maker's understanding of a 'good' solution set. In fact, the Minimality Theorem of Section 4.2.5 proposes a guideline for building a minimal set of quality indexes based on the input from a decision-maker.

4.1.3 Binary / Unary Quality Indexes and Outperformance Relations

By definition, a unary quality index yields an absolute value to quantify the goodness of a given solution set. On the other hand, a binary index only compares the quality of two solution sets and return a relative value. For reasons that will be discussed at the end of this section, the focus of this research is only on binary indexes.

Definition 4.1.3-1: If A and B are two NDS's, then the binary index $Q(A,B)$ returns a scalar that reflects how much better set A is when compared to set B. (The discussion in this chapter also applies to other indexes that can be transformed into the binary format.)

Moreover, the binary index Q is said to be symmetric and homogeneous iff $Q(B,A) = -Q(A,B)$. Note that the general definition for a symmetric index is: $Q'(B,A) = C - Q'(A,B)$. However, without loss of generality, this latter format can be easily transformed into a homogenous symmetric index by assuming: $Q(A,B) = C/2 - Q'(A,B)$. Also, without loss of generality, it is assumed that $Q(A,B) > 0$ iff A is strictly better than B. If we define the set of all possible NDSs by U, A binary quality index as defined above constructs a total order in U and compares any two NDS's on a quantitative basis.

Other than quality indexes that compare the quality of all NDS's, Hansen and Jaszkiewicz (1998) defined the outperformance relations to establish a strict partial order among NDS's, where some pairs of solution sets are objectively comparable in terms of Pareto optimality (or dominance), as in the following:

Definition 4.1.3-2: A non-dominated set A strongly outperforms a non-dominated set B, denoted by AR_OB, iff $A \neq B$ and (in the objective space) for each $y \in B$, there exists $x \in A$ such that $x \succeq y$. (In addition to the strong relation, Hansen and Jaszkiewicz (1998) also defined weak and complete outperformance relations.)

The notation '$\succeq$' in $x \succeq y$ indicates that point x is either equal to or dominates point y with respect to all objectives. Figure 4.1 demonstrates two non-dominated solution sets generated for a 2-objective maximization problem. According to Definition 4.1.3-2, we observe that: AR_OB.

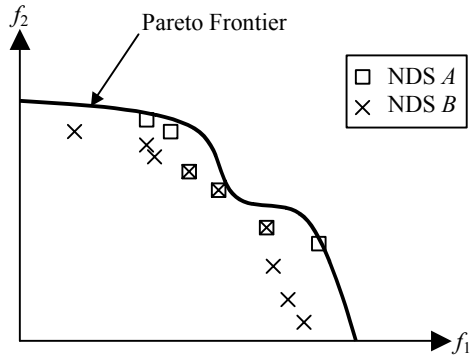


Figure 4.1**:** AR_OB: set A strongly outperforms set B

Clearly, not every two NDS's are comparable by outperformance relations. Then the above relation constructs a strict partial order in U. Here we define the *partially ordered domain* of this comparison in UxU -- denoted by Λ_{Ro} -- as the set of all 2-tuples of NDS's that are comparable via R_O, i.e.,

$$\Lambda_{Ro} = \{(A,B) \in U\text{x}U \mid \text{either } AR_OB \text{ or } BR_OA\} \tag{4.1}$$

Outperformance relation accounts for pairs of NDS's where one solution set is objectively better (based only on the notion of dominance) than the other set and thus, they establish *strict partial orders* among NDS's. Put another way, not all solution sets are comparable in this way, but at least one can verify the validity of a quality index by examining its compatibility with these relations. Based on this idea, Zitzler et al. (2002) investigated compatibility and completeness of different comparison methods and derived a set of theoretical restrictions for the existence of compatible and complete *unary* quality indexes. In fact, they prove that a finite combination of unary indexes that is compatible and complete at the same time does not exist in general. They also mention that this limitation does not apply to *binary* quality indexes, and

therefore, one can construct compatible and complete indexes with respect to any dominance relations.

As such, the focus of this chapter is on minimality and compatibility of *binary* quality indexes and their correspondence with partial order relations (referred to as excellence relations in the following section.) Although there are relatively fewer binary quality indexes in the literature as compared to unary indexes (Zitzler et al. 2003), the above property makes them very attractive for a comparison study of multiobjective optimization algorithms, mainly because one could select a minimal set of binary indexes to address all desired aspects of quality exhaustively and distinctly. (This is not possible in general with unary indexes.)

4.2 SET MINIMALITY OF BINARY QUALITY INDEXES

In this section, we will derive a theoretical framework for selection of a minimal set of symmetric and homogeneous binary indexes.

4.2.1 Excellence Relations

Outperformance relation by itself does not provide a tool to compare any given pair of NDS's. This relation is based only on the concept of dominance (or closeness to the Pareto frontier), and therefore, if a quality index aims at comparing NDS's in terms of dominance, it must be compatible with this relation in the first place. (The formal definition of compatibility is given in Section 4.2.2.)

However, there are other aspects of quality that are especially important in the assessment of solution sets obtained from MOGAs, e.g., diversity of solution sets, or extent of the Pareto frontier that is covered by the solutions. Very similar to the

outperformance relation, one can collect all 2-tuples of NDS's that are objectively comparable with respect to any aspect of quality and construct a strict partial order accordingly. This prompts for the definition of a more general concept, hereafter referred to as *excellence relations* (given next).

Definition 4.2.1-1: An excellence relation, denoted by R, is defined as a strict partial order in U that relates all non-dominated sets that are objectively comparable with respect to a common aspect of quality. The partially ordered domain of R in UxU -- denoted by Λ_R -- is defined as: $\Lambda_R = \{(A,B) \in U\text{x}U \mid \textit{either } ARB \text{ or } BRA\}$.

For example, an outperformance relation is an excellence relation with respect to dominance. As another example, in the following, we define a new excellence relation (i.e., *coverage relation*) to address a different aspect of quality: *coverage* (i.e., the span of a solution set over the Pareto frontier). In this example, it is assumed that all objective functions are positive.

Definition 4.2.1-2: A non-dominated set, B, is strictly superior to another non-dominated set, A, in terms of *coverage*, denoted by BR_CA, iff all solution points of Set A are contained in a convex cone generated by Set B, while there exists at least one solution point in Set B that is not contained in a convex cone generated by Set A.

Here the convex cone generated by a solution set $A=\{\mathbf{a}_1, \mathbf{a}_2, \ldots\}$ is defined as all nonnegative linear combinations of $\mathbf{a}_i$'s, i.e., $\{\mathbf{v} \mid \mathbf{v} = \Sigma w_i \mathbf{a}_i \; ; \quad w_i \geq 0\}$. The shaded area in Figure 4.2(b) demonstrates the convex cone generated by Set B. This cone clearly contains all solution points of Set A. In contrast, the convex cone of Set A (Figure 4.2(a)) does not include all solution points of Set B. Thus, according to

Definition 4.2.1-2, we have: BR_CA. Finally, note that in this definition it is assumed that all objectives are to be maximized. Also, the nadir point is assumed to be located at the origin of the Cartesian objective space. If these assumptions do not hold for a given problem, one could always transform the objectives to meet these assumptions.

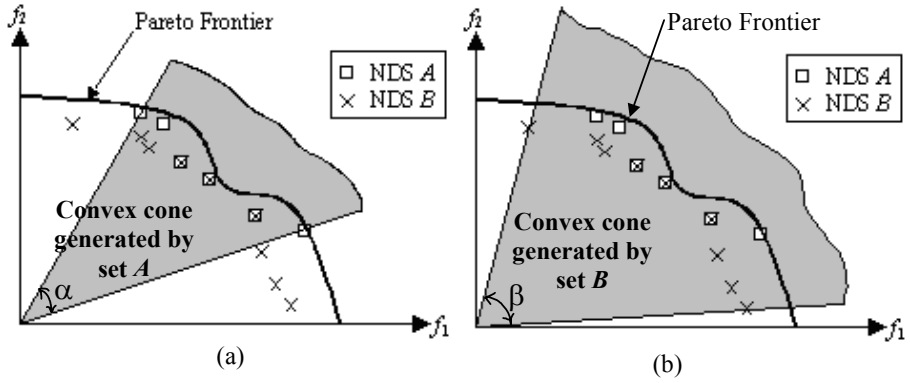


Figure 4.2**:** Convex cones generated by: (a) NDS *A*; and (b) NDS *B*. According to Definition 3: BR_CA.

As expected, the excellence relation for this example establishes a strict partial order in *U*, i.e., objectively compares some pairs of NDS's in terms of coverage.

Note that the first step in a comparison study of NDS's is to determine different aspects of quality that are of interest to the decision-maker (e.g., Pareto optimality, coverage, diversity, and so on). Then, we collect all pairs of NDS's that are objectively comparable with respect to any of these quality aspects. These collections establish strict partial orders in *U* that we refer to as excellence relations. However, not all solution sets are comparable using these relations, and one must formulate quality indexes that enable an exhaustive comparison of all non-dominated solution sets. Each

binary quality index constructs a *total order* that quantitatively compares all NDS's pairs in U.

The correspondence of these quality indexes (i.e., total order) with excellence relations (i.e., partial orders) is the subject of the next section. We prove two key lemmas that will be used later in Section 4 to investigate the properties of a minimal set of quality indexes.

4.2.2 Compatibility and Concordance

Suppose Q is a symmetric and homogeneous binary quality index that compares any two NDS's in terms of a certain aspect of quality. If R is an excellence relation that addresses the same aspect of quality, it is natural to expect Q to be compatible with R, as defined formally in the following. (This is very similar to the definition of compatibility with outperformance relation given by Hansen and Jaszkiewicz (1998); tailored for symmetric indexes, and generalized for excellence relations.)

Definition 4.2.2-1: A symmetric homogeneous binary index, Q, is *compatible* with an excellence relation, R, iff: for any pair of non-dominated sets A and B such that ARB, we also have $Q(A,B) > 0$, which implies set A has a better quality than set B. (The compatibility of index Q with relation R is denoted as $Q \sim R$.)

Knowles (2002) studied the compatibility of several unary and binary quality indexes with respect to an outperformance relation. The same study can be carried out to determine the compatibility of those indexes with respect to any other excellence

relations, such as the coverage relation, R_C. Obviously, if an index is not intended to compare two NDS's in terms of a certain aspect of quality (e.g., a diversity assessment index is not intended to account for closeness to the Pareto frontier) it does not need to be compatible with that excellence relation. In fact, we will prove that each quality index in a minimal set must be compatible with one and only one excellence relation. Nevertheless, this compatibility is dependant on the definition of the excellence relation itself.

Before formally defining minimal sets of quality indexes and their correspondence with excellence relations, in the following we introduce the notion of *concordance* among excellence relations.

Definition 4.2.2-2: Two excellence relations R and R' are *concordant* iff for each $(A, B) \in \Lambda_R \cap \Lambda_{R'}$ such that ARB, we also have $AR'B$.

Concordance basically implies that the two excellence relations *cannot* work against each other. Put another way, R and R' are concordant iff there do not exist two non-dominated sets A and B such that: ARB and $BR'A$. If two excellence relations are referring to different aspects of quality (e.g., diversity and Pareto optimality), there are always examples of NDS's that are better with respect to one aspect of quality and worse with respect to another, and therefore, those relations are not concordant (or are *non-concordant*). In contrast, if being better in terms of one relation always implies better with respect to another, then it implies that the two relations have essentially the same nature (i.e., refer to the same aspect or notion of quality) and thus are concordant. The two excellence relations of Definitions 4.1.3-2 and 4.2.1-2, i.e., R_C and R_O, are

non-concordant because set A in Figures 4.1 and 4.2 is better than set B in terms of the outperformance relation (AR_OB), but worse in terms of coverage (BR_CA). Moreover, from the above definition two relations R and R', such that $R \subset R'$, are always concordant. Therefore, the family of outperformance relations defined by Hansen and Jaszkiewicz (1998) are concordant because: complete outperfromance is a subset of strong outperformance, which in turn is a subset of weak outperformance. Therefore, although these relations are different, they are concordant according to Definition 4.2.2-2.

Concordance is a very strong assumption in the sense that if $(A,B) \in U \mathbf{x} U$ is comparable via two given concordant excellence relations, the outcome of the comparison from the first relation is always the same as that of the second one. On the other hand, non-concordance is a weak assumption in the sense that: two excellence relations are non-concordant even if there exists only one pair of non-dominated solution sets, (A,B), such that A is better than B with respect to one relation and worse with respect to another.

4.2.3 Compatibility Lemma

In the following, we prove an important lemma for non-concordant relations that will be used later in this chapter as a basis for the Minimality Theorem.

Lemma 4.2.3: There does not exist a symmetric and homogeneous quality index that is compatible with two (or more) non-concordant excellence relations.

Proof. For the sake of contradiction, suppose there exists a quality index, Q, which is compatible with two non-concordant excellence relations, namely R and R'. Since R and R' are non-concordant, there exists a pair of non-dominated sets, namely $A,B \in U$, $(A \neq B)$, such that ARB and $BR'A$. But since Q is compatible with R, from ARB we conclude $Q(A,B) > 0$. Similarly, Q is compatible with R' and from $BR'A$ we have $Q(B,A) > 0$, which is a contradiction because $Q(A,B) = -Q(B,A)$. □

In fact, this lemma is somewhat intuitive from the definition of concordance and compatibility: an index cannot simultaneously be compatible with two excellence relations that work against each other. For example a symmetric and homogeneous coverage index, which is compatible with R_C, is necessarily incompatible with R_O (i.e., recall that R_O and R_C are non-concordant according to Definition 4.2.2-2 and Figures 4.1 and 4.2). The above lemma indicates that there must be at least one separate index in a minimal set to individually address each aspect of quality, e.g., at least one index compatible with diversity, another one compatible with Pareto optimality, and so on.

Next, we show that any two given indexes that address the same aspect of quality are necessarily correlated.

4.2.4 Correlation Lemma

In this section, we prove another important lemma for non-correlated binary indexes that will be used in Section 4.2.5 as a basis for Minimality Theorem.

Lemma 4.2.4: If two symmetric homogeneous indexes are both compatible with an excellence relation, R, they are positively correlated within Λ_R.

Proof. Assume two symmetric and homogeneous indexes, Q and Q', are both compatible with R. Then the covariance of Q and Q' within Λ_R can be written as:

$$\text{Cov}\,[Q(A,B), Q'(A,B)] = \langle Q(A,B)Q'(A,B)\rangle - \langle Q(A,B)\rangle \langle Q'(A,B)\rangle \quad ; (A,B) \in \Lambda_R$$

where the expected value of Q within Λ_R, i.e., $\langle Q(A,B)\rangle$, is zero, because Q is symmetric, and therefore, for each $(A,B) \in \Lambda_R$, we also have $(B,A) \in \Lambda_R$, and $Q(A,B) = -Q(B,A)$. Similarly: $\langle Q'(A,B)\rangle = 0$. On the other hand, Q and Q' are both compatible with R, and therefore, for each $(A,B) \in \Lambda_R$, $Q(A,B)$ and $Q'(A,B)$ have the same sign (both negative or both positive). Thus, $\langle Q(A,B)Q'(A,B)\rangle$ is strictly positive, and the Lemma follows. □

Note that being 'positively correlated' is a necessary and not sufficient condition for 'compatibility with the same relation', i.e., two indexes that are not compatible with the same relation are not guaranteed to be uncorrelated. Here we denote the standard deviation of two indexes, Q and Q', by σ and σ' respectively. Then the correlation of two indexes are defined by:

$$\rho = \text{Corr}(Q,Q') = \text{Cov}(Q,Q')/(\sigma\sigma') \tag{4.2}$$

where $-1 \le \rho \le 1$. Note that $\rho = \pm 1$ implies that the two indexes are strictly correlated, i.e., one index is a linear function of the other. In that case, using both indexes to assess the quality of a solution set is redundant. On the other hand, a correlation of $\rho = 0$ shows that the two indexes are absolutely uncorrelated, which is a

very desired property. As mentioned before, we expect the indexes in a minimal set to be uncorrelated.

In the following section, we take advantage of the above Lemmas to investigate the properties of minimal sets of quality indexes.

4.2.5 Minimality Theorem

In this section, we formally state the minimality conditions for a set of binary quality indexes.

Definition 4.2.5-1: A set of quality indexes, namely $\mathbf{\Gamma}$, is said to be *minimal* with respect to a given set of non-concordant excellence relations, $\mathbf{\Phi}$, iff:

1- Each quality index, $Q \in \mathbf{\Gamma}$, is compatible with at least one excellence relation in $\mathbf{\Phi}$. Formally, $\forall Q \in \mathbf{\Gamma} : \exists R \in \mathbf{\Phi}$ such that $Q \sim R$.

2- For each excellence relation in $\mathbf{\Phi}$, there is at least one compatible quality index in $\mathbf{\Gamma}$. Formally, $\forall R \in \mathbf{\Phi} : \exists Q \in \mathbf{\Gamma}$ such that $Q \sim R$.

3- There is minimum (or no) correlation among quality indexes of $\mathbf{\Gamma}$ within the partially ordered domain of excellence relations.

The first property rejects unnecessary indexes that are not compatible with any of the excellence relations. The second property guarantees that $\mathbf{\Gamma}$ is exhaustive, in the sense that it addresses all aspects of quality that are of any interest to the decision-maker (i.e., expressed via excellence relations in $\mathbf{\Phi}$). The last property eliminates or

minimizes the redundancy within the set, i.e., the selected quality indexes should have minimum or no correlation. From this definition and Lemmas 4.2.3 and 4.2.4 we observe the following.

Minimality Theorem: Given a set of *n* non-concordant excellence relations, $\mathbf{\Phi}$, a corresponding minimal set of symmetric and homogeneous indexes, $\mathbf{\Gamma}$, contains *n* and only *n* quality indexes. (Also, there is a one-to-one correspondence between $\mathbf{\Gamma}$ and $\mathbf{\Phi}$.)

Proof. From Lemma 4.2.3, an index in $\mathbf{\Gamma}$ cannot be compatible with more than one excellence relation in $\mathbf{\Phi}$ (because the excellence relations in $\mathbf{\Phi}$ are non-concordant). Therefore, following the first property of Definition 4.2.5-1, each index is compatible with exactly one excellence relation. Also, Lemma 4.2.4 indicates that two uncorrelated indexes cannot be compatible with the same excellence relation (because otherwise they would be positively correlated according to this lemma). Therefore, following the second property in Definition 4.2.5-1, there is a one-to-one correspondence between $\mathbf{\Gamma}$ and $\mathbf{\Phi}$. □

Minimality Theorem suggests a recipe with a set of steps to be followed for the selection of a minimal set of quality indexes, as given next.

Step 1. The general aspects of quality (or goodness) of solution set that are of interest to the decision-maker are determined (e.g., closeness to the Pareto frontier, coverage, diversity, etc.)

Step 2. An excellence relation is considered that accounts for all pairs of NDS's that are objectively comparable with respect to the given aspect of quality.

Step 3. Suppose $\mathbf{\Phi}$ consists of n excellence relations. To construct a minimal set, select one and only one quality index, compatible with each excellence relation in $\mathbf{\Phi}$ (recall Minimality Theorem).

For instance, outperformance relation addresses the closeness to the Pareto frontier; coverage relation of Definition 4.2.1-2 addresses coverage of the set, and so on. If the aspects of quality are defined properly in Step 1, these excellence relations are non-concordant (because if they are referring to different aspects of quality, there exists a non-dominated sets that is better than another set with respect to one excellence relation while worse with respect to another). These non-concordant excellence relations constitute $\mathbf{\Phi}$. The minimal set therefore, contains exactly 2 quality indexes, each compatible with one excellence relation. Note that since Lemma 4.2.4 provides only a necessary condition for being uncorrelated, establishing a one-to-one compatibility correspondence does not guarantee a minimal set. However, it rules out many of non-minimal collections of indexes and significantly narrows the search for minimum correlations. The result is a set of size n of performance assessment indexes, i.e., $\mathbf{\Gamma}$.

4.2.6 Discussion

$\mathbf{\Gamma}$ constructs exactly n total orders in U, and can be used to compare any two given NDS's in terms of the quality aspects expressed in Step 1, and formulated as excellence relations (i.e., strict partial orders) in Step 2. It exhaustively and distinctly covers all desired aspects of quality, without unnecessary correlation among indexes. According to the Minimality Theorem, collections of more than n indexes are

necessarily correlated, while less than n indexes cannot distinctly address all desired aspects of quality. Note that the suggested guideline is only an abstraction of the notion of quality indexes and their desired properties, and therefore, it does not define or formulate new indexes by itself.

Finally, although the theoretical framework of this chapter provides an approach for an objective selection of minimal sets of binary quality indexes, its real-world application may be hindered by several factors:

- A decision-maker may not be able to state his/her idea of 'quality of a solution set' explicitly in the form of excellence relations.
- Even if the decision-maker is able to state a set of excellence relations, $\mathbf{\Phi}$, as a basis for quality assessment, it is not always possible to find a corresponding set of minimal binary and symmetric quality indexes, $\mathbf{\Gamma}$, such that each index in $\mathbf{\Gamma}$ is compatible with an excellence relation in $\mathbf{\Phi}$.

In the example of the next section, we further discuss the difficulties involved with finding a minimal set, and suggest ways to address them.

4.3 EXAMPLE

In the following example, we investigate the practical issues involved with finding a minimal set of quality indexes.

4.3.1 Two-Relation Example

Suppose the decision-maker desires only two aspects of quality: 1) Pareto optimality, and 2) Coverage. Therefore, $\mathbf{\Phi}$ consists of exactly two relations: $\mathbf{\Phi} = \{R_O,$

$R_C\}$ (recall that these two relations are non-concordant). A minimal set of size two of quality indexes, $\mathbf{\Gamma}=\{Q_1, Q_2\}$, should then be selected from the pool of existing indexes such that $Q_1 \sim R_O$, and $Q_2 \sim R_C$. No other combination can address both of these quality aspects without redundancy. According to the Minimality Theorem, the same argument holds for any number of non-concordant excellence relations in $\mathbf{\Phi}$. However, as discussed in the following, finding $\mathbf{\Gamma}=\{Q_1, Q_2\}$ such that the Minimality Theorem holds is not always a trivial task. For example consider a case where $\mathbf{\Phi}$ consists of the two non-concordant excellence relations of outperformance and coverage of Section 4.2.1, i.e., $\mathbf{\Phi} = \{R_O, R_C\}$. Table 4.1 shows the compatibility of several quality indexes in the literature with these relations.

	Strong Outperformance Relation (R_O) (Definition 4.1.3-2)	Coverage Relation (R_C) (Definition 4.2.1-2)
C index	C	I
Inferiority Index (*InfI*) (Farhang-Mehr et al. 2001)	C	I
k-th Objective Pareto Spread (OS_k) (Wu and Azarm, 2001)	I	I
Entropy Index [Chapter 5]	I	I

Table 4.1: Compatibility of quality indexes with $\Phi = \{R_O, R_C\}$
('C' and 'I' indicate compatible and incompatible, respectively.)

Note that Wu and Azarm's *k*-th Objective Pareto Spread (OS_k) is a unary index. So, here we revised this index, i.e., computed the difference in the value of OS_k between two given NDS's to create a binary index: $OS_k(A,B)=OS_k(A)-OS_k(B)$, which constructs a cardinal total order in U. Table 4.1 shows that only a portion of the examined indexes are compatible with R_O, while we were not able to find any quality index in the literature to be compatible with R_c. Indeed, as shown by Knowles and

Corne (2002), a relatively small portion of the existing quality indexes is compatible with one of the outperformance relations introduced by Hansen and Jaszkiewicz (1998). In fact, obtaining a set of indexes to address all aspects of quality may become very difficult in practice because such indexes may not actually exist in the literature. Therefore, one may seek alternative excellence relations, or modify existing indexes to satisfy the Minimality Theorem -- as discussed in the following two sections.

4.3.2 Modifying Quality Indexes (Γ)

As mentioned in Section 4.3.1, constructing a one-to-one correspondence between excellence relations and quality indexes is not always a trivial task. In a situation like this, one may try to modify previous indexes or create new ones to match the compatibility criterion with a given excellence relation. In the following, for example, we propose a new binary quality index to quantify the difference between the spans of two non-dominated sets as a measure of extent of coverage. Later we show that this index is compatible with R_c.

Definition 4.3.2-1: *Binary coverage index*, denoted by $Q_c(A,B)$, is defined as:

$Q_c(A,B)$= inf $\{(\mathbf{b}_i.\mathbf{b}_j)/(||\mathbf{b}_i||\ ||\mathbf{b}_j||)$ s.t. $\mathbf{b}_i,\mathbf{b}_j \in \mathbb{B}\}$- inf $\{(\mathbf{a}_k.\mathbf{a}_l)/(||\mathbf{a}_k||\ ||\mathbf{a}_l||)$ s.t. $a_k,a_l \in \mathbb{A}\}$

where $\mathbb{A}$ and $\mathbb{B}$ are the convex cones generated by solution sets: $A=\{\mathbf{a}_1, \mathbf{a}_2, \ldots\}$ and $B=\{\mathbf{b}_1, \mathbf{b}_2, \ldots\}$. The term: inf $\{(\mathbf{a}_k.\mathbf{a}_l)/(||\mathbf{a}_k||\ ||\mathbf{a}_l||)$ s.t. $a_k,a_l \in \mathbb{A}\}$ measures the cosine of the largest possible angle between two vectors in the convex cone generated by the solution set A. This for example corresponds to $cos(\alpha)$ in Figure 4.2, and $Q_c(A,B)$= $cos(\beta)$ - $cos(\alpha) < 0$. We use this as a measure of maximum span of the solution sets on

the Pareto frontier. Q_c is compatible with R_c because: if BR_cA, we have $\mathbb{A} \subset \mathbb{B}$, and therefore, the second term in the above equation is greater than the first term. Therefore, $Q_c(A,B)<0$. Similarly, if AR_cB we obtain $Q_c(A,B)>0$, and compatibility follows.

From Table 4.1, Definition 4.3.2-1, and Minimality Theorem we observe that $\mathbf{\Gamma} = \{InfI; Q_c\}$ is a candidate minimal set of binary quality indexes with respect to $\mathbf{\Phi} = \{R_O, R_C\}$.

4.3.3 Modifying Excellence Relations ($\mathbf{\Phi}$)

Alternatively, one may go back to the initial set of the excellence relations, $\mathbf{\Phi}$, and try to modify these relations such that a corresponding minimal set of quality indexes can be found. For example, if the decision-maker modifies the definition of the coverage excellence relation (Definition 4.2.1-2), $\mathbf{\Gamma}$ may or may not remain minimal. In the following, for example, we introduce a modified definition for coverage excellence relation that makes Wu and Azarm's OS_k index compatible with it.

Definition 4.3.3-1: (Modified Coverage Relation; Compare to Definition 4.2.1-2). In a normalized multi-objective maximization, a non-dominated set $B=\{\mathbf{b}_1, \mathbf{b}_2, \ldots\}$ is strictly superior to another non-dominated set $A=\{\mathbf{a}_1, \mathbf{a}_2, \ldots\}$ in terms of a *modified coverage*, denoted by BR'_CA, iff: $\max_{i,j}(b_i^k - b_j^k)$ is strictly greater than $\max_{i,j}(a_i^k - a_j^k)$ for all k's (where b_i^k refers to the k-th objective value of the i-th solution in B).

OS_k is compatible with R'_C, and therefore, a combination of this quality index and Inferiority Index (*InfI*) -- i.e. $\mathbf{\Gamma'} = \{InfI, OS_k\}$ -- is a candidate minimal set with respect to $\mathbf{\Phi'} = \{R_O, R'_C\}$. In contrast, $\mathbf{\Gamma} = \{InfI, Q_c\}$ which is minimal with respect to $\mathbf{\Phi} = \{R_O, R_C\}$, is not minimal with respect to $\mathbf{\Phi'}$, because Q_c is not compatible with R'_C (See Definition 4.3.2-1 for Q_C). Note that the process of defining and redefining $\mathbf{\Phi}$ becomes increasingly difficult as more excellence relations are included. Nonetheless, it provides a formal platform and an objective starting point for selection of binary quality indexes.

4.3.4 Checking Minimum-Correlation Condition

Table 4.2 summarizes the compatibility of the above-mentioned indexes with R_O, R_C, and R'_C.

	Strong Outperformance Relation (R_O) (Definition 4.1.3-2)	Coverage Relation (R_C) (Definition 4.2.1-2)	Modified Coverage Relation (R'_C) (Definition 4.3.3-1)
C index	C	I	I
Inferiority Index (*InfI*) (Farhang-Mehr et al. 2001)	C	I	I
k-th Objective Pareto Spread (OS_k) (Wu and Azarm, 2001)	I	I	C
Binary Coverage Index (Definition 4.3.2-1)	I	C	I

Table 4.2: Compatibility of quality indexes with R_O, R_C, and R'_C
('C' and 'I' indicate compatible and incompatible, respectively.)

From the Compatibility Lemma (Lemma 4.2.3), each quality index can be compatible with at most one relation (since R_O, R_C, and R'_C are non-concordant relations). This can be easily observed from this table, and is true in general for all quality indexes. Furthermore, from Minimality Theorem, there are four different

combination of quality indexes that are minimal with respect to either $\mathbf{\Phi} = \{R_O, R_C\}$ or $\mathbf{\Phi} = \{R_O, R'_C\}$, as shown in Table 4.3.

Index 1	Index 2	Correlation (ρ)
C index	k-th Objective Pareto Spread (OS_k)	0.03
C index	Binary Coverage Index (Definition 4.3.2-1)	- 0.08
Inferiority Index (*InfI*)	k-th Objective Pareto Spread (OS_k)	-0.02
Inferiority Index (*InfI*)	Binary Coverage Index (Definition 4.3.2-1)	0.09

Table 4.3: Minimal sets and the corresponding correlation

Recall that Minimality Theorem provides only a necessary condition for uncorrelated sets, thus one must also check the correlation of each combination. In Table 4.3 therefore, we also estimated the correlation of different combinations of indexes from the following experiment: For each combination of indexes, namely Q and Q', we randomly selected 1000 pairs of non-dominated sets of size 5 such that they are comparable with respect to $\mathbf{\Phi}$ (i.e., we generated pairs of non-dominated sets and checked whether they are comparable with respect to both relations in $\mathbf{\Phi}$. If yes, we kept the pair, otherwise we replaced it with another randomly generated pair.) For this example, we selected a 2-dimensional solution space and the solution points are constrained in a rectangle between (0,0) and (1,1). The correlation of any given pair of indexes can be easily obtained from the following:

$$\text{Cov}\,[Q(A,B), Q'(A,B)] = \langle Q(A,B)Q'(A,B)\rangle - \langle Q(A,B)\rangle \langle Q'(A,B)\rangle \qquad (4.3)$$

where $<Q(A,B)>$ and $<Q'(A,B)>$ are approximately zero (because of symmetry). So the correlation can be estimated as:

$$\rho = \text{Corr}(Q,Q') = <Q(A,B)Q'(A,B)> / (\sigma\sigma') \quad (4.4)$$

For example, the following is the correlation between *C* index and Inferiority Index:

Corr (*C* index, Inferiority Index) = 0.86

Clearly, these two indexes are highly correlated. This is consistent with Compatibility Lemma, i.e., these indexes are both compatible with outperformance relation and therefore are expected to be positively correlated. Table 4.3 lists the correlations of combinations of quality indexes that follow the Minimality Theorem. Although Minimality Theorem provides only a necessary condition for zero-correlation, these pairs of indexes have relatively negligible correlation values.

4.4 CONCLUDING REMARKS

In this chapter, we presented a theoretical framework for selection of a minimal set of quality indexes based on inputs provided by a decision maker. These indexes can account for all desired aspects of quality in non-dominated solution sets exhaustively and without redundancy. In this framework, once the decision-maker's desired aspects of performance are determined, it is necessary to find all pairs of non-dominated sets that are objectively comparable. This in turn constructs partial orders in the set of all possible non-dominated sets, referred to as excellence relations in this dissertation. We proved that there is a one-to-one compatibility correspondence between these excellence relations (partial orders) and a minimal set of quality indexes

(total orders), i.e., for each excellence relation there is one and only one compatible quality index in a minimal set. This important result (referred to as the Minimality Theorem) helps the decision-maker select a minimal set among the existing quality indexes in the literature, and thus, enables a quantitative and objective comparison of the solution sets obtained from different MOGAs.

The next chapter is devoted to developing a new quality index -- referred to as entropy index. This new quality index is based on the concept of information-theoretic entropy (Section 2.1.5,) and provides a measure for diversity of NDS's.

CHAPTER 5

ENTROPY INDEX

This chapter presents a new quality index that can be used to measure the diversity of solution points. The new index -- hereafter referred to as the *entropy* of a solution set -- is based on the notion of information-theoretic entropy and encapsulates different aspects of the distribution quality such as uniformity of distribution, coverage (i.e., portion of the Pareto frontier covered by the observed solution set), number of solution points and clustering. This entropy scalar essentially reflects whether a set of solutions provides a full representation of the Pareto frontier. This is of paramount interest in comparison of different MOGAs where final solution sets are likely to cluster, leaving the rest of the Pareto frontier empty or sparsely populated. The proposed entropy index has a linear computational complexity and thus can be easily computed throughout the evolution process to monitor and compare the diversity of solution sets.

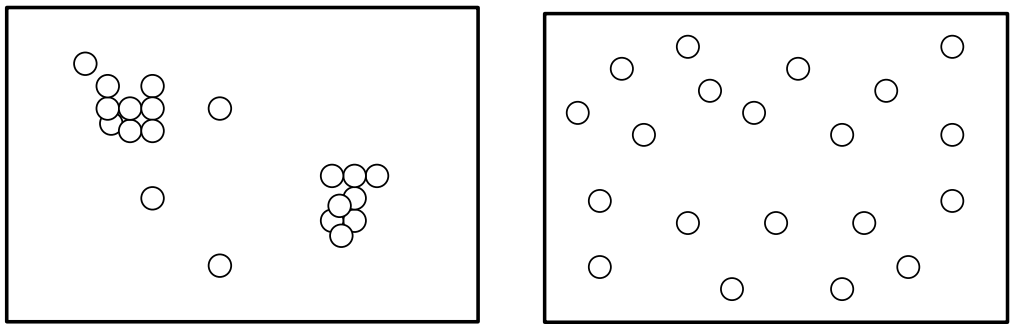
This chapter is organized as follows: Section 5.1 provides a background on intuitive interpretation and visualization of diversity in a solution set. The notions of influence and density functions are also defined. Section 5.2 takes advantage of formal similarities between information-theoretic entropy of Section 2.1 and flatness of density function; and entropy index is formulated accordingly. Finally, Section 5.3 presents a simple test case to investigate the behavior of entropy index during optimization of a problem using MOGA.

5.1 BACKGROUND AND PREVIOUS WORK

The following subsections discuss uniformity of solution sets, and define a few terms that are used in this chapter.

5.1.1 Visualization of Diversity in a Solution Set

Figure 5.1 shows the distribution of two sets of the same number of feasible solution points in a rectangular region that represents the best estimate of a feasible design space. Each of these solution points corresponds to a feasible design alternative and the rectangular region is chosen such that it encloses the entire feasible design space. Clearly, no feasible design alternative is allowed to be outside this rectangular domain, yet there might be points inside the rectangle that are infeasible. For now, assume that none of the objectives are being optimized and therefore all of these feasible solution points are equally desirable. Our goal is to determine how good each of these sets of alternatives, i.e., the feasible solution set shown in Figure 5.1(a) or the set in Figure 5.1(b) represents the design space. In other words, which one of these sets gives a wider variety of options to the decision maker.



(a) (b)

Figure 5.1: Two different feasible solution sets

Intuitively, one may argue that the solutions in Figure 1(a) are grouped into two clusters, as shown in Figure 5.2(a). There are several design alternatives (or solution points) in each cluster that are not significantly different from one another, while there are areas in the domain where there are few or no solution points. In contrast, the solution points in Figure 5.2(b) are spread uniformly and cover almost the entire domain. It is clear that such a uniform distribution provides the decision maker with a larger variety of design alternatives, and consequently, it is preferred over the first set.

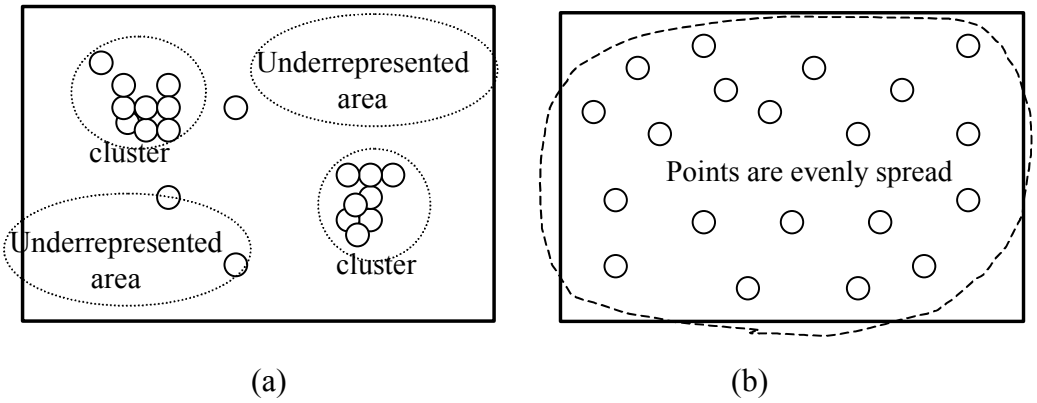


Figure 5.2:Visual assessment of solution sets

However, as the number of alternatives increases, making visual judgment becomes increasingly difficult. For problems with higher dimensions, relying upon the decision maker's intuition is either impossible or quite misleading.

5.1.2 Influence and Density Functions

The basic idea behind our proposed entropy index is that each solution point provides some information about its neighborhood in the feasible space that can be modeled as a function, called an influence function. The notion of influence function

has been used extensively in data-mining to identify data clusters in huge databases (e.g., Fukunaga and Hostler, 1975; Hinneburg and Keim, 1998).

Before introducing the entropy index, a formal definition of influence and density functions is given next.

Influence Function: We denote the *m*-dimensional feasible domain in the objective space of a multiobjective optimization problem by $F^m \subseteq R^m$. The influence function of the *i*-th solution point in this space is then defined as a function (or mapping), $\Omega_i : F^m \to R$, such that Ω_i is a decreasing function of the distance to the *i*-th solution point. Different function forms, e.g., parabolic, square wave or Gaussian, can be used for an influence function. The influence function of the *i*-th solution point is maximum at that point and decreases gradually with the distance from that point. In this chapter, a Gaussian influence function is assumed.

Density Function: The density function at each point of the feasible objective space is defined as the aggregation of the influence functions from all solution points. Assuming that the solution set contains *N* solution points ($N > 0$), the density function at an arbitrary point, *y*, in the feasible objective space F^m can be obtained as follows:

$$D(y) = \sum_{i=1}^{N} \Omega_i \left(r_{i \to y} \right) \tag{5.1}$$

where $r_{i \to y}$ is a scalar that represents the Euclidean distance of the point *y* and the *i*-th solution point, and $\Omega_i(.)$ is the influence function of the *i*-th solution point. Figure 5.3 illustrates the influence function of a few points in a one-dimensional feasible domain (or line segment). The resulting density function is obtained and graphed by simply

aggregating the influence functions of all solution points along the feasible line segment.

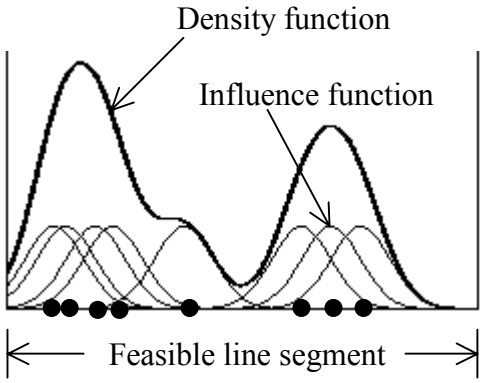


Figure 5.3: A set of solution points in a one-dimensional feasible space with the corresponding influence and density functions

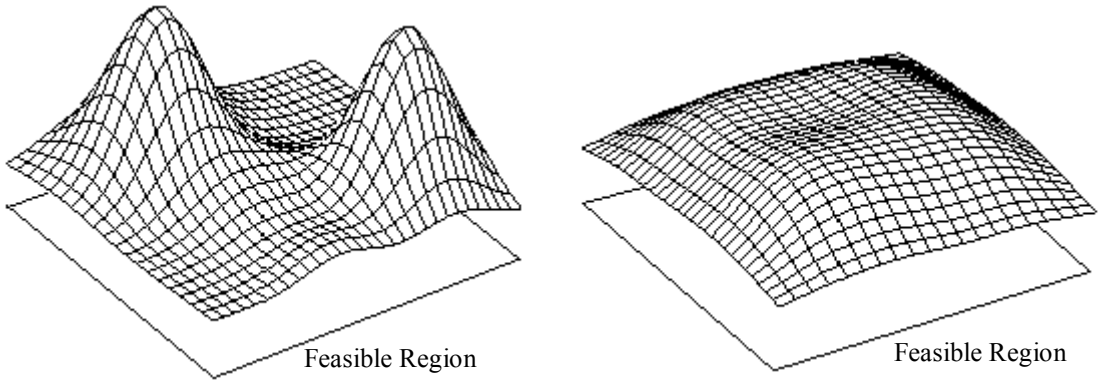
Assuming that the chosen influence function is continuous and differentiable, the corresponding density function, as defined above, is also continuous and differentiable at any point in the feasible space. This smooth curve (or hyper-surface in more than two dimensions), hereafter referred to as *density hyper-surface*, consists of peaks and valleys that can be easily identified. As shown in Figure 5.3, the peaks correspond to dense areas with lots of points nearby and the valleys correspond to sparse areas with a few adjacent points.

5.2 Entropy Index

In the following, entropy index is defined and related to the distribution quality of a solution set.

5.2.1 Flatness of Density Hypersurface

Figures 5.4(a) and (b) illustrate the density surface of the two solution sets of Figures 5.1(a) and (b), respectively. There are relatively sharp peaks and deep valleys in Figure 5.4(a) while there is no significant peak or valley in Figure 5.4(b). This confirms our previous observation that the second solution set is spread more evenly over the feasible region as compared to the first set. A good distribution of solutions yields a relatively even surface without significant peaks and valleys. In contrast, if the solution points are grouped into one or more clusters and the rest of the feasible region is populated sparsely they do not fully represent the entire domain. The density surface of such solution set contains sharp peaks and deep valleys.



(a) (b)

Figure 5.4 (a) Density hyper-surface of the solution set in Figure 1(a),
(b) Density hyper-surface of the solution set in Figure 1(b)

So far, we have established a relationship between the goodness of a distribution of a solution set and the flatness of its corresponding density surface. In the following section, an entropy index is defined to measure this flatness.

5.2.2 Entropy of Density Hypersurface

In Section 2.1.5, Shannon entropy of a given probability distribution, π, was defined as:

$$H = - E_U [\log \pi \ ] \tag{5.2}$$

Now, assume a discrete stochastic process with n possible outcomes where the probability of the i-th outcome is p_i.

$$\mathbf{P} = [p_1, \ldots, p_i, \ldots, p_n] \ ; \qquad \sum_{i=1}^{n} p_i = 1 \ ; \qquad p_i \geq 0 \tag{5.3}$$

The entropy of this process can be written as:

$$H(\mathbf{P}) = -\sum_{i=1}^{n} p_i ln(p_i) \tag{5.4}$$

where $p_i ln(p_i)$ is assumed zero for $p_i = 0$ (because $p_i ln(p_i) \rightarrow 0$ when $p_i \rightarrow 0$). This function is at its maximum, $H_{max} = ln(n)$, when all probabilities have the same value, and at its minimum of zero when one component of **P** is 1 and the rest of entries are zero. In fact, the Shannon's entropy measures the flatness of **P**, i.e., if the values of the entries in the vector are approximately the same then the entropy is high, but if the values are very different (uneven probability distribution), the corresponding entropy is low. In the previous section, we mentioned that a desirable solution set must have a 'flat' density surface within the feasible domain. To quantify this flatness, one may take advantage of the formal similarities between this problem and the Shannon's entropy, which also measures the flatness of a distribution.

Consider the 2-dimensional objective space of Figure 5.5. In order to apply Equation 5.4, a grid of size $a_1 \times a_2$ is constructed in the feasible domain. The density

function at each cell, D_{ij}, can be obtained from Equation 5.1 assuming that the point y_{ij} is located at the center of the cell, i.e., $D_{ij}=D(y_{ij})$. The quantities a_1 and a_2 are determined such that the size of each cell becomes less than or equal to an indifference region. (Indifference region is defined as the size of a cell in which any two solution points are considered to be the same, or that the decision maker is indifferent to such solutions.) These a_1 and a_2 quantities can be determined subjectively based on the decision-maker's experience or knowledge of similar problems, or objectively based on the available computational power and desired accuracy. Assuming a very small grid size helps improve the accuracy but it also increases the computational burden of calculating the entropy, which in turn might make the quality assessment process very slow or even computationally infeasible. Clearly the appropriate grid size is problem-dependent and differs from one situation to another.

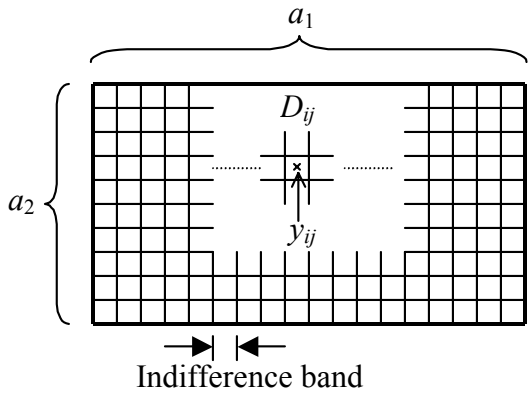


Figure 5.5: The feasible region is divided into a grid of cells

However, since the sum of the entries (i.e., probabilities) in the Shannon's definition of entropy is one, we define a normalized density, ρ_{ij}, as:

$$\rho_{ij} = \frac{D_{ij}}{\sum_{k_1=1}^{a_1}\sum_{k_2=1}^{a_2} D_{k_1 k_2}} \tag{5.5}$$

Note that if there is no solution point (i.e., $N = 0$ in Equation 5.1) the denominator in Equation 5.5 becomes zero. Indeed, the above definition of the normalized density is not well defined for an empty solution set, which is why the solution set is assumed to be non-empty in the definition of the density function. We will assign a value of zero to the entropy of empty solution sets to indicate the worst case, i.e., zero diversity. Now we have:

$$\sum_{k_1=1}^{a_1}\sum_{k_2=1}^{a_2}\rho_{k_1k_2} = 1\,; \qquad \rho_{k_1k_2} \geq 0 \quad \forall k_1, k_2 \tag{5.6}$$

The entropy of such a distribution can then be defined as:

$$H = -\sum_{k_1=1}^{a_1}\sum_{k_2=1}^{a_2}\rho_{k_1k_2} ln\left(\rho_{k_1k_2}\right) \tag{5.7}$$

In general, for an m-dimensional objective space, the feasible region in objective space is divided into $a_1 \text{x} a_2 \text{x} \ldots \text{x} a_m$ cells and the entropy is defined as:

$$H = -\sum_{k_1=1}^{a_1}\sum_{k_2=1}^{a_2}\cdots\sum_{k_m=1}^{a_m}\rho_{k_1k_2\ldots k_m} ln\left(\rho_{k_1k_2\ldots k_m}\right) \tag{5.8}$$

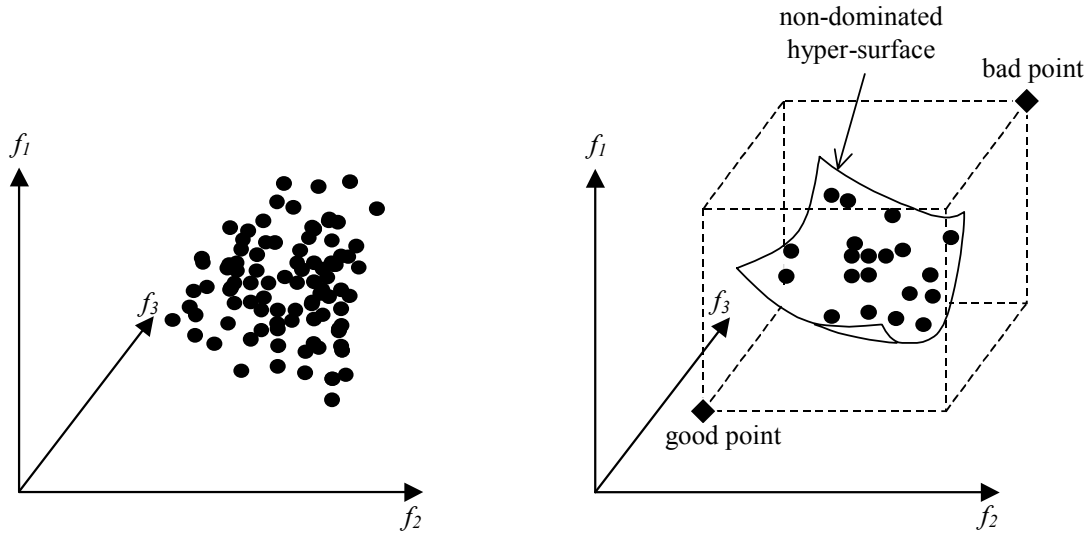
The entropy of the sets in Figure 5.4(a) and (b) can be calculated by constructing a grid of cells and applying Equation 5.8. The entropy of the second set (Figure 5.4(b)) turns out to be larger than that of the first set (Figure 5.4(a)). This indicates a more evenly spread of solution points in the second set as compared to the first one, which confirms our preliminary intuitive judgment and leads to the following conclusion: *a solution set with a higher entropy is spread more evenly throughout the feasible region in objective space and provides a better coverage of the space.* Note that, if the grid consists of N_G cells, the calculation of entropy from Equation 5.8 has a computational complexity of $O(N_G)$.

The formulation of entropy, has found many applications in numerous fields and can be derived based on a variety of assumptions. In this chapter, however, entropy is used to measure the flatness of the density distribution provided by a set of solution points, and hence it is desirable to be maximized (i.e., it corresponds to a uniform distribution of solution points over the feasible domain). Yet, confusion may arise if entropy is considered as a measure of uncertainty in a stochastic event with a given probability distribution of the outcome (for example, the application of entropy in Chapter 2). In that case, the entropy corresponds to the ignorance (of information content) of the distribution, in the sense that a flat probability distribution provides no information about the probable outcome of the event, whereas a sharp peak in the probability distribution function can help capture the outcome with a certain confidence level and thus, is very informative. The entropy of such problems is desirable to be minimized towards zero that corresponds to a well-informed, ideal state for the problem. For example in Chapter 2, by maximizing the 'reduction of entropy' we indeed minimized the posterior entropy. Clearly, this is in contrast to the problem of assessing the distribution quality of a solution set over a given domain where there is no preference toward any particular sub-region and the solutions are desired to be 'spread' as uniformly as possible to provide a diverse set of options to the decision maker. Indeed, Peters (1975) proposed that different names should be given to the entropy H depending on the underlying idea and the nature of the problem. (For different interpretations of the notion of entropy and their implications, see, Jaynes, 1957; and Jessop, 1995.)

Now that an index for the goodness of the spread of points in an *m*-dimensional objective space is formulated, the next issue is whether this index can be applied to capture the spread of a population in an evolutionary multiobjective optimization process. The next section is devoted to this issue.

5.2.3 Projection

Consider a minimization problem with respect to all objectives. Figure 5.6(a) shows a population of design points in a three-dimensional multi-objective evolutionary optimization process. The observed Pareto set is shown separately in Figure 5.6(b) along with the corresponding good and bad points. A surface that passes through all non-dominated solutions is called a 'non-dominated hypersurface'. This hypersurface, which represents the current estimate of the Pareto frontier, approaches the true Pareto frontier as the multiobjective evolutionary algorithm converges. As shown in this figure, a visual detection of clusters on this hypersurface can be fairly difficult even in three dimensions, and virtually impossible for higher dimensions.



(a) (b)

Figure 5.6: (a) A population of solution points (b) A non-dominated hypersurface with good and bad points

To avoid aggregation and comparison of different non-commensurable objectives, the coordinates in the objective space need to be normalized with respect to the good and bad points (see Figure 5.7). In an m-dimensional objective space whose coordinates are denoted by $f_1,\ldots,f_i,\ldots,f_m$, the normalized set of objectives can be obtained as,

$$f_i' = \frac{(f_i - g_i)}{(b_i - g_i)} \; ; \quad i = 1,\ldots, m \tag{5.9}$$

where $G_0 = (g_1,\ldots, g_i,\ldots, g_m)$ and $B_0 = (b_1,\ldots, b_i,\ldots, b_m)$ are the original (not normalized) good and bad points respectively.

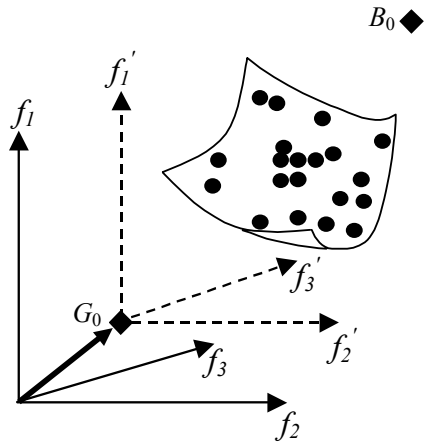


Figure 5.7: The origin of the Cartesian coordinate system is transferred to the good point

In the normalized coordinate system, as shown in Figure 5.8(a), the good and points are G=(0,…, 0) and B=(1,…,1), respectively. The normalized design space is then a hyper-cube, whose two opposite vertices are the points G and B, and whose

edges have unit lengths. In Figure 5.8(a), $\mathbf{u_1}$, $\mathbf{u_2}$ and $\mathbf{u_3}$ are the cartesian unit vectors along f_1', f_2' and f_3', respectively. We define:

Definition: The *projection direction*, denoted by $\mathbf{v_1}$, is defined as a unit vector in the objective space along the *GB* direction. We also refer to the (*m*-1)-dimensional hyper-plane that passes through the origin (*G*) of the normalized objective space (which is *m*-dimensional) and is normal to $\mathbf{v_1}$ as the *projection hyper-plane.*

The projection direction and the corresponding projection plane are shown for the three-dimensional example of Figure 5.8(a). All solution points are projected on this hyper-plane to enable a calculation of entropy. To perform this projection, first we need to construct a rotated coordinate system, one of whose axes is along the projection direction and the rest lie on the projection hyper-plane. This can be done using a procedure known as Gram-Schmidt orthogonalization (Noble and Daniel, 1988) that creates a new set of *m* perpendicular unit vectors in the *m*-dimensional objective space, starting with $\mathbf{v_1}$. The following is a step by step implementation of this procedure for the three-dimensional example of Figure 5.8. After obtaining $\mathbf{v_1}$ (from its definition above), the second unit vector, i.e., $\mathbf{v_2}$, can be constructed from $\mathbf{u_2}$ by subtracting the component of $\mathbf{u_2}$ that lies along $\mathbf{v_1}$ and normalizing the residue:

$$\mathbf{v}_2 = \frac{\left[\mathbf{u}_2 - (\mathbf{u}_2.\mathbf{v}_1)\mathbf{v}_1\right]}{\left\|\mathbf{u}_2 - (\mathbf{u}_2.\mathbf{v}_1)\mathbf{v}_1\right\|} \tag{5.10}$$

It can be shown that $\mathbf{v_2}$ as obtained in Equation 5.10 is a unit vector normal to $\mathbf{v_1}$. In a similar fashion, $\mathbf{v_3}$ is constructed from $\mathbf{u_3}$ by subtracting the components of $\mathbf{u_3}$ along $\mathbf{v_1}$ and $\mathbf{v_2}$ and normalizing the residue:

$$\mathbf{v}_3 = \frac{\left[\mathbf{u}_3 - (\mathbf{u}_3 . \mathbf{v}_1)\mathbf{v}_1 - (\mathbf{u}_3 . \mathbf{v}_2)\mathbf{v}_2\right]}{\left\|\mathbf{u}_3 - (\mathbf{u}_3 . \mathbf{v}_1)\mathbf{v}_1 - (\mathbf{u}_3 . \mathbf{v}_2)\mathbf{v}_2\right\|} \tag{5.11}$$

$\mathbf{v_3}$ as obtained from Equation 5.11 is a unit vector normal to $\mathbf{v_1}$ and $\mathbf{v_2}$ and thus the new set, $\{\mathbf{v_1}, \mathbf{v_2}, \mathbf{v_3}\}$, forms an orthonormal basis that describes a rotated cartesian coordinate system. The first axis, $\mathbf{v_1}$, is the projection vector and points from origin to the bad point while the other two axes lie on the projection plane. If a solution point is represented as a vector in a three dimensional objective space:

$$\mathbf{f}' = (f_1', f_2', f_3') \tag{5.12}$$

then the projection is defined as a mapping from R^3 to R^2, such that:

$$(X, Y) = ((\mathbf{f}' . \mathbf{v_2}) , (\mathbf{f}' . \mathbf{v_3})) \tag{5.13}$$

(X,Y) represents a two-dimensional cartesian coordinate system whose coordinates are along $\mathbf{v_2}$ and $\mathbf{v_3}$ and thus lies on the projection plane.

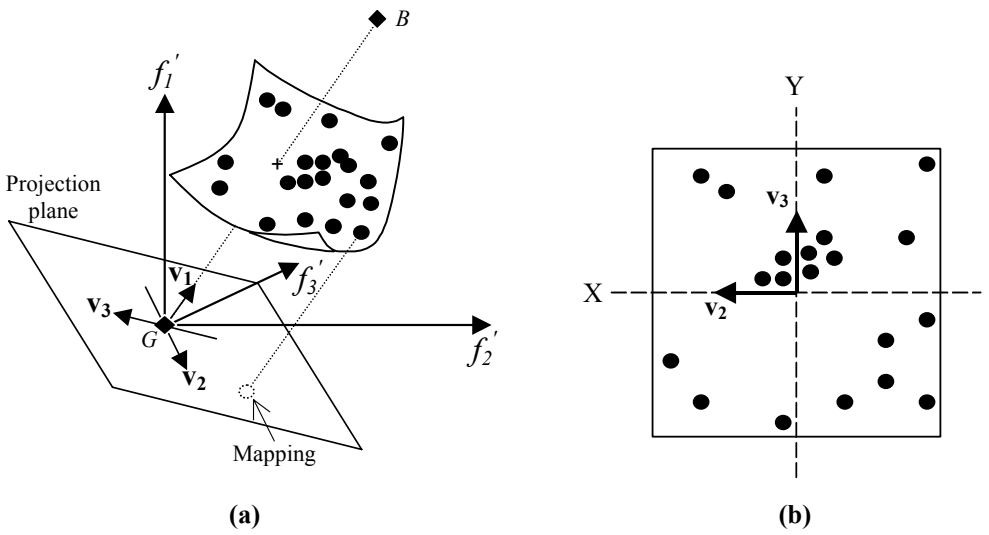


Figure 5.8: The projection mapping of solution points

Figure 5.8(b) illustrates the projection of solution points on the X-Y coordinate system. (Note that $\mathbf{v}_2$ and $\mathbf{v}_3$ lie along X and Y axes, respectively.) The same procedure, as described above, can be applied to a multi-objective optimization problem with any number of objectives to project the observed Pareto set on the projection hyper-plane. The feasible area can be determined by projecting the constraints on the projection plane. However, this can be difficult and computationally prohibitive for complicated constraints. In Figure 5.8(b), the feasible region is estimated by a rectangle, using the maximum and minimum values of X and Y, so that it contains all the projected points. It can also be estimated by projecting the cube between the good and bad points on the X-Y plane. Note that the estimated feasible area must contain all the solution points and remain the same during the comparison study of different solution sets.

5.2.4 Entropy Index of Projected Solution Set

Now, the density surface of the projected set of solution points can be constructed using a Gaussian influence function. But first we need to estimate the standard deviation, σ, of this distribution. Assuming a very small σ causes the influence functions to decay very rapidly (i.e., sharp influence distributions). As a result, even two neighboring points do not significantly influence each other. The density surface is basically a set of N_s sharp peaks (corresponding to N_s solution points) scattered on a flat function with zero elevation. The entropy of such a density surface is not very sensitive to the distribution of the solution points. In contrast, a very large σ implies a flat influence function. The density surface of such a set of solutions is therefore flat no matter where in the design space the solution points are located. Again, the entropy is insensitive to the distribution of solution points. Therefore, the value of σ should be set appropriately. Here we suggest a simple assumption for σ that is proved to be very effective for the example of this chapter. A Gaussian influence function has the general form of:

$$\Omega(r) = \frac{1}{\sigma\sqrt{2\pi}} e^{-r^2/2\sigma^2} \tag{5.14}$$

The standard deviation, σ, of Equation 5.14 is set such that for a solution point that is located at the center of the normalized design space, the 3σ-tail of the Gaussian distribution coincides with the edges of the design space. As mentioned earlier, the normalized design space is basically a hyper-cube between the normalized good and bad points whose edges have unit lengths. Therefore, for a solution point located at the

center of this normalized hyper-cube, we have: $6\sigma=1$ or $\sigma=1/6$. The influence of a central solution point spreads in every direction and almost vanishes at the boundaries. Note that this estimation for the standard deviation is valid only in the normalized coordinate system. Finally, having this estimation of σ, a density hyper-surface can be constructed for the projected solution points on the projection hyper-plane. Now, the entropy index as defined in Equation 5.8 can be used on the projection hyper-plane to measure the flatness of the density hyper-surface. The result will be a scalar that reflects the distribution quality of the solution points. This quality index can be used to observe the evolution of a population during an optimization process or simply to compare the performance of two (or more) different multi-objective optimization algorithms.

5.2.5 Computational Complexity

Clearly, the density function of a solution set can be constructed and updated incrementally by choosing any new feasible solution point and aggregating its influence function with the previous density function. In the real-world design problems in particular, this is extremely helpful, because there are usually new design alternatives generated during or even after the design-generation phase of MOGA. To add these new solution points to the previous set, one can simply update the density surface by aggregating the influence functions of the new solution points and the previous density surface without recalculating the influence of the entire solution set that may need considerable amount of computational effort.

Assume that there are N_s solution points. The influence of a solution point on its neighboring cells is calculated once at the beginning of the process and stored. Since the influence function decays rapidly with the distance, we need to calculate the influence of that solution point only on a limited number of its neighboring cells, N_I cells, for which the influence is significant. The stored influence values for N_I neighboring cells can be used for all other solution points without recalculation (since the same influence function is assumed for the entire solution set). Now the overall density surface can be constructed iteratively: solution points are considered one-by-one and their influence function values on the neighboring cells are added up to the previous density values of those cells (refer to Equation 3). Therefore, the construction of a density surface has a computational complexity of $O(N_s N_I)$. The computational effort increases linearly with the number of solution points in the set. This is a very important advantage, especially when the designer is dealing with a huge database of design alternatives.

5.2.6 Boundary Effect

A closer look at Figures 5.1(b) and 5.4(b) reveals that although the solution set is spread rather uniformly over the feasible region, the density surface has a lower elevation near the boundaries. This is because the density function at point A in Figure 5.9 for example, is the aggregation of influence functions of all solution points in the vicinity located along different directions around that point. In contrast, point B which is located very close to the boundary is influenced only by the solution points in the feasible half-space (located on the right-hand side of the boundary) while there are no feasible solutions in the infeasible half-space. So even for a very evenly distributed

solution set, a point near the boundary is expected to have a lower density function as compared to the one in the middle. It is clear that the boundary effect gradually vanishes as we move farther from the boundary to the middle of the feasible region.

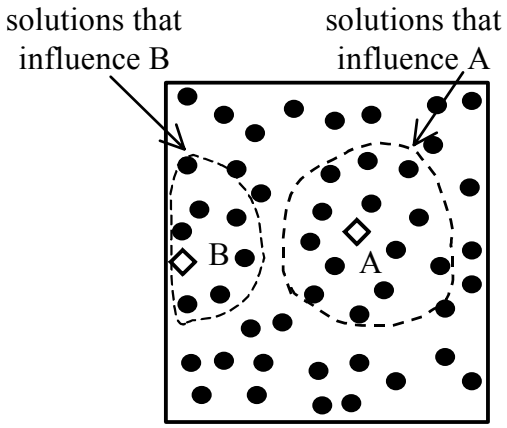


Figure 5.9: The density function values at points A and B are mainly the aggregation of the influence functions of the solutions in the vicinity

Figure 5.10 suggests a modification to the construction of the density surface that compensates for the above mentioned boundary effect. The solution points in the feasible area are mirrored by the boundary to create a set of virtual solution points in the infeasible area. Then the density function at point B is defined to be the aggregation of the influence functions of all real and virtual solution points. This compensates the absence of the solution points in the infeasible half-space by duplicating the solution points in the feasible half-space. Clearly, by moving farther from the boundary, the influence of these virtual solution points decreases (i.e., as the boundary effect diminishes). The same virtual solution points should be constructed for all of the boundaries of the feasible region to compensate for the boundary effect. Note that it is not necessary to mirror all the solution points into the infeasible area and only the solution points near the boundary lines are of interest, since the corresponding

mirrored points heavily influence the density surface near those boundaries. (In Figure 5.10 all of the solution points are mirrored for the purpose of demonstration.)

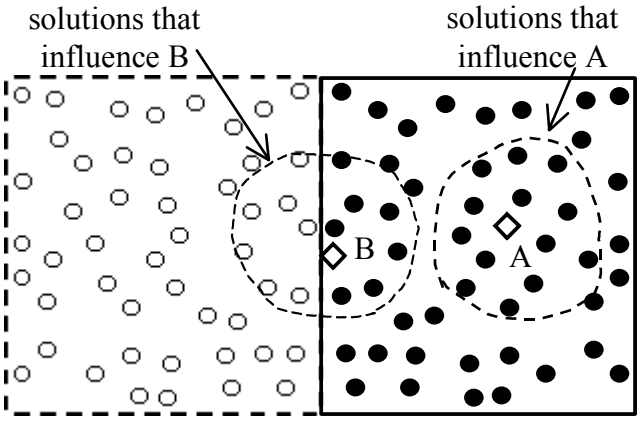


Figure 5.10: Compensation of a boundary effect

Figure 5.11 shows the density surface of Figure 5.4(b) after compensation. As expected from the uniform distribution of this solution set, the constructed density surface has about the same elevation everywhere within the feasible region.

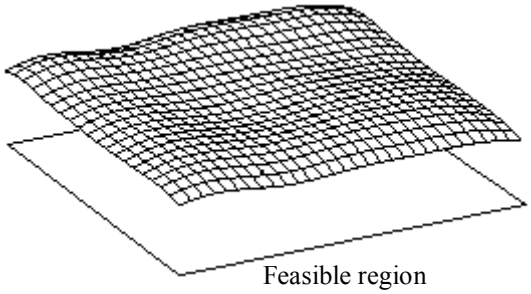


Figure 5.11: The density surface of Figure 5.4(b) after the boundary effect correction

The boundary effect, as explained above has also been observed in the 'maximum-entropy design of experiments' (Chapter 2), where the experiments on the edges of the design space are favored because of a degraded correlation with the

interior solution points. In fact, a local degradation in the density surface near the boundaries is an inevitable result of asymmetry at the boundaries (where half of the neighborhood is feasible and the other half is not) and assuming a decreasing influence function. The above suggested correction should compensate the boundary effect of rectangular feasible domains, but it may become computationally very complex for non-rectangular and complicated shapes. Note that, as mentioned before, we use a rectangular estimation of the feasible region, based on the maximum and minimum values of the projected objective values in the rotated coordinate system (on the projection plane.) However, if an irregular feasible region is assumed, the problem of finding the virtual solution points is very similar to that of finding the virtual images of an object surrounded by several mirrors with irregular shapes, which is computationally very complex. Therefore, here we suggest an alternative strategy, which is much simpler to implement:

1) Only those solution points that are located close to the boundaries are considered for compensation. The boundary effect is significant only in the proximity of the boundaries.
2) For a solution point, namely *P*, which is located close to a boundary, we find the smallest circle tangent to that boundary, with *P* as the center (see Figure 5.12). This can be done by gradually increasing the radius of the circle and checking the feasibility of points on the perimeter until a point on the perimeter becomes infeasible. The virtual image of this point then lies on the extension of

the radius connecting P to the point of tangency (the first infeasible point on the perimeter).

Finally even this simplified approach is difficult to implement in general, especially if the constraints are very expensive to compute. One may prefer not to account for the boundary effect at all in such situations to avoid excessive computational effort. After all, the boundary effect is only a local degradation in the density surface and thus its impact on the entropy is usually small and negligible.

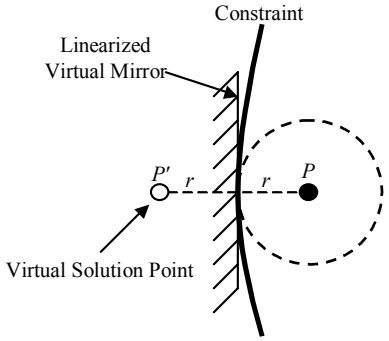


Figure 5.12: Virtual image of a solution point, located close to a boundary

In the next section, the entropy index is applied to an engineering optimization problem to demonstrate its ability to assess the distribution quality of an observed Pareto set. The entropy index, as introduced in this chapter, is applicable not only to a MOGA but to any population-based multi-objective optimization technique that approximates the true Pareto frontier by observing a set of solution points (i.e., observed Pareto set).

5.3 DEMONSTRATION EXAMPLE

In this section, entropy index is used to monitor and compare the performance of two MOGAs in terms of their ability to manipulate diversity of solutions: T-MOGA (Chapter 3) and MOGA-NA. Both algorithms claim to improve the uniformity of the solution set over the Pareto frontier. Thus, to fairly assess this aspect of their performance, a three-objective test example is constructed with an artificial bias toward a sub-region on the Pareto frontier. As a result, solution points tend to cluster in one particular portion of the Pareto optimal frontier, leaving the rest of the optimal region empty or sparsely populated. As a result of this non-uniformity, MOGAs will face difficulty in maintaining diversity among solutions. The entropy index can then be calculated for the solution sets generated by each algorithm and the results are compared. Consider a three-objective minimization problem of the following form:

(5.15)

$$\begin{aligned} &\textit{Minimize } f_1(\mathbf{x}) = f_1(x_1) \\ &\textit{Minimize } f_2(\mathbf{x}) = g(x_2)h(f_1(x_1), g(x_2)) \\ &\textit{Minimize } f_3(\mathbf{x}) = g'(x_3)h'(f_1(x_1), g'(x_3)) \end{aligned}$$

In order to ensure the continuity of the Pareto frontier, following Deb's guideline (Deb, 1998), h and h' are chosen to be monotonically decreasing functions of f_1. Moreover, we choose unimodal, single-variable and linear functions for g's: $g(x_2) = x_2 + 1,\ g'(x_3) = x_3 + 1$, to create no bias in the search space toward or from the Pareto front. However, a lateral bias against finding some sub-regions on the Pareto frontier is introduced by choosing a non-linear function for f_1 as: $f_1(x_1) = \frac{1 - e^{-4x_1}}{1 - e^{-4}}$. This

non-linearity in f_1 will result in a non-uniformity in the distribution of the solutions points over the Pareto front, testing the MOGAs' ability to maintain diversity in the solution set. Since we want to test only this aspect of the MOGAs' performances, the rest of the problem features are designed to be at the lowest complexity level. Thus, h and h' are chosen to be:

$$h(f_1, g) = 1 - \left(\frac{f_1}{g}\right)^{0.5} ; \; h'(f_1, g') = 1 - \left(\frac{f_1}{g'}\right)^{0.1} \tag{5.16}$$

for a convex Pareto frontier with no local Pareto-optimal front (because of uni-modality of g functions). The optimization problem can then be formulated as:

$$\begin{aligned}
&\textit{Minimize} \qquad f_1(\mathbf{x}) = \frac{1 - e^{-4x_1}}{1 - e^{-4}} \\
&\textit{Minimize} \qquad f_2(\mathbf{x}) = (x_2 + 1)\left\{1 - \left[\frac{f_1(x_1)}{x_2 + 1}\right]^{0.5}\right\} \\
&\textit{Minimize} \qquad f_3(\mathbf{x}) = (x_3 + 1)\left\{1 - \left[\frac{f_1(x_1)}{x_3 + 1}\right]^{0.1}\right\} \\
&\text{s.t.:} \qquad 0 \le x_i \le 1; \; i = 1,2,3
\end{aligned} \tag{5.17}$$

A number of solutions are generated randomly and depicted in Figure 5.13. As expected, due to the lateral bias imposed on the problem, there is a tendency among the solutions to concentrate in a sub-region of the Pareto frontier.

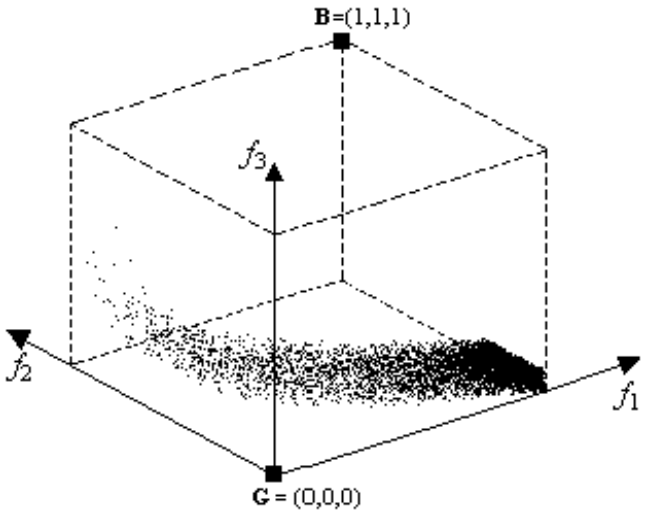


Figure 5.13: Bias of the optimization problem

Both T-MOGA and MOGA-NA are executed for this optimization problem for 40 generations and the solution sets are stored. The entropy of each non-dominated solution set is also computed during optimization. To demonstrate the procedure, the entropy calculation of the 4th generation of E-MOGA is presented here in detail. This generation of chromosomes contains only 9 non-dominated solutions, listed in Table 5.1. The good and bad points of this problem are **G** = (0,0,0) and **B** = (1,1,1) respectively, so:

$$\mathbf{v}_1 = \frac{(\mathbf{G}-\mathbf{B})}{\|\mathbf{G}-\mathbf{B}\|} = \left(\frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}\right) \tag{5.18}$$

From equations 5.10 and 5.11 we obtain,

$$\mathbf{v}_2 = \frac{[\mathbf{u}_2 - (\mathbf{u}_2.\mathbf{v}_1)\mathbf{v}_1]}{\|\mathbf{u}_2 - (\mathbf{u}_2.\mathbf{v}_1)\mathbf{v}_1\|} = \left(-\frac{1}{\sqrt{6}}, \frac{2}{\sqrt{6}}, -\frac{1}{\sqrt{6}}\right) \tag{5.19}$$

$$\mathbf{v}_3 = \frac{[\mathbf{u}_3 - (\mathbf{u}_3.\mathbf{v}_1)\mathbf{v}_1 - (\mathbf{u}_3.\mathbf{v}_2)\mathbf{v}_2]}{\|\mathbf{u}_3 - (\mathbf{u}_3.\mathbf{v}_1)\mathbf{v}_1 - (\mathbf{u}_3.\mathbf{v}_2)\mathbf{v}_2\|} = \left(-\frac{1}{\sqrt{2}}, 0, \frac{1}{\sqrt{2}}\right) \tag{5.20}$$

where $\mathbf{u}_1 = (1,0,0),\ \mathbf{u}_2 = (0,1,0),\ \mathbf{u}_3 = (0,0,1)$ are the unit vectors along f_1, f_2 and f_3 axes. All the solution points in Table 5.1 are projected to the X-Y plane, using equation 5.13:

$$X = \mathbf{f} \cdot \mathbf{v_1};\ Y = \mathbf{f} \cdot \mathbf{v_3};\ \text{where}\quad \mathbf{f} = (f_1, f_2, f_3) \tag{5.21}$$

The X-Y values of the points are listed in Table 5.1.

	f_1	f_2	f_3	X	Y
Point 1	0.9087	0.0414	0.0101	-0.3412	0.6354
Point 2	0.8986	0.0393	0.0140	-0.3404	0.6255
Point 3	0.9154	0.0388	0.0101	-0.3461	0.6401
Point 4	0.9527	0.0256	0.0044	-0.3698	0.6705
Point 5	0.9648	0.0279	0.0031	-0.3723	0.6800
Point 6	0.9680	0.0236	0.0033	-0.3772	0.6821
Point 7	0.5221	0.1839	0.0834	-0.0970	0.3102
Point 8	0.5083	0.1647	0.1192	-0.1216	0.2751
Point 9	0.2178	0.3141	0.0993	0.1270	0.0837

Table 5.1: Non-Dominated Set (4th generation of T-MOGA)

A normal distribution is chosen for the influence function: $\Omega(r) = \frac{1}{\sigma\sqrt{2\pi}} e^{-r^2/2\sigma^2}$. The standard deviation of this function is chosen according to the guideline of Section 5.2.4: $3\sigma = 0.5$. Now, the domain is divided into a 20×20 grid of cells, The density function at the center of each cell can be calculated by aggregating the influences of all other solution points, i.e. equation 5.1. The normalized density, ρ_{ij}, at each cell can then be obtained from equation 5.5. The entropy equation 5.7 yields:

$$H = -\sum_{k_1=1}^{20} \sum_{k_2=1}^{20} \rho_{k_1 k_2} ln\left(\rho_{k_1 k_2}\right) = 3.321 \tag{5.22}$$

In a similar fashion, entropy is computed for every generation of T-MOGA and MOGA-NA and plotted in Figure 6.

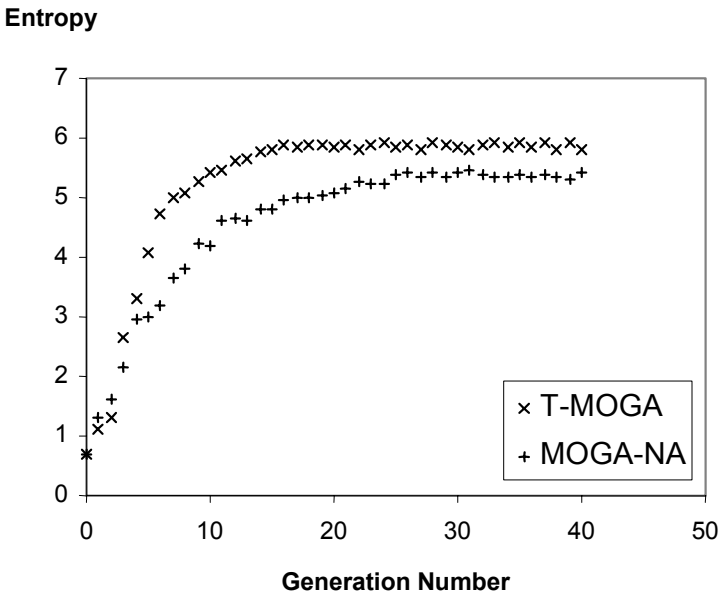


Figure 5.14: Entropy behavior of T-MOGA and MOGA-NA

The optimization processes of both MOGA-NA and T-MOGA start from the same initial population. As the populations evolve, more non-dominated solutions are generated and distributed over the Pareto frontier resulting in entropy to grow rapidly. However, since T-MOGA is specifically designed to maintain diversity in the solution set, the clusters in its population are rapidly dissolved and the solutions are spread more uniformly over the non-dominated hyper-surface. As a result, the entropy of this algorithm grows and saturates faster than MOGA-NA. (A saturation of entropy occurs as soon as the population fully matures and no further improvement in the distribution of the solution points over the Pareto frontier is observed, see Figure 5.14.) Moreover, the final distribution quality (i.e., entropy) of T-MOGA is somewhat better than that of the MOGA-NA.

5.4 CONCLUDING REMARKS

In this chapter, we defined and formulated entropy as an index that could quantitatively assess the distribution quality of an observed Pareto set. The new index can be used to capture and compare the capability of different population-based multi-objective optimization algorithms in generating well-distributed solution sets. This is especially beneficial in comparison study of different MOGAs. In an ideal MOGA, as the population evolves towards maturity, the diversity of observed Pareto set improves, which in turn increases the entropy. This improvement continues until a saturation point wherein the population becomes almost all feasible and non-dominated, spreads as uniformly as possible along the Pareto frontier. At that point, the population has reached its maturity in terms of the distribution quality of solution points and the entropy remains almost constant subsequently.

To demonstrate the applicability and merits of the proposed index, in the example, the entropy of each generation of observed Pareto solutions was computed and graphed throughout the optimization process. The results clearly indicate that as the distribution quality of the solution set improves (i.e., the density surface flattens), the entropy index increases accordingly. Since the evaluation of this index is computationally easy, one can continuously compute and monitor the entropy as a measure of population diversity.

In the next chapter, the proposed approach of this research will be applied several test examples. Some of these examples are real-world design problems, while others are designed specifically to test different performance aspects of the proposed approach.

CHAPTER 6

FURTHER NUMERICAL AND ENGINEERING TEST EXAMPLES

In order to further investigate the performance of the proposed methodologies introduced in the previous chapters, several test examples are presented in this chapter. These test examples are either: 1) numerical test examples that can be tailored to address one particular aspect of performance; or 2) engineering test examples that aim at demonstrating the application of the proposed methodologies. In Section 6.1, a real-world design problem is introduced -- high speed civil transport aircraft. This problem has many design variable and is computationally expensive. Therefore, SMAXED is used to approximate this high-dimensional simulation model and the resulting surrogate model is compared against other conventional techniques. RMSE (root-mean square), max deviation and average error are computed for each surrogate model and the results are discussed. In Section 6.2, several optimization problems are presented to test different aspects of T-MOGA. In particular, Sections 6.2.1 and 6.2.2 present several numerical and unconstrained optimization problems crafted specifically to hinder the performance of MOGAs in one way or another. It is discussed that test problems with difficult optimization features are particularly helpful in comparison study of different MOGAs. A constrained design problem (i.e. design of a vibrating platform) is presented in Section 6.2.3 along with two other numerical test problems to test the performance of T-MOGA in the presence of constraints. In Section 6.3 another

engineering problem is introduced: speed reducer gearbox. In this case, entropy index is used to monitor and compare the performance of T-MOGA versus another MOGA and the results are discussed. Finally, concluding remarks are given in Section 6.4.

6.1 APPROXIMATION

A response function with one design variable was used in Chapter 2 to demonstrate and compare the performance of the SMAXED approach. A more complicated real-world application of SMAXED will be discussed in Chapter 7 as part of a case study. In the following section, another test example is introduced: High Speed Civil Transport (Balabanov et al. 1996) that has 26 design variables.

6.1.1 High Speed Civil Transport (HSCT Commercial Aircraft)

Aircraft designers have given considerable attention to metamodeling of Navier-Stokes fluid flow analyses and finite element structural simulations as they attempt to improve the time-to-market and performance of their products. These simulations are usually very time consuming and a direct application of optimization techniques (without approximating first) is not feasible with time constraints and today's computational capabilities. Therefore, current approaches reported in the literature first employ statistical DOE techniques and perform a limited number of computational experiments. With the resulting data, the designer creates mathematical models using some sort of meta-modeling technique, which is then used in an optimization process.

Among current aircraft design challenges, optimization of the geometry and specifications of a High Speed Civil Transport (HSCT) has been addressed by many

researchers in the past few years (see for instance Srivastava et al. 1999). HSCT is a supersonic aircraft with a cruise speed of 2 to 3 Mach and capacity of more than 200 passengers (see Figure 6.1). This is an ideal test problem for meta-modeling techniques because economic and technical feasibility of such an aircraft mainly lies on the boundaries of the aircraft technology. Typically, this design problem has many design variables and shows to be a noisy (with many local optima) and non-trivial response function.

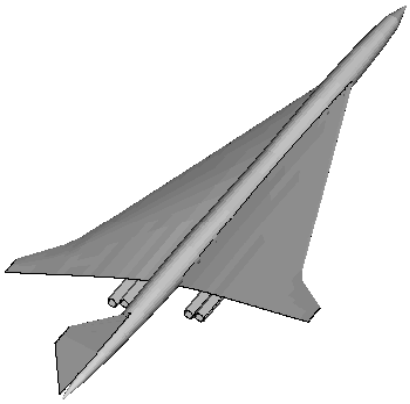


Figure 6.1: Schematic view of HSCT aircraft
(reproduced from Guinta 1997)

A finite-element model for HSCT was first developed at the Multidisciplinary Analysis and Design Center for Advance Vehicles at Virginia Tech (see Giunta et al. 1995; and Balabanov et al. 1996). The version of this model used in this thesis has 25 design variables that determine geometry of the aircraft (listed in Table 6.1).

TE=Trailing Edge, LE=Leading Edge, t/c=thickness/chord ratio

No.	Description	Value	Units
1	Wing root chord	181.48	ft
2	LE break point, x	155.9	ft

3	LE break point, y	49.2	ft
4	TE break point, x	181.6	ft
5	TE break point, y	64.2	ft
6	LE wing tip, x	169.5	ft
7	Wing tip chord	7.00	ft
8	Wing semi-span	75.9	ft
9	Max t/c location	40	% chord
10	Airfoil t/c at root	2.58	%
11	Airfoil t/c at LE break	2.16	%
12	Airfoil t/c at tip	1.80	%
13	Fuselage restraint 1,x	2.20	ft
14	Fuselage restraint 1,r	1.06	ft
15	Fuselage restraint 2,x	12.20	ft
16	Fuselage restraint 2,r	3.50	ft
17	Fuselage restraint 3,x	132.46	ft
18	Fuselage restraint 3,r	5.34	ft
19	Fuselage restraint 4,x	248.67	ft
20	Fuselage restraint 4,r	4.67	ft
21	Nacelle 1, y	26.23	ft
22	Nacelle 2, y	33.09	ft
23	Mission fuel	322,617	lbs
24	Vertical tail area	697.9	sq-ft
25	Horizontal tail area	713.0	sq-ft

Table 6.1: Break down of 25 geometric design variables used in HSCT aircraft design (from NASA MDOB library[1])

The geometric variables can be grouped into five categories: wing platform, airfoil shape, tail areas, nacelle placement, and fuselage shape. Total of eight variables describe the wing platform (variables x_1-x_8 in Figure 6.2), eight variables define the fuselage shape, five variables define the wing leading edge and airfoil section properties (variables x_9-x_{13} in Figure 6.2), two variable define the engine nacelle locations (variables x_{22} and x_{23} in Figure 6.2), and two variables define the horizontal and vertical tail areas. For this HSCT design problem the fuselage has a fixed length of

[1] Online access URL: http://mdob.larc.nasa.gov/mdo.test/index.html

300 ft and an internal volume of 23720 ft^3 (See Guinta et al. 1995 and also Balabanov et al. 1996).

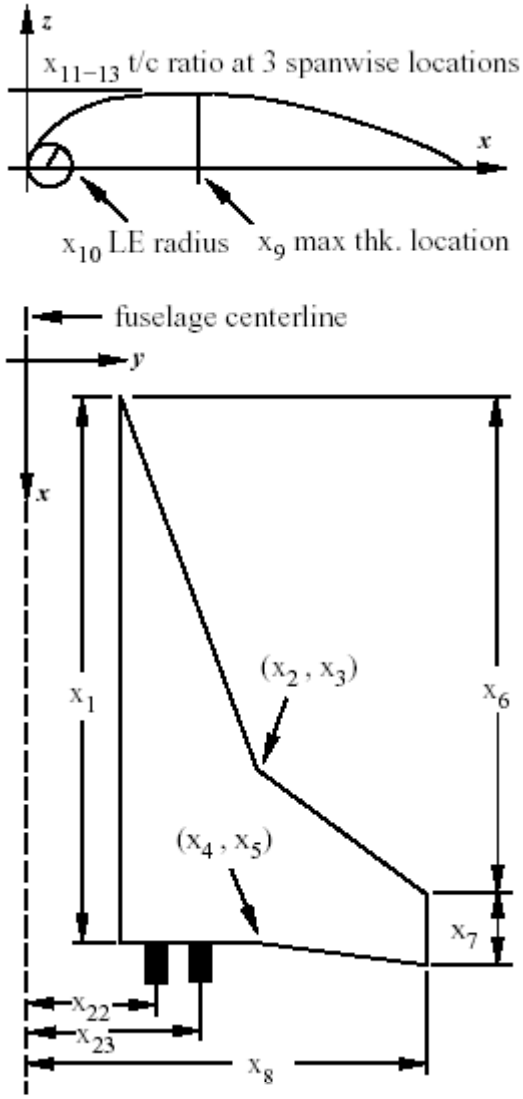


Figure 6.2: Wing design variable for HSCT aircraft
(reproduced from Guinta et al. 1995)

The response function is the take-off gross weight for a range of 5500 nautical miles and cruise speed of 2.4 Mach with a capacity of 251 passengers. This response function represents an overall measure for merits of the aircraft. For instance, if takeoff gross weight is broken down into the sum of dry weight (weight with cargo and

passengers without fuel) and the fuel weight, the dry weight can be used to estimate the cost of the aircraft (using historical data) fuel weight can be related to the yearly costs of maintaining and operating the aircraft. So roughly speaking, minimum weight aircraft corresponds to minimum cost and low-maintenance aircraft (Balabanov et al. 1996). This response function should be minimized, however, due to the computational expenses of conducting experiments (hours on super computers for one evaluation), direct application of optimization techniques to this problem is computationally prohibitive. Moreover, the large number of design variables (25) makes meta-modeling of this response function a challenging undertaking.

The input variables and response data for 2490 experiments are available (at the time of writing this dissertation) on NASA MDOB test suite library[2]. Therefore, new experiments are not necessary. Instead, a handful of the experiments are chosen among these available data points using SMAXED approach. The rest of the available experiment data are used to compute the RMSE (refer to equation 7.5) of the obtained meta-models. The accuracy of several approximation techniques is then compared against SMAXED. To assume a realistic situation (where the data is not already available), it is assumed that each experiment takes about 30 minutes to run, and about 3 days of computational resources are available for this purpose. Therefore, one could conduct only a total of about 60 experiments.

For the first part of this example, a total of 60 experiments are conducted using different techniques (all techniques are implemented by the author). Table 6.2 shows

[2] Online access URL: http://mdob.larc.nasa.gov/mdo.test/index.html

RMSE (equation 7.5), Max Error[3], and Average Error[4] percentages (See Appendix I for definitions) obtained from different techniques, for 60 experiments. For Orthogonal Array design, since the designed experiments may not be actually among the available 2490 experiments, we select the nearest neighbor among the existing data points. For all other design techniques, experiments are selected directly from the existing pool of experiments. The percentage errors are determined based on the deviation of the meta-models from the remaining data points, i.e. 2490-60 = 2430 points.

DOE/Meta-modeling technique	**RMSE (%)**	**Average Error (%)**	**Max Error (%)**
Random Sampling/Kriging	75	51	642
Latin Hypercube/Kriging (McKay et al. 1979)	37	22	315
Orthogonal Array/Kriging (Owen 1992)	52	46	389
Orthogonal Array/Response Surface Analysis (Quadratic)	311	270	1343
Maximum Entropy/Kriging (Koehler and Owen 1996)	32	19	298
SMAXED (6 blocks of 10 experiments)	13	9	218
SMAXED (12 blocks of 5 experiments)	15	11	193

Table 6.2: Approximation using 60 experiments

[3] Similar to RMSE (equation 7.5), Maximum Error is used to determine the accuracy of a meta-model. This is defined as maximum deviation of a meta-model from its original response function, i.e. maximum of $|\hat{y}(\mathbf{x}) - y(\mathbf{x})|$, where **x** is a design vector in the design space, $\hat{y}(\mathbf{x})$ is the corresponding predicted value by the meta-model, and $y(\mathbf{x})$ is the value of the original response function.

[4] This is defined as the average deviation of meta-model from the original response function, i.e. average of $|\hat{y}(\mathbf{x}) - y(\mathbf{x})|$, where **x** is a design vector in the design space.

Table 6.2 shows that Latin Hypercube Sampling, Maximum Entropy and SMAXED all yield relatively lower Max Errors. However, on average SMAXED performs much better, i.e., it has a much lower RMSE and Average Error.

For the second part of this example, 180 experiments are used. The results are listed in Table 6.3.

DOE/Meta-modeling technique	**RMSE (%)**	**Average Error (%)**	**Max Error (%)**
Random Sampling/Kriging	7.4	3.4	56
Latin Hypercube/Kriging (McKay et al. 1979)	3.9	2.4	23
Orthogonal Array/Kriging (Owen 1992)	5.3	4.9	45
Orthogonal Array/Response Surface Analysis (Quadratic)	6.9	3.1	10.2
Maximum Entropy/Kriging (Koehler and Owen 1996)	3.8	2.5	21
SMAXED (18 blocks of 10 experiments)	2.1	1.8	22
SMAXED (36 blocks of 5 experiments)	2.3	1.8	18

Table 6.3: Approximation using 180 experiments

Again, SMAXED performs better on average (i.e. lower RMSE and Average Error). However, Maximum Error is lower with Orthogonal Array/Response Surface Analysis. Note that Response Surface Analysis is a non-interpolative approach, however, it yields a lower maximum error (overshoot) than interpolative approaches for higher number of experiments (a similar observation for response function meta-models versus kriging is reported in Srivastava et al. 1999).

In the next section, several optimization problems are presented to test the performance of T-MOGA (recall Chapter 3).

6.2 TEST CASES FOR T-MOGA

Choosing appropriate test problems is perhaps the most important step in evaluating the merits of a new optimization algorithm. Deb (2001) states that 'difficult' problems must be used to assess the performance of a new algorithm. Easy problems can be solved by almost any algorithm, however, problems with difficult features (e.g. bias, discontinuity, etc.) discriminate among different algorithms. As such, Deb (2001) identified a set of difficult features and defined a set of test problems accordingly.

In the next section, we briefly introduce these test problems and assess the performance of T-MOGA. In Section 6.2.2, another class of problems -- by Zitzler et al. 2000 -- is also introduced to further test T-MOGA.

6.2.1 Deb's Test Problems with Difficult Features (Unconstrained, 2 Variables)

According to Deb (2001) there are two major tasks that a MOGA should address to obtain a good Pareto solution set:

1) Convergence to the Pareto-optimal frontier
2) Maintaining population diversity

Numerical test examples can be set up with several problem features such as multi-modality, deceptive functions and isolated optima that are recognized to hinder the convergence of the population in a MOGA. Similarly, features such as non-convexity, discontinuity of the Pareto frontier and non-uniformity (or bias) in the distribution of the solutions may cause difficulties for MOGAs in maintaining solution

set diversity (Deb 2001). In order to fairly and consistently compare the performance of different MOGAs in one particular aspect, appropriate numerical test examples must be formulated. Deb (2001) formulated a generic (unconstrained) 2-objective optimization problem that can be tuned to address different sources of optimization difficulty:

$$\text{Minimize} \qquad f_1 = f_1(x_1,...,x_m) \tag{6.1}$$

$$\text{Minimize} \qquad f_2 = g(x_{m+1},...,x_N)h(f_1,g)$$

By defining different functional forms for f_1, g, and h, Deb created test problems that are difficult in terms of either convergence or diversity, as listed below. In the first three problems, MOGA will face difficulty in maintaining diversity. In problems 4 and 5, however, the difficulty is in converging to the Pareto frontier (For a detailed description of these difficult features and how to manipulate them in test problems see Deb 2001.) Unlike the first 5 problems, the 6th problem has no difficult feature and thus, is very easy to converge and maintain diversity. Later in this section, it is shown that all examined MOGAs perform equally well for this easy problem and therefore, they cannot be distinguished based on the results from this problem. This further emphasizes the need for using problems with difficult features (such as Problems 1-5) to compare the performance of MOGAs.

Problem 1 (difficult to maintain diversity, Deb 2001): The following test problem has a non-convex local Pareto front and a convex global Pareto front (See Figure 6.3). This poses a great deal of difficulty to MOGAs in obtaining a well-distributed set of

solutions. Especially, those algorithms that exploit the shape of the Pareto frontier to manipulate uniformity of the solution set have to adopt themselves from a non-convex shape to a convex shape as the population evolves from local front to global front.

$$
\begin{aligned}
f_1 &= 4x_1 \\
g &= \begin{cases} 4 - 3\exp\{-\left(\frac{x_2 - 0.2}{0.02}\right)^2\} & 0 \le x_2 \le 0.4 \\ 4 - 2\exp\{-\left(\frac{x_2 - 0.7}{0.2}\right)^2\} & 0.4 \le x_2 \le 1 \end{cases} \\
h &= \begin{cases} 1 - (\frac{f_1}{g})^4 & f_1 \le g \\ 0 & otherwise \end{cases}
\end{aligned}
\tag{6.2}
$$

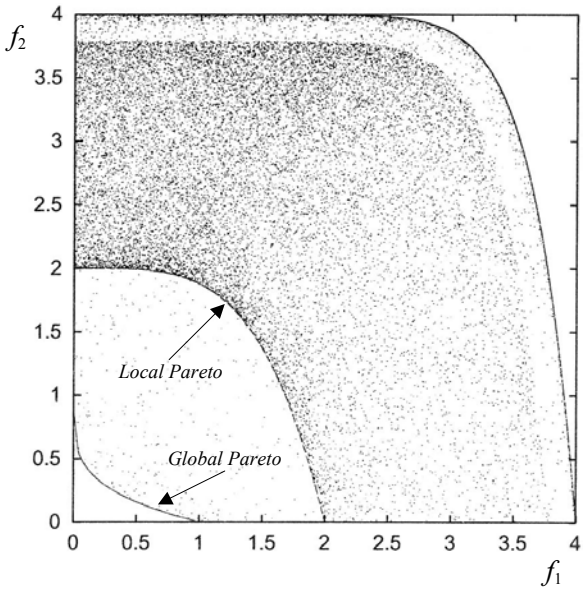


Figure 6.3: Test Problem 1 (Deb, 2001)

Problem 2 (difficult to maintain diversity): The following test problem has a discontinuous Pareto front (See Figure 6.4). Discontinuous Pareto frontier causes difficulty for MOGAs in obtaining a diverse solution set.

$$
\begin{aligned}
f_1 &= x_1 \\
g &= 1 + 10x_2 \\
h &= 1 - \left(\frac{f_1}{g}\right)^2 - \frac{f_1}{g}\sin(8\pi f_1)
\end{aligned}
\tag{6.3}
$$

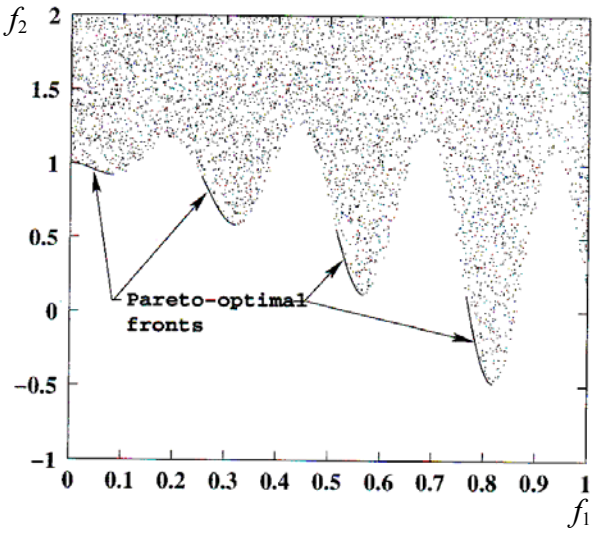


Figure 6.4: Test Problem 2 (Deb, 2001)

Problem 3 (difficult to maintain diversity): The following test problem has a solution space that is laterally biased, i.e., the density of solutions in the right half space of the Pareto frontier (Figure 6.5) is higher than that of the left half. Therefore, MOGA tends to find more solutions in the dense part, leaving the rest of the Pareto frontier empty or sparsely populated.

$$
\begin{aligned}
f_1 &= 1 - \exp\{-4x_1\}\sin^4\{5\pi x_1\} \\
g &= 1 + 10x_2 \\
h &= \begin{cases} 1 - \left(\frac{f_1}{g}\right)^4 & f_1 \le g \\ 0 & otherwise \end{cases}
\end{aligned}
\tag{6.4}
$$

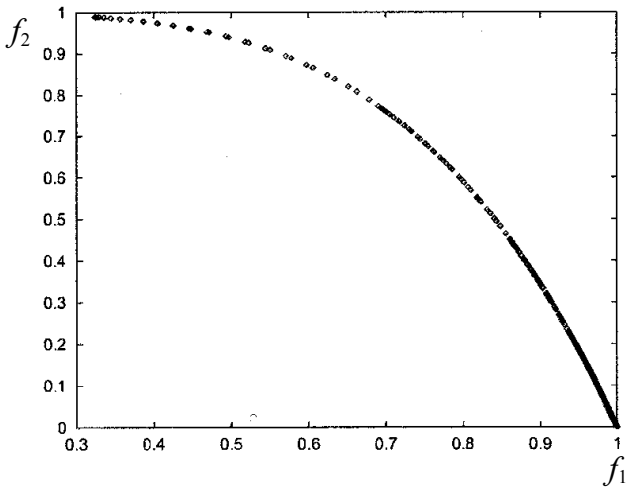


Figure 6.5: Problem 3 (Deb, 2001)

Problem 4 (difficult to converge): The following test problem has local Pareto frontiers with significantly dense solutions, but the global Pareto is very sparsely populated (See Figure 6.6). A MOGA will face difficulty in converging to the global Pareto.

$$\begin{aligned} f_1 &= x_1 \\ h &= 1/f_1 \\ g &= 2 - \exp\{-\left(\frac{x_2 - 0.2}{0.004}\right)^2\} - 0.8\exp\{-\left(\frac{x_2 - 0.6}{0.4}\right)^2\} \end{aligned} \tag{6.5}$$

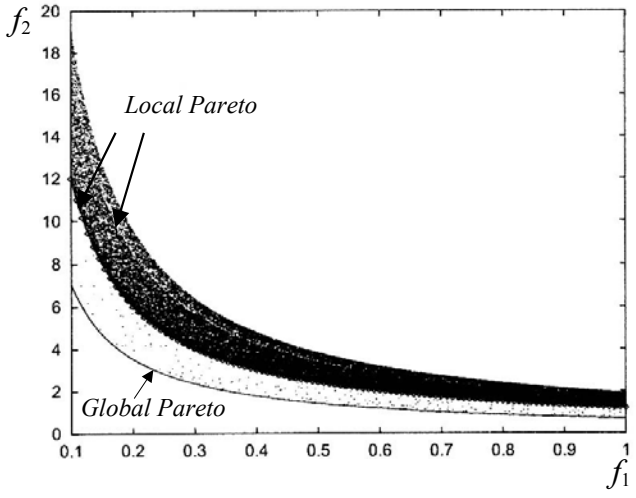


Figure 6.6: Problem 4 (Deb, 2001)

Problem 5 (difficult to converge): The solution space of this problem has a strong bias against the non-convex Pareto frontier. In Figure 6.7, the density of solutions significantly decrease as a MOGA attempts to converge to the Pareto frontier.

$$\begin{aligned} f_1 &= x_1 \\ g &= 1 + x_2^{0.15} \\ h &= 1 - (\frac{f_1}{g})^2 \end{aligned} \tag{6.6}$$

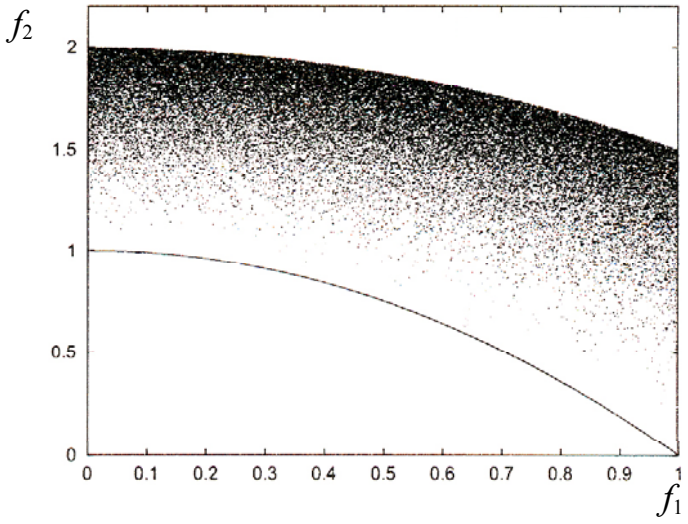


Figure 6.7: Test Problem 5 (Deb, 2001)

Unlike above test problems, the following problem has no difficult features.

Problem 6 (easy to converge and maintain diversity): The Pareto frontier of the following problem is convex, continuous, and unbiased.

$$\begin{aligned} f_1 &= x_1 \\ g &= 1 \\ h &= 1 / f_1 \end{aligned} \tag{6.7}$$

To assess the relative performance of T-MOGA, three other well-known MOGAs are selected:

- NSGA (Nondominated Sorting Genetic Algorithm): Srinavas and Deb, 1994
- MOGA-NA: Naryanan and Azarm 1999
- VEGA (Vector Evaluated Genetic Algorithm): Schaffer 1985.

Two quality indexes are selected to compare the results: 1) *Entropy Index (H)* to measure diversity (from Chapter 5); 2) *Hypercubic Size of the Non-dominated Space* denoted by *S* (Fonseca and Fleming 1996, See Appendix I for definition) to measure closeness to the Pareto frontier. The MOGA properties are listed in Table 6.4.

Parameter	**Value**
Population size	100
Replacement per generation	10
Function calls	550
Crossover type	2-point
Crossover probability	0.8
Mutation probability	0.05
Bits per variable	10
Selection type	Stochastic universal selection (Levine, 1996)

Table 6.4: MOGA Parameters

We performed each of the above MOGAs 10 times for each of the above-mentioned problems. Table 6.5 provides the following information (for a sample of 10 runs):

- Best (highest) obtained entropy index value: H_g
- Worst (lowest) obtained entropy index value: H_b
- Sample mean of obtained entropy index values: H_m

- Best (lowest) obtained *S* index value: S_g
- Worst (highest) obtained *S* index value: S_b
- Sample mean of obtained *S* index values: S_m

	Feature	NSGA	MOGA-NA	VEGA	T-MOGA
PROBLEM 1	Non-Convex Local and Convex Global Front (Difficult to maintain diversity)	H_g= 6.4 H_b= 5.8 **H_m= 6.0** S_g = 0.62 S_b = 0.71 **S_m = 0.64**	H_g= 5.3 H_b= 4.9 **H_m= 5.2** S_g =0.68 S_b = 0.82 **S_m = 0.71**	H_g= 4.2 H_b= 4.1 **H_m= 4.1** S_g = 0.71 S_b = 0.75 **S_m = 0.72**	H_g= 6.8 H_b= 6.4 **H_m= 6.6** S_g = 0.49 S_b = 0.69 **S_m = 0.63**
PROBLEM 2	Disconnected Pareto (Difficult to maintain diversity)	H_g= 5.1 H_b= 4.0 **H_m= 4.2** S_g = 0.27 S_b = 0.31 **S_m = 0.29**	H_g= 4.4 H_b= 4.0 **H_m= 4.4** S_g =0.28 S_b = 0.33 **S_m = 0.30**	H_g= 3.8 H_b= 3.8 **H_m= 3.8** S_g = 0.73 S_b = 0.89 **S_m = 0.77**	H_g= 5.7 H_b= 4.8 **H_m= 5.5** S_g = 0.25 S_b = 0.30 **S_m = 0.27**
PROBLEM 3	Lateral Bias (Difficult to maintain diversity)	H_g= 5.9 H_b= 5.7 **H_m= 5.8** S_g = 0.44 S_b = 0.49 **S_m = 0.45**	H_g= 5.1 H_b= 5.0 **H_m= 5.1** S_g = 0.55 S_b = 0.68 **S_m = 0.61**	H_g= 5.2 H_b= 5.0 **H_m= 5.1** S_g = 0.67 S_b = 0.68 **S_m =0.67**	H_g= 7.6 H_b= 7.0 **H_m=7.1** S_g = 0.36 S_b = 0.38 **S_m = 0.38**
PROBLEM 4	Isolated Global Pareto (Difficult to converge)	H_g= 5.9 H_b= 5.8 **H_m=5.8** S_g = 0.37 S_b = 0.39 **S_m= 0.37**	H_g= 6.0 H_b= 5.6 **H_m= 5.8** S_g = 0.38 S_b = 0.39 **S_m =0.39**	H_g= 5.0 H_b= 4.3 **H_m= 4.9** S_g = 0.53 S_b = 0.61 **S_m =0.55**	H_g= 6.1 H_b= 5.8 **H_m= 5.9** S_g = 0.37 S_b = 0.40 **S_m = 0.39**
PROBLEM 5	Bias against Pareto (Difficult to converge)	H_g= 6.8 H_b= 6.6 **H_m=6.7** S_g = 0.23 S_b = 0.29 **S_m = 0.25**	H_g= 6.1 H_b= 6.0 **H_m=6.1** S_g = 0.24 S_b = 0.32 **S_m = 0.29**	H_g=5.9 H_b= 5.2 **H_m= 5.7** S_g = 0.62 S_b = 0.70 **S_m = 0.64**	H_g= 7.8 H_b= 7.1 **H_m=7.3** S_g = 0.26 S_b = 0.30 **S_m = 0.27**
PROBLEM 6	Easy	H_g=6.2 H_b=6.1 **H_m=6.2** S_g = 0.19 S_b = 0.19 **S_m = 0.19**	H_g= 6.3 H_b= 6.2 **H_m=6.2** S_g = 0.16 S_b = 0.17 **S_m = 0.17**	H_g= 6.4 H_b= 6.4 **H_m=6.4** S_g = 0.15 S_b = 0.15 **S_m = 0.15**	H_g= 6.4 H_b= 6.3 **H_m=6.3** S_g = 0.15 S_b = 0.15 **S_m = 0.15**

Table 6.5: Quality of solutions obtained from 4 MOGAs. The first 3 problems (with gray shadow) pose difficulties in achieving a diverse solution set. In Problems 4 and 5, MOGAs face difficulty to converge to the Pareto frontier. Problem 6 is easy to converge and maintain diversity. Note that the data in this table is based on 10 runs.

From the above table it can be observed that:

- **In terms of diversity:** In problems with diversity difficulties (Problems 1-3) and Problem 5, T-MOGA performed better (higher entropy) than MOGA-NA, NSGA, and VEGA. In problem 4, all MOGAs except VEGA performed relatively well in terms of obtaining diverse solutions, however, T-MOGA yielded slightly higher entropy values. In problem 6, entropies are almost the same (all algorithms performed well). This easy problem does not discriminate among these MOGAs, and therefore, is not helpful in our comparison study (This easy example was included only to demonstrate the need for using test problems with difficult features.)
- **In terms of convergence:** In problems 4 and 5 (difficult to converge), NSGA showed a slightly better performance than MOGA-NA and T-MOGA in terms of *S* index (Note that the lower the value of *S*, the better). VEGA showed a relatively weak performance. For problems 1, 2, 3 and 6, however, T-MOGA performed as well as or better than other algorithms in terms of *S*.

To summarize: *T-MOGA performed much better than others in terms of obtaining diverse solutions, and almost as good as MOGA-NA and NSGA (and better than VEGA) in terms of converging to the Pareto frontier.*

Test problems 1, 2, 3, 4, and 5 had difficult features to magnify the differences among MOGAs, however, they had only 2 design variables. In the next section, we present another class of test problems that is based on equation 6.2.1 and has many variables. These test problems were first introduced by Zitzler et al. (2000).

6.2.2 ZTD Test Problems (Unconstrained, Many Variables)

Zitzler et al. (2000) formulated six test problems with many variables. In the following, 4 of these problems are listed with their properties. The other 2 functions are Boolean functions defined over bit-strings. Since T-MOGA is a phenotype alteration in MOGA (i.e. expansion operator manipulates the population in the objective space not in the chromosome level) and there is no reason to believe that T-MOGA is any different than other baseline MOGAs in terms of bit-strings, these two Boolean test functions are omitted in the following. The numbering of these problems is also modified from the original work.

ZTD 1 (difficult to maintain diversity): non-convex with local Pareto frontier with 30 variables

$$\begin{aligned} f_1 &= x_1 \\ g &= 1 + \frac{9}{29}\sum_{i=2}^{30} x_i \\ h &= 1 - \left(\frac{f_1}{g}\right)^2 \end{aligned} \tag{6.8}$$

ZTD 2 (difficult to maintain diversity): disconnected with 30 variables.

$$\begin{aligned} f_1 &= x_1 \\ g &= 1 + \frac{9}{29}\sum_{i=2}^{30} x_i \\ h &= 1 - \sqrt{f_1 / g} - \left(f_1 / g\right)\sin(10\pi f_1) \end{aligned} \tag{6.9}$$

ZTD 3 (difficult to converge): biased against Pareto frontier with 10 variables.

$$f_1 = 1 - \exp\{-4x_1\}\sin^6\{6\pi x_1\}$$
$$g = 1 + 1 + 9\left[\sum_{i=2}^{30} x_i / 9\right]^{0,25} \tag{6.10}$$
$$h = 1 - (\frac{f_1}{g})^2$$

ZTD 4 (easy to converge and maintain diversity): convex and unbiased with 30 variables

$$f_1 = x_1$$
$$g = 1 + \frac{9}{29}\sum_{i=2}^{30} x_i \tag{6.11}$$
$$h = 1 - \sqrt{f_1 / g}$$

In a similar fashion to the previous section, the results are listed in Table 6.6.

	Feature	NSGA	MOGA-NA	VEGA	T-MOGA
ZTD 1	Non-Convex Local and Convex Global Pareto	H_g = 3.1 H_b= 2.7 **H_m= 3.0** S_g = 0.30 S_b = 0.49 **S_m = 0.32**	H_g = 3.3 H_b= 3.1 **H_m=3.3** S_g = 0.29 S_b = 0.38 **S_m = 0.30**	H_g = 3.8 H_b= 3.7 **H_m=3.7** S_g = 0.83 S_b = 0.88 **S_m =0.84**	H_g = 4.3 H_b= 3.9 **H_m= 4.1** S_g = 0.23 S_b = 0.40 **S_m =0.26**
ZTD 2	Disconnected Pareto	H_g = 3.9 H_b= 3.0 **H_m= 3.7** S_g = 0.36 S_b = 0.40 **S_m = 0.39**	H_g = 4.6 H_b= 3.5 **H_m= 4.1** S_g = 0.51 S_b = 0.55 **S_m =0.51**	H_g = 4.4 H_b= 2.8 **H_m=3.1** S_g = 0.62 S_b = 0.82 **S_m = 0.69**	H_g = 5.4 H_b= 4.9 **H_m= 5.3** S_g = 0.21 S_b = 0.39 **S_m =0.30**
ZTD 3	Biased against Pareto	H_g = 3.9 H_b= 3.6 **H_m=3.8** S_g = 0.41 S_b = 0.46 **S_m = 0.44**	H_g = 3.4 H_b= 3.0 **H_m= 3.3** S_g = 0.49 S_b = 0.58 **S_m = 0.53**	H_g = 2.3 H_b= 2.0 **H_m= 2.2** S_g = 0.88 S_b = 0.93 **S_m = 0.90**	H_g = 3.7 H_b= 3.6 **H_m=3.7** S_g = 0.43 S_b = 0.48 **S_m = 0.46**
ZTD 4	Easy	H_g = 5.1 H_b= 4.8 **H_m= 5.0** S_g = 0.48 S_b = 0.51 **S_m =0.49**	H_g = 4.9 H_b= 4.8 **H_m= 4.8** S_g =0.41 S_b = 0.55 **S_m =0.47**	H_g =3.6 H_b= 3.3 **H_m= 3.5** S_g = 0.81 S_b = 0.88 **S_m =0.84**	H_g = 5.4 H_b= 4.9 **H_m= 5.1** S_g =0.45 S_b = 0.51 **S_m =0.50**

Table 6.6: Quality of solutions obtained from 4 MOGAs. The first 2 problems (with gray shadow) pose difficulties in achieving a diverse solution set. In Problem ZTD 3,

MOGAs face difficulty to converge to the Pareto frontier. ZTD 4 is easy to converge and maintain diversity. Note that the data in this table is based on the best result out of 10 runs.

Similar to our observation in the previous section, T-MOGA performed much better in terms of maintaining diversity in the first two problems (with features that work against achieving diversity). Except for VEGA, all MOGAs yielded similarly high entropies for ZTD 3 and 4. In terms of convergence to the Pareto frontier, T-MOGA, NSGA and MOGA-NA all performed almost in the same way, while VEGA shows a slower convergence.

The test examples in the past two sections were all unconstrained. In the next section, we investigate the effect of constraints on the performance of T-MOGA.

6.2.3 Constrained Test Cases

In this section three constrained problems are chosen. The first problem (problem C1) is an engineering optimization problem: Design of a Vibrating Platform. Problems C2 and C3 are two numerical optimization problems with constraints.

Problem C1- Design of a Vibrating Platform (Messac 1996)

The design problem is taken from Messac (1996) and consists of a pinned-pinned sandwich beam with a vibrating motor on its top. As shown in Figure 6.8, the beam has five layers of three different materials. There is a middle layer and two sandwiched layers. The distance from the center of the beam to the outer edge of each layer comprises three of the sizing design variables, d_1, d_2, and d_3. The width of the beam, b, and the length of the beam, L, are the other two sizing design variables. There

are also three combinatorial variables for the material type M_i, where i=1,2,3, for the different materials that can be used for each layer. Hence, there are 8 design variables, 3 combinatorial variables for the material type of the 3 layers, and 5 sizing variables.

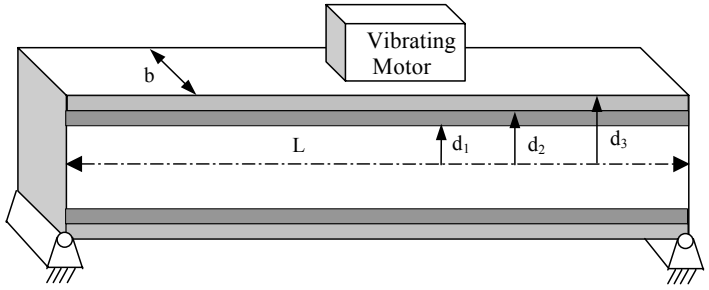


Figure 6.8: Vibrating platform design problem

The problem has two design objectives: 1) Maximize the fundamental frequency of the beam, and 2) Minimize the material cost. The maximization of the first objective is converted to a minimization form by assuming a negative sign. Therefore, the problem can be formulated as follows:

$$
\begin{aligned}
&\textit{Minimize} && f_1(d_1,d_2,d_3,b,L,M_i) = -(\pi/2L^2)(EI/\mu)^{0.5} \\
&\textit{Minimize} && f_2(d_1,d_2,d_3,b,M_i) = 2b[c_1 d_1 + c_2(d_2 - d_1) + c_3(d_3 - d_2)]
\end{aligned}
$$

Subject to: (6.12)

$$
\begin{aligned}
&\mu L - 2800 \le 0 \\
&d_2 - d_1 - 0.15 \le 0 \\
&d_3 - d_2 - 0.01 \le 0 \\
&0.05 \le d_1 \le 0.5 \\
&0.2 \le d_2 \le 0.5 \\
&0.2 \le d_3 \le 0.6 \\
&0.35 \le b \le 0.5 \\
&3 \le L \le 6
\end{aligned}
$$

where,

$$
\begin{aligned}
&(EI) = (2b/3)[E_1 d_1^3 + E_2(d_2^3 - d_1^3) + E_3(d_3^3 - d_2^3)] \\
&(\mu) = 2b[\rho_1 d_1 + \rho_2(d_2 - d_1) + \rho_3(d_3 - d_2)]
\end{aligned}
$$

E_i is the modulus of elasticity of material M_i, while ρ_i is the density, and c_i is the cost. According to the material type variable M_i, the value of the parameters E_i, ρ_i, and

c_i is different for different layer material, as given in Table 6.7. It is assumed that the material types for the three layers are mutually exclusive. In other words, the same material cannot be used for more than one layer. However, the layers are allowed to have zero thickness. The first three constraints refer to upper bounds on the mass of the beam, thickness of layer 2, and thickness of layer 3, respectively. The last 5 constraints are the set constraints on the sizing variables.

Material M_i	ρ_i (Kg/m^3)	E_i (N/m^2)	C_i (\$/volume)
1	100	1.6×10^9	500
2	2,770	70×10^9	1,500
3	7,780	200×10^9	800

Table 6.7: Material properties of the vibrating platform design problem

The next two constrained problems (Problems C2 and C3) are two numerical examples with simple objective functions. The only challenge in solving these two problems is the fact that they are constrained and thus, they are appropriate for comparing the performance of MOGAs only in terms of their ability to handle constraints.

Problem C2 (Binh and Korn 1997)

The formulation of this two-objective optimization problem is given below.

$$
\begin{aligned}
&\textit{Minimize} \quad && f_1 = 4x_1^2 + 4x_2^2 \\
&\textit{Minimize} \quad && f_2 = (x_1 - 5)^2 + (x_2 - 5)^2 \\
&\textit{subject to} && \\
&&& (x_1 - 5)^2 + x_2^{\,2} \le 25 \\
&&& (x_1 - 8)^2 + (x_2 + 3)^2 \ge 7.7 \\
&&& 0 \le x_1 \le 5 \\
&&& 0 \le x_2 \le 3
\end{aligned}
\qquad (6.13)
$$

For this test problem (Binh and Korn 1997), the constraint handling technique of Section 3.2.5 is used. The constraints have simple mathematical forms (circles in this example) and therefore, the proposed technique of reflection can be employed analytically (similar to reflection of gas molecules from walls of a container).

Problem C3 (Deb 2001)

The formulation of this test problem is given below.

$$
\begin{aligned}
&\textit{Minimize} \qquad f_1 = x_1 \\
&\textit{Minimize} \qquad f_2 = x_2\left(1-\sqrt{\frac{x_1}{x_2}}\right) \\
&\textit{subject to} \\
&\qquad 0.2(\sin(10\pi(\sin(-0.2\pi)(f_2-1)+\cos(-0.2\pi)f_1)))^6 - \\
&\qquad\qquad \cos(-0.2\pi)(f_2-1)+\sin(-0.2\pi)f_1 \le 0 \qquad (6.14) \\
&\qquad 0 \le x_1 \le 1 \\
&\qquad 1 \le x_2 \le 5
\end{aligned}
$$

This constraint makes parts of the unconstrained Pareto frontier infeasible. Moreover, the periodic nature of this constraint creates alternating feasible-infeasible regions that pose hardship on convergence and diversity of MOGAs. For this test problem, reflection technique of Chapter 3 is not easy to perform analytically. Computational methods (similar to Section 5.2.6) could also become computationally expensive. Therefore, for this problem an alternative approach is used: points that would become infeasible by expansion operator are frozen (not expanded anymore). This is not consistent with Maxwellian assumption; however, T-MOGA still shows a tendency to yield a high entropy.

10 MOGA runs are performed for the above three problems (and for each MOGA). The results are listed below.

	Feature	NSGA	MOGA-NA	VEGA	T-MOGA
C1	Multiple constraints (engineering design limitations)	$H = 5.7$ $S = 0.32$	$H = 6.4$ $S = 0.22$	$H = 6.1$ $S = 0.34$	$H = 7.5$ $S = 0.23$
C2	Constraints with easy Objectives	$H = 6.0$ $S = 0.33$	$H = 5.1$ $S = 0.28$	$H = 5.9$ $S = 0.32$	$H = 7.9$ $S = 0.31$
C3	Difficult constraint with easy Objectives (Periodic nature of constraint function creates alternating feasible-infeasible bands)	$H = 6.9$ $S = 0.30$	$H = 7.3$ $S = 0.38$	$H = 5.9$ $S = 0.37$	$H = 7.3$ $S = 0.31$

Table 6.8: Quality of solutions obtained from 4 MOGAs.

Each MOGAs is run 10 times for 200 function calls. The best values of quality indexes are listed in Table 6.8. In terms of entropy, T-MOGA performs equally well or better than all other algorithms in all cases. In Problems C1 and C2 in particular, T-MOGA yields much higher entropy.

6.3 ENGINEERING TEST CASE FOR ENTROPY INDEX

In Chapters 4 and 5 we introduced quality indexes for comparison study of different MOGAs. In Chapter 5 in particular, we compared the performance of two MOGAs by monitoring the entropy index during optimization of a numerical test example. In this section, an engineering design problem is introduced and optimized using T-MOGA (Chapter 3) as well as another MOGA (MOGA-NA). The entropy index of both MOGAs are calculated and plotted throughout the process.

6.3.1 Speed Reducer Gearbox for Light Aircrafts

This example was originally formulated by Golinski (1970) as a single-objective optimization problem. The problem has been converted into a two-objective optimization problem. A simple gear-box is shown in Figure 6.9 that can be used in light airplanes between engine and propeller. There are seven design variables in the formulation: gear face width (x_1), teeth module (x_2), number of teeth of pinion (x_3 – *integer variable*), distance between bearings 1 (x_4), distance between bearings 2 (x_5), diameter of shaft 1 (x_6), and diameter of shaft 2 (x_7). The first design objective, f_1, is to minimize the volume. The second objective, f_2, is to minimize the stress in one of the two gear shafts.

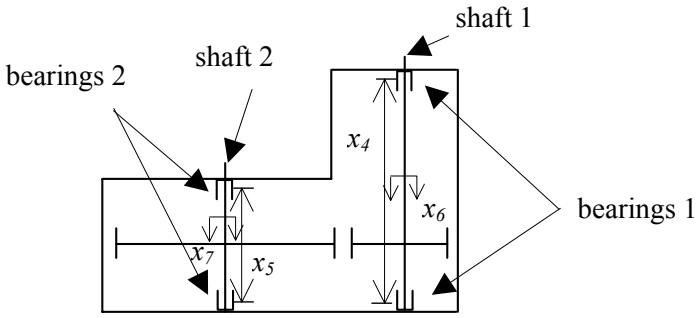


Figure 6.9: Speed reducer

The design is subject to a number of constraints imposed by gear and shaft design practices. An upper and lower limit is imposed on each of the seven design variables. There are 11 other inequality constraints (of which one is a constraint imposed on the first objective), as follows: g_1 is an upper bound of the bending stress of the gear tooth; g_2 : upper bound of the contact stress of the gear tooth; g_3, g_4 are upper bounds of the transverse deflection of the shafts;g_5-g_7 are dimensional

restrictions based on space and/or experience; g_8, g_9 are design requirements on the shaft based on experience; and g_{10}, g_{11} are constraints on stress in the gear shafts. The optimization formulation is:

$$\text{Minimize } f_{weight} = f_1 = 0.7854x_1x_2^2(10x_3^2/3 + 14.933x_3 - 43.0934) - 1.508x_1(x_6^2 + x_7^2) + 7.477(x_6^3 + x_7^3) + 0.7854(x_4x_6^2 + x_5x_7^2)$$

$$\text{Minimize } f_{stress} = f_2 = \frac{\sqrt{(745x_4/x_2x_3)^2 + 1.69\text{x}10^7}}{0.1x_6^3}$$

Subject to: (6.15)

$$g_1 : \frac{1}{(x_1x_2^2x_3)} - \frac{1}{27} \le 0 \qquad g_2 : \frac{1}{(x_1x_2^2x_3^2)} - \frac{1}{397.5} \le 0$$

$$g_3 : \frac{x_4^3}{(x_2x_3x_6^4)} - \frac{1}{1.93} \le 0 \qquad g_4 : \frac{x_5^3}{(x_2x_3x_7^4)} - \frac{1}{1.93} \le 0$$

$$g_5 : x_2x_3 - 40 \le 0 \qquad g_6 : \frac{x_1}{x_2} - 12 \le 0$$

$$g_7 : 5 - \frac{x_1}{x_2} \le 0 \qquad g_8 : 1.9 - x_4 + 1.5x_6 \le 0$$

$$g_9 : 1.9 - x_5 + 1.1x_7 \le 0 \qquad g_{10} : f_1(x) \le 1300$$

$$g_{11} : \frac{\sqrt{(745x_5/x_2x_3)^2 + 1.575\text{x}10^8}}{0.1x_7^3} \le 1100$$

The lower and upper limits on the seven variables are:

$$g_{12,13} : 2.6 \le x_1 \le 3.6 \qquad g_{14,15} : 0.7 \le x_2 \le 0.8$$

$$g_{16,17} : 17 \le x_3 \le 28 \qquad g_{18,19} : 7.3 \le x_4 \le 8.3$$

$$g_{20,21} : 7.3 \le x_5 \le 8.3 \qquad g_{22,23} : 2.9 \le x_6 \le 3.9$$

$$g_{24,25} : 5.0 \le x_7 \le 5.5$$

The Pareto solutions obtained using T-MOGA and MOGA-NA are shown in Figure 6.10 and the parameters are listed in Table 6.9. From the results shown in Figure 6.10 , one can see that T-MOGA has outperformed MOGA-NA in terms of both the coverage of the Pareto frontier and the uniformity of spread, i.e., the range of Pareto frontier covered by solutions from T-MOGA is significantly larger than that of MOGA-NA. Also, the solution points are spread uniformly in the T-MOGA while the solution points are mostly clustered in MOGA-NA.

	MOGA	T-MOGA
Population size	50	50
Replacement per generation	10	10
Function calls	550	550
Crossover type	2-point	2-point
Crossover probability	0.8	0.8
Mutation probability	0.05	0.05
Bits per variable	10	10
Selection type	Stochastic universal selection (Levine, 1996)	Stochastic universal selection (Levine, 1996)
Expansion percentage	NA	10%
Expansion start	NA	5^{th}
Expansion finish	NA	45^{th}

Table 6.9: MOGA parameters in the speed-reducer example

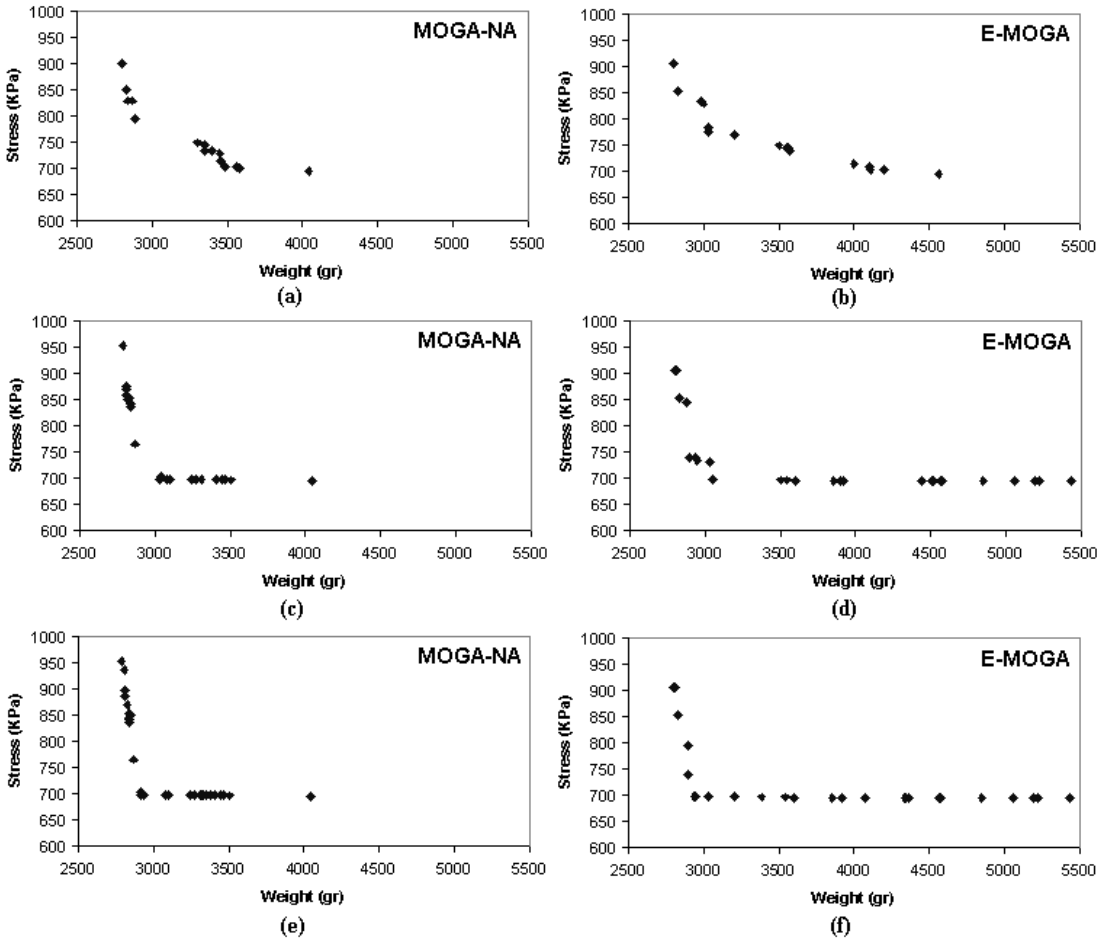


Figure 6.10: Pareto solution sets for speed-reducer example: (a) MOGA-NA; 150 function calls, (b) T-MOGA; 150 function calls, (c) MOGA-NA; 250 function calls, (d) T-MOGA; 250 function calls, (e) MOGA-NA; 550 function calls, (f) T-MOGA; 550 function calls

As mentioned in Chapter 4, the above visual assessment is not possible in general (especially in problems with 3 or more objectives). Therefore, here we use entropy index (Chapter 5) to monitor the diversity of solution sets on a quantitative basis during optimization of the speed reducer problem. The good and bad points (see Appendix I for definition) are determined subjectively, based on the upper bounds of stress in the two shafts (constraints C_{10} and C_{11}), and a sample of points obtained from a single run of MOGA. We arbitrarily overestimate the good and bad points to be:

$\mathbf{G}_0$=(2500, 600) and $\mathbf{B}_0$=(7000, 1300). After normalization, the normalized good and bad points become **G** = (0,0) and **B** = (1,1), respectively. So the projection direction can be obtained:

$$\mathbf{v}_1 = \frac{(\mathbf{G}-\mathbf{B})}{\|\mathbf{G}-\mathbf{B}\|} = -\left(\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}\right) \tag{6.16}$$

Following Gram-Schmidt procedure, Equation 5.10, we obtain:

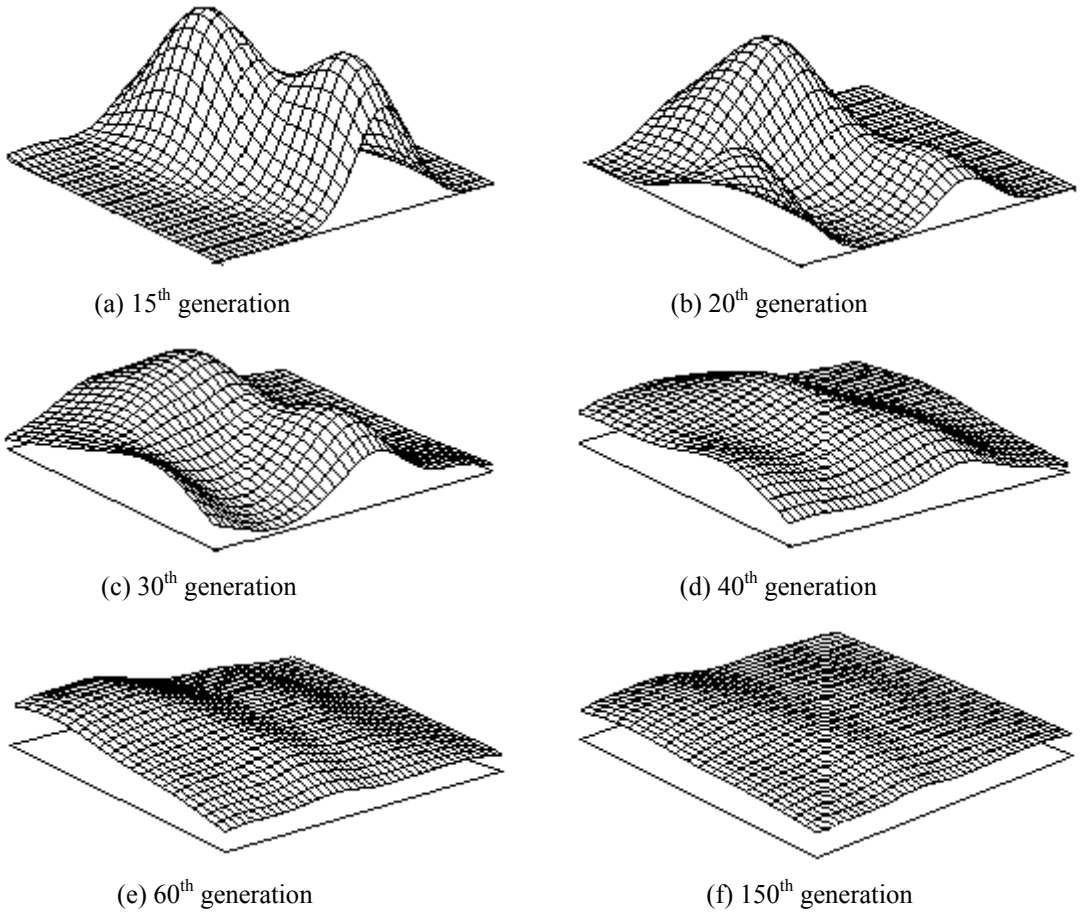
$$\mathbf{v}_2 = \frac{[\mathbf{u}_2 - (\mathbf{u}_2.\mathbf{v}_1)\mathbf{v}_1]}{\|\mathbf{u}_2 - (\mathbf{u}_2.\mathbf{v}_1)\mathbf{v}_1\|} = \left(\frac{1}{\sqrt{2}}, \frac{-1}{\sqrt{2}}\right) \tag{6.17}$$

where $\mathbf{u}_1 = (1,0)$, and $\mathbf{u}_2 = (0,1)$ unit vectors along f_1, and f_2 axes. All of the solution points are projected on the projection plane, using Equation 5.12:

$$X = \mathbf{f} \cdot \mathbf{v_1} \quad ; \quad Y = \mathbf{f} \cdot \mathbf{v_2} \qquad \text{where:} \quad \mathbf{f} = (f_1, f_2) \tag{6.18}$$

Now, to construct a density surface on the projection hyper-plane, we assume a Gaussian influence function, i.e., $\Omega(r) = \frac{1}{\sigma\sqrt{2\pi}} e^{-r^2/2\sigma^2}$. The standard deviation, σ, is set according to the guideline of Section 5.2.4, i.e., 6σ=1 or σ=1/6. The projection of the feasible domain on the projection plane is estimated by a rectangle as explained in Section 5.2.4. A 20×20 grid of cells is constructed on the estimated feasible domain. The density function at the center of each cell can be calculated by aggregating the

influence of all solution points on that cell (recall Equation 5.1). T-MOGA and MOGA-NA are executed for 150 generations and the solution sets are stored. A density surface can then be constructed for every generation of the observed Pareto set as the population evolves (Figure 6.11.)



(a) 15th generation (b) 20th generation

(c) 30th generation (d) 40th generation

(e) 60th generation (f) 150th generation

Figure 6.11: The density surface of non-dominated solution sets in T-MOGA

It is clear from this figure that as the population matures (i.e., the observed Pareto solutions converge), the density surface gradually flattens. For instance, the density surface of the 15th generation consists of two sharp peaks and the rest of the domain has a low density-function value. In contrast, the 150th generation forms a

nearly flat density surface. This is because of continuous improvement in the population diversity during the evolution of T-MOGA. In order to observe the distribution quality of the generations during T-MOGA and MOGA-NA, the entropy of each generation is calculated from Equation 5.7 and graphed in Figure 6.12.

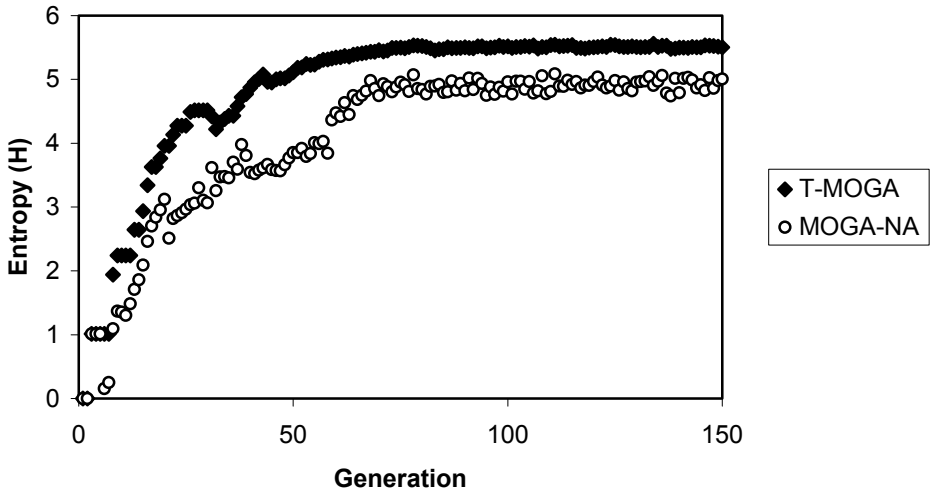


Figure 6.12: Entropy history.

Clearly, the entropy of T-MOGA grows faster and saturates before MOGA-NA (maximum achieved diversity). Moreover, the steady state entropy of T-MOGA is higher than that of MOGA-NA, which means that T-MOGA managed to obtain a more diverse solution set. In the first two generations of T-MOGA, there are no feasible solutions and thus the entropy is zero. The 3rd to 7th generations contain only one feasible solution, so their entropy remains relatively low. As the population evolves, more feasible non-dominated solutions are generated and thus, the corresponding entropy grows rapidly as the non-dominated hyper-surface is covered with more and more solutions. However, after the 30th generation, a large portion of the population is

feasible and therefore, there is no improvement in terms of number of feasible solutions. However, the entropy still increases due to expansion operators -- clusters in the population are gradually dissolved and the solutions are spread more uniformly over the non-dominated hyper-surface covering a larger portion of the feasible domain. The population reaches maturity around the 80^{th} generation and the entropy remains almost constant until T-MOGA stops after a total of 150 generations. During this saturation period, from the 80^{th} generation to the 150^{th} generation, no significant improvement is observed in the distribution of the solutions over the Pareto frontier. Recall that this is equivalent to equilibrium state in Maxwellian systems where the system tends to achieve a steady-state (time independent) state with maximum entropy.

6.4 CONCLUDING REMARKS

In addition to the test examples of Chapters 2, 3, 4, and 5, we presented several additional test cases in this chapter to further investigate the performance of the methodologies of this thesis. In particular:

- SMAXED created a more accurate surrogate model for the large-scale response function of high-speed civil transport example (26 variables) in Section 6.1.
- A collection of numerical test examples in Section 6.2 demonstrated more diverse (higher entropy) solution sets from T-MOGA as compared to other methods.
- In the speed reducer gearbox example, entropy index was successfully used to monitor and compare the performance of T-MOGA.

The next chapter presents a detailed case study: Crashworthiness Design of Front-End of a Pickup Truck. This case study takes advantage of SMAXED, T-MOGA, and Entropy Index in the context of an overall approximation-assisted optimization approach.

CHAPTER 7

CASE STUDY: CRASHWORTHINESS DESIGN OF FRONT-END OF A PICKUP TRUCK

This chapter presents an overall framework for approximation-assisted multi-objective crashworthiness optimization of front-end of a pick-up truck[5]. The finite element simulation of physical phenomena that occur during a crash event is highly nonlinear, with variables and response functions interacting in a complicated fashion. It is shown that while a direct application of evolutionary multi-objective optimization techniques to such a problem is computationally intractable, we first employ SMAXED (our approach from Chapter 2) to create a meta-model for the finite-element simulation. Then, the meta-model is optimized using T-MOGA (recall Chapter 3). Entropy index (recall Chapter 5) is used to monitor the quality of solutions obtained by optimization process to ensure a diverse representation of the approximated Pareto-frontier.

This chapter is organized as follows: Section 7.1 describes an approximation-assisted optimization framework. The case study of this chapter is introduced in Section 7.2 as a demonstration for the proposed framework. The concluding remarks of this chapter are given in Section 7.3.

[5] This case study was done as part of a collaboration with Professor A. Diaz and his student Mr. A. Ravisekar from Michigan State University; see Farhang-Mehr et al. (2003)

7.1 OVERALL FRAMEWORK: APPROXIMATION-ASSISTED OPTIMIZATION

Figure 7.1 illustrates the overall framework of this research -- approximation-assisted multi-objective optimization and quality assessment. As shown in Figure 7.1, the overall approach starts with choosing those objectives and constraints that are computationally expensive and need to be approximated. The computationally expensive responses are approximated sequentially using SMAXED approach described in Chapter 2, while simpler functions are used directly by the optimization engine. Next, T-MOGA is performed (as described in Chapter 3) and a set of Pareto-optimal design solutions are generated and evolved to fully represent the optimal design space (or achieve the best possible representation). The distribution quality of the solution set is monitored during the optimization process to determine whether or not the desired level of solution diversity is observed along the Pareto frontier. The result will be a set of non-dominate solutions, distributed such that the best possible description of the entire optimal feasible space can be given. Note that based on the required accuracy and resource availability, the results can be refined by confining the approximation and search region within a neighborhood of interest around the selected design, updating the approximation models and obtaining a more accurate optimal solution set to fully represent the region of interest.

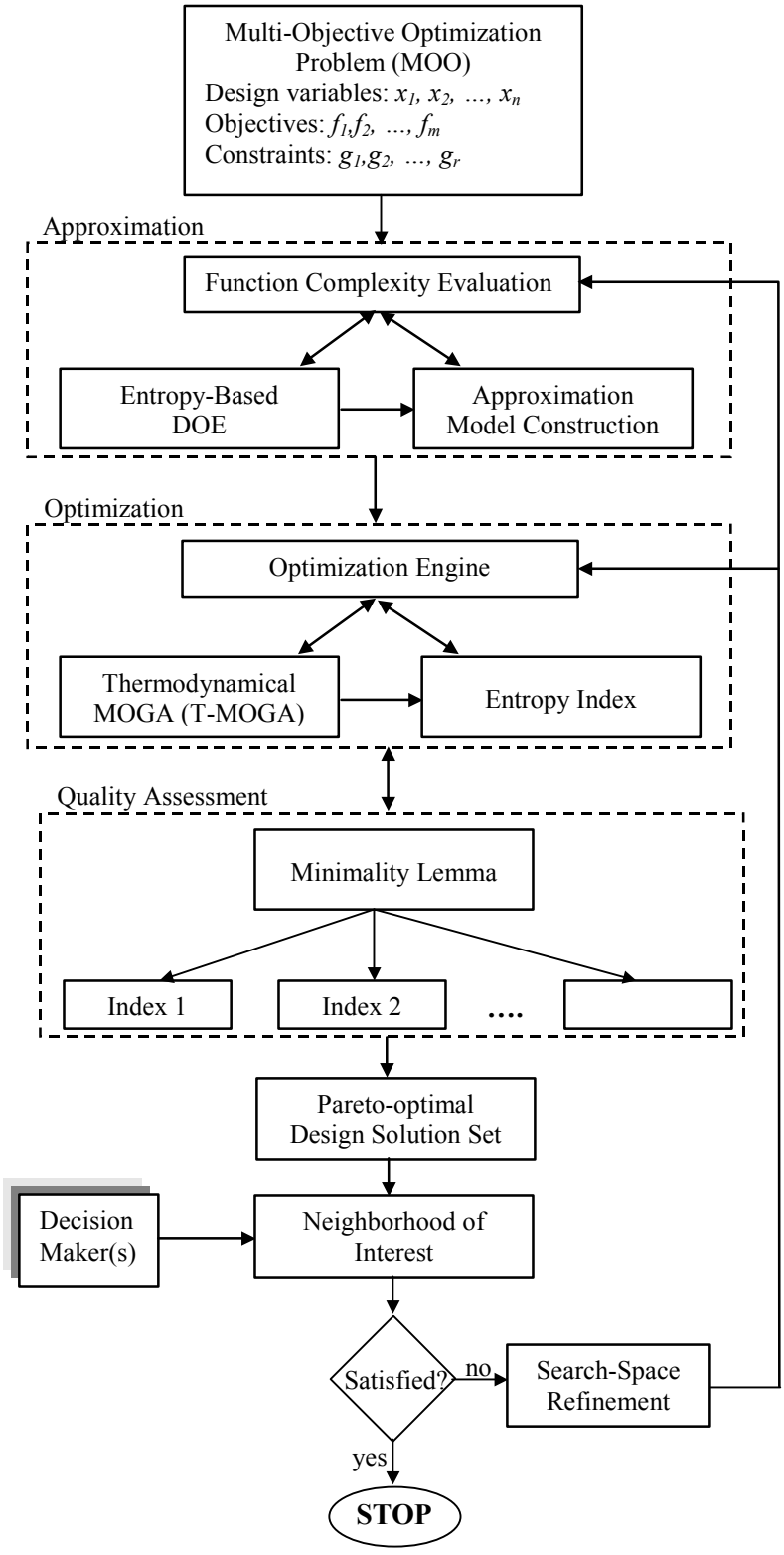


Figure 7.1: Proposed approximation and optimization framework

7.2 CRASHWORTHINESS DESIGN OF FRONT-END OF A PICKUP TRUCK

By most measures simulation of a crash event involving a typical vehicle is a computationally intensive task. A complete detailed computer model of a passenger vehicle typically involves $10^5 – 10^6$ degrees of freedom and one performance evaluation may require many hours or days of computer time. The complexity of the problem makes design optimization of even a small component of the vehicle a challenging undertaking. The physical phenomena involved are highly non-linear and the simulated responses tend to be quite ill-behaved which in turn makes a direct application of most optimization techniques very time-consuming or even impossible.

7.2.1 Crashworthiness Problem Definition

A detailed multi-purpose finite element model of a 1994 Chevrolet C-2500 pick-up truck was developed at the National Crash Analysis Center at George Washington University (Bedewi et al. 1996). This model is the first of its kind developed specifically to address vehicle safety issues, including front and side performance. As shown in Figure 7.2, the bumper-rail assembly from this model is used in this analysis.

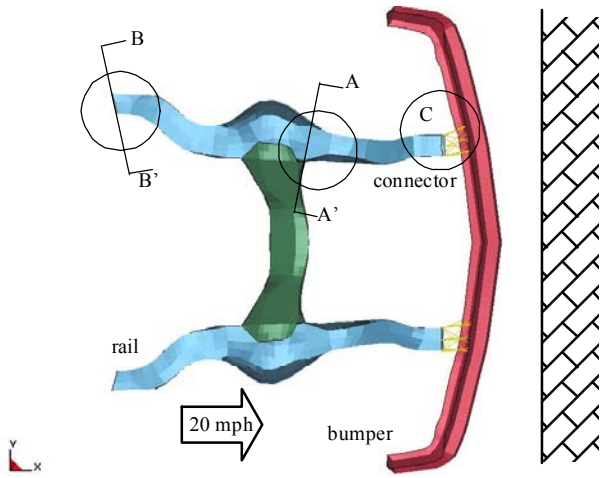


Figure 7.2: The bumper-rail assembly

The assembly consists of the bumper, the left and right rails, and the cross rail connector. The rail mountings, which connect the bumper and the rails, were replaced with connectors modeled by beam elements whose purpose is simply to engage/disengage the rail and the bumper. Lumped masses are attached at the rear end of the rails, at section B-B' (see Appendix-III for additional details). The assembly is moving forward at a 20 mph when it hits a rigid wall. The analysis was performed at Michigan State University using a standard finite element package for large deformation and impact analysis (LSDyna). Each analysis run cost about 15 minutes on a Sun Ultra 80 workstation. The meta-modeling and optimization part of this research was performed at The University of Maryland.

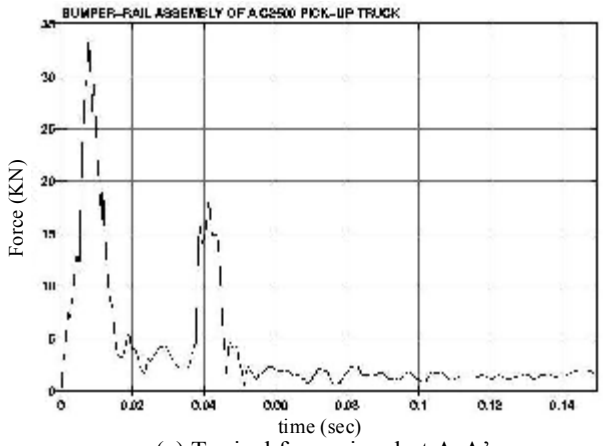
In design for enhanced crashworthiness, the objective is to improve the protection of the passenger, e.g., by controlling the accelerations experienced by the passenger and the deformation of the structure in the immediate vicinity of the passenger. In our case this is accomplished indirectly by (1) minimizing the

maximum force, *F*, transmitted through the rail (measured at section A-A'); and (2) minimizing the (X-) displacement *D* of the section at B-B'. Sample signals for *F* and *D* are shown in Figures 7.3 (a) and (b), respectively, as a function of time (t=0 at the moment of impact). Therefore, the objective functions are:

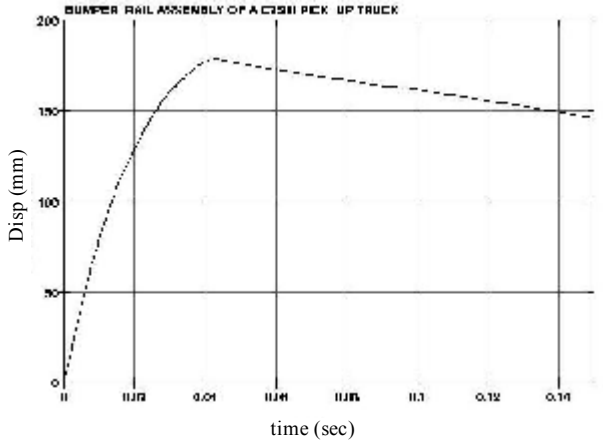
$$f_1 = \max_{t>0} | F(t) | \tag{7.1}$$

and

$$f_2 = \max_{t>0} | D(t) | \tag{7.2}$$



(a) Typical force signal at A-A'



(b) Typical displacement at B-B'

Figure 7.3: Typical performance functions in rail assembly problem

There are two design variables in this problem:

x_1 = collapse strength of connector material (*MPa*)

x_2 = sheet metal thickness of rail forward of section A-A' (*mm*)

The first variable acts as a "switch" that controls the timing of the failure of the mountings that connect the bumper and the rail. The second variable has a strong effect on both the amount of deformation and the accelerations transmitted to the rear of the vehicle. The overall problem can be formulated as a two-objective minimization problem:

$$\begin{aligned} &\text{Minimize } (f_1, f_2) \\ &\text{subject to:} \\ &1 \le x_1 \le 70 \\ &2 \le x_2 \le 5 \end{aligned} \tag{7.3}$$

Given the computational cost of evaluating f_1 and f_2 (i.e., 15 minutes for a single function call), a direct application of a multi-objective optimization technique could be very time consuming. For instance, a typical multi-objective genetic algorithm requires about 1000 function evaluations to obtain a good approximation of the Pareto-frontier, i.e., about 11 days of computation for this problem! The situation becomes even worse if we select more design variables and choose additional objective and/or constraint functions.

As mentioned before, the computational complexity of calculating the objective functions in Equation 7.3 prohibits a direct application of multi-objective optimization techniques. Therefore, we use SMAXED to construct a global approximation for the

two objectives: f_1 and f_2, which will be used later to find an approximation for the Pareto-optimal frontier.

The upper and lower bounds for the two variables of the problem, x_1 (collapse strength) and x_2 (sheet metal thickness), are given in equation 7.4. The following is a linear mapping of these variables onto $[0,1]^2$:

$$z_1 = (x_1 - 1)/69 \quad ; z_2 = (x_2 - 2)/3 \tag{7.4}$$

A 50x50 grid is constructed in the design space (a total of 2500 nodes). We also limit our experimentation for each objective to 39 experiments, as 3 blocks of 13 experiments each. Recall that the response evaluation for each experiment takes about 15 minutes on a Sun Ultra 80, so, 39 experiments for each objective function is estimated to take about 10 hours of computation. The step-by-step approximation of the two objective functions using SMAXED is demonstrated in the following sections.

7.2.2 f_1: Maximum Force (over time)

At the beginning of the approximation process the non-informative (stationary assumption) holds.

STEP 1 (1st block of 13 experiments) – The goal is to find a set of 13 nodes such that the determinant of the covariance matrix is maximized. To estimate $\theta_0 = 1/(2\tau^2)$ (where τ is the standard deviation of the correlation function), we assume that for a node located exactly in the middle of the design space (i.e. point (0.5,0.5) in $[0,1]^2$), the 6τ boundaries of the normal correlation function coincides with the boundaries of the design space: $6\tau = 1$, and thus $\theta_0 = 18$. To analyze the sensitivity of the outcome with respect to θ_0, we also examined other values for θ_0: θ_0=1, 10, 36; and observed that the

first iteration of the SMAXED process remains unchanged for these values of θ_0, while the second and third iterations change only slightly. Figure 7.4 shows the first block of 13 points that maximizes the covariance matrix. The symmetry of this design is due to the fact that there is no prior information about the behavior of the response function at this stage (i.e., no response function evaluations have been performed yet). Compare this to Figure 2.1 (c).

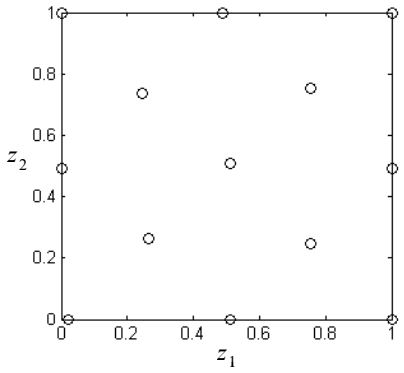
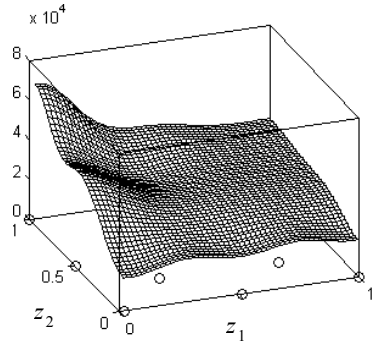
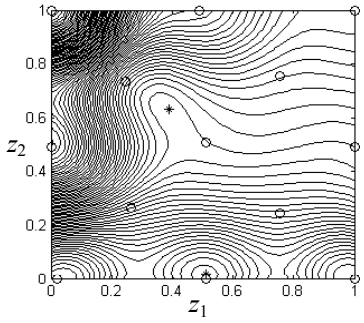


Figure 7.4: The first block of 13 experiments for f_1

STEPS 2 and 3 (1st block) – The response values for the above 13 experiments are computed and listed in Appendix-III. An initial approximation model is constructed accordingly as shown in Figures 7.5 (a) and (b). Note that the mean of the response values from the first block of experiments is used as an estimate for μ (μ=31,745 N, for the response values listed in Appendix-III).



(a)



(b)

Figure 7.5: (a) Intermediate approximation model based on 13 experiments; (b) The contour plot of this approximation model and its optima in the design domain (denoted by '*'). Small circles denote the first block of 13 experiments

STEP 4 (1st block) – The optima of the above approximation model are found (Figure 7.5(b)). Since the approximation model is very fast to compute, the optimum nodes are obtained by simply comparing the values of the approximation model at each node with its immediate neighbors. (This requires little computational burden. The L_i's are computed from Definition 1 for each node in the design space.

STEPS 5, 6 and 7 (2nd block) – A 26x26 covariance matrix is constructed such that the first 13 rows and columns correspond to the 1st block of experiments. 13 new nodes are

selected in the design space such that the covariance of the matrix is maximized.

Figure 7.6 shows the first and second blocks of experiments, a total of 26 experiments.

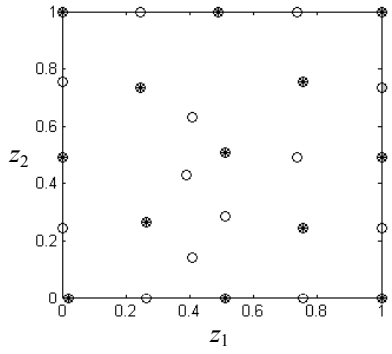


Figure 7.6: Solid circles mark the first block of 13 experiments (already performed) while hollow circles are new experiments (to be performed)

Since 13 more experiments need to be designed to obtain a total of 39 experiments, we return to STEP 2 and repeat the process. (The response values are listed in Appendix-III.) Figure 7.7 shows the second intermediate approximation model (based on the 26 experiments).

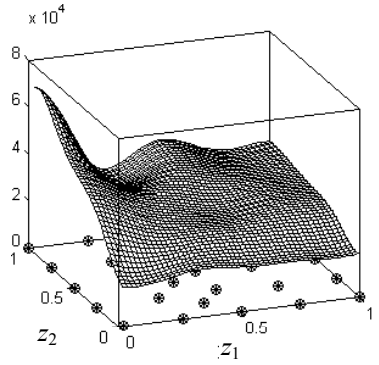


Figure 7.7: The second block of experiments are evaluated and a new approximation model is obtained based on these 26 experiments.

Again, we find the optima of the above intermediate approximation function, and by maximization of the covariance matrix, the 3rd block of experiments is designed and shown in Figure 7.8.

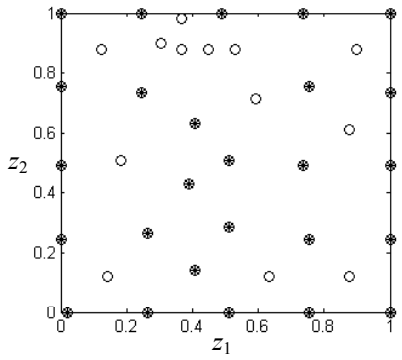


Figure 7.8: Hollow circles mark the 3rd block of 13 experiments (to be performed)

The response values for these new experiments are evaluated and a final approximation model is obtained based on all 39 experiments (See Figure 7.9).

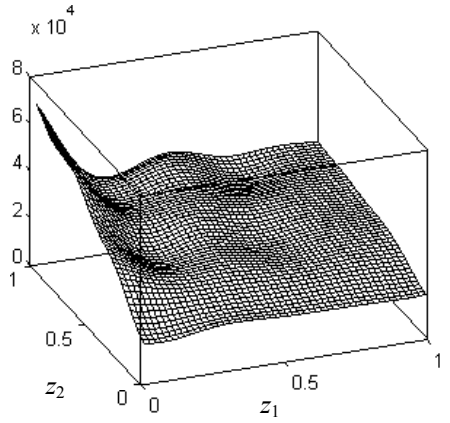
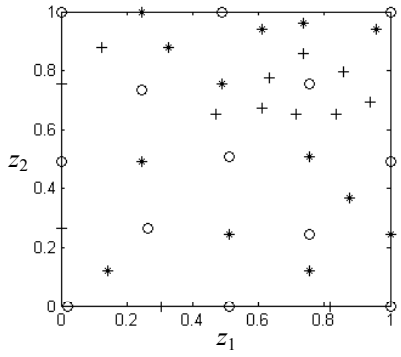


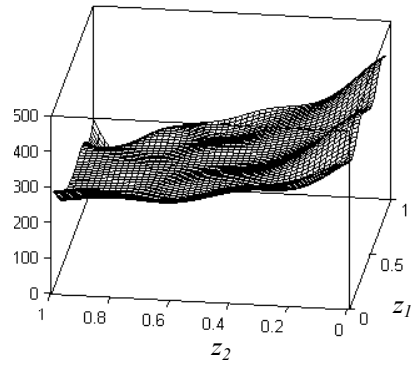
Figure 7.9: Approximation model for the maximum force objective function (f_1) based on 39 experiments.

7.2.3 f_2: Maximum Displacement (over time)

In a similar fashion for the maximum displacement response function (f_2), we design a total of 39 experiments as 3 blocks of 13 experiments each. Figure 7.10 (a) and (b) show the final design and the resulting approximation model. (See Appendix-III for numerical data.)



(a)



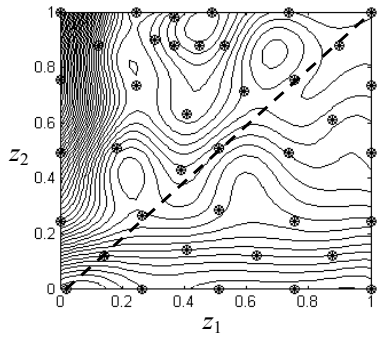
(b)

Figure 7.10: (a) Empty circles, asterix, and plus signs, mark the 1st , 2nd, and 3rd blocks of experiments respectively. (b) Approximation model for the Maximum Displacement objective function (f_2) based on 39 experiments.

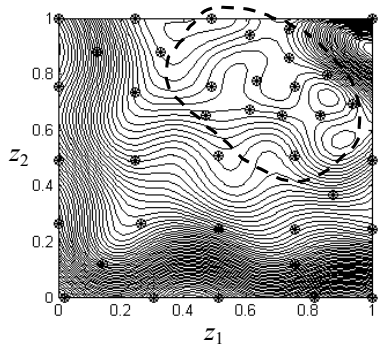
7.2.4 Discussions and Verification

Figures 7.11 (a) and (b) show the contour plots of the final approximation models for f_1 and f_2, respectively. The adaptive nature of SMAXED can be easily

observed in these figures: The concentration of experiments is slightly higher in the irregular regions of the design space where uncertainty is higher (larger variance) due to the fluctuations in the response function. This is a key feature of SMAXED that helps obtain a more accurate approximation model with fewer experiments.



(a) Maximum Force (f_1)



(b) Maximum Displacement (f_2)

Figure 7.11: There is a higher concentration of experiments in the irregular regions of the design space where uncertainty is higher. For example in (a), compare the number of experiments in the upper-left and lower-right triangles.

To verify the accuracy of the above approximation models, one can select a random sample of points in the design space and compute the deviation between the actual response function and the approximation model. The Root-Mean-Square-Error (RMSE) of the approximation model can then be estimated:

$$RMSE = \left\{ \sum [\hat{y}(\mathbf{z}_i) - y(\mathbf{z}_i)]^2 / n \right\}^{1/2} \tag{7.5}$$

where n is the size of the random sample. We selected a random sample of 40 points in $[0,1]^2$, and estimated the normalized RMSE (RMSE divided by sample mean) of both approximation model:

$$RMSE(f_1) / Mean(f_1) \sim 5\% \tag{7.6}$$

$$RMSE(f_2) / Mean(f_2) \sim 6\%$$

This indicates that the error is well within the acceptable range. In fact, the finite element model itself is expected to deviate from the actual physical phenomenon by at least that much. Note that the above verification process by itself is computationally very expensive (40 random experiment to obtain the above RMSE takes about 10 hours to compute). One could either skip the computational burden of the verification process altogether, or take advantage of these additional random experiments, and incorporate them into the approximation model to improve accuracy. In this paper, however, the approximation models are based only on 39 experiments (3 blocks of 13 experiments each, obtained from SMAXED).

7.2.5 Approximation-Assisted Multi-Objective Optimization

Going back to the original optimization problem of equation 7.3, we can now replace the computaionally expensive objective functions with the corresponding surrogate models, i.e.,

$$\text{Minimize } (\hat{y}_{f_1}, \hat{y}_{f_2})$$

$$\text{subject to:} \tag{7.7}$$

$$0 \le z_1 \le 1$$

$$0 \le z_2 \le 1$$

Solving equation 7.7 using T-MOGA provides an approximation for the Pareto-optimal frontier of the original problem. As mentioned before, the meta-models are very fast to compute, and therefore, application of T-MOGA with a total of 1000 function calls is computationally feasible. Had we decided to use the original response functions directly, 1000 function evaluations could have taken about 11 days of computational time on a Sun Ultra 80; while with the approximated models of Section 4, the optimization process ended in less than 5 minutes on a Pentium-4, 2.26GHz, 256 MB RAM (plus 10 hours for running 39 experiments per objective function (for a total of 20 hours) on a Sun Ultra 80). The resulting Pareto optimal solution set is shown in Figure 7.12.

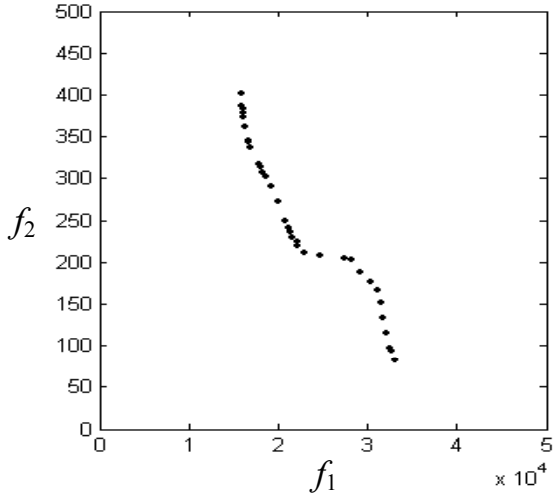


Figure 7.12: Approximated Pareto-optimal solution set based on 39 experiments for each objective

Figure 7.12 demonstrates the tradeoff between the two objectives, maximum force and maximum displacement. In Figure 7.13, the entropy index (Chapter 5) is

plotted with number of function evalutions. T-MOGA increases the diversity of solutions as the population evolves, until it saturates i.e. entropy does not improve anymore. The non-dominated solution set of Figure 7.12 corresponds to the last generation of solutions in Figure 7.13 (function evaluations=1000).

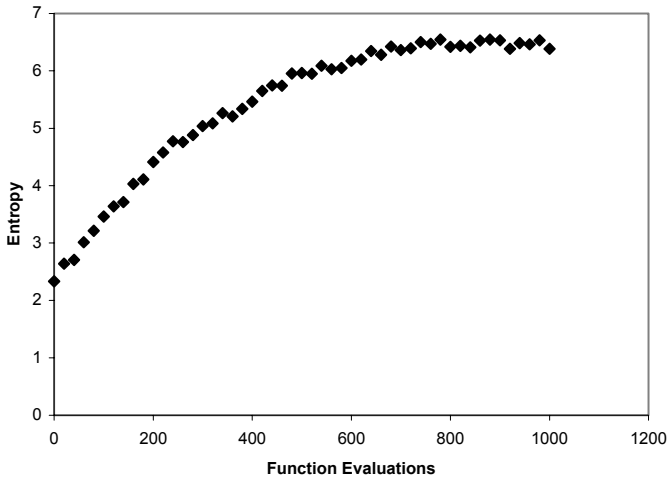


Figure 7.13: Entropy history of T-MOGA

7.3 CONCLUDING REMARKS

A finite-element model is developed for simulation of a crash event involving the front-end assembly of a 1994 Chevrolet C-2500 pickup truck. The simulation model is computationally expensive, taking about 15 minutes of computational time for a single evaluation. For such a problem, a direct application of multi-objective optimization techniques that require numerous function evaluations (e.g., T-MOGA) becomes computationally intractable. Therefore, SMAXED is used to construct a meta-model for the finite-element model, with T-MOGA to find an estimate to the Pareto optimal solutions. Using this approach, it is shown that the accuracy of the meta-models is within an acceptable range while the computational costs are reduced dramatically.

The next chapter is devoted to a brief summary and contributions of this research. It also includes suggestions for future research.

CHAPTER 8

CONCLUSIONS

This dissertation has four research thrusts in the context of multi-objective optimization for engineering design problems: 1) Meta-Modeling Research Thrust: Produced a meta-modeling approach for expensive engineering simulations; 2) Multi-Objective Genetic Algorithm Research Thrust: Produced a multi-objective evolutionary optimization technique with expansion operators to preserve diversity of solutions; 3) Minimality of Quality Indexes Research Thrust: Produced a framework for selection of a minimal set of quality indexes to assess the obtained solution sets; and 4) Entropy Index Research Thrust: Produced an entropy quality index to measure the distribution quality of a solution set.

An overview of these four research thrusts are given in the following:

- *Meta-Modeling Research Thrust*: Sequential Maximum Entropy Design (SMAXED) was developed as a new approximation technique. The main idea behind SMAXED is to sequentially use the information from all performed computer experiments (i.e., setting the input design variables and evaluating the outputs) to identify 'irregularities' in the response function behavior, and design new experiments such that the most accurate global meta-model can be obtained with fewer experiments (which means less computational effort).

- *Multi-Objective Genetic Algorithm Research Thrust*: Thermodynamical MOGA (T-MOGA) was developed. T-MOGA manipulates the diversity of a population of solutions according to Maxwellian model in order to achieve a uniform and thorough representation of the Pareto frontier.

- *Minimality of Quality Indexes Research Thrust*: A theoretical framework for selection of a minimal set of quality indexes is developed that exhaustively addresses all desired aspects of quality without redundancy. Two key lemmas -- Concordance and Compatibility Lemmas -- were proved to derive a key Theorem: Minimality Theorem. This theorem asserts that in order to construct a minimal set of quality indexes that exhaustively and distinctively address all aspects of quality (as put forward by a decision-maker), one must provide a one-to-one correspondence between quality indexes and quality aspects (expressed in the form of excellence relations).

- *Entropy Quality Index Research Thrust*: It measures the distribution quality of a solution set over the Pareto frontier. This new quality index is very easy to compute, and provides a scalar that can be used to compare different MOGAs in terms of their ability to produce diverse solution sets.

In Chapters 6 and 7, a set of engineering and numerical test examples were solved to further demonstrate the applicability and performance of the proposed methodology. Chapter 7 also suggested a general framework: approximation-assisted multi-objective optimization of expensive engineering simulations. The proposed

framework was demonstrated in the context of an engineering design problem: Crashworthiness design of front-end of a pickup truck.

8.1 CONCLUDING REMARKS

In the following subsections, concluding remarks are given for each research thrust.

8.1.1 Meta-Modeling Research Thrust

The SMAXED approach of this thesis was applied to several engineering and numerical test problems. In Section 6.1.1 for instance, the HSCT aircraft design was used as a test bed to compare the performance of SMAXED with other meta-modeling techniques. In this large-scale problem (with 25 design variables) SMAXED showed a significant improvement over other techniques. Therefore,

- The SMAXED approach is applicable to small- and large-scale problems alike (in terms of number of design variables).

In the HSCT problem, SMAXED showed a better (lower) RMSE and Average Error for both small and large number of experiments (60 and 180 experiments respectively). However, polynomial response surface meta-model showed a lower Maximum Error than other interpolating techniques for larger number of experiments (smaller overshoot). The same observation was reported by other researchers for response surface analysis technique as compared to interpolative approaches (such as SMAXED and kriging). This does not mean that response surface analysis is a better approach than interpolative techniques because on average all interpolative techniques

that were used in the HSCT aircraft design example (Section 6.2.1) performed better than response surface analysis for both small and large number of experiments.

- On average, SMAXED performed better than all other techniques (including response surface analysis) for both large and small number of experiments.

One shortcoming, however, that requires further research emerged during application of SMAXED to the crashworthiness case study of Chapter 7. The two response functions of this design problem, i.e. maximum force and maximum displacement were treated as two separate 'models', and were approximated separately. In practice, however, these two response functions were obtained from one code base. So anytime an experiment was performed to calculate the maximum force, the code base also evaluated maximum displacement with no additional computational cost. In Chapter 7, however, this additional (and perhaps very valuable) information was simply disregarded in creating the meta-models.

- In Section 7.2.4 we suggested to simply incorporate the additional information obtained from running simulations with multiple response outputs into the kriging model (equation 2.9) to improve accuracy. While in some cases this ad-hoc approach could improve the accuracy, it is not necessarily an optimal approach.

8.1.2 Multi-Objective Genetic Algorithm Research Thrust

In Section 6.2.2, T-MOGA is applied to a set of test problems. These test problems were specifically designed to introduce difficulty in a certain aspect of MOGA performance (e.g., convergence of MOGAs). Through these examples it was

shown that T-MOGA creates more diverse solution sets while it converges to the Pareto frontier almost as fast as other prominent techniques in the literature (convergence was measured using the *S* quality index).

- The expansion operators in T-MOGA manipulate a higher diversity of solutions without decreasing the convergence rate of MOGA

One shortcoming of T-MOGA lies in its computationally expensive constraint handling technique. For the first two constrained test problems in Section 6.2.3, the constraints had analytical forms that could be easily used to find the reflection of solution points. This, however, is not possible in general for problems with more complex constraints. Computational methods, such as the one suggested in Section 5.2.6, may become computationally expensive because of many functions calls. For constrained problem C3 in Section 6.2.3 we used an alternative approach that was based on freezing (not expanding) solutions that otherwise would become infeasible with expansion. Although this ad-hoc approach still yielded an increasing entropy, it makes the analogy between T-MOGA and Maxwellian System -- that is guaranteed to achieve maximum *H* -- incomplete. One other approach (similar to the suggested approach of Section 5.2.3) is to estimate the feasible domain by a box-like region. The solution points can be reflected easily from the walls of this box-like region (Each wall is perpendicular to one axis). This latter approach is easy to implement and maintains the good properties of Maxwellian systems, however, it may render parts of the population infeasible because of those portions of the box-like region that are indeed infeasible.

8.1.3 Minimality of Quality Indexes Research Thrust

In the two-relation example of Section 4.3, Minimality Theorem (Chapter 4) was used to find a minimal set of quality indexes that address both aspects of quality in that example (i.e. closeness and span of solutions over the Pareto frontier). Indeed, Minimality Theorem provides a recipe for selection a comprehensive yet non-redundant set of quality indexes, that is

- One can form many combinations of quality indexes to assess the performance of population-based optimization techniques. Minimality Theorem, however, can significantly narrow the search by eliminating all combinations that do not satisfy the stated conditions and thus, are guaranteed to be either correlated or non-comprehensive.

In that example (Section 4.3), however, we encountered some difficulties in finding a minimal set of indexes. In short,

- In short, a decision-maker is not always able to express his/her preferences mathematically and accurately in terms of excellence relations.
- Even if excellence relations are formed, it is not always easily possible to find a quality index that is compatible with that excellence relation.

For the example of Section 4.3, we suggested ways to deal with these issues. One possible approach could be to modify the excellence relations such that a corresponding compatible quality index can be found. This was shown for the example of the coverage excellence relation where this relation was modified such that a corresponding compatible index was found. This is, however, not an easy or well-defined task in general. Especially, for more complicated cases where decision-maker

expresses many excellence relations, modifying excellence relations and quality indexes such that a one-to-one correspondence can be built becomes increasingly difficult.

8.1.4 Entropy Quality Index Research Thrust

The Entropy Index of Chapter 5 was used in the gearbox design problem of Section 6.3.1 to monitor the entropy of population during MOGA process. The entropy index has two properties that make it very desirable for monitoring the quality during optimization (not only after optimization is done and the results are obtained):

- Linear computational complexity
- Linearly additivity, i.e., if a solution point is added or removed from a population, entropy can be update easily without repeating the computation for the entire population.

Entropy index, however, has an overhead computation (once at the beginning of the process) to compute and store the influence function. Also, assigning the value of entropy for a certain solution set does not carry an absolute meaning and can only be used relatively in comparison with other solution sets.

8.2 MAIN CONTRIBUTIONS OF THE THESIS

The main challenge in approximation of computationally-expensive (deterministic) computer simulations is to create surrogate models that are as accurate as possible using minimum number of experiments. Bayesian approximation is particularly appropriate for this class of problems because it generates interpolative and

global surrogate models. SMAXED was developed in this dissertation as an extension to Bayesian maximum entropy approximation:

- *SMAXED is the first sequentially-adaptive Bayesian meta-modeling technique in the sense that it sequentially takes into account response function behavior and adjusts a priori parameters for the next iteration of experiments.*

This new technique is applied to several small- to large-scale problems (Chapters 2, 6, and 7). The results showed a significant improvement in the accuracy of the obtained surrogate models (recall Section 6.1.1).

In the Multi-Objective Genetic Algorithm research thrust, there are two major challenges: 1) Obtaining a diverse set of solutions that represent the entire Pareto frontier; and 2) Fast convergence to the Pareto frontier. A novel approach to manipulating diversity in MOGAs was introduced in this thesis and a new algorithm (T-MOGA) was developed accordingly. It is shown through many examples that while T-MOGA performs much better than other algorithms in terms of achieving a better diversity of solutions (first challenge), it converges to the Pareto frontier at least as fast as other approaches.

- *The major contribution of this part of research is to introduce expansion operators in T-MOGA that expands the population according to Maxwellian model. Therefore (according to Boltzmann's maximum H theorem), T-MOGA guarantees to achieve the highest-entropy equilibrium state, i.e., steady state (time-invariant) with uniform density.*

In quality indexes, the major challenge is to select a handful of indexes such that: 1) they are not redundant (too many indexes); and 2) they are comprehensive (too few indexes).

- *The main contribution of this part of research is to give a formal definition for minimal sets of quality indexes. It is argued that an appropriate set of quality indexes must be minimal with respect to a set of given excellence relations. More importantly, The Minimality Theorem is proved as a necessary condition for minimal sets. This condition will significantly narrow the search for a minimal set among a myriad of quality indexes in the literature.*

Finally, the major challenge in evaluating the diversity of a solution set is to develop a quality index that is: 1) computationally easy to compute so that it can be constantly evaluated throughout the optimization process; and 2) it encapsulates all aspects of diversity such as uniformity and coverage into one scalar.

- *In this thesis, a new diversity assessment quality index is developed -- referred to as the entropy index. The entropy index takes advantage of a formal similarity between Shannon entropy function and distribution quality of solution points. Because of its low computational complexity entropy index can be evaluated throughout MOGA process.*

Through many examples, it is shown that entropy index can play a major role in comparison study of different MOGAs.

8.3 DISCUSSIONS AND FUTURE RESEARCH DIRECTIONS

In this section, several suggestions are given for future research. Some of these suggestions are based on currently known shortcomings of the proposed approach (found during application to various examples and summarized in Section 8.3). Other suggestions are made based on extended research and recommendations by other researchers.

SMAXED: In Section 8.2.1, it was mentioned that the SMAXED approach of Chapter 2 does not account for simulation models that create several response values with one run. That is,

- Approximation of simulation models that simultaneously compute two or more response values are left mainly untreated in this thesis. This is an area of research that could significantly improve the accuracy of meta-models in real-world application where one code base (e.g. the finite element crash model of Chapter 7) computes several response values simultaneously with no additional computational cost.

Another area of further research for SMAXED lies in estimating fixed parameters for the kriging model:

- While maximum likelihood estimation can become computationally burdensome, the ad-hoc alternative 6σ approach of Chapter 2 can become a source of error. The accuracy of the meta-model depends on a priori selection of appropriate values for parameters such as prior mean, standard deviation, and correlation (although it was shown through examples that the design is not that sensitive to this value).

T-MOGA: In a study by Deb, Mohan and Mishra (2003), a *sparsity measure* is introduced (again based on the entropy idea of this research) and a new MOGA is developed that increases this measure. Their comparisons show that their new MOGA performs relatively well and achieves a good convergence and diversity. This triggers a possible direction for future research:

- In Chapter 2, it is shown that for a phenotypic MOGA with expansion operators (that move solution points in the objective space) a Maxwellian system such as T-MOGA is the only expansion pattern that guarantees uniformity and time-invariance (maximum entropy). However, it is not known to us whether a genotypic entropy-based approach that manipulates variables in the bit level or in the fitness assignment stage can perform better than T-MOGA. This issue deserves a deeper investigation.

Minimality of Quality Indexes: In chapter 4, we provided a platform to classify and select quality indexes based on their compatibility with the excellence relations. In practice, however, a certain set of excellence relations (and the corresponding set of minimal quality indexes) can be used in different occasions. In other words, one can create a library of possible excellence relations and profile quality indexes based on their compatibility with the excellence relations in this library.

- A library of possible excellence relations and the corresponding compatible quality indexes could make the job of a decision-maker much easier. For each particular instance of comparison study, a decision-maker goes through this library (catalog of excellence relations) and selects a handful of them. For each

excellence relation, one quality index is chosen. Correlation among these quality indexes is checked, if there is minimum or no correlation one can proceed with performance assessment. Otherwise, one can choose a different set of quality indexes (compatible with the same excellence relations) and repeat the process.

Entropy Index: In a recent study, Deb and Jain (2002) investigated the merits and setbacks of the entropy index (first published in Farhang-Mehr and Azarm 2002). While they found our proposed approach of projection on a hyper-plane and applying entropy function to be 'a good approach', they also identified several shortcomings, as in the following:

- Deb and Jain (2002) state that entropy index (as developed in Farhang-Mehr and Azarm 2002) largely depends on the chosen value of σ, i.e. if σ is chosen too large, the density hypersurface is always flat and the entropy is always high. On the other hand, if σ is chosen to be too small, the density surface is always peaky and entropy index does not change significantly from one set to another.
- Deb and Jain (2002) also maintain that entropy index may be low (that indicates bad distribution) for a degenerate[6] but perfectly distributed Pareto-optimal solution set, as compared to a non-optimal solution set with the same dimension as that of the objective space.

[6] Degenerate Pareto is defined as a Pareto frontier that has a dimension lower than that of the objective space.

Based on the above-mentioned shortcomings, Deb and Jain (2002) proceeded to suggest an extension to the entropy index of this thesis by modifying the definition of H and defining a value function for quality of distribution, denoted by $m(x)$. Although this modified approach seems to have none of the above-mentioned shortcomings, however, it faces a new challenge: The value function, $m(x)$ must be chosen intuitively for every problem. This creates difficulties in high-dimensional problems where intuitive approaches tend to be misleading, if not impossible. Therefore, one potential future research direction can be identified as follows:

- Both estimation guideline of Section 5.2.4 and value function approach are subjective, and may become a source of error in higher dimensional problems. An entirely objective approach (with no arbitrary parameters or functions) is therefore very desirable and deserves a much deeper study.

APPENDIX-I

TERMINOLOGY AND DEFINITIONS

A multi-objective optimization problem with m objective functions ($m > 1$) can be shown in the following minimization form.

$$\begin{aligned} &\text{Minimize} \quad \mathbf{f}(\mathbf{x}) = \{f_1(\mathbf{x}),...,f_i(\mathbf{x}),...,f_m(\mathbf{x})\} \\ &\text{s.t.} \quad \mathbf{x} \in D \\ &D = \{\mathbf{x} : g_j(\mathbf{x}) \le 0, j = 1,...,J; h_k(\mathbf{x}) = 0, k = 1,...,K\} \end{aligned} \tag{A.I-1}$$

where $\mathbf{x}$ is an n-dimensional design variable vector, and D is the set of all such vectors that satisfy the constraints. g_j's and h_k's are the inequality and equality constraints, respectively. Below, the definitions of some terms used in this dissertation are given.

Variable space: An n-dimensional space whose coordinates are the design variables.

Objective space: An m-dimensional space whose coordinates are the objectives.

Feasible Design Space: The set of all feasible points to an optimization problem is called feasible design space, denoted by D.

Dominance: Let $\mathbf{x}_1, \mathbf{x}_2 \in D$ be two design points. Then, $\mathbf{x}_1$ dominates $\mathbf{x}_2$ iff $f_i(\mathbf{x}_1) \le f_i(\mathbf{x}_2)$ for all i=1,…, m, with strict inequality for at least one i (Steuer, 1986).

Pareto-Optimal Solution (or Pareto Solution): A feasible solution point, namely $\mathbf{x}^* \in D$, is Pareto optimal iff there does not exist another solution point, $\mathbf{x} \in D$, such that $f_i(\mathbf{x}) \le f_i(\mathbf{x}^*)$ for all i=1,…, m, with strict inequality for at least one i (Steuer, 1986; Miettinen, 1999).

Pareto Frontier: The set of all Pareto optimal solutions to the multi-objective optimization problem of Equation A.I-1 is referred to as Pareto frontier.

Normalization of Objectives: The objective functions of a generic multi-objective optimization problem are usually incommensurable in the sense that they have different units and therefore any comparison or aggregation among them is meaningless. To address this issue, in Chapters 3 and 5, the objectives are normalized with respect to two reference points: The *ideal* and *nadir* points.

Ideal/good points: The ideal point is defined as a point in the objective space, whose components are obtained by constrained minimization of each of the objective functions individually, that is (Miettinen, 1999):

$$\text{Minimize } f_i(\mathbf{x}) \qquad \text{subject to: } \mathbf{x} \in D; \text{ for } i=1,\ldots, m. \tag{A.I-2}$$

In practice, however, performing several optimization routines to obtain the ideal point is often time-consuming. In most cases, an experienced designer is able to estimate this ideal point even without optimizing the objectives. In this paper we refer to the ideal point or its best estimation as a *good point*. The good point is basically a lower bound for all objectives and should be selected such that it dominates all solution points.

Nadir/bad points: The nadir point is the opposite of the ideal point, i.e., the upper bounds of the Pareto frontier (Miettinen, 1999). Finding the nadir point is still an open research problem in general. There are a few attempts in the literature to further improve the estimation of the nadir point, but most of them require several optimization routines (see, for instance, Lewandowski and Wierzbicki, 1989; Korhonen and Steuer, 1997). Instead, in this thesis, we arbitrarily overestimate the

ranges of objectives such that no design point is encountered that violates the estimated upper bounds. These estimated upper bounds for the objective functions constitute a point in the objective space that is referred to as the bad point. All solution points are normalized in the objective space with respect to these two reference points: good and bad points.

Non-Dominated Set (NDS): Population-based multi-objective techniques usually generate a finite set of design points to the optimization problem of Equation A.I-1. If we denote the population of all feasible design points by *S*, then an NDS (or Observed Pareto Set) is defined as the set of all $\mathbf{x} \in S$, such that there does not exist another design point in *S* that dominates **x**. Note that a design solution point in an NDS is not necessarily Pareto optimal. However, a good optimization algorithm will provide an NDS that approximates the Pareto frontier as closely as possible.

Inferiority Index (*InfI*): Assume two NDSs *A* and *B*. We denote the number of solutions in set *A* that are not dominated by any solution in Set *B* by n_A^{nd}. Similarly the number of solutions in Set *B* that are not dominated by any solution in Set *A* is denoted by n_B^{nd}. The inferiority index of set *A* as compared to *B* (denoted by *InfI*(*A,B*)) is the ratio of n_A^{nd} over the sum of n_A^{nd} and n_B^{nd} , i.e. $InfI\,(A,B)= n_A^{nd}/(n_A^{nd} + n_B^{nd})$. Similarly: $InfI\,(B,A)= n_B^{nd}/(n_A^{nd} + n_B^{nd})$. Clearly this is a symmetric metric because: $InfI(A,B)+InfI(B,A)=1$. In order to rewrite the above metric in a homogeneous format according to the guideline of Chapter 4, the following mapping is necessary: $Q_{Infl}(A,B)=1/2-InfI(A,B)$ which is homogeneous and symmetric.

Size of Dominated Space (*S*): (Fonseca and Fleming 1996, also Zizler 1999) Consider a non-dominated solution set $A=\{\mathbf{a}_1, \ldots,\mathbf{a}_i, \ldots\}$ in an objective space that is normalized

with respect to good and bad points. The size of dominated space by set A, denoted by V_A, is defined as the volume of the union of hypercubes $\{C_1, \ldots, C_i, \ldots\}$, where C_i is a hypercube whose two opposite vertices are $\mathbf{a}_i$ and good (ideal) point. To form a symmetric and homogenous quality index (as required in Chapter 4), we write: $S(A,B)=V_B-V_A$. Therefore, $S(A,B)>0$ indicates that set A is closer to the Pareto frontier, i.e. there is a smaller dominated volume between solution set A and good point, as compared to solution set B.

APPENDIX-II

OVERVIEW OF SINGLE- AND MULTI-OBJECTIVE GENETIC ALGORITHMS

Among different evolutionary algorithms, Genetic Algorithms (GAs) -- originally introduced by Holland (1975) -- have been receiving significant attention because of their wide range of applicability to mixed discrete-continuous optimization problems. The main idea behind Genetic Algorithms is described next (For more information on GA and MOGA, refer to Holland 1975; Goldberg 1989; Fonseca and Fleming 1995; Van Veldhuizen and Lamont 2000; and Deb 2001).

GENTIC ALGORITHMS

GAs are based upon the process of natural evolution and selection. GA starts with an initial set of randomly selected binary strings -- referred to as *initial population*. Each individual in this population is a binary representation of a possible design. Binary strings are partitioned into several groups of bits, encoding values for design variables. Similar to natural evolution, bits are referred to as *genes* and solution strings made of genes are called *chromosomes*. GAs are iterative processes, i.e., during each iteration the individuals of the population are evaluated on the basis of certain *fitness* criterion to determine their goodness (or fitness) for survival and reproduction. There are several different fitness assignment criteria and selection methods. For

instance in roulette wheel selection, two individuals among the population are selected as *parents* with probabilities proportional to their fitness. In other words, the better fitness an individual has, the higher probability that it will be chosen. The operators are called crossover and mutation and inversion, which are analogous to their counterparts in the natural evolution process. These genetics operations are applied to the parents to combine their genes and produce new individual(s) -- referred to as *offspring*. The offspring are then evaluated and a new generation is formed by including some of the parents and the offspring on the basis of their fitness values such that the size of the population remains the same throughout the procedure.

As the tendency is to select high fitness individuals to generate offspring and the likelihood of survival of the better individuals is higher (i.e. the weak individuals are more likely to be discarded from the next population), therefore the next generation is likely to have the individuals with higher fitness values. Hence, the fitness of the entire population improves over the generations. That means the overall solution quality improves. We need to keep in mind that at the same time some bad genes are also inherited from the previous generation even though the probability of that event is quite low. In this way, the algorithm does not get stuck in a local optimum. This is a very important mechanism in GAs that usually results in globally optimum solutions. The framework of the described procedure is shown in the Figure AII-1.

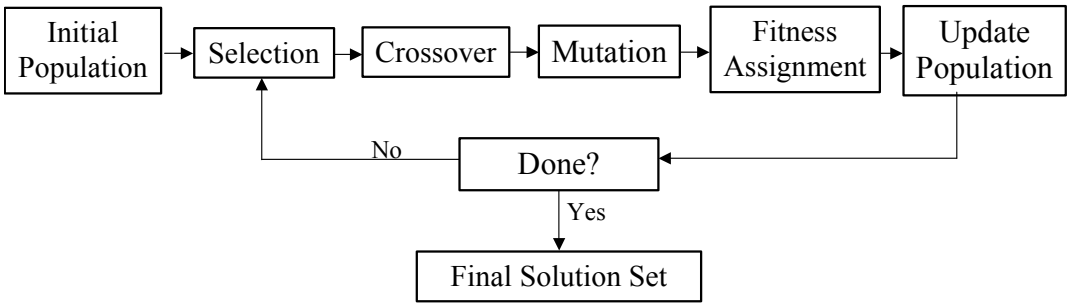


Figure AII-1: Basic operations of a GA

The three genetic operators that are used to produce offspring are discussed in the following:

1. **Crossover:** Crossover generates offspring by combining chromosomes of two individuals at a time. This could be achieved by choosing a random cut point and generating the offspring by combining the left segment of one parent with right segment of the other. However, after doing so, some blocks may be repeated while some other blocks may get deleted. This problem has been dealt with in many different ways. The amount of crossover is controlled by the crossover rate (or crossover probability) which is defined as the ratio of the number of offspring produced in each generation to the population size. The crossover rate determines the ratio of the number of searches in the region of high average fitness to the number of searches in the other region. The schematic view of the crossover operation is shown in Figure AII-2.

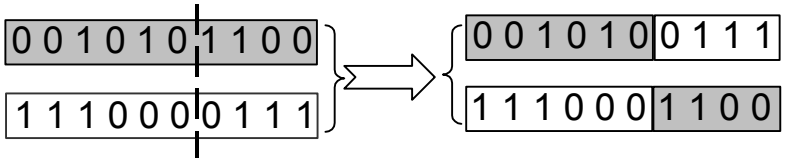


Figure AII-2: Crossover operator

2. **Mutation:** This operator is not directly responsible for producing new offspring, but it causes incremental random changes in the offspring produced by crossover operator. The most commonly used mutation is pair-wise interchange. This is the process by which new genes, which did not exist in the original generation, can be generated. The mutation rate (or mutation probability) is defined as the percentage of the total number of genes in the population, which are mutated in each generation. Lets keep in mind that it should be carefully chosen so that it can introduce more useful genes, and at the same time do not destroy the resemblance of offspring to their parents. The schematic view of the mutation operation is shown in Figure AII-3.

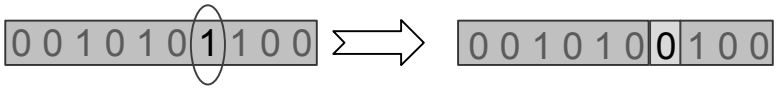


Figure AII-3: Mutation Operator

3. **Selection:** After generating offspring, a portion of individuals is chosen based on fitness criteria to survive for the next generation. There are many

such selection functions used by various researchers. In competitive selection, all the parents and offspring compete with each other and the fittest individuals are selected so that the population remains constant. In random selection, the individuals for the next generation are randomly selected so that the population remains constant. This could be advantageous considering the fact that by selecting the fittest individuals the population converges to individuals that share the same genes and the search might not converge to the global optimum. However, if the individuals are chosen randomly, there is no way to gain improvement from older generation to new generation. By compromising two methods stochastic selection make selections with probabilities based on fitness of each individual (This latter technique is used in the MOGAs of this paper; See Table 3.1.)

MULTI-OBJECTIVE GENETIC ALGORITHMS

Since GAs are population-based approaches, they have a tremendous advantage over classical search techniques for handling multi-objective optimization problems (Deb, 2001). In fact, GAs can be easily modified to find many Pareto solution points in one single run. Unlike classical techniques that require weighting, e-constraint or other methods to transform multiple objectives into a single scalar, MOGAs are capable of handling multiple objectives only by redefining fitness criteria. In Sections 3.1.1 and 3.1.2 we briefly mentioned several existing MOGAs and their characteristics. Here, a baseline approach to MOGA is presented that was used as the basis for many recent

MOGAs including Fonseca and Fleming's MOGA (FFMGA); MOGA-NA; and T-MOGA (See Fonseca and Fleming 1993).

The fitness assignment criterion is based on dominance number, as defined in the following.

Dominance Number: Consider a finite feasible set for a multi-objective optimization problem, $\{\mathbf{P}_1, \mathbf{P}_2, ...\}$. The dominant number of an element, $\mathbf{P}_i$, in this set is defined as the number of points, $\mathbf{P}_j (j \neq i)$, such that $\mathbf{P}_j \succ \mathbf{P}_i$. Clearly, the dominance number of non-dominated solution is zero.

A rank number of (1+dominance number) is assigned to each solution. Therefore, all non-dominates solutions are assigned a rank of 1. Then a fitness is assigned to each solution based on this ranking, as described below:

- All solutions are sorted in ascending order of their ranking.
- A raw fitness is assigned to each individual based on a mapping function (usually linear) that maps the best ranking solution to N and the worst ranking solution to 1.
- The raw fitness of solutions that have similar ranks (but now have different raw fitness values) is averages. This average is assigned as fitness value to each individual with that ranking.

The rest of MOGA process (including Stochastic Universal Selection, Mutation and Crossover) is similar to that of single-objective GA. In this MOGA, non-dominated solutions are emphasized (are assigned higher fitness values). Therefore, the population tends to evolve towards Pareto frontier. There are many variations of this same concept that also incorporate niche punishment or fitness sharing approaches to

also increase diversity in the population. Some researchers also suggested specific techniques to handle constraints (other than classical techniques such as penalty function) .For a thorough review of these approaches and their merits refer to Deb (2001).

APPENDIX-III

ADDITIONAL INFORMATION FOR CRASHWORTHINESS DESIGN PROBLEM

All components except the connectors are modeled in LSDyna using shell elements. The material is elastic-perfectly plastic with elastic modulus of 210 GPa, Poisson's ratio of 0.3, yield strength of 100 MPa, and density of 7890 kg/m^3. Sheet-metal thicknesses are:

Rail (rear of A-A'): 3.14 mm

Bumper: 2.7 mm

Cross rail connector: 3.61 mm

Beams (in bumper/rail connector) have a solid square cross section, 20 mm per side.

Lumped masses totaling 100 Kg are attached at the rear of the rails, at section B-B'.

RESPONSE VALUES FROM MAXIMUM FORCE SIMULATION

	Experiment	$x_1(MPa)$	$x_2(mm)$	$f_1(N)$
1st Block	1	68.62	4.94	33319
	2	2.38	2	15903
	3	1	4.94	68141
	4	68.62	2	18017
	5	35.5	3.5	29288
	6	1	3.44	45929
	7	68.62	3.44	29302
	8	34.12	4.94	38500
	9	35.5	2	18226
	10	17.56	4.16	32282

	11	18.94	2.78	26765
	12	52.06	4.22	31485
	13	52.06	2.72	25528
2nd Block	14	35.5	2.84	26246
	15	28.6	3.86	34670
	16	50.68	4.94	33042
	17	1	2.72	33951
	18	68.62	4.16	31531
	19	27.22	3.26	29739
	20	1	4.22	58604
	21	18.94	2	18670
	22	17.56	4.94	32391
	23	68.62	2.72	25501
	24	52.06	2	18002
	25	50.68	3.44	29057
	26	28.6	2.42	22812
3rd Block	27	25.84	4.58	36573
	28	36.88	4.58	35695
	29	25.84	4.88	36409
	30	31.36	4.58	37952
	31	13.42	3.5	28744
	32	21.7	4.64	34567
	33	41.02	4.1	31908
	34	10.66	2.36	21983
	35	43.78	2.36	21796
	36	61.72	4.58	32492
	37	60.34	2.36	21785
	38	60.34	3.8	30422
	39	9.28	4.58	38809

RESPONSE VALUES FROM MAXIMUM DISPLACEMENT SIMULATION

	Experiment	$x_1(MPa)$	$x_2(mm)$	$f_2(mm)$
1st Block	1	68.62	4.94	180.48
	2	2.38	2	401.77
	3	1	4.94	281.98
	4	68.62	2	400.54
	5	35.5	3.5	215.99
	6	1	3.44	295.98
	7	68.62	3.44	213.93
	8	34.12	4.94	175.32
	9	35.5	2	400.25

	10	17.56	4.16	197.66
	11	18.94	2.78	262.68
	12	52.06	4.22	192.1
	13	52.06	2.72	259.58
2nd Block	14	50.68	4.82	183.64
	15	17.56	4.94	184.59
	16	68.62	2.72	258.89
	17	52.06	2.36	298.53
	18	34.12	4.22	193.18
	19	35.5	2.72	262.16
	20	52.06	3.5	213.16
	21	17.56	3.44	227.7
	22	42.4	4.76	184.79
	23	10.66	2.36	301.01
	24	65.86	4.76	95.9
	25	23.08	4.58	180.1
	26	60.34	3.08	232.42
3rd Block	27	64.48	4.04	194.72
	28	1	4.22	282.11
	29	42.4	3.98	198.86
	30	1	2.78	331.57
	31	21.7	2	387.28
	32	57.58	3.92	198.26
	33	32.74	3.92	200.76
	34	58.96	4.34	189.32
	35	43.78	4.28	192.54
	36	56.2	2	398.35
	37	9.28	4.58	245.8
	38	49.3	3.92	199.16
	39	50.68	4.52	187.67

REFERENCES

Andrews, F.C., 1963, *Equilibrium Statistical Mechanics*, Wiley, New York

Balabanov, V., Kaufman, M., Knill, D.L., Haim, D., Golovidov, O., Giunta, A.A., Haftka, R.T., Grossman, B., Mason W.H., and Watson, L.T., 1996, "*Dependence of Optimal Structural Weight on Aerodynamic Shape for a High Speed Civil Transport,"* Proceedings of the 6th AIAA/NASA/USAF Multidisciplinary Analysis and Optimization Symposium, Bellevue, WA, Sept. 1996, pp. 599-612

Barthelemy, J.F., Haftka, R.T., 1993, "Recent Advances in Approximation Concepts for Optimum Structural Design", Structural Optimization, Vol. 5(3), pp. 129-144

Bedewi, Z.A., Kan, A., Marzougui, D., 1996 "Validation of a Non-linear Finite Element Vehicle Model Using Multiple Impact Data," ASME Winter Annual Congress and Exposition, Atlanta GA

Binh, T. T., Korn, U., 1997, "A MultiObjective Evolution Strategy for Constrained Optimization Problems", 3rd International Conference on Genetic Algorithms, Brno, Czech Republic, pp. 176-182

Bosman, P. A. N., Thierens, D., 2003, "The Balance Between Proximity and Diversity in Multiobjective Evolutionary Algorithms," IEEE Transactions on Evolution Computation, Vol. 7(2), pp. 174-188

Brillouin, L., 1962, *Science and Information Theory*, 2nd Edition Academic Press, NY

Casella, G., and Berger, R. L., 1990, *Statistical Inference*, Duxbury Press, Belmont, California.

Camponogara, E., Talukdar, S., 1997, A Genetic Algorithm for Constrained and Multi-Objective Optimization. Jarmo T. A. (ed.), In: *3rd Nordic Workshop on Genetic Algorithms and Their Applications (3NWGA)*, pp. 49-62, Vaasa, Finland

Coello Coello, A., 1999, "An Updated Survey of Evolutionary Multiobjective Optimization Techniques: State of the Art and Future Trends", Proceedings of Congress on Evolutionary Computation, Washington D.C., Vol. 1, pp. 3-12

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

Colorni, A., Dorigo, M., Maniezzo, V., 1992, "Distributed Optimization by Ant Colonies", Proceedings of the First European Conference on Artificial Life, Paris, France, pp. 134-142

Cunha, a. G., Oliviera, P., Covas, J., 1997, "Use of Genetic Algorithms in Multi-criteria optimization to solve industrial problems, Proceedings of the Sevenths International Conference on Genetic Algorithms, San Francisco, CA, pp. 682-688

Currin, C., Mitchell, M., Morris, M., and Ylvisaker, D., 1991, "Bayesian Prediction of Deterministic Functions, with Applications to the Design and Analysis of Computer Experiments," Journal of the American Statistical Association, Vol. 86, pp. 953-963

Deb, K., 1998, *Multi-Objective Genetic Algorithms: Problem Difficulties and Construction of Test Problems*, Technical Report CI-49/98, Department of Computer Science/LS11, University of Dortmund, Germany

Deb, K., 2001, *Multi-Objective Optimization Using Evolutionary Algorithms*, John Wiley and Sons, Chichester, UK

Deb, K., Jain, S., 2002, "Running Performance Metrics for Evolutionary Multi-Objective Optimization," Proceedings of Simulated Evolution and Learning (SEAL-02), pp. 13-20

Deb, K., Manikanth, M., Mishra, S., 2003, "Towards a Quick Computation of Well-Spread Pareto-Optimal Solutions," Proceedings of the Second International Conference on Evolutionary Multi-Criterion Optimization (EMO 2003), Faro, Portugal pp. 222-236

Farhang-Mehr, A., Wu, J., and Azarm, S., 2001, "Some Preliminary Results on the Development and Comparison of a New Multi-Objective Genetic Algorithm," Proceedings of the ASME 2001 Design Engineering Technical Conference, Pittsburgh, PA

Farhang-Mehr, A., Azarm, S., 2002, "Diversity Assessment of Pareto Optimal Solutions: An Entropy Approach", CD Proceedings of the IEEE World Congress on Computational Intelligence, Honolulu, Hawaii.

Farhang-Mehr, A., Azarm, S., Diaz, A., and Ravisekar, A., 2003, "Approximation Assisted Crashworithness Design of Front-End of a Pickup Truck," Proceedings of the ASME 2003 Design Engineering Technical Conference, Chicago, IL

Fay, J. A., 1965, *Molecular Thermodynamics*, Addison Wesley, Reading, MA

Fonseca, C. M., Fleming, P. J., 1993, "Genetic Algorithms for Multiobjective Optimization: Formulation, Discussion and Generalization", The Fifth International Conference on Genetic Algorithms, San Mateo, CA, pp. 416-423

Fonseca, C. M., Fleming, P. J., 1995, "An Overview of Evolutionary Algorithms in Multiobjective Optimization", Evolutionary Computation, Vol. 3(1), pp. 1-16

Fonseca, C.M., Fleming, P.J., 1996, "On the performance assessment and comparison of stochastic multiobjective optimizers," Fourth International Conference on Parallel Problem Solving from Nature, Berlin, Germany, pp. 584-593

Fonseca, C. M., Fleming, P. J., 1998, "Multiobjective Optimization and Multiple Constraint Handling with Evolutionary Algorithms-Part II: Application Example", IEEE Transactions on Systems Man. and Cybernetics, Vol. 28(1), pp. 38-47

Fukunaga, K., and Hostetler, L.D., 1975, "The Estimation of the Gradient of a Density Function with Applications in Pattern Recognition", IEEE Transactions on Information Theory, Vol. 21(1), pp. 32-87

Geoffrion, A.M., Dyer, J.S., Feinberg, 1972, "An Interactive Approach For Multiple Criteria Optimization with an Application to the Operation of an Academic Department", Management Science, Vol. 19(4), pp. 357-368

Golinski, J., 1970, "Optimal Synthesis Problems Solved by Means of Nonlinear Programming and Random Methods", Journal of Mechanisms, Vol. 5, pp. 287-309

Giunta, A.A., Balabanov, V., Kaufman, M., Burgee, S., Grossman, B., Haftka, R.T., Mason, W.H., and Watson, L.T., 1995, "*Variable-Complexity Response Surface Design of an HSCT Configuration*," Proceedings of ICASE/LaRC Workshop on Multidisciplinary Design Optimization, Hampton, VA

Giunta, A., 1997, *Aircraft Multidisciplinary Optimization Using Design of Experiments Theory and Response Surface Modeling Methods*, Ph.D. Dissertation, Department of Aerospace and Ocean Engineering, Virginia Tech, Blacksburg, VA

Giunta, A., Watson, L. T. and Koehler, J., 1998, "A Comparison of Approximation Modeling Techniques: Polynomial Versus Interpolating Models," 7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis & Optimization, St. Louis, MI

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

Haar, D. T., 1964, *Elements of Statistical Mechanics*, Reinehardt, New York

Haimes, Y.Y., Wismer, D.A., Lasdon, L.S., 1971, "On a Bicriterion Formulation of the Problems of Integrated System Identification and System Optimization", IEEE Transactions on Manufacturing Systems, pp. 296-97

Hansen, M. P., Jaszkiewicz, A., 1998, Evaluating the Quality of Approximations to the Non-dominated Sets, Technical Report IMM-REP-1998-7, Technical University of Denmark, Lyngby, Denmark

Horn, J., Nafpliotis, N., Goldberg, D. E., 1994, "A Niched Pareto Genetic Algorithm for Multiobjective Optimization", Proceedings of the First IEEE Conference on Evolutionary Computation, Piscataway, NJ, pp. 82-87

Horn, J., 1997, *Multicriteria Decision Making*, in *Handbook of Evolutionary Computation*, Institute of Physics Publishing, Bristol, UK

Holland, J.H., 1975, *Adaptation in Natural and Artificial Systems*, The University of Michigan Press, Ann Arbor, MI

Hinneburg, A., and Keim, D.A., 1998, "An Efficient Approach to Clustering in Large Multimedia Databases with Noise," Proc. 4th Int. Conf. on Knowledge Discovery and Data Mining (KDD98), New York, pp. 58-65

Ishibuchi, H., and Murata, T., 1996, "Multi-objective Genetic Local Search Algorithm", Proceedings of 1996 IEEE International Conference on Evolutionary Computation, Piscataway, NJ, pp. 119-124

Jaynes, E. T., 1957, "Information Theory and Statistical Mechanics," Phys. Rev., Vol. 106, pp. 620-630

Jessop, A., 1995, *Informed Assessments: An Introduction to Information, Entropy and Statistics*, Hemel Hempstead, Ellis Horwood, Chichester, U.K.

Kennedy, J., and Eberhart, R. C., 1995, "Particle Swarm Optimization", Proceedings of the IEEE conference on Neural Networks, Piscataway, NJ, pp. 1942-1948

Kirkpatrick, S., Gelatt, C. D. Jr., and Vecchi, M. P., 1983, "Optimization by Simulated Annealing", Science, Vol. 13(4), pp. 563-571

Knowles, J. D., 2002, *Local-Search and Hybrid Evolutionary Algorithms for Pareto Optimization*, Ph.D. Dissertation, University of Reading, Department of Computer Science, Reading, U.K

Knowles, J. D., Corne, D. W., 2002, "On Metrics for Comparing Non-Dominated Sets", Proceedings of the 2002 Congress on Evolutionary Computation Conference, Honolulu, Hawaii, pp. 711-716

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

Koehler, J. R., Owen, A. B., 1996, *Computer Experiments*, in *Handbook of Statistics*, Elsevier Science, New York, pp. 261-308

Korhonen, P., S., Steuer, R. E., 1997, "A Heuristic for Estimating Nadir Criterion Values in Multiple Objective Linear Programming," Operations Research, Vol. 45(5), pp. 751-757

Levin, R. D., Tribus, M., 1979, *The Maximum Entropy Formalism*, The MIT Press, Cambridge, MA

Lewandowski, A., Wierzbicki, A.P., 1989, *Aspiration Based Decision Support Systems*, Lecture Notes in Economics and Mathematical Systems, Vol. 331

Mardia, K. V., and Marshall, R. J., 1984, "Maximum Likelihood Estimation of Models for Residual Covariance in Spatial Regression," Biometrika, Vol. 71, pp. 135-146

McKay, M., Beckman, R., Conover, W., 1979, "A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code", Technometrics, Vol. 21(2), pp. 239-245

Messac, A., 1996, "Physical Programming: Effective Optimization for Computational Design," AIAA Journal, Vol. 34 (1), pp. 1490158.

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

Myers, R. H., Montgomery, D. C., 1995, *Response Surface Methodology: Process and Product Optimization Using Designed Experiments*, John Wiley & Sons, New York, NY

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

Noble, B., and Daniel, J. W., 1988, *Applied Linear Algebra*, Third Edition, Prentice-Hall, Englewood Cliffs, NJ

Obayashi, S., Takahashi, S., and Takeguchi, Y., 1998, "Niching and Elitist Models for MOGAs", Fifth International conference on Parallel Problem Solving from Nature, Berlin, Germany, pp. 260-269

Owen, A. B., 1992, "Orthogonal Arrays for Computer Experiments, integration, and visualization", Statistical Science, Vol. 2, pp. 439-452

Pham, D. T., and Karaboga, D., 2000, *Intelligent Optimization Techniques, Genetic Algorithms, Tabu Search, Simulated Annealing, and Neural Netweorks,* Springer-Verlag, Berlin, Heidelburg, New York

Prugel-Bennett, A., and Shapiro, J. L., 1994, "An Analysis of Genetic Algorithms Using Statistical Mechanics", Phys. Rev. Lett., Vol. 72(9), pp. 1305-1309

Prugel-Bennett, A., and Shapiro, J. L., 1997, "The dynamics of a genetic algorithm in simple randomising systems", Physical Review Journal D, Vol. 104(1), pp. 75-114

Reynolds, B. J., 2000, *Hybrid Multi-Objective Genetic Algorithm with Heuristics Based on Archived Information*. M.S. Thesis, University of Maryland, College Park, MD,

Rudolph, G., 1998, "On a Multi-objective Evolutionary Algorithm and its Convergence to the Pareto Set", IEEE International Conference on Evolutionary Computation (ICEC'98), Piscataway, NJ, pp. 511-516

Sacks, J., Welch, W. J., Mitchell, T. J., and Wynn, H. P., 1989, "Design and Analysis of Computer Experiments", Statistical Science, Vol. 4(4), pp. 409-435

Sayin, S., 2000, "Measuring the Quality of Discrete Representations of Efficient Sets in Multiple Objective Mathematical Programming", Mathematical Programming, Vol. 87, pp. 543-560

Schaffer, J. D., 1985, "Multiple Objective Optimization with Vector Evaluated Genetic Algorithms", Genetic Algorithms and Their Applications, Proceedings of the First International Conference on Genetic Algorithms, Hillsdale, NJ, pp. 93-100

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

Schott, J. R., 1995, *Fault Tolerant Design Using Single and Multicriteria Genetic Algorithm Optimization*, M.S. Thesis, Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, MA

Shannon, C. E., 1948, "A mathematical theory of communication," Bell System Technical Journal, Vol. 27, pp. 379-423 and 623-656 (July and October)

Simpson, T. W., Pepliski, J., Koch, P. N., Allen, J. K., 1997, "On the Use of Statistics in Design and Implications for Deterministic Computer Experiments," Design Theory and Methodology, DTM'C97, Sacramento, CA

Simpson, T. W., Mauery, T. M., Korte, J. J., Mistree, F., 1998a, "Comparison of Response Surface and Kriging Models for Multidisciplinary Design Optimization", 7th

AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis & Optimization, St. Louis, MI

Simpson, T. W., Allen, J. K., Mistree, F., 1998b, "Spatial Correlation Metamodels for Global Approximation in Structural Design Optimization", ASME Design Engineering Technical Conferences, Design Automation Conference (DETC98), Atlanta, GA

Srinivas, N., Deb, K., 1994, "Multiobjective Optimization using Non-dominated sorting in Genetic Algorithms", Evolutionary Computation, Vol. 2(3), pp. 221-248

Srivastava, A., Hacker, K., Lewis, K., 1999, "Investigation of Different Approximation Techniques in the Design of the High Speed Civil Transport Aircraft", The Third World Congress of Structural and Multidisciplinary Optimization, Buffalo, NY

Steuer, R. E., 1986, *Multiple Criteria Optimization: Theory, Computation, and Application*, John Wiley & Sons, New York

Tamaki, H., Kita, H., Kobayashi, S., 1996, "Multi-objective Optimization by Genetic Algorithms: A Review", Proceedings of IEEE International Conference on Evolutionary Computation (ICEC'96), Piscataway, NJ, pp. 517-522

Valenzuela-Rendon, M., Uresti-Charre, E., 1997, "A Non-Generational Genetic Algorithm for Multiobjective Optimization", Proceedings of the Seventh International Conference on Genetic Algoriths, San Francisco, CA, pp. 658-665

Van Veldhuizen, D. A., Lamont, G. B., 1998, "Evolutionary Computation and Convergence to a Pareto front", Proceedings of the Third Annual Conference on Genetic Programming, San Francisco, CA, pp. 22-25

Van Veldhuizen, D. A., 1999, *Multiobjective Evolutionary Algorithm: Classification, Analyses and New Innovations*, Ph.D. Dissertation, Dept. of Electrical and Computer Engineering, Air Force Institute of Technology, Wright-Patterson AFB, Ohio

Van Valdhuizen, D.A., Lamont G.B., 2000, "Multiobjective Evolutionary Algorithms: Analyzing the State-of-the-Art", Evolutionary Computation, Vol. 8(2), pp. 125-147

Wolpert, D. H., Macready, W. G., 1997, "No Free Lunch Theorems for Optimization", IEEE Transactions on Evolutionary Computation, Vol. 1(1), pp. 67-82

Wu, J., and S. Azarm, 2001, "Metrics for Quality Assessment of a Multiobjective Design Optimization Solution Set", Transactions of the ASME, Journal of Mechanical Design, Vol. 123, pp. 18-25

Wu, J., 2001, "Quality Assisted Multiobjective and Multidisciplinary Genetic Algorithms", Ph.D. Dissertation, Mechanical Eng. Dept., University of Maryland, College Park, MD

Zitzler, E., Thiele, L., 1998, "Multiobjective Optimization Using Evolutionary Algorithms – A Comparative Study", Proc. 5th International Conference, Parallel Problem Solving from Nature, Amsterdam, The Netherlands, pp. 292-301

Zitzler, E., 1999, *Evolutionary Algorithms for Multiobjective Optimization: Methods and Applications*, PhD Thesis, Swiss Federal Institute of Technology (ETH) Zurich

Zitzler, E., Deb, K., Thiele, L., 2000, "Comparison of multiobjective evolutionary algorithms: Empirical Results", Evolutionary Computation Journal, Vol. 8(2), pp. 125-148

Zitzler, E., Laumanns, M., Thiele, L., Fonseca, C. M., Grunert da Fonseca V., 2002, "Why Quality Assessment of Multiobjective Optimizers Is Difficult", Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2002), New York, NY, pp. 666-674

Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C., Fonseca, V., 2003, "Performance Assessment of Multiobjective Optimizers: An Analysis and Review," IEEE Transactions on Evolutionary Computation, Vol. 7(2), pp. 117-132