

A New Threshold Based Penalty Function Embedded MOEA/D

Muhammad Asif Jan
Department of Mathematics
Kohat University of Science & Technology
Khyber Pakhtunkhwa, Pakistan

Nasser Mansoor Tairan
College of Computer Science
King Khalid University Abha,
Saudi Arabia

Rashida Adeeb Khanum
Jinnah College for Women
University of Peshawar Khyber
Pakhtunkhwa, Pakistan

Wali Khan Mashwani
Department of Mathematics
Kohat University of Science & Technology
Khyber Pakhtunkhwa, Pakistan

Abstract—Recently, we proposed a new threshold based penalty function. The threshold dynamically controls the penalty to infeasible solutions. This paper implants the two different forms of the proposed penalty function in the multiobjective evolutionary algorithm based on decomposition (MOEA/D) framework to solve constrained multiobjective optimization problems. This led to a new algorithm, denoted by CMOEA/D-DE-ATP. The performance of CMOEA/D-DE-ATP is tested on hard CEC-series test instances in terms of the values of IGD-metric and SC-metric. The experimental results are compared with the three best performers of CEC 2009 MOEA competition. Experimental results show that the proposed penalty function is very promising, and it works well in the MOEA/D framework.

Keywords—Constrained multiobjective optimization; decomposition; MOEA/D; penalty function; threshold.

I. INTRODUCTION

In this paper, we consider the following constrained multiobjective optimization problem (CMOP) [1]:

$$\begin{aligned} \text{Minimize} \quad & F(x) = (f_1(x), f_2(x), \dots, f_m(x))^T; \\ \text{Subject to} \quad & g_j(x) \geq 0, j = 1, \dots, p; \\ & l_k \leq x_k \leq u_k, k = 1, \dots, n, \end{aligned} \quad (1)$$

where $x = (x_1, \dots, x_n)^T \in \mathcal{R}^n$ is an n dimensional vector of decision variables, F is the objective vector function that consists of m real-valued objective functions, and $g_i(x) \geq 0$ are inequality constraints. The objective and constraint functions, f_i 's and g_j 's, could be linear or non linear real-valued functions. l_k and u_k are the lower and upper bounds (called bound constraints) of x_k , $k = 1, \dots, n$, respectively, which define the search region $\mathcal{S} = \{x = (x_1, \dots, x_n)^T \mid l_k \leq x_k \leq u_k, k = 1, \dots, n\}$.

A solution $x \in \mathcal{S}$ is called a feasible solution, if it satisfies all the inequality constraints in (1). The set of all feasible solutions is called the feasible region. Mathematically, we can write:

$$\mathcal{F} = \{x \in \mathcal{S} \subset \mathcal{R}^n \mid g_j(x) \geq 0, j = 1, \dots, p\}.$$

However, If a solution is not feasible, we call it infeasible. The set of all infeasible solutions is called the infeasible region.

The feasible attainable objective set (AOS) can be defined as $\{F(x) \mid x \in \mathcal{F}\}$.

Since the objectives in (1) more often contradict each other, so it is hard to find a single solution in \mathcal{F} that could minimize all the objectives at the same time. Instead, one looks for a set of optimal compromising/tradeoff feasible solutions. The best tradeoffs among the objectives can be defined in terms of Pareto-optimality [2], [3].

A solution x is said to Pareto-dominate or simply dominate another solution y , mathematically denoted as $x \preceq y$, if $f_i(x) \leq f_i(y)$, $\forall i = 1, \dots, m$ and $f_j(x) < f_j(y)$ for at least one $j \in \{1, \dots, m\}$ ¹. This definition of domination is sometimes referred to as a weak dominance relation.

A solution $x^* \in \mathcal{F}$ is Pareto-optimal to (1) if there is no solution $x \in \mathcal{F}$ such that $F(x) \preceq F(x^*)$. $F(x^*)$ is then called a Pareto-optimal (objective) vector. The set of all Pareto-optimal solutions is called the Pareto Set (PS) in the decision space and Pareto Front (PF) in the objective space [2].

In the majority of constrained optimization problems, the optimal solutions lie on the constraints' boundaries. Thus, to arrive at these solutions, some algorithms evolve some good infeasible solutions with less constraint violation along with their feasible counterparts during the evolutionary process (e.g., see [4]–[6]). The primary purpose of evolving infeasible solutions in the search procedure is to utilize the information they transport. As EAs are stochastic search and optimization methods, rejecting infeasible individuals might lead the EA being stuck in local optima, particularly in problems with disconnected search space [7], [8]. Moreover, in some highly constrained optimization problems, it could be a demanding problem to find a single feasible solution [9], [10]. Therefore, constraint handling techniques used in multiobjective optimization (MOO) can be mainly distinguished by knowing how infeasible solutions are mixed up and evolved in the evolutionary process.

¹This definition of domination is for minimization. All the inequalities should be reversed if the goal is to maximize the objectives in (1). “dominate” means “be better than”.

In [1], we introduced a new threshold based penalty function in the replacement and update scheme of MOEA/D-DE [11], an improved version of MOEA/D [12], to penalize infeasible solutions. The threshold is adaptively adjusted by using the minimum and maximum constraint violation in the neighborhood of a solution. The infeasible solutions with constraint violation less than the threshold are less penalized than the ones with constraint violation greater than the threshold. As a result, we expect that some good infeasible solutions with less constraint violation will have a chance to evolve in the evolutionary process. The some preliminary experimental results, presented in [1], have proven the capability of the proposed algorithm for solving CF-series [13] test instances. In this paper, we present detailed experimental results and comment on the pitfalls of the proposed algorithm.

The rest of this paper is organized as follows. Section II presents some basic concepts and the two versions of the proposed penalty function. Section III briefly introduces MOEA/D and the modified algorithmic framework of MOEA/D-DE. Section IV discusses the experimental settings. Section V presents and discusses experimental results on CF-series [13] test instances. Section VI compares our experimental results with the three best performers [14]–[16] of CEC 2009 MOEA competition. Finally, Section VII concludes this paper with an outline of the work carried out.

II. BASIC CONCEPTS AND THE PROPOSED PENALTY FUNCTION

A. Degree of Constraint Violation

The degree of constraint violation of a solution $x \in S$ can be defined as [1], [3]:

$$V(x) = \left| \sum_{j=1}^p \min(g_j(x), 0) \right|. \quad (2)$$

Obviously, if $V(x) = 0$, x is feasible; otherwise, it is infeasible.

B. Tchebycheff Aggregation Function

MOEA/D [12] decomposes an MOP into a number of single objective subproblems. This paper uses the Tchebycheff aggregation function for this purpose, which is given as under [17]:

$$\begin{aligned} \text{Minimize } & g^{te}(x|\lambda, z^*) = \max_{1 \leq i \leq m} \{ \lambda_i |f_i(x) - z_i^*| \}; \\ \text{Subject to } & x \in \mathcal{F} \subset \mathcal{R}^n; \end{aligned} \quad (3)$$

where $z^* = (z_1^*, \dots, z_m^*)^T$ is the reference point, i.e., $z_i^* = \min\{f_i(x) | x \in \mathcal{F}\} \forall i = 1, \dots, m$ and $\lambda = (\lambda_1, \dots, \lambda_m)^T$ is a weight vector such that $\lambda_i \geq 0 \forall i = 1, \dots, m$ and $\sum_{i=1}^m \lambda_i = 1$. Some theorems related to the Pareto-optimality of Tchebycheff aggregation function can be found in [2].

C. The Proposed Penalty Function

The proposed penalty function uses a threshold value, τ for dynamically controlling the amount of penalty.

Suppose MOEA/D [12] decomposes the MOP into N subproblems. At each generation, MOEA/D retains N solutions x^1, \dots, x^N , where x^i is the current solution to subproblem i . Let P be the mating and update range set in MOEA/D. Then define [1]:

$$V_{min} = \min\{V(x^i), i \in P\}, \quad (4)$$

$$V_{max} = \max\{V(x^i), i \in P\}, \quad (5)$$

where $V(x^i)$ is the degree of constraint violation of solution x^i .

The threshold value, τ is defined as [1]:

$$\tau = V_{min} + s(V_{max} - V_{min}), \quad (6)$$

where the parameter s controls the threshold value. In [1], we used $s = 0.3$.

Our suggested penalty function encourages the algorithm to search the feasible region and the infeasible region near the feasible region. It is defined in the following two different ways: For $i = 1, \dots, m$

$$f_p^i(x) = \begin{cases} f_i(x) + s_1 V^2(x), & \text{if } V(x) < \tau; \\ f_i(x) + s_1 \tau^2 + s_2 (V(x) - \tau), & \text{otherwise,} \end{cases} \quad (7)$$

$$g_p^{te}(x|\lambda, z^*) = \begin{cases} g^{te}(x|\lambda, z^*) + s_1 V^2(x), & \text{if } V(x) < \tau; \\ g^{te}(x|\lambda, z^*) + s_1 \tau^2 + s_2 (V(x) - \tau), & \text{otherwise,} \end{cases} \quad (8)$$

where s_1 and s_2 are two scaling parameters with $s_1 \ll s_2$. In the penalty functions, the penalty increases sharply when $V(x)$ exceeds the threshold. This is realized by scaling the degree of constraint violation, $V(x)$ of an infeasible solution by relatively high value of parameter s_2 than parameter s_1 in our penalty function formulations. In Eq. 7, the penalty is added to individual objective function values of an infeasible solution, while in Eq. 8, it is added directly to Tchebycheff aggregation function value of an infeasible solution. Furthermore, in [1], we tested Eq. 8 only.

III. MULTIOBJECTIVE EVOLUTIONARY ALGORITHM BASED ON DECOMPOSITION

Zhang and Li [12] suggested a simple yet efficient MOEA, multiobjective evolutionary algorithm based on decomposition (MOEA/D). MOEA/D approximates the PF by explicitly decomposing an MOP into several single objective optimization subproblems. These subproblems are then optimized concurrently and collaboratively by evolving population of solutions. An EA is employed for this purpose. The Euclidean distances between the aggregation coefficient vectors of these subproblems are calculated to identify the neighborhood of each subproblem. The information gathered from the neighboring subproblems is then used to optimize a subproblem.

In this work, we employed the penalty functions defined by Eqs. 7, 8 in the update scheme of MOEA/D-DE [11], one

of the efficient versions of MOEA/D to solve CF-series [13] test instances. This resulted in a new algorithm, denoted by CMOEA/D-DE-ATP (For details of CMOEA/D-DE, please see [18]). The pseudo-code of the modified update scheme is given in Algorithm 1.

Algorithm 1 Pseudo-code of the update scheme of CMOEA/D-DE-ATP.

```

1: Each new child solution  $y$  updates  $n_r$  solutions from the set  $P$  of its neighboring solutions as follows:
2: Set  $c = 0$  and then do the following:
3: if  $c = n_r$  or  $P$  is empty then
4:   return;
5: else
6:   Randomly pick an index  $j$  from  $P$ ;
7:   Compute the Tchebycheff aggregation function values of  $y$  and  $x^j$  with the new objective values of Eq. 7 (or the new aggregation function values of  $y$  and  $x^j$  with Eq. 8);
8:   if  $g^{te}(y|\lambda^j, z) \leq g^{te}(x^j|\lambda^j, z)$  (or  $g_p^{te}(y|\lambda^j, z) \leq g_p^{te}(x^j|\lambda^j, z)$ ) then
9:      $x^j = y$ ,  $F(x^j) = F(y)$ ,  $V(x^j) = V(y)$ , and  $c = c + 1$ ;
10:  end if
11:  Remove  $j$  from  $P$  and go to step 3;
12: end if
```

IV. EXPERIMENTAL SETTINGS

In our experiments, we use the same parameters' settings and weight vectors' selection criteria as is used in [13]. Further, we use statistics of the inverted generational distance metric (IGD-metric) [12], [19] for comparing results on CF-series test instances, CF1-CF10. Also, the set coverage metric (SC-metric) [12] is used to compare the nondominated solutions obtained by different algorithms. Unless otherwise stated, we will use Eq. 6 with $s = 0.7$ and Eqs. 7, 8 with $s_1 = 0.01$ and $s_2 = 20$ in all experiments.

V. EXPERIMENTAL RESULTS

TABLE I: THE IGD-METRIC STATISTICS OF CMOEA/D-DE-ATP USING EQS. 7, 8. THE RESULTS IN **BOLDFACE** INDICATE THE BETTER RESULTS; IF NOT, THEY ARE IDENTICAL.

Test Instance	best (lowest)		mean		st. dev.	
	Eq. 7	Eq. 8	Eq. 7	Eq. 8	Eq. 7	Eq. 8
CF1	0.0003	0.0003	0.0006	0.0005	0.0003	0.0002
CF2	0.0028	0.0027	0.0037	0.0041	0.0013	0.0019
CF3	0.0632	0.0632	0.1382	0.1382	0.0441	0.0441
CF4	0.0060	0.0051	0.0097	0.0095	0.0042	0.0043
CF5	0.0406	0.0297	0.1606	0.1663	0.1084	0.1107
CF6	0.0049	0.0053	0.0197	0.0192	0.0141	0.0144
CF7	0.0344	0.0304	0.1188	0.1310	0.0729	0.0722
CF8	0.0332	0.0356	0.0370	0.0371	0.0020	0.0010
CF9	0.0428	0.0434	0.0468	0.0479	0.0022	0.0030
CF10	0.1068	0.1108	0.1509	0.1630	0.0396	0.0409

Table I presents the best (i.e., lowest), mean, and standard deviation of the IGD-metric values for CF-series test instances

found by CMOEA/D-DE-ATP with Eqs. 7, 8. These statistics are based on 30 independent runs. As it can be seen from this table that CMOEA/D-DE-ATP can find better best values with Eq. 7 for one 2-objective, CF6 and three 3-objective, CF8-CF10 and with Eq. 8 for four 2-objective, CF2, CF4, CF5, and CF7 test instances. The best values for test instance CF1 are identical. This table also shows that CMOEA/D-DE-ATP with both Eqs. 7, 8 performs similarly on test instance CF3. However, improved mean and st. dev. values can be found when CMOEA/D-DE-ATP employs Eq. 7 for most of the test instances. In particular, the improved performance can be seen for the three 3-objective test instances, CF8-CF10. This suggests that adding the penalty to individual objective function values as is done in Eq. 7 before calculating the aggregation function values is a good choice for better performance on CF-series test instances.

It can also be seen from Table I that CMOEA/D-DE-ATP with both Eqs. 7, 8 finds small values for the mean of IGD-metric on CF1, CF2, CF4, CF6, CF8, CF9. Empirically, these results illustrate that the final nondominated solutions found by CMOEA/D-DE-ATP for these test instances approximate the PF very well in a sense.

TABLE II: THE AVERAGE SET COVERAGE BETWEEN CMOEA/D-DE-ATP WITH EQ. 7 AND WITH EQ. 8 ON CF-SERIES TEST INSTANCES. THE RESULTS IN **BOLDFACE** INDICATE THE BETTER RESULTS; IF NOT, THEY ARE IDENTICAL.

Test Instance	C(Eq. 7, Eq. 8)	C(Eq. 8, Eq. 7)
CF1	0.46	0.48
CF2	0.13	0.13
CF3	0.65	0.65
CF4	0.26	0.28
CF5	0.19	0.22
CF6	0.17	0.21
CF7	0.24	0.24
CF8	0.05	0.04
CF9	0.03	0.03
CF10	0.39	0.31

Table II presents the average set coverage between the nondominated solutions of CMOEA/D-DE-ATP with Eq. 7 and Eq. 8. The results of this table reveal that, in terms of the SC-metric, the nondominated solutions found by CMOEA/D-DE-ATP with Eq. 8 are better than those obtained with Eq. 7 for test instances CF1, CF4-CF6, but are worse for test instances CF8 and CF10 vice versa. The table also shows that the nondominated solutions acquired from CMOEA/D-DE-ATP with both Eqs. 7, 8 are same for test instances CF2, CF3, CF7, and CF9. However, looking at the results of this table, it can be inferred that the performance of CMOEA/D-DE-ATP is comparable with both Eqs. 7, 8, as there is no big difference in the SC-metric values.

Figures 1 and 2 show, in the objective space, the distributions of the 100 and 150 nondominated population members for the seven 2-objective, CF1-CF7, and the three 3-objective, CF8-CF10, CF-series test instances. These solutions are selected based on the criteria as mentioned in [13] from the final population of the run with the best (i.e., lowest) IGD-metric value among the 30 independent runs. These figures also show

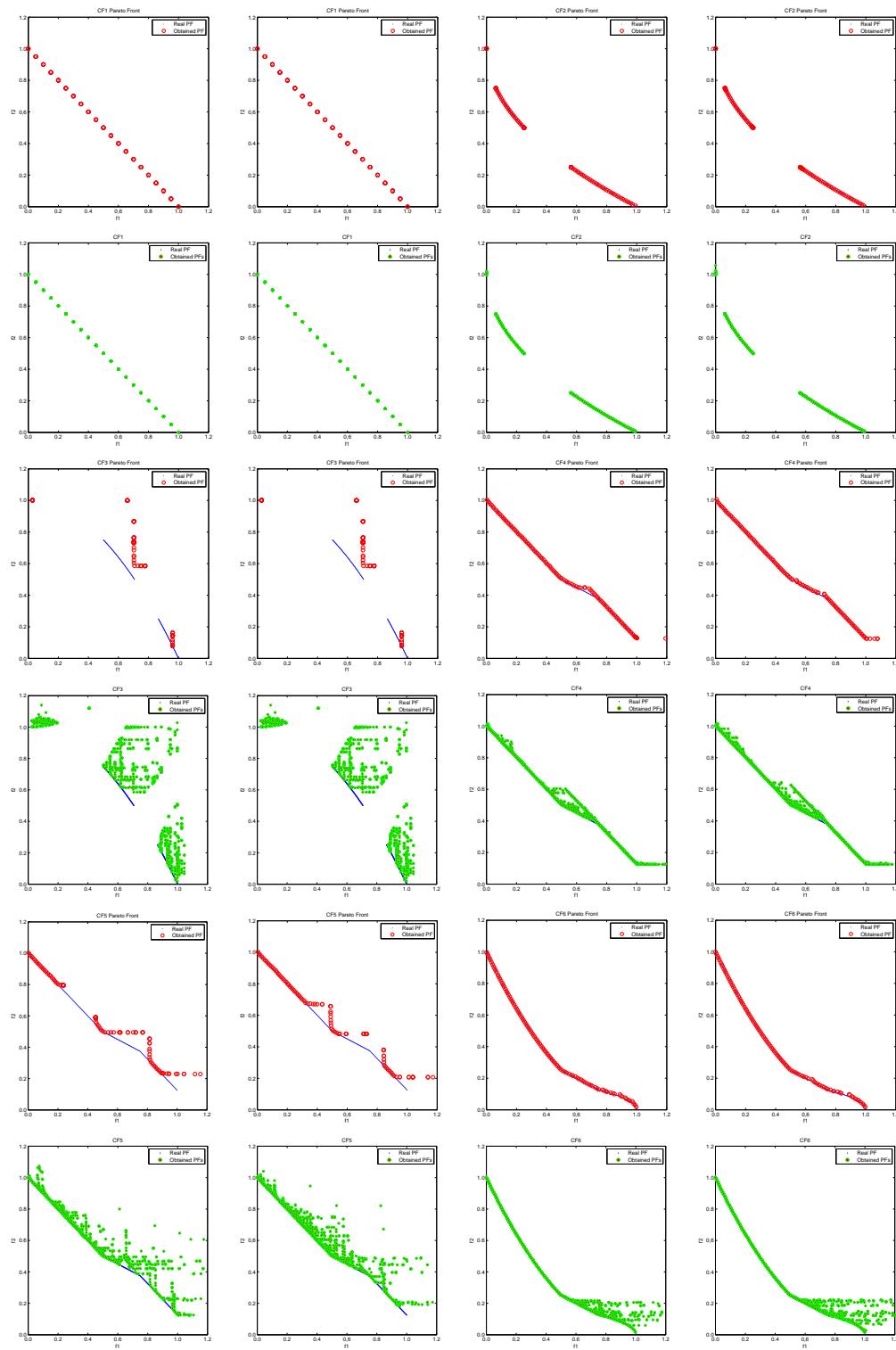


Fig. 1: Plots of the nondominated front with the best IGD value and all the 30 final nondominated fronts found by CMOEA/D-DE-ATP when using Eq. 7 (columns 1 and 3) and Eq. 8 (columns 2 and 4) for CF1-CF6.

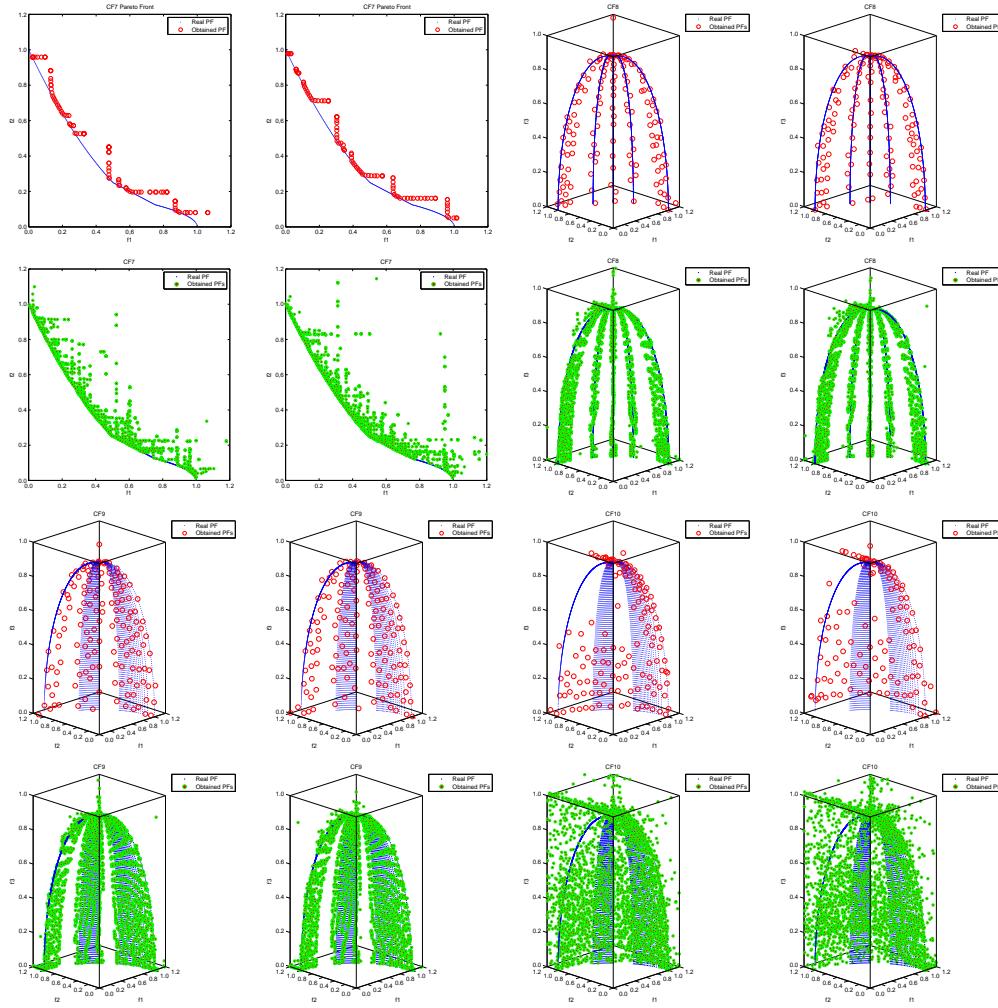


Fig. 2: Plots of the nondominated front with the best IGD value and all the 30 final nondominated fronts found by CMOEA/D-DE-ATP when using Eq. 7 (columns 1 and 3) and Eq. 8 (columns 2 and 4) for CF7-CF10.

all the 30 final nondominated fronts of these selected 100 and 150 nondominated solutions.

It is very clear from these figures that CMOEA/D-DE-ATP with both Eqs. 7, 8 found good approximations for the four 2-objective, CF1, CF2, CF4, and CF6, and two 3-objective, CF8, CF9 test instances. However, it performed poorly on test instances CF3, CF5, CF7, and CF10. It is also apparent from the plots of 30 nondominated fronts of test instances CF4 and CF6 that CMOEA/D-DE-ATP fails to find the whole PF in some runs.

The PF of CF3 is concave and discontinuous. Therefore, it could be hard for the algorithm than all other 2-objective test instances. Although the PFs of CF4 and CF5, CF6 and CF7, and CF9 and CF10 are identical, the poor performance of CMOEA/D-DE-ATP on test instances CF5, CF7 and CF10 could be due to the presence of harder objective and constraint functions in these test instances than test instances CF4, CF6, and CF9.

Figure 3 shows the evolution of the average IGD-metric values versus function evaluations of the nondominated so-

lutions in the current population. This figure shows that CMOEA/D-DE-ATP with both Eqs. 7, 8 converges at the same rate in terms of IGD-metric values for six CF-series test instances CF1, CF3, CF4, CF6, CF8 and CF9. However, it converges slightly faster in terms of IGD-metric values for the other four CF-series test instances CF2, CF5, CF7, and CF10 with Eq. 7 than with Eq. 8.

Figure 4 depicts the average generation feasibility versus generations' graphs. This figure demonstrates that CMOEA/D-DE-ATP with both Eqs. 7, 8 approaches to the feasible regions at the same rate for test instances CF1-CF6. It converges slower to the feasible regions with Eq. 7 than with Eq. 8 for the test instances CF7-CF10. This permits further exploration of the infeasible regions near the PF and could be one of the reasons for the better performance of CMOEA/D-DE-ATP with Eq. 7 on the three 3-objective test instances, CF8-CF10.

As it can be seen from Figure 4 that 50 % or more of the initial populations for test instances CF1, CF2, CF4 and CF5 are feasible. These feasible solutions are propagated in the subsequent generations by the replacement and update scheme

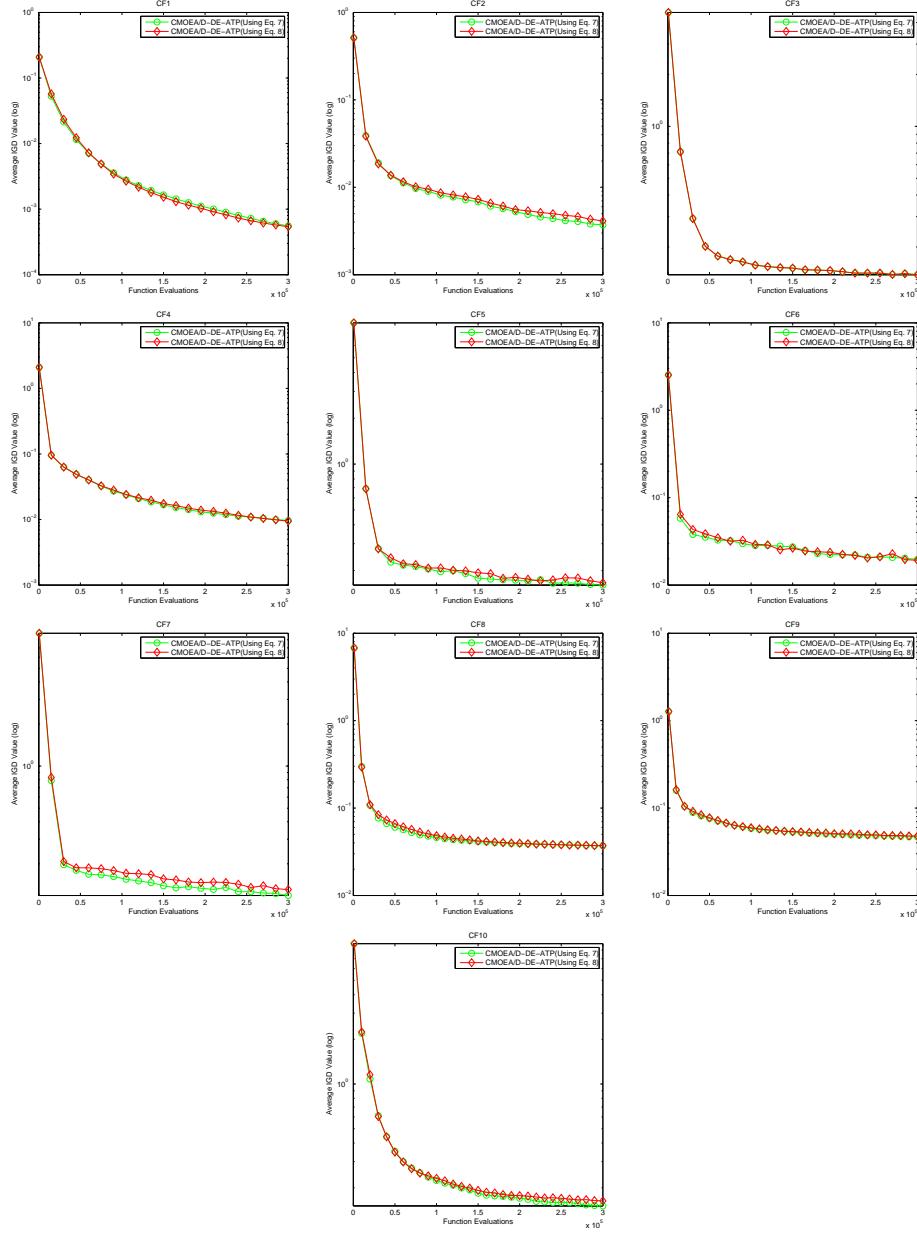


Fig. 3: Evolution of the IGD-metric values versus function evaluations when CMOEA/D-DE-ATP uses Eqs. 7, 8 for CF1-CF10.

of the algorithm and thus produce better feasible solutions due to the DE operator. Furthermore, the feasibility ratio becomes 1 after the initial 30 to 40 generations for these test instances. The reason for the quick convergence to the feasible region is the higher adopted update number of neighboring parent solutions (as in our settings $n_r = 6$ when $T = 60$ and $n_r = 10$ when $T = 100$) that are replaced by a better child solution in the update scheme of the algorithm. This speedy convergence to the feasible region is good for test instances like CF1, CF2, and CF4, but it causes problems for harder test instance like CF5. The PF of CF5 is a piecewise continuous curve with three pieces like CF4, but its objective and constraint functions are quite different and harder than CF4.

On the other hand, about 25 % or below of the initial

populations for test instances CF6-CF10 is feasible. Here, the proposed constraint handling technique has more chances to evolve better infeasible solutions during the evolutionary process. Particularly, in the two 3-objective test instances CF8 and CF9, the average feasibility ratio at the last generations of the algorithmic runs is 0.6 and about 0.9, respectively (see Figure 4). This way the infeasible regions near the feasibility boundaries in these two instances are well explored and could be a reason for the better performance of the algorithm on these two instances.

Moreover, in test instance CF6, the feasibility ratio of CMOEA/D-DE-ATP with both Eqs. 7, 8 becomes 1 after the initial 40 generations, while in test instance CF7, it takes 200 generations of CMOEA/D-DE-ATP with Eq. 7 and 50 gener-

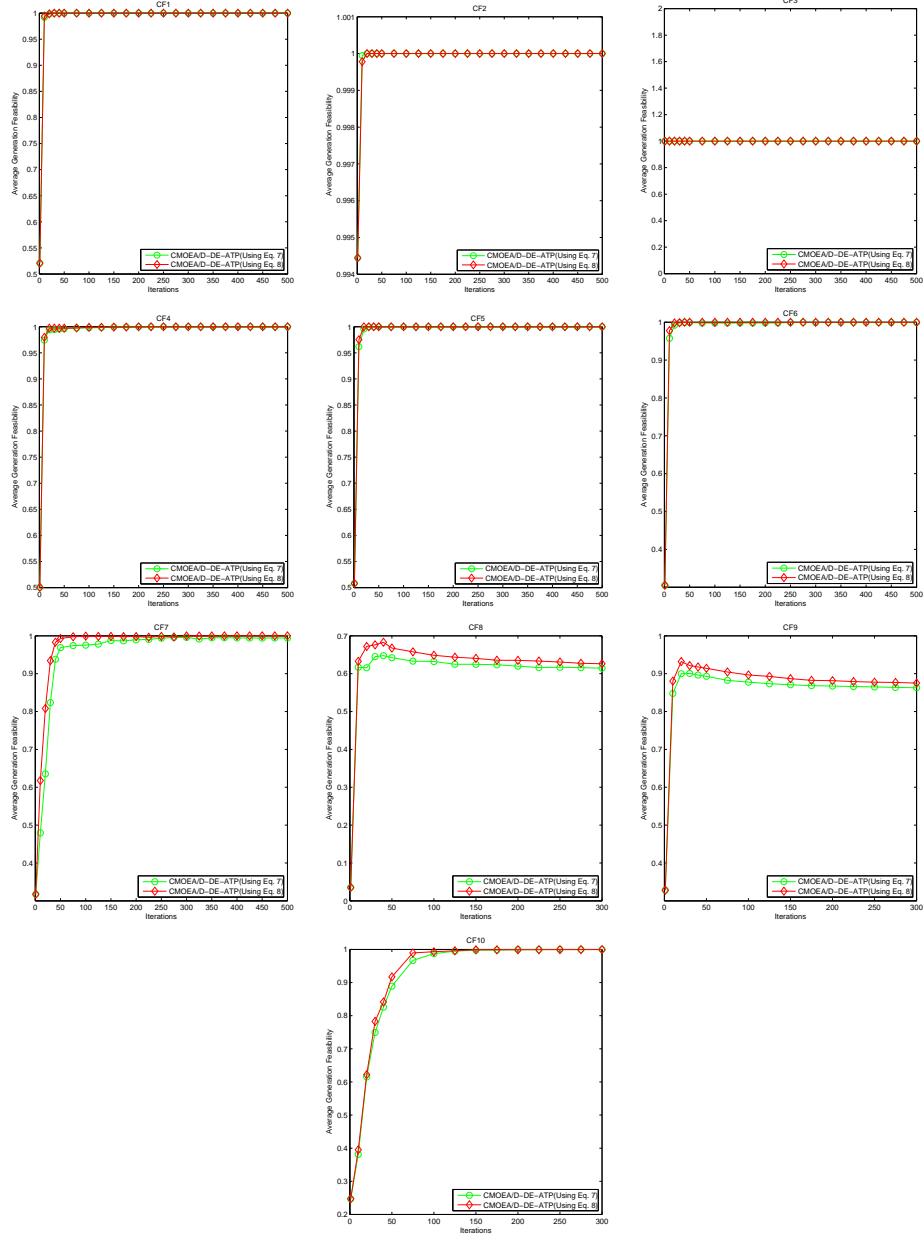


Fig. 4: Evolution of the generation feasibility versus generations when CMOEA/D-DE-ATP uses Eqs. 7, 8 for CF1-CF10.

ations with Eq. 8 to become 1. Thus, the quick convergence to the feasible region is good in case of CF6, but could be a reason for the poor performance of CMOEA/D-DE-ATP with both Eqs. 7, 8 in case of CF7.

In test instance CF10, the feasibility ratio takes about 140 generations of CMOEA/D-DE-ATP with Eq. 7 and about 80 generations of CMOEA/D-DE-ATP with Eq. 8 to become 1. Again, the less exploration of the infeasible regions could be the reason for the poor performance of CMOEA/D-DE-ATP on test instance CF10.

VI. COMPARISON WITH THE THREE BEST PERFORMERS OF CEC 2009 MOEA COMPETITION

In this section, we compare the results of CMOEA/D-DE-ATP with Eqs. 7, 8 with the three best performers [14]–[16] in CEC 2009 MOEA competition on the CF-series test instances.

Table III compares the best (i.e., lowest), mean, and standard deviation values of the IGD-metric obtained from our algorithm, CMOEA/D-DE-ATP with Eqs. 7, 8, and the three best performers [14]–[16] in CEC 2009 MOEA competition for the CF-series test instances. The table clearly shows that CMOEA/D-DE-ATP has found the best (i.e., lowest) IGD-metric values for four test instances CF1, CF6, CF8 and CF9 and the second best value for one test instance CF3 with Eq.

TABLE III: COMPARISON BETWEEN CMOEA/D-DE-ATP WITH EQ. 7 (INDICATED BY JZ1) AND WITH EQ. 8 (INDICATED BY JZ2), TSENG AND CHEN'S [14] (INDICATED BY TC), LIU AND LI'S [15] (INDICATED BY LL), AND LIU ET. AL'S [16] (INDICATED BY LI) ALGORITHMS IN TERMS OF THE IGD VALUES BASED ON 30 INDEPENDENT RUNS. THE RESULTS IN **BOLDFACE** AND IN *ITALIC* INDICATE THE BETTER AND THE SECOND BETTER RESULTS.

Test Instance	best (lowest)					mean					st. dev.				
	JZ1	JZ2	TC	LL	LI	JZ1	JZ2	TC	LL	LI	JZ1	JZ2	TC	LL	LI
CF1	0.0003	0.0003	0.0139	<i>0.0007</i>	0.0071	0.0006	0.0005	0.0192	0.0009	0.0113	0.0003	0.0002	0.0026	0.0001	0.0028
CF2	0.0028	0.0027	0.0041	0.0027	0.0016	0.0037	0.0041	0.0268	0.0042	0.0021	<i>0.0013</i>	0.0019	0.0147	0.0026	0.0005
CF3	0.0632	0.0632	0.0753	0.0908	0.0381	0.1382	0.1382	0.1045	0.1829	0.0563	0.0441	0.0441	0.0156	0.0421	0.0076
CF4	0.0060	0.0051	0.0089	0.0090	0.0055	0.0097	0.0095	0.0111	0.0142	0.0070	0.0042	0.0043	0.0014	0.0033	0.0015
CF5	0.0406	0.0297	0.0176	0.0588	0.0079	0.1606	0.1663	0.0208	0.1097	0.0158	0.1084	0.1107	0.0024	0.0307	0.0067
CF6	0.0049	0.0053	0.0096	0.0090	0.0062	0.0197	0.0192	0.0162	0.0139	0.0150	0.0141	0.0144	0.0060	0.0026	0.0065
CF7	0.0344	0.0304	0.0187	0.0535	0.0104	0.1188	0.1310	0.0247	0.1045	0.0191	0.0729	0.0722	0.0047	0.0351	0.0061
CF8	0.0332	0.0356	0.6220	0.0473	0.0388	0.0370	0.0371	1.0854	0.0607	0.0475	0.0020	0.0010	0.2191	0.0130	0.0064
CF9	0.0428	0.0434	0.0721	0.0460	0.1191	0.0468	0.0479	0.0851	0.0505	0.1434	0.0022	0.0030	0.0082	0.0034	0.0214
CF10	0.1068	0.1108	0.1173	0.1055	0.0984	0.1509	0.1630	0.1376	0.1974	0.1621	0.0396	0.0409	0.0092	0.0760	0.0316

7. It has also found the best IGD-metric values for two test instances CF1 and CF4 and the second best values for four test instances CF2, CF3, CF8, and CF9 with Eq. 8. Particularly, for test instances CF1, CF8 and CF9 better statistics are found by our algorithm except the standard deviation value on CF1 (although both our standard deviation values are very close to the best standard deviation value).

VII. CONCLUSIONS

A penalty function that penalizes infeasible solutions based on an adaptive threshold value has been introduced into the update and replacement scheme of MOEA/D-DE. This resulted in a new algorithm, CMOEA/D-DE-ATP for CMOO. The proposed penalty function is presented in two forms given by Eqs. 7, 8. The performance of CMOEA/D-DE-ATP is tested on CF-series test instances in terms of the values of IGD-metric and SC-metric.

From the experimental results in this paper, we can make the following conclusions.

- Overall, CMOEA/D-DE-ATP produced better results with the proposed penalty function defined by Eq. 7 than when it is defined by Eq. 8. That is, it is better to add the penalty to individual objective function values before calculating the aggregation function values than directly adding the penalty to aggregation function values of an infeasible solution for better performance achievement on CF-series test instances.
- The comparison of CMOEA/D-DE-ATP with the three best performers in CEC 2009 special session and competition indicated that CMOEA/D-DE-ATP has found the best (i.e., lowest) IGD-metric values for five test instances CF1, CF4, CF6, CF8 and CF9 and the second best values for two test instances CF2 and CF3. In particular, our algorithm overall found better statistics for tests instances CF1, CF8, and CF9.

REFERENCES

- [1] M. A. Jan and Q. Zhang, "MOEA/D for constrained multiobjective optimization: Some preliminary experimental results," in *UK Workshop on Computational Intelligence (UKCI)*. IEEE, 2010, pp. 1–6.
- [2] K. Miettinen, *Nonlinear Multiobjective Optimization*. Kluwer Academic Publishers, 1999.
- [3] K. Deb, *Multi-Objective Optimization using Evolutionary Algorithms*. John Wiley & Sons LTD, 2001.
- [4] M. Gen and R. Cheng, "Interval Programming using Genetic Algorithms," in *Proceedings of the Sixth International Symposium on Robotics and Manufacturing*, Montpellier, France, 1996.
- [5] E. M. Montes and C. A. Coello Coello, "A simple multimembered evolution strategy to solve constrained optimization problems: Smes," *IEEE Transactions on Evolutionary Computation*, vol. 9, no. 1, pp. 1–17, February 2005.
- [6] T. Ray, H. K. Singh, A. Isaacs, and W. Smith, "Infeasibility driven evolutionary algorithm for constrained optimization," in *Constraint-Handling in Evolutionary Computation*, E. Mezura-Montes, Ed. Berlin: Springer. Studies in Computational Intelligence, Volume 198, 2009, ch. 7, pp. 145–165, ISBN 978-3-642-00618-0.
- [7] Y. G. Woldesenbet, G. G. Yen, and B. G. Tessema, "Constraint handling in multiobjective evolutionary optimization," *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 3, June 2009.
- [8] G. Yen, "An adaptive penalty function for handling constraint in multi-objective evolutionary optimization," *Constraint-Handling in Evolutionary Optimization*, pp. 121–143, 2009.
- [9] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W H Freeman and Co San Francisco, 1979.
- [10] S. S. Rao, *Engineering Optimization*, 3rd ed. John Wiley & Sons, 1996.
- [11] H. Li and Q. Zhang, "Multiobjective Optimization Problems with Complicated Pareto Sets, MOEA/D and NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 2, pp. 284–302, April 2009.
- [12] Q. Zhang and H. Li, "MOEA/D: A multiobjective evolutionary algorithm based on decomposition," *IEEE Transactions on Evolutionary Computation*, vol. 11, no. 6, pp. 712–731, 2007.
- [13] Q. Zhang, W. Liu, and H. Li, "The Performance of a New Version of MOEA/D on CEC09 Unconstrained Mop Test Instances," in *Special Session on Performance Assessment of Multiobjective Optimization Algorithms/CEC 09 MOEA Competition*, Norway, 18-21 May 2009.

- [14] L. Tseng and C. Chen, "Multiple trajectory search for unconstrained/constrained multi-objective optimization," in *IEEE Congress on Evolutionary Computation, CEC2009*. IEEE, 2009, pp. 1951–1958.
- [15] H. Liu and X. Li, "The multiobjective evolutionary algorithm based on determined weight and sub-regional search," in *IEEE Congress on Evolutionary Computation, CEC2009*. IEEE, pp. 1928–1934.
- [16] M. Liu, X. Zou, Y. Chen, and Z. Wu, "Performance assessment of DMOEA-DD with CEC 2009 MOEA competition test instances," in *IEEE Congress on Evolutionary Computation, CEC2009*. IEEE, 2009, pp. 2913–2918.
- [17] K. M. Miettinen, *Nonlinear Multiobjective Optimization*. Kluwer Academic Publishers, 1999.
- [18] M. A. Jan and R. A. Khanum, "A study of two penalty-parameterless constraint handling techniques in the framework of MOEA/D," *Appl. Soft Comput.*, vol. 13, no. 1, pp. 128–148, 2013.
- [19] Q. Zhang, A. Zhou, S. Zhao, P. N. Suganthan, W. Liu, and S. Tiwari, "Multiobjective optimization test instances for the CEC 2009 special session and competition," University of Essex and Nanyang Technological University, Tech. Rep. CES-487, 2008.