

# Machine Learning-Aided Constraint Handling Under Variable Uncertainty for Preference-Based EMO

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

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

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

COIN Report 2025004

## ABSTRACT

Evolutionary Multi-objective Optimization (EMO) algorithms are widely used to solve real-world multi-objective optimization problems, aiming to obtain a set of non-dominated solutions close to the Pareto front. However, most EMO methods assume deterministic decision variables, ignoring inherent uncertainties in engineering applications, which can lead to design failures, especially in reliability-based designs. Reliability-based Multi-objective Optimization (ReMOO) addresses this issue by incorporating variable uncertainty and probabilistic constraints to generate a *Reliable Front*. ReMOO operates using a bi-level framework: the outer level optimizes objective functions, while the inner level estimates reliability through computationally intensive methods, like Monte Carlo Simulation (MCS) or the Performance Measure Approach (PMA). Additionally, decision-makers (DMs) often select only a subset of reliable solutions, limiting computational efficiency. To overcome these challenges, this paper proposes a Machine Learning-assisted reliability-based Multi-Criteria Decision-Making (ML-ReMCDM) technique. ML models are trained on reliability-based constraints within the decision space before an EMO execution. In the inner loop, ML models predict probabilistic constraints and reliability indices, significantly reducing computational costs. Moreover, the outer loop computes only the DM-preferred segment of the reliable front, further enhancing efficiency. The ML-ReMCDM approach, implemented on several benchmark and real-world examples, demonstrates substantial improvements in computational efficiency as well as practical applicability.

## KEYWORDS

Evolutionary Algorithms, Multi-Criteria Decision-Making, Machine Learning, Reference Point, Reliability, Uncertainty.

### ACM Reference Format:

Deepanshu Yadav, Palaniappan Ramu, and Kalyanmoy Deb. . Machine Learning-Aided Constraint Handling Under Variable Uncertainty for Preference-Based EMO

## 1 INTRODUCTION

Real-world applications often involve multiple conflicting objectives [4], which are typically formulated as multi-objective optimization (MOO) problems in the literature [7, 34]. Solving an MOO

problem does not yield a single unique solution but instead results in a set of non-dominated (ND) solutions located near the Pareto-optimal front (PF) [26]. The PF represents the set of optimal solutions in the objective space, where improving one objective is only possible by compromising at least one other objective [8]. Evolutionary Multi-Objective Optimization (EMO) algorithms, such as NSGA-II [12], NSGA-III [16], RVEA [5], C-TAEA [21], and MOEA/D [50], are designed to efficiently compute a set of ND solutions approximating the PF in a single computational effort. However, in practice, a decision maker (DM) must select one or a few solutions from the PF that align with their preferences, discarding the rest [2]. Given the computational expense of obtaining the complete PF, Multi-Criteria Decision-Making (MCDM) techniques have been developed to integrate the DM's preferences, enabling the identification of a portion of the PF that meets the DM's aspiration levels. These MCDM algorithms include R-NSGA-II [13], RD-NSGA-II [10], R-NSGA-III [36], *g*-NSGA-III [28], light beam search [11, 17], and various other scalarization-based methods [27].

In practical applications, variable uncertainty is unavoidable in design and optimization processes [30]. These uncertainties arise from various sources, such as process parameters, manufacturing tolerances, instrument precision, and measurement errors [19]. As a result, it is essential to account for variable uncertainty in multi- and many-objective problems (M/MaOPs). Singh and Branke [33] proposed evaluating non-dominated solutions considering stochastic decision variables. For constrained M/MaOPs, optimal solutions often lie on the boundaries of constraints [14, 32]. In such cases, uncertainty in design variables can lead to infeasible solutions, potentially causing design failures. Therefore, constraint handling under variable uncertainty requires a distinct approach compared to standard MOO formulations. The goal is to provide reliable solutions where the probability of design failure due to constraint violations is minimized. Reliability-Based Design Optimization (RBDO) methods [1, 47] have been developed to address these challenges. Among these, double-loop approaches (DLA) are widely used. In a DLA, the outer loop identifies optimal design variables by solving the regular optimization problem, while the inner loop calculates the reliability index [9] for a design solution to assess its reliability against constraint violations or failures.

Reliability estimation methods can be broadly classified into sampling-based and optimization-based approaches. Sampling-based methods are detailed in [19, 31, 44], while optimization-based approaches include first-order reliability methods (FORM), such as the Reliability Index Approach (RIA) [6] and the Performance Measure Approach (PMA) [48], which compute the Most Probable Target Point (MPTP). Advanced methods like the Advanced Mean Value

(AMV) method [37], Conjugate Mean Value (CMV) method [49], and Hybrid Mean Value (HMV) method [18] were introduced to improve MPTP computation. The limitations of these approaches have been addressed by Chaos Control theory-based method [45], the Modified Chaos Control (MCC) method [23], and the Adaptive Chaos Control (ACC) method [20]. Relevant literature on reliability-based design optimization can be found in [9, 24, 25].

It is worth noting that computing the complete Reliable Front (ReF) is computationally expensive [9]. To address this, we propose using Reference Point-Based Evolutionary Multi-Objective Optimization (R-EMO) algorithms to compute a subset of the ReF, as defined by the optimization formulation in Equation 2. This proposed approach, termed Reliability-Based Multi-Criteria Decision-Making (ReMCDM), employs a Double Loop Architecture (DLA) or a bi-level approach. In this framework, the outer loop solves the optimization problem in Equation 1 using R-NSGA-III, while the inner loop uses the Hybrid Mean Value (HMV) method to handle constraints from Equation 2 under variable uncertainty and perform reliability estimation. This approach eliminates the need to compute the complete PF or ReF. Onward, we refer this proposed approach as Bi-ReMCDM.

To further reduce the computational effort required for reliability estimation in the inner loop of the Bi-ReMCDM approach, we propose the Machine Learning-Assisted Reliability-Based Multi-Criteria Decision-Making (ML-ReMCDM) method. First, a sample is generated using the Latin Hypercube Sampling (LHS) within the design variable space, and the corresponding reliability-based constraints are evaluated using the HMV method. Subsequently, for each constraint, an ML model [29] is trained using LHS samples as inputs and the reliability-based constraint as output. We compare the computational costs of Bi-ReMCDM and ML-ReMCDM approaches and propose metrics to evaluate their performance. The key contributions of this paper are outlined as follows:

- (1) Proposing a bi-level approach for Reliability-Based Multi-Criteria Decision-Making (Bi-ReMCDM) approach.
- (2) Highlighting the limitations of the Bi-ReMCDM approach and proposing Machine Learning-Assisted Reliability-Based Multi-Criteria Decision-Making (ML-ReMCDM) approach.
- (3) Proposing metrics for performance evaluation of the Bi-ReMCDM and ML-ReMCDM approaches.

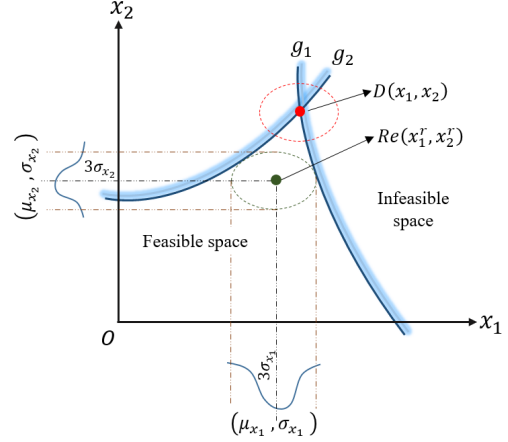
The remainder of the paper is organized as follows. Section 2 presents the constraint handling under variable uncertainty. Reliability-based Multi-Criteria Decision-Making approach is discussed in Section 3. The proposed Machine Learning-assisted Reliability-based Multi-Criteria Decision-Making (ML-ReMCDM) approach is outlined in Section 4. Simulation results are presented in Section 5 followed by the conclusions in Section 6.

## 2 CONSTRAINT HANDLING UNDER VARIABLE UNCERTAINTY

Consider the following multi-/many-objective optimization problem (M/MaOP) formulation [9] presented in Equation 1:

$$\begin{aligned} & \min_{(\mathbf{x}, \mathbf{d})} \left[ f_1(\mathbf{x}, \mathbf{d}, \mathbf{p}), f_2(\mathbf{x}, \mathbf{d}, \mathbf{p}), \dots, f_M(\mathbf{x}, \mathbf{d}, \mathbf{p}) \right], \\ & \text{subject to: } g_j(\mathbf{x}, \mathbf{d}, \mathbf{p}) \leq 0, \quad j = 1, \dots, J, \\ & \mathbf{d}^{(lb)} \leq \mathbf{d} \leq \mathbf{d}^{(ub)}; \mathbf{x}^{(lb)} \leq \mathbf{x} \leq \mathbf{x}^{(ub)}. \end{aligned} \quad (1)$$

Here,  $M$  represents the number of objective functions and  $J$  represents the number of inequality constraints (equality constraints are omitted for simplicity). In this formulation,  $\mathbf{x}$  is the vector of uncertain design variables,  $\mathbf{d}$  is the vector of deterministic design variables unaffected by uncertainty, and  $\mathbf{p}$  represents uncertain parameters that are not variables in the optimization problem. Figure 1 illustrates a deterministic optimal solution  $D$ , obtained at the intersection of two constraints,  $g_1$  and  $g_2$ . However, under variable uncertainties  $\sigma_{x_1}$  and  $\sigma_{x_2}$ , the optimal solution might shift within the red ellipse.



**Figure 1: Reliability-based design optimization (RBDO) task. Constraints under variable uncertainty makes a solution risky for constraint violation.  $x_1$  and  $x_2$  are two design variables.  $\sigma_{x_1}$  and  $\sigma_{x_2}$  are uncertainty associated with them.  $g_1$ , and  $g_2$  are two constraints.  $D$  is deterministic solution and  $Re$  marks a reliable solution.**

Many solutions within this ellipse violate the constraints, making them infeasible. This creates the possibility of design failure due to variable uncertainty. Thus, handling constraints under variable uncertainty requires a probabilistic multi-objective optimization (MOO) formulation to ensure that the probability of design failure does not exceed a pre-specified threshold ( $p_f$ ), which is derived from a desired reliability index  $\beta$ . The probabilistic formulation for M/MaOPs is presented in Equation 2:

$$\begin{aligned} & \min_{(\mu_{\mathbf{x}}, \mathbf{d})} \left[ f_1(\mu_{\mathbf{x}}, \mathbf{d}, \mu_{\mathbf{p}}), f_2(\mu_{\mathbf{x}}, \mathbf{d}, \mu_{\mathbf{p}}), \dots, f_M(\mu_{\mathbf{x}}, \mathbf{d}, \mu_{\mathbf{p}}) \right], \\ & \text{subject to: } \Pr(g_j(\mu_{\mathbf{x}}, \mathbf{d}, \mu_{\mathbf{p}}) > 0) \leq p_f, \quad j = 1, \dots, J, \\ & \mathbf{d}^{(lb)} \leq \mathbf{d} \leq \mathbf{d}^{(ub)}; \mathbf{x}^{(lb)} \leq \mu_{\mathbf{x}} \leq \mathbf{x}^{(ub)}. \end{aligned} \quad (2)$$

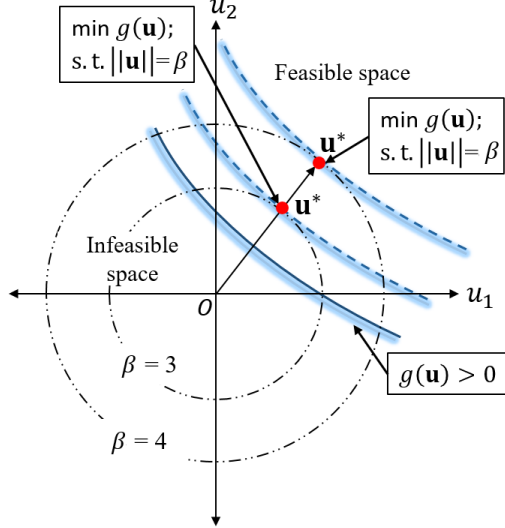
The solution to this formulation consists of the mean value of the uncertain design variables ( $\mu_{\mathbf{x}}$ ) and the deterministic design variables ( $\mathbf{d}$ ). A deterministic constraint in Equation 1 is replaced with a probabilistic constraint in Equation 2. Here,  $\Pr(\cdot)$  denotes the probability of constraint violation (or failure). This formulation provides a reliable solution ( $Re$  in Figure 1) within the feasible space, ensuring that the probability of failure does not exceed the threshold ( $p_f$ ).

The reliability index  $\beta$  is defined as  $-\Phi^{-1}(p_f)$ , where  $\Phi$  is the cumulative standard Normal distribution function. It can be approximated using Monte Carlo Simulation (MCS) in the vicinity

of the solution, but this method is computationally expensive [47]. Alternative methods for estimating  $\beta$  with reduced computational cost include adaptive sampling techniques [31], surrogate models, and extreme-value statistics [19, 44] based techniques. Another approach focuses on optimization-based reliability measures, identifying the “most probable target point” (MPTP) on the constraint boundary closest to the deterministic solution [9, 24]. This method linearly approximates constraints near the MPTP to estimate reliability, forming the basis of first-order reliability methods (FORMs). Examples of FORMs include the Performance Measure Approach (PMA) and Reliability Index Approach (RIA). The current work utilizes the PMA to compute the MPTP for  $\beta$  estimation.

### 3 RELIABILITY-BASED MULTI-CRITERIA DECISION-MAKING APPROACH

The proposed Reliability-based Multi-Criteria Decision-Making (ReMCDM) framework uses a bi-level approach to perform reliability-based MCDM tasks. We refer this bi-level approach as Bi-ReMCDM. In the outer loop, the original optimization problem in Equation 2 is solved using  $(\mathbf{x}, \mathbf{d})$  as design variables. For each  $(\mathbf{x}, \mathbf{d})$  in the outer loop, a *chance* constraint (involving  $\Pr(\cdot)$  in Equation 2) is evaluated in the inner loop using another optimization formulation i.e. PMA or RIA. Instead of solving an M/MaOP to compute the ReF in the outer loop, the proposed Bi-ReMCDM method uses preference-based evolutionary multi-objective optimization (R-EMO) algorithms to conduct the MCDM task.



**Figure 2: Concept of Performance Measure Approach (PMA) for RBDO.**  $g$  is a specific constraint and  $\beta$  is target reliability index.  $u_1$  and  $u_2$  are the variables in transformed standard normal space.  $\mathbf{u}^*$  (in red dots) is MPTP.

#### 3.1 Performance Measure Approach (PMA)

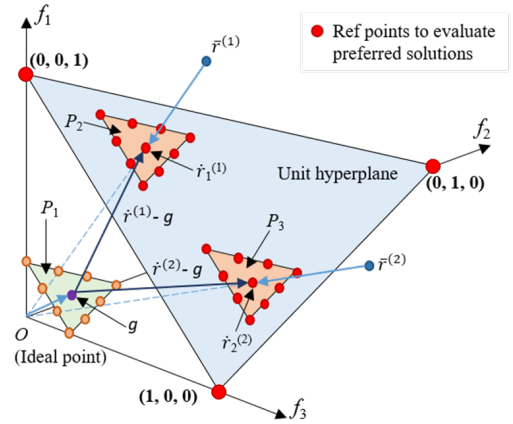
In the Performance Measure Approach (PMA), at each design point  $(\mathbf{x}, \mathbf{d})$ , the uncertain design variables and parameters  $(\mathbf{v} = (\mathbf{x}, \mathbf{p}))$

are transformed into standard Normal space  $(\mathbf{u})$ , resulting in the following optimization formulation:

$$\begin{aligned} \min_{\mathbf{u}} \quad & g_j(\mathbf{d}, \mathbf{u}), \quad j = 1, \dots, J, \\ \text{subject to:} \quad & \|\mathbf{u}\|_2 = \beta, \end{aligned} \quad (3)$$

where  $u_k = \frac{v_k - \mu_{v_k}}{\sigma_{v_k}}$  for  $k = 1, 2, \dots, K$ , with  $K$  being the dimension of the vector  $\mathbf{v}$ , and  $\beta$  representing the reliability index. Solving the optimization problem in Equation 3 yields the Most Probable Target Point (MPTP) solution  $(\mathbf{u}^*)$  at point  $\mathbf{d}$ .

The solution  $\mathbf{u}^*$  is then transformed back into the original space  $\mathbf{v}^* = (\mathbf{x}^*, \mathbf{p}^*)$  using  $v_k = \mu_{v_k} + u_k \sigma_{v_k}$ . This transformed point is then used in the constraint function of Equation 1. The updated constraint  $g_j(\mathbf{v}^*)$  replaces the probabilistic constraint in Equation 2, thereby avoiding computationally expensive simulations, such as Monte Carlo Simulation (MCS). However, finding MPTP is a challenging task [37]. In current paper, we use HMV approach for MPTP computation. For details on hybrid mean value (HMV) calculation procedure, readers are referred to [46] and Appendix A.1.



**Figure 3: A sketch of the R-NSGA-III's reference point computation procedure [41].**  $\bar{r}^{(1)}$  and  $\bar{r}^{(2)}$  are two reference points provided by DM.  $P_1$  is the shrunk hyperplane depending on parameter  $\mu$ .  $P_2$  and  $P_3$  are reference planes on unit hyperplane corresponding to the DM preferences  $\bar{r}^{(1)}$  and  $\bar{r}^{(2)}$ .  $\hat{r}^{(1)}$  and  $\hat{r}^{(2)}$  are the intercept of unit hyperplane and normalized reference point vectors  $\bar{r}^{(1)}$  and  $\bar{r}^{(2)}$ . Extreme reference points are also included for finding extreme PF solutions along with preferred solutions.

#### 3.2 Preference-based Evolutionary Multi-objective Optimization

Using PMA for reliability estimation in the optimization formulation defined in Equation 2, the Reliable Front (ReF) can be obtained. However, computing the complete ReF is computationally expensive. The Bi-ReMCDM approach utilizes an R-EMO algorithm [41] to integrate the decision maker's (DM's) objective preferences, allowing the assessment of only a subset of the ReF. In the current paper, R-NSGA-III [36] is used as the R-EMO algorithm.

The procedure for generating reference points in R-NSGA-III is illustrated in Figure 3. It requires user-defined reference points ( $\bar{r}^{(k)}$ ), population per reference point (pop/RP), and a shrinkage factor ( $\mu$ ) as input. These inputs are used to generate reference points (red dots) on a unit hyperplane, where  $\mu$  controls the spread of the Multi-Criteria Decision-Making (MCDM) solutions. Additionally,  $M$  extreme points on the unit simplex are provided, enabling R-NSGA-III to focus on extreme Pareto-optimal (PO) solutions for normalization of objective function. The extreme solutions help identify the ideal point ( $O$ ), which is used to translate the shrunk reference points from  $O$  toward  $\bar{r}^{(k)}$  on the unit simplex. These translated reference points guide R-NSGA-III to prioritize solutions near them, thereby advancing toward a preferred subset of the reliable front over successive generations.

#### 4 PROPOSED MACHINE LEARNING-ASSISTED RELIABILITY-BASED MCDM APPROACH

The proposed Bi-ReMCDM approach discussed in Section 3 performs MCDM in outer level using R-NSGA-III and estimates reliability using HMV method in the inner loop. In each generation of R-NSGA-III, at each design point, HMV method is executed to estimate the MPTP for a given targeted reliability index ( $\beta_t$ ) value. Although Bi-ReMCDM only evaluates preferred reliable solutions instead of evaluating the complete ReF, the computational cost of reliability estimation in the inner level is still high. To address this limitation, we propose the ML-ReMCDM approach presented in Figure 4.

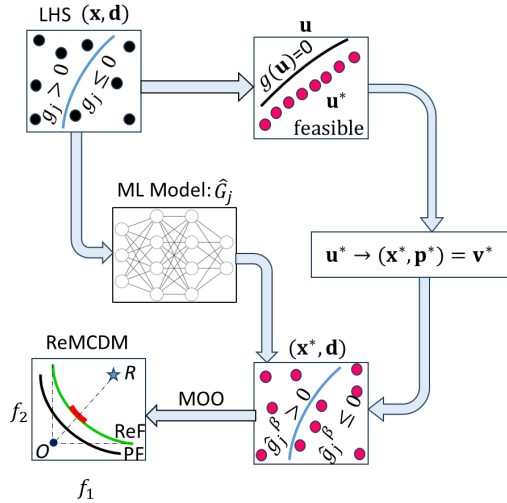


Figure 4: Outline of ML-ReMCDM approach

##### 4.1 ML-ReMCDM Approach

The outline of the proposed ML-MCDM approach is presented in Figure 4. It starts with performing Latin Hypercube Sampling (LHS) [15] in variable  $(\mathbf{x}, \mathbf{d})$  space. Then the uncertain parameters and the uncertain variables ( $\mathbf{v} = (\mathbf{p}, \mathbf{x})$ ) are transformed into standard Normal space  $(\mathbf{u})$ . Next, for each constraint function ( $g_j$ ), MPTP ( $\mathbf{u}^*$ ) values are computed on the LHS samples of  $(\mathbf{x}, \mathbf{d})$ , using the HMV method.

The MPTP ( $\mathbf{u}^*$ ) are converted back into the uncertain variables ( $\mathbf{v}^* = (\mathbf{x}^*, \mathbf{p}^*)$ ) and reliability-based constraint ( $g_j^\beta$ ) is computed at each  $\mathbf{v}^*$  values. Next, a Machine Learning (ML) model ( $\hat{G}_j$ ) using Multi-Layer Perceptron (MLP) [29] is trained considering  $(\mathbf{x}, \mathbf{d})$  as input and  $g_j^\beta$  as output for each constraint  $j, j = 1, \dots, J$ . Finally, the original constraint in MOO formulation presented in Equation 2 is substituted by reliability-based constraints' ML model(s)  $\hat{G}_j$ . This eliminates the constraint evaluation in the inner loop of Bi-ReMCDM approach reducing the computational cost of reliability estimation. The stepwise procedure of ML-ReMCDM approach is provided as follows:

- Step 1:** Generate LHS samples [15] of  $(\mathbf{x}, \mathbf{d})$ , where  $\mathbf{x}$  represents design variables and  $\mathbf{d}$  represents deterministic parameters.
- Step 2:** Extract uncertain parameters, denoted as  $\mathbf{p}$ .
- Step 3:** Transform  $\mathbf{v} = (\mathbf{x}, \mathbf{p})$  into  $\mathbf{u}$ -space.
- Step 4:** For each design point  $(\mathbf{x}, \mathbf{d})$ , compute the MPTP  $\mathbf{u}^*$ .
- Step 5:** Transform  $\mathbf{u}^*$  back to the original space, obtaining  $\mathbf{v}^* = (\mathbf{x}^*, \mathbf{p}^*)$ .
- Step 6:** Evaluate the reliability-based constraint as:  $g_j^\beta = g_j(\mathbf{v}^*)$
- Step 7:** Construct Machine Learning (ML) model ( $\hat{G}_j$ ) such that:  $g_j^\beta = \hat{G}_j(\mathbf{x}, \mathbf{d})$ . First, build a Regression Model using Multi-layer Perceptron (RegMLP) for  $g_j^\beta \in \mathbb{R}$ . If the accuracy of RegMLP is less than 0.95, convert  $g_j^\beta \in [0, 1]$  based on feasible and infeasible region and build a classification model using Multi-layer Perceptron (ClassMLP). If the accuracy of both the models are less than 0.95, add more points in LHS samples and rebuild the ML models to further improve the accuracy of ML models.
- Step 8:** Substitute the original constraints ( $g_j$ ) with the reliability-based constraint's ML model ( $\hat{G}_j$ ) in the multi-objective optimization (MOO) formulation given in Equation 2 to perform ML-ReMCDM.

##### 4.2 Comparison and Performance Evaluation

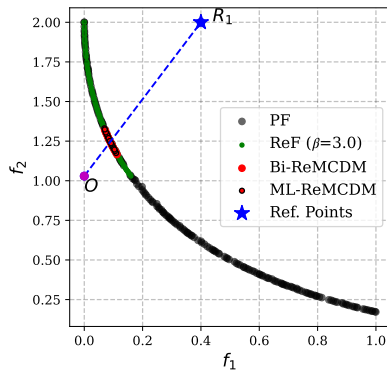
The comparison of Bi-ReMCDM approach and ML-ReMCDM approach is conducted on the basis of their performance evaluation and computational cost. The performance of Bi-ReMCDM and ML-ReMCDM approach is evaluated based on (i) number of reliable solutions found per reference direction in R-NSGA-III, and (ii) convergence and diversity of reliable solutions in terms of their R-HV values [22]. The computational cost of Bi-ReMCDM and ML-ReMCDM is compared on the basis of evaluation of objective and constraint functions. Further details are provided in Appendix A.2.

## 5 SIMULATION RESULTS

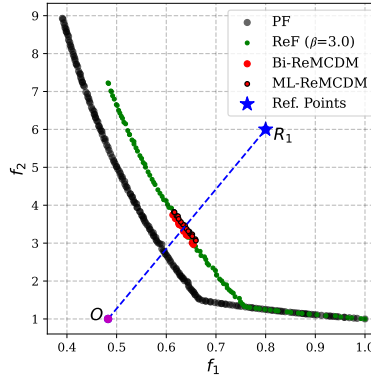
The proposed Bi-ReMECM approach and ML-ReMCDM approach is implemented on four benchmark problems and six real-world engineering examples. A detailed description of these examples is presented in Table 1. For generating the Pareto Front (PF) and Reliable Front (ReF), NSGA-II algorithm is used for bi-objective cases and NSGA-III algorithm is used for three- and five-objective cases. Details of the population size and the number of generations used in these algorithms for each example are available in

**Table 1: Problem description for ReMCDM.**  $N$ ,  $M$ , and  $G$  denote number of variables, objectives, and constraints, respectively.  $R_P$  is the reference point(s) used to compute ReMCDM solutions.  $\sigma$  indicates the variable uncertainty. # LHS indicates number of LHS sample used to build ML-model for reliability constraint.  $\eta_{min}$  indicates the accuracy of least accurate ML models among all the ML models for reliability constraints. Type indicates the type of ML model, where R stands for Regression-based model and C stands for Classifier-based model. Further details on ML hyper-parameter are provided in Appendix.

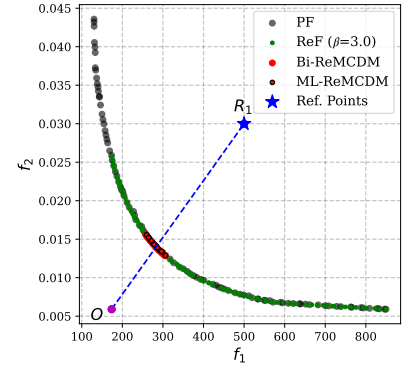
Examples	# Var. (N)	# Obj. (M)	# Cons. (G)	Reference Point(s) ( $R_P$ )	Variable Uncertainty ( $\sigma$ )	# LHS Sample	Accuracy ( $\eta_{min}$ )	Type (R/C)
Benchmark 1	2	2	1	[0.4, 1.8]	[0.2, 0.2]	60	0.99	R
Benchmark 2	2	2	2	[0.8, 6.0]	[0.03, 0.03]	200	1.0	C
Benchmark 3	2	3	3	[3.4,0.2,0.5]	[0.05,0.05]	60	0.96	R
Benchmark 4	2	3	3	[2.0,1.0,0.7]	[0.035,0.035]	60	0.98	R
I-Beam	4	2	1	[500.0, 0.03]	[0.2, 0.2, 0.2, 0.2]	400	0.99	R
Disc Brake	4	2	4	[4.0, 2.25]	[0.25, 0.25, 0.25, 0.25]	800	0.96	C
Welded-Beam	4	2	4	[15.0, 0.006]	[1.0, 1.0, 5.0, 1.0]	400	0.95	C
Two-Bar Truss	3	2	3	[0.09,6000.0]	[0.001, 0.001, 0.25]	300	0.98	C
Car Side-Impact	7	3	10	[40.0, 4.0, 60.0]	[0.1,0.1,0.1,0.1,0.1,0.1,0.1]	700	0.99	R
WATER	3	5	7	$z^{nad}=[8.267e4, 1.349e3, 2.853e6, 1.597e7, 3.747e4]$	[0.002,0.002,0.002]	1000	0.98	C



(a) Benchmark Example 1



(b) Benchmark Example 2



(c) I-Beam Design

**Figure 5: Machine Learning-assisted reliability-based Multi-Criteria Decision-Making (ML-ReMCDM) approach on (a) Benchmark Example 1, (b) Benchmark Example 2, and (c) I-Beam Design Example. Black and green dots highlights PF and ReF, respectively. Blue star indicates reference point. Preferred reliable solutions obtained from Bi-ReMCDM approach and ML-ReMCDM approach is highlighted in red dots and red dots with black boundary, respectively.  $O$  indicates ideal point obtained from R-NSGA-III's extreme solutions while executing ML-ReMCDM approach.**

the Appendix. It is important to note that the Bi-ReMCDM and ML-ReMCDM approaches do not require the information on PF and ReF. Here, PF and ReF are computed solely for reference purposes. The Bi-ReMCDM and ML-ReMCDM solutions are obtained by executing the R-NSGA-III algorithm. The details on R-NSGA-III parameters is provided in Table 2. The GA parameters i.e. simulated binary crossover: SBX (cross-over probability  $p_c=1.0$  and crossover operator  $\eta_c=30$ ) and polynomial mutation: PM (mutation probability  $\eta_m=30$ ) are kept the same for all examples. The HMV method is implemented with a maximum iterations ( $T=100$ ) and a convergence parameter ( $\epsilon = 10^{-4}$ ) as the termination criteria, whichever is reached first. In all the examples, a reliability index value of  $\beta=3.0$  is used. While training the ML models, ReLU activation function and Adam optimizer are used in MLP Regressor and

Classifier. Additional details on ML hyper-parameters are provided in Appendix. The computational cost and performance evaluation of Bi-ReMCDM and ML-ReMCDM is outlined in Table 2.

## 5.1 Benchmark Examples

The proposed Bi-ReMCDM and ML-ReMCDM approaches are applied to four benchmark problems. The analytical expressions for these benchmark problems are available in Appendix A.3, while a detailed description of the problems is presented in Table 1.

**5.1.1 Benchmark Example 1.** Benchmark example 1 is a constrained bi-objective problem with two variables and a single constraint. The variable uncertainty information is provided in Table 1. The results for Benchmark Example 1 is presented in Figure 5a. The black and green dots present PF and ReF, respectively, which is

**Table 2: Performance evaluation of Bi-ReMCDM and ML-ReMCDM approach.**  $\mu$  and pop/RP are the shrinkage factor and population size per reference point in R-NSGA-III algorithm.  $gen_{Bi}$  and  $gen_{ML}$  are the number of R-NSGA-III generations in Bi-ReMCDM and ML-ReMCDM approach.  $\bar{k}_j$  is average number of iterations in HMV method for a given constraint  $g_j$ . The objective function and constraint function evaluations are compared as constraint cost and objective cost of Bi-ReMCDM and ML-ReMCDM approach. The objective performance and constraint performance is evaluated according to section 6. It can be observed that the constraint cost for ML-ReMCDM approach is always better than Bi-ReMCDM approach. However, in few instances objective cost for ML-ReMCDM approach is worse than Bi-ReMCDM approach.

Examples	R-NSGA-III parameters				$\Sigma \bar{k}_j$	Obj. Cost		Cons. Cost		Obj. Perf.		Cons. Perf.	
	$\mu$	pop/RP	$gen_{Bi}$	$gen_{ML}$		$E_{Bi}^{obj}$	$E_{ML}^{obj}$	$E_{Bi}^{cons}$	$E_{ML}^{cons}$	$P_{Bi}^{obj}$	$P_{ML}^{obj}$	$p_{Bi}^{cons}$	$p_{ML}^{cons}$
Benchmark 1	0.25	20	100	23	2	4,000	920	20,000	660	0.288	0.272	1	1
Benchmark 2	0.15	20	100	209	4	4,000	8,360	40,000	4,400	0.547	0.544	1	1
Benchmark 3	0.15	21	100	40	6	6,300	2,520	63,000	1,980	0.238	0.238	1	1
Benchmark 4	0.15	21	100	26	6	6,300	1,638	63,000	1,980	0.103	0.103	1	1
I-Beam Design	0.20	20	100	30	5	4,000	1,200	90,000	18,400	4.089	4.033	1	1
Disc Brake Design	0.15	20	100	157	10	4,000	6,280	1,80,000	75,200	0.808	0.776	1	1
Welded-Beam Design	0.20	20	100	390	25	4,000	15,600	4,50,000	91,600	0.027	0.026	1	0.65
Two-Bar Truss	0.15	10	100	32	24	4,000	1,280	1,68,000	51,300	44.04	43.47	1	1
Car Side Impact	0.10	21	200	92	80	12,600	5,796	50,40,000	8,47,000	3.014	2.876	1	1
WATER Example	0.15	35	100	99	27	17,500	17,325	6,61,500	1,96,000	0.411	0.394	0.31	0.31

shown for reference only. It is evident from the figure that the part of PF, with objective  $f_2 < 1$ , does not satisfy the reliability constraint corresponding to  $\beta = 3.0$ , which does not provide the preferred reliable solutions for  $f_2 < 1$ . For this example, ReF is part of PF. The effect of  $\beta$  on ReF for Benchmark Example 1 is provided in Appendix A.4. A single reference point  $R_1 = [0.4, 2.0]$  is used in R-NSGA-III with population size per reference point as 20 and shrinkage factor ( $\mu = 0.25$ ) to compute preferred reliable solutions using Bi-ReMCDM approach and ML-ReMCDM approach. The Bi-ReMCDM solutions are represented in red dots. The ML-ReMCDM solutions are represented by red dots with black boundary. Both Bi-ReMCDM and ML-ReMCDM approaches evaluate the preferred reliable solution corresponding to the supplied reference point  $R_1$ . However, Table 2 suggests that the computational cost in terms of objective and constraint function evaluation is significantly lower for ML-ReMCDM approach than ReMCDM approach. Constraint performance, in terms of generating reliable solutions per number of reference directions in R-NSGA-III, is the same for both the Bi-ReMCDM and ML-ReMCDM approaches. The convergence and diversity of the reliable solution obtained using Bi-ReMCDM and ML-ReMCDM approaches are indicated in terms of their R-HV values. It can be observed from Table 2 that R-HV values of preferred reliable solution obtained from both the approaches are almost same, which can also be observed from Figure 5a. The further details on ML hyper-parameters are provided in Appendix and details on R-NSGA-III parameters are provided in Table 2.

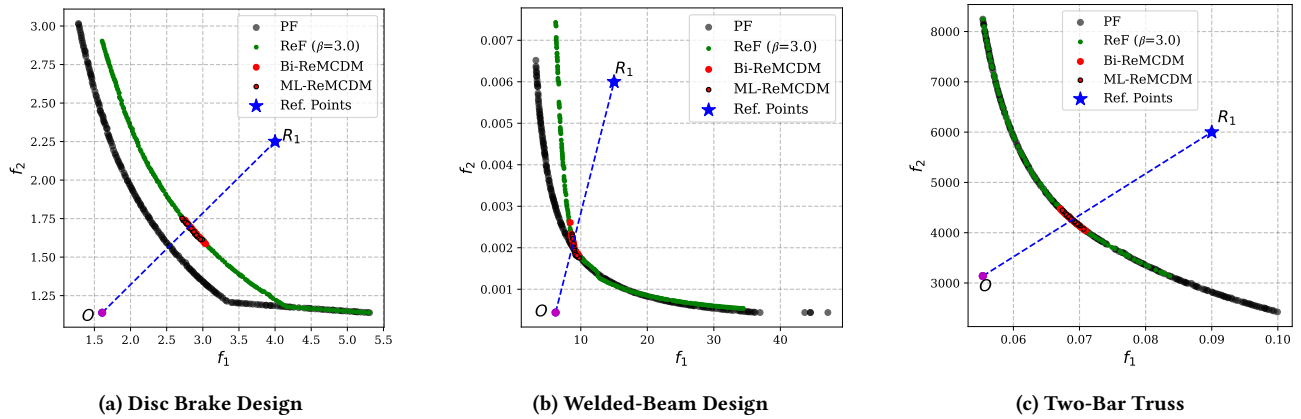
**5.1.2 Benchmark Example 2.** This is a constraint bi-objective example with two variables and two constraints. Problem description in terms of reference points, variable uncertainty, number of samples used in Latin Hypercube Sampling (LHS) to train ML model and its accuracy for Benchmark Example 2 is provided in Table 1. Figure 5b presents the results for Benchmark Example 2 obtained using Bi-ReMCDM and ML-ReMCDM approaches. The black and green dots represent PF and ReF ( $\beta = 3.0$ ), respectively. For higher values of  $f_1$

( $>0.75$ ), ReF and PF coincides with each other. However, for approximately  $f_1 < 0.75$ , ReF shifts away from the PF and moves inside the criterion space to satisfy the reliability constraint. With a reference point  $R_1 = [0.8, 6.0]$ , Bi-ReMCDM and ML-ReMCDM approaches are executed to evaluate preferred reliable solutions, presented in red dots and red dots with black boundary, respectively. Figure 5b indicates that the preferred reliable solutions obtained from both the Bi-ReMCDM and ML-ReMCDM approaches are very close. Table 2 presents the computational cost of Bi-ReMCDM and ML-ReMCDM approaches in terms of objective and constraint functions evaluation. The constraint evaluation is significantly lower, whereas the objective function evaluation is relatively higher for ML-ReMCDM approach compared to Bi-ReMCDM approach. This is because the ML-ReMCDM approach requires a relatively higher number of generations when executing the R-NSGA-III algorithm to achieve the same performance level as the Bi-ReMCDM approach. Constraint performance and objective performance as defined in Section 6 are outlined in Table 2, which suggest that both the approaches have similar performances. Details on Benchmark Example 3 and Benchmark Example 4 are provided in Appendix A.5.

## 5.2 Real-World Engineering Examples

The proposed ML-ReMCDM approach is implemented on four bi-objective, one three-objective and one five-objective examples. Analytical expression for Real-world Engineering examples are provided in [35]. The description in these examples are provided in Table 1. Additional details on R-NSGA-III parameters used in Bi-ReMCDM and ML-ReMCDM approaches along with the computational cost and performance evaluation is presented in Table 2.

**5.2.1 I-Beam Design.** This is a bi-objective constrained MOO problem. It has four design variables and a single constraint. Black and green dots in Figure 5c indicate the PF and ReF ( $\beta = 3.0$ ). There exist no reliable solutions for  $f_2 > 0.026$  as the reliability constraint ( $\beta \geq 3.0$ ) is not satisfied for that region. A reference point  $R_1 =$



**Figure 6: ML-ReMCDM approach on (a) Disc Brake Design, (b) Welded-Beam Design, and (c) Two-Bar Truss Example. Black and green dots highlights PF and ReF, respectively. Blue star indicates reference point. Reliable solutions obtained from Bi-ReMCDM and ML-ReMCDM approach is highlighted in red dots and red dots with black boundary, respectively.**

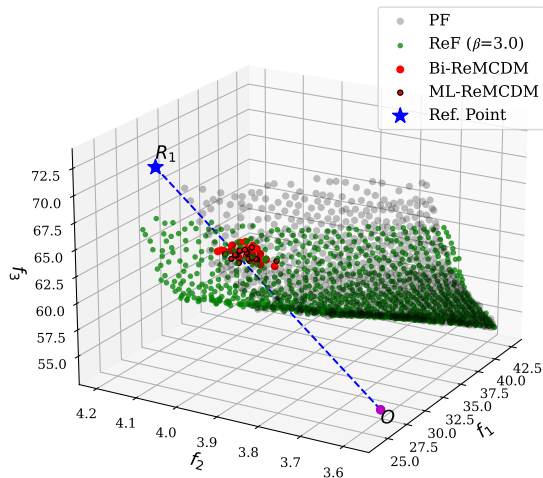
[500, 0.03] is supplied to R-NSGA-III algorithm for computing the preferred reliable solutions using Bi-ReMCDM and ML-ReMCDM approaches. The Bi-ReMCDM and ML-ReMCDM obtained preferred reliable solutions are highlighted in red dots and red dots with black boundary, respectively. The computational cost and performance of objective and constraint functions for I-Beam example is outlined in Table 2. The computational cost for ML-ReMCDM approach is significantly lower than Bi-ReMCDM approach.

**5.2.2 Disc Brake Design.** This example is a bi-objective constrained MOO problem with four variables and four constraints. Further details on the Disc Brake Design example are provided in Table 1. Figure 6a depicts the PF and ReF, highlighted in red and green dots, respectively. It can be observed from Figure 6a that for approximately  $f_1 > 4.0$ , the ReF coincides with the PF, whereas for  $f_1 < 4.0$ , the ReF shifts away from the PF and moves inside the criterion space to satisfy the reliability-based constraints. A reference point  $R_1 = [4.0, 2.25]$  is supplied to the R-NSGA-III algorithm to compute the preferred reliable solutions using the Bi-ReMCDM and ML-ReMCDM approaches. In Figure 6a, the preferred reliable solutions obtained from Bi-ReMCDM and ML-ReMCDM are highlighted using red dots and red dots with a black boundary, respectively. Figure 6a highlights that the spread of Bi-ReMCDM solution is slightly better than ML-ReMCDM solutions which is also reflected in terms of objective performance provided in Table 2. Table 2 also indicates that the constraint function evaluation is significantly less costly, while the objective function evaluation is relatively more expensive for the ML-ReMCDM approach compared to the Bi-ReMCDM approach. The high computational cost of objective function evaluation in the ML-ReMCDM approach arises from the increased number of generations required for R-NSGA-III to achieve performance comparable to the Bi-ReMCDM approach.

**5.2.3 Welded Beam Design.** The problem description for Welded-beam design example is provided in Table 1. Figure 6b presents the PF and ReF in black and green dots, respectively. Some part of the ReF and PF coincides with each other. For approximately  $f_2 > 0.02$ , the ReF shifts away from the PF and moves inside the criterion

space to satisfy the reliability constraint. A reference point  $R_1 = [15.0, 0.006]$  is supplied to R-NSGA-III algorithm to compute the preferred reliable solutions using Bi-ReMCDM approach and ML-ReMCDM approach highlighted in red dots and red dots with black boundary, respectively, in figure 6b. The computational cost and performance are calculated according to the definitions provided in Section 6 and presented in Table 2. It is observed from the Table 2 that constraint evaluation is less expensive and objective evaluation is more expensive for ML-ReMCDM approach compared to DLA-based Bi-ReMCDM approach. The constrained performance as defined in Section 6 for ML-ReMCDM approach is worse than Bi-ReMCDM approach because number of reliable solutions obtained using ML-ReMCDM approach is less than the number of reference direction supplied to R-NSGA-III algorithm. The objective performance for ReMCDM and ML-ReMCDM is similar. Further details in R-NSGA-III parameters, ML models' hyper-parameters is provided in Appendix.

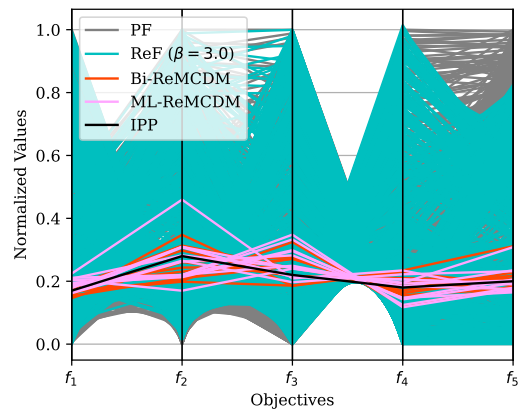
**5.2.4 Two-Bar Truss.** The problem description for Two-Bar Truss example is provided in Table 1, and the reliable solutions obtained using ReMCDM and ML-ReMCDM approaches by executing R-NSGA-III algorithm are presented in Figure 6c. The PF and ReF are highlighted using black and green dots, respectively. For this example, the ReF is part of the PF, as the reliability constraints are not satisfied for values of  $f_2 < 3000$ . A reference point,  $R_1 = [0.09, 6000]$ , is supplied to the R-NSGA-III algorithm to compute preferred reliable solutions using the ReMCDM and ML-ReMCDM approaches. The reliable solutions obtained from the ReMCDM and ML-ReMCDM approaches are highlighted as red dots and red dots with a black boundary, respectively. Table 2 indicates that the objective and constraint function evaluations for the ML-ReMCDM approach are better than those of Bi-ReMCDM approach, with the same constraint performance. That is, both the Bi-ReMCDM and ML-ReMCDM approaches generated the same number of reliable solutions per reference direction in R-NSGA-III. The spread of reliable solutions obtained using ML-ReMCDM approach is slightly smaller than that of Bi-ReMCDM approach. Therefore, the R-HV value, i.e., the objective performance of ML-ReMCDM approach, is slightly worse than that of Bi-ReMCDM approach.



**Figure 7: Car Side-Impact: Scatter plot indicates Pareto front (black dots) and Reliable front (green dots). Bi-ReMCDM solutions and ML-ReMCDM solutions are indicated by red dots and red dots with black boundary, respectively.**

**5.2.5 Car Side-Impact.** The problem details of the Car Side-Impact example are provided in Table 1. The results of the Car Side-Impact example are presented in Figure 7. The PF and ReF are highlighted using black and green dots, respectively. A single reference point,  $R_1 = [40.0, 4.0, 60.0]$ , is provided by the user to execute the R-NSGA-III algorithm and obtain preferred reliable solutions using the ReMCDM and ML-ReMCDM approaches. The reliable solutions obtained by the ReMCDM and ML-ReMCDM approaches are depicted as red dots and red dots with a black boundary, respectively. The spread of ML-ReMCDM solutions is smaller than that of ReMCDM solutions, resulting in lower R-HV values for ML-ReMCDM. Table 2 supports this inference in terms of objective function performance. As defined in Section 6, the constraint performance for both approaches is the same.

**5.2.6 WATER Example.** This five-objective constrained problem consists of three design variables and seven constraints. Figure 8 presents the PF, ReF, and preferred reliable solutions using the Parallel Coordinate Plot (PCP). All solutions are normalized according to the range of PF solutions before being highlighted on the PCP. The gray and cyan lines in Figure 8 represent the PF and ReF, respectively. It can be observed from the figure that ReF is a subset of PF, and no reliable solutions are obtained for higher values of the  $f_5$  objective function. Preferred reliable solutions obtained using the ReMCDM and ML-ReMCDM approaches are highlighted in red and pink, respectively. A black line represents the intersection of the reference point  $R_1$  and the reliable front, referred to as the Ideal Preferred Point (IPP). Table 2 indicates that the constraint function evaluation for ML-ReMCDM is significantly better than that of the ReMCDM approach. However, the objective function evaluation for ML-ReMCDM is only slightly better than that of the ReMCDM approach. This suggests that the number of generations required by R-NSGA-III to achieve the performance level of ML-ReMCDM, comparable to that of the ReMCDM approach, is almost the same as the number of generations used by R-NSGA-III in the ReMCDM



**Figure 8: WATER Example: PCP indicates Pareto front (gray lines) and reliable front (cyan lines). Bi-ReMCDM solutions and ML-ReMCDM solutions are highlighted in red and pink lines, respectively.**

approach. The constraint performance, in terms of the number of preferred reliable solutions per reference direction of the R-NSGA-III algorithm, is the same for both the ReMCDM and ML-ReMCDM approaches, as outlined in Table 2.

## 6 CONCLUSIONS

In this paper, we have provided the outline for Reliability-based Multi-Criteria Decision-Making (ReMCDM) using bi-level approach to solve  $M/MaOPs$  under variable uncertainty given objective preferences of DMs. To address this limitation of a standard ReMCDM approach, we have proposed Machine Learning-assisted Reliability-based Multi-Criteria Decision-Making (ML-ReMCDM) in which the ML models serve as reliability-based constraint eliminating the need of actual expensive constraint evaluation in inner loop of ReMCDM approach. The results obtained from implementing both the approaches on several benchmark and real-world engineering examples reveal that the proposed ML-ReMCDM approach is computationally efficient in handling the constraint under the variable uncertainty, for the same level of performance.

In future research, performance evaluation of R-EMO algorithms, such as R-NSGA-III, R-NSGA-II, C-TAEA, and RVEA, for the Bi-ReMCDM and ML-ReMCDM tasks can be conducted using the updated  $\hat{R}$ -HV metric proposed in a recent study [42]. Visualization is crucial for assisting decision-makers (DMs) in expressing preferences and selecting final solutions. Integrating the ReMCDM framework with suitable visualization techniques can enable interactive decision-making on the reliable front [38, 40]. While the proposed approach focuses on constraint handling under variable uncertainty, integrating objective function robustness [39, 41] can lead to a combined robustness and reliability-based decision-making framework.

## REFERENCES

- [1] Younes Aoues and Alaa Chateaneuf. 2010. Benchmark study of numerical methods for reliability-based design optimization. *Structural and multidisciplinary optimization* 41, 2 (2010), 277–294.

- [2] Kalyan Shankar Bhattacharjee, Hemant Kumar Singh, Michael Ryan, and Tapabrata Ray. 2017. Bridging the gap: Many-objective optimization and informed decision-making. *IEEE Transactions on Evolutionary Computation* 21, 5 (2017), 813–820.
- [3] Julian Blank and Kalyanmoy Deb. 2020. Pymoo: Multi-objective optimization in python. *IEEE access* 8 (2020), 89497–89509.
- [4] Vira Chankong and Yacov Y Haimes. 2008. *Multiobjective decision making: theory and methodology*. Courier Dover Publications.
- [5] Ran Cheng, Yaochu Jin, Markus Olhofer, and Bernhard Sendhoff. 2016. A reference vector guided evolutionary algorithm for many-objective optimization. *IEEE Transactions on Evolutionary Computation* 20, 5 (2016), 773–791.
- [6] Tae Min Cho and Byung Chai Lee. 2010. Reliability-based design optimization using convex approximations and sequential optimization and reliability assessment method. *Journal of mechanical science and technology* 24 (2010), 279–283.
- [7] Carlos A Coello Coello, Gary B Lamont, David A Van Veldhuizen, et al. 2007. *Evolutionary algorithms for solving multi-objective problems*. Vol. 5. Springer.
- [8] Kalyanmoy Deb. 2011. Multi-objective optimisation using evolutionary algorithms: an introduction. In *Multi-objective evolutionary optimisation for product design and manufacturing*. Springer, 3–34.
- [9] Kalyanmoy Deb, Shubham Gupta, David Daum, Jürgen Branke, Abhishek Kumar Mall, and Dhanesh Padmanabhan. 2009. Reliability-based optimization using evolutionary algorithms. *IEEE transactions on evolutionary computation* 13, 5 (2009), 1054–1074.
- [10] Kalyanmoy Deb and Abhishek Kumar. 2007. Interactive evolutionary multi-objective optimization and decision-making using reference direction method. In *Proceedings of the 9th annual conference on Genetic and evolutionary computation*. 781–788.
- [11] Kalyanmoy Deb and Abhay Kumar. 2007. Light beam search based multi-objective optimization using evolutionary algorithms. In *2007 IEEE Congress on Evolutionary Computation*. IEEE, 2125–2132.
- [12] Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and Tanaka Meyarivan. 2002. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE transactions on evolutionary computation* 6, 2 (2002), 182–197.
- [13] Kalyanmoy Deb and J Sundar. 2006. Reference point based multi-objective optimization using evolutionary algorithms. In *Proceedings of the 8th annual conference on Genetic and evolutionary computation*. 635–642.
- [14] Ahsanul Habib, Hemant Kumar Singh, and Tapabrata Ray. 2016. A study on the effectiveness of constraint handling schemes within Efficient Global Optimization framework. In *2016 IEEE Symposium Series on Computational Intelligence (SSCI)*. IEEE, 1–8.
- [15] Ronald L Iman. 2008. Latin hypercube sampling. In *John Wiley & Sons, Ltd*.
- [16] Himanshu Jain and Kalyanmoy Deb. 2013. 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* 18, 4 (2013), 602–622.
- [17] Andrzej Jaszkiewicz and Roman Słowiński. 1999. The ‘Light Beam Search’ approach—an overview of methodology applications. *European Journal of Operational Research* 113, 2 (1999), 300–314.
- [18] Behrooz Keshtegar and Peng Hao. 2017. A hybrid self-adjusted mean value method for reliability-based design optimization using sufficient descent condition. *Applied Mathematical Modelling* 41 (2017), 257–270.
- [19] Ikjin Lee, Ungki Lee, Palaniappan Ramu, Deepanshu Yadav, Gamze Bayrak, and Erdem Acar. 2022. Small failure probability: principles, progress and perspectives. *Structural and Multidisciplinary Optimization* 65, 11 (2022), 326.
- [20] Gang Li, Zeng Meng, and Hao Hu. 2015. An adaptive hybrid approach for reliability-based design optimization. *Structural and Multidisciplinary Optimization* 51 (2015), 1051–1065.
- [21] Ke Li, Renzhi Chen, Guangtao Fu, and Xin Yao. 2018. Two-archive evolutionary algorithm for constrained multiobjective optimization. *IEEE Transactions on Evolutionary Computation* 23, 2 (2018), 303–315.
- [22] Ke Li, Kalyanmoy Deb, and Xin Yao. 2017. R-metric: Evaluating the performance of preference-based evolutionary multiobjective optimization using reference points. *IEEE Transactions on Evolutionary Computation* 22, 6 (2017), 821–835.
- [23] Zeng Meng, Gang Li, Bo Ping Wang, and Peng Hao. 2015. A hybrid chaos control approach of the performance measure functions for reliability-based design optimization. *Computers & Structures* 146 (2015), 32–43.
- [24] Zeng Meng, Gang Li, Xuan Wang, Sadiq M Sait, and Ali Rıza Yıldız. 2021. A comparative study of metaheuristic algorithms for reliability-based design optimization problems. *Archives of Computational Methods in Engineering* 28 (2021), 1853–1869.
- [25] Zeng Meng, Betül Sultan Yıldız, Gang Li, Changting Zhong, Seyedali Mirjalili, and Ali Rıza Yıldız. 2023. Application of state-of-the-art multiobjective metaheuristic algorithms in reliability-based design optimization: a comparative study. *Structural and Multidisciplinary Optimization* 66, 8 (2023), 191.
- [26] Kaisa Miettinen. 1998. Nonlinear multiobjective optimization. In *International series in operations research and management science*.
- [27] Kaisa Miettinen and Marko M Mäkelä. 2002. On scalarizing functions in multiobjective optimization. *OR spectrum* 24 (2002), 193–213.
- [28] Julián Molina, Luis V Santana, Alfredo G Hernández-Díaz, Carlos A Coello Coello, and Rafael Caballero. 2009. g-dominance: Reference point based dominance for multiobjective metaheuristics. *European Journal of Operational Research* 197, 2 (2009), 685–692.
- [29] Fionn Murtagh. 1991. Multilayer perceptrons for classification and regression. *Neurocomputing* 2, 5-6 (1991), 183–197.
- [30] Frank Neumann, Aneta Neumann, and Hemant Kumar Singh. 2024. Evolutionary computation for stochastic problems. In *Proceedings of the Genetic and Evolutionary Computation Conference Companion*. 1352–1368.
- [31] Kiran Pannierselvam, Deepanshu Yadav, and Palaniappan Ramu. 2022. Scarce Sample-Based Reliability Estimation and Optimization Using Importance Sampling. *Mathematical and Computational Applications* 27, 6 (2022), 99.
- [32] Tapabrata Ray, Hemant Kumar Singh, Amitay Isaacs, and Warren Smith. 2009. Infeasibility driven evolutionary algorithm for constrained optimization. *Constraint handling in evolutionary optimization* (2009), 145–165.
- [33] Hemant Kumar Singh and Juergen Branke. 2022. Identifying stochastically non-dominated solutions using evolutionary computation. In *International Conference on Parallel Problem Solving from Nature*. Springer, 193–206.
- [34] Ralph E Steuer. 1986. Multiple criteria optimization: Theory, computation and applications. Wiley (1986).
- [35] Ryoji Tanabe and Hisao Ishibuchi. 2020. An easy-to-use real-world multi-objective optimization problem suite. *Applied Soft Computing* 89 (2020), 106078.
- [36] Yash Vesikar, Kalyanmoy Deb, and Julian Blank. 2018. Reference point based NSGA-III for preferred solutions. In *2018 IEEE symposium series on computational intelligence (SSCI)*. IEEE, 1587–1594.
- [37] Y-T Wu, HR Millwater, and TA Cruse. 1990. Advanced probabilistic structural analysis method for implicit performance functions. *AIAA journal* 28, 9 (1990), 1663–1669.
- [38] Deepanshu Yadav, Deepak Nagar, Palaniappan Ramu, and Kalyanmoy Deb. 2023. Visualization-aided multi-criteria decision-making using interpretable self-organizing maps. *European Journal of Operational Research* 309, 3 (2023), 1183–1200.
- [39] Deepanshu Yadav, Palaniappan Ramu, and Kalyanmoy Deb. 2023. Finding Robust Solutions for Many-Objective Optimization Using NSGA-III. In *2023 IEEE Congress on Evolutionary Computation (CEC)*. IEEE, 1–8.
- [40] Deepanshu Yadav, Palaniappan Ramu, and Kalyanmoy Deb. 2023. Interpretable self-organizing map assisted interactive multi-criteria decision-making following Pareto-Race. *Applied Soft Computing* 149 (2023), 111032.
- [41] Deepanshu Yadav, Palaniappan Ramu, and Kalyanmoy Deb. 2023. Multi-objective Robust Optimization and Decision-Making Using Evolutionary Algorithms. In *Proceedings of the Genetic and Evolutionary Computation Conference*. 786–794.
- [42] Deepanshu Yadav, Palaniappan Ramu, and Kalyanmoy Deb. 2024. An Updated Performance Metric for Preference-Based Evolutionary Multi-Objective Optimization Algorithms. In *Proceedings of the Genetic and Evolutionary Computation Conference*. 612–620.
- [43] Deepanshu Yadav, Palaniappan Ramu, and Kalyanmoy Deb. 2025. Reliability-Based MCDM Using Objective Preferences Under Variable Uncertainty. In *International Conference on Evolutionary Multi-Criterion Optimization*. Springer, 225–240.
- [44] Deepanshu Yadav, Kannan Sekar, and Palaniappan Ramu. 2024. Adaptive sampling based estimation of small probability of failure using interpretable Self-Organising Map. *Structural Safety* 109 (2024), 102470.
- [45] Dixiong Yang, Gang Li, and Gengdong Cheng. 2006. Convergence analysis of first order reliability method using chaos theory. *Computers & structures* 84, 8-9 (2006), 563–571.
- [46] Byeng D Youn, KK Choi, and Liu Du. 2005. Adaptive probability analysis using an enhanced hybrid mean value method. *Structural and Multidisciplinary Optimization* 29 (2005), 134–148.
- [47] Byeng D Youn and Kyung K Choi. 2004. Selecting probabilistic approaches for reliability-based design optimization. *AIAA journal* 42, 1 (2004), 124–131.
- [48] Byeng D Youn, Kyung K Choi, and Liu Du. 2005. Enriched Performance Measure Approach for Reliability-Based Design Optimization. *AIAA journal* 43, 4 (2005), 874–884.
- [49] Byeng D Youn, Kyung K Choi, and Young H Park. 2003. Hybrid analysis method for reliability-based design optimization. *J. Mech. Des.* 125, 2 (2003), 221–232.
- [50] Qingfu Zhang and Hui Li. 2007. MOEA/D: A multiobjective evolutionary algorithm based on decomposition. *IEEE Transactions on evolutionary computation* 11, 6 (2007), 712–731.

## APPENDIX

### A.1 HYBRID MEAN VALUE (HMV) METHOD

We use a gradient-based hybrid mean value (HMV) method [18, 46] to compute “most probable target point” (MPTP):  $\mathbf{u}_{HMV}$ . The gradient ( $\mathbf{n}_j$ ) of the  $j$ -th constraint is computed at point  $(\mathbf{d}, \mathbf{u})$ . The

steps in HMV method to compute MPTP ( $\mathbf{u}_{\text{HMV}}$ ) is provided as follows:

**Step 1:** Set iteration counter  $t=0$ ,  $t_{\text{max}} = T$ , convergence parameter  $\epsilon$ , Target Reliability Index  $\beta$ , and  $\mathbf{u}_{\text{HMV}}^0 = [0]_{1 \times K}$ .

**Step 2:** If  $t \leq 2$ , compute:

$$\mathbf{n}_j(\mathbf{u}_{\text{HMV}}^t) = -\frac{\nabla_{\mathbf{u}} g_j(\mathbf{d}, \mathbf{u}_{\text{HMV}}^t)}{\left\| \nabla_{\mathbf{u}} g_j(\mathbf{d}, \mathbf{u}_{\text{HMV}}^t) \right\|_2}; \quad \mathbf{u}_{\text{HMV}}^{t+1} = \beta \cdot \mathbf{n}_j(\mathbf{u}_{\text{HMV}}^t). \quad (4)$$

**Step 3:** If  $t > 2$ , compute the convexity estimate of  $g_j$  as follows:

$$\zeta_{\text{HMV}}^t = \left( \mathbf{n}_j(\mathbf{u}_{\text{HMV}}^t) - \mathbf{n}_j(\mathbf{u}_{\text{HMV}}^{t-1}) \right) \cdot \left( \mathbf{n}_j(\mathbf{u}_{\text{HMV}}^{t-1}) - \mathbf{n}_j(\mathbf{u}_{\text{HMV}}^{t-2}) \right). \quad (5)$$

**Step 4:** If  $\zeta_{\text{HMV}}^t > 0$ , that is,  $g_j$  is convex at  $\mathbf{d}$ , compute  $\mathbf{u}_{\text{HMV}}^{t+1}$  using Equation 4.

**Step 5:** Else,  $g_j(\mathbf{u}_{\text{HMV}}^t)$  is concave at point  $\mathbf{d}$ , compute:

$$\mathbf{u}_{\text{HMV}}^{t+1} = \beta \cdot \frac{\mathbf{n}_j(\mathbf{u}_{\text{HMV}}^t) + \mathbf{n}_j(\mathbf{u}_{\text{HMV}}^{t-1}) + \mathbf{n}_j(\mathbf{u}_{\text{HMV}}^{t-2})}{\left\| \mathbf{n}_j(\mathbf{u}_{\text{HMV}}^t) + \mathbf{n}_j(\mathbf{u}_{\text{HMV}}^{t-1}) + \mathbf{n}_j(\mathbf{u}_{\text{HMV}}^{t-2}) \right\|_2}. \quad (6)$$

**Step 6:** If  $|g_j(\mathbf{u}_{\text{HMV}}^{t+1}) - g_j(\mathbf{u}_{\text{HMV}}^t)| < \epsilon$ , **Stop**. Else,  $t = t + 1$ .

**Step 7:** If  $t \leq T$ , go to **Step 3**. Else **Stop**.

Based on the convexity check performed in Step 3 using Equation 5, the Hybrid Mean Value (HMV) method alternates between the Advanced Mean Value (AMV) method [37] in Step 4 and the Conjugate Mean Value (CMV) method [49] in Step 5. This allows HMV to leverage the advantages of both AMV and CMV methods, making it suitable for handling convex and moderately nonlinear concave functions [24]. For further details on HMV, readers are referred to [46].

## A.2 COMPARISON AND PERFORMANCE EVALUATION

The comparison of Bi-ReMCDM approach and ML-ReMCDM approach is conducted on the basis of their performance evaluation and computational cost of both the objective function and constraint function. Consider that the R-NSGA-III algorithm is executed with the population size per reference point as  $\text{pop}_{\text{RP}}$  and the number of reference points supplied by DM as  $N_{\text{RP}}$ . The population size in R-NSGA-III algorithm is computed as:  $\text{pop} = \text{pop}_{\text{RP}} \times N_{\text{RP}}$ .

### A.2.a Performance Evaluation

For a fair comparison of Bi-ReMCDM and ML-ReMCDM approaches, we adapt the following procedure. First, the number of generation in Bi-ReMCDM approach is fixed to  $\text{gen}_{\text{Bi}}$  and preferred reliable solutions are computed by executing the R-NSGA-III algorithm. The R-HV metric –a hypervolume-based performance metric for R-EMO algorithms– is computed for the preferred reliable solutions obtained using Bi-ReMCDM approach. Further details on R-HV metric is provided in [22, 42]. Next, with the same R-NSGA-III parameters, ML-ReMCDM approach is executed to arrive at the same convergence in terms of R-HV values of preferred reliable solutions. If  $\text{R-HV}_{\text{Bi}}$  is the R-HV value of Bi-ReMCDM solutions, we apply the following convergence criteria for R-NSGA-III in ML-ReMCDM approach:

$$\text{R-HV}_{\text{ML}} \in [0.95 \times \text{R-HV}_{\text{Bi}}, 1.05 \times \text{R-HV}_{\text{Bi}}] \quad (7)$$

The R-NSGA-III termination criteria for ML-ReMCDM approach is to satisfy the convergence presented in Equation 7 for five consecutive generations computed in interval of 20 generations. Further details on R-HV based convergence is provided in [3]. The objective performance for both the Bi-ReMCDM and ML-ReMCDM approaches is evaluated as  $\text{R-HV}_{\text{Bi}}$  and  $\text{R-HV}_{\text{ML}}$ , respectively. Hence, the objective performance for these methods is computed as:

$$P_{\text{Bi}}^{\text{obj}} = \text{R-HV}_{\text{Bi}}, \quad P_{\text{ML}}^{\text{obj}} = \text{R-HV}_{\text{ML}}.$$

In our case, the objective performance of both the ReMCDM and ML-ReMCDM approaches are expected to be similar.

For a given reliability index ( $\beta$ ), consider that after executing the R-NSGA-III,  $\text{pop}_{\text{Bi}}$  and  $\text{pop}_{\text{ML}}$  are the number of preferred reliable solutions obtained in Bi-ReMCDM and ML-ReMCDM approach, respectively. The number of preferred reliable solutions  $\text{pop}_{\text{Bi}}$  and  $\text{pop}_{\text{ML}}$  can vary depending on the efficiency of constraint handling and reliability estimation in Bi-ReMCDM and ML-ReMCDM approaches. Hence, the constraint performance for these methods is computed as:

$$p_{\text{Bi}}^{\text{cons}} = \frac{\text{pop}_{\text{Bi}}^{\beta}}{\text{pop}}, \quad p_{\text{ML}}^{\text{cons}} = \frac{\text{pop}_{\text{ML}}^{\beta}}{\text{pop}}; \quad p_{\text{Bi}}^{\text{cons}}, p_{\text{ML}}^{\text{cons}} \in [0, 1].$$

where 1 corresponds to the best constraint performance and 0 indicates the worst constraint performance.

### A.2.b Computational Cost

Consider that the number of generations in R-NSGA-III while executing ML-ReMCDM approach to arrive the termination criteria defined above is  $\text{gen}_{\text{ML}}$ . The number of objective function evaluations for Bi-ReMCDM approach is:

$$E_{\text{Bi}}^{\text{Obj}} = \text{gen}_{\text{Bi}} \times \text{pop} \times M,$$

whereas, the number of objective function evaluations for ML-ReMCDM approach is:

$$E_{\text{ML}}^{\text{Obj}} = \text{gen}_{\text{ML}} \times \text{pop} \times M,$$

where  $M$  is the number of objective functions.

Consider that at a given design point, on average  $\bar{k}_j$  iterations are required in HMV method for the  $g_j$ th constraint to estimate the MPTP. Gradient evaluation in the HMV method using the central difference technique requires  $(2N + 1)$  evaluation per constraint at each design point, where  $N$  is the number of variables [8]. The constraint cost for the Bi-ReMCDM approach is computed as:

$$E_{\text{Bi}}^{\text{Cons}} = (2N + 1) \times \sum \bar{k}_j \times \text{pop} \times \text{gen}_{\text{Bi}}.$$

However, the ML-ReMCDM approach requires  $S_j$  LHS samples for the  $g_j$ th constraint to build the ML model using Multi-layer perceptron. During execution of R-NSGA-III, ML-ReMCDM approach does not require actual constraint evaluation; instead it uses trained ML models of reliability-based constraints to replace the constraints in MOO formulation presented in Equation 2 (in Main paper). So, the constraint functions are only evaluated to generate  $S_j$  samples and to perform gradient computation on those samples using HMV

method. The constraint evaluation for ML-ReMCDM approach is provided as:

$$E_{ML}^{\text{Cons}} = (2N + 1) \times \sum \bar{k}_j \times S_j + \sum S_j.$$

Since the metric  $E_{ML}^{\text{Cons}}$  is independent of the term  $\text{pop}_{ML}$  and  $\text{gen}_{Bi}$ , the computational cost of constraint evaluation for the ML-ReMCDM approach is expected to be significantly lower than that of the Bi-MCDM approach. In the result section, we use same number of LHS samples for each constraint. However, for computational efficiency the LHS samples can vary for each constraints.

### A.3 ANALYTICAL EXPRESSIONS OF BENCHMARK EXAMPLES

The analytical expressions for objective functions, constraint functions, and bounds of design variables are provided in this section.

#### A.3.a Benchmark Example 1

Analytical expression of objectives and constraints for Benchmark Example 1 is provided as follows:

$$\begin{aligned} f_1(\mathbf{x}) &= x_1^2 + x_2^2, \\ f_2(\mathbf{x}) &= (x_1 - 1)^2 + (x_2 - 1)^2, \\ g(\mathbf{x}) &= x_1^2 + x_2^2 - 1 \leq 0, \\ [\mathbf{x}^{lb}, \mathbf{x}^{ub}]_i &= [0, 1], \quad i = 1, 2. \end{aligned} \quad (8)$$

#### A.3.b Benchmark Example 2

Analytical expression of objectives and constraints for Benchmark Example 2 is provided in [9] as follows:

$$\begin{aligned} f_1(\mathbf{x}) &= x_1, \\ f_2(\mathbf{x}) &= \frac{1 + x_2}{x_1}, \\ g_1(\mathbf{x}) &= 9x_1 + x_2 - 6 \leq 0, \\ g_2(\mathbf{x}) &= 9x_1 - x_2 - 1 \leq 0, \\ \mathbf{x}^{lb} &= [0.1, 0.0], \quad \mathbf{x}^{ub} = [1.0, 5.0]. \end{aligned} \quad (9)$$

#### A.3.c Benchmark Example 3

Analytical expression of objectives and constraints for Benchmark Example 3 is provided as follows:

$$\begin{aligned} f_1(\mathbf{x}) &= x_1 + x_2, \\ f_2(\mathbf{x}) &= -x_1 + x_2, \\ f_3(\mathbf{x}) &= \frac{1}{1 + x_2}, \\ g_1(\mathbf{x}) &= 1 - (x_1 - 1)^2 - (x_2 - 1)^2 \leq 0, \\ g_2(\mathbf{x}) &= 1 - (x_1 - 2)^2 - (x_2 - 1)^2 \leq 0, \\ g_3(\mathbf{x}) &= 1 - (x_1 - 1.5)^2 - \left(x_2 - 1 - \frac{\sqrt{3}}{2}\right)^2 \leq 0, \\ \mathbf{x}^{lb} &= [1.0, 0.85], \quad \mathbf{x}^{ub} = [2.15, 2.0] \end{aligned} \quad (10)$$

#### A.3.d Benchmark Example 4

Analytical expression of objectives and constraints for Benchmark Example 4 is provided as follows:

$$\begin{aligned} f_1(\mathbf{x}) &= x_1 + x_2, \\ f_2(\mathbf{x}) &= -x_1 + x_2, \\ f_3(\mathbf{x}) &= \frac{1}{1 + x_2}, \end{aligned} \quad (11)$$

$$\begin{aligned} g_1(\mathbf{x}) &= (x_1 + 1.5)^2 + \left(x_2 - 1 - \frac{\sqrt{3}}{2}\right)^2 - 4 \leq 0, \\ g_2(\mathbf{x}) &= (x_1 - 2.5)^2 + \left(x_2 - 1 - \frac{\sqrt{3}}{2}\right)^2 - 4 \leq 0, \\ g_3(\mathbf{x}) &= (x_1 - 0.5)^2 + \left(x_2 + 1 + \frac{\sqrt{3}}{2}\right)^2 - 5 \leq 0, \\ \mathbf{x}^{lb} &= [-1.5, 0.0], \quad \mathbf{x}^{ub} = [2.5, 2.0] \end{aligned} \quad (12)$$

### A.4 EFFECT OF RELIABILITY INDEX ( $\beta$ ) ON RELIABLE FRONT

In this section, we discuss the effect of reliability index ( $\beta$ ) on the Reliable front (ReF). We conduct this study on two Benchmark Examples discussed in the Main paper.

#### A.4.a Benchmark Example 1

This is a bi-objective problem having two uncertain design variables and one constraint. Figure 9a represents PF in black dots; and ReF in blue, green, and magenta dots, corresponding to reliability indices  $\beta=2.0$ ,  $\beta=3.0$ , and  $\beta=4.0$ , respectively. Note that two extreme points of the ReF is also found to provide stability in estimating the ideal point  $O$ . It can be observed that the ReF is a part of PF shifted towards the higher values of  $f_2$  leaving out the lower  $f_2$  as infeasible solutions for accommodating the variable uncertainty while handling the reliability constraint. As the target reliability increases from 2.0 to 4.0 the ReF (being the part of PF) shifts toward the higher  $f_2$  values.

Figure 9b presents the scatter plot of Pareto solutions in black dots and reliable solutions (blue, green, and magenta) in the design space at different target reliability indices (2.0, 3.0, and 4.0). It can be observed that as the target reliability index ( $\beta$ ) increases, the reliable solutions move away farther from the constraint boundary. The adjacent color bar suggests that the reliable solutions shrink towards the region corresponding to higher  $f_2$  and lower  $f_1$  values.

#### A.4.b Benchmark Example 2

Benchmark Example 2 is referred from [9]. Figures 10a and Figure 10b and represent deterministic Pareto solutions and reliable solutions in objective space and design space, respectively. Figure 10b suggests that the Pareto solutions are located at the constraint boundary. As the target reliability index is increased from 2.0 to 4.0 the reliable solutions shift inside the feasible space to accommodate the variable uncertainty and to satisfy the reliability constraints. Similarly, it can be observed from Figure 10a that a part of the ReF shifts inside the objective space. As the target reliability

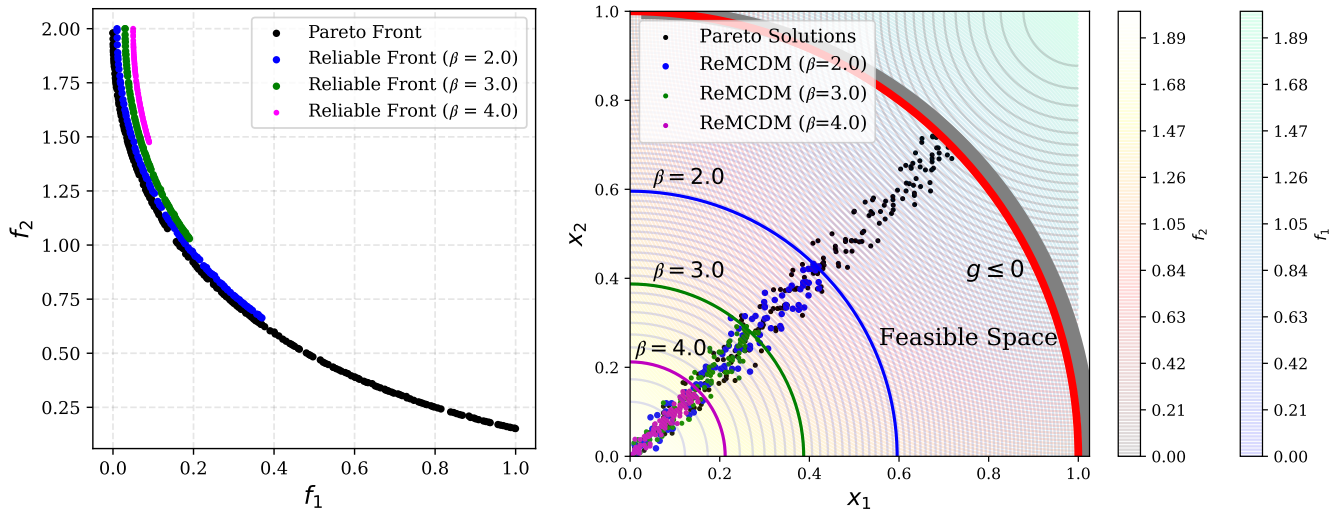
(a) Reliable fronts at different reliability index ( $\beta$ ) values(b) Reliable solutions at different reliability index ( $\beta$ ) values

Figure 9: Benchmark Example 1: Reliable solutions obtained using NSGA-II algorithm [43] (a) Pareto optimal front (black dots) and reliable fronts (ReFs) at different values of  $\beta = [2.0, 3.0, 4.0]$  are highlighted in blue, green, and magenta dots, respectively. The contour plots for  $f_1$  and  $f_2$  with color legends shown on the right of the figures indicate the conflicting behavior of two objectives, (b) Scatter plot of Pareto front (black dots) and reliable front at different values of  $\beta = [2.0, 3.0, 4.0]$  highlighted in blue, green, and magenta dots, respectively. Although ReFs lie on top of each other, ReFs are plotted with a slight shift to show clearly the extent of the respective fronts.

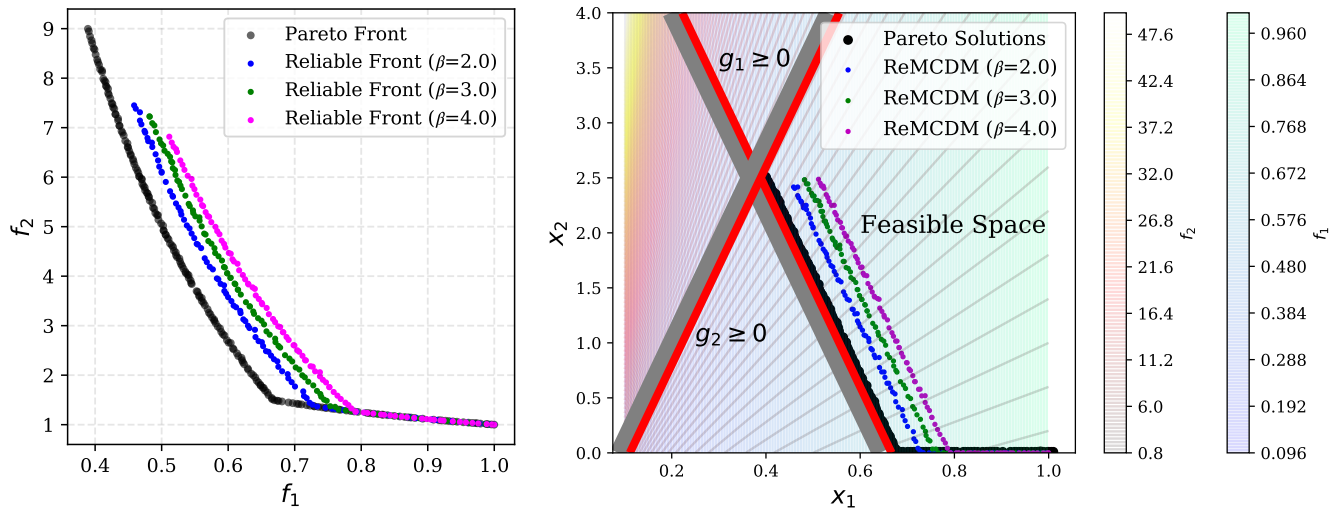
(a) Reliable fronts at different reliability index ( $\beta$ ) values(b) Reliable solutions at different reliability index ( $\beta$ ) values

Figure 10: Benchmark Example 2: Reliable solutions obtained using NSGA-II algorithm [43] (a) Pareto optimal front (black dots) and reliable fronts (ReFs) at different values of  $\beta = [2.0, 3.0, 4.0]$  are highlighted in blue, green, and magenta dots, respectively. Here, the fronts are not deliberately shifted, rather they are naturally shifted due to higher clearances needed from the constraint boundaries to achieve a higher reliability index. The contour plots for  $f_1$  and  $f_2$  with color legends shown on the right of the figures indicate the conflicting behavior of two objectives, (b) Scatter plot of Pareto front (black dots) and reliable front at different values of  $\beta = [2.0, 3.0, 4.0]$  highlighted in blue, green, and magenta dots, respectively.

**Table 3: Hyperparameters used in Multi-layer Perceptron (MLP) for benchmark and real-world examples implemented in proposed ML-ReMCDM approach. For a given problem, we use the same configuration of MLP hyperparameters for each reliability-based constraints. However, different configurations can be used to further improve the accuracy of an individual ML model.**

Examples	Hidden Layers	Max Iter.	Learning Rate	Decay Factor
Benchmark 1	(10, 10)	200	0.015	0.0025
Benchmark 2	(100, 100)	5,000	0.001	0.00125
Benchmark 3	(100, 100)	10,000	0.01	0.0025
Benchmark 4	(100, 100)	10,000	0.02	0.0015
I-Beam	(100, 100)	2,000	0.0015	0.0025
Disc Brake	(100, 100)	5,000	0.001	0.00125
Welded-Beam	(100, 100)	1,000	0.001	0.00125
Two-Bar Truss	(100, 100)	500	0.001	0.00125
Car Side Impact	(100, 100)	5,000	0.01	0.00125
WATER Example	(100, 100)	500	0.001	0.00125

**Table 4: Population size and number of generations used in NSGA-II and NSGA-III algorithm to generate PF and ReF on the benchmark and real-world examples used in the Main paper.**

Examples	Algorithm	PF		ReF	
		Pop	Gen	Pop	Gen
Benchmark 1	NSGA-II	100	100	100	100
Benchmark 2	NSGA-II	100	100	100	100
Benchmark 3	NSGA-III	351	100	351	100
Benchmark 4	NSGA-III	351	100	351	100
I-Beam Design	NSGA-II	100	100	100	100
Disc Brake Design	NSGA-II	200	200	200	100
Welded-Beam Design	NSGA-II	200	200	200	100
Two-Bar Truss	NSGA-II	200	100	100	100
Car Side Impact	NSGA-III	1,326	50	1,326	50
WATER Example	NSGA-III	3,876	50	3,876	50

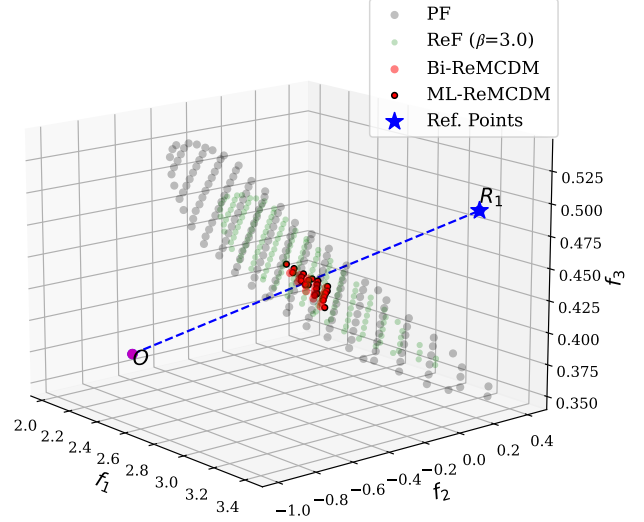
index increases from 2.0 to 4.0, the extent of the shifting in objective space also increases.

## A.5 ADDITIONAL RESULTS

This section discusses the addition results for the Benchmark Example 3 and Benchmark Example 4 mentioned in the Main paper. Both the benchmark examples are three-objective constrained MOO problem, consisting of two design variables and two constraints. Table 3 provides the details on ML hyper-parameters used for training Multi-layer Perceptron. Table 4 outlines the NSGA-II and NSGA-III parameters used for computing Pareto front and Reliable front.

### A.5.a Benchmark Example 3

The analytical expression for Benchmark Example 3 is provided in Section 6 and additional descriptions are provided in Table 1 of the



**Figure 11: ML-ReMCDM approach implemented on Benchmark Example 3. Scatter plot indicates Pareto front (black dots) and reliable front at  $\beta = 3.0$  (green dots). The ReMCDM solution corresponding to reference point  $R_1$  is highlighted in red dots. ML-ReMCDM solutions are indicated by red dots with black boundary.  $R_1$  is the user-defined reference point.  $O$  is the Ideal point estimated from the extreme reference directions of R-NSGA-III algorithm.**

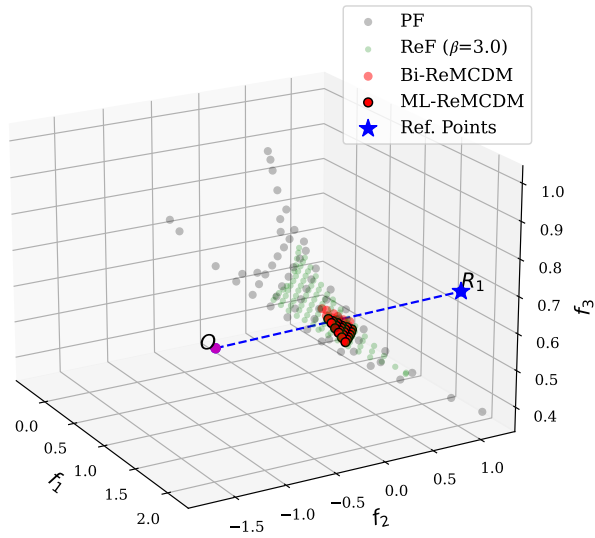
Main paper. The results for Benchmark Example 3 is presented in Figure 11. Black and green dots represent PF and ReF, respectively. In this example, ReF is part of PF. A single reference point  $R_1 = [3.4, 0.2, 0.5]$  is used to obtain preferred reliable solutions using Bi-ReMCDM and ML-ReMCDM approaches.

In Figure 11, red dots represent Bi-ReMCDM solutions and red dots with black boundary represent ML-ReMCDM solutions. The details on ML Model hyper-parameters are provided in Table 2 of the Main paper. R-NSGA-III parameters used in both approaches; and computational cost and performance of Bi-ReMCDM and ML-ReMCDM approaches are provided in Table 3 of the Main paper. Table 3 in Main paper indicates that the objective and constraint evaluation in ML-ReMCDM approach is significantly less than objective and constraint evaluation in Bi-ReMCDM approach.

Additional details on problem description and ML model hyper-parameter details for Benchmark Example 3 is provided in Table 1 and Table 2, respectively, of the Main paper.

### A.5.b Benchmark Example 4

The analytical expression for Benchmark Example 4 is provided in Section 6 and additional descriptions are provided in Table 1 of the Main paper. The results for Benchmark Example 3 are shown in Figure 12. Black and green dots represent the PF and ReF, respectively. In this case, the ReF is a subset of the PF. A single reference point,  $R_1 = [2.0, 1.0, 0.7]$ , is used to evaluate preferred reliable solutions using the Bi-ReMCDM and ML-ReMCDM approaches.



**Figure 12: ML-ReMCDM approach implemented on Benchmark Example 4.** Scatter plot indicates Pareto front (black dots) and reliable front at  $\beta = 3.0$  (green dots). The ReMCDM solution corresponding to reference point  $R_1$  is highlighted in red dots. ML-ReMCDM solutions are indicated by red dots with black boundary.

In Figure 11, red dots indicate Bi-ReMCDM solutions, while red dots with black boundaries represent ML-ReMCDM solutions. The hyper-parameter details of the ML model are provided in Table 2 of the main paper. The R-NSGA-III parameters used in both approaches, along with the computational cost and performance comparison of Bi-ReMCDM and ML-ReMCDM, are presented in Table 3 of the main paper. Table 3 also highlights that the number of objective and constraint evaluations in the ML-ReMCDM approach is significantly lower than in the Bi-ReMCDM approach.