Analysis of two variants of the Generalized Differential Evolution algorithm with ordered mutation for real world engineering multi-objective optimization problems

Rafael P. Garcia[1], Dênis E. C. Vargas[2], Afonso C. C. Lemonge[3]

[1]*Department of Architecture and Urbanism, Federal University of Viçosa*
Viçosa, Brazil
rafael.pgarcia@ufv.br
[2]*Department of Mathematics, Federal Center for Technological Education of Minas Gerais*
Belo Horizonte, Brazil
denis.vargas@cefetmg.br
[3]*Department of Applied and Computational Mechanics, Federal University of Juiz de Fora*
Juiz de Fora, Brazil
afonso.lemonge@ufjf.edu.br

Abstract. Differential Evolution (DE) is one of the most powerful commonly used metaheuristics for global multi-objective optimization. New strategies to improve the DE's performance are an important and attractive research study. The third Evolution Step of Generalized Differential Evolution (GDE3) is a widely used DE-based multi-objective evolutionary algorithm in the literature, especially in real-world multi-objective optimization problems with two or three conflicting objectives in its formulation. GDE3 uses the most popular mutation strategy of the DE, DE/rand/1, which randomly selects three candidate solutions from the population without considering any order. The fourth version of the Generalized Differential Evolution (GDE4) was recently proposed, which presents an ordered mutation operator based on two well-known schemes: Non-dominated Ranking and Crowding Distance. Previous studies have shown that GDE4 outperforms GDE3 on a set of many-objective optimization problems. In this paper, the second version of GDE4 is proposed, GDE4-II, considering a local ordering among the three randomly selected individuals instead of the entire population as GDE4. Besides, experiments are conducted to evaluate the performance of the two GDE4 variants in benchmark and engineering multi-objective optimization problems with two and three objective functions. Metrics such as Hypervolume and Inverted Generational Distance plus (IGD+) combined with performance profiles are used to point out the robustness of the GDE4 and GDE4-II.

Keywords: Multi-Objective Optimization, Differential Evolution, Ordered Mutation.

1 Introduction

Differential Evolution (DE) (Storn and Price [1]) is a simple and efficient evolutionary algorithm for solving complex numerical optimization problems in continuous search spaces. Several extensions of DE to solve Multi-objective Optimization Problems (MOP) have already been proposed in the literature. Among them, the third Evolution Step of Generalized Differential Evolution (GDE3) (Kukkonen and Lampinen [2]) can be found as a widely used DE-based multi-objective evolutionary algorithm (MOEA). Recently, a new variant was proposed, GDE4 (Bidgoli et al. [3]), with the promise of improving the performance of the previous version.

Since many real-world optimization problems have conflicting objectives, the demand for efficient optimization algorithms both computationally and in finding attractive solutions becomes indispensable. GDE3 attracts attention in this scenario, mainly when applied to two- and three-dimensional problems (i.e., with two or three objective functions), due to the ease of being implemented, robustness, computationally fast, and for its few control parameters. GDE3 was applied by Lemonge et al. [4], Vargas et al. [5], and Zavala et al. [6] to solve bi-dimensional MOPs involving the design of truss structures, optimizing both the structure's weight and the displacement of its nodes. Ibrahim et al. [7] uses GDE3 to optimize the geometric features of a thermoelectric generator for improved efficiency and output power while incorporating different operating conditions. In a three-dimensional scenario, Yin et al. [8] and Goudos et al. [9] applied GDE3 in real-word MOPs. Yin et al. [8] uses GDE3 as a MOEAs in the context of a reliability-aware multi-objective predictive control approach for a wind farm. The three control objectives considered were maximizing the averaged wind farm power production, minimizing the averaged wind farm thrust loads, and maximizing the actuator health-informed wind farm reliability. Also, GDE3 was one of the MOEAs considered by Goudos et al. [9] in a multi-objective optimization approach for indoor wireless network planning, in which the three objectives considered were exposure minimization, coverage maximization, and power consumption minimization.

Both original DE and GDE3 use DE/rand/1 (the most popular mutation DE strategy), randomly selecting three candidate solutions from the population without considering any ordering in its mutation scheme. Lately, Bidgoli et al. [3] proposed an enhanced version of GDE3, which orders those solutions before applying the mutation process. This ordering is based on two well-known strategies: Non-dominated Ranking and Crowding Distance. This new approach is referred to as GDE4. GDE4 sorts the three candidate solutions randomly selected from the population in the best, the second-best, and the worst, according to its Non-dominated Ranking and Crowding Distance measures concerning the entire population. The authors called this scheme by DE/order/1 strategy and conducted experiments on a benchmark set of many-objective optimization with 5, 10, and 15 objectives. All of them were unconstrained optimization problems. The results showed that GDE4 outperforms GDE3.

This paper proposes to evaluate the performance of GDE4 in MOPs with two and three objective functions. Also, a second version of the GDE4 is proposed, called GDE4-II, which applies the order strategy of GDE4 comparing only the three selected individuals instead of the entire population. This work also extends the knowledge about the robustness of those algorithms, with experiments on constrained real multi-objective structural optimization problems.

Computational experiments with GDE3, GDE4, and GDE4-II in MOPs with 2 and 3 objectives are performed. The aim is to verify if the GDE4 and the GDE4-II present good performances as aforementioned, making GDE4 and GDE4-II also options for future works in MOPs with two or three objective functions, especially real-world MOPs. The benchmarks ZDT (Zitzler et al. [10]), DTLZ (Deb et al. [11]), WFG (Huband et al. [12]), and continuous structural multi-objective optimization design of the 10-, 25-, and 72-bar trusses (Vargas et al. [5]) problems were adopted for the numerical experiments. The obtained results were evaluated according to the Hypervolume (Zitzler and Thiele [13]) and Inverted Generational Distance plus (IGD+) (Ishibuchi et al. [14]) metrics and Performance Profiles (Dolan and Moré [15]).

This paper is organized as follows: Section 2 describes the general formulation of the multi-objective optimization problem. Section 3 provides the steps of generalized differential algorithms used and the extensions proposed in this paper. The numerical experiments are presented and analyzed in Section 4. Finally, the paper ends with the conclusions reported in Section 5.

2 Multi-Objective Optimization Problem

A MOP is formulated as

$$\begin{aligned} \min_{\mathbf{x}} \quad & [f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_m(\mathbf{x})] \\ s.t. \quad & \mathbf{x} = (x_1, \ldots, x_n) \quad \text{with} \quad x_i \in [l_i, u_i] \end{aligned} \tag{1}$$

where l_i and u_i are the lower and upper bounds, respectively, of each x_i of vector $\mathbf{x} \in \mathbb{R}^n$, $\forall i = 1, \ldots, n$. Furthermore, equality ($h(\mathbf{x}) = 0$) and inequality ($g(\mathbf{x}) \leq 0$) constraints can be considered. In this case, we are looking for solutions that optimize the objective functions and satisfy the constraints.

A MOP with more than three objective functions ($m > 3$) is called a many-objective optimization problem. Since the solution of a MOP involves conflicting objectives, the aim is to find good compromises (trade-offs), represented by the Pareto optimum set (PS), which is based on the concept of dominance defined as follows: a solution $\mathbf{x} \in \mathbb{R}^n$ dominates another solution $\mathbf{y} \in \mathbb{R}^n$ ($\mathbf{x} \prec \mathbf{y}$) iff $f_i(\mathbf{x}) \leq f_i(\mathbf{y})$, $\forall i = 1, \ldots, n$, and $\exists j (1 \leq j \leq n)$ with $f_j(\mathbf{x}) < f_j(\mathbf{y})$. Two solutions $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ where $\mathbf{x} \not\prec \mathbf{y}$ and $\mathbf{y} \not\prec \mathbf{x}$ are termed non-dominated solutions.

This definition says that PS is formed by all non-dominated solutions that represent the best trade-offs as possible of a MOP, that is, $\mathbf{x} \in PS$ of the MOP if there exists no $\mathbf{y}$ which $\mathbf{y} \prec \mathbf{x}$. The Pareto Front of the MOP is the set $PF = \{(f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_m(\mathbf{x})), \forall \mathbf{x} \in PS\}$. Population evolutionary algorithms like DE are indicated to solve MOPs since they can obtain robust approximations for the PS in a single execution.

3 Generalized Differential Evolution

The Generalized Differential Evolution (GDE) has extended the DE for MOP. GDE3, its most popular version, starts by randomly generating an initial population and then improves it using DE's classical mutation strategy (DE/rand/1), which produces a trial vector from three vectors randomly selected in the population. The crossover rate ($CR \in [0, 1]$), the mutation factor ($F \in \mathbb{R}$), and the population size (N) are user-defined parameters.

Mathematically, let P_G be a population of N decision vectors $\mathbf{x}_{i,G}$ in generation G, where $i \in \{1, \ldots, N\}$. Each $\mathbf{x}_{i,G}$ is an n-dimensional vector and $\mathbf{x}_{j,i,G}$ is its j-th component, with $j \in \{1, \ldots, n\}$. Then, a trial vector $\mathbf{u}_{i,G}$ is constructed through of DE/rand/1 strategy. After the mutation and crossover operations, the trial vector $\mathbf{u}_{i,G}$ is compared to the target vector $\mathbf{x}_{i,G}$. The trial vector $\mathbf{u}_{i,G}$ is selected to replace the target vector $\mathbf{x}_{i,G}$ in P_{G+1} if $\mathbf{u}_{i,G} \prec \mathbf{x}_{i,G}$. If $\mathbf{x}_{i,G} \prec \mathbf{u}_{i,G}$, $\mathbf{u}_{i,G}$ is discarded and $\mathbf{x}_{i,G}$ remains in the population. Otherwise, both are included in P_{G+1}. To resize P_{G+1} to N solutions, Non-dominated Ranking and Crowding Distance are applied to P_{G+1}. Non-dominated Ranking sorts all the solutions, assigning them to different Pareto ranks according to the domination among the solutions. To assign the Pareto ranks, we search for all non-dominated solutions of P_{G+1} to assign them to rank 1. The solutions of rank 2 will be those that are non-dominated of P_{G+1} when we discard all solutions of rank 1, and so on. Solutions in a smaller rank are better than those in a larger one. The Crowding Distance is introduced to estimate the density of solutions and select solutions from the same rank. Solutions with higher Crowding Distance are preferred.

The functioning of GDE4 is similar to GDE3, except for the introduction of a new ordered mutation scheme. The GDE4 ordered mutation scheme, called by DE/order/1, sorts $\mathbf{x}_{r_1,G}$, $\mathbf{x}_{r_2,G}$, and $\mathbf{x}_{r_3,G}$ in the best ($\mathbf{x}_b$), the second-best ($\mathbf{x}_{sb}$), and the worst ($\mathbf{x}_w$), according to their Non-dominated Ranking and Crowding Distance measures in relation to the entire population. Then, the ordered mutation DE/order/1 calculates the trial vector by $\mathbf{u}_{i,G} = \mathbf{x}_{i,b,G} + F(\mathbf{x}_{i,sb,G} - \mathbf{x}_{i,w,G})$.

Four possible cases can be considered:

1. All three candidate solutions are in different Pareto Fronts. In this case, the lowest rank will be $\mathbf{x}_b$, the middle rank will be $\mathbf{x}_{sb}$, and the greatest rank will be $\mathbf{x}_w$;
2. Two candidate solutions are in the same Pareto Front and the third in a lower rank. In this case, the lowest rank will be $\mathbf{x}_b$. Between the two of the same rank, one with the biggest Crowding Distance will be $\mathbf{x}_{sb}$, and the other will be $\mathbf{x}_w$;
3. Two candidate solutions are in the same Pareto Front and the third in a bigger rank. In this case, the biggest rank will be $\mathbf{x}_w$. Between the two of the same rank, the one whose Crowding Distance is bigger will be $\mathbf{x}_b$, and the one whose Crowding Distance is smaller will be $\mathbf{x}_{sb}$;
4. All three candidate solutions are in the same Pareto Front. In this case, the biggest Crowding Distance will be $\mathbf{x}_b$, the middle Crowding Distance will be $\mathbf{x}_{sb}$, and the lowest Crowding Distance will be $\mathbf{x}_w$;

We propose a modification to the GDE4, referred to as GDE4-II. GDE4-II applies the order also based on dominance and crowding distance but here considering only the three randomly selected solutions $\mathbf{x}_{r_1,G}$, $\mathbf{x}_{r_2,G}$, and $\mathbf{x}_{r_3,G}$, unlike the GDE4 that orders based on the entire population. We established this approach by realizing the highly elitist condition of the GDE4, indicating a tendency of the algorithm towards rapid convergence and stagnation in local optima. Although GDE4-II still has an elitist character, the relaxation of the mutation operator for local ordering may favor discovering new promising areas of the search space.

The other components in GDE3 remain the same in both GDE4 and GDE4-II: the trial vector $\mathbf{u}_{i,G}$ is compared to the target vector $\mathbf{x}_{i,G}$ to replace it or not in the next generation P_{G+1}, whose size is kept fixed at N by Non-dominated Ranking and Crowding Distance.

In Bidgoli et al. [3], no experiments were performed with problems with constraints. Another contribution of this paper was to extend the scheme DE/order/1 to problems with constraints. For this, it was used the GDE3 constraint-domination concept, denoted by ($\prec_c$): $x_1 \prec_c x_2$ iff any of the following conditions is true: 1) x_1 is feasible and x_2 is not; 2) x_1 and x_2 are infeasible and x_1 dominates x_2 in constraint function violation space; 3) x_1 and x_2 are feasible and x_1 dominates x_2 in objective function space. So, $\mathbf{x}_b$, $\mathbf{x}_{sb}$, $\mathbf{x}_w$ are chosen in such a way that $\mathbf{x}_b \prec_c \mathbf{x}_{sb} \prec_c \mathbf{x}_w$ in problems with constraints.

4 Numerical Experiments

We compare the performance of the GDE family on four benchmark sets of problems: (i) ZDT, 5 bi-dimensional problems (Zitzler et al. [10]); (ii) DTLZ, 7 tri-dimensional problems (Deb et al. [11]); (iii) WFG, 9 tri-dimensional problems (Huband et al. [12]); (iv) continuous structural multi-objective optimization design of the 10-, 25-, and 72-bar trusses (Vargas et al. [5]).

Concerning ZDT problems, 150 generations and 100 candidate solutions in the population were set. The DE user-defined parameters $CR = 0.9$ and $F = 0.5$ were adopted, except for ZDT4 problem whose $CR = 0.1$. As for DTLZ problems, 250 generations and a population size of 100 solutions were used. $CR = 0.1$ and $F = 0.5$ for all DTLZ problems. These parameters are commonly found in the literature. Due to DTLZ are scalable problems, the following dimensions were adopted: DTLZ1, $n = 7$; DTLZ 2-6, $n = 12$; and DTLZ7, $n = 22$. The parameter $\alpha = 100$ was adopted for DTLZ4. All of these values are suggested by Deb et al. [11]. The following parameters were adopted for all WFG (1-9) problems: $n = 12$, $CR = 0.1$ and $F = 0.5$, 250 generations and 100 candidate solutions in the population. In the case of continuous structural multi-objective optimization design of the 10-, 25-, and 72-bar trusses (whose full description can be found in Vargas et al. [5]), $CR = 0.1$ and $F = 0.5$ (DE parameters), 500 generations and 100 candidate solutions in the population were set. It was performed 20 independent runs of each algorithm for each MOP.

4.1 Performance Metrics

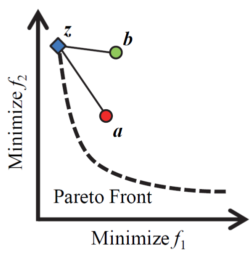
Two characteristics are considered to evaluate the performance of the algorithms: convergence and diversity. It means finding solutions as close as possible to the true Pareto Front while maintaining a good spread along with it. Many performance metrics for measuring these two criteria either separately or together have been proposed in the literature. For this work, it was chosen the Hypervolume (HV) (Zitzler and Thiele [13]) and Inverted Generational Distance plus (IGD+) (Ishibuchi et al. [14]) due to its ability to measure obtained Pareto set both convergence and diversity to the true Pareto front. Also, Performance Profiles (PP) (Dolan and Moré [15]) are presented to reinforce the analyses.

The HV calculates the hypervolume enclosed between the obtained Pareto set and a reference point. In MOPs with two objective functions, the hypervolume is the stairway polygon area whose solutions obtained represent the corner of steps. The point formed by each objective function's maximums is usually used as the reference point (RP) for HV value calculation.

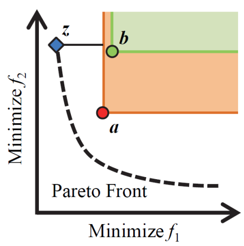
The original Inverted Generational Distance (IGD) (Sierra and Coello [16]) measures the average of the Euclidean distances of each solution in true Pareto Front to the nearest element in the obtained Pareto set. Except for the change from Euclidean distances to the distance from each true Pareto point $\mathbf{z} = (z_1, ..., z_m)$ to the dominated region by the obtained solution $\mathbf{a} = (a_1, ..., a_m)$, calculated as

$$d^+(\mathbf{z}, \mathbf{a}) = \sqrt{\sum_{i=1}^{m} (max\{z_i - a_i, 0\})^2}, \tag{2}$$

there is no difference between the IGD and the Inverted Generational Distance plus (IGD+) (Ishibuchi et al. [14]). The IGD+ has the same advantage as the IGD (simplicity of its computation). At the same time, the IGD+ has an additional advantage: Weakly Pareto-compliant, as shown in Figure 1. The closer the obtained Pareto set approximation is to the true Pareto Front, the smaller the IGD+ value.



(a) IGD



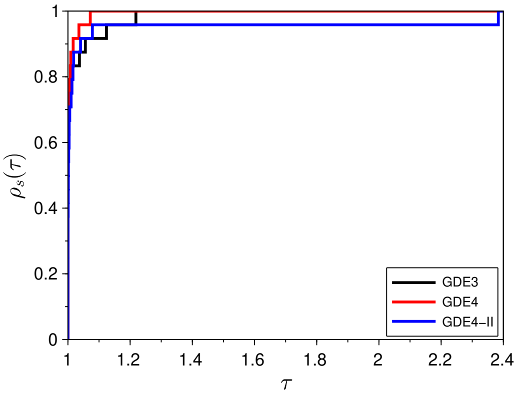
(b) IGD+

Figure 1. Distance calculation in the IGD and the IGD+ (Extracted from [17]).

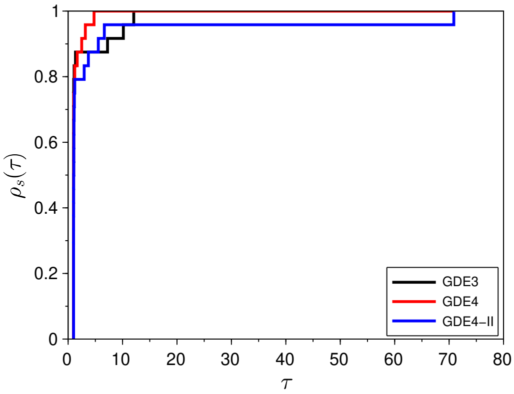
Performance profiles (PP) are an important tool for evaluating and comparing the performance of a set of solvers in a set of problems, according to a given performance metric that one wants to minimize. The area under the curves produced by the PP is a good indicator of the overall performance of the algorithms (Barbosa et al. [18]). The larger the area value, the better the overall performance of the algorithm.

4.2 Results

Figure 2 shows the performance profiles (PP) curves with both HV and IGD+ metrics considering the 20 independent runs performed of each algorithm for each MOP. In general, GDE4 is the algorithm that obtains performance slightly superior to the others according to both metrics HV and IGD+. This is confirmed by the area under its performance curve in the Figure 2, since the larger the area under its PP, the better the overall performance of the algorithm. The PPs also indicate the difficulty of convergence of the GDE4-II for some problems, verified by the high value of τ to achieve $\rho(\tau) = 1$.



(a) HV: GDE4 (1); GDE3 (0.99110); GDE4-II (0.95713).



(b) IGD+: GDE4 (1); GDE3 (0.98901); GDE4-II (0.95400).

Figure 2. Performance profile curves of the HV and IGD+. The normalized areas under their curves are shown between parenthesis.

An example of the good performance of GDE4 is shown in Figure 3, where it was able to obtain higher f_1 values than the others. This behavior is welcome in multi-objective optimization problems. Whether on the one hand, we are looking for convergence to the Pareto curve. On the other hand, we are also interested in the solutions being as spread out as possible.

Although GDE4-II had the worst PP values, it got better performance on problems such as the 10-bar truss (Table 2) and was competitive in many others. Its poor performance was mainly due to the problem ZDT2 (Table 1), a problem with non-convex true Pareto Front.

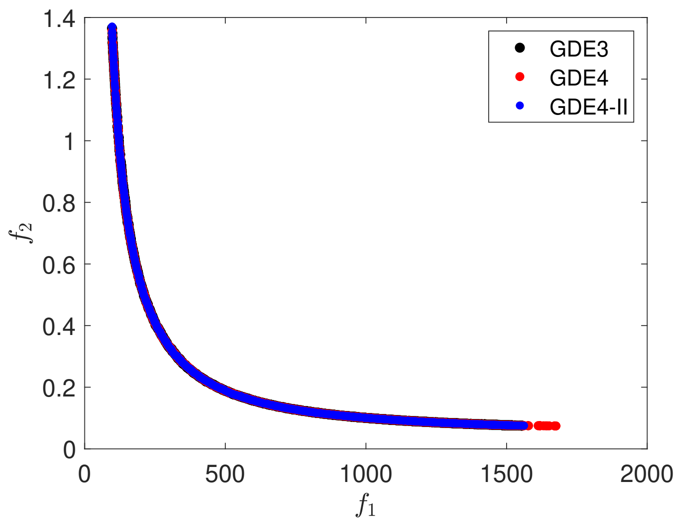


Figure 3. Pareto front obtained in the continuous structural multi-objective optimization design of the 72-bar trus.

Table 1. Hypervolume(HV) and IGD+ mean calculated from the Pareto Front in ZDT2 problems

Problem		GDE3	GDE4	GDE4-II
ZDT2	Mean	0.30322	**0.36951**	0.15500
	Mean	0.04774	**0.00396**	0.28062

Table 2. Hypervolume(HV) and IGD+ mean calculated from the Pareto Front in truss problems

Problem		GDE3	GDE4	GDE4-II
10b	HV	0.86430	0.85668	**0.86460**
	IGD+	0.00147	0.00334	**0.00134**
25b	HV	**0.88551**	0.88550	0.88546
	IGD+	**0.00069**	0.00070	0.00072
72b	HV	0.90923	**0.90928**	0.90907
	IGD+	**0.00124**	0.00129	0.00137

Even though GDE4 was the algorithm with the best overall performance when considering the entire set of problems in this work (Figure 2), it can be seen in Table 2 that GDE4 does not deliver the best results for the truss problems. On the other hand, GDE4-II finds, on average, better results for the 10-bar problem, while GDE3 performs better on the 25-bar problem. Considering the IGD+ metric, GDE3 is also the algorithm that finds the best results for the 72-bar problem.

5 Conclusions

This paper compared the performance of GDE3, GDE4, and GDE4-II when solving benchmark problems and structural multi-objective optimization design of the 10-, 25-, and 72-bar trusses. We emphasize that this paper proposed a second version of GDE4, called here by GDE4-II, considering a local ordering among the three randomly selected individuals instead of the entire population as GDE4. Another contribution of this paper was to extend the scheme DE/order/1 to problems with constraints. The algorithms were evaluated using Hypervolume and Inverted Generational Distance plus (IGD+) combined with its performance profiles. It was observed that GDE4 has the best both HV and IGD+ performance profile in general, although GDE4-II was also competitive on most problems.

Acknowledgements. The third author wish to thank the Conselho Nacional de Desenvolvimento Científico e Tecnológico - CNPq (Grant 306186/2017-9) and FAPEMIG (Grant TEC PPM-00174-18).

Authorship statement. The authors hereby confirm that they are the sole liable persons responsible for the authorship of this work, and that all material that has been herein included as part of the present paper is either the property (and authorship) of the authors, or has the permission of the owners to be included here.

References

[1] R. Storn and K. Price. Differential evolution a simple and efficient adaptive scheme for global optimization over continuous spaces. *Journal of Global Optimization*, vol. 11(4), pp. 341–359, 1997.

[2] S. Kukkonen and J. Lampinen. GDE3: The third evolution step of generalized differential evolution. In *IEEE Congress on Evolutionary Computation (CEC 2005)*, pp. 443–450. IEEE, 2005.

[3] A. A. Bidgoli, S. Mahdavi, S. Rahnamayan, and H. Ebrahimpour-Komle. Gde4: The generalized differential evolution with ordered mutation. In *Proceedings of the 10th International Conference on Evolutionary Multi-Criterion Optimization (EMO 2019)*, volume 11411, pp. 101–113. Lecture Notes in Computer Science, Springer, Cham, 2019.

[4] A. C. Lemonge, J. P. Carvalho, P. H. Hallak, and D. Vargas. Multi-objective truss structural optimization considering natural frequencies of vibration and global stability. *Expert Systems with Applications*, vol. 165, pp. 113777, 2021.

[5] D. E. C. Vargas, A. C. C. Lemonge, H. J. C. Barbosa, and H. S. Bernardino. Differential evolution with the adaptive penalty method for structural multi-objective optimization. *Optimization and Engineering*, vol. 20, pp. 65–88, 2019.

[6] G. Zavala, A. J. Nebro, F. Luna, and C. A. C. Coello. Structural design using multi-objective metaheuristics. comparative study and application to a real-world problem. *Struct Multidisc Optim*, vol. 53, pp. 545–566, 2016.

[7] A. Ibrahim, S. Rahnamayan, M. Vargas Martin, and B. Yilbas. Multi-objective thermal analysis of a thermoelectric device: Influence of geometric features on device characteristics. *Energy*, vol. 77, pp. 305–317, 2014.
[8] X. Yin, X. Zhao, J. Lin, and A. Karcanias. Reliability aware multi-objective predictive control for wind farm based on machine learning and heuristic optimizations. *Energy*, vol. 202, pp. 117739, 2020.
[9] S. K. Goudos, D. Plets, N. Liu, L. Martens, and W. Joseph. A multi-objective approach to indoor wireless heterogeneous networks planning based on biogeography-based optimization. *Computer Networks*, vol. 91, pp. 564–576, 2015.
[10] E. Zitzler, K. Deb, and L. Thiele. Comparison of multiobjective evolutionary algorithms: Empirical results. *Evol. Comput*, vol. 8, pp. 173–195, 2000.
[11] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler. Scalable multi-objective optimization test problems. In *Proceedings of the 2002 Congress on Evolutionary Computation. CEC'02 (Cat. No.02TH8600)*, volume 1, pp. 825–830 vol.1, 2002.
[12] S. Huband, P. Hingston, L. Barone, and L. While. A review of multiobjective test problems and a scalable test problem toolkit. *IEEE Transactions on Evolutionary Computation*, vol. 10, n. 5, pp. 477–506, 2006.
[13] E. Zitzler and L. Thiele. Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach. *IEEE Trans. on Evolutionary Computation*, vol. 3(4), pp. 257–271, 1999.
[14] H. Ishibuchi, H. Masuda, Y. Tanigaki, and Y. Nojima. Modified distance calculation in generational distance and inverted generational distance. In *Proceedings of the 8th International Conference on Evolutionary Multi-Criterion Optimization (EMO 2015)*, volume 9019, pp. 110–125. Lecture Notes in Computer Science, Springer, Cham, 2015.
[15] E. D. Dolan and J. J. Moré. Benchmarking optimization software with performance profiles. *Mathematical Programming*, vol. 91, pp. 201–213, 2002.
[16] M. R. Sierra and C. A. C. Coello. Improving PSO-based multi-objective optimization using crowding, mutation and epsilon-dominance. In *EMO 2005*, pp. 505–519. Springer-Verlag, 2005.
[17] H. Ishibuchi, H. Masuda, and Y. Nojima. Sensitivity of performance evaluation results by inverted generational distance to reference points. In *2016 IEEE Congress on Evolutionary Computation (CEC)*, pp. 1107–1114, 2016.
[18] H. J. C. Barbosa, H. S. Bernardino, and A. M. Barreto. Using performance profiles to analyze the results of the 2006 cec constrained optimization competition. In *IEEE Congress on Evolutionary Computation*, pp. 1–8, 2010.