Finding Robust Solutions for Many-Objective Optimization Using NSGA-III

Deepanshu Yadav
Indian Institute of Technology Madras
Chennai, India
deepanshu.yadav380@gmail.com

Palaniappan Ramu
Indian Institute of Technology Madras
Chennai, India
palramu@iitm.ac.in

Kalyanmoy Deb
Michigan State University
East Lansing, USA
kdeb@egr.msu.edu

COIN Report Number 2023002

***Abstract*—The primary task of evolutionary multi-objective optimization (EMO) is to find the globally best Pareto-optimal front. However, often decision makers (DMs) are not interested in obtaining the global frontier. Instead, they prefer a set of solutions for which there is no significant change in objective values within a small neighborhood of each decision variable vector, resulting in a set of robust solutions. The corresponding objective vectors are said to lie on the *robust front*. In practical applications, such as in engineering designs, designers are interested in robust designs which are less sensitive to the perturbation in the design variables and parameters, caused by the manufacturing process tolerances, material non-uniformity, uncertainties in supply-chain process, and by many other practical matters. An earlier robust EMO study proposed two different robustness measures and used the elitist non-dominated sorting genetic algorithms (NSGA-II) to find respective robust fronts. However, the limitation of NSGA-II in generating well-distributed and diverse set solutions for many-objective optimization, the robust optimization concept must be extended with evolutionary many-objective optimization (EMaO) algorithms to investigate the efficacy in more than three-objective problems. This study proposes an extension of test problems for robust many-objective optimization tasks and demonstrates the performance of updated NSGA-III procedure to two to eight-objective test and real-world problems.**

***Index Terms*—Robust design, Pareto-optimal front, Efficient frontier, Multi-objective optimization, NSGA-III.**

I. Introduction

EVOLUTIONARY multi-objective optimization (EMO) algorithms have been primarily focused on obtaining a well-distributed and diverse set of Pareto-optimal (PO) solutions [1], [2]. Although, these non-dominated solutions are globally the best solutions from a theoretical optimization point of view, the solutions can be very sensitive to small perturbations in the neighborhood of decision variable vectors for one or more PO solutions. Practitioners usually do not like to adopt sensitive solutions, despite them being optimal, due to the uncertainties involved in implementing and deploying sensitive solutions in their desired forms. If an intended solution gets implemented slightly differently to a different solution, the respective objective vector would be different from the intended objective vector. In the context of multi- or many-objective optimization, the perturbed solution can now become dominated, infeasible, better or stay non-dominated. For these reasons, usually, practitioners are interested in finding out a set of *robust* solutions that are less sensitive to perturbations in the neighborhood of solutions.

Robustness measure can be defined in many ways. In the context of EMO algorithms, the concept of robustness was introduced and discussed for single- and multi-objective optimization problems. An earlier study [3] highlighted the challenges of computing robust solutions in the EMO framework and observed that it involves high computational complexity in terms of more solution evaluations. A study [4] proposed modification to the basic formulation to accommodate robustness in EMO and suggested that EA's ability to create robust solutions can be enhanced without significantly increasing the computational complexity. Interpolation and regression-based fitness estimation methods are proposed for faster convergence of EAs [5]. Another study proposed two robustness measures based on the function variance for the single objective optimization problem in which the search for robust optimal solutions is formulated as a bi-objective optimization problem that minimized the fitness function and the robustness measure [6]. Tsutsui and Ghosh [7] proposed the concept of genetic algorithms with a robust solution-searching scheme (GAs/RS) for evaluating the robust solutions for single-objective problems.

The ranking process for probabilities of selection to compute the robust solution(s) for both single and multi-objective optimization (MOO) problems is re-formulated in [8]. Deb and Gupta [9] presented two procedures for computing robust front for four test problems using NSGA-II [10]. The first procedure is based on minimizing the mean effective function, instead of the original function and the second imposes a robustness constraint for solving the reformulated MOO. They also extended the procedure to handle constraints [11]. Zhou et al [12] proposed a multi-objective robust optimization (MORO) approach based on Gaussian process (GP) model to reduce the computational burden under interval uncertainty. The performance metrics of MORO algorithms are measured in terms of the convergence, coverage, and success rate of robust solutions obtained by robust multi-objective algorithms

979-8-3503-1458-8/23/$31.00 ©2023 IEEE

[13]. Mirjalili [14], [15] studied the current challenges in test problems and proposed a framework to generate robust multi-objective test problems with different adjustable characteristics and difficulty levels.

Despite a number of MORO studies in literature, robustness studies in the context of many-objective optimization involving four or more objectives are rare [16]–[18]. The novelty of the current work is to enable uncertainty handling in many-objective optimization specifically using NSGA-III. This will enable dealing with uncertainty in parameters and variables in solving practical many-objective optimization problems.

The rest of the paper is organized as follows. Section II discusses the concept of multi-objective robust optimization (MORO). A brief discussion on the results of NSGA-III for robust many-objective optimization is presented in Section III. Next, the simulation results of the proposed approach to two test problems and two real-world engineering problems are discussed in Section IV. Section V studies the effect of robustness parameters (δ, H, and η) on hypervolume (HV) metric followed by the conclusions in Section VI.

II. Multi-objective Robust Optimization (MORO)

Contrary to the single-objective optimization problems where a single optimal solution is sought, EMO approaches are used to find a set of Pareto-optimal solutions for multi-objective optimization (MOO) problems [1]. These so-called Pareto-optimal solutions are the best solutions globally, available in the feasible decision space. On a Pareto front, an objective can only be improved by sacrificing at least one other objective.

For the case of multi-objective optimization problems, the concept of robust optimization was introduced and presented in [9]. Consider the following multi-objective optimization problem:

$$\begin{aligned} \min_{\mathbf{x}} \quad & \{f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_M(\mathbf{x})\}, \\ \text{subject to} \quad & \mathbf{x} \in \mathcal{S}, \end{aligned} \tag{1}$$

where $\mathcal{S}$ is the feasible search space. A robust solution $\mathbf{x}$ is defined as the one for which the objective vector $\mathbf{f}(\mathbf{x})$ is insensitive up to a certain level for a perturbation in the neighborhood of $\mathbf{x}$.

Fig. 1 illustrates the sensitivity of the two decision vectors A and B in the objective space. It can be observed that a small perturbation in decision point B causes a large change in objective values as compared to the decision point A. Hence, solution A can be considered more robust than solution B. Unlike the robustness in the case of single-objective optimization problems, in MOO the robustness is defined for M objectives and a set of non-dominated solutions.

Two ideas are widely used in the literature for executing a robust single-objective optimization task [4]:

1) **Robust Solution of Type I:** Mean effective objective function is used for optimization instead of the original objective function.
2) **Robust Solution of Type II:** The normalized difference between the perturbed objective function and the original objective function is used as a constraint to have better control in defining a robust solution.

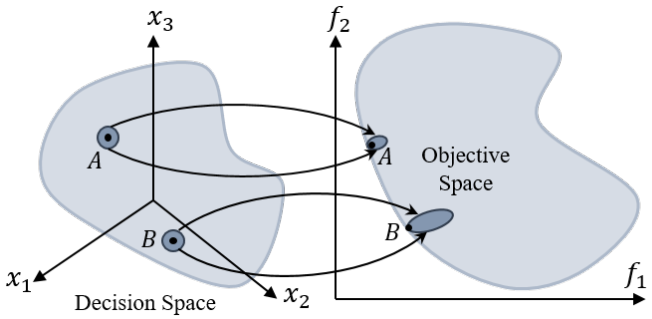


Fig. 1: A and B are two points in decision space. In objective space, solution A is less sensitive as compared to solution B for a small perturbation in the decision variable space. Hence, A is more robust than B.

The above two types of robustness ideas are extended for computing a robust front in the case of multi-objective optimization (MOO) [9]. The definition of the Robust Solution of Type I and Type II for multi-objective optimization defined in [9] is presented as follows:

*Definition 1 (**Robust Solution of Type I**):* Defined in a δ-neighborhood $\big(\mathcal{B}_\delta(\mathbf{x})\big)$, a solution $\mathbf{x}^*$ is called a multi-objective robust solution of Type I, if it is a global feasible Pareto-optimal solution to the following multi-objective minimization problem:

$$\begin{aligned} \min_{\mathbf{x}} \quad & \{f_1^{\text{eff}}(\mathbf{x}), f_2^{\text{eff}}(\mathbf{x}), \ldots, f_M^{\text{eff}}(\mathbf{x})\}, \\ \text{subject to} \quad & \mathbf{x} \in \mathcal{S}, \end{aligned} \tag{2}$$

where f_j^{eff} is defined as follows:

$$f_j^{\text{eff}} = \frac{1}{\mathcal{B}_\delta(\mathbf{x})} \int_{\mathbf{y} \in \mathcal{B}_\delta(\mathbf{x})} f_j(\mathbf{y}) d\mathbf{y}. \tag{3}$$

*Definition 2 (**Robust Solution of Type II**):* A solution $\mathbf{x}^*$ is called a multi-objective robust solution of Type II, if it is a global feasible Pareto-optimal solution to the following multi-objective minimization problem:

$$\begin{aligned} \min_{\mathbf{x}} \quad & \{f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_M(\mathbf{x})\}, \\ \text{subject to} \quad & \frac{||\mathbf{f}^{\text{eff}}(\mathbf{x}) - \mathbf{f}(\mathbf{x})||_2}{||\mathbf{f}(\mathbf{x})||_2} \leq \eta, \\ & \mathbf{x} \in \mathcal{S}. \end{aligned} \tag{4}$$

In formulation (2), the effective objective function of $\mathbf{x}$ can be computed as the mean of H original objective functions evaluated from H solutions in the δ-neighborhood of $\mathbf{x}$. The formulation in (4) optimizes the original multi-objective function in the feasible decision space $\mathcal{S}$ with an additional constraint. The additional constraint only considers those solutions feasible for which the normalized change in the objective vector is less than a user-defined parameter η. Along with the constraint handling strategy, alternate forms of the constraint

defined in (4) are presented and implemented in [11] using the NSGA-II algorithm [10].

Though robust multi-objective optimization problems (MOPs) are introduced and implemented using NSGA-II, such formulations for many-objective problems (MaOPs) rarely exist. We propose robustness for many-objective optimization (MaOP) using NSGA-III here. The key features of NSGA-III such as scalability, diversity preservation, and ability to compute well-distributed and converged solutions [19] have motivated us to extend it for computing robust front for MaOP.

III. Robustness in Multi-objective Optimization (MOO) using NSGA-III

Diversity preservation, well-distribution, and good convergence are the desirable properties of non-dominated solutions obtained upon solving MaOP using EMO algorithms. In high-dimensional problems, maintaining diversity as well as obtaining a converged Pareto-optimal front is a challenging and difficult task. Since NSGA-II [10] has limitations in addressing the properties stated above in MaOPs, NSGA-III is used in this study. Moreover, unlike in NSGA-II, the diversity preservation among population members in NSGA-III is aided by supplying and adaptively updating a number of well-spread reference points on a (M-1)-dimensional unit hyper-plane, where M is the number of objective function [19]. Given the divisions or partitions (p) for an M-objective problem, the number of reference points (R_p) is computed as follows:

$$R_p = \binom{M+p-1}{p}. \tag{5}$$

Fig.2a depicts the concept of reference point located on a unit hyperplane. These well-distributed reference points produce reference lines or reference vectors. In each generation, non-dominated solutions that are closer to these reference vectors are used to create the offspring. This procedure allows the solutions to be well-distributed and spread in objective space. Also, NSGA-III was extended for constraint MaOPs by incorporating a constraint-handling strategy [20].

Fig. 2b presents non-dominated front obtained upon solving a five-objective MaOP known as the river pollution problem using NSGA-II algorithm [10] and Fig. 2c presents non-dominated front obtained using NSGA-III algorithm [20]. It can be inferred that the non-dominated solutions obtained from NSGA-III are better distributed and more diverse, as compared to NSGA-II solutions. This can be attributed to the use of a well-distributed set of reference points, presented in Fig. 2a, used to guide the progress of the NSGA-III approach.

Convergence, coverage, and success rate of robust solutions are the performance metrics of MORO/MaORO algorithms [13]. A modified selection scheme, association operator, and niche preserving operator in NSGA-III algorithm improve the convergence and coverage of MaOPs and hence is suitable for applying to MaORO problems. The mean effective function $\left(f^{\text{eff}}\right)$ is computed using H neighboring solutions in the δ-neighborhood of each reference point of NSGA-III. This mean effective function (f^{eff}) can be supplied to the robustness definition of Type I and Type II defined in (2) and (4), respectively.

In the next section, NSGA-III is implemented for solving two test problems and two real-world MaORO engineering problems. Also, the effect of robustness parameters (δ, H, η) on robust front is studied for MaORO problems. Details on these parameters and generating H solutions in δ-neighborhood are explained in [9].

IV. Simulation Results

This section presents the results of NSGA-III in solving MORO and MaORO problems.

A. 4-Bar Truss Problem with Two Objectives

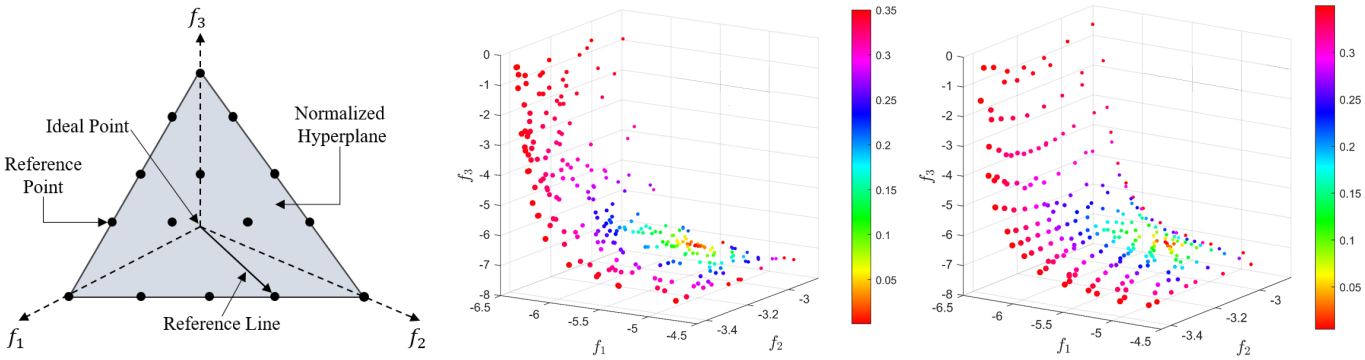
The 4-bar truss design problem is an unconstrained bi-objective problem that has four decision variables. The first objective is structural volume and the second objective is joint displacement. The details of the problem are discussed in [21].

Type I and Type II robust fronts are computed for different parameters (δ, H, η) using the NSGA-III procedure. For this problem, 201 reference directions are created using the Das-Dennis method [22] and the NSGA-III is executed for 200 generations with a tournament selection strategy. Other GA parameters used are binary crossover (η_c = 30, p_c = 1.0), mutation ($p_m = 1/n$, η_m = 20). The simulation is performed in *pymoo* [23].

1) Effect of δ on robust front: A simulation is performed on 4-bar truss problem with different values of perturbation extents (δ = [0.2, 0.3, 0.4]) in decision variables. For variables x_i, the perturbation $\delta_i = 2\delta$ ($i = 1, 2, 4$) and $\delta_3 = \delta$ is used. H = 50 solutions are generated in the neighborhood of a solution to compute the mean effective objective function values. The strategy for generating the neighboring solutions in the vicinity of a decision variable point is being adopted from [9], where Latin Hypercube Sampling (LHS) was performed.

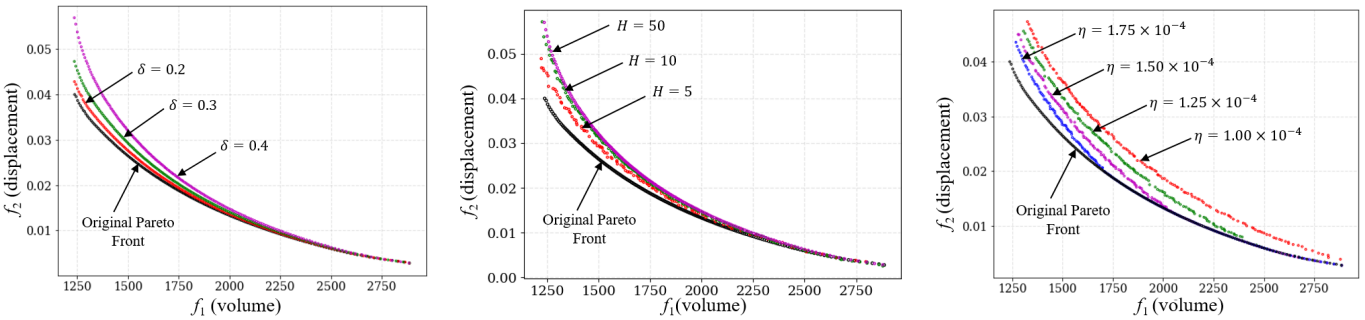
Fig. 3a represents the original non-dominated front along with the robust front (Type I) for different δ values while keeping H constant. It can be inferred from Fig. 3a that as the value of δ (domain of neighborhood) increases, the robust front shifts away from the original non-dominated front and moves inside the objective space. For 4-bar truss problem, a part of the robust front coincides with the original Pareto front. As the value of δ increases, the difference between the robust front and the original Pareto front becomes significant.

It is to be noted that the robust front obtained upon solving the 4-bar truss example using NSGA-III is well distributed. Again, the non-dominated solutions obtained from the original non-dominated front are pretty robust as it requires a large perturbation in the neighborhood of a solution in design space (δ = [0.2, 0.3, 0.4]) to obtain a robust front that is significantly different from the original non-dominated front. However, here we focused on demonstrating the working of NSGA-III as MORO algorithm in generating a well-distributed and diverse set of non-dominated solutions and the effect of robustness parameter δ on the robust front.



(a) Reference point-based NSGA-III (b) Non-dominated front (NSGA-II) (c) Non-dominated front (NSGA-III)

Fig. 2: (a) Reference points are highlighted on the normalized hyper-plane for a 3-objective problem with 4 divisions/partitions i.e. p = 4, (b) River Pollution Problem: non-dominated front obtained from NSGA-II, (c) River Pollution Problem: non-dominated front obtained from NSGA-III. f_1, f_2, and f_3 are represented on three axes, whereas the fourth objective (f_4) is represented by the marker size and the fifth objective (f_5) is represented by marker color provided on a color scale.



(a) Type I: Effect of δ (H = 50) (b) Type I: Effect of H (δ = 0.4) (c) Type II: Effect of η (δ = 0.4, H = 500)

Fig. 3: Scatter plot for 4-bar truss problem: (a) Minimum-f_2 solutions are naturally robust, whereas minimum-f_1 solutions are not. With increasing δ, the robust front shifts away from the original non-dominated front for minimum-f_1 region, (b) With increasing H, the robust front shifts away from the original non-dominated front as well, (c) With decreasing value of η (constraint value for Type-II robustness), the robust front shifts away from the original non-dominated front more pronouncing than previous two cases.

2) Effect of H on robust front: The number of neighboring solutions H generated in the δ-neighbourhood of a solution ($\mathbf{x}$) is studied and analyzed in this section. For this, simulation is performed for different values H in the fixed δ-neighborhood. Fig. 3b presents the original non-dominated front along with robust fronts (Type I) obtained for H = [5, 10, 50] values by keeping δ (= 0.4) constant. It can be inferred that as the value of H increases, the robust front shifts away from the original non-dominated front and moves inside the objective space.

It can also be observed that there is less difference between the robust front corresponding to H = 10 and H = 50 as compared to H = 5 and H = 10. Hence, at higher values of H, the robust front of Type I is converging to a specific non-dominated front. One interesting observation is that for lower values of H (5,10), the robust front is relatively not-so-well-converged as compared to the non-dominated front at higher values of H (50).

3) Effect of η on robust front: In this section, effect of parameter η on the robust front (Type II robustness) is studied by keeping other parameters δ and H constant. The simulation is performed for $\eta = [1.00 \times 10^{-4}, 1.25 \times 10^{-4}, 1.50 \times 10^{-4}, 1.75 \times 10^{-4}]$ with δ set at 0.4 and H at 500.

Fig. 3c presents the original non-dominated front along with the robust front of Type II at different values of η. Here the value η is a normalized change in the original objective function and mean effective function and is introduced as a constraint as per robustness *definition II*. It can be inferred from Fig. 3c that as the value of η increases i.e. the constraint is relaxed Type II robust solution obtained from NSGA-III moves closer to the original non-dominated front. It can be inferred from the plots in Fig. 3 that structural volume is more sensitive than displacement in robust fronts.

Also, a part of Type II robust solution set coincides with the original non-dominated set. As the value of η decreases, i.e. constraint is tightened, the robust front shifts away from the original non-dominated front and its coinciding part shortens. It is to be noted that the original non-dominated front is robust,

as it requires small values of η to generate robust solutions, which significantly differ from the original non-dominated solutions. The HV calculation is performed for the original and Type I and Type II robust fronts at different neighborhood parameters (δ, H, and η) values and is presented in Table I.

B. Four-Objective Test Problem

This test problem is created by extending the first three-objective test problem presented in [9] to have four objectives. Five variables are used in this problem. A simulation was performed to evaluate Type I and II robust fronts. Each simulation is performed for 200 generations using NSGA-III. The number of divisions to generate the reference points is used as p = 10 and other GA parameters are kept the same as in the previous example. Next, we discuss the effect of neighborhood parameters (δ, and η) on the resulting robust front. For this problem, we kept $\delta_i = \delta$ for i = 1, 2, 3; and δ_i = 2δ for $i > 3$. The value of parameters α and β used in the objective function are kept the same as discussed in [9].

1) Effect of δ on the Robust Front: The effect of δ is studied on the robust front by keeping H (= 50) constant. A set of values of δ = [0.005, 0.01, 0.02] is used to compute the Type I robust fronts. The left upper diagonal plots in Fig. 4 represents the scatter plot of the objective functions of original as well as robust fronts at different δ values. The marker color is used to differentiate the original and robust front. The blue color markers represent the original non-dominated front, orange color markers represent the Type I robust front corresponding to δ = 0.005. The green and the red color markers represent the Type I robust fronts corresponding to δ = 0.01 and δ = 0.02, respectively. It can be inferred from the f_1 vs. f_2 and f_1 vs. f_3 plots that as the value of δ increases the range and maximum values of objective functions f_1, f_2, and f_3 reduces in robust front (orange, green, and red markers). On the other hand, the plots f_1-f_3 vs. f_4 reveal that as the value of δ increases the objective f_4 values in robust front also increase. Also, with an increasing value of δ, the range as well as the maximum value of f_4 increases for the robust front. Hence, it can be concluded that with increasing the values of δ, the robust front shrinks in objective functions f_1, f_2, and f_3 space, whereas it shifts in the dominated region of the original non-dominated front in objective function f_4 space.

It is to be noted that increasing the δ values increases the region or range of the neighborhood at a design point. The mean effective objective function ($\mathbf{f}^{\text{eff}}$) computed at design point corresponding to higher values of δ becomes dominated in some part of the design space. This leads to shrinking down the robust front in f_1, f_2, and f_3 objective space and shifting it in f_4 objective function space towards the dominated region of the original non-dominated front. The left upper diagonal scatter plot in Fig. 4 reveals the effectiveness of NSGA-III in generating a well-distributed and diverse set of Type I robust front for MaOPs.

2) Effect of η on the Robust Front: Type II robustness is considered on the 4-objective test problem. A simulation is performed at different η = [0.4, 0.6, 1.0] values by keeping other parameters δ = 0.02 and H = 100 fixed. The right lower diagonal plots in Fig. 4 represents the scatter plot of the original non-dominated front along with the robust fronts obtained at a different set of η values. The marker colors in the scatter plot are utilized to differentiate the original and robust fronts. The blue color markers represent the original non-dominated front, orange color markers represent the Type II robust front corresponding to η = 0.4. The green and the red color markers represent the Type II robust fronts corresponding to η = 0.6 and η = 1.0, respectively.

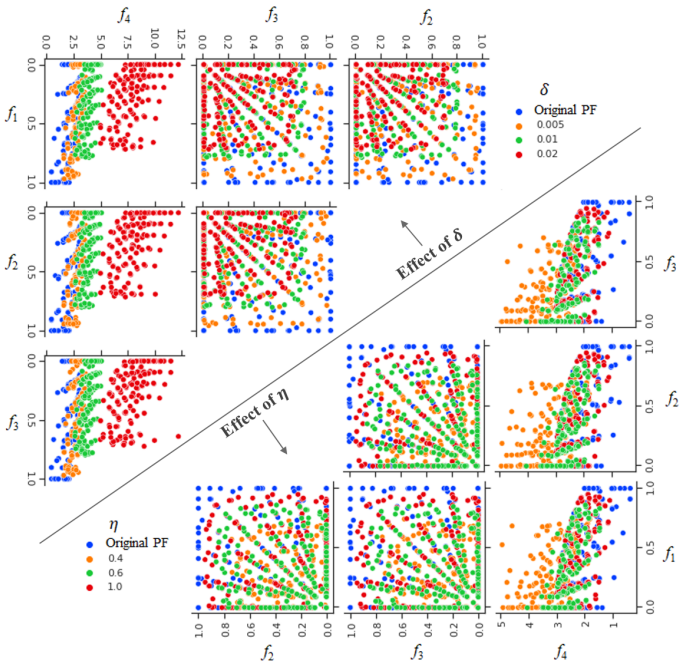


Fig. 4: Scatter plot for 4-objective test problem: (a) The left upper diagonal scatter plot represents the effect of parameter δ on robust front. With an increasing value of δ, the robust front shrinks in f_1, f_2, and f_3 objective space, whereas it shifts inside the dominated region of the original non-dominated front in f_4 objective space, (b) The right lower diagonal scatter plot represents the effect of parameter η on robust front. With decreasing the value of η, the robust front shrinks in f_1, f_2, and f_3 objective space, whereas it shifts inside dominated region of the original non-dominated front in f_4 objective space.

f_1 vs. f_2 and f_1 vs. f_3 scatter plots presented in the right lower diagonal plot of Fig. 4 reveal that as the value of η decreases, the range and maximum values of objective functions f_1, f_2, and f_3 also decreases in Type II robust fronts highlighted in orange, green, and red color markers. Again, the scatter plots f_1-f_3 vs. f_4 present that with decreasing values of η, the objective function f_4 shifts towards its higher values, and the range of the objective function f_4 also increases. This signifies that as the value of the parameter η decreases, the robust fronts shrink in f_1, f_2, and f_3 space, whereas it shifts in the dominated region of the original non-dominated front in f_4 objective function space.

TABLE I: Hyper Volume (HV) of robust fronts at the different values of neighboring parameters (δ, H, and η). It can be inferred from the HV values of Type I and Type II robust front that as the parameter δ and H increases the HV decreases, whereas on increasing the parameter η, HV of robust front increases.

4-bar truss problem	Effect of δ values (H = 50)			Effect of H values (δ = 0.40)			Effect of η values (δ=0.40, H = 500)			**Original non-dom. front**
	δ = 0.2	δ = 0.3	δ = 0.4	H = 5	H = 10	H = 50	η = 1.00e-4	η = 1.25e-4	η = 1.50e-4	
HV	71.27	69.97	67.69	70.02	68.51	67.69	64.10	66.77	68.89	**72.17**
4-obj. test prob.	Effect of δ values (H = 50)			Effect of H values (δ = 0.020)			Effect of η values (δ=0.020, H = 100)			**Original non-dom. front**
	δ = 0.005	δ = 0.010	δ = 0.020	H = 2	H = 10	H = 50	η = 0.4	η = 0.6	η = 1.0	
HV	14.60	13.41	9.36	14.91	13.54	13.43	14.43	14.87	15.03	**15.17**
River poll. problem	Effect of δ values (H = 50)			Effect of H values (δ = 0.035)			Effect of η values (δ=0.025, H = 100)			**Original non-dom. front**
	δ = 0.010	δ = 0.025	δ = 0.035	H = 2	H = 5	H = 100	η = 0.005	η = 0.010	η = 0.025	
HV	320.25	303.32	292.66	323.13	305.63	292.34	285.03	301.23	316.33	**331.06**
8-obj. test prob.	Effect of δ values (H = 100)			Effect of H values (δ = 0.05)			Effect of η values (δ=0.05, H = 100)			**Original non-dom. front**
	δ = 0.050	δ = 0.075	δ = 0.100	H = 5	H = 10	H = 100	η = 0.2	η = 0.3	η = 0.4	
HV	40.08	28.32	15.06	42.73	40.64	28.32	40.40	45.15	48.01	**53.18**

It is to be noted that the parameter η takes part as an inequality constraint in Type II robustness definition (4). By tightening the constraint, the robust front shrinks in f_1, f_2, and f_3 objective space, whereas it shifts in the dominated region of the original non-dominated front in f_4 space. The right lower diagonal part of Fig. 4 highlights the usefulness of NSGA-III in finding a well-distributed and diverse set of Type II robust solutions for many-objective test problems. The HV of the original and Type I and Type II robust fronts at different combinations of δ, H, and η values are presented in Table I.

C. River Pollution Problem

The river pollution problem [24] is an unconstrained multi-objective optimization problem containing 5 objectives and 2 design variables. Further details on this problem can be found in [25]. Type I and Type II robustness formulation is solved for the river pollution problem to evaluate robust solutions using NSGA-III algorithm. The river pollution problem is studied for various sets of parameter values (δ, H, η) and their effect on robust fronts. For this problem, 20 divisions are provided to compute the reference directions using the Das-Dennis method [22] and the GA is executed for 50 generations with a tournament selection strategy. Other GA parameters used are binary crossover (η_c = 30, p_c = 1.0) and mutation ($p_m = 1/n$, η_m = 20). From Fig. 2c, it can be inferred that objectives f_1 and f_3, and objectives f_2 and f_4 (marker size) are conflicting with each other. The objective f_5 (marker color) decreases slowly from the edges of non-dominated front and decreases relatively faster to attain a minimum value at a point that corresponds to relatively lower values of f_3 and f_4, and higher values of f_1 and f_2.

1) Effect of δ on the Robust Front: Type I robustness is solved for the river pollution problem with different δ values and keeping H (= 50) constant. The robust fronts at δ = [0.010, 0.025, 0.035] are evaluated using the NSGA-III algorithm. The left upper diagonal plots in Fig. 5 represent the scatter plot of the original non-dominated front and Type I robust fronts corresponding to δ = [0.010, 0.025, 0.035]. The marker colors in the scatter plot differentiate the original and robust fronts. The blue color markers in left upper diagonal plots in Fig. 5 represent the original non-dominated front, and orange color markers represent the Type I robust front corresponding to δ = 0.01. The green and the red color markers represent the Type I robust fronts corresponding to δ = 0.025 and δ = 0.035, respectively.

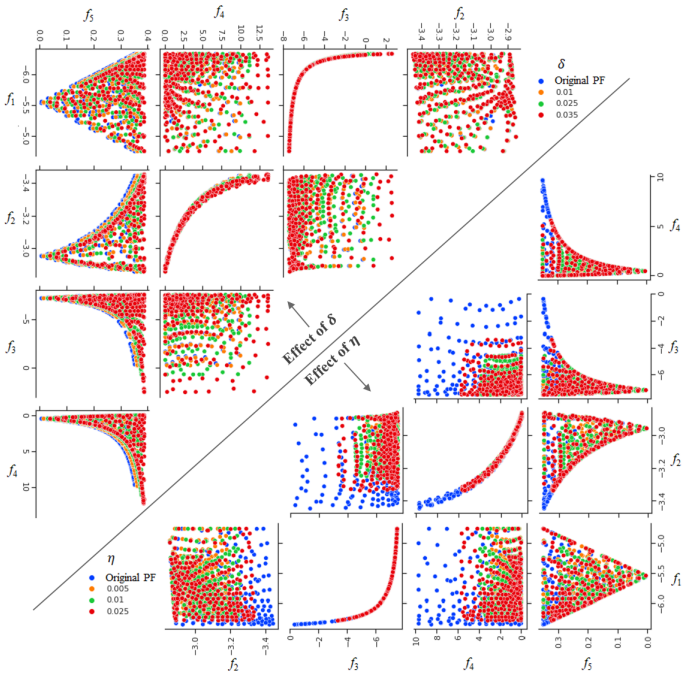


Fig. 5: Scatter plot for river pollution problem: (a) The left upper diagonal scatter plot represents the effect of parameter δ on robust front, (b) The right lower diagonal scatter plot represents the effect of parameter η on robust front.

From the f_1-f_4 vs. f_5 plot in the left upper diagonal part of Fig. 5 reveals that by increasing the value of parameter δ, the Type I robust front (orange, green, and red markers) shift towards the dominated region of the original non-dominated front (blue markers). Also, with increasing the value of δ, the maximum value and the range of objectives f_3 and f_4 increases, whereas the range of objective f_1 reduces.

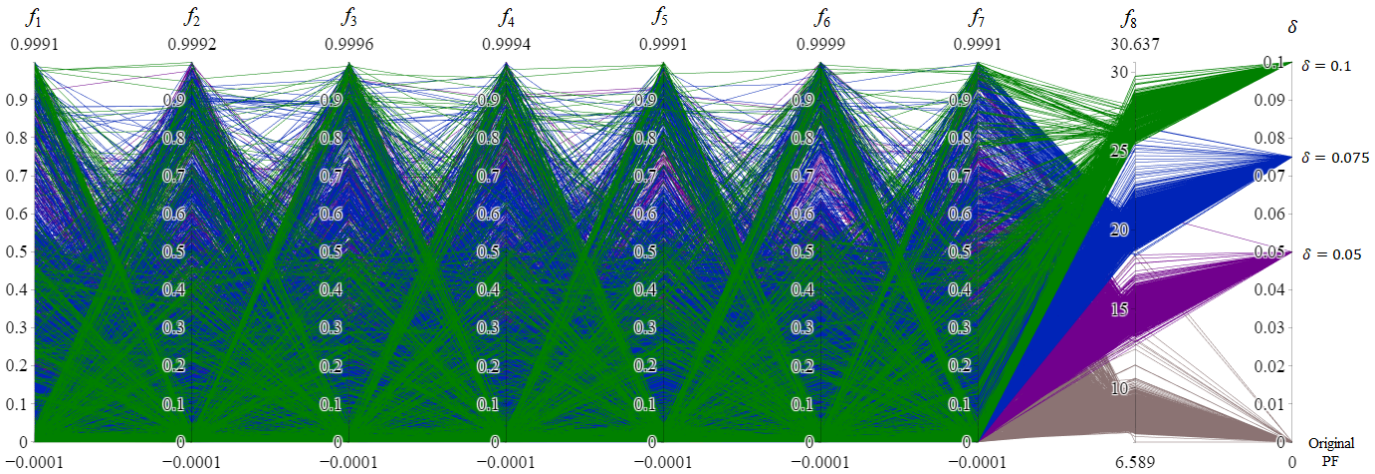


Fig. 6: Parallel coordinate plot (PCP) of 8-Objective Problem: The eight objective functions (f_1-f_8) are represented on separate parallel axis. An additional axis representing a set of δ values is used to differentiate the original and robust front. This plot highlights the effect of δ on robust front(s). As the value of parameter δ increases, the maximum and minimum value f_8 also increase, however, the other function values span from 0 to 1.

2) Effect of η on the Robust Front: In this section, Type II robustness definition (4) is applied to the river pollution problem to evaluate the robust fronts using NSGA-III algorithm. Type II robustness definition considers an inequality constraint of normalized difference of original solution to mean effective solution by introducing a parameter η. A simulation is performed with different sets of η values. Original non-dominated front and robust front corresponding to η = 0.025, 0.010, and 0.005 are evaluated using NSGA-III algorithm and presented in right lower diagonal scatter plots in Fig. 5. The blue color markers in the right lower diagonal plots in Fig. 5 represent the original non-dominated front, and orange color markers represent the Type II robust front corresponding to η = 0.005. The green and red color markers represent Type II robust fronts corresponding to η = 0.01 and η = 0.025, respectively.

In this simulation, the neighborhood size parameter (δ = 0.025) and the number of the neighboring points to compute the mean effective function (H = 100) are kept constant. It can be inferred from the right lower diagonal part of Fig. 5 that as the value of parameter η decreases the robust fronts highlighted in orange, green, and red color markers shrink and shorten which is evident from f_1 vs. f_2 and f_2 vs. f_3 plot. This effect of η on the robust front is justified, as decreasing η means tightening the constraint that consecutively reduces the feasible space. This forces some solutions that were optimal before to become infeasible, thereby shrinking the original non-dominated front to produce the robust front corresponding to the respective η value.

For river pollution problem, Type I and Type II robust fronts are evaluated using NSGA-III algorithm for different sets of neighboring parameters δ, H, and η, and their HV is calculated and compared with HV of original non-dominated front.

D. Eight-objective Test Problem

This test problem is obtained by scaling a 3-objective test problem 2 discussed in [9]. It has 8 objectives and 10 design variables. The number of partitions on the unit hyperplane is taken as p = 6, and NSGA-III algorithm is implemented for 200 generations. Other GA parameters are kept the same as in the previous examples. For performing the Type I and Type II robustness, the neighborhood size at different design variables are chosen as $\delta_i = \delta$, for i= 1,...,7; and $\delta_i = 2\delta$ for $i > 7$. A simulation is performed at different sets of δ (= [0.05, 0.075, 0.1]), H = [5, 10, 100], and η (= [0.2, 0.3, 0.4]) values. The HV calculation for original and robust front at different sets of δ, H, and η values is performed and reported in Table I.

The robust front corresponding to different δ values along with the actual non-dominated front is plotted in Fig. 6 using parallel coordinate plots (PCP). In the PCP, 8 objectives (f_1-f_8) are plotted at 8 parallel axes. An additional axis named δ highlights and differentiates the original and robust fronts according to their δ values. From Fig. 6, it can be observed that the objective function f_8 values vary at different ranges for different δ = [0.05, 0.075, 0.1] values. As the value of parameter δ increases, the maximum and minimum values of f_8 also increase. On the other hand, the other function (f_1-f_7) values vary in the range from 0 to 1.

The effectiveness of NSGA-III implemented to 2-objective to 8-objective problems for applying the Type I and Type II robustness definition highlights the usefulness in generating a well-distributed and diverse set of non-dominated solutions and robust solutions for MaOPs.

V. Effect of δ, H, and η on Hyper-Volume (HV)

This section studies the effect of robust fronts obtained at different parameter values (δ, H, and η) on HV [26]. Table I presents the HV values of original and robust fronts for the test problems and real-world problems discussed in the

result section. The reference point used for computing HV is computed as follows:

$$f_j^{\text{ref}} = f_j^{\max} + 0.1(f_j^{\max} - f_j^{\min});\ j = 1, \ldots, M,$$

where $f_j^{\max}$ and $f_j^{\min}$ are the maximum and minimum values of j^{th} objective functions among all the robust front and original non-dominated front, respectively. From Table I, it can be concluded that as the parameter δ and H increase the HV decreases. On the contrary, on increasing the parameter η, HV of the robust front increases. It is to be noted that for all four examples, the trend of HV with neighborhood parameters (δ, H, and η) is consistent. Increasing the parameter η brings the Type II robust front closer to the original non-dominated front that leads to a higher HV value as compared to HV value of the robust fronts corresponding to the lower η values.

VI. Conclusions

This paper has extended the idea of finding robust multi-objective Pareto-optimal solutions (termed here as 'robust solutions') into the context of many-objective optimization. Two types of robustness measures are used to modify the application of a specific evolutionary many-objective optimization (EMaO) algorithm – NSGA-III – to find the robust front for MaOPs. Robust NSGA-III procedure has then been applied to two benchmark and two real-world engineering problems. The effect of robustness parameters (δ, H, η) on the resulting robust front has been studied.

As the future extension of this work, the novel performance metrics for robust multi-optimization algorithms proposed in [13] can be computed for NSGA-III and can be compared with existing algorithms such as MOEA/D to evaluate its effectiveness for solving robust many-objective test problems. Also, the challenging robust multi-objective test problems proposed in [14] can be scaled to MaOPs and the effectiveness of NSGA-III can be studied on these test problems. As generating the robust front is computationally costly, it is desirable to develop approaches such as one discussed in [4] and [12] that have the ability to create robust solutions without significantly increasing the computational complexity of the EAs.

An efficient visualization method to visualize the solutions of EMaO-RO problems can be studied as future work [27]. Though the current work is implemented the Type I and II robustness definition, NSGA-III is not limited to these two types of definitions, and can be extended and applied to other robustness definitions used in MORO and MaORO studies. Moreover, Type I and Type II definitions of robustness can be extended for scalarization functions or utility functions that can further be extended to perform interactive and robust multi-criteria decision-making (MCDM) tasks [28].

References

[1] K. Deb, "Multi-objective optimisation using evolutionary algorithms: An introduction," in *Multi-objective evolutionary optimisation for product design and manufacturing*. Springer, 2011, pp. 3–34.

[2] C. A. C. Coello, G. B. Lamont, D. A. Van Veldhuizen *et al.*, *Evolutionary algorithms for solving multi-objective problems*. Springer, 2007, vol. 5.

[3] J. Branke , "Creating robust solutions by means of evolutionary algorithms," in *International Conference on Parallel Problem Solving from Nature*. Springer, 1998, pp. 119–128.

[4] J. Branke, "Efficient evolutionary algorithms for searching robust solutions," in *Evol. Design and Manufacture*. Springer, 2000, pp. 275–285.

[5] J. Branke and C. Schmidt, "Faster convergence by means of fitness estimation," *Soft Computing*, vol. 9, no. 1, pp. 13–20, 2005.

[6] Y. Jin and B. Sendhoff, "Trade-off between performance and robustness: An evolutionary multiobjective approach," in *International Conference on Evolutionary Multi-Criterion Optimization*. Springer, 2003, pp. 237–251.

[7] S. Tsutsui and A. Ghosh, "Genetic algorithms with a robust solution searching scheme," *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 3, pp. 201–208, 1997.

[8] E. J. Hughes, "Evolutionary multi-objective ranking with uncertainty and noise," in *International Conference on Evolutionary Multi-Criterion Optimization*. Springer, 2001, pp. 329–343.

[9] K. Deb and H. Gupta, "Searching for robust Pareto-optimal solutions in multi-objective optimization," in *International Conference on Evolutionary Multi-Criterion Optimization*. Springer, 2005, pp. 150–164.

[10] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.

[11] H. Gupta and K. Deb, "Handling constraints in robust multi-objective optimization," in *2005 IEEE Congress on Evolutionary Computation*, vol. 1. IEEE, 2005, pp. 25–32.

[12] Q. Zhou, P. Jiang, X. Huang, F. Zhang, and T. Zhou, "A multi-objective robust optimization approach based on Gaussian process model," *Structural and Multidisc. Optimization*, vol. 57, no. 1, pp. 213–233, 2018.

[13] S. Mirjalili and A. Lewis, "Novel performance metrics for robust multi-objective optimization algorithms," *Swarm and Evolutionary Computation*, vol. 21, pp. 1–23, 2015.

[14] S. Mirjalili and A. Lewis , "Hindrances for robust multi-objective test problems," *Applied Soft Computing*, vol. 35, pp. 333–348, 2015.

[15] S. Mirjalili and A. Lewis, "Novel frameworks for creating robust multi-objective benchmark problems," *Information Sciences*, vol. 300, pp. 158–192, 2015.

[16] Y. Chi, Y. Xu, and R. Zhang, "Many-objective robust optimization for dynamic var planning to enhance voltage stability of a wind-energy power system," *IEEE Transactions on Power Delivery*, vol. 36, no. 1, pp. 30–42, 2020.

[17] F. Jalalvand, H. H. Turan, S. Elsawah, and M. J. Ryan, "A multi-objective risk-averse workforce planning under uncertainty," in *2020 IEEE Symposium Series on Computational Intelligence (SSCI)*. IEEE, 2020, pp. 1626–1633.

[18] S. Pourshahabi, G. Rakhshandehroo, N. Talebbeydokhti, M. R. Nikoo, and F. Masoumi, "Handling uncertainty in optimal design of reservoir water quality monitoring systems," *Environmental Pollution*, vol. 266, pp. 115–211, 2020.

[19] K. Deb and H. Jain, "An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: solving problems with box constraints," *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 4, pp. 577–601, 2014.

[20] H. Jain and K. Deb, "An evolutionary many-objective optimization algorithm using reference-point based nondominated sorting approach, part II: Handling constraints and extending to an adaptive approach," *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 4, pp. 602–622, 2014.

[21] F. Cheng and X. Li, "Generalized center method for multiobjective engineering optimization," *Engg. Opt.*, vol. 31, no. 5, pp. 641–661, 1999.

[22] I. Das and J. E. Dennis, "Normal-boundary intersection: A new method for generating the Pareto surface in nonlinear multicriteria optimization problems," *SIAM J. on Optimization*, vol. 8, no. 3, pp. 631–657, 1998.

[23] J. Blank and K. Deb, "pymoo: multi-objective optimization in python," *IEEE Access*, vol. 8, pp. 89 497–89 509, 2020.

[24] K. Miettinen and M. M. Mäkelä, "Interactive method NIMBUS for nondifferentiable multiobjective optimization problems," in *Multicriteria Analysis*. Springer, 1997, pp. 310–319.

[25] S. C. Narula and H. Weistroffer, "A flexible method for nonlinear multicriteria decision-making problems," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 19, no. 4, pp. 883–887, 1989.

[26] C. M. Fonseca, L. Paquete, and M. López-Ibánez, "An improved dimension-sweep algorithm for the hypervolume indicator," in *IEEE Conference on Evolutionary Computation*. IEEE, 2006, pp. 1157–1163.

[27] D. Yadav, D. Nagar, P. Ramu, and K. Deb, "Visualization-aided multi-criteria decision-making using interpretable self-organizing maps," *European Journal of Operational Research*, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0377221723001145

[28] D. Yadav, P. Ramu, and K. Deb, "Visualization-aided multi-criterion decision-making using reference direction based Pareto race," in *2022 IEEE Symposium Series on Computational Intelligence (SSCI)*, 2022, pp. 125–132.