Introducing Robustness in Multi-Objective Optimization

Kalyanmoy Deb and Himanshu Gupta
Kanpur Genetic Algorithms Laboratory (KanGAL)
Indian Institute of Technology Kanpur
Kanpur, PIN 208016, India
{deb,himg}@iitk.ac.in
http://www.iitk.ac.in/kangal

KanGAL Report Number 2004016

Abstract

In optimization studies including multi-objective optimization, the main focus is placed in 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 global best solutions, particularly if these solutions are sensitive to the variable perturbations which cannot be avoided in practice. In such cases, practitioners are interested in finding the so-called *robust* solutions which are less sensitive to small changes in variables. Although robust optimization is dealt in detail in single-objective 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 efficient 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 developed six two and three-objective test problems and show simulation results using an evolutionary multi-objective optimization (EMO) algorithm. The interesting results of this paper should encourage further studies and applications of robust multi-objective optimization.

1 Introduction

For the past decade and more, the primary focus of the research and application in the area of evolutionary multi-objective optimization (EMO) has been to find the globally best Pareto-optimal solutions [1, 2]. Such solutions are non-dominated to each other. Simply stated, there exists no other solution in the entire search space which dominate 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 cannot not be implemented to the desired accuracy and the implemented solution may be somewhat different from the theoretical global optimal solution. If a global optimal solution is quite *sensitive* to such variable perturbation in its vicinity, the implemented solution may result in a different set of objective values than that of the theoretical optimal solution. Thus, from a practical standpoint, such solutions are of not much importance and the emphasis must be made in finding *robust* solutions, which are less sensitive to variable perturbations in their neighborhoods.

In single-objective optimization, a number of studies have been devoted for finding robust solutions. Branke [3] suggested a number of heuristics for searching robust solutions. In another study, Branke [4] suggested a number of methods for alternate fitness estimation. Later, Branke

[3] also pointed out key differences between searching optimal solutions in a noisy environment and searching for robust solutions. Jin and Sendhoff [5] 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. Tsutsui and Ghosh [6] presented a mathematical model for obtaining robust solutions using the schema theorem for single-objective genetic algorithms. Parmee [7] suggested a hierarchical strategy of searching several high performance regions in a fitness landscape simultaneously. However, to our knowledge there exists no study on robust multi-objective optimization till to date.

In this paper, we make an effort to extend an existing approach 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 efficient front may turn out to be the robust frontier. To illustrate the working of this approach, we suggest six different test problems and employ NSGA-II [8] 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 user has a control on 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.

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 one of the two robust optimization schemes and simulation results obtained using NSGA-II. Section 5 discusses the second robust approach and presents simulation results. Finally, a conclusion of this study is presented in Section 6.

2 Robustness in 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 the neighborhood. Let us consider Figure 1. Of the two optimal solutions, solution A is considered robust, as a small perturbation of the decision variables does not alter the objective function value of the solution by any 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 suggested different procedures of defining and finding such robust solutions (like solution A) in a single-objective optimization context [9, 4, 5, 6, 7]. There could exist several other ways to define and find a robust solution.

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*

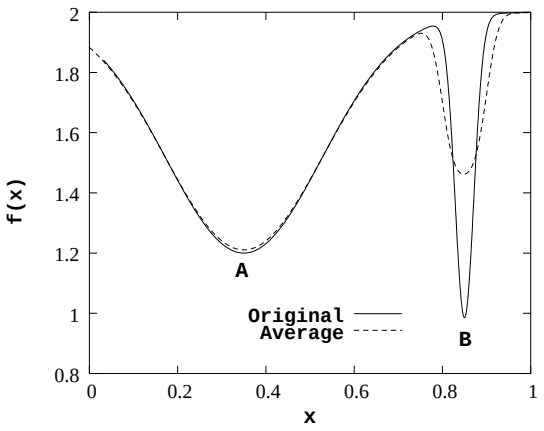

Figure 1: Illustration of global versus robust solutions in a single-objective optimization problem.

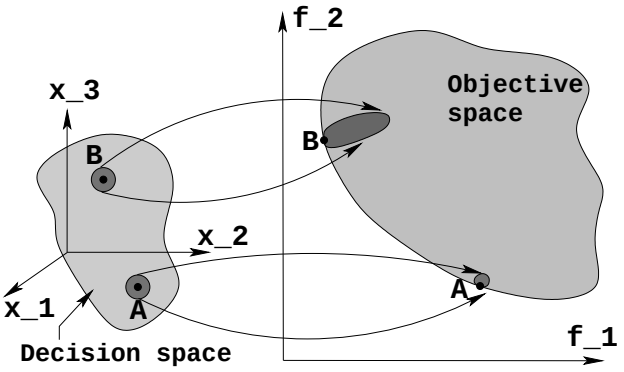

Figure 2: Point A is less sensitive to variable perturbation than point B.

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) 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 used for a faster computation [4].

Another approach would be to calculate the normalized difference in values between the perturbed function value f^{p} 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:*

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

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. 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 a direct control of defining the extent of desired robustness through the parameter η. This method also requires the computation of H neighboring solutions. In some applications, the difference $\|f^{\text{p}}(\mathbf{x}) - f(\mathbf{x})\|$, instead of the normalized difference, can be restricted to η.

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 all are considered to be better than any other solution [1, 10]. 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 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* $\mathbf{x}$*):*

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

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

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

Due to the variable sensitivities, a part of the original global 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 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. 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.

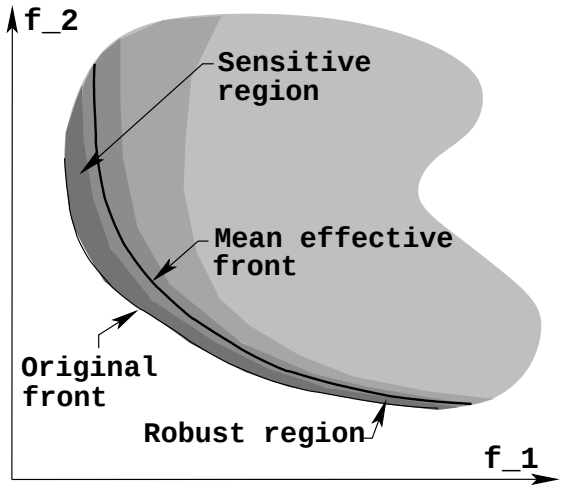

Figure 3: Case 1: Complete efficient front is robust.

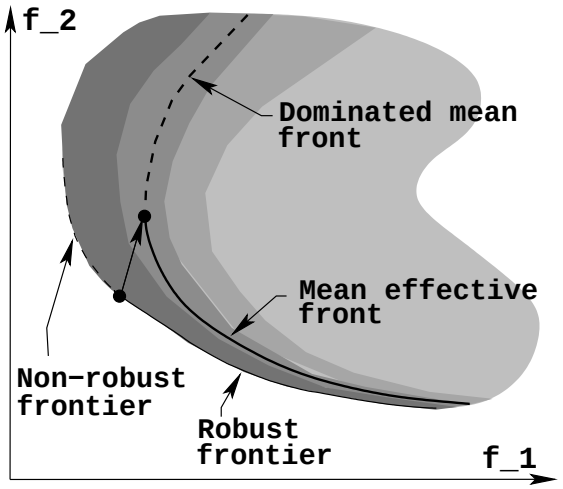

Figure 4: Case 2: A part of the efficient front is robust.

3.2 Case 2

Here, the entire original efficient front is not robust with respect to the above definition of robustness of type I. In most real-world scenarios 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, the task of a multi-objective 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.

3.3 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 are sensitive to local perturbation. Figure 5 demonstrates such a problem. This type of problems, if encountered, must be solved for finding the robust efficient front, instead of the sensitive global efficient front.

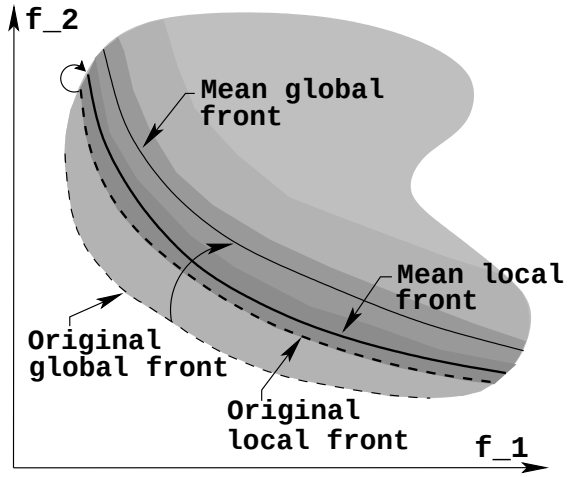

Figure 5: Case 3: The complete global efficient front is not robust.

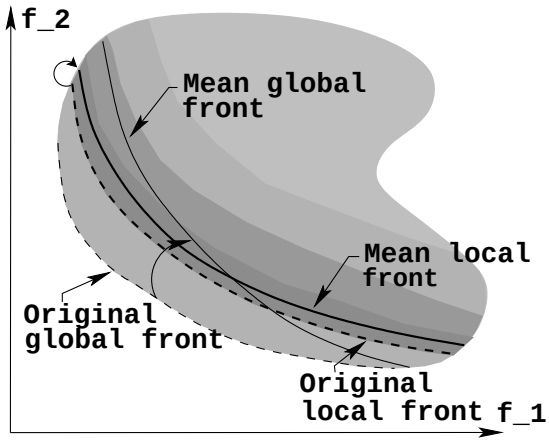

Figure 6: Case 4: A part of the global efficient front is not robust.

3.4 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. 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. However, we argue that the above four scenarios most likely cover different types of robust multi-objective optimization problems which can be encountered in practice and an algorithm capable of solving these scenarios would be adequate in solving other simpler kinds.

3.5 Test Problems

In this section, we now construct mathematical test problems for each of the above four cases.

3.5.1 Test Problem 1

This problem is an illustration to 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 making

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 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:

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

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

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). However, for $0 < \delta_i \leq \delta_i^{\text{cr}}$, all original Pareto-optimal solutions are robust solutions – a matter we shall discuss further in Section 4.

3.5.2 Test Problem 2

This problem is an illustration to 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 that 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.

3.5.3 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:

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

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:

$$f_2 = 1 - \sqrt{f_1} \quad \text{(global)},$$
$$f_2 = 1.2(1 - \sqrt{f_1}) \quad \text{(local)}.$$

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 $H(x_2^*, \delta_2)$ is given as follows:

$$\begin{aligned} H(x_2^*, \delta_2) &= \frac{1}{2\delta_2}\int_{x_2^*-\delta_2}^{x_2^*+\delta_2} h(y)dy \\ &= 2 - \frac{0.1\sqrt{\pi}}{2\delta_2}\left(\text{erf}\left(\frac{x_2^* - 0.35 + \delta_2}{0.25}\right) - \text{erf}\left(\frac{x_2^* - 0.35 - \delta_2}{0.25}\right)\right) \\ &\quad - \frac{0.015\sqrt{\pi}}{2\delta_2}\left(\text{erf}\left(\frac{x_2^* - 0.85 + \delta_2}{0.03}\right) - \text{erf}\left(\frac{x_2^* - 0.85 - \delta_2}{0.03}\right)\right). \end{aligned}$$

For the global and local Pareto-optimal solutions, $x_2^* \approx 0.85$ and 0.35, respectively, can be substituted. Figure 7 plots $H(0.85, \delta_2)$ and $H(0.35, \delta_2)$ as they vary with δ_2. It is clear that at $\delta_2 = 0$ and its neighborhood, the the front corresponding to $x_2^* = 0.85$ has a smaller $H()$ value, thereby making it the global frontier. On the other hand, when δ_2 is greater than about 0.02723, the reverse happens. For example, for $\delta_2 = 0.03$, $H(0.85, 0.03) > H(0.35, 0.03)$, thereby making the front corresponding to $x_2^* = 0.35$ the robust front. We discuss more about the robust frontier of this problem in Section 4.

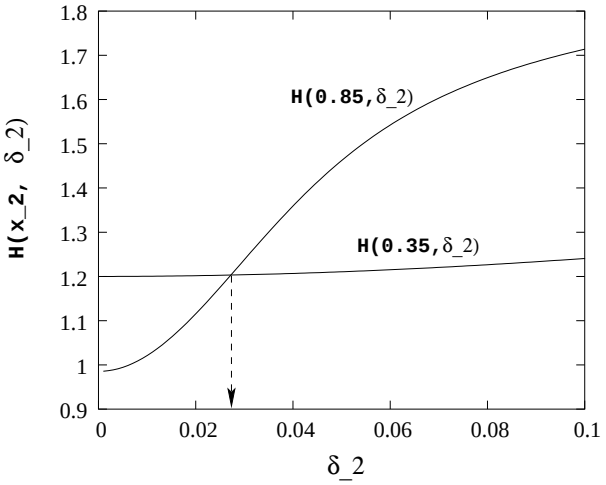

Figure 7: Variation of $H(x_2, \delta_2)$ with δ_2 for $x_2 = 0.85$ and $x_2 = 0.35$ for test problem 3.

3.5.4 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 is $h()$ 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 $x_1 + x_2 = 0.35$. Although the above relationship is true for all global Pareto-optimal solutions in the range $f_1 \in [0, 1]$, the above relationship for local Pareto-optimal solutions cannot exist for $f_1 > 0.5$. At $f_1 = 0.5$ (that is, $x_1 = 0.5$), the variable x_2 takes the lower bound ($= -0.15$) and thereafter the local front cannot satisfy $x_1 + x_2 = 0.35$ constraint. Thus, for $f_1 > 0.5$, solutions on the local efficient front follows $x_2 = -0.15$. 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.

3.5.5 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 making the following relationship between optimal objective values:

$$f_3 = 2 - f_1^2 - f_2^2. \tag{15}$$

The mean effective objective values in a δ-neighborhood for a Pareto-optimal solution, $\mathbf{x}$, are given as follows:

$$f_1^{\text{eff}}(\mathbf{x}) = x_1, \tag{16}$$

$$f_2^{\text{eff}}(\mathbf{x}) = x_2, \tag{17}$$

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

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.

3.5.6 Three-Objective Test Problem 2

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

$$\begin{array}{rl}
\text{Minimize} & f_1(\mathbf{x}) = x_1, \\
\text{Minimize} & f_2(\mathbf{x}) = x_2, \\
\text{Minimize} & f_3(\mathbf{x}) = h(x_3)\left(g(\mathbf{x}) + S(x_1, x_2)\right), \\
\text{Subject to} & 0 \le x_1, x_2, x_3 \le 1, \\
& -1 \le x_i \le 1, \quad i = 4, 5, \ldots, n, \\
\text{where} & h(x_3) = 2 - 0.8\exp\left(-\left(\frac{x_3-0.35}{0.25}\right)^2\right) - \exp\left(-\left(\frac{x_3-0.85}{0.03}\right)^2\right), \\
& g(\mathbf{x}) = \sum_{i=4}^{n}(10 + x_i^2 - 10\cos(4\pi x_i)), \\
& S(x_1, x_2) = 10 - \sqrt{x_1} - \sqrt{x_2}.
\end{array} \tag{19}$$

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:

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

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:

$$f_1^{\text{eff}}(\mathbf{x}) = x_1, \tag{20}$$

$$f_2^{\text{eff}}(\mathbf{x}) = x_2, \tag{21}$$

$$\begin{array}{rcl}
f_3^{\text{eff}}(\mathbf{x}) & = & H(x_3^*, \delta_3)\left[\sum_{i=4}^{n}\left(10 + \frac{\delta_i^2}{3} - \frac{10}{4\pi\delta_i}\sin(4\pi\delta_i)\right)\right. \\
& & \left. +10 - \sum_{i=1}^{2}\left(\frac{1}{3\delta_i}\left((x_i+\delta_i)^{1.5} - (x_i-\delta_i)^{1.5}\right)\right)\right]
\end{array} \tag{22}$$

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

4 Simulation Results

Here, we use NSGA-II [8] procedure to find the robust Pareto-optimal solutions, although any other EMO algorithm can also be used. Various parameters which would 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 to 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, we show the effect of this parameter to particularly show its relation with the extent of robustness in the Pareto-optimal solutions.

The effect of all three parameters was 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 [9]. The simplest strategy can be to randomly create H points in the neighborhood of every solution. 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. To create a pattern systematically, we divide the perturbation domain of each variable (around $[-\delta_i, \delta_i]$) into exactly H equal grids, thereby dividing the δ-neighborhood into n^H small hyperboxes. Thereafter, we pick exactly H hyperboxes randomly from n^H hyperboxes so that in each dimension all H distinct grids are represented. Figure 8 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. Such a procedure also ensures that the chosen H neighboring points are distributed all around a solution.

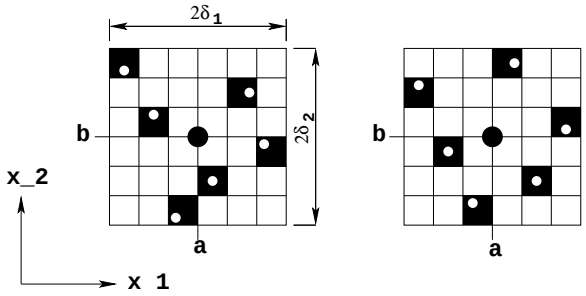

Figure 8: Two examples of creating $H = 6$ neighboring points around a solution $\mathbf{x} = (\mathtt{a}, \mathtt{b})$.

In all simulations, we have used the simulated binary crossover (SBX) and the polynomial mutation operator with distribution indices of 10 and 50, respectively [1]. A population size of 100 is run for a long enough (10,000) generations to have confidence in the location of the robust optimal 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 9 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 was convex. It is important to highlight here that for a robust optimization of type I an EMO algorithm works with the mean effective objectives and thus may have difficulty in solving the robust optimization problem of handling a non-convex problem compared to the original convex problem.

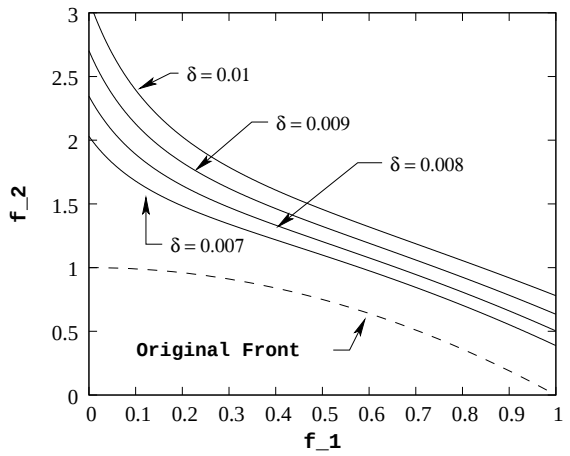

Figure 9: Theoretical mean effective fronts showing the effect of δ on test problem 1.

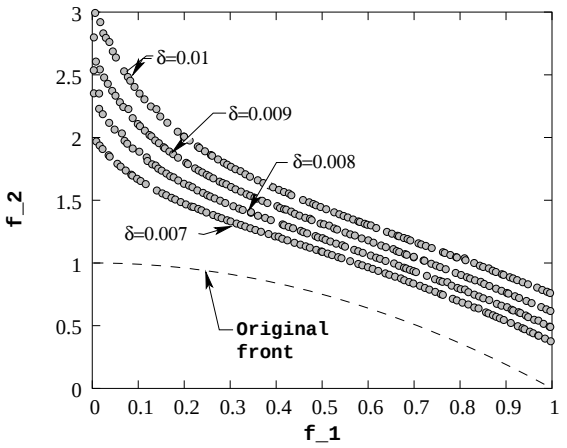

Figure 10: Robust solutions obtained using NSGA-II show the effect of δ on test problem 1.

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, assume that $A = \sum_{i=2}^{n} 80\pi^2\delta_i^2/3$. 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 ($\bar{f}_1^{\text{eff}}$):

$$\bar{f}_1^{\text{eff}} \left(0.2 + \bar{f}_1^{\text{eff}}\right)^2 = \frac{0.5\alpha}{\beta - 1/A}. \tag{23}$$

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

$$A \leq 1/(\beta - 0.3472\alpha) \tag{24}$$

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 9, δ values smaller than this critical value are used, all original Pareto-optimal solutions are robust.

Figure 10 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 9).

Figures 11 and 12 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 one, some original Pareto-optimal solutions are no more optimal. Using Equation 23 and substituting $\alpha = 1$ and $\beta = 10$, we obtain $A \leq 0.1036$ for the test problem 2. This results in 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 11 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. When performing a robust multi-objective optimization, an algorithm should then only find the Pareto-optimal solutions which are robust. Figure 12 shows that NSGA-II finds only the non-dominated (or robust) portion of this efficient front. By finding the root of Equation 23 for different values of δ, we find the boundary of different robust fronts and show it on both the

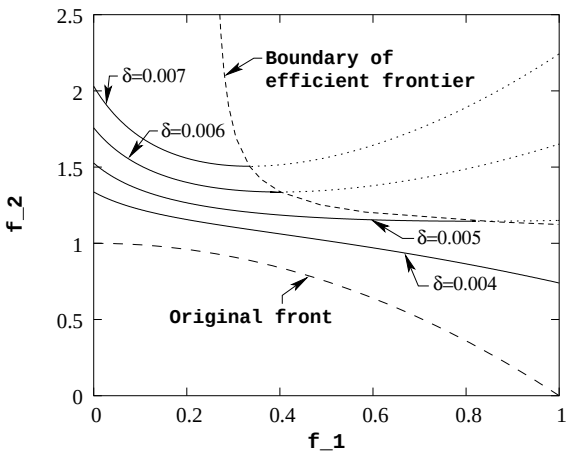

Figure 11: Theoretical mean effective fronts showing the effect of δ on test problem 2.

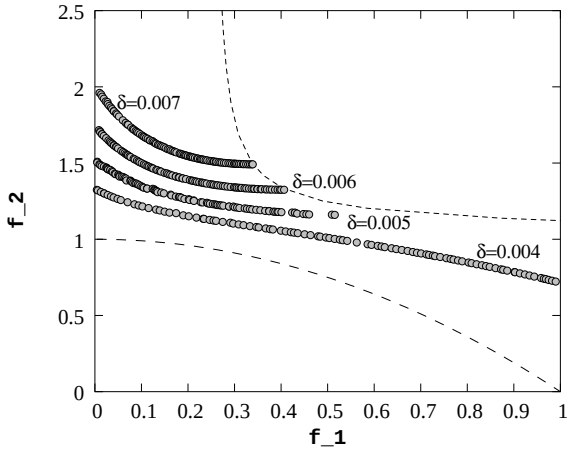

Figure 12: Robust solutions obtained using NSGA-II show the effect of δ on test problem 2.

figures with a dashed line. It is interesting to 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.

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 13 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 14 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.

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 15 and 16 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 the 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 16 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 16 as well. It is intuitive that the

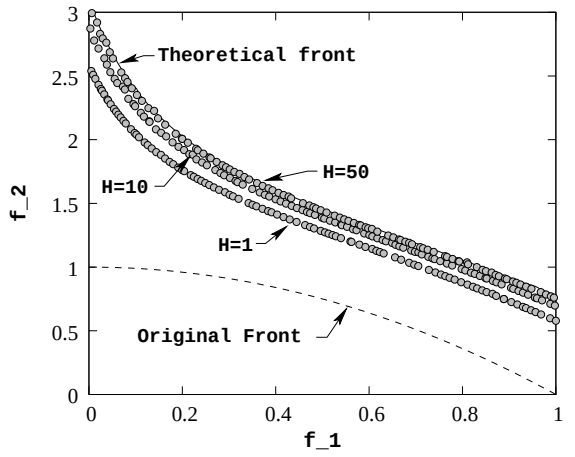

Figure 13: Effect of H (theoretical and NSGA-II) on test problem 1 ($\delta = 0.01$ and $n = 5$).

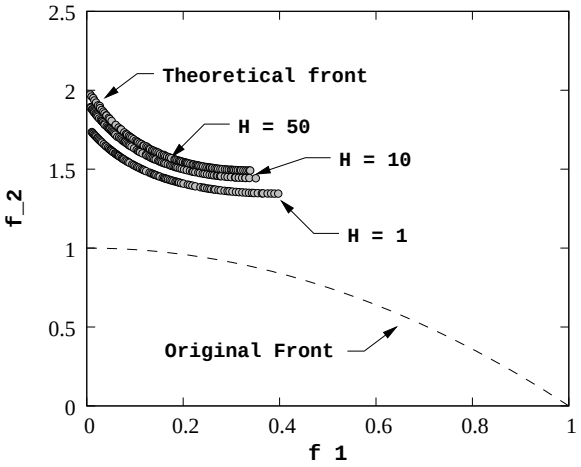

Figure 14: Effect of H (theoretical and NSGA-II) on test problem 2 ($\delta = 0.007$ and $n = 5$).

effect of robustness can be more dramatic in a higher dimensional problem, as depicted by both simulation results.

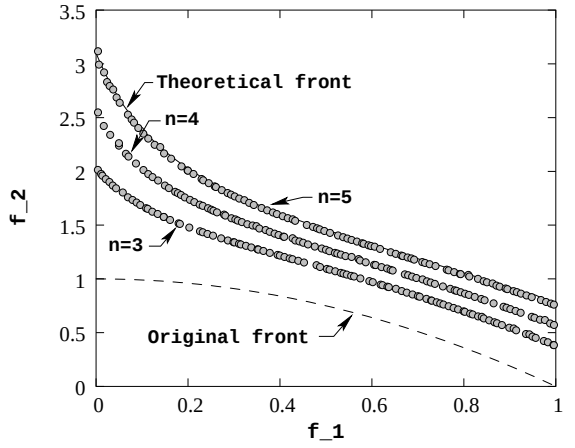

Figure 15: Effect of n (theoretical and NSGA-II) on test problem 1 ($\delta = 0.01$ and $H = 50$).

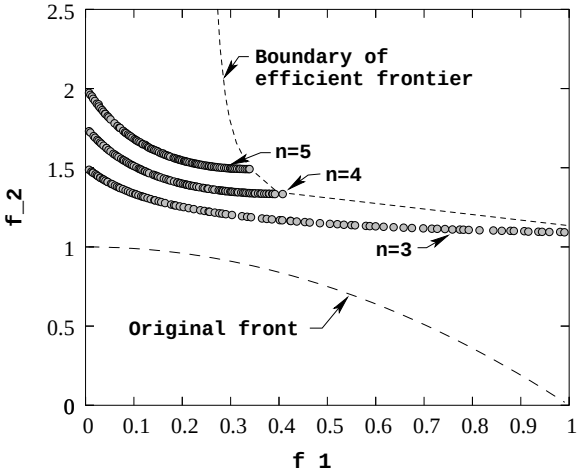

Figure 16: Effect of n (theoretical and NSGA-II) on test problem 2 ($\delta = 0.007$ and $H = 50$).

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 17 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, as depicted in Figure 7.

Figure 18 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, as also can be seen by comparing both figures.

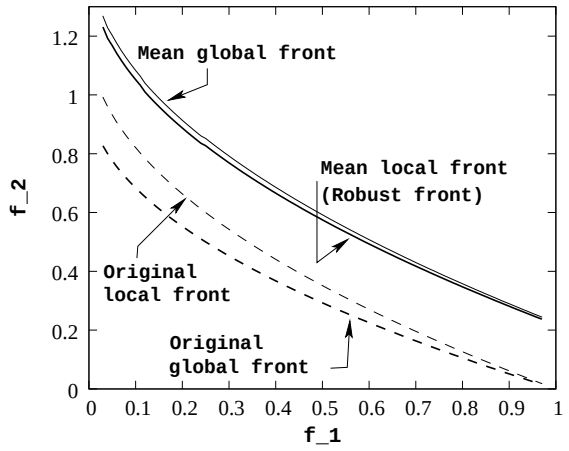

Figure 17: Theoretical robust front for test problem 3 ($\delta = 0.03$).

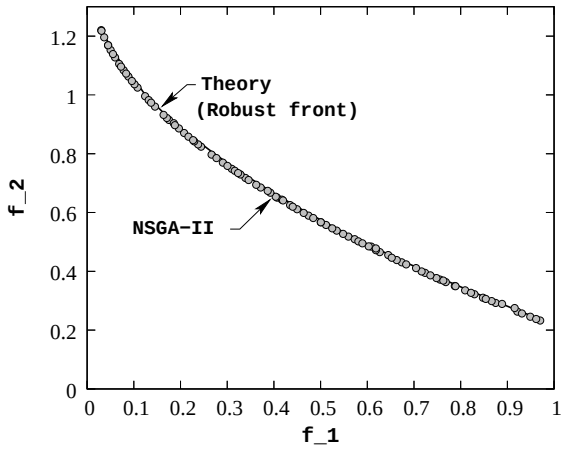

Figure 18: NSGA-II robust front for test problem 3 ($\delta = 0.03$).

To show the difference between the original efficient front and the robust front, we show all 100 obtained NSGA-II solutions for the two cases in Figures 19 and 20, respectively. 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 20 shows the robust solutions. Here, all solutions take a value close to $x_2 = 0.35$.

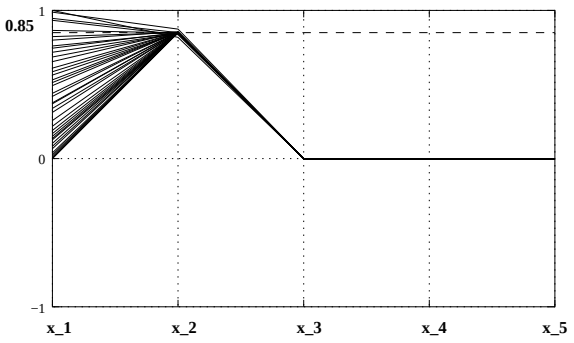

Figure 19: NSGA-II solutions of the original test problem 3.

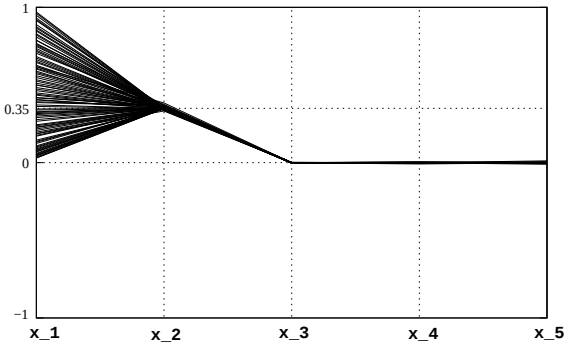

Figure 20: NSGA-II robust solutions for test problem 3.

Next, we consider the test problem 4. Theoretical fronts for the original problem are shown in Figure 21 in dashed lines and corresponding mean effective fronts are shown in solid lines. In both cases, the local efficient frontier takes 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 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 22 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 $H = 500$,

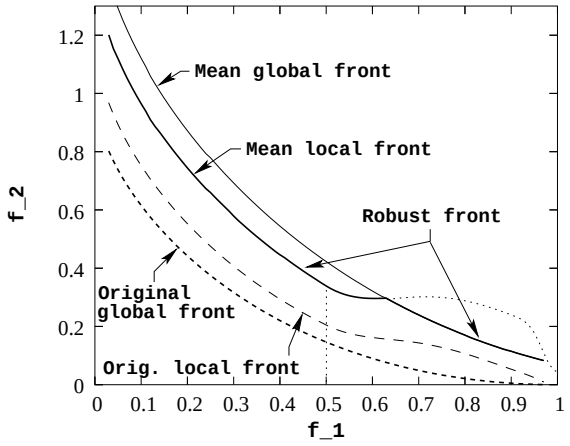

Figure 21: Theoretical robust front for test problem 4.

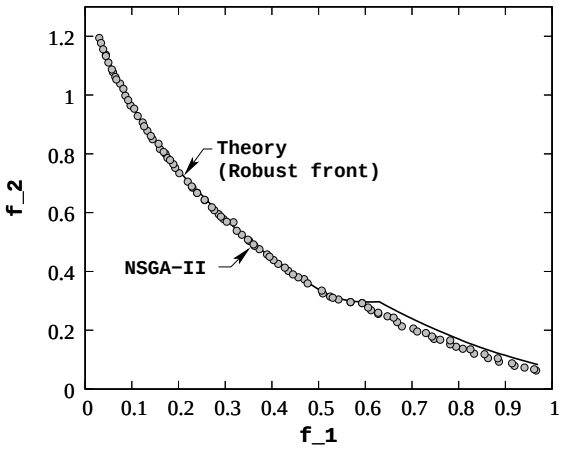

Figure 22: NSGA-II robust front for test problem 4.

the obtained solutions are closer to the theoretical frontier.

Figures 23 and 24 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 23. For $0.5 < f_1 < 0.63$, x_2 gets fixed to its lower bound (-0.15).

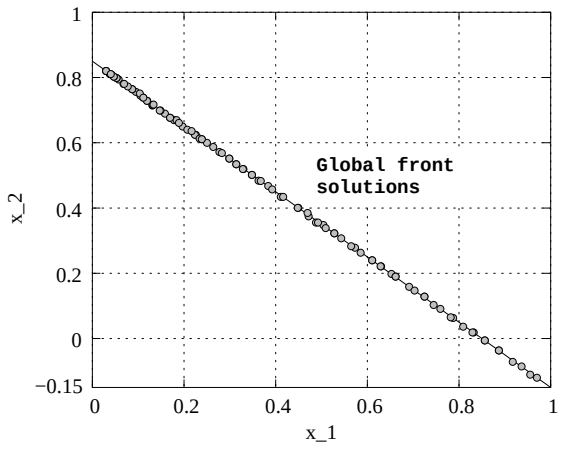

Figure 23: NSGA-II solutions of the original test problem 4.

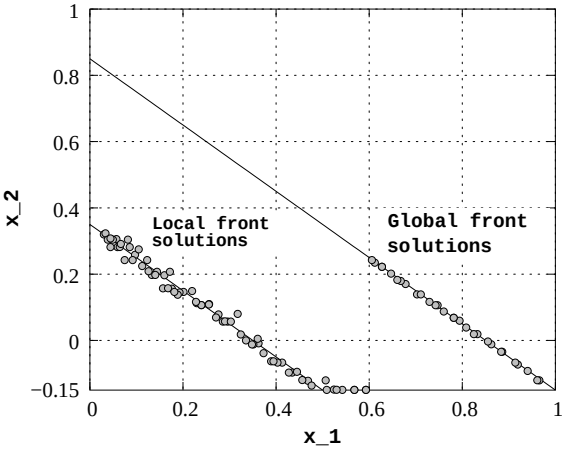

Figure 24: NSGA-II robust solutions for test problem 4.

4.3 Three-Objective Test Problem 1

Figure 25 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 16 to 18. 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 26 shows the robust solutions obtained using NSGA-II. It is amply clear from the figure

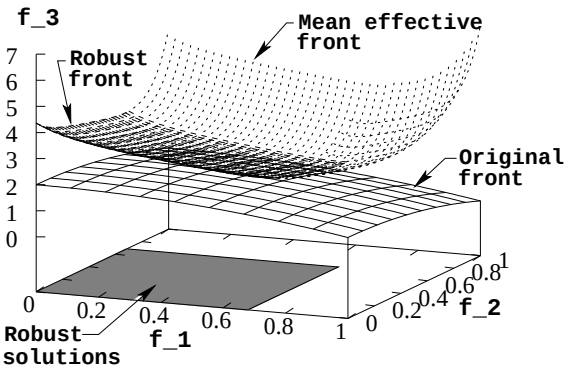

Figure 25: Theoretical Robust Front for three-objective test problem 1

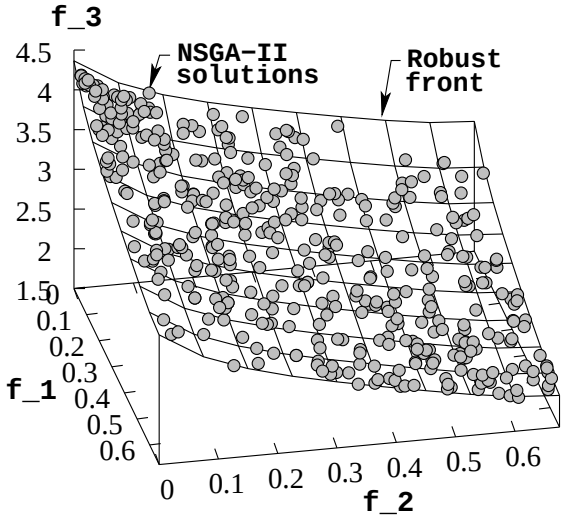

Figure 26: NSGA-II robust solutions for three-objective test problem 1.

that the obtained robust solutions lie on the theoretical efficient frontier shown in Figure 25.

Figure 27 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 28 shows the theoretical efficient fronts (local and global) for this problem. The mean effective local and global fronts are obtained using Equations 20 to 22. It is clear from Figure 28 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 29 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 30 represents the solutions obtained for the original optimization problem. All solutions have x_3 values almost equal to 0.85, while Figure 31 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.

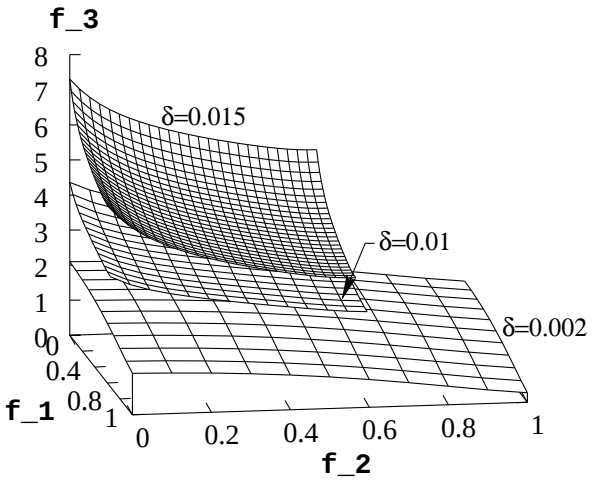

Figure 27: Theoretical effective fronts for different values of δ.

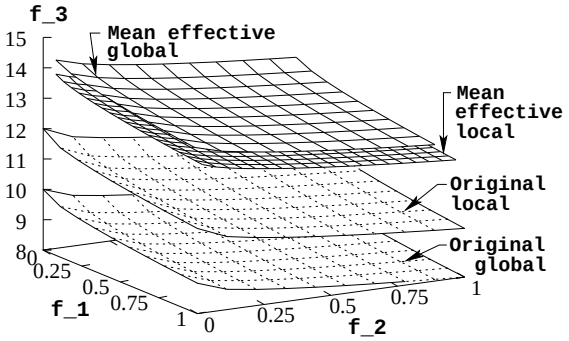

Figure 28: Theoretical robust front of type I for the three-objective test problem 2.

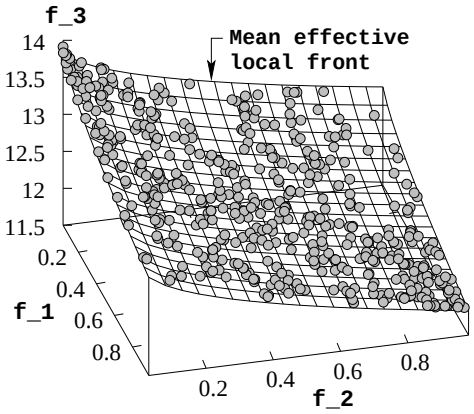

Figure 29: NSGA-II robust solutions of type I for the three-objective test problem 2.

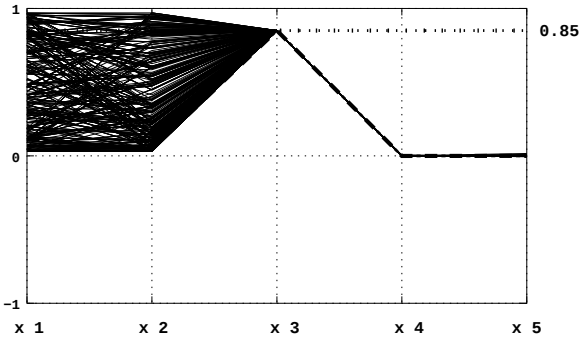

Figure 30: Pareto-optimal solutions of the original three-objective test problem 2.

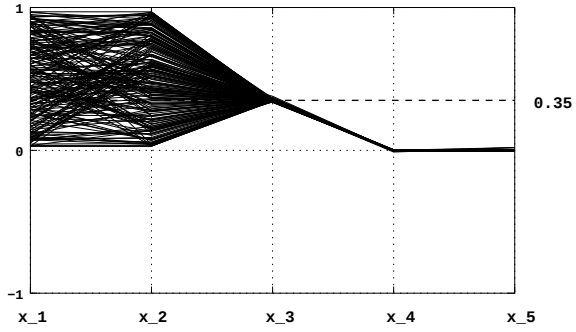

Figure 31: Robust solutions of type I for the three-objective test problem 2.

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 been also 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 in 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 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 the following multi-objective minimization problem:*

$$\left.\begin{array}{ll} \textit{Minimize} & \mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_M(\mathbf{x})), \\ \textit{subject to} & \frac{\|\mathbf{f}^{\mathrm{p}}(\mathbf{x}) - \mathbf{f}(\mathbf{x})\|}{\|\mathbf{f}(\mathbf{x})\|} \leq \eta, \\ & \mathbf{x} \in \mathcal{S}. \end{array}\right\} \tag{25}$$

Here, we use f_j^{eff} for f_j^p and the Euclidean norm for $\|\cdot\|$ operator, but any other suitable functional and norm can also be used. 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 [1].

5.1 Test Problem 1

Figure 32 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 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 32 and 33. To show how the decision variables are distributed across the robust frontier, we plot $g()$ function for different values of η in Figure 33. 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

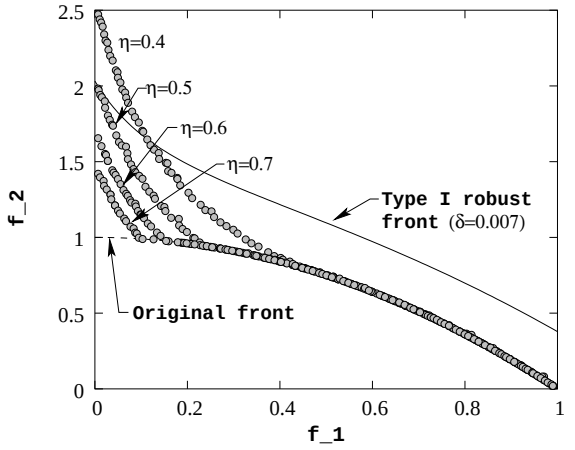

Figure 32: Robust fronts for different values of η obtained by minimizing exact f^{eff} for problem 1.

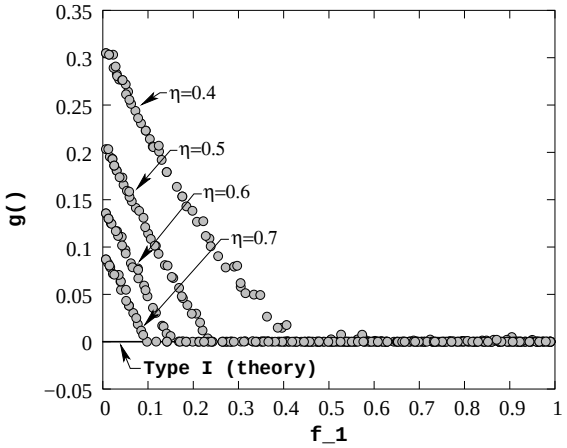

Figure 33: Function $g()$ of the robust solutions shown in Figure 17 for problem 1.

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 solutions shown in Figure 33 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 34 shows the corresponding NSGA-II results obtained by minimizing 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 32).

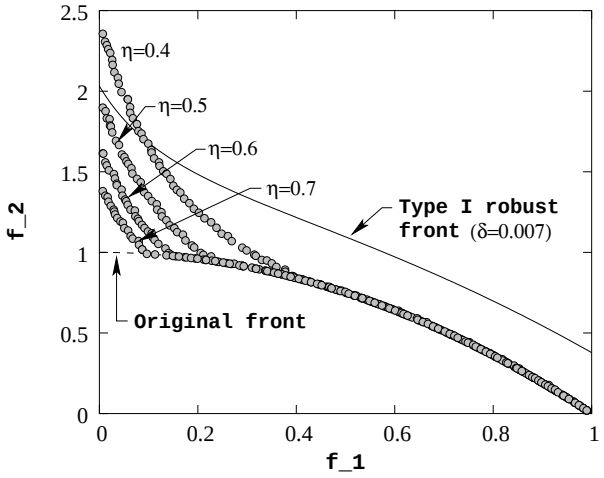

Figure 34: Robust fronts for different values of η obtained by minimizing an average of H neighboring points for test problem 1.

5.2 Test Problem 2

Figure 35 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 $\eta = 0.6$. 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.

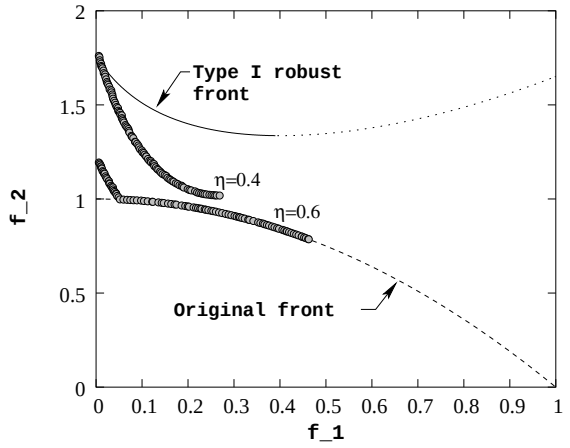

Figure 35: Robust fronts for different values of η for problem 2.

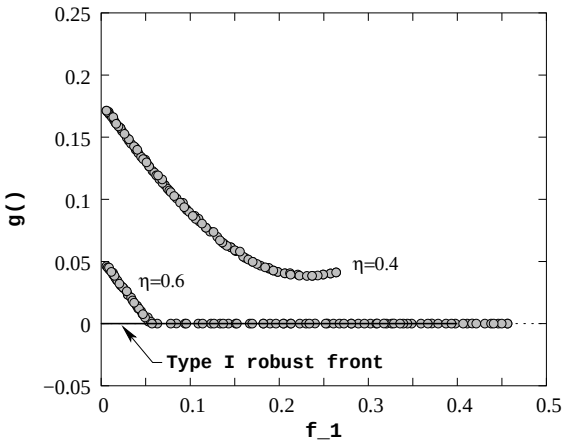

Figure 36: Function $g()$ of the robust solutions shown in Figure 17 for problem 2.

Figure 36 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.

5.3 Test Problems 3 and 4

Figure 37 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 38 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 39 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 the original global efficient front for $f_1 \leq 0.7$. In the remaining portion, the original global efficient front remains to be robust. It is also interesting to note from Figure 40 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.

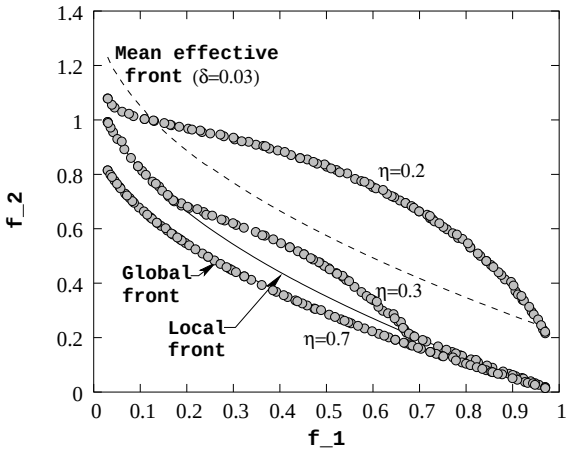

Figure 37: Robust fronts for different values of η for problem 3.

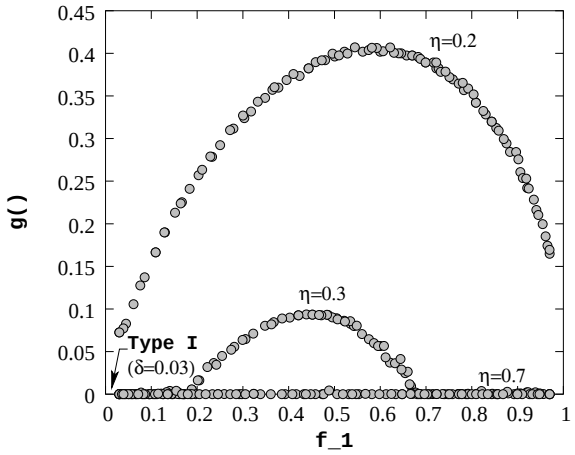

Figure 38: Corresponding $g()$ values show different Pareto-optimal sets obtained by different η. For type I ($\delta = 0.03$) robust solutions, $g() = 0$.

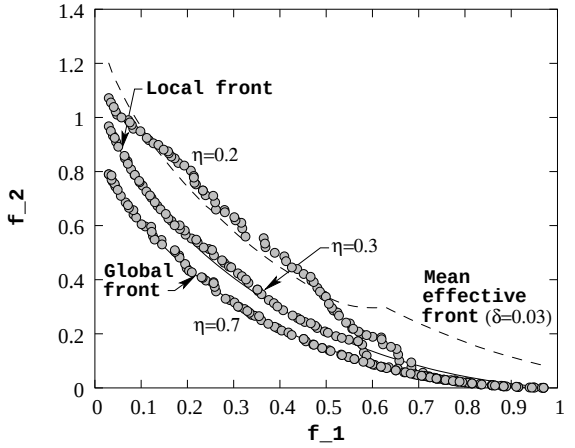

Figure 39: Robust fronts for different values of η for problem 4.

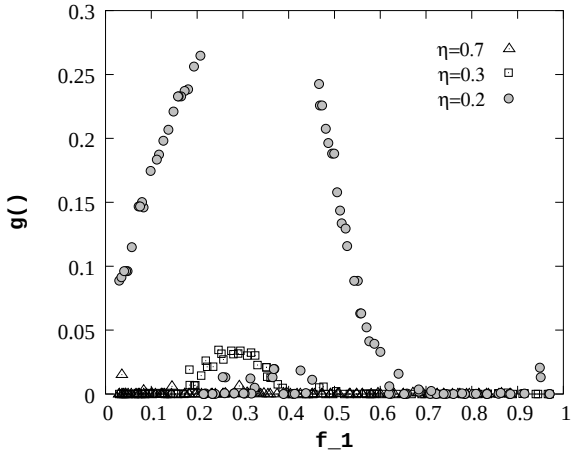

Figure 40: Corresponding $g()$ values show different Pareto-optimal sets obtained by different η.

5.4 Three-Objective Test Problem 1

We now show simulation results on the three-objective test problem 1. Figures 41 and 42 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 41. However, for $\eta = 0.4$ (shown in Figure 42), 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 more 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 $\delta = 0.01$.

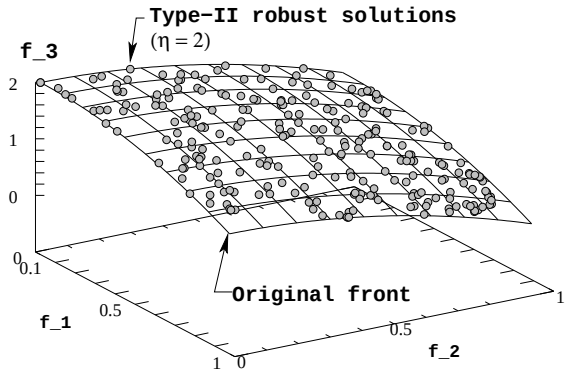

Figure 41: Type-II Robust solutions for $\eta = 2$ for the three-objective test problem 1. Original Pareto-optimal solutions are all robust.

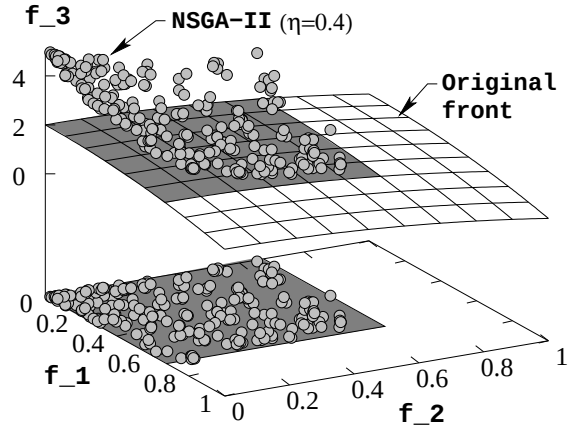

Figure 42: Type-II Robust solutions for $\eta = 0.4$ for the three-objective test problem 1. A part of the original Pareto-optimal solutions are robust.

6 Conclusions

This paper takes a first step towards defining robust multi-objective solutions. First, a straightforward extension of a mean effective objective approach suggested for single-objective optimization is defined for multiple objectives. In this approach (we 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 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 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.

This paper has made an attempt to just scratch 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 of new evaluations. Such a technique has been successfully tried for single-objective robust optimization [4] 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.

Acknowledgements

The first author wishes to acknowledge and thank Juergen Branke, Hartmut Schmeck, and Marco Farina, for some initial discussions the author had with them during his visit to University of Karlsruhe during the summer of 2003.

References

[1] K. Deb. *Multi-objective optimization using evolutionary algorithms.* Chichester, UK: Wiley, 2001.

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

[3] Jurgen Branke. Creating robust solutions by means of an evolutionary algorithm. *Parallel Problem Solving from Nature*, pages 119–128, 1998.

[4] Jurgen Branke and C. Schmidt. Faster convergence by means of fitness estimation. In *Soft Computing*, 2000.

[5] Yaochu Jin and Bernhard Sendhoff. Trade-off between performance and robustness: An evolutionary multiobjective approach. In *EMO2003*, pages 237–251, 2003.

[6] S. Tsutsui and A. Ghosh. Genetic algorithms with a robust solution searching scheme. *IEEE transactions on Evolutionary Computation*, pages 201–219, 1997.

[7] I.C. Parmee. The maintenance of search diversity for effective design space decomposition using cluster-oriented genetic algorithms(cogas) and multi-agent strategies(gaant). In *ACEDC*, 1996.

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

[9] Jurgen Branke. Efficient evolutionary algorithms for searching robust solutions. *ACDM*, pages 275–286, 2000.

[10] K. Miettinen. *Nonlinear Multiobjective Optimization.* Kluwer, Boston, 1999.