# Infeasibility Driven Approach for Bi-objective Evolutionary Optimization

Deepak Sharma and Prem Soren
Department of Mechanical Engineering,
Indian Institute of Technology Guwahati, Assam, India, PIN– 781039
Email: {dsharma,p.soren}@iitg.ernet.in

***Abstract*—Infeasibility driven approach is proposed in this paper for constrained bi-objective optimization using evolutionary algorithm. The idea is motivated from one of the constraint handling techniques in which infeasible solutions are preserved in the population for focusing the optimal solution lying on the boundary of feasible region. In the proposed approach, extreme solutions of the current non-dominated front are allowed to recombine only with extreme infeasible solutions. This restricted mating is expected to generate offspring towards the "Pareto-optimal" front and reduces number of generations required to evolve comparative results against existing multi-objective evolutionary algorithm (MOEA). Although the proposed approach is generic and can be coupled with any MOEA, but for bench-marking purpose it is coupled with NSGA-II (refer as IDMOEA) and is tested on four engineering optimization problems. On an average for 30 different runs, IDMOEA shows quicker convergence than NSGA-II with equivalent quality of solutions assessed by indicator analysis.**

## I. INTRODUCTION

Performance of any evolutionary algorithm (EA) for real-world optimization problems in which constraints often appear, depends on the constraint handling technique. In constrained optimization, mostly the EA population is divided into feasible and infeasible sets of solutions. The survival of a solution in the population then depends on the overall fitness that is based on objective function and constraint violation. There exist many constraint handling techniques in the literature for EA [1]. Among them, one of the constraint handling techniques prefers infeasible solutions close to the feasible region than feasible solutions. It generally happens with real-world optimization problems where optimal solution lies on the boundary of feasible region [2]. In such situation, infeasible solutions closer to the optimal solutions can help EA to converge quickly than feasible solutions away from the optima. Therefore, it is more desirable to focus the search on the constraint boundary between the feasible and infeasible regions [1].

In the literature, few methods have been suggested to prefer infeasible solutions over feasible solutions. In one of the approaches, problem specific genetic operators can be developed that can search the boundary of feasible region [2], [3]. In another approach, the fitness of infeasible solutions close to the feasible region can be assigned such that these solutions become competitive and can survive in the population against feasible solutions. It can be done by separate handling of objective function and constraints for the given problem [4], [5]. When dealing with constraints, infeasible solutions with the lowest sum of constraint violation in the population can be kept for the next generation.

In study [6], a probability factor is used to calculate the rank of each solution that depends on either objective function value or constraint violation. A multi-objective optimization technique is used in the study [7] in which a constrained optimization problem is converted into multi-objective optimization problem. In this study, the population based algorithm generators and infeasible solutions archiving and replacement mechanism are used. Three different non-dominated ranking procedures are adopted in [8] where ranking of a solution is done (1) using the objective function values, (2) using all constraints, and (3) using all objective functions and constraints together. In the study [9], a two-phase algorithm is used where the objective function is completely disregarded and the constraint optimization problem is treated as a constraint satisfaction problem in first phase. In the second phase, both constraint satisfaction and objective optimization are treated as a bi-objective optimization problem. An approach adapted in [10], [11] also maintains infeasible solutions in the population by reformulating the original $k-$objective constrained minimization problem into an unconstrained $k+1-$objective minimization problem. An additional objective function such as the number of constraint violations [10] or violation measure [11] can be used. In this approach, the higher ranks are assigned to the infeasible solutions than the feasible solutions, thereby focusing the search near to the constraint boundaries for the optimal solution through infeasible region. Alternative approach in the study [12] uses a norm of objective value and constraint violation when the population is mixed of feasible and infeasible solutions. When the complete population is infeasible, the norm of constraint violation is used.

It can be observed from above studies that preferring infeasible solutions in the populations can drive EAs to locate optima easily and quickly. Moreover, a small proportion of infeasible solutions in the population is enough to improve the performance of EA [10], [13]. In this paper, the infeasibility driven approach is adopted from the above constraint handling technique and only extreme infeasible solutions in the objective space are preserved in the population. The proposed approach is developed to use restricted mating of extreme current non-dominated solutions with extreme infeasible solutions, and generate solutions towards the "Pareto-optimal" front. The remaining paper is organized in five sections. In section II, the proposed approach is discussed and its implementation with EA scheme is shown. In section III, four engineering optimization problems are solved and statistical results obtained from number of generations, $R$ and hypervolume indicators, and attainment plots are shown. The outcome from the proposed approach and comparison with NSGA-II is discussed in section

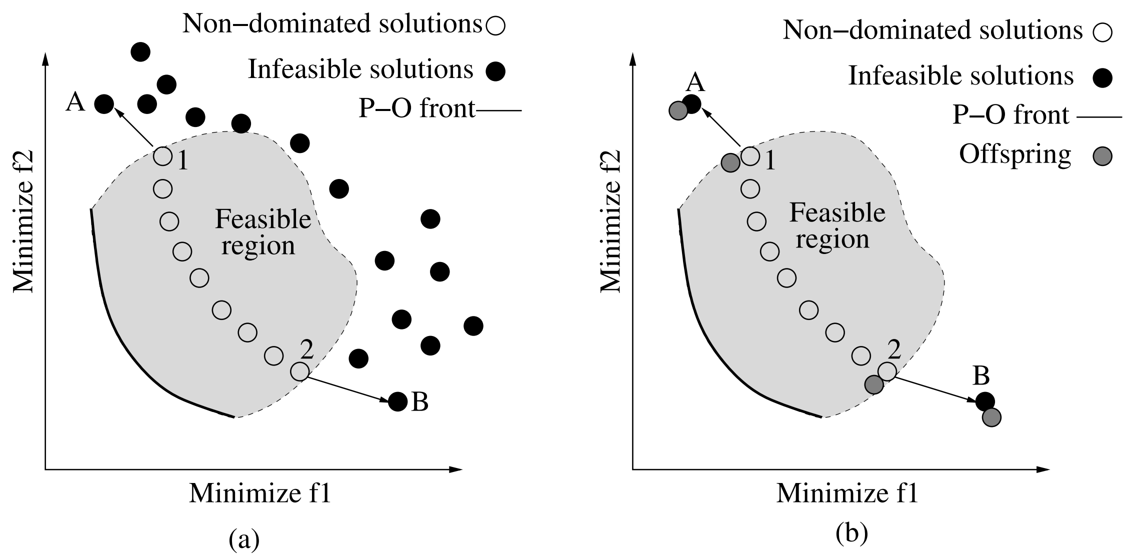


Fig. 1. Part (a) shows the P-O front, a set of current non-dominated solutions, and infeasible solutions. Part (b) shows the expected outcome from the proposed approach where offspring solutions are created closer to the P-O front.

IV and the paper is concluded in section V with suggested further work.

**Algorithm 1** Scheme for MOEA

1: Generation counter $(t = 1)$;
2: Initialize random population $(P(t))$;
3: Evaluate $(P(t))$ and assigned fitness;
4: Find extreme infeasible solutions
5: **while** Termination condition not met **do**
6: $P'(t)$ := Copy extreme infeasible solutions + Selection$(P(t))$;
7: $P''(t)$ := Variation$(P'(t))$;
8: Evaluate $P''(t)$ and assign fitness;
9: Update extreme infeasible solution if found better in $P''(t)$
10: $P(t+1)$ := Extreme infeasible solutions + Survivor$(P(t) + P''(t))$;
11: $t := t + 1$;
12: **end while**

## II. Infeasibility Driven Approach

The idea is motivated from one of the constraint handling techniques in which infeasible solutions are preserved in the population for focusing the optimal solution lying on the constraint boundary [1]. The idea is primarily developed for constrained bi-objective optimization. In the proposed approach, two extreme infeasible solutions are preserved in the population. A hypothetical case is shown in Fig. 1(a). An infeasible solution $A$ is extreme solution in $f_1-$objective. Meaning, solution $A$ is having minimum value of $f_1-$objective value among all the infeasible solutions in the current population. Similarly, an infeasible solution $B$ is extreme in $f_2-$objective. These two extreme infeasible solutions are preserved to allow restricted mating with extreme solutions of the current non-dominated front. As shown in Fig. 1(a), non-dominated solution 1 is extreme in $f_1-$objective and non-dominated solution 2 is extreme in $f_2-$objective. In this case, solution 1 is allowed to recombine with solution $A$ only, and solution 2 can recombine with solution $B$ only. A recombination operator that can generate offspring near to parent solutions is preferred for the proposed approach as shown in Fig. 1(b). It is expected that offspring solutions generated closer to the "Pareto-optimal" (P-O) front can assist MOEAs for quicker convergence towards the P-O front. Moreover, the proposed approach is focusing on extreme solutions, it can help MOEAs to generate a wider P-O front. If offspring solutions are not better or infeasible, then convergence of MOEAs towards the P-O front will be guided by their original procedure.

### A. Implementation

Implementation of the proposed idea is done according to the scheme shown in algorithm 1. After evaluating initial population in step 3, extreme infeasible solutions in the objective space are preserved in the population by deleting two worst solutions of the current population.

*1) Termination Conditions (Step 5 of algorithm 1):* In this paper, two termination conditions are set. First termination condition is based on the normalized distance $(ND)$ metric [14] which is calculated as:

$$ND = \sqrt{\frac{1}{M}\sum_{i=1}^{M}\left(\frac{z_i^{est} - z_i^*}{z_i^w - z_i^*}\right)^2} \tag{1}$$

$M$ is number of objectives, $z_i^*$ is the ideal point, $z_i^w$ is the worst point and $z_i^{est}$ is the estimated Nadir point at any generation of MOEA. Pictorially, these points are shown in Fig. 2 for the P-O front. In this work, $z_i^*$ and $z_i^w$ are the best ideal and worst points in the population found so far. If value of ideal point and/or worst point in the current generation is better than $z_i^*$ and/or $z_i^w$, then values of $z_i^*$ and/or $z_i^w$ get updated. $z_i^{est}$ can be found from the extreme solutions of the current non-dominated front.

$ND$ values are stored for every generation, and maximum $(ND_{max})$, minimum $(ND_{min})$ and average $(ND_{avg})$

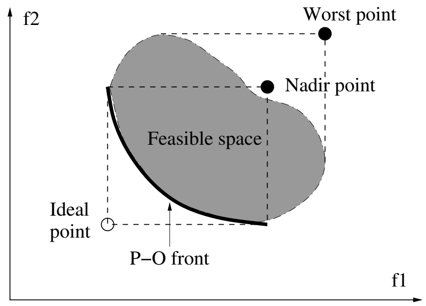


Fig. 2. Ideal point, Nadir point, and worst point in two-objective space.

values of $ND$ are calculated from last $\tau$ generations. When $(ND_{max} - ND_{min}/ND_{avg}) \leq \Delta$, then MOEA terminates. Otherwise, MOEA gets terminated after completion of 1000 generations. An adaptive terminating condition is helpful to determine number of generations required by the proposed approach against existing MOEAs. This termination condition signifies that the non-dominated front of MOEA is not progressing over last $\tau$ generations by finding the movement of ideal, nadir and worst points. Here, less number of generations signifies less functional evaluations and quicker convergence. An alternative approach like targeting a specified value of hypervolume or other indicator can be used. But for the given set of engineering optimization problems, the set of P-O solutions or the reference set of non-dominated solutions are not known.

*2) Selection (Step 6 of algorithm 1):* In this step, extreme infeasible solutions are directly copied into $P'(t)$ by replacing two worst solutions in the current population. A binary tournament selection operator without replacement variant is adopted in which two solutions are randomly picked-up for tournament but not replaced back them into the population. This variant of binary tournament selection operator is applied twice on the population to keep the same size $N$ for $P'(t)$. Here, the best solution will get the two copies and rest of the solutions can have 0, 1 or 2 copies. Interested readers can refer [15] for more details of this variant of binary tournament selection operator.

During the selection, solution 1 and solution 2 (refer to Fig. 1) are not allowed to compete with each other. But, these two solutions always get selected when they compete with other solutions in the current population. By performing binary tournament selection operator twice on $P(t)$, two copies of solutions 1 and 2 are generated in $P'(t)$. One copy of solutions 1 and 2 is allowed to recombine with solutions $A$ and $B$, respectively. Another copy of solutions 1 and 2 can recombine with any solution in the population, randomly. Similarly, other solutions in the population can recombine, randomly.

If the restricted mating generates offspring solutions closer to the P-O front, then these offspring can become extreme solutions in the next generation. In the selection process of next generation, these offspring will get two copies in the population for recombination. A copy of offspring solutions which is allowed to recombine with any solution in the population, can assist MOEAs to generate intermediate solutions of the non-dominated front towards the P-O front in later generations.

*3) Variation (Step 7 of algorithm 1):* In this work, SBX crossover operator and polynomial mutation operator are used for variation in the population [16]. SBX operator is preferred because offspring solutions near to the parents are monotonically more likely to be chosen than solutions distant from parents [15]. However, other crossover operator can also be used for generating near-parent offspring solutions. When SBX operator is applied on solutions 1 and $A$, it can generate two offspring solutions closer to the parent solutions as shown in Fig. 1 (b). If offspring solutions are generated towards the P-O front, then these solutions can pull the current non-dominated front towards the P-O front in later generations via recombination. If offspring solutions are not better or infeasible, then convergence of MOEAs towards the P-O front will be guided by their original procedure.

*4) Updating extreme infeasible solutions (Step 9 of algorithm 1):* As indicated in Fig. 1(b), offspring solutions are also created near to extreme infeasible solutions $A$ and $B$. It can be possible that offspring solutions closer to these extreme infeasible solutions are infeasible and further extreme in objectives $f_1$ and $f_2$. Therefore, an updating of extreme infeasible solutions is done to make sure that $f_1^A \leq f_1^1$ and $f_2^B \leq f_2^2$, where $f_1^A$, $f_1^1$, $f_2^B$ and $f_2^2$ are $1^{st}$ objective value of solution $A$ and solution 1, and $2^{nd}$ objective value of solution $B$ and solution 2, respectively.

*5) Next generation population (Step 10 of algorithm 1):* The population for next generation is created by copying two extreme infeasible solutions. Rest of the population is filled from offspring and parent population by following elitist $(\mu + \lambda)$ strategy.

## III. SIMULATION RESULTS

The proposed idea is generic and can be coupled with any existing MOEA. In this paper, the infeasibility drive approach is coupled with NSGA-II [17] for benchmarking purpose and is tested on four engineering optimization problems from mechanical engineering domain. The proposed approach with NSGA-II is referred as infeasibility driven MOEA (IDMOEA). Two indicators and attainment plots are used for comparing the results from IDMOEA against the results from NSGA-II. Details of these indicators are as follows:

*R-indicator ($I_{R2}$) [18]:*

$$I_{R2} = \frac{\sum_{\lambda \in \Lambda} u^\star(\lambda, A) - u^\star(\lambda, R)}{|\Lambda|} \tag{2}$$

where $R$ is a reference set, $u^\star$ is the maximum value reached by the utility function $u$ with weight vector $\lambda$ on an approximate set $A$, that is, $u^\star = \max_{z \in A} u_\lambda(z)$. The augmented Tchebycheff function is used as the utility function. The second order $R$-indicator gives the idea of proximity with respect to the reference set $A$ [19].

*Hypervolume indicator ($I_{\bar{H}}$) [20]:* The hypervolume indicator $I_H$ measures the hypervolume of that portion of the objective space that is weakly dominated by an approximate set $A$. This indicator gives the idea of spread quality and has to be maximized. As recommended in the study [19], the difference in values of hypervolume indicator between the approximate set $A$ and the reference set $R$ is calculated in this paper, that is, $I_{\bar{H}} = I_H(R) - I_H(A)$. The smaller value suggests good spread [19].

*Attainment surface [21]:* An approximate set $A$ is called the $k\% - approximate\ set$ of the empirical attainment function (EAF) $\alpha_r(z)$, iff it weakly dominates exactly those objective vectors that have been attained in at least $k$ percentage of the $r$ runs. Formally, $\forall \in Z : \alpha_r(z) >= k/100 \leftrightarrow A \preceq \{z\}$ where $\alpha_r(z) = \frac{1}{r}\sum_{i=1}^{r} I(A^i \preceq \{z\})$. $A^i$ is the $i$th approximation set (run) of the optimizer and $I(.)$ is the indicator function, which evaluates to one if its argument is true and zero if its argument is false.

An attainment surface of a given approximate set $A$ is the union of all tightest goals that are known to be attainable as a result of $A$. Formally, this is the set $\{z \in \Re^n : A \preceq z \wedge \neg A \prec\prec z\}$ [19].

As recommended in [19], minimum and maximum limits on each objective are used to normalize and further scale the approximate sets of MOEAs and the reference set between 1 and 2. On this basis, the reference set ($z^*$) of (1, 1) and (2, 2) are used to evaluate $I_{R2}$ values whereas, (2.1, 2.1) is assigned as the reference point ($z^+$) for calculating the $I_{\bar{H}}$ values. The suggested range of $I_{R2}$ and $I_{\bar{H}}$ lies between $-1$ and $+1$, where $-1$ is the best and $+1$ is considered as worst. The reference set for four engineering optimization problems are developed by finding the non-dominated solutions from the approximate sets evolved by IDMOEA and NSGA-II over different runs.

*Parameters*

A few parameters are kept constant for IDMOEA and NSGA-II which are given below:

| | | | |
|---|---|---|---|
| Population: | 100 | Generation: | 1000 |
| $\tau$: | 50 | $\Delta$: | $1e^{-6}$ |
| Prob. of crossover: | 0.9 | Prob. of mutation: | 0.1 |
| $\eta_c$ for SBX: | 15 | $\eta_m$ for poly. mut. | 20 |

Both the algorithms are run for 30 different times on four engineering optimization problems.

### A. Machining Parameter Optimization of Milling Process

Milling is a metal cutting process in which rotary cutter moves on the workpiece and removes material. In milling operation, depth of cut, feed rate and cutting speed are three variables which are responsible for optimum cutting. A workpiece that is produced by milling is shown in Fig. 3. In the given problem, five milling operations are performed to get a final design of product that includes step, pocket, corner and two slot milling operations. The problem formulation [22] for multi-tool milling is given below:

| | | |
|---|---|---|
| Minimize: | $C_u$ | (Unit cost in \$) |
| Minimize: | $T_u$ | (Unit time in min) |
| Subjected to: | | |
| | $C_5 V f^0.8 \leq 1$ | (Power constraint) |
| | $C_6 f^2 \leq 1$ | (Surface finish constraint for end milling) |
| | $C_7 f \leq 1$ | (Surface finish constraint face milling) |
| | $C_8 F_C \leq 1$ | (Cutting force constraint) |

(3)

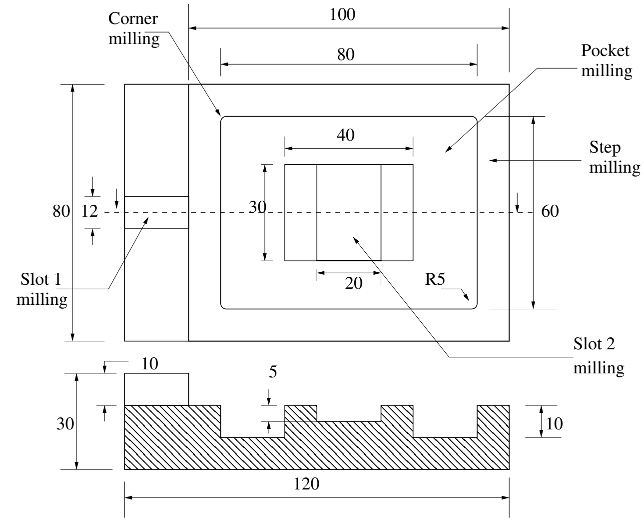


Fig. 3. Final design of workpiece that is manufactured using five milling operations.

The unit cost ($C_u = C_1 + \sum_{i=1}^{m} C_{2i} V_i^{-1} f_i^{-1} + \sum_{i=1}^{m} C_{3i} V_i^{(1/n)-1} f_i^{[(w+g)/n]-1} + \sum_{i=1}^{m} C_{4i}$) is the sum of material cost, setup cost, machining cost, and tool changing cost. The unit time ($T_u = t_s + \sum_{i=1}^{m} K_{1i} V_i^{-1} f_i^{-1} + \sum_{i=1}^{m} t_{tci}$) is comprised of setup time, machining time, and tool changing time. In constraints, $C_5 = 0.78 K_p W z a_{rad}(a/60\pi d e P_m)$, $C_6 = 318(4d)^{-1}/R_{a(at)}$, $C_7 = 318[tan(la) + cot(ca)]^{-1}/R_{a(at)}$, and $C_8 = 1/F_{C(per)}$. The nomenclature and values of various constant parameters used in milling formulation are given in appendix A.

As two termination conditions are used for IDMOEA and NSGA-II, the statistical data of number of generations required for terminating MOEAs for 30 different runs is shown in Table I. We can see that NSGA-II shows best value for number of generations based on $ND$ metric value as compared to IDMOEA. But, few of the runs in both algorithms need complete 1000 generations. Values of mean and median show that IDMOEA requires less number of generations to get terminated by $ND$ metric value. This suggested that on an average the proposed approach can reduce function evaluations by reducing number of generations. Variance in the generation data is also found less in IDMOEA against NSAG-II.

Now it's a time to check performance of IDMOEA. The motive is to generate similar or better results by IDMOEA against NSGA-II. Otherwise, the improved values of mean, median and standard deviation of generation data (cf. Table

TABLE I. NUMBER OF GENERATIONS REQUIRED FOR TERMINATING MOEAS.

| MOEAs ↓ | Mean | Median | Standard deviation | Best | Worst |
|---|---|---|---|---|---|
| IDMOEA | **678.9** | **677.5** | **175.9** | 401 | **1000** |
| NSGA-II | 706.5 | 694.5 | 220 | **291** | **1000** |

TABLE II. STATISTICAL VALUES OF $I_{R2}$ INDICATOR FOR MILLING. THE SUGGESTED RANGE OF $I_{R2}$ LIES BETWEEN $-1$ AND $+1$, WHERE $-1$ IS THE BEST AND $+1$ IS THE WORST.

| MOEAs ↓ | Mean | Median | Standard deviation | Best | Worst |
|---|---|---|---|---|---|
| IDMOEA | **0.0325** | **0.0167** | 0.0468 | 0.0009 | 0.2214 |
| NSGA-II | 0.0340 | 0.0203 | **0.0435** | **0.0006** | **0.1474** |

TABLE III. STATISTICAL VALUES OF $I_{\bar{H}}$ INDICATOR FOR MILLING. THE SUGGESTED RANGE OF $I_{\bar{H}}$ LIES BETWEEN $-1$ AND $+1$, WHERE $-1$ IS THE BEST AND $+1$ IS THE WORST.

| MOEAs ↓ | Mean | Median | Standard deviation | Best | Worst |
|---|---|---|---|---|---|
| IDMOEA | **0.1071** | **0.0659** | 0.1323 | 0.0021 | 0.6038 |
| NSGA-II | 0.1112 | 0.0787 | **0.1302** | **0.0013** | **0.4303** |

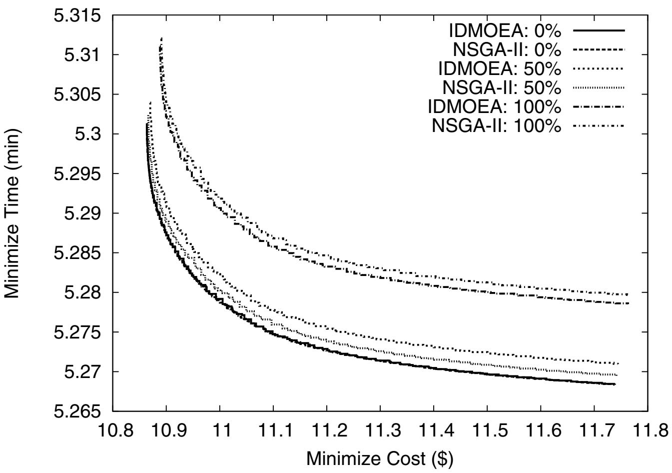


Fig. 4. 0%, 50% and 100% attainment plots of MOEAs for milling parameter optimization.

I) are less significant. Therefore, statistical analysis using $I_{R2}$ and $I_{\bar{H}}$ indicators are shown in Tables II and III. Mean and median values of IDMOEA for both indicators are better than NSGA-II over 30 different runs. Values of standard deviation and best of both the indicators show similar performance of IDMOEA as NSGA-II. But, NSGA-II shows better worst values of indicators than IDMOEA.

For visual assessment, 0%, 50% and 100% attainment surface plots are drawn in Fig. 4. 0% attainment plot shows equivalent performance of IDMOEA against NSGA-II. However, IDMOEA is little worse in 50% attainment plot but better in 100% attainment plots against NSGA-II.

### B. Two-Bar Truss Design

In this problem, the truss has to carry a certain load (100 kN) without elastic failure as shown in Fig. 5. Two-bar truss design problem is designed to minimize the volume and stresses, simultaneously. This is a three-variable, two objectives truss design problem [15] which is as follows:

$$\begin{aligned}
&\text{Minimize:} && f_1(x,y) = x_1\sqrt{16+y^2} + x_2\sqrt{1+y^2}, \\
&&& \qquad\qquad (\text{Total volume, } m^2) \\
&\text{Minimize:} && f_2(x,y) = max(\sigma_{AC}, \sigma_{BC}) \\
&&& \qquad\qquad (\text{Maximum stress, } MPa) \\
&\text{Subjected to:} && \\
&&& max(\sigma_{AC}, \sigma_{BC}) \leq 10^5 \text{ (Stress constraint)} \\
&&& 1 \leq y \leq 3 \text{ and } 0 \leq x_1, x_2 \leq 0.01 \\
&\text{where} && \sigma_{AC} = \frac{20\sqrt{16+y^2}}{yx_1},\ \sigma_{BC} = \frac{80\sqrt{1+y^2}}{yx_2}
\end{aligned} \tag{4}$$

Table IV shows similar outcome as observed in last section III-A. Values of mean, median and standard deviation of generation data is better for IDMOEA against NSGA-II. One difference can be seen in the worst value of generation data for IDMOEA which requires less generations than NSGA-II.

TABLE IV. NUMBER OF GENERATIONS REQUIRED FOR TERMINATING MOEAS.

| MOEAs ↓ | Mean | Median | Standard deviation | Best | Worst |
|---|---|---|---|---|---|
| IDMOEA | **340.3** | **351** | **79.3** | 200 | **502** |
| NSGA-II | 408 | 390 | 131.79 | **158** | 685 |

TABLE V. STATISTICAL VALUES OF $I_{R2}$ INDICATOR FOR TWO-BAR TRUSS DESIGN. THE SUGGESTED RANGE OF $I_{R2}$ LIES BETWEEN $-1$ AND $+1$, WHERE $-1$ IS THE BEST AND $+1$ IS THE WORST.

| MOEAs ↓ | Mean | Median | Standard deviation | Best | Worst |
|---|---|---|---|---|---|
| IDMOEA | 0.0019 | 0.0019 | **0.0001** | **0.0016** | **0.0021** |
| NSGA-II | **0.0018** | **0.0018** | **0.0001** | **0.0016** | **0.0021** |

TABLE VI. STATISTICAL VALUES OF $I_{\bar{H}}$ INDICATOR FOR TWO-BAR TRUSS DESIGN. THE SUGGESTED RANGE OF $I_{\bar{H}}$ LIES BETWEEN $-1$ AND $+1$, WHERE $-1$ IS THE BEST AND $+1$ IS THE WORST.

| MOEAs ↓ | Mean | Median | Standard deviation | Best | Worst |
|---|---|---|---|---|---|
| IDMOEA | 0.0065 | 0.0065 | **0.0003** | **0.0059** | **0.0072** |
| NSGA-II | **0.0064** | **0.0064** | **0.0003** | **0.0059** | **0.0072** |

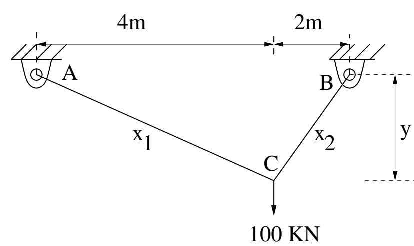


Fig. 5. A two-bar truss.

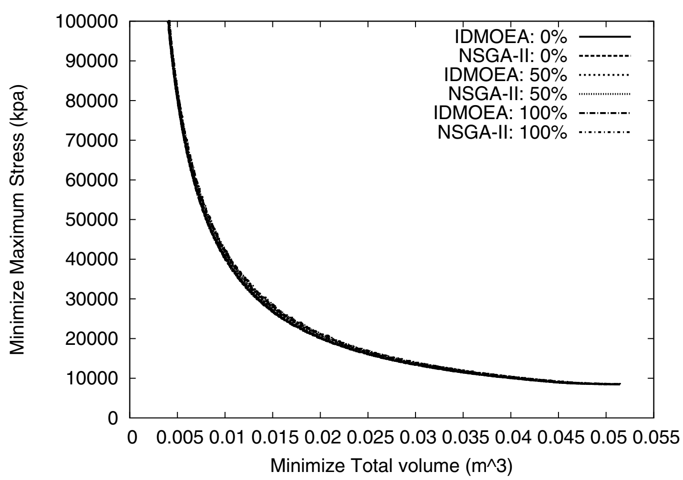


Fig. 6. 0%, 50% and 100% attainment plots of MOEAs for two-bar truss design problem.

The statistical values of $I_{R2}$ and $I_{\bar{H}}$ for both MOEAs are very close and mostly a difference can be seen at fourth decimal place. By looking at 0%, 50% and 100% attainment plots in Fig. 6, an equivalent performance of IDMOEA is observed against NSGA-II.

### C. Gear Train Design

A compound gear train is designed to achieve a specific gear ratio between the driver and driven shafts as shown in

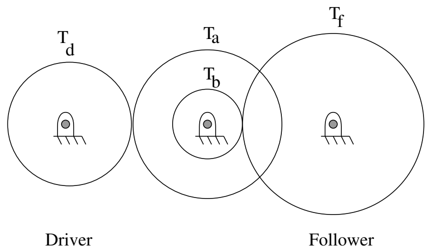


Fig. 7. A compound gear train.

Fig 7. Number of teeth on four gears must be integers which are represented as $(x_1, x_2, x_3, x_4) = (T_d, T_b, T_a, T_f)$. The optimization problem [15] can be written as:

$$
\begin{aligned}
&\text{Minimize:} && \left(\frac{1}{6.931} - \frac{x_1 x_2}{x_3 x_4}\right) && \text{(Gear ratio error)},\\
&\text{Minimize:} && max(x_1, x_2, x_3, x_4) && \text{(Maximum gear size)},\\
&\text{Subjected to:} && && \\
& && 12 \leq x_1, x_2, x_3, x_4 \leq 60, && \\
& && \text{all } x_i \text{are integers.} &&
\end{aligned} \tag{5}
$$

Table VII shows generation data in which average generations required to terminated MOEAs are same. A small difference in the values of median, standard deviation and worst is observed. Values of $I_{R2}$ and $I_{\bar{H}}$ indicators in Tables VIII and IX are also same or very close to each other. An equivalent performance of IDMOEA against NSGA-II can be seen in 0%, 50% and 100% attainment plots in Fig. 8. As variables in this problem are integers, the proposed idea shows identical performance with NSGA-II.

TABLE VII. NUMBER OF GENERATIONS REQUIRED FOR TERMINATING MOEAS.

| MOEAs ↓ | Mean | Median | Standard deviation | Best | Worst |
|---|---|---|---|---|---|
| IDMOEA | **185.3** | **169.5** | 66.7 | **73** | 376 |
| NSGA-II | **185.3** | 177 | **54.9** | 104 | **342** |

TABLE VIII. STATISTICAL VALUES OF $I_{R2}$ INDICATOR FOR GEAR TRAIN DESIGN. THE SUGGESTED RANGE OF $I_{R2}$ LIES BETWEEN $-1$ AND $+1$, WHERE $-1$ IS THE BEST AND $+1$ IS THE WORST. AS WORST.

| MOEAs ↓ | Mean | Median | Standard deviation | Best | Worst |
|---|---|---|---|---|---|
| IDMOEA | **0.0000** | **0.0000** | **0.0000** | **0.0000** | 0.0002 |
| NSGA-II | **0.0000** | **0.0000** | **0.0000** | **0.0000** | **0.0001** |

TABLE IX. STATISTICAL VALUES OF $I_{\bar{H}}$ INDICATOR FOR GEAR TRAIN DESIGN. THE SUGGESTED RANGE OF $I_{\bar{H}}$ LIES BETWEEN $-1$ AND $+1$, WHERE $-1$ IS THE BEST AND $+1$ IS THE WORST.

| MOEAs ↓ | Mean | Median | Standard deviation | Best | Worst |
|---|---|---|---|---|---|
| IDMOEA | **0.0011** | **0.0011** | 0.0007 | 0.0004 | 0.0037 |
| NSGA-II | **0.0011** | **0.0011** | **0.0005** | **0.0003** | **0.0024** |

### D. Welded Beam Design

In this problem, a beam which is welded on another beam must carry a certain load $F$ as shown in Fig. 9. The

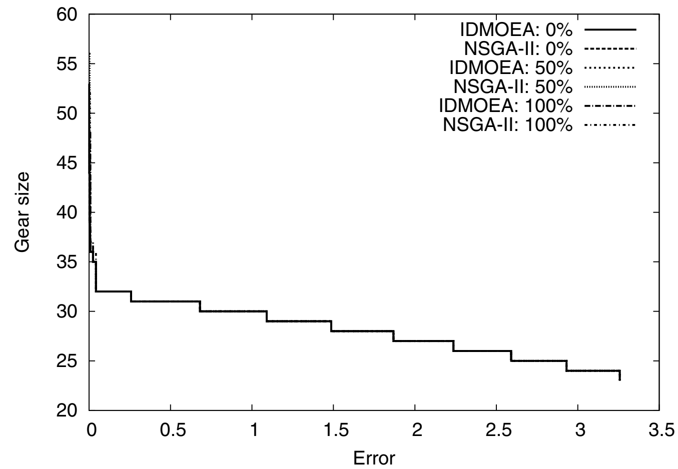


Fig. 8. 0%, 50% and 100% attainment plots of MOEAs for gear train design problem.

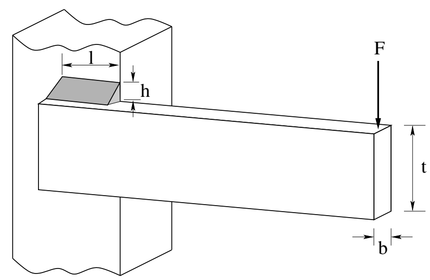


Fig. 9. Welded beam design problem.

optimization problem can be written as:

$$
\begin{aligned}
&\text{Minimize:} && 1.10471h^l + 0.04811tb(14.0 + l), \quad \text{(cost of the beam)},\\
&\text{Minimize:} && 2.1952/t^3 b \quad \text{(vertical deflection)},\\
&\text{Subjected to:} && \\
& && 13,600 - \tau(\vec{x}) \geq 0\\
& && 30,000 - \sigma(\vec{x}) \geq 0\\
& && b - h \geq 0\\
& && P_c(\vec{x}) - 6000 \geq 0\\
& && 0.125 \leq h, b \leq 5 \text{ and } 0.1 \leq l, t \leq 10\\
&\text{where} && \tau(\vec{x}) = \sqrt{\frac{(\tau')^2 + (\tau'')^2 + (l\tau'\tau'')}{\sqrt{0.25(l^2 + (h+t)^2)}}}\\
& && \tau' = 6,000/\sqrt{2}hl\\
& && \tau'' = \frac{6,000(14 + 0.5l)\sqrt{0.25(l^2 + (h+t)^2)}}{2\{0.707hl(l^2/12 + 0.25(h+t)^2)\}},\\
& && \sigma(\vec{x}) = 504,000/t^2 b,\\
& && P_c(\vec{x}) = 64,746.022(1 - 0.0282346t)tb^3
\end{aligned} \tag{6}
$$

There are four design parameters (thickness of the beam, $b$, width of the beam $t$, length of weld, $l$, and weld thickness $h$) for which the cost of the beam is minimum and simultaneously the vertical deflection at the end of the beam is minimum. The overhang portion of the beam has a length of 14in and $F = 6,000$ lb force is applied at the end of the beam.

For welded beam design problem, better values of mean, median, standard deviation and worst number of generations for terminating IDMOEA are observed in Table X. A marginal

TABLE X. NUMBER OF GENERATIONS REQUIRED FOR TERMINATING MOEAS.

| MOEAs ↓ | Mean | Median | Standard deviation | Best | Worst |
|---|---|---|---|---|---|
| IDMOEA | **475.5** | **442.5** | **195.1** | 206 | **905** |
| NSGA-II | 505.8 | 455 | 230.7 | **126** | 1000 |

TABLE XI. STATISTICAL VALUES OF $I_{R2}$ INDICATOR FOR WELDED BEAM DESIGN. THE SUGGESTED RANGE OF $I_{R2}$ LIES BETWEEN $-1$ AND $+1$, WHERE $-1$ IS THE BEST AND $+1$ IS THE WORST.

| MOEAs ↓ | Mean | Median | Standard deviation | Best | Worst |
|---|---|---|---|---|---|
| IDMOEA | 0.0031 | **0.0021** | 0.0036 | **0.0012** | 0.0164 |
| NSGA-II | **0.0030** | **0.0021** | **0.0029** | **0.0012** | **0.0140** |

TABLE XII. STATISTICAL VALUES OF $I_{\bar{H}}$ INDICATOR FOR WELDED BEAM DESIGN. THE SUGGESTED RANGE OF $I_{\bar{H}}$ LIES BETWEEN $-1$ AND $+1$, WHERE $-1$ IS THE BEST AND $+1$ IS THE WORST.

| MOEAs ↓ | Mean | Median | Standard deviation | Best | Worst |
|---|---|---|---|---|---|
| IDMOEA | 0.0242 | 0.0217 | 0.0196 | **0.0033** | 0.0846 |
| NSGA-II | **0.0222** | **0.0192** | **0.0178** | 0.0035 | **0.0772** |

difference can be observed in the statistical values of $I_{R2}$ and $I_{\bar{H}}$ indicators from Tables XI and XII. Visually also, the performance of IDMOEA is similar to NSGA-II as shown in 0%, 50% and 100% attainment plots in Fig. 10.

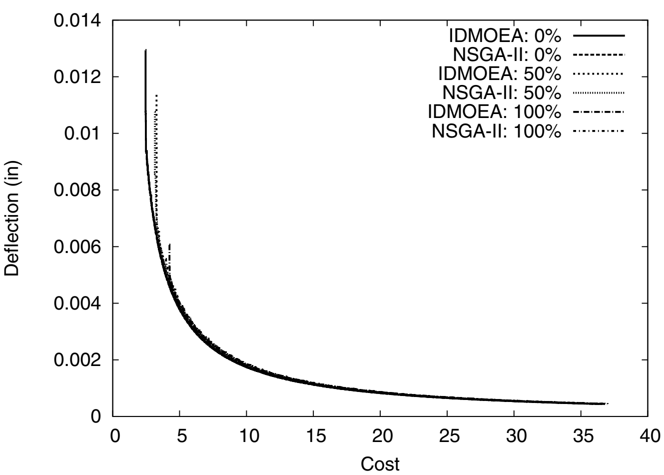


Fig. 10. 0%, 50% and 100% attainment plots of MOEAs for welded beam design problem.

## IV. DISCUSSION

The proposed approach was coupled with NSGA-II and tested on four engineering optimization problems. The aim was to obtain the results quickly without compromising much in the quality of solutions. Number of generations required for terminating MOEAs showed that on an average IDMOEA required less number of generations than NSGA-II over 30 different runs. Median of generation data for IDMOEA was also better than NSGA-II. Mixed results were observed for standard deviation, best and worst values of generation data on four engineering optimization problem. Regarding the quality of solutions evolved from MOEAs, statistical values of $I_{R2}$ and $I_{\bar{H}}$, and attainment plots showed equivalent performance of IDMOEA with NSGA-II. However, 100% attainment plot of IDMOEA in milling parameter optimization problem showed better performance than NSGA-II. One more interesting thing was also observed when IDMOEA was used for gear train design problem. In this problem, decision variables were integers due to which performance of IDMOEA was identical with NSGA-II. It may be concluded that the proposed approach was more favorable for real parameters than integers.

## V. CONCLUSIONS

In this paper, infeasibility driven approach was adopted in which extreme solutions of current non-dominated front were allowed to recombine only with extreme infeasible solutions. The approach was developed to use binary tournament selection and any crossover operator that can generate offspring solutions near to parents. The generic approach proposed in this paper was coupled with NSGA-II (IDMOEA) and tested on four engineering optimization problem. IDMOEA's performance was assessed using indicators and attainment plots against NSGA-II. On an average, less number of generations was required by IDMOEA to perform equivalently as NSGA-II. Therefore, the proposed approach was successful in quicker convergence than NSGA-II where some function evaluations can be saved by reducing number of required generations. In the future work, the idea can be enhanced for more number of objectives so that function evaluations can be reduced by reducing the number of generations. Moreover, the proposed approach can be tested with other existing MOEAs in the literature.

## APPENDIX A
## NOMENCLATURE AND VALUES OF CONSTANT PARAMETER OF MILLING FORMULATION

$a, a_{rad}$: axial depth of cut, radial depth of cut (mm); $C = 33.98, 100.45$: constant in cutting speed for HSS tools and carbide tools, respectively; $ca$: clearance angle; $c_l = 0.45, c_o = 1.45$: labour and over-head cost (\$/min); $c_m, c_{mat} = 0.50, c_t$: costs of machining, material per part and cutting tool (\$); $d$: cutter diameter (mm); $e = 95\%$: machine tool efficiency factor; $F$: feed rate (mm/min); $f$: feed rate (mm/tooth); $F_C, F_C(per)$: cutting force and permitted cutting force (N); $g = 0.14$: exponent of slenderness ratio; $K$: distance to be traveled by the tool to perform the operation (mm); $K_p = 2.24$: power constant depending on the workpiece material; $la$: lead (corner) angle of tool; $m = 5$: number of machining operation required to produce the product; $n = 0.15, 0.3$: tool life exponent for HSS tools and carbide tools, respectively; $P_m = 8.5$: motor power (kW); $Q$: contact proportion of cutting edge with the workpiece per revolution; $R_a, R_{a(at)}$: arithmetic value of surface finish, and attainable surface finish ($\mu$m); $t_m, t_s = 2, t_{tc} = 0.5$: machining time, set-up time, tool changing time (min); $V$: cutting speed; $w = 0.28$: exponent of chip cross-sectional area; $W = 1.1$: tool wear factor; $z$: number of cutting teeth of the tool.

Constants used for formulating unit cost: $C_1 = c_{mat} + (c_l + c_0)t_s$, $C_{2i} = (c_l + c_0)K_{1i}$, $K_{1i} = \pi d_i K_i / 1000 V_i f_i z_i$, $C_{3i} = c_{ti} K_{31}$, $K_{3i} = K_{1i}/K_{2i}$, $K_{2i} = 60 Q_i^{-1} C_i^{1/n_i} 5^{-g/n_i} a_i^{(g-w)/n_i}$, $C_{4i} = (c_l + c_o)t_{tci}$.

Variable bounds: $60 \leq V_1 \leq 120$ (face milling), $40 \leq V_2 \leq 70$ (corner milling), $40 \leq V_3 \leq 70$ (pocket milling), $30 \leq V_4 \leq 50$ (slot milling 1), $30 \leq V_5 \leq 50$ (slot milling 2); $0.05 \leq f_1 \leq 0.4$ (face milling), $0.05 \leq f_2 \leq 0.5$ (corner milling), $0.05 \leq f_3 \leq 0.5$ (pocket milling), $0.05 \leq f_4 \leq 0.5$ (slot milling 1), $0.05 \leq f_5 \leq 0.5$ (slot milling 2).

TABLE XIII. REQUIRED DATA FOR MACHINING

| Oper. no. | Oper. no. | Tool no. | $a$ | $K$ | $R_a$ | Face for surface roughness | $Q$ | $a_{rad}$ |
|---|---|---|---|---|---|---|---|---|
| 1 | Face milling | 1 | 10 | 450 | 2 | bottom | 0.45 | 50 |
| 2 | corner milling | 2 | 5 | 90 | 6 | bottom | 1.0 | 10 |
| 3 | pocket milling | 2 | 10 | 450 | 5 | bottom | 0.5 | 10 |
| 4 | slot milling | 3 | 10 | 32 | - | - | 1.0 | 12 |
| 5 | slot milling | 3 | 5 | 84 | 1 | side | 1.0 | 12 |

TABLE XIV. TOOLS DATA

| Tool No. | Tool type | Quality | $d$ | $CL$ | $z$ | Price | $SD$ | Helx angle | $la$ | $ca$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | face mill | Carbide | 50 | 20 | 6 | 49.50 | 25 | 15 | 45 | 5 |
| 2 | end mill | HSS | 10 | 35 | 4 | 7.55 | 10 | 45 | 0 | 5 |
| 3 | end mill | HSS | 12 | 40 | 4 | 7.55 | 10 | 45 | 0 | 5 |

## REFERENCES


[1] C. A. C. Coello, "Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: a survey of the state of the art," *Computer Methods in Applied Mechanics and Engineering*, vol. 191, no. 11 - 12, pp. 1245 – 1287, 2002. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0045782501003231

[2] M. Schoenauer and Z. Michalewicz, "Evolutionary computation at the edge of feasibility," in *Proceedings of the 4th International Conference on Parallel Problem Solving from Nature*, ser. PPSN IV. London, UK, UK: Springer-Verlag, 1996, pp. 245–254. [Online]. Available: http://dl.acm.org/citation.cfm?id=645823.670543

[3] ——, "Sphere operators and their applicability for constrained parameter optimization problems," in *Proceedings of the 7th International Conference on Evolutionary Programming VII*, ser. EP '98. London, UK, UK: Springer-Verlag, 1998, pp. 241–250. [Online]. Available: http://dl.acm.org/citation.cfm?id=647902.738966

[4] E. Mezura-Montes and C. A. C. Coello, "Useful infeasible solutions in engineering optimization with evolutionary algorithms," in *MICAI 2005: Advances in Artificial Intelligence*, ser. Lecture Notes in Computer Science, A. Gelbukh, A. d. Albornoz, and H. Terashima, Eds. Springer Berlin Heidelberg, 2005, vol. 3789, pp. 652–662. [Online]. Available: http://dx.doi.org/10.1007/11579427_66

[5] ——, "A simple multimembered evolution strategy to solve constrained optimization problems," *Evolutionary Computation, IEEE Transactions on*, vol. 9, no. 1, pp. 1 – 17, feb. 2005.

[6] T. Runarsson and X. Yao, "Stochastic ranking for constrained evolutionary optimization," *Evolutionary Computation, IEEE Transactions on*, vol. 4, no. 3, pp. 284 –294, sep 2000.

[7] Z. Cai and Y. Wang, "A multiobjective optimization-based evolutionary algorithm for constrained optimization," *Evolutionary Computation, IEEE Transactions on*, vol. 10, no. 6, pp. 658 –675, dec. 2006.

[8] T. Ray, k. Tai, and K. C. Seow, "Multiobjective design optimization by an evolutionary algorithm," *Engineering Optimization*, vol. 33, no. 4, pp. 399–424, 2001. [Online]. Available: http://www.tandfonline.com/doi/abs/10.1080/03052150108940926

[9] S. Venkatraman and G. Yen, "A generic framework for constrained optimization using genetic algorithms," *Evolutionary Computation, IEEE Transactions on*, vol. 9, no. 4, pp. 424 – 435, aug. 2005.

[10] A. Isaacs, T. Ray, and W. Smith, "Blessings of maintaining infeasible solutions for constrained multi-objective optimization problems," in *Evolutionary Computation, 2008. CEC 2008. (IEEE World Congress on Computational Intelligence). IEEE Congress on*. IEEE Press, june 2008, pp. 2780 –2787.

[11] H. Singh, A. Isaacs, T. Ray, and W. Smith, "Infeasibility driven evolutionary algorithm (IDEA) for engineering design optimization," in *AI 2008: Advances in Artificial Intelligence*, ser. Lecture Notes in Computer Science, W. Wobcke and M. Zhang, Eds. Springer Berlin Heidelberg, 2008, vol. 5360, pp. 104–115. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-89378-3_11

[12] Y. Woldesenbet, G. Yen, and B. Tessema, "Constraint handling in multiobjective evolutionary optimization," *Evolutionary Computation, IEEE Transactions on*, vol. 13, no. 3, pp. 514–525, 2009.

[13] E. Mezura-Montes and C. A. C. Coello, "Adding a diversity mechanism to a simple evolution strategy to solve constrained optimization problems," in *Evolutionary Computation, 2003. CEC '03. The 2003 Congress on*, vol. 1, dec. 2003, pp. 6 – 13 Vol.1.

[14] K. Deb, K. Miettinen, and S. Chaudhuri, "Toward an estimation of nadir objective vector using a hybrid of evolutionary and local search approaches," *Evolutionary Computation, IEEE Transactions on*, vol. 14, no. 6, pp. 821 –841, dec. 2010.

[15] K. Deb, *Multi-Objective Optimization using Evolutionary Algorithms*, 1st ed. Chichester, UK: Wiley, 2001.

[16] K. Deb and R. B. Agrawal, "Simulated binary crossover for continuous search space," *Complex Systems*, vol. 9, no. 2, pp. 115–148, 1995.

[17] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II," *Evolutionary Compuation, IEEE Transactions on*, vol. 6, no. 2, pp. 182 –197, apr 2002.

[18] M. P. Hansen and A. Jaszkiewicz, "Evaluating the Quality of Approximations to the Non-Dominated Set," Technical University of Denmark, IMM-REP-1998-7, 1998.

[19] J. Knowles, L. Thiele, and E. Zitzler, "A Tutorial on the Performance Assessment of Stochastic Multiobjective Optimizers," Computer Engineering and Networks Laboratory (TIK), ETH Zurich, TIK Report 214, Feb. 2006.

[20] E. Zitzler and L. Thiele, "Multiobjective Evolutionary Algorithms: A Comparative Case Study and the Strength Pareto Approach," *Evolutionary Computation, IEEE Transactions on*, vol. 3, no. 4, pp. 257–271, 1999.

[21] C. Fonseca, V. Fonseca, and L. Paquete, "Exploring the performance of stochastic multiobjective optimisers with the second-order attainment function," in *Evolutionary Multi-Criterion Optimization*, ser. Lecture Notes in Computer Science, C. Coello Coello, A. Hernndez Aguirre, and E. Zitzler, Eds. Springer Berlin Heidelberg, 2005, vol. 3410, pp. 250–264. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-31880-4_18

[22] M. Tolouei-Rad and I. Bidhendi, "On the optimization of machining parameters for milling operations," *International Journal of Machine Tools and Manufacture*, vol. 37, no. 1, pp. 1 – 16, 1997. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0890695596000442