Introducing Robustness in Multi-Objective Optimization

Kalyanmoy Deb deb@iitk.ac.in
Kanpur Genetic Algorithms Laboratory (KanGAL), Indian Institute of Technology Kanpur, Kanpur, PIN 208016, INDIA

Himanshu Gupta higupta3@in.ibm.com
IBM India Research Lab, Block 1, Indian Institute of Technology, Hauz Khas, New Delhi, PIN 110016, India

Abstract
In optimization studies including multi-objective optimization, the main focus is placed on finding the global optimum or global Pareto-optimal solutions, representing the best possible objective values. However, in practice, users may not always be interested in finding the so-called global best solutions, particularly when these solutions are quite sensitive to the variable perturbations which cannot be avoided in practice. In such cases, practitioners are interested in finding the *robust* solutions which are less sensitive to small perturbations in variables. Although robust optimization is dealt with in detail in single-objective evolutionary optimization studies, in this paper, we present two different robust multi-objective optimization procedures, where the emphasis is to find a robust frontier, instead of the global Pareto-optimal frontier in a problem. The first procedure is a straightforward extension of a technique used for single-objective optimization and the second procedure is a more practical approach enabling a user to set the extent of robustness desired in a problem. To demonstrate the differences between global and robust multi-objective optimization principles and the differences between the two robust optimization procedures suggested here, we develop a number of constrained and unconstrained test problems having two and three objectives and show simulation results using an evolutionary multi-objective optimization (EMO) algorithm. Finally, we also apply both robust optimization methodologies to an engineering design problem.

Keywords
Multi-objective optimization, evolutionary algorithms, robust solutions, Pareto-optimal solutions, global and local optimal solutions.

1 Introduction

For the past decade and more, the primary focus of research and applications in the area of evolutionary multi-objective optimization (EMO) has been to find the globally best Pareto-optimal solutions (Deb, 2001; Coello et al., 2002). Such solutions are non-dominated to each other. Simply stated, there exists no other solution in the entire search space which dominates any of these solutions. From a theoretical point of view, such solutions are of utmost importance in a multi-objective optimization problem. However, in practice, often a solution can not be implemented with an arbitrary accuracy and the implemented solution may be somewhat different from the theoretical global optimal solution. If a global optimal solution is *sensitive* to such variable perturbation in its vicinity, the implemented solution may result in a different set of objective

©2006 by the Massachusetts Institute of Technology

values from that of the theoretical optimal solution. Thus, from a practical standpoint, such solutions are not of much importance and the emphasis must be made in finding *robust* solutions, which are comparatively less sensitive to variable perturbations in their neighborhoods.

In single-objective evolutionary optimization, a number of studies have been devoted to finding robust solutions. Parmee (1996) suggested a hierarchical strategy of searching several high performance regions in a fitness landscape simultaneously. Tsutsui and Ghosh (1997) presented a mathematical model for obtaining robust solutions using the schema theorem for single-objective genetic algorithms. Branke (1998) suggested a number of heuristics for searching robust solutions. In another study, Branke (1998) also pointed out key differences between searching optimal solutions in a noisy environment and searching for robust solutions. In an optimization with a noisy environment, the objective function computation is not precise and is usually associated with some error, mainly due to the error associated with modeling the objective functions. However, robust optimization deals with uncertainties in variables which in effect cause the objective functions to be stochastic. But Branke's studies were made in the context of single-objective optimization. Later, Branke and Schmidt (2000) suggested a number of methods for alternate fitness estimation. Stagge (2001) suggested a computationally efficient procedure in which only a few best solutions in the population are evaluated multiple times to get an idea of the true objective value. Jin and Sendhoff (2003) considered the issue of finding robust solutions in a single-objective optimization problem as a multi-objective optimization problem with the objectives being maximizing robustness and performance.

In the context of multi-objective optimization, Teich (1998) handled errors in modeling objective functions by modifying the domination operator. By considering the modeling error to vary uniformly over a given range of values for each objective, a probabilistic domination operator was defined. Hughes (2001) followed a similar principle but used a sampling method to learn the probability distribution of noise arising from the modeling error. However, none of these studies considered sensitivities in objective values due to variable uncertainties in a multi-objective scenario.

Recently, we suggested two different ways of introducing robustness in multi-objective optimization and showed some proof-of-principle results on unconstrained test problems (Deb and Gupta, 2005). In this paper, we investigate in detail the two approaches, extend the concepts to constrained optimization and demonstrate the usefulness of robust multi-objective optimization by solving an engineering design problem. Specifically, we concentrate on the effect of parameter uncertainty in defining the optimal solution of a multi-objective optimization problem and leave the issue of noise associated with the modeling of objective functions for a later study. We make an effort to extend an existing averaging approach (Tsutsui and Ghosh, 1997; Branke, 1998) for finding robust solutions in single-objective optimization for multi-objective optimization. Essentially, in this approach, instead of optimizing the original objective functions, we optimize the *mean effective* objective functions computed by averaging a representative set of neighboring solutions. Solutions which are less sensitive to such local perturbations will fair well in terms of the mean effective objective values and the resulting Pareto-optimal front may turn out to be the robust frontier. To illustrate how this approach works, we suggest a number of different test problems and employ NSGA-II (elitist non-dominated sorting genetic algorithm) (Deb et al., 2002) to find the robust frontier. Thereafter, we present a new definition of robustness by optimizing the original objectives but adding a constraint limiting the extent of functional change

by local perturbations to a user-defined value. Thus, the second approach is more pragmatic and the user has control of the desired level of robustness on the obtained solutions. The differences between these two robust procedures and fundamental differences between global optimization and robust optimization principles in the context of multi-objective optimization are clearly demonstrated through an analysis of the simulation results. Then, we extend the robust optimization idea for constrained problems and show limited simulation results. Finally, to demonstrate the usefulness of the proposed methodology in practice, we take up an engineering design problem and show the effect of robust optimization on the optimized solutions.

In the remainder of the paper, Section 2 introduces the concept of robustness in multi-objective optimization and stresses its importance in practice. Sections 3 and 4 discuss the first robust optimization scheme and simulation results obtained using NSGA-II. Section 5 discusses the second robust approach and presents simulation results. Section 6 extends the idea for constrained problem solving and Section 7 applies the proposed methodologies to an engineering design problem. Finally, a conclusion of this study is presented in Section 8.

2 Robustness in Single-Objective Optimization

For a single-objective optimization of the following type:

$$\left.\begin{array}{ll} \text{Minimize} & f(\mathbf{x}), \\ \text{subject to} & \mathbf{x} \in \mathcal{S}, \end{array}\right\} \tag{1}$$

where $\mathcal{S}$ is the feasible search space, a robust solution is defined as the one which is insensitive (to a limit) to the perturbation in the decision variables in their neighborhood. Let us consider Figure 1. Of the two minimum solutions, solution A is considered

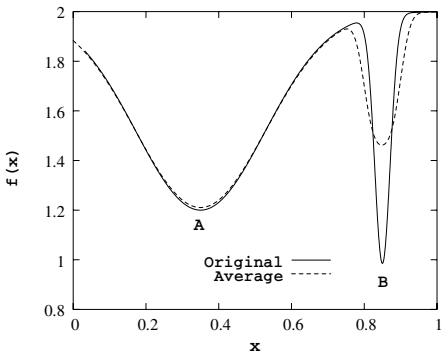


Figure 1: Illustration of global versus robust solutions in a single-objective optimization problem.

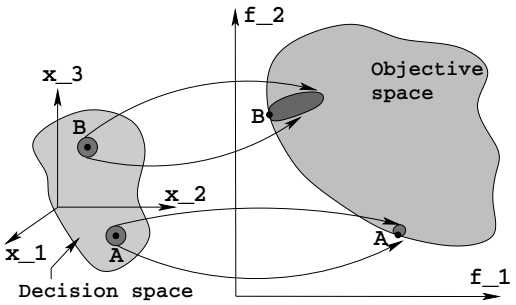


Figure 2: Point A is less sensitive to variable perturbation than point B in a multi-objective optimization scenario.

robust, as a small perturbation of the decision variables does not alter the objective function value of the solution by a significant amount. On the other hand, solution B is quite sensitive to the variable perturbation and often cannot be recommended in practice, despite having a better function value than solution A. Several EA researchers have suggested different procedures of defining and finding such robust solutions (like solution A) in a single-objective optimization context (Parmee, 1996; Tsutsui and Ghosh,

1997; Branke, 2000; Branke and Schmidt, 2000; Jin and Sendhoff, 2003). It is possible that there may also exist several other ways to define and find a robust solution. However, one of the main ideas portrayed in the literature is to use a mean effective objective function for optimization, instead of the objective function itself:

Definition 1 (Robust Solution of Type I): *For the minimization of an objective function $f(\mathbf{x})$, a solution $\mathbf{x}^*$ is called a robust solution of type I, if it is the global minimum of the mean effective function $f^{\text{eff}}(\mathbf{x})$ defined with respect to a δ-neighborhood as follows:*

$$\left.\begin{array}{ll} \textit{Minimize} & f^{\text{eff}}(\mathbf{x}) = \frac{1}{|\mathcal{B}_\delta(\mathbf{X})|}\int_{\mathbf{y}\in\mathcal{B}_\delta(\mathbf{x})} f(\mathbf{y})d\mathbf{y}, \\ \textit{subject to} & \mathbf{x}\in\mathcal{S}, \end{array}\right\} \tag{2}$$

where $\mathcal{B}_\delta(\mathbf{x})$ is the δ-neighborhood of the solution $\mathbf{x}$ and $|\mathcal{B}_\delta(\mathbf{x})|$ is the hypervolume of the neighborhood.

To use it in practice, a finite set of H solutions can be randomly (or in some structured manner, such as the Latin hypercube method, which we describe later in Section 4) chosen around a δ-neighborhood ($\mathcal{B}_\delta(\mathbf{x})$) of a solution $\mathbf{x}$ in the variable space and the mean function value (f^{eff}) is used as the fitness to an EA. This way, instead of an individual's own function value (f), an agglomerate objective value in its vicinity is used as the objective for optimization. Since this causes H times more function evaluations than the usual approach of optimizing the objective function itself, the use of a dynamically updated archive of a fixed size for choosing neighboring solutions is recommended and offers faster computation (Branke and Schmidt, 2000).

The new approach proposed here would be to calculate the normalized difference in values between the perturbed function value f^{p} (can be f^{eff}) and the original function f itself and declare a solution to be robust, if the normalized difference is smaller than a chosen threshold (η):

Definition 2 (Robust Solution of Type II): *For the minimization of an objective function $f(\mathbf{x})$, a solution $\mathbf{x}^*$ is called a robust solution of type II, if it is the global minimum solution of the following problem:*

$$\left.\begin{array}{ll} \textit{Minimize} & f(\mathbf{x}), \\ \textit{subject to} & \frac{\|f^{\text{p}}(\mathbf{x})-f(\mathbf{x})\|}{\|f(\mathbf{x})\|} \le \eta, \\ & \mathbf{x}\in\mathcal{S}. \end{array}\right\} \tag{3}$$

The perturbed function value f^{p} can be chosen as the mean effective function value (f^{eff}) or the worst function value (among H chosen solutions) in the neighborhood or any other aggregate function which would denote the amount of perturbation in the function $f(\mathbf{x})$. The operator $\|\cdot\|$ can be any suitable norm measure. The use of this definition may be more practical than the previous definition, as the user has direct control over the extent of desired robustness through the parameter η. This method also requires the computation of H neighboring solutions. In some applications, the additional constraint can be replaced by any of the following constraints:

- $\|\mathbf{f}^{\text{p}}(\mathbf{x}) - \mathbf{f}(\mathbf{x})\| \le \eta$: The decision-maker may be interested in limiting the absolute difference between the perturbed and original objective vectors.
- $\frac{1}{M}\sum_{i=1}^{M}\frac{f_i^{\text{p}}(\mathbf{x})-f_i(\mathbf{x})}{f_i(\mathbf{x})} \le \eta$: The decision-maker may be interested in limiting an average objective-wise normalized difference.

- $\frac{f_i^{\mathrm{p}}(\mathbf{x})-f_i(\mathbf{x})}{f_i(\mathbf{x})} \leq \eta_i, \quad i = 1, 2, \ldots, M$: The decision-maker may be interested in limiting the normalized difference in each objective. However, this will require the decision-maker to supply a η-vector, instead of a single η value.
- $\max_{i=1,\ldots,M} \frac{f_i^{\mathrm{p}}(\mathbf{x})-f_i(\mathbf{x})}{f_i(\mathbf{x})} \leq \eta$: This will allow the maximum normalized perturbation to be within a limit.

3 Robustness in Multi-Objective Optimization

A multi-objective optimization problem has a number of conflicting objectives:

$$\left.\begin{array}{ll} \text{Minimize} & (f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_M(\mathbf{x})), \\ \text{subject to} & \mathbf{x} \in \mathcal{S}. \end{array}\right\} \tag{4}$$

The goal in an evolutionary multi-objective optimization is to find a finite number of Pareto-optimal solutions, instead of a single optimum, to the above problem. Since Pareto-optimal solutions collectively *dominate* any other feasible solution in the search space, they are all considered to be better than any other solution (Miettinen, 1999; Deb, 2001; Coello et al., 2002). A solution is said to dominate another solution if it is not worse in any of the objectives and is strictly better in at least one of the objectives. The two concepts of robustness illustrated above for single-objective optimization can be extended for multi-objective optimization as well and are worth performing from a practical standpoint. In Figure 2, two Pareto-optimal solutions (A and B) are checked for their sensitivity in the decision variable space. Since the local perturbation of point B causes a large change in objective values, this solution may not qualify as a robust solution. To qualify as a robust solution, each Pareto-optimal solution now has to demonstrate its insensitivity towards small perturbations in its decision variable values. The main differences with a single-objective robust solution are as follows:

1. The sensitivity now has to be established with respect to all M objectives (or to the ones preferred by the decision-maker). That is, a combined effect of variations in all M objectives has to be used as a *measure* of sensitivity to variable perturbation.
2. There are many solutions to be checked for robustness, instead of one or two solutions as in the case of single-objective optimization.

Extending the ideas portrayed in two definitions of robustness to multi-objective optimization raises some interesting issues. We discuss them next one at a time.

3.1 Multi-Objective Robust Solutions of Type I

Since there exist multiple conflicting objectives in a multi-objective optimization, Definition 1 now has to be changed as follows:

Definition 3 (Multi-objective Robust Solution of Type I): *A solution $\mathbf{x}^*$ is called a multi-objective robust solution of type I, if it is the global feasible Pareto-optimal solution to the following multi-objective minimization problem (defined with respect to a δ-neighborhood ($\mathcal{B}_\delta(\mathbf{x})$ of a solution $\mathbf{x}$):*

$$\left.\begin{array}{ll} \textit{Minimize} & (f_1^{\mathrm{eff}}(\mathbf{x}), f_2^{\mathrm{eff}}(\mathbf{x}), \ldots, f_M^{\mathrm{eff}}(\mathbf{x})), \\ \textit{subject to} & \mathbf{x} \in \mathcal{S}, \end{array}\right\} \tag{5}$$

where $f_j^{\mathrm{eff}}(\mathbf{x})$ is defined as follows:

$$f_j^{\mathrm{eff}}(\mathbf{x}) = \frac{1}{|\mathcal{B}_\delta(\mathbf{x})|} \int_{\mathbf{y} \in \mathcal{B}_\delta(\mathbf{x})} f_j(\mathbf{y}) d\mathbf{y}. \tag{6}$$

Due to the variable sensitivities, a part of the original global Pareto-optimal (efficient) front may not qualify as a robust front. In some scenarios, the original global efficient front (corresponding to the problem stated in Equation 4) may be completely non-robust and an original local efficient or an original sub-optimal front may now become robust. Depending on how robust the original global efficient front is with respect to the above definition, there can be the following four main scenarios:

- **Case 1:** The complete original efficient front is robust.
- **Case 2:** A part of the original efficient front is no more robust.
- **Case 3:** The complete original global efficient front is non-robust; instead an original local efficient front is robust.
- **Case 4:** A part of the original global efficient front is robust together with a part of an original local efficient front.

We illustrate and discuss each of the above four scenarios in the following. Later, we develop one test problem for each scenario.

3.1.1 Case 1

This is the simplest case in which the original efficient front remains as an efficient front with respect to the mean effective objective functions. Figure 3 illustrates such a problem for a two-objective optimization problem. Although it is expected that the global efficient front constructed with the mean effective objectives will be somewhat worse than that constructed with the original objectives, the entire set of original Pareto-optimal solutions is robust and is the target in this type of optimization problems. The

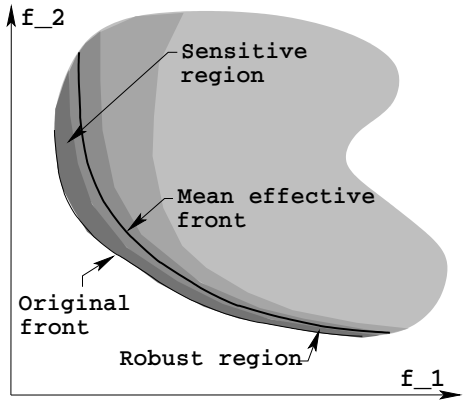


Figure 3: Case 1: Complete efficient front is robust.

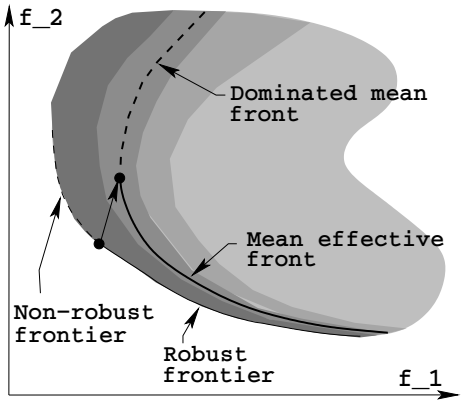


Figure 4: Case 2: A part of the efficient front is robust.

concept can also be extended for more than two objectives.

3.1.2 Case 2

Here, the entire original efficient front is not robust with respect to the above definition for robustness of type I. In most real-world scenarios (including the welded-beam design problem described in Section 7) such a problem is expected, as some portion of the original efficient front may lie in a sensitive region in the decision variable space.

In such a problem, an important task of a robust optimizer would be to identify only that part of the efficient front which is robust (that is, less sensitive to the variable perturbation). Figure 4 shows that the efficient front corresponding to the mean effective objectives does not span over the entire original efficient region in a two-objective optimization problem.

3.1.3 Case 3

Cases 3 and 4 correspond to more difficult problems in which the original problem may have more than one efficient frontiers (global and local). In a Case 3 problem, the global efficient front of the original problem is completely dominated by a local efficient front with respect to the mean effective objectives, thereby meaning that the original global Pareto-optimal solutions are not robust solutions and that they are sensitive to local perturbation. Figure 5 demonstrates such a problem with two objectives. This type of problem, if encountered, must be solved for finding the robust efficient front, instead of the sensitive global efficient front.

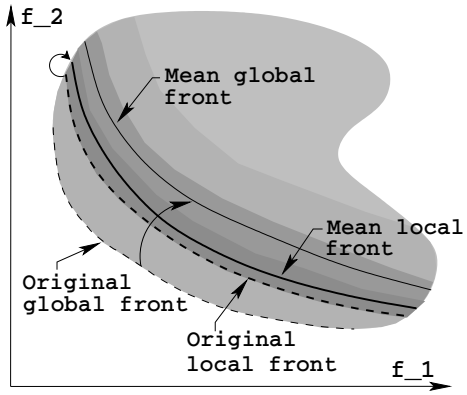


Figure 5: Case 3: The complete global efficient front is not robust.

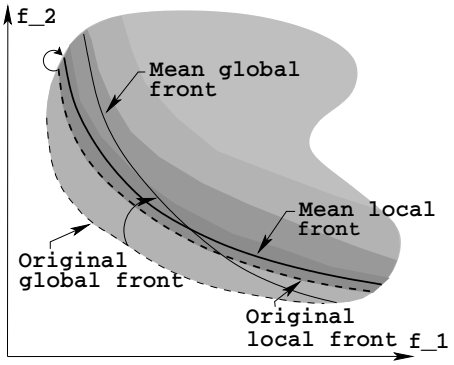


Figure 6: Case 4: A part of the global efficient front is not robust.

3.1.4 Case 4

Instead of the complete original global efficient front being sensitive to the variable perturbation, Case 4 problems cause a part of it to be adequately insensitive. In the remaining part, a new front appears to be robust. Figure 6 illustrates this problem with two objectives. A part of the robust frontier corresponds to the original global efficient frontier and the rest corresponds to the original local efficient frontier.

Certainly, other scenarios are possible, where instead of an original local efficient front becoming robust, a completely new frontier emerges to be robust. This is likely in the case of constrained optimization with Pareto-optimal solutions lying on the constraint boundary. The concepts can also be extended to problems having more than two objectives. We discuss more about constrained problems in Section 6.

3.2 Test Problems for Multi-Objective Robust Optimization

In this section, we now construct mathematical test problems for each of the above four cases for evaluating and illustrating the differences between them.

3.2.1 Test Problem 1

This problem is an illustration of Case 1 discussed above.

$$\begin{array}{rl} \text{Minimize} & f_1(\mathbf{x}) = x_1, \\ \text{Minimize} & f_2(\mathbf{x}) = h(x_1) + g(\mathbf{x})S(x_1), \\ \text{Subject to} & 0 \le x_1 \le 1, -1 \le x_i \le 1, \quad i = 2, 3, \ldots, n, \\ \text{where} & h(x_1) = 1 - x_1^2, \\ & g(\mathbf{x}) = \sum_{i=2}^{n} 10 + x_i^2 - 10\cos(4\pi x_i), \\ & S(x_1) = \frac{\alpha}{0.2+x_1} + \beta x_1^2. \end{array} \tag{7}$$

Here, we suggest $\alpha = 1$ and $\beta = 1$. The efficient front corresponds to $x_i = 0$ for $i = 2, 3, \ldots, n$ and for any value of x_1 in the prescribed domain $[0, 1]$. At these solutions, $g(\mathbf{x}) = 0$, thereby resulting in the following relationship between original objectives:

$$f_2 = 1 - {f_1}^2. \tag{8}$$

The use of a multi-modal $g()$ function causes optimization algorithms a difficulty in converging to the true efficient frontier. The mean effective objectives in a δ-neighborhood (x_i is perturbed in the neighborhood $[x_i - \delta_i, x_i + \delta_i]$) for a Pareto-optimal solution, $\mathbf{x}$, are given as follows:

$$f_1^{\text{eff}}(\mathbf{x}) = x_1, \tag{9}$$

$$\begin{aligned} f_2^{\text{eff}}(\mathbf{x}) = {} & (1 - x_1^2) - \frac{1}{3}\delta_1^2 + \left[\alpha \frac{1}{2\delta_1} \log\left(\frac{0.2 + x_1 + \delta_1}{0.2 + x_1 - \delta_1}\right)\right. \\ & \left. + \beta\left(x_1^2 + \frac{1}{3}\delta_1^2\right)\right] \sum_{i=2}^{n} \left(10 + \frac{1}{3}\delta_i^2 - \frac{10}{4\pi\delta_i}\sin 4\pi\delta_i\right). \end{aligned} \tag{10}$$

The corresponding efficient front can be obtained by substituting f_1^{eff} in place of x_1 in the latter equation. It is interesting to note that the mean effective front depends on the chosen δ-neighborhood. We shall demonstrate its effect in Section 4. In the limit, $\lim_{\delta_i \to 0} f_2^{\text{eff}}(\mathbf{x}) = 1 - (f_1^{\text{eff}}(\mathbf{x}))^2$, which is identical to the original efficient front (given in Equation 8). Further, we shall discuss in Section 4 about the existence of a limiting δ ($= \delta_i^{\text{cr}}$), above which the entire original Pareto-optimal front does not remain robust.

3.2.2 Test Problem 2

This problem is an illustration of Case 2. The mathematical formulation of this problem is identical to that in test problem 1, except that here we use $\alpha = 1$ and $\beta = 10$. The corresponding efficient frontier for the original problem and the one for the mean effective objectives can be obtained from Equation 8, 9 and 10 by substituting the above parameter values. For these parameter values, the entire original efficient frontier is not robust.

3.2.3 Test Problem 3

This problem is an instantiation of Case 3. Since, this problem requires a swapping of local and global efficient frontiers when evaluated using the mean effective objectives,

we construct a bi-modal, two-objective optimization problem:

$$\begin{array}{rl} \text{Minimize} & f_1(\mathbf{x}) = x_1, \\ \text{Minimize} & f_2(\mathbf{x}) = h(x_2)\left(g(\mathbf{x}) + S(x_1)\right), \\ \text{Subject to} & 0 \le x_1, x_2 \le 1, \\ & -1 \le x_i \le 1, \quad i = 3, 4, \ldots, n, \\ \text{where} & h(x_2) = 2 - 0.8\exp\left(-\left(\frac{x_2-0.35}{0.25}\right)^2\right) - \exp\left(-\left(\frac{x_2-0.85}{0.03}\right)^2\right), \\ & g(\mathbf{x}) = \sum_{i=3}^{n} 50x_i^2, \\ & S(x_1) = 1 - \sqrt{x_1}. \end{array} \tag{11}$$

Both local and global efficient fronts correspond to $x_i = 0$ for all $i = 3, 4, \ldots, n$, so that $g(\mathbf{x}) = 0$. Thus, at these fronts, $f_2(x_1, x_2) = h(x_2)S(x_1)$. Since, $f_1(x_1) = x_1$, the local and global efficient frontiers will correspond to the local and global minima of $h(x_2)$, respectively. A careful look at $h()$ function will reveal that there are two minima, of which the global minimum is at $x_2^* \approx 0.85$ (with $h(x_2^*) \approx 1.0$). Thus, the construction of the above problem is such that the global efficient front corresponds to $x_2^* \approx 0.85$. Similarly, the local efficient front corresponds to $x_2^* \approx 0.35$ (with $h(x_2^*) \approx 1.2$). The approximate relationship between f_1 and f_2 at these two fronts are as follows:

$$\begin{array}{rcl} f_2 & = & 1 - \sqrt{f_1} \quad \text{(global)}, \\ f_2 & = & 1.2(1 - \sqrt{f_1}) \quad \text{(local)}. \end{array}$$

The mean effective objective values of the solutions at these two fronts are as follows:

$$f_1^{\text{eff}}(\mathbf{x}) = x_1, \tag{12}$$

$$f_2^{\text{eff}}(\mathbf{x}) = H(x_2^*, \delta_2)\left[\sum_{i=3}^{n}\frac{50}{3}\delta_i^2 + \left(1 - \frac{1}{3\delta_1}\left((x_1+\delta_i)^{1.5} - (x_1-\delta_i)^{1.5}\right)\right)\right], \tag{13}$$

where $H(x_2^*, \delta_2) = \frac{1}{2\delta_2}\int_{x_2^*-\delta_2}^{x_2^*+\delta_2} h(y)dy$. An analysis reveals that for $\delta \approx 0$ (specifically, $\delta \le 0.02723$), $H(0.85, \delta_2) < H(0.35, \delta_2)$, thereby making the original global solutions with $x_2^* = 0.85$ remain as robust solutions. But, for $\delta_2 > 0.02723$, the solutions corresponding to the original local frontier (with $x_2^* = 0.35$) become the robust solutions.

3.2.4 Test Problem 4

To represent Case 4, we construct a problem which is the same as test problem 3, with a couple of modifications:

1. The function $h()$ is dependent on two variables:

$$h(x_1, x_2) = 2 - x_1 - 0.8\exp\left(-\left(\frac{x_1+x_2-0.35}{0.25}\right)^2\right) - \exp\left(-\left(\frac{x_1+x_2-0.85}{0.03}\right)^2\right).$$

2. The variable bound on x_2 is different: $-0.15 < x_2 < 1$.

The problem has its global efficient front somewhere near $x_1 + x_2 = 0.85$ and the local efficient front near $x_1 + x_2 = 0.35$. Thus, when the mean effective objectives are minimized, the efficient frontier corresponds to a mix of three sets: the local Pareto-optimal solutions satisfying $x_1 + x_2 = 0.35$ (for $f_1 < 0.5$), the global Pareto-optimal solutions satisfying $x_1 + x_2 = 0.85$ (for $f_1 > 0.63$), and an intermediate front for which $x_2 = -0.15$. We discuss more about the robust frontier of this problem in Section 4.

3.2.5 Three-Objective Test Problem 1

We also construct a couple of three-objective test problems. The first problem is given as follows:

$$\begin{array}{rl} \text{Minimize} & f_1(\mathbf{x}) = x_1, \\ \text{Minimize} & f_2(\mathbf{x}) = x_2, \\ \text{Minimize} & f_3(\mathbf{x}) = h(x_1, x_2) + g(\mathbf{x})S(x_1, x_2), \\ \text{Subject to} & 0 \le x_1, x_2 \le 1, \\ & -1 \le x_i \le 1, \quad i = 3, 4, \ldots, n, \\ \text{where} & h(x_1, x_2) = 2 - x_1^2 - x_2^2, \\ & g(\mathbf{x}) = \sum_{i=3}^{n}(10 + x_i^2 - 10\cos(4\pi x_i)), \\ & S(x_1, x_2) = \frac{\alpha}{0.2+x_1} + \beta x_1^8 + \frac{\alpha}{0.2+x_2} + \beta x_2^8. \end{array} \tag{14}$$

Here we suggest $\alpha = 0.75$ and $\beta = 10$. The efficient front corresponds to $x_i = 0$ for $i = 3, 4, \ldots, n$ and for any value of x_1 and x_2 in the prescribed domain $[0, 1]$. At these solutions, $g(\mathbf{x}) = 0$, thereby resulting in the following relationship between optimal objective values: $f_3 = 2 - f_1^2 - f_2^2$. The mean effective objective values in a δ-neighborhood for a Pareto-optimal solution, $\mathbf{x}$, are given as follows:

$$f_1^{\text{eff}}(\mathbf{x}) = x_1, \tag{15}$$

$$f_2^{\text{eff}}(\mathbf{x}) = x_2, \tag{16}$$

$$\begin{aligned} f_3^{\text{eff}}(\mathbf{x}) = {} & (2 - x_1^2 - x_2^2) - \frac{1}{3}(\delta_1^2 + \delta_2^2) \\ & + \left[\sum_{i=1}^{2}\left(\alpha\frac{1}{2\delta_i}\log\left(\frac{0.2 + x_i + \delta_i}{0.2 + x_i - \delta_i}\right) + \beta\frac{(x_i + \delta_i)^9 - (x_1 - \delta_i)^9}{18\delta_i}\right)\right] \\ & \times \sum_{i=3}^{n}\left(10 + \frac{1}{3}\delta_i^2 - \frac{10}{4\pi\delta_i}\sin 4\pi\delta_i\right). \end{aligned} \tag{17}$$

The corresponding efficient frontier can be obtained by substituting f_1^{eff} and f_2^{eff} in place of x_1 and x_2 in the latter equation.

3.2.6 Three-Objective Test Problem 2

Like the test problem 3 suggested for two objectives, we construct a bi-modal, three-objective optimization problem:

$$\begin{array}{rl} \text{Minimize} & f_1(\mathbf{x}) = x_1, \\ \text{Minimize} & f_2(\mathbf{x}) = x_2, \\ \text{Minimize} & f_3(\mathbf{x}) = h(x_3)\left(g(\mathbf{x}) + S(x_1, x_2)\right), \\ \text{Subject to} & 0 \le x_1, x_2, x_3 \le 1, \\ & -1 \le x_i \le 1, \quad i = 4, 5, \ldots, n, \\ \text{where} & h(x_3) = 2 - 0.8\exp\left(-\left(\frac{x_3 - 0.35}{0.25}\right)^2\right) - \exp\left(-\left(\frac{x_3 - 0.85}{0.03}\right)^2\right), \\ & g(\mathbf{x}) = \sum_{i=4}^{n}(10 + x_i^2 - 10\cos(4\pi x_i)), \\ & S(x_1, x_2) = 10 - \sqrt{x_1} - \sqrt{x_2}. \end{array} \tag{18}$$

Similar to the test problem 3, the relationship among optimal f_1, f_2 and f_3 at the local and global efficient fronts are well-approximated as follows:

$$\begin{aligned} f_3 &= 10 - \sqrt{f_1} - \sqrt{f_2} \quad \text{(global)}, \\ f_3 &= 1.2(10 - \sqrt{f_1} - \sqrt{f_2}) \quad \text{(local)}. \end{aligned}$$

The corresponding $x_i = 0$ for $i > 3$. The local and global efficient frontiers correspond to $x_3 = 0.35$ and 0.85, respectively. The mean effective objective values for the solutions of these two fronts are as follows:

$$f_1^{\text{eff}}(\mathbf{x}) = x_1, \tag{19}$$

$$f_2^{\text{eff}}(\mathbf{x}) = x_2, \tag{20}$$

$$f_3^{\text{eff}}(\mathbf{x}) = H(x_3^*, \delta_3) \left[\sum_{i=4}^{n} \left(10 + \frac{\delta_i^2}{3} - \frac{10}{4\pi\delta_i} \sin(4\pi\delta_i) \right) + 10 - \sum_{i=1}^{2} \left(\frac{1}{3\delta_i} \left((x_i + \delta_i)^{1.5} - (x_i - \delta_i)^{1.5} \right) \right) \right]. \tag{21}$$

The expression for $H(x_3^*, \delta_3)$ is analogous to that presented in the case of the two-objective test problem 3. Recall that for $\delta_3 > 0.02723$ the original local Pareto-optimal solutions are robust.

4 Simulation Results for Robust Solutions of Type-I

Here, we use the NSGA-II (Deb et al., 2002) procedure to find the robust Pareto-optimal solutions, although any other EMO algorithm can also be used. Various parameters which determine the extent and nature of shift of the mean effective front from the original front are as follows:

- The extent of neighborhood (δ) considered for each variable.
- Number of neighboring points (H) used to compute the mean effective objectives.
- Number of variables (n) in the problem. Although this is not a parameter a user has a control over, we demonstrate the effect of this parameter to particularly show its relation with the extent of robustness in the Pareto-optimal solutions.

The effects of all three parameters are analyzed in detail for the first two test problems. However, before we discuss the results, there is an important matter which we discuss first.

There can be a number of ways of generating H neighboring points in the vicinity of a solution to compute the mean effective objective values (Branke, 2000). The simplest strategy can be to randomly create H points in the neighborhood of every solution (using the Monte Carlo approach). However, this introduces additional randomness in evaluating the same solution more than once. It was suggested that a random pattern of points around a solution is created in the beginning of a simulation and the same pattern be used for every solution (Branke, 1998). Many statistical space-filling sampling methods exist for creating a diverse and well-distributed set of points in a specific region (Montgomery, 2001). Here, to create a pattern systematically, we use the commonly-used Latin hypercube strategy in which we divide the perturbation domain of each variable (around $[-\delta_i, \delta_i]$) into exactly H equal grids, thereby dividing the δ-neighborhood into H^n small hyperboxes. Thereafter, we pick exactly H hyperboxes randomly from H^n hyperboxes so that in each dimension all H distinct grids are represented. Figure 7 shows two such patterns for a two-variable problem. Once the hyperboxes are identified, a random point within each hyperbox is chosen and is used for the computation of the mean effective objective values. We use a random pattern in the beginning of each generation and use it for evaluating all solutions during the generation. For the next generation, we create another random pattern.

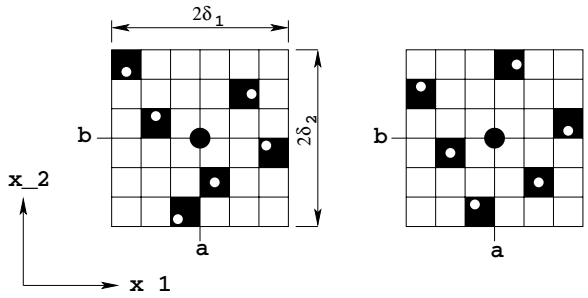


Figure 7: Two examples of creating $H = 6$ neighboring points around a solution $\mathbf{x} = (\mathtt{a}, \mathtt{b})$.

In all simulations, we have used the simulated binary crossover (SBX) and the polynomial mutation operator with distribution indices of 10 and 50, respectively (Deb, 2001). A population size of 100 is used in all cases. Although the purpose of this study is not to suggest an algorithm or compare different algorithms for multi-objective robust optimization, we performed multiple runs in each case with different initial populations. Here, we simply present the result of a typical run. However, despite an early convergence to the desired front by the chosen algorithm, we extend a specific run for a long enough (10,000) number of generations to have confidence in the location of the obtained robust front.

4.1 Test Problems 1 and 2

We first show the effect of various parameters on the first two test problems.

4.1.1 Effect of Neighborhood Size, δ

First, we show the effect of δ_i on the test problem 1. To not have a significant effect due to finite neighboring points and variation in problem size, we use $H = 50$ and $n = 5$. Figure 8 shows the theoretical mean effective front obtained using Equations 9 and 10 for four different values of neighborhood size defined by a parameter δ as follows: $\delta_1 = \delta$ and $\delta_i = 2\delta$ for all $i > 1$ to have an identical neighborhood size in all variables. It is clear from the figure that as δ increases, the shift in the mean effective efficient front moves away from the original efficient front. Although for this test problem, all solutions corresponding to the mean effective Pareto-optimal front are identical to those lying on the original efficient front for the chosen neighborhood size, the change in shape of the front is interesting. For the four δ used here, the mean effective front is non-convex, whereas the original front is convex. Thus, algorithms having difficulties in solving non-convex problems may face trouble in finding the robust frontier, although they may be able to find the original convex (yet non-robust) efficient frontier.

To analyze the extent of perturbation for which the above problem remains a Case 1 problem, we observe that the quantity summed for the second to n-th variable in Equation 10 can be approximately written as $80\pi^2\delta_i^2/3$. Let us also define a parameter A as $A = \sum_{i=2}^{n} 80\pi^2\delta_i^2/3$. Differentiating f_2^{eff} with respect to f_1^{eff} (or $\mathbf{x}$) and equating the term to zero, we can compute the minimum solution corresponding to f_2^{eff}. A solution to the following equation will result in the corresponding f_1^{eff} objective value ($\bar{f}_1^{\text{eff}}$):

$$\bar{f}_1^{\text{eff}} \left(0.2 + \bar{f}_1^{\text{eff}}\right)^2 = \frac{0.5\alpha}{\beta - 1/A}. \tag{22}$$

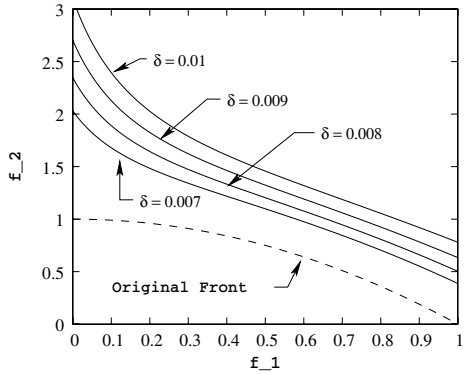


Figure 8: Theoretical mean effective fronts showing the effect of δ on test problem 1.

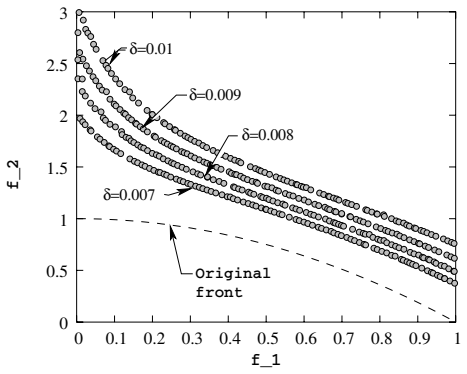


Figure 9: Robust solutions obtained using NSGA-II show the effect of δ on test problem 1.

In order to have the entire original efficient front to remain as robust, $\bar{f}_1^{\text{eff}} \geq 1$, yielding

$$A \leq 1/(\beta - 0.3472\alpha) \tag{23}$$

from the above equation. For the test problem 1 with $\alpha = \beta = 1$, this means $A \leq 1.532$. Substituting the expression of A in terms of $n = 5$ and δ-neighborhood ($\delta_1 = \delta$ and $\delta_i = 2\delta$ for $i > 1$), the maximum δ which would cause all original Pareto-optimal solutions to remain as robust is $\delta \leq \delta^{\text{cr}} = 0.019$. Since in Figure 8, δ values smaller than this critical value are used, all original Pareto-optimal solutions are robust.

Figure 9 verifies that the obtained NSGA-II (robust) solutions for the same four δ values span the entire range of f_1. A close investigation will reveal that the obtained front is exactly the same as that obtained using the exact mathematical analysis (Figure 8).

Figures 10 and 11 show theoretical and NSGA-II results on test problem 2. In this problem, not only the shape of the mean effective front is different from the original

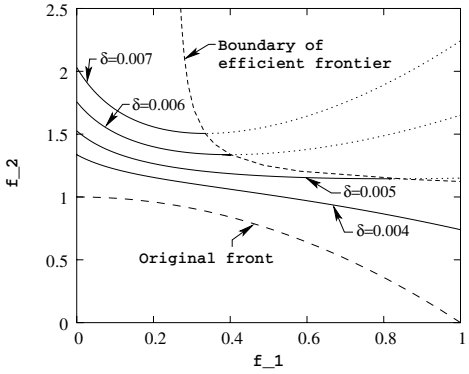


Figure 10: Theoretical mean effective fronts showing the effect of δ on test problem 2.

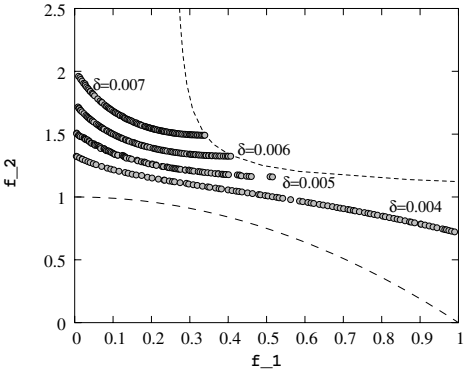


Figure 11: Robust solutions obtained using NSGA-II show the effect of δ on test problem 2.

one, some original Pareto-optimal solutions are no longer optimal. Using Equation 22 and substituting $\alpha = 1$ and $\beta = 10$, we obtain $A \leq 0.1036$ for the test problem 2. Hence, the maximum value of δ which will cause all original Pareto-optimal solutions to remain as robust solutions is $\delta = \delta^{\text{cr}} = 0.00496$. Figure 10 shows that for $\delta = 0.004$ all original Pareto-optimal solutions are robust, while for $\delta = 0.005$ or more, not all original Pareto-optimal solutions are robust. For example, solutions with x_1 greater than about 0.403 are not robust solutions in the case of $\delta = 0.006$. This simply means that these Pareto-optimal solutions are very sensitive to variable perturbation and are not robust. Figure 11 shows that NSGA-II finds only the non-dominated (or robust) portion of this efficient front. By finding the root of Equation 22 for different values of δ, we find the boundary of different robust fronts and show it on both the figures with a dashed line. It is interesting to note how sensitive the robust solutions of test problem 2 are to the choice of neighborhood size. With a larger neighborhood size, fewer solutions become robust. Once again, the prediction by theory is verified by simulation results of NSGA-II in this specific problem.

4.1.2 Effect of Neighboring Points, H

It is intuitive that if more neighboring points are chosen for computing the mean effective objectives, the objective values will be closer to the theoretical average values; however, the computation time will be more. Figure 12 shows the effect of using different values of H on test problem 1. Here, we use $\delta = 0.01$ and $n = 5$. The theoretical mean effective front (ideally for $H = \infty$) is also shown with a solid line in the figure. It is clear that as H is increased, the shift of the mean effective front from the original front is more and asymptotically approaches the theoretical front. Figure 13 shows the effect of H on test problem 2 (with $n = 5$ and $\delta = 0.007$). The front obtained using a small H overestimates the true robust front, but with a much smaller computational time. It is also interesting to note from these two figures that the frontier computed using only one neighboring solution makes a good approximation of the true robust frontier in these problems.

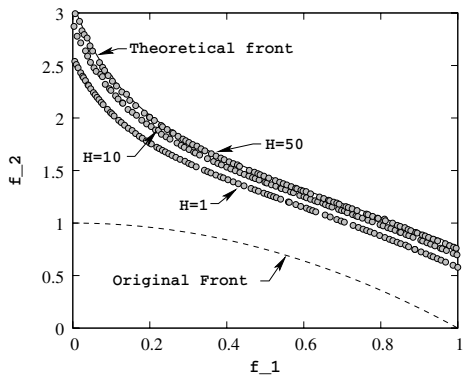


Figure 12: Effect of H (theoretical and NSGA-II) on test problem 1 ($\delta = 0.01$ and $n = 5$).

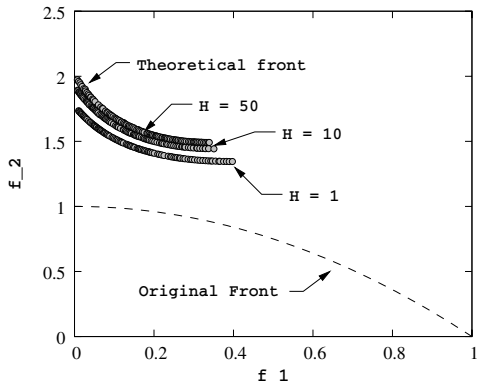


Figure 13: Effect of H (theoretical and NSGA-II) on test problem 2 ($\delta = 0.007$ and $n = 5$).

4.1.3 Effect of Number of Variables, n

Next, we investigate the effect of n on the robustness of the test problems 1 and 2. We use $H = 50$ in each problem and fix $\delta = 0.01$ and 0.007 for problems 1 and 2, respectively. Figures 14 and 15 show the change in the mean effective front with n for the two problems. Recall that for the test problem 1, the limiting A was 1.532. Substituting $A = (n-1)320\pi^2\delta^2/3$ and using $\delta = 0.01$, we obtain $n \leq 15.55$ or $n \leq 15$. This suggests that up to a 15-variable version of test problem 1 all original Pareto-optimal solutions remain robust. For 16 variables or more, not all such solutions will remain as robust with $\delta = 0.01$. Since we have performed simulations for $n = 3$, 4, and 5, in all cases the complete original Pareto-optimal solutions are found to be robust by NSGA-II.

Similarly, in the case of test problem 2 with $\delta = 0.007$, the condition $A \leq 0.1036$ yields $n \leq 3.008$ or $n \leq 3$. It is clear from Figure 15 that for $n = 3$, the obtained robust front corresponds to the original Pareto-optimal solutions. But when $n = 4$ or more is used, a part of the original Pareto-optimal solutions (up to $x_1 \leq 0.397$ for $n = 4$) are robust. The boundary of the robust front for different problem sizes is shown in Figure 15 as well. It is intuitive that the effect of robustness can be more dramatic in a higher dimensional problem, as depicted by both simulation results.

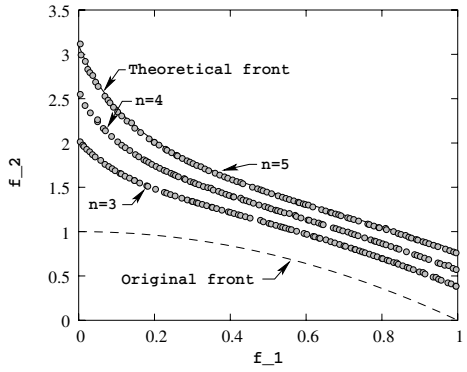


Figure 14: Effect of n (theoretical and NSGA-II) on test problem 1 ($\delta = 0.01$ and $H = 50$).

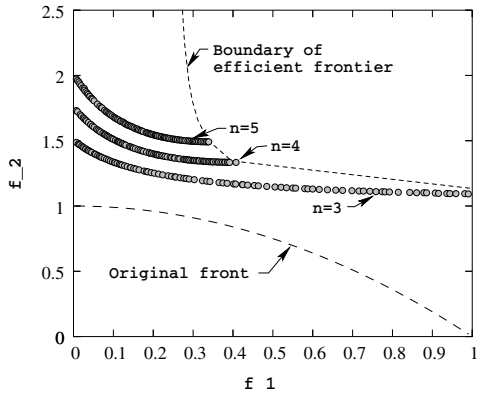


Figure 15: Effect of n (theoretical and NSGA-II) on test problem 2 ($\delta = 0.007$ and $H = 50$).

4.2 Test Problems 3 and 4

For problems 3 and 4, we show the effect of local and global fronts of the original problem in deciding on the true robust front. Here, we choose $\delta_1 = \delta_2 = \delta$ and $\delta_i = 2\delta$ for all $i > 2$, to have an identical precision in all variables. For both problems, we use $\delta = 0.03$, $H = 100$, and $n = 5$. Figure 16 shows the theoretical results obtained using Equations 12 to 13. The original local and global efficient fronts are shown in dashed lines. The mean effective local and global fronts are also shown in the figure with solid lines. It is clear that the mean effective local front is the robust frontier of this problem, meaning that the original local Pareto-optimal solutions are robust solutions and original global Pareto-optimal solutions are too sensitive to the variable perturbation to qualify as robust solutions. For a larger choice of δ_2 (or δ), the gap between these two fronts would be larger.

Figure 17 shows NSGA-II solutions applied to mean objective values obtained by averaging H function values in the δ-neighborhood of a solution. The NSGA-II front corresponds to the theoretical local mean effective front (which is also the robust frontier), as can also be seen by comparing both figures.

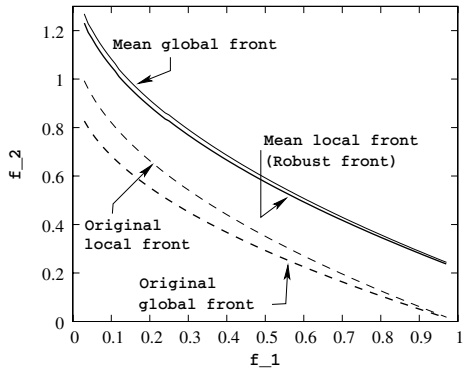


Figure 16: Theoretical robust front for test problem 3 ($\delta = 0.03$).

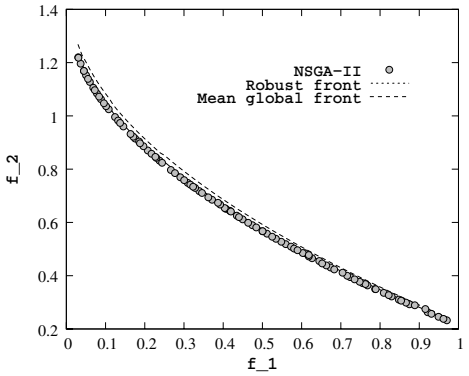


Figure 17: NSGA-II robust front for test problem 3 ($\delta = 0.03$).

The difference in two frontiers (robust and mean global frontiers) is not significant in the objective space plot shown in Figure 17. To show the real difference between the original efficient front and the robust front, we plot all 100 obtained NSGA-II solutions in the decision space for the two cases in Figures 18 and 19, respectively. Each solution consisting of five variables x_1 to x_5 are joined by a solid line. It is clear that for all solutions of the original front, x_2 is close to 0.85. Variables x_3 to x_5 are all settled to a value zero and the variation of solutions on the front appears due to the variation in x_1 alone. Figure 19 shows the robust solutions. Here, all solutions take a value close to $x_2 = 0.35$.

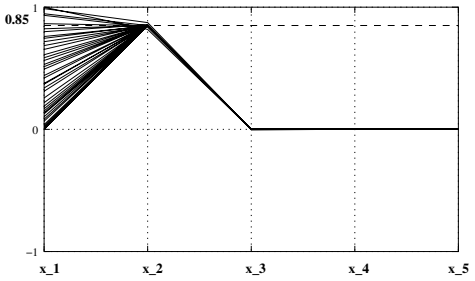


Figure 18: NSGA-II solutions (decision variable space) of the original test problem 3.

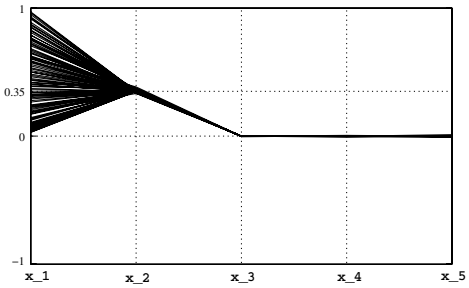


Figure 19: NSGA-II robust solutions (decision variable space) for test problem 3.

Next, we consider test problem 4. Theoretical fronts for the original problem are shown in Figure 20 in dashed lines and corresponding mean effective fronts are shown in solid lines. In both cases, the local efficient frontier takes a different functional form for $f_1 \leq 0.5$ and for $f_1 > 0.5$, as discussed earlier. It is clear from the figure that a part of the robust frontier is constituted with some local Pareto-optimal solutions

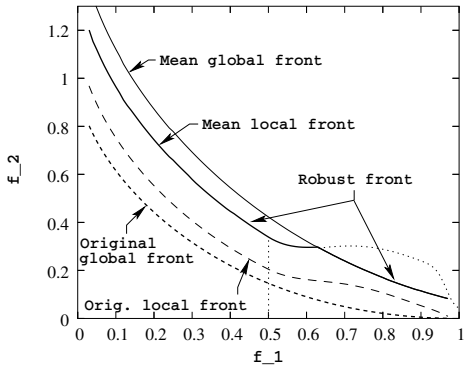


Figure 20: Theoretical robust front for test problem 4.

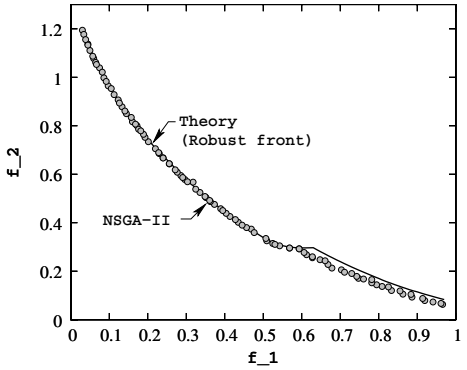


Figure 21: NSGA-II robust front for test problem 4.

and another part with some global Pareto-optimal solutions. An intermediate portion ($f_1 \in [0.5, 0.63]$) corresponds to $x_2 = -0.15$. Figure 21 shows the robust solutions obtained using NSGA-II. The deviation in the global part of the robust frontier from theory is due to the choice of a finite H (50 here). The original function landscape at the global frontier is quite sensitive to parameter changes, and it becomes difficult for an optimization algorithm to converge to the exact global frontier. When we rerun the problem with $H = 500$, the obtained solutions are closer to the theoretical frontier.

Figures 22 and 23 show dramatically the relationship between x_1 and x_2 in the solutions obtained for the original problem and that obtained for the mean effective objectives, respectively. It is clear that for solutions $f_1 \leq 0.5$ the relationship more or less follows $x_1 + x_2 = 0.35$ and for $f_1 \geq 0.63$ the relationship is $x_1 + x_2 = 0.85$. The latter condition corresponds to the original global efficient front, as shown in Figure 22. For $0.5 < f_1 < 0.63$, x_2 gets fixed to its lower bound (-0.15).

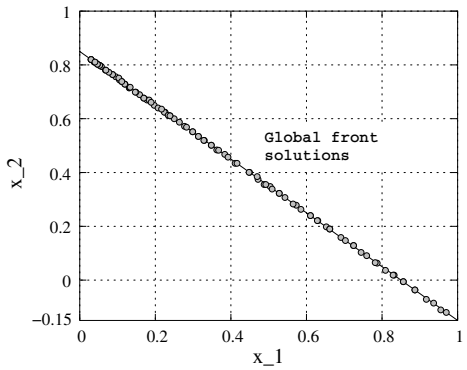


Figure 22: NSGA-II solutions of the original test problem 4.

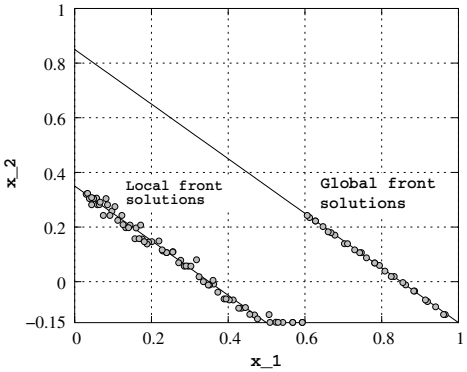


Figure 23: NSGA-II robust solutions for test problem 4.

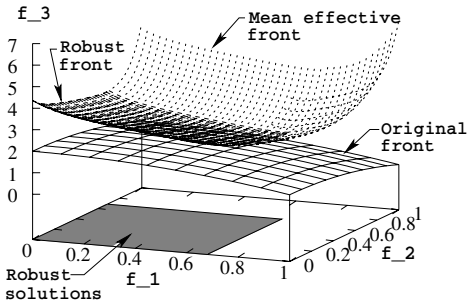


Figure 24: Theoretical Robust Front for three-objective test problem 1

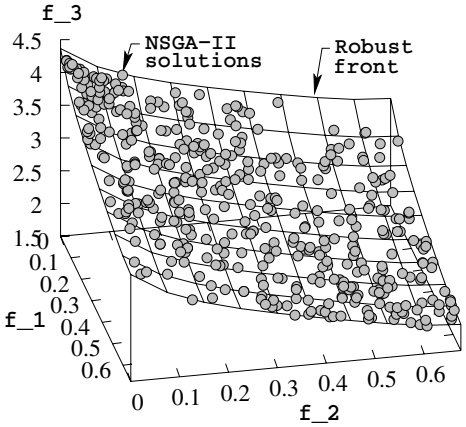


Figure 25: NSGA-II robust solutions for three-objective test problem 1.

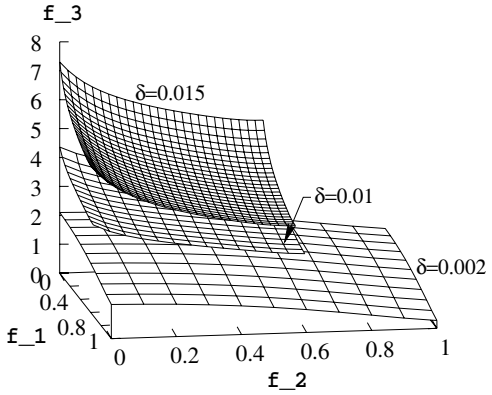


Figure 26: Theoretical effective fronts for different values of δ.

4.3 Three-Objective Test Problem 1

Figure 24 shows the theoretical efficient fronts obtained for three-objective test problem 1. Here, we use $\delta_1 = \delta_2 = \delta$ $(= 0.01)$ and $\delta_i = 2\delta$ for all $i > 2$. Also, we choose $H = 50$ and $n = 5$. The theoretical efficient front is obtained using Equations 15 to 17. Solutions corresponding to the non-dominated part of this theoretical effective frontier are robust solutions of type I. The robust frontier and its projection on f_1-f_2 plane are also shown as a shaded region.

Figure 25 shows the robust solutions obtained using NSGA-II. It is amply clear from the figure that the obtained robust solutions lie on the theoretical efficient frontier shown in Figure 24.

Figure 26 shows the effect of δ on the three-objective test problem 1. As expected, with an increase in δ, the robust frontier moves further away from the original efficient frontier. It is also evident that in this problem the non-dominated region of the robust frontier shrinks with an increase in δ.

4.4 Three-Objective Test Problem 2

The three-objective test problem 2 has a local and a global efficient frontier, similar to that observed in the two-objective test problem 3. Here, we use $\delta = 0.03$, $n = 5$, and $H = 50$. Figure 27 shows the theoretical efficient fronts (local and global) for this problem. The mean effective local and global fronts are obtained using Equations 19 to 21. It is clear from Figure 27 that the global Pareto-optimal solutions are more sensitive to variable perturbation than those corresponding to the local efficient frontier. The difference between the original global efficient front and the mean global efficient front is much more than that for the local efficient front. Thus, it is expected that the robust frontier of type I will correspond to the local Pareto-optimal solutions in this problem.

Figure 28 shows the simulation results obtained using NSGA-II. The mean effective local front is also shown. It is apparent that the obtained NSGA-II solutions lie on the theoretical mean effective local front.

To show the difference between the original global efficient front and the robust front of type I, we show obtained solutions for both cases in the variable space. In both cases, a variation in the Pareto-optimal solutions occurs due to a variation in x_1 and x_2 values. Figure 29 represents the solutions obtained for the original optimization problem. All solutions have x_3 values almost equal to 0.85, while Figure 30 shows the robust solutions of type I, where all solutions have x_3 values close to 0.35. Thus, the consideration of robustness of type I in this problem causes a completely different (less locally sensitive) set of solutions to emerge, compared to the original global Pareto-optimal solutions.

4.5 Critical Comments on Robustness of Type I

The above discussion and simulation results amply demonstrate that by optimizing the mean effective objectives (instead of the original objective functions) computed by averaging a few neighboring solutions, the robust frontier of type I can be found by using an EMO procedure. In a problem, the computation of the robust front instead of the original efficient front is more useful and provides a user with the information about robust solutions directly. It has also been found that the neighborhood size and the number of neighboring points used to compute the mean objective values are important parameters in obtaining the true robust frontier.

However, the definition of type I robustness is somewhat less practical and yields a robust frontier which gets fixed for a particular choice of δ-neighborhood. For a given problem, the above definition constitutes a particular front as a robust front, mainly from the consideration of mean objective values. However, a user may like a preferred limiting change in function values for defining robustness and would be interested in knowing the corresponding robust frontier. For this purpose, we have defined robust solutions of type II earlier and will discuss it in the next section.

5 Multi-objective Robust Solutions of Type II

The robust solution of type II for multi-objective optimization can be defined by following Definition 2:

Definition 4 (Multi-objective Robust Solution of Type II): *A solution* $\mathbf{x}^*$ *is called a multi-objective robust solution of type II, if it is the global feasible Pareto-optimal solution to*

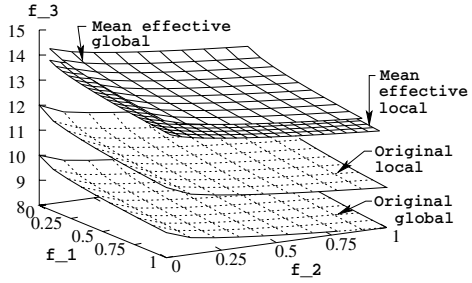


Figure 27: Theoretical robust front of type I for the three-objective test problem 2.

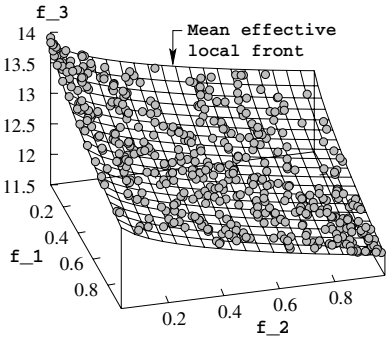


Figure 28: NSGA-II robust solutions of type I for the three-objective test problem 2.

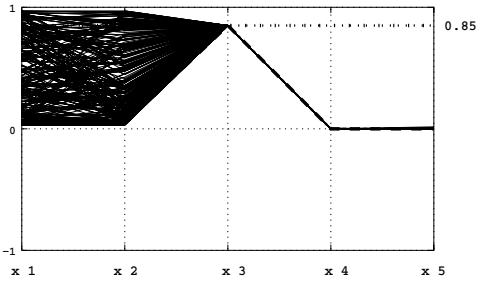


Figure 29: Pareto-optimal solutions of the original three-objective test problem 2.

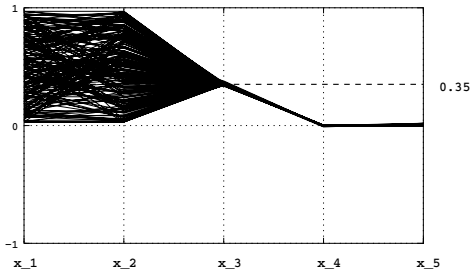


Figure 30: Robust solutions of type I for the three-objective test problem 2.

the following multi-objective minimization problem:

$$\left.\begin{array}{ll} \textit{Minimize} & \mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_M(\mathbf{x})), \\ \textit{subject to} & \frac{\|\mathbf{f}^{\mathrm{p}}(\mathbf{x}) - \mathbf{f}(\mathbf{x})\|}{\|\mathbf{f}(\mathbf{x})\|} \leq \eta, \\ & \mathbf{x} \in \mathcal{S}. \end{array}\right\} \tag{24}$$

The limiting parameter η is considered constant in a simulation run and is a user-defined parameter. We simply employ NSGA-II to solve the corresponding constrained optimization problem by using the constrained-domination principle, described elsewhere (Deb, 2001; Deb et al., 2002).

5.1 Test Problem 1

Figure 31 shows the NSGA-II solutions obtained for different pre-defined η values on test problem 1. We use $n = 5$, $H = 100$, and $\delta = 0.007$. Here, the theoretical mean effective objective functions (Equations 9 and 10) are optimized with the additional η constraint by using NSGA-II. The figure demonstrates that the sensitive region of the original efficient front is once again vulnerable to the chosen value of η. For a more tight (smaller) limiting η, the corresponding front is further away from the original front. As η is increased, the robust frontier gets closer to the original front in this sensitive region. However, on the less sensitive portion of the original frontier, the solutions are

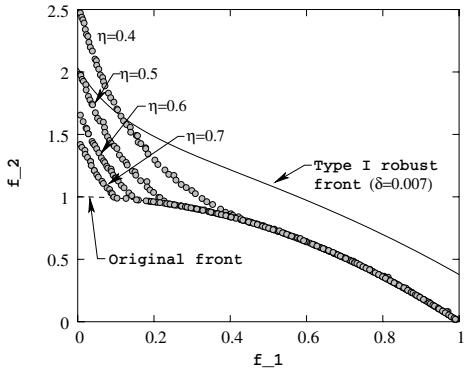


Figure 31: Robust fronts for different values of η obtained by minimizing exact f^{eff} for problem 1.

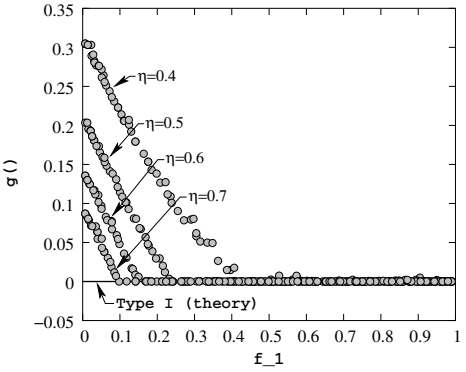


Figure 32: Function $g()$ of the robust solutions shown in Figure 16 for problem 1.

independent of η.

For a comparison, the robust front obtained with type I robustness is also shown for identical δ and H parameter values in Figures 31 and 32. To show how the decision variables are distributed across the robust frontier, we plot $g()$ function for different values of η in Figure 32. Recall that in the case of type I robustness, the robust solutions correspond to $x_i = 0$ for $i > 1$ (thereby making the $g(\mathbf{x}^*) = 0$). However, with type II robustness, different solutions appear in the sensitive portion of the robust frontier and $g(\mathbf{x}^*)$ need not be zero. To demonstrate this aspect, we plot $g(\mathbf{x}^*)$ values for two cases: Type I robust frontier (theoretical) and type II robust frontier with various η. Although the solutions shown in Figure 32 with $g(\mathbf{x}^*) \neq 0$ were not the Pareto-optimal solutions of the original problem, the definition of robustness of type II causes them to be robust with respect to a particular η.

Finally, Figure 33 shows the corresponding NSGA-II results obtained by minimizing the mean objective functions computed using H neighboring points. The obtained solutions are close to those obtained using the theoretical mean objective functions (shown in Figure 31).

5.2 Test Problem 2

Figure 34 shows the NSGA-II solutions obtained for $n = 5$, $H = 100$ neighboring points, and $\delta = 0.006$. We also use two values of η: $\eta = 0.4$ and $\eta = 0.6$. Constraints are handled using the constraint-domination principle (Deb et al., 2002). As discussed earlier, the complete efficient front was not robust of type I in this problem. For both η values, the robust frontiers of type II also do not cover the entire range of the original efficient front. However, as η is increased the robust frontier comes closer to the original front.

Figure 35 compares the $g(\mathbf{x}^*)$ values for all robust solutions of type I (theoretical) and type II ($\eta = 0.4$ and $\eta = 0.6$). The theoretical type I robust solutions span for $f_1 \leq 0.403$ and the corresponding $g()$ value for all solutions is zero. However, for the robust solutions of type II, we observe that the $g()$ values are nonzero in the most sensitive region. The NSGA-II procedure finds solutions which were non-optimal before but are robust with respect to the chosen η parameter. But, in the relatively insensitive region, the original Pareto-optimal solutions are still robust.

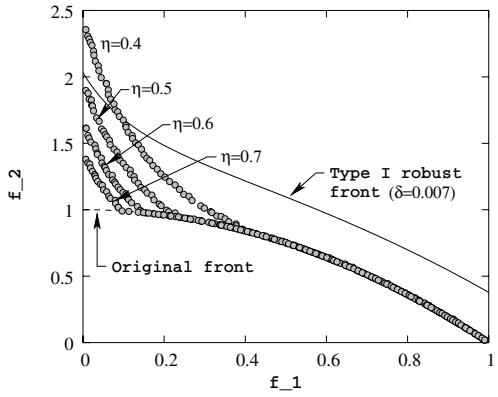


Figure 33: Robust fronts for different values of η obtained by minimizing an average of H neighboring points for test problem 1.

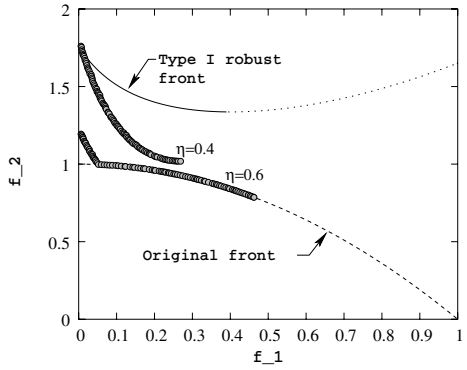


Figure 34: Robust fronts for different values of η for problem 2.

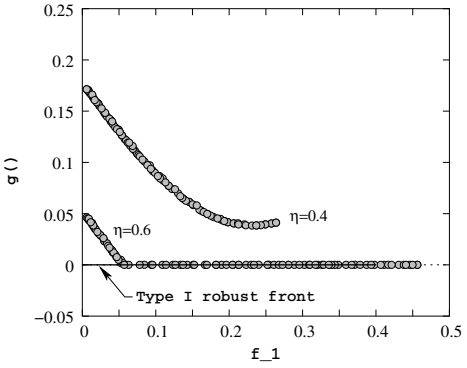


Figure 35: Function $g()$ of the robust solutions shown in Figure 16 for problem 2.

5.3 Test Problems 3 and 4

Figure 36 shows the type-II robust frontiers obtained for the test problem 3 with $H = 50$, $n = 5$ and $\delta = 0.03$. For a large value of $\eta \geq 0.7$, the type-II robust solutions are identical to the original global Pareto-optimal solutions. However, for $\eta = 0.3$ or 0.2, the original global Pareto-optimal solutions are more sensitive than allowed and a completely different set of robust solutions emerge. Figure 37 shows the $g()$ function value corresponding to each obtained robust solution for different η values. It is clear from the figure that for $\eta = 0.7$, robust solutions are identical to that of the global Pareto-optimal solutions. On the other hand, for $\eta = 0.3$ or 0.2, they are different. With a decrease in the limiting η value, the deviation of robust solutions from the original global Pareto-optimal solutions is more. We also plot the type I robust frontier (with an identical $\delta = 0.03$) of this problem in a dashed line to compare the effect of two types of robustness considered in this paper.

Figure 38 shows the type-II robust fronts obtained for the test problem 4. Here also, we observe that as the value of η decreases, the type-II robust front moves away from

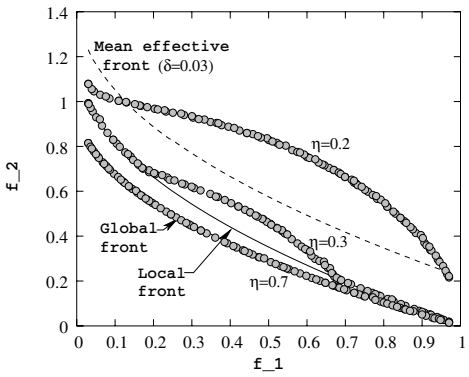


Figure 36: Robust fronts for different values of η for problem 3.

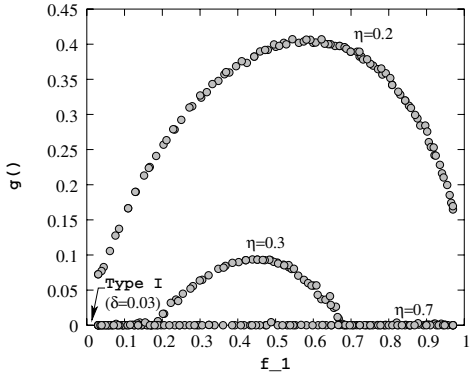


Figure 37: Corresponding $g()$ values show different Pareto-optimal sets obtained by different η. For type I ($\delta = 0.03$) robust solutions, $g() = 0$.

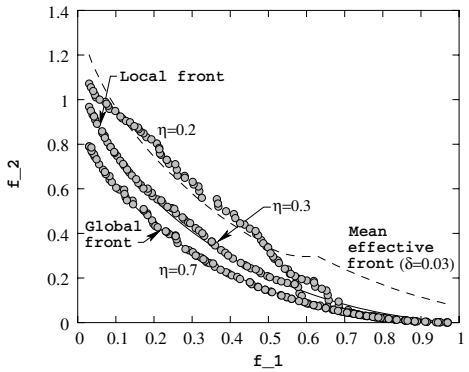


Figure 38: Robust fronts for different values of η for problem 4.

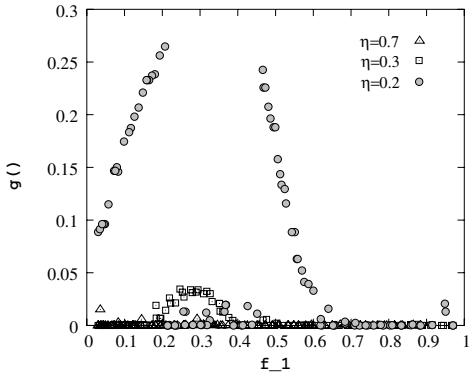


Figure 39: Corresponding $g()$ values show different Pareto-optimal sets obtained by different η.

the original global efficient front for $f_1 \leq 0.7$. In the remaining portion, the original global efficient front is robust. It is also interesting to note from Figure 39 that for smaller values of η, the robust solutions are different and the nature of variation is different from that observed in the test problem 3. All original global Pareto-optimal solutions are too sensitive with respect to a small allowable normalized perturbation (η) to qualify as robust solutions of type II. However, all original global Pareto-optimal solutions having f_1 larger than about 0.7 are still robust of type II.

5.4 Three-Objective Test Problem 1

We now show simulation results on the three-objective test problem 1. Figures 40 and 41 show the robust solutions for $\eta = 2$ and 0.4, respectively. Here, we have used $H = 50$, $n = 5$ and $\delta = 0.01$. For $\eta = 2$, the limiting difference between the mean effective and the original objective values is large enough to have the original efficient front to remain as the robust front of type II, as apparent from Figure 40. However, for

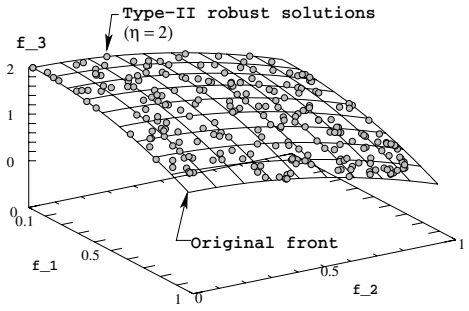


Figure 40: Type II robust solutions for $\eta = 2$ for the three-objective test problem 1. Original Pareto-optimal solutions are all robust.

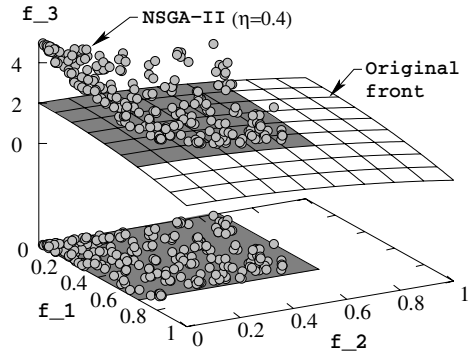


Figure 41: Type II robust solutions for $\eta = 0.4$ for the three-objective test problem 1. A part of the original Pareto-optimal solutions are robust.

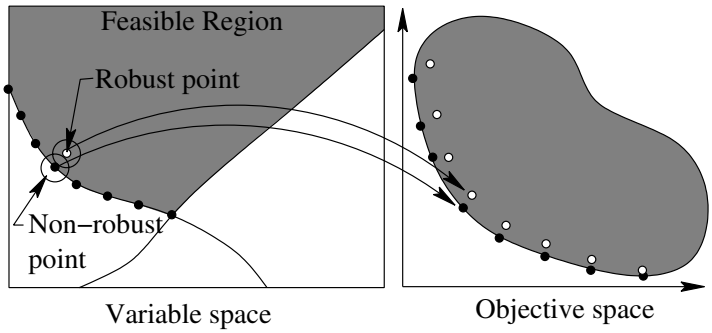


Figure 42: Illustration of a constrained robust solution.

$\eta = 0.4$ (shown in Figure 41), the limiting difference is small, and the original Pareto-optimal solutions which make a large difference between the mean effective objective values and the original efficient objective values are no longer robust of type II. Thus, only a part of the original efficient frontier is the robust front of type II. The extent of this robust front depends on the chosen η. For a comparison, we also mark (as a shaded region) the Pareto-optimal solutions obtained in the case of type I robustness with $\delta = 0.01$.

6 Constrained Robust Optimization

Figure 42 explains the necessity for finding the robust constrained solutions in a multi-objective optimization problem. On most interesting problems, a Pareto-optimal solution is likely to lie on the constraint boundary, as shown by the filled circles on the figure. Since the solutions lie on the constraint boundary, they are also precarious for use in practice, particularly if the solutions are expected to be uncertain. In such cases, feasible solutions which are somewhat away from the constraint boundary turn out to be robust (like the solutions marked with open circles). The extent of movement of solutions from the constraint boundary will depend on the neighborhood size, chosen

for performing the robust optimization. Similar to the definitions of the unconstrained robust solutions, we define a constrained robust solution in the following two definitions.

Definition 5 Robust Feasible Optimal Solution of Type I *A solution $\mathbf{x}^*$ is called a robust feasible optimal solution of type I, if it is a feasible global optimal solution and all solutions in its δ-neighborhood are also feasible:*

$$\left.\begin{array}{ll} \textit{Minimize} & (f_1^{\text{eff}}(\mathbf{x}), f_2^{\text{eff}}(\mathbf{x}), \ldots, f_M^{\text{eff}}(\mathbf{x})), \\ \textit{subject to} & \mathbf{y} \in \mathcal{S}, \quad \textit{for all } \mathbf{y} \in \mathcal{B}_\delta(\mathbf{x}), \end{array}\right\} \tag{25}$$

where $\mathcal{B}_\delta$ is the δ-neighborhood of the solution.

The inclusion of above constraint ensures that the solutions lying on an active constraint boundary will not be robust, as a perturbed solution is likely to be infeasible. In our implementation, instead of checking all neighboring solutions to be feasible, we shall check the feasibility of each of the H solutions used to compute effective function values. If any of them is infeasible, the solution is declared infeasible. To implement this idea, we compute the robust constraint violation (RCV) of each solution ($\mathbf{x}$), as follows:

$$\text{RCV}(\mathbf{x}) = \sum_{\mathbf{y} \in \mathcal{B}_\delta(\mathbf{x})} \text{CV}(\mathbf{y}), \tag{26}$$

where, the constraint violation of a solution $\mathbf{y}$ is defined as follows: $\text{CV}(\mathbf{y}) = \sum_j \langle g_j(\mathbf{y}) \rangle$, where the bracket operator $\langle \gamma \rangle$ is defined as follows: $\langle \gamma \rangle = \gamma$, if $\gamma < 0$; zero, otherwise. Thus, if the robust constraint violation is negative, some neighboring solutions used for computing the mean effective objective computation is infeasible. The RCV value computed this way is used along with NSGA-II's constrained tournament selection operator (Deb et al., 2002).

Similarly, we modify the second robustness definition as follows:

Definition 6 Robust Feasible Optimal Solution of Type II: *For the minimization of a multi-objective problem, a solution $\mathbf{x}^*$ is called a robust feasible optimal solution of type II, if it is the Pareto-optimal solution to the following problem:*

$$\left.\begin{array}{ll} \textit{Minimize} & (f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_M(\mathbf{x})), \\ \textit{subject to} & \frac{\|\mathbf{f}^{\text{p}}(\mathbf{x}) - \mathbf{f}(\mathbf{x})\|}{\|\mathbf{f}(\mathbf{x})\|} \leq \eta, \\ & \mathbf{y} \in \mathcal{S}, \quad \textit{for all } \mathbf{y} \in \mathcal{B}_\delta(\mathbf{x}). \end{array}\right\} \tag{27}$$

The inclusion of the additional constraint (last line) ensures that all solutions in the vicinity of a robust feasible optimal solution are feasible.

6.1 Simulation Results

To investigate the effect of constraints on robust solutions, we propose the following test problem:

$$\begin{array}{rl} \text{Minimize} & f_1(\mathbf{x}) = x_1, \\ \text{Minimize} & f_2(\mathbf{x}) = h(x_1) + G(\mathbf{x}) S(x_1), \\ \text{Subject to} & g(\mathbf{x}) \equiv 0.2x_1 + x_2 - 0.1 \geq 0, \\ & 0 \leq x_1 \leq 1, \\ & -1 \leq x_i \leq 1, \quad i = 2, 3, \ldots, n, \\ \text{where} & h(x_1) = 1 - x_1^2, \\ & G(\mathbf{x}) = \sum_{i=2}^{n} 50 x_i^2, \\ & S(x_1) = \frac{\alpha}{0.2 + x_1} + \beta x_1^2. \end{array} \tag{28}$$

If the constraint was absent, the corresponding Pareto-optimal front is as follows: $f_2 = 1 - f_1^2$ and is marked with a dotted line in Figure 43. The corresponding robust frontier (with a δ-neighborhood) can be written as follows:

$$\begin{aligned} f_2 &= (1 - f_1^2) - \frac{1}{3}\delta_1^2 + \left[\alpha \frac{1}{2\delta_1} \log\left(\frac{0.2 + f_1 + \delta_1}{0.2 + f_1 - \delta_1}\right)\right. \\ &\quad \left. + \beta\left({f_1}^2 + \frac{1}{3}\delta_1^2\right)\right] \sum_{i=2}^{n} \left(\frac{50}{3}\delta_i^2\right). \end{aligned} \tag{29}$$

This front is shown as the 'simple effective front' in Figure 43. Here, we have chosen $\delta_1 = 0.01$ and $\delta_i = 2\delta_1$ for $i \geq 2$. With the linear constraint, the following relationship between the two objectives emerge for the constrained original Pareto-optimal front (without any robustness consideration):

$$f_2 = \begin{cases} 1 - f_1^2, & \text{if } f_1 \geq 0.5, \\ 1 - f_1^2 + 50(0.1 - 0.2 f_1)^2 (1/(0.2 + f_1) + f_1^2), & \text{otherwise.} \end{cases} \tag{30}$$

This front is marked as 'constrained original front' in Figure 43. Now, for robustness consideration, since every variable x_i is perturbed in the δ_i-neighborhood, the constraint function restricts $x_2 \geq 0.1 - 0.2(x_1 - \delta_1) + \delta_2$ for $x_1 < 0.5$. Substituting this condition in the objective functions, we obtained the constrained effective frontier (or the robust frontier), as follows:

$$f_2 = \begin{cases} 1 - f_1^2, & \text{if } f_1 \geq 0.5, \\ (1 - x_1^2) - \frac{1}{3}\delta_1^2 + \left[\alpha \frac{1}{2\delta_1} \log\left(\frac{0.2 + x_1 + \delta_1}{0.2 + x_1 - \delta_1}\right) + \beta\left(x_1^2 + \frac{1}{3}\delta_1^2\right)\right] \sum_{i=3}^{n} \left(\frac{100}{3}\delta_i^2\right) & \\ \quad + \frac{50}{2\delta_1} \int_{x_1 - \delta_1}^{x_1 + \delta_1} \left(\frac{\alpha}{0.2 + x_1} + \beta x_1^2\right) (0.1 - 0.2 x_1 + 0.2\delta_1 + \delta_2)^2 \, dx_1. & \text{otherwise.} \end{cases} \tag{31}$$

First, we use definition 1 for a robust feasible optimal solution. Figure 44 shows the obtained NSGA-II solutions with $H = 5$ and $H = 100$ neighboring points. The effect of H is clear from the plot. As the number of neighboring points are increased, the obtained solutions get closer to the theoretical frontier (which can be viewed as a robust optimal frontier with $H = \infty$). The RCV values (computed using freshly created 1,000 neighboring points) are plotted for all obtained solutions in Figure 45. It is interesting to note that the solutions obtained without any robustness consideration (marked as 'simple effective front'), all solutions become infeasible when perturbed. However, for the robust optimization runs obtained using more neighboring points (H), constraint violation is less significant. For $H = 100$, the constraint violation is zero, meaning that all solutions are feasible.

Next, we apply the second definition of robustness and compare the results with the above study. Figure 46 shows the robust frontier obtained for $\eta = 0.025$ and $\delta = 0.01$. The corresponding robust frontier with the first definition and $\delta = 0.01$ is also shown in the figure. It is clear that in order to meet the stringent requirement of a maximum normalized perturbation of objective values to 2.5%, the entire robust frontier needs to be moved inside the feasible region. Moreover, although the first definition finds that a portion ($f_1 > 0.5$) of the original Pareto-optimal frontier is robust, the second definition finds the entire original Pareto-optimal front to be non-robust. A completely new frontier emerges to be a robust frontier according to the second definition.

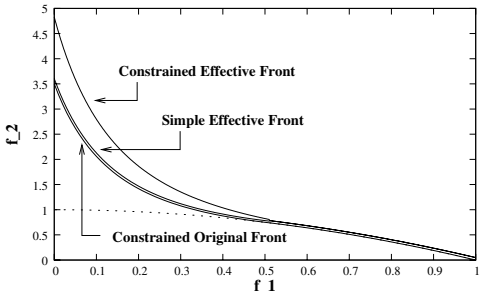


Figure 43: Theoretical simple and constrained effective fronts on test problem 1.

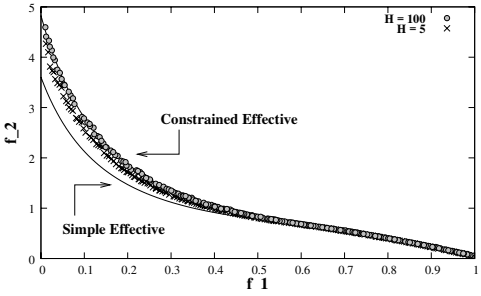


Figure 44: Constrained effective fronts showing the effect of number of points H on test problem 1.

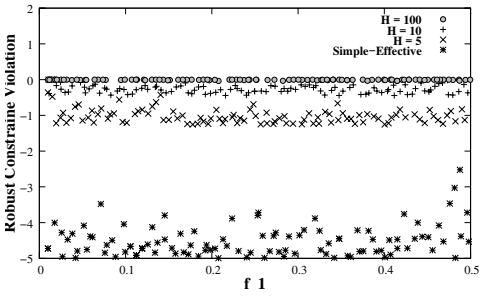


Figure 45: Robust constraint violation for the points obtained on test problem 1 with different values of H.

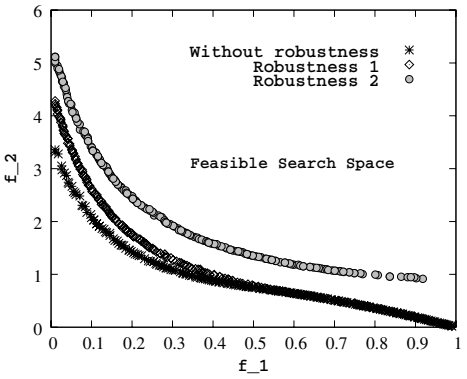


Figure 46: The effect of two approaches of robust optimization is shown on the constrained Pareto-optimal front.

7 Robust Engineering Design: The Welded Beam Problem

This section presents a case study of a robust optimization of a welded beam problem, the single-objective version of which has been commonly used in classical and evolutionary optimization studies (Reklaitis et al., 1983; Deb, 1991). The welded beam design problem is a four-variable constrained optimization problem and was modified to have two objective functions and four highly non-linear constraints (Deb, 2000). A beam needs to be welded on another beam and must carry a certain load F (Figure 47). In the context of single-objective optimal design, it is desired to find four design parameters (weld thickness $x_1 = h$, length of weld $x_2 = \ell$, width of the beam $x_3 = t$, and thickness of the beam, $x_4 = b$) for which the cost of the beam is minimum. The overhang portion of the beam has a length of 14 inches and $F = 6,000$ lb force is applied at the end of the beam. It is intuitive that an optimal design for cost will make all four design variables take small values. When the beam dimensions are small, it is likely that the deflection at the end of the beam will be large. Thus, the design solutions for minimum cost and minimum end-deflection are conflicting with each other. In the following, we present the mathematical formulation of the two-objective optimization

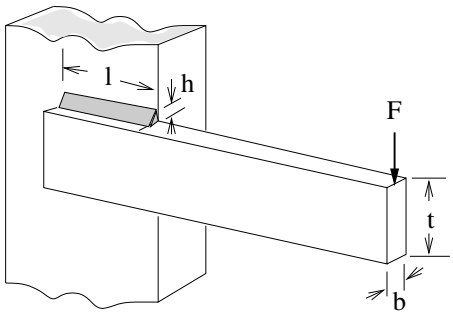


Figure 47: The welded beam design problem. Minimizations of cost and end deflection are two objectives.

problem of minimizing cost and the end-deflection:

$$\begin{array}{ll} \text{Minimize} & f_1(\vec{x}) = 1.10471h^2\ell + 0.04811tb(14.0+\ell), \\ \text{Minimize} & f_2(\vec{x}) = \delta(\vec{x}), \\ \text{Subject to} & g_1(\vec{x}) \equiv 13,600 - \tau(\vec{x}) \geq 0, \\ & g_2(\vec{x}) \equiv 30,000 - \sigma(\vec{x}) \geq 0, \\ & g_3(\vec{x}) \equiv b - h \geq 0, \\ & g_4(\vec{x}) \equiv P_c(\vec{x}) - 6,000 \geq 0. \\ & 0.125 \leq h, b \leq 5.0, \quad 0.1 \leq \ell, t \leq 10.0. \end{array} \tag{32}$$

The deflection term $\delta(\vec{x})$ is given as follows:

$$\delta(\vec{x}) = \frac{2.1952}{t^3 b}.$$

The first constraint makes sure that the shear stress developed at the support location of the beam is smaller than the allowable shear strength of the material ($13,600$ psi). The second constraint makes sure that normal stress developed at the support location of the beam is smaller than the allowable yield strength of the material ($30,000$ psi). The third constraint makes sure that the thickness of the beam is not smaller than the weld thickness from a practical standpoint. The fourth constraint makes sure that the allowable buckling load (along t direction) of the beam is more than the applied load F. A violation of any of the above four constraints will make the design unacceptable. The stress and buckling terms are given as follows (Reklaitis et al., 1983):

$$\begin{aligned} \tau(\vec{x}) &= \sqrt{(\tau')^2 + (\tau'')^2 + (\ell\tau'\tau'')/\sqrt{0.25(\ell^2+(h+t)^2)}}, \\ \tau' &= \frac{6,000}{\sqrt{2}h\ell}, \\ \tau'' &= \frac{6,000(14+0.5\ell)\sqrt{0.25(\ell^2+(h+t)^2)}}{2\left\{0.707h\ell(\ell^2/12+0.25(h+t)^2)\right\}}, \\ \sigma(\vec{x}) &= \frac{504,000}{t^2 b}, \\ P_c(\vec{x}) &= 64,746.022(1-0.0282346t)tb^3. \end{aligned}$$

We use real-parameter GAs with simulated binary crossover (SBX) and polynomial mutation operator (Deb, 2001) to solve this problem.

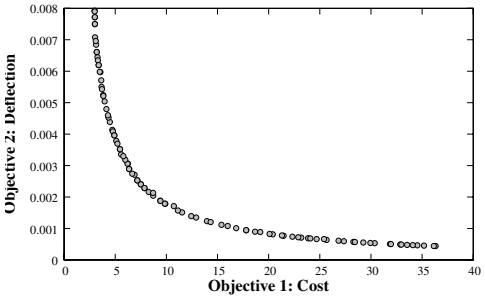


Figure 48: Original Pareto-optimal front for welded beam problem (without any robustness consideration).

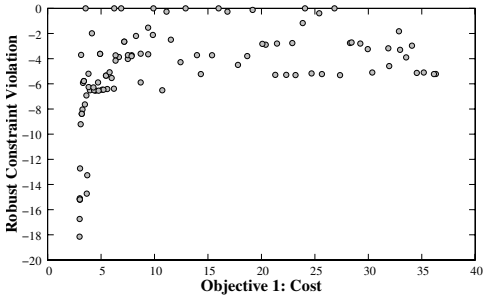


Figure 49: Robust constraint violation for the solutions on original Pareto-optimal front for welded beam problem.

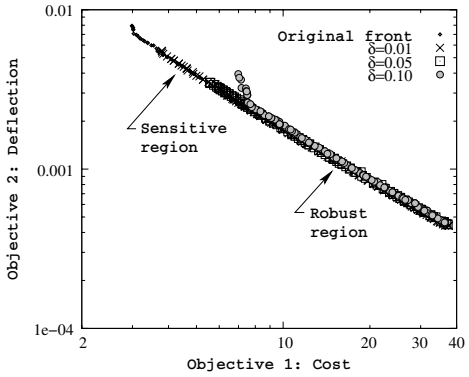


Figure 50: Effect of δ for welded beam problem (Type I robustness).

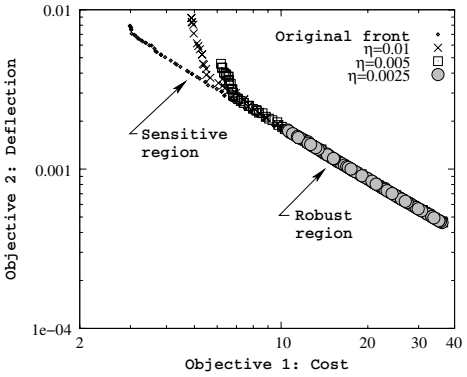


Figure 51: Effect of η for welded beam problem (Type II robustness).

Figure 48 shows the original Pareto-optimal front (without any robustness consideration) for the welded beam optimization problem. The trade-off between the objectives is clear and the result agrees with previously reported results (Deb, 2000). However, Figure 49 shows the robust constraint violation value (computed with 1,000 new solutions in the δ-neighborhood) for every obtained Pareto-optimal solution. It can be seen that solutions close to a minimum-cost solution are quite sensitive to variable perturbation (causing a large negative value of RCV) and thus are likely to be non-robust solutions according to the definitions of robustness.

Figure 50 shows the resulting robust frontier (type I) for three different values of perturbation factor δ and for $H = 50$. To show the differences among different δ-cases, we have used a log-log plot of two objectives. The original (non-robust) front is also shown in the figure. It can be seen that as δ is increased, the minimum-cost portion of the original Pareto-optimal frontier becomes more and more non-robust. There is a slight movement robust frontier from the original Pareto-optimal frontier along the complete range of optimal solutions due to the use of mean function values in plotting the figure, instead of the actual function value at each point. Interestingly, the minimum-cost region is more sensitive to parameter fluctuations than the minimum-end-deflection region, for which almost an identical front is found by all three δ values.

Thus, this welded beam design problem is an instance of the Case 2 problem set, discussed in Section 3.1.2 earlier.

Next, we apply the type 2 robustness principle with the following additional constraint:

$$\max_{i=1,2} \left(\frac{|f_i - f_i^{\text{eff}}|}{|f_i|} \right) \leq \eta.$$

The constraint is handled using the constraint-domination principle (Deb et al., 2002) in NSGA-II. Figure 51 shows the effect of varying η. Here, we have used $\delta = 0.01$ and $H = 50$ on a log-log plot of two objectives. It can be seen that to achieve a tighter restriction on the normalized change in effective objective values from the original objective values (small η), the constrained effective frontier must have to deviate significantly inside the feasible region around the minimum-cost portion of the Pareto-optimal frontier. There is also a shrink in the range of the frontier with smaller values of η. However, the minimum-end-deflection region is relatively unperturbed, as was also found in the type 1 study above. Another interesting aspect emerges from Figure 52, in which the variable x_3 is plotted with the cost objective for respective non-dominated solutions obtained for the original problem, type 1 (with $\delta = 0.01$) and type 2 (with $\eta = 0.0025$ and $\delta = 0.01$) robustness considerations. It is clear that the type 1 robust solutions

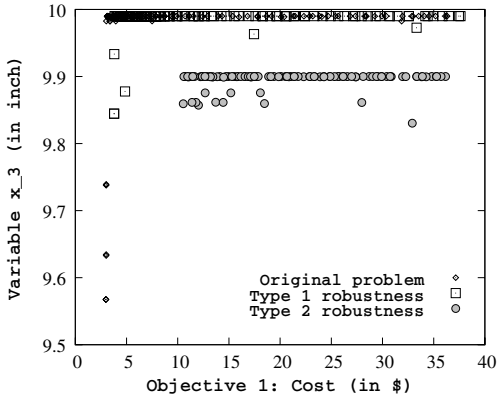


Figure 52: Effect of robustness consideration on variable x_3 (t).

are not significantly different from the original solutions (except near the minimum-cost solution). These solutions are very close to the upper bound used for variable x_3. However, the type 2 robust solutions are significantly different from the original optimal solutions. To make the solutions more reliable with respect to the η constraint, the solutions now must be placed well inside the allowable range of values of x_3. The application of the robust EMO methodology not only discovers the robust frontier in the objective space, the resulting solution vectors also indicate how the original solution vectors must have to be changed in an optimal manner to make them robust or insensitive to the variable perturbations.

8 Conclusions

This paper has taken an important step towards defining robust multi-objective solutions. First, a straightforward extension of a mean effective objective approach, suggested for single-objective evolutionary optimization, has been defined for multiple

objectives. In this approach (we have redefined it as a robust optimization of type I), an EMO methodology has been applied to minimize the mean effective objectives obtained by averaging a finite set of neighboring solutions. Second, we have suggested a robust optimization of type II, in which the original objectives are optimized, but they have an additional constraint restricting the change in objective values to remain within a pre-defined threshold. We have argued that such a procedure is more practical, as it allows a user to find robust solutions with a user-defined limit to the extent of change in objective values with respect to local perturbations.

Additionally, we have identified four different scenarios which can occur in a robust frontier in real-world problems and suggested a number of variable-wise scalable two and three-objective test problems. Simulation results of NSGA-II on these test problems have been illustrated and explained to understand the differences between the two robust optimization procedures. Finally, both approaches for robustness have been applied to an engineering design problem and the importance of this study in *applied optimization* has been highlighted.

However, the authors are aware that this paper has only scratched the surface of an important and pragmatic research topic in applied optimization. There still exists a number of salient implementational issues of a robust optimization procedure. In this research, we have considered $H = 50$ or 100 neighboring solutions to compute the mean effective objectives. Thus, in principle, this method is 50 or 100 times more computationally expensive than the regular non-robust optimization methods. This issue needs an immediate solution before such a method becomes really practical. We are currently pursuing the use of an updatable archive to store a large number of previously-computed solutions. To compute the mean effective objective value of a new solution, the neighboring solutions from the archive can be borrowed, thereby reducing the need for new evaluations. Such a technique has been successfully tried for single-objective robust optimization (Branke and Schmidt, 2000) and may be useful for multi-objective robust optimization as well. However, new insertion and deletion rules honoring the two distinct goals of multi-objective optimization – convergence and distribution – may have to be considered.

Hopefully, this proof-of-the-principle study will motivate more detailed studies in the future and encourage interested readers to understand and apply robust optimization procedures to real-world multi-objective optimization problems.

References

Branke, J. (1998). Creating robust solutions by means of an evolutionary algorithm. *Parallel Problem Solving from Nature*, pages 119–128, LNCS 1498, Springer-Verlag.

Branke, J. (2000). Efficient evolutionary algorithms for searching robust solutions. *ACDM*, pages 275–286.

Branke, J. and Schmidt, C. (2000). Faster convergence by means of fitness estimation. In *Soft Computing*, 9(1):13–20.

Coello, C. A. C., VanVeldhuizen, D. A., and Lamont, G. (2002). *Evolutionary Algorithms for Solving Multi-Objective Problems*. Boston, MA: Kluwer Academic Publishers.

Deb, K. (1991). Optimal design of a welded beam structure via genetic algorithms. *AIAA Journal*, 29(11):2013–2015.

Deb, K. (2000). Multi-objective evolutionary optimization: Past, present and future. In Parmee, I. C., editor, *Evolutionary Design and Manufacture*, pages 225–236. London: Springer.

Deb, K. (2001). *Multi-Objective Optimization Using Evolutionary Algorithms*. Chichester, UK: Wiley.

Deb, K., Agrawal, S., Pratap, A., and Meyarivan, T. (2002). A fast and elitist multi-objective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197.

Deb, K. and Gupta, H. (2005). Searching for robust pareto-optimal solutions in multi-objective optimization. In *Proceedings of the third International Conference on Evolutionary Multi-Criterion Optimization (EMO-05)*, pages 150–164.

Hughes, E. J. (2001). Evolutionary multi-objective ranking with uncertainty and noise. In *Proceedings of the First International Conference on Evolutionary Multi-Criterion Optimization (EMO-01)*, pages 329–343.

Jin, Y. and Sendhoff, B. (2003). Trade-off between performance and robustness: An evolutionary multiobjective approach. In *EMO2003*, pages 237–251.

Miettinen, K. (1999). *Nonlinear Multiobjective Optimization*. Kluwer, Boston.

Montgomery, D. C. (2001). *Design and analysis of experiments*. New York, Wiley.

Parmee, I. (1996). The maintenance of search diversity for effective design space decomposition using cluster-oriented genetic algorithms(cogas) and multi-agent strategies(gaant). In Ian Parmee and M. J. Denham, Eds., *Adaptive Computing in Engineering Design and Control'96 (ACEDC'96)*.

Reklaitis, G. V., Ravindran, A., and Ragsdell, K. M. (1983). *Engineering Optimization Methods and Applications*. New York : Wiley.

Stagge, P. (2001). Averaging efficiently in the presence of noise. In *Proceedings of the First International Conference on Evolutionary Multi-Criterion Optimization (EMO-01)*, pages 314–328.

Teich, J. (1998). Pareto-front exploration with uncertain objectives. In Eiben, A. E., äck, T., Schoenauer, M., and Schwefel, H. P., editors, *PPSN V: Proceedings of the 5th International Conference on Parallel Problem Solving from Nature*, pages 188–200. Springer.

Tsutsui, S. and Ghosh, A. (1997). Genetic algorithms with a robust solution searching scheme. *IEEE Transactions on Evolutionary Computation*, pages 201–219.