Uluslararası İleri Doğa Bilimleri ve Mühendislik Araştırmaları Dergisi
Sayı 7, S. 390-395, 10, 2023

Araştırma Makalesi

https://alls-academy.com/index.php/ijanser
ISSN: 2980-0811

International Journal of Advanced Natural Sciences and Engineering Researches
Volume 7, pp. 390-395, 10, 2023

Research Article

Comparing the Transform-based Algorithms to Handle Constraint Multiobjective Optimization Problems

Okkes Tolga Altinoz[1*]

[1]*Department of Electrical and Electronics Engineering, Ankara University, Turkey*

[*]*(taltinoz@ankara.edu.tr) Email of the corresponding author*

(Received: 16 November 2023, Accepted: 26 November 2023)

(4th International Conference on Engineering and Applied Natural Sciences ICEANS 2023, November 20-21, 2023)

ATIF/REFERENCE: Altinoz, O. T. (2023). Comparing the Transform-based Algorithms to Handle Constraint Multiobjective Optimization Problems. *International Journal of Advanced Natural Sciences and Engineering Researches*, 7(10), 390-395.

Abstract – Engineering problems are converted to multiobjective problems and the methods to solve these problems have progressed significantly in recent years. However, most of these algorithms are designed to solve unconstrained multiobjective optimization problems. In fact, many engineering problems contains s large number of constraints. For this reason, handling the constraints is relatively hard challenge for multiobjective algorithms. In recent years, the constraint handling methods have achieved promising performance. Among these handling techniques, transforming the problem to a different problem to handle the constraint looks promising and relatively new papers have been focusing on these frameworks. Hence in this research this constraint handling algorithms discussed on a set of multiobjective algorithms and analyzed their performances by comparing them with each other.

Keywords – Multiobjective Optimization, Constraint Handling, Transform, Evolutionary Algorithms.

I. Introduction

Engineering problems usually involve multiple conflicting objectives with lots of constraints. These problems denoted as constrained multi-objective optimization problems (CMOPs), and they can be defined as follows:

$$F(x) = (f_1, \dots, f_m)^T \quad (1)$$

subject to

$$g_i(x) \geq 0, i = 1, \dots, g$$
$$h_j(x) = 0, j = 1, \dots, h$$

where F is the m-dimensional objective function vector, there are g number of inequality constraint and h number of equality constraints. A solution is said to be feasible if both constraints are met, equality and inequality constraints.

Assume that two solutions are feasible x and y, the solution x said to dominate y if $f_i(x) \leq f_i(y)$ for each $i \in \{1, \dots, m\}$ and $f_j(x) < f_j(y)$ for at least one $j \in \{1, \dots, m\}$. If a feasible solution z dominating x, x said to be a feasible Pareto optimal solution. The set of all feasible Pareto solutions is called the Pareto set. The shape of the Pareto set on the objective space is denoted as the Pareto front.

Unlike bound constraints, the equality and inequality constraints need additional methodologies. In order to present more efficient solution for CMOPs, penalty-based methods [6], objective/constraint separation [7], additive optimization methods [8], hybrid methods [9], altering crossover/operators [10] are methods to solve CMOPs.

In this research transforming CMOPs into other problems are investigated. More specifically, they are converting two stage optimization problems. Generally, in the first stage the algorithm aims to converge to Pareto front. At the second stage similar to the filter, the improvement at the solutions makes [11]. The push-pull search (PPS) framework is embedded to solve CMOPS [12]. In PPS method, at the push stage of PPS the unconstraint Pareto Front (UPF) is violated and at the pull stage the solutions are search in constraint Pareto Front (CPF). In [13], PPS framework is combined with the cellular genetic algorithm.

In addition of these studies, relatively recent algorithms ToP [1], CMOEA-MS [2], CTAEA [3], and CCMO [4] are selected as case studies for this study. Therefore, in this research the aim is to analyze the transform-based constraint handling mechanism on four CMOEAs by comparing them under two and three objective benchmark problems.

The remainder of this paper is structured as follows. The basic concepts of algorithms, benchmark problems, metrics and statistical tests are introduced after the introduction section. In the following section the implementation results are given. Finally, the conclusions are drawn in the final section.

II. Problems and Solutions

The benchmark problems, the CMOEAs, metrics to compare results and the statistical tests used in this research are introduced in this section.

A. *Constraint Benchmark Problems: Difficulty-Adjustable and Scalable CMOPs (DAS-CMOP)*

Three difficulty levels are defined in [5]. Diversity-hardness, Feasibility-hardness, and Convergence-hardness. The authors in [5] formularized the CMOP with two parts: objective function and constraint function. They renamed as shape function and distance function respectively. By proposing 5these functions a set of problems are proposed in [5]. In this research, among them three two objective benchmark problems (Eq.s 2-4) and one three objective benchmark problem (Eq. 5) is employed to discuss the performances of the CMOEAs. They can be defined as follows.

DAS-CMOP1:

$$f_1(x) = x_1 + g(x) \quad (2)$$
$$f_2(x) = 1 + x_1^2 + g(x)$$
$$g(x) = \sum_{j=1}^{n} \left(x_j - sin\left(x_1 \frac{\pi}{2} \right) \right)^2$$

Subject to

$$c_1(x) = sin(20\pi x_1) - b \geq 0$$
$$c_2(x) = \big(e - g(x)\big)\big(g(x) - 0.5\big) \geq 0$$
$$c_{k+2}(x) = \frac{\big((f_1 - p_k)\cos(-0.25\pi) - (f_2 - q_k)sin(-0.25\pi)\big)^2}{0.3} + \frac{\big((f_1 - p_k)\sin(-0.25\pi) - (f_2 - q_k)cos(-0.25\pi)\big)^2}{1.2} \geq r$$
$$p_k = [0,1,0,1,2,0,1,2,3]$$
$$q_k = [1.5,0.5,2.5,1.5,0.5,3.5,2.5,1.5,0.5]$$

DAS-CMOP2:

$$f_1(x) = x_1 + g(x) \quad (3)$$
$$f_2(x) = 1 + \sqrt{x_1} + g(x)$$

g(x) and constraints are same with DAS-CMOP1

DAS-CMOP3:

$$f_1(x) = x_1 + g(x) \quad (4)$$
$$f_2(x) = 1 + \sqrt{x_1} + 0.5 * |\sin(5\pi x_1)| g(x)$$

g(x) and constraints are same with DAS-CMOP1

DAS-CMOP9:

$$f_1(x) = \cos(0.5\pi x_1) * \cos(0.5\pi x_2) + g(x) \quad (5)$$
$$f_2(x) = \cos(0.5\pi x_1) * \sin(0.5\pi x_1) + g(x)$$
$$f_3(x) = \sin(0.5\pi x_1) + g(x)$$
$$g(x) = \sum_{j=3}^{n} \left(x_j - cos\left((x_1 + x_2)\frac{0.25j\pi}{n} \right) \right)^2$$

Subject to

$$c_1(x) = sin(20\pi x_1) - b \geq 0$$
$$c_2(x) = cos(20\pi x_2) - b \geq 0$$
$$c_3(x) = \big(e - g(x)\big)\big(g(x) - 0.5\big) \geq 0$$
$$c_{k+3} = \sum_{j=1, j\neq k}^{3} f_j^2 + (f_k - 1)^2 - r^2 \geq 0$$
$$c_7 = \sum_{j=1}^{3} \big(f_j - 1/\sqrt{3}\big)^2 - r^2 \geq 0$$

B. *Two-Phase Framework -ToP [1]*

Liu et al have discussed the constrained MOEAs and they constructed a set of CMOPs, named DOC. These are the benchmark problem sets that are constructed by considering both decision and objective constraints from real-word application with different properties. In [1], after investigating the constraints on the real-word applications and proposing the set of benchmark problems, a simple but efficient two-phase framework, called ToP is proposed.

The ToP is a two stage MOEA so that at the first phase a feasible area is detected from a set of constrained single objective problems. That means the algorithm converts CMOP into a single objective COP (SCOP). At the second phase the final solutions are "pulled" from the solution space. The Algorithm begins with the conversion of the CMOP into SCOP with the aid of weighted sum scalarization method, then *N* constrained single objective optimization problems with *N* different weight vectors are generated. After that the algorithm has a repeated loop so that the first and second phases are repeated consequently.

First Phase: (constrained single objective optimization) The aim of this phase is to detect the best candidates to transfer to the second phase. For this reason, an optimization method is used with two components, constraint handling technique and search engine.

First Phase Constraint handling: Three rules are followed to handle the constraints. For two solutions (decision variables) 1) if both of them are feasible and select the variable with the best objective value. 2)If only one of them is feasible than select feasible variable. 3)If both of them are infeasible than calculate the degree of constraint violation by selecting the maximum positive value from the equality and inequality constraints. Then select the decision variable with the smaller constraint violation degree. Based on these three rules, the decision variables are stored and transferred to the next phase (This is the constraint selection operator).

First Phase Search Engine: Two of the Differential Evolution (DE)'s crossover algorithms are evaluated with the same probability to generate offspring (DE/current-to-rand/1 and DE/rand-to-best/1/bin).

Second Phase: (Constrained Multiobjective Optimization) At the first phase the set of feasible area has been detected. However, some of the individuals in the population may still needs to be improved. Therefore, a CMOEA algorithm is implemented in the second phase.

Second Phase: (Constrained Multiobjective Evolutionary Algorithm CMOEA) CMOEA has two components, multiobjective evolutionary algorithm and constraint handling method. The dominance-base CMOEA is used in the paper (NSGA-II and IDEA algorithms are preferred and named as ToP-NSGA-II and ToP-IDEA).

The results from all the comparisons in the ToP researched showed that without first phase, CMOEA not able to enter the feasible region or stall in the local region.

C. *Two-Stage Framework Constraint Multiobjective Evolutionary Algorithm - CMOEA-MS*

Tian at al. proposed a two-stage framework for the CMOP [2]. Unlike ToP this method can switch between two stages according to the status of the current population. The first stage is for reaching the feasible region from the infeasible region, and the stage two is given for spreading among the feasible regions. In the first stage it is assumed that mostly of the individuals at the population are in the infeasible region, and the objective is to "push" them to the feasible region. In the second state, it is assumed that most of the individuals are in the feasible region, and the objective has the lower priority to push them into the feasible region.

The algorithm begins with the randomly assignment of the individuals in the population and calculation of the objective functions. At the beginning of the iterations, by using the binary tournament selection method the parents are selected and stored in P. Then two stages are repeated with a parameter λ which is used to determining the stage. That parameter is a constant value compared with the number of feasible solutions in P. If it is true than Stage A else Stage B executed.

Stage A: Constraint violation and shift-based density estimation-based distance (SDE) calculation is evaluated as the first stage. The minimum one among these two values are selected. To calculate SDE, the objective valuers are needed. The objective is the basically the number of solutions who violated the feasibility boarders and the inverse distance to the nearest neighbour. This became the objective value of the Stage A

Stage B: Similar calculations are repeated in Stage B. But this time a rule-based approach, which is used in ToP is used as the domination principle. Stage B always prefers the solution with lower constraint violation. This is used as constraint violation value, and it also used to calculate objective value.

D. Two-Archive Framework Constrained Evolutionary Algorithm - CTAEA

Li et al. proposed a parameter-free two-archive framework for CMOPs [3]. Unlike previous algorithms this method maintains the collaboration of archives simultaneously. One of the archives is denoted for convergency (CA) and the other is for the diversity (DA). The contribution of this study is the restricted mating selection for the crossover. İt is adaptively chooses appropriate parents based on their evolution status.

Two-archives CA and DA produce solution candidates and proportion of the nondominated solutions is the criteria for the assignment of the solution set from DA or CA to the Tournament Selection operator (to select mating parents). If proportional of the nondominated solutions of CA in unified set from DA and CA is smaller than DA means that convergency of the CA is better than DA, vice versa. The selected parents are applied to the reproduction mechanism and these steps are repeated.

CA: It is begun with a form of hybrid population from CA and offsprings. The feasible solutions are chosen from this set and recorded to the temporary archive. Then this archive is divided into several nondominated fronts by using nondominated sorting algorithm. The feasible solutions are selected from each front.

DA: Unlike similar algorithm, in the second stage it is not consider the constraint violation. The data from CA is selected as reference set. First DA and offsprings are formed to get a hybrid population. Each solution assigned the subregions in CA and then iteratively investigate each subregion to survived to the next generation.

E. Coevolutionary Constrained Multi-Objective Optimization framework - CCMO

Tianj et al. proposed two-population method for solving CMOPs [4]. The CMOP is changed as two problems. The original CMOP is solved with one population and t5he other population solve a helper problem. These two populations share information with each other when they are evolved separately.

The algorithm begins with the initialization randomly of the individuals in the population. However, this framework work on two populations. Each population applied to same mating selection operator; offspring are generated from crossover. And environmental selection is applied to each population separately. However, the only difference is the evaluation of the objective function. The firs population applies to the original objective functions and the other is evaluated into the helper function. The helper function is derived from the original function and consists of a part of the original function and the constraint. And this function is easier than the original function. The aim of the population 2 is to jump out from the local optimum. The information is share by exchanging the offspring between populations. The offspring of the first population is transferred to the second population vice versa.

F. Metrics and Statistical Tests

Two widely accepted metrics are chosen to assess the performance of the algorithms. The inverted generational distance (IGD) is calculated to demonstrate how the solution set converges to the Pareto Front [14]. And the hypervolume is calculated as the total volume of the objective space dominated by the solution set [15]. This metric used to demonstrate how the solution set converges and distributed among the objective space.

The implementations are repeated many times (independent run). Therefore, statistical properties of these metrics are recorded and analysed. The first approach for statistics are the mean and standard deviation. In addition, for analyse significance of the experimental results from different experiments on the same problem Rank sum test is conducted at the significance level of 5%. This test is used for comparing statistics of two test results. The assumptions are the independency of the dataset, with the same variance and normal distributed solution set. However, the results from algorithms are not same variance and they are not normally distributed. But if the samples are large enough it is said that these assumptions are robust and approximately succeeded the assumption. By using this test approximately similar results and significant differences emphasized. The idea of rank sum test is to calculate the differences of the dataset. The absolute differences are ranked beginning with the largest difference. The sum of the ranks for the negative and positive differences calculates. The minimum among the sum of these negative and positive ranks is selected. Then this value is compared with the expected value of the ranks.

Finally, the test check for significance by comparing these values.

TABLE 1. Comparison Results on IGD Metric (Median and Standard Deviation) on Benchmark Functions

Problem	M	D	ToP	CMOEA_MS	CTAEA	CCMO
DASCMOP1	2	30	7.6596e-1 (4.43e-2) -	7.4657e-1 (3.13e-2) =	1.9001e-1 (1.30e-2) +	7.1571e-1 (4.11e-2)
DASCMOP2	2	30	5.0828e-1 (2.75e-1) =	2.7856e-1 (3.39e-2) =	1.1168e-1 (4.28e-2) +	2.5515e-1 (2.73e-2)
DASCMOP3	2	30	6.6287e-1 (1.50e-1) -	3.3364e-1 (3.40e-2) =	1.8367e-1 (5.96e-2) +	3.3844e-1 (2.55e-2)
DASCMOP9	3	30	6.9464e-1 (1.87e-1) -	4.0874e-1 (1.09e-1) =	2.6087e-1 (7.62e-2) =	3.5742e-1 (7.30e-2)
+/-/=			0/3/1	0/0/4	3/0/1	

TABLE 2. Comparison Results on Hypervolume Metric (Median and Standard Deviation) on Benchmark Functions

Problem	M	D	ToP	CMOEA_MS	CTAEA	CCMO
DASCMOP1	2	30	4.6387e-3 (7.08e-3) =	4.8604e-3 (4.51e-3) =	1.6604e-1 (3.45e-3) +	9.6823e-3 (7.26e-3)
DASCMOP2	2	30	1.2605e-1 (1.07e-1) -	2.5412e-1 (4.86e-3) -	3.0131e-1 (1.22e-2) +	2.5880e-1 (2.29e-3)
DASCMOP3	2	30	4.6548e-2 (5.90e-2) -	2.1145e-1 (9.13e-3) -	2.3981e-1 (1.81e-2) +	2.1245e-1 (1.41e-2)
DASCMOP9	3	30	6.7651e-2 (2.44e-2) -	1.2106e-1 (1.58e-2) =	1.4217e-1 (1.41e-2) =	1.3120e-1 (1.32e-2)
+/-/=			0/3/1	0/2/2	3/0/1	

III. IMPLEMENTATION

Three two objective benchmark problems DASCMOP1- DASCMOP3 and one three objective benchmark problem DASCMOP9 are selected to form a test suite. The benchmark problems have solved by four CMOEAs with 15 independent runs.

Comparison results of the IGD values are given in Table 1 and hypervolume values are given in Table 2. In general speaking CTAEA algorithm produce superior values on all benchmark problems, especially on two objective CMOPs.

The most cost effective and simple algorithm CCMO with two population framework produce almost same values statistically with CMOEA-MS algorithm. Among two-stage algorithms, ToP produces worst performance for both two and three objective problems.

For three objective problem the algorithms expect ToP gives very closed results, almost same with speaking statistical test results. Since the nature of the CTAEA algorithm is based on two methods for diversity and convergency, the hypervolume metric gives the best with the IGD metric values.

Another important contribution of the CTAES it the hybrid population idea. This idea also presenting in CCMO as information sharing. So that the hybrid population is constructed from CA, DA, and offspring. By this way the information related to the feasible region and the constraint, and also the diversity and convergency data is shared among the population. Finally, the DA and CA is not executed in serial manner. It is possible to shuffle the order of DA and CA based on the number of nondominated individuals.

IV. CONCLUSION

The constraint handling problem in Multiobjective optimization problems (CMOPS) is critical for specially to solve the engineering problems, real-word applications. Generally, these applications need many equality and inequality constraints that must be succeeded. It is possible to categorize these methods. One of the strategies is to conversion/transform the problem into different form. The framework in this category is dependent on two levels. The order of these level and the defined formulations has a direct effect on the performance. Among the compared algorithms CTAES produced the best metric values. The main reason is the information sharing and evaluation of the individuals in the population at the infeasible region. The DA and CA techniques not only improved the diversity and convergency of the solutions but also at each iteration with the number of nondominated individuals the offspring and parents construct a hybrid population that cause to share information related to constrain, diversity and convergency.

As future study, this idea will be improved and a novel constraint handling algorithm will be presented.

REFERENCES

[1] Z. Liu and Y. Wang, “Handling constrained multiobjective optimization problems with constraints in both the decision and objective spaces,” IEEE Transactions on Evolutionary Computation, vol. 23, no. 5, pp. 870-884, 2019.
[2] Y. Tian, Y. Zhang, Y. Su, X. Zhang, K. C. Tan, and Y. Jin, “Balancing objective optimization and constraint satisfaction in constrained evolutionary multi-objective optimization,” IEEE Transactions on Cybernetics, vol. 52, no. 9, pp. 9559-9572, 2022.
[3] K. Li, R. Chen, G. Fu, and X. Yao, “Two-archive evolutionary algorithm for constrained multi-objective optimization,” IEEE Transactions on Evolutionary Computation, vol. 23, no. 2, pp. 303-315, 2018.
[4] Y. Tian, T. Zhang, J. Xiao, X. Zhang, and Y. Jin, “A coevolutionary framework for constrained multi-objective optimization problems,” IEEE Transactions on Evolutionary Computation, vol. 25, no. 1, pp. 102-116, 2021.
[5] Z. Fan, W. Li, X. Cai, H. Li, C. Wei, Q. Zhang, K. Deb, and E. Goodman, “Difficulty adjustable and scalable constrained multi objective test problem toolkit,” Evolutionary Computation, vol. 28, no. 3, pp. 339-378, 2020.
[6] Z. Xia, Y. Liu, J. Lu, J. Cao, and L. Rutkowski, “Penalty method for constrained distributed quaternion-variable optimization,” IEEE Trans. Cybern., vol. 51, no. 11, pp. 5631–5636, 2021.
[7] B.-C. Wang, H.-X. Li, J.-P. Li, and Y. Wang, “Composite differential evolution for constrained evolutionary optimization,” IEEE Trans. Syst., Man, Cybern., Syst., vol. 49, no. 7, pp. 1482–1495, 2019.
[8] C. Segura, C. A. Coello Coello, G. Miranda, and C. León, “Using multi-objective evolutionary algorithms for single-objective constrained and unconstrained optimization,” Ann. Oper. Res., vol. 240, no. 1, pp. 217–250, 2016.
[9] Z. Fan et al., “Push and pull search for solving constrained multi-objective optimization problems,” Swarm Evol. Comput., vol. 44, pp. 665–679, 2019.
[10] V. Kelner, F. Capitanescu, O. Léonard, and L. Wehenkel, “A hybrid optimization technique coupling an evolutionary and a local search algorithm,” J. Comput. Appl. Math., vol. 215, no. 2, pp. 448–456, 2008.
[11] L. V. Santana-Quintero, A. G. Hernández-Díaz, J. Molina, C. A. Coello Coello, and R. Caballero, “DEMORS: A hybrid multi-objective optimization algorithm using differential evolution and rough set theory for constrained problems,” Comput. Oper. Res., vol. 37, no. 3, pp. 470–480, 2010.
[12] Z. Fan et al., “Push and pull search for solving constrained multi-objective optimization problems,” Swarm Evol. Comput., vol. 44, pp. 665–679, 2019.
[13] C. Garcia-Garcia, M.-G. Martínez-Peñaloza, and A. Morales-Reyes, “CMOGA/D: A novel cellular ga based on decomposition to tackle constrained multiobjetive problems,” in Proc. Genet. Evol. Comput. Conf. Compan., pp. 1721–1729, 2020.
[14] P. A. N. Bosman and D. Thierens, “The balance between proximity and diversity in multiobjective evolutionary algorithms,” IEEE Trans. Evol. Comput., vol. 7, no. 2, pp. 174–188, 2003.
[15] E. Zitzler and L. Thiele, “Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach,” IEEE Trans. Evol. Comput., vol. 3, no. 4, pp. 257–271, 1999.