CILAMCE PANACM
RIO DE JANEIRO 2021

Comparison of multi-objective particle swarm algorithms for truss design optimization

Érica C.R. Carvalho[1], Afonso C.C. Lemonge[2], José Pedro G. Carvalho[3], Patrícia H. Hallak[2]

[1]*Graduate Program of Computational Modeling, Federal University of Juiz de Fora*
José Lourenço Kelmer St, São Pedro, Juiz de Fora, MG 36036-900, Brazil
ericacrcarvalho@gmail.com
[2]*Dept. of Applied and Computational Mechanics, School of Engineering, Federal University of Juiz de Fora*
José Lourenço Kelmer St, São Pedro, Juiz de Fora, MG 36036-900, Brazil
afonso.lemonge@ufjf.edu.br, patricia.hallak@ufjf.edu.br
[3]*Civil Engineering Program, COPPE/Federal University of Rio de Janeiro*
550 Pedro Calmon Ave, Cidade Universitária, Rio de Janeiro, RJ 21941-901, Brazil
jose.carvalho@engenharia.ufjf.br

Abstract. Multi-objective structural optimization problems (MOSOPs) with two or more objectives are extensively considered in the literature. Due to the great interest in solving these types of problems, several Multi-objective Evolutionary Optimization Algorithms (MOEAs) have been developed. They are applied to problems in several fields, mainly engineering. This paper compares multi-objective optimization algorithms based on swarm intelligence and applies them to solve structural optimization problems concerning three objectives. The objective functions are the weight, the natural frequencies of vibration, and the maximum nodal displacement, considering stress constraints. The design variables, discrete or continuous, are the cross-sectional areas of the bars. Some traditional benchmark problems in the literature in structural engineering applications were performed. Finally, Pareto sets are presented where a Multi Tournament Decision (MTD) method is adopted to extract the desired solutions from these problems.

Keywords: Multi-objective truss optimization, Particle Swarm Optimization, Structural optimization.

1 Introduction

It is common for designers to search for light and economic structures that meet safety criteria in structural engineering. The objective function is usually single in most of these optimization problems. However, this formulation may contain more objectives (multi-objective) and several constraints, which leads to constraints such as maximum displacements and stresses and minimum values for natural frequencies of vibration, among others.

Evolutionary Algorithms (EAs), especially population-based meta-heuristics, have grown in recent decades and successfully applied in the field of structural optimization. This kind of technique can be considered robust and free of derivatives of objective functions and constraints. Researchers have developed many meta-heuristics. The Particle Swarm Optimization (PSO) algorithm [1] is a widespread example of EA and provides computational models based on the concept of collective intelligence.

In this paper, multi-objective structural optimization problems are solved concerning the minimization of the mass of truss structures, the maximization of the first natural frequency of vibration, and the minimization of the maximum displacement, concerning discrete or continuous design variables. The axial stresses are the constraints. The computational experiment is on benchmark single-objective structural optimization problems widely discussed in the literature: the 10-bar truss. Four multi-objective PSO algorithms are used to solve the MOSOPs. Pareto fronts are used to show non-dominated solutions to optimization problems, and Multi-criteria decision-making (MCDM) is adopted to extract solutions from the Pareto front according to the decision maker's (DM) preferences.

The remainder of the paper is organized as follows. Section 2 describes the formulation of the MOSOPs. Section 3 presents the algorithms adopted in this paper. Performance indicators to evaluate the algorithms are described in Section 4. The computational experiments and some discussions are presented in Section 5. Finally, the paper ends with conclusions in Section 6.

2 Multi-objective optimization

In engineering, most decision problems are Multi-objective Structural Optimization Problem (MOSOPs) [2]. A MOSOP has some objective functions that are to be minimized (or maximized) simultaneously. Although single-objective structural optimization problems are commonly found in the literature, the formulation of optimization problems involving multiple objectives appears naturally due to two or more conflicting objectives.

The MOSOP discussed in this paper is formulated as:

$$\begin{aligned} &\min\ W(\mathbf{x}) \quad \text{and} \quad \max\ \omega_1(\mathbf{x}) \quad \text{and} \quad \min\ \text{maximum}(u_j(\mathbf{x})),\ j=1,...,n_{dof}, \\ &\text{subject to} \qquad \sigma_i(\mathbf{x}) \leq \overline{\sigma} \\ &\qquad\qquad \mathbf{x}^L \leq \mathbf{x} \leq \mathbf{x}^U, \end{aligned} \tag{1}$$

where $W(\mathbf{x})$ is the weight of the structure, $\omega_1(\mathbf{x})$ is the first natural frequency of vibration, $u_j(\mathbf{x})$ is the displacement at the j-th node, n_{dof} is the number of degree of freedom, and $\sigma_i(\mathbf{x})$ is the axial stress at the i-th bar. The design variables are $\mathbf{x} = \{A_1, A_2, ..., A_N\}$, where A_i are the sizing design variables indicating the cross-sectional areas of the N bars (continuous or discrete) that must be in the lower $\mathbf{x}^L$ and upper $\mathbf{x}^U$ bounds. $W(\mathbf{x})$ is written as:

$$W(\mathbf{x}) = \sum_{i=1}^{N} \rho A_i L_i, \tag{2}$$

where ρ is the specific mass of the material and L_i is the length of the i-th bar of the structure. $\omega_1(\mathbf{x})$ is obtained by the evaluation of the eigenvalues of the matrix

$$\left[(\omega_{m_f}^2 \times [M]) + [K]\right] = 0, \tag{3}$$

where $[M]$ is the mass matrix and ω_{m_f} are the equivalent eigenvalues with respect to the m_f natural frequencies of vibration of the structure [3]. The nodal displacements $\{u\}$ are obtained by the equilibrium equation for a discrete system of bars, which is written as:

$$[K]\{u_j(\mathbf{x})\} = \{p\}, \tag{4}$$

where $\{p\}$ are the load components.

3 Particle swarm algorithms

The particle swarm algorithms used to solve the MOSOPs formulated in this paper are the Multi-objective Craziness based Particle Swarm Optimization (MOCRPSO) [4], the Competitive Multi-Objective Particle Swarm Optimizer (CMOPSO) [5], a novel MOPSO algorithm with Multiple search strategies (MMOPSO) [6], and a novel MOPSO (NMPSO) [7].

MOCRPSO operates with a special code based on CRPSO [8] and incorporates a crowding distance mechanism, non-dominated solutions, and an external archive, together with a mutation operator based on the MOPSO-CD [9]. The algorithm was adopted in Carvalho [4] to solve benchmark and engineering problems. Its performance was discussed and compared with GDE3 [10], GDE3-APM [11], and MOA and MOAS [12] and achieved competitive results. The Adaptive Penalty Method (APM) [13] is the constraint-handling technique coupled to MOCRPSO. CMOPSO was recently proposed, and no additional storage is required to record the historical information in the search process, such that it does not need any external archive. The algorithm is verified by comparing it with six existing algorithms on 21 benchmark multi-objective optimization problems. The experimental results demonstrate that the CMOPSO shows significantly better overall performance than the compared algorithms in terms of both qualities of the solution set and convergence speed.

MMOPSO uses two search strategies to update the velocity of each particle, which is respectively beneficial for the acceleration of convergence speed and the keeping of population diversity. In addition, all the non-dominated solutions visited by the particles are preserved in an external archive, where an evolutionary search strategy is further performed to exchange useful information among them. MMOPSO was compared with some algorithms, and simulation results show that MMOPSO performs better on most test problems. The NMPSO algorithm uses a balanceable fitness estimation method and a novel velocity update equation to compose it. DTLZ and

WFG test suites with 4-10 objectives are used to assess its performance. They indicate that NMPSO has superior performance over four current algorithms considering four competitive multi-objective evolutionary algorithms when solving most of the test problems adopted.

4 Performance indicators of the particle swarm algorithms

To assess the quality of the algorithms, some performance indicators can be found in the literature, which usually considers two aspects: whether the obtained solutions are as close as possible to the Pareto front (convergence) and whether there is a widely spread distribution of solutions on it (diversity). In this study, the Hypervolume (HV) [14], and the Empirical Attainment Function (EAF) [15] are adopted to evaluate the performance of the algorithms. These were chosen due to their popularity and efficiency.

4.1 Hypervolume (HV)

HV metric was proposed by [14]. It did not require knowledge of the real Pareto front of the analyzed problem and to evaluate both convergence and diversity simultaneously. As its name implies, it provides the hypervolume of the space limited by the solutions of the non-dominated set (that belongs to a Pareto front) and a reference point.

The free code developed by [16][1] was used here to obtain the HV. For its evaluation, each objective function of the obtained Pareto front was normalized in the interval $[0, 1]$, and all the coordinates of the adopted reference point were designated equal to 1.

4.2 Empirical attainment function (EAF)

The Empirical Attainment Function (EAF), proposed by [15], returns a probability distribution of the results obtained by an algorithm concerning the values of the objective functions. It is a summary of the results of several executions of an algorithm. The EAF Best shows where an algorithm produces the best solutions in the objective space.

A computational code in C language provided in [17][2] was adopted to compute the EAF from non-dominated sets. EAF Best was adopted for three objective vectors as all feasible non-dominated solutions were obtained in all independent runs. [15] and [18] provided a discussion on how to obtain the EAF curves.

5 Computational experiments

This section assesses the performance of the algorithms, and the computational experiment refers to a well-known structural optimization problem, named a 10-bar truss for cases with continuous and discrete variables. The first objective is to minimize the structure's weight, the second is to maximize the first natural frequency of vibration, and the third is to minimize the maximum displacement, considering axial stresses as constraints.

The initial population was randomly generated considering the maximum number of objective function evaluations is 50000 (50 particles and 1000 generations). The number of independent runs is 20, and all of the presented solutions are rigorously feasible considering all algorithms. The MOCRPSO was developed using C language. Its parameters are: $c_1 = c_2 = 2.05$, $v^{craziness} = 0.001$, $Pcr = 0.5$, global neighborhood topology. The external file limit $ARQ = 500$. For all remaining algorithms was used the PlatEMO [19] to performed the test problem. PlatEMO is a MATLAB platform for MOEAs which includes more than 50 algorithms and more than 100 multi-objective test problems, along with several widely used performance indicators. The structure was analyzed by the Finite Element Method (FEM) [20] during the evolutionary process.

5.1 The 10-bar truss

A well-known structural multi-objective optimization problem, named as 10-bar truss [21], and depicted in Fig. 1, has ρ = 0.1 lb/in^3, and E = 10^4 ksi. Vertical downward loads of 100 kips are applied at nodes 2 and 4, and the stress in each bar is limited to $\pm$ 25 ksi. A non-structural mass of 1000 lb is attached to the free nodes. For the discrete case, the values of the cross-sectional areas are chosen from the set (in^2): $\{$1.62, 1.80, 1.99, 2.13, 2.38, 2.62, 2.63, 2.88, 2.93, 3.09, 3.13, 3.38, 3.47, 3.55, 3.63, 3.84, 3.87, 3.88, 4.18, 4.22, 4.49, 4.59, 4.80, 4.97,

[1] http://lopez-ibanez.eu/hypervolume.
[2] https://eden.dei.uc.pt/ cmfonsec/aft.html

5.12, 5.74, 7.22, 7.97, 11.50, 13.50, 13.90, 14.20, 15.50, 16.00, 16.90, 18.80, 19.90, 22.00, 22.90, 26.50, 30.00, 33.50}, resulting in 42 options. For the continuous case, the lower and upper bounds for the cross-sectional areas are defined by $[0.1; 40]$ (in^2).

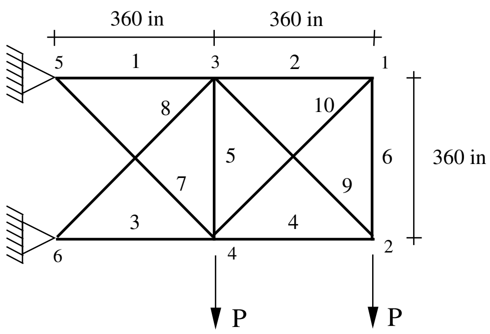
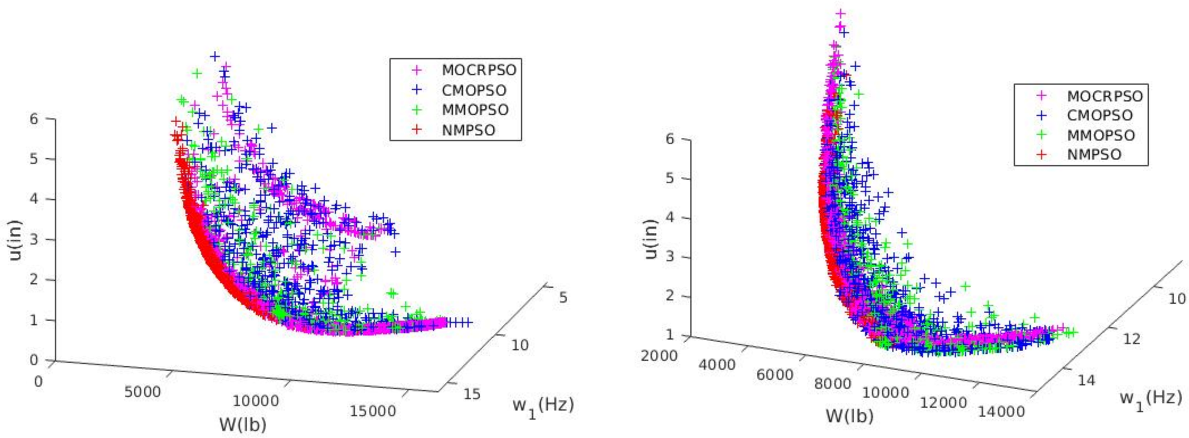


Figure 1. 10-bar truss, taken from [13].

5.2 Results and discussions

The results obtained for the MOSOPs are presented in Figures 2a and 2b, concerning the EAF_{best} obtained by MOCRPSO, CMOPSO, MMOPSO, and NMPSO algorithms for continuous and discrete cases, respectively.



(a) Continuous case

(b) Discrete case

Figure 2. EAF_{best} of the MOCRPSO, CMOPSO, MMOPSO, and NMPSO algorithms for the 10-bar truss.

The values of the normalized HVs are shown in Table 1. CMOPSO algorithm obtained the highest value of the HV for the continuous case. For the discrete case, MOCRPSO algorithm obtained the highest value. NMPSO algorithm obtained low values for the HV for both cases.

Table 1. HV values obtained for all independent runs.

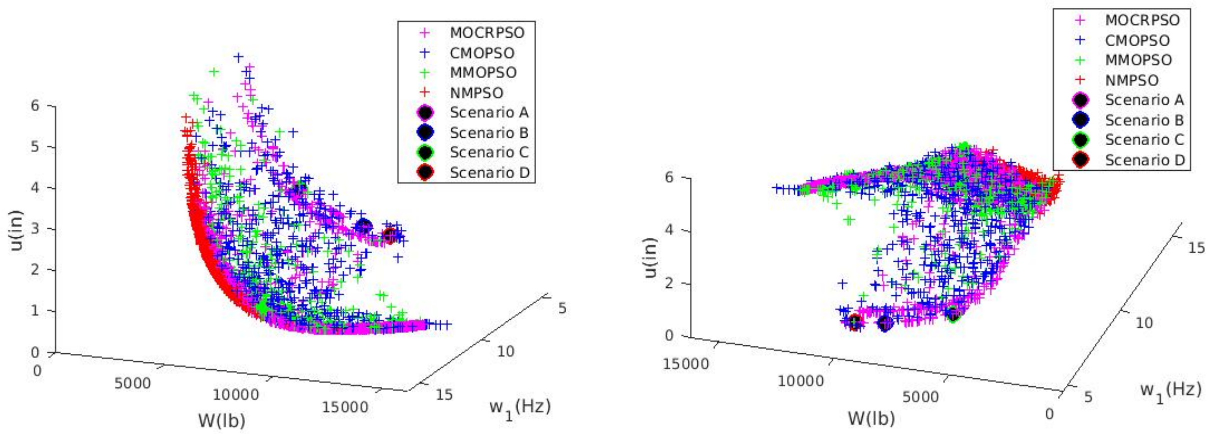
10-bar truss	MOCRPSO	CMOPSO	MMOPSO	NMPSO
Continuous	0.807732	**0.849822**	0.826196	0.673550
Discrete	**0.752447**	0.686654	0.688117	0.603232

The analysis takes into account all the information obtained by the Pareto fronts was performed. The DM has a nontrivial task of extracting a solution from the Pareto set. Based on that, a tournament-based method that ranks the best and the worst possible solutions in the Pareto set according to objectives and preferences (weights) established by the DM was proposed by Parreiras & Vasconcelos [22] and named as Multicriteria Tournament Decision (MTD) method. More details and pseudocode for the MTD can be found in [22].

This method is used in this study to find the best solutions, according to some importance. Four decision scenarios are used considering three criteria: (i) the weight, (ii) the first natural frequency, and (iii) the maximum displacement. The scenarios are described as follows:

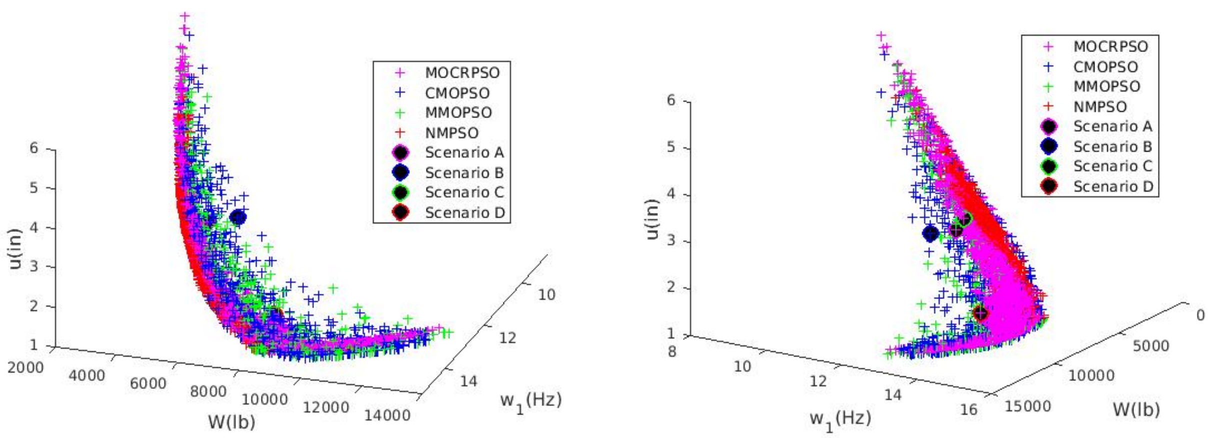
- Scenario A: all criteria have the same importance, i.e. $(w_1, w_2, w_3) = (0.3333, 0.3333, 0.3333)$.
- Scenario B: criterion (ii) is the most important and criteria (i) and (iii) have the same importance, i.e. $(w_1, w_2, w_3) = (0.2, 0.6, 0.2)$.
- Scenario C: criterion (i) is the most important and criteria (ii) and (iii) have the same importance, i.e. $(w_1, w_2, w_3) = (0.6, 0.2, 0.2)$.
- Scenario D: criterion (iii) is the most important and criteria (i) and (ii) have the same importance, i.e. $(w_1, w_2, w_3) = (0.2, 0.2, 0.6)$.

Figures 3a and 4a show the non-dominated solutions for the 10-bar truss continuous and discrete cases, respectively. The solid circles in these figures represents the solutions extracted by the MTD method corresponding to each scenario. Also, Figures 3b and 4b present the same data as the previous figures, however, with a different view to better expose the MTD solutions.



(a) (b)

Figure 3. MTD solutions of the MOCRPSO, CMOPSO, MMOPSO, and NMPSO algorithms for the 10-bar truss (continuous case).



(a) (b)

Figure 4. MTD solutions of the MOCRPSO, CMOPSO, MMOPSO, and NMPSO algorithms for the 10-bar truss (discrete case).

Table 2 provides the optimized design variables (dv) and their objective function values of the MTD extracted solutions from Figures 3 and 4, discrete and continuous cases, respectively, considering the four scenarios. From the complete extraction data, such as the values of the objective functions, the DM expands the range of solutions that are of interest more easily. If any of these are not within the DM expectations, new scenarios can be tested, and new solutions can be extracted. For instance, the DM may be interested in obtaining a solution that the importance for the frequency is higher, so a possible scenario would be $(w_1, w_2, w_3) = (0.1, 0.8, 0.1)$. Besides, the DM also may be interested that the weight and maximum displacement would be of high importance,, so $(w_1, w_2, w_3) = (0.45, 0.1, 0.45)$.

Table 2. Design variables (dv) and objective function values of the MTD solutions (Scenarios (Sc.) A, B, C, and D) of the 10-bar truss. $W(\mathbf{x})$ in lb, $\omega_1(\mathbf{x})$ in Hz, and $u(\mathbf{x})$ in inches.

	Continuous case				Discrete case			
dv	Sc. A	Sc. B	Sc. C	Sc. D	Sc. A	Sc. B	Sc. C	Sc. D
A_1	40.0000	40.0000	33.0891	40.0000	13.5000	5.7400	13.5000	14.2000
A_2	0.1000	0.1000	0.1000	0.1000	2.6200	3.1300	1.6200	1.6200
A_3	40.0000	40.0000	19.6707	40.0000	5.7400	5.1200	5.7400	14.2000
A_4	22.3460	22.3460	16.5185	40.0000	4.9700	7.2200	4.5900	14.2000
A_5	0.1000	0.1000	0.10337	0.1000	1.9900	1.6200	1.6200	1.6200
A_6	0.1000	0.1000	0.1000	0.1000	1.6200	1.8000	1.6200	1.6200
A_7	14.7962	14.7962	9.2151	18.1558	4.2200	4.4900	4.4900	4.8000
A_8	35.9561	35.9561	21.1957	40.0000	5.7400	7.2200	7.2200	14.2000
A_9	34.8341	34.8341	20.1569	40.0000	11.5000	13.9000	4.9700	14.2000
A_{10}	0.1000	0.1000	0.1000	0.1000	1.6200	3.1300	1.6200	1.6200
$W(\mathbf{x})$	8057.7036	8057.7036	5084.5312	9333.1702	4653.0546	5365.0168	4206.1557	7973.8278
$\omega_1(\mathbf{x})$	5.1355	5.1355	5.1055	5.1406	11.5561	11.1228	11.6008	13.3188
$u(\mathbf{x})$	1.2803	1.2803	2.0177	1.1417	2.4658	2.4148	2.6713	1.4060

6 Conclusions

This study presented structural optimization problems considering three conflicting objectives and compared four multi-objective algorithms based on swarm intelligence. The algorithms' performance was evaluated using a 10-bar truss take into account continuous and discrete search spaces.

A set of non-dominated solutions was extracted from the experiments, and their respect HVs were presented. The results indicated that MOCRPSO and CMOPSO algorithms achieved superior performance in the two analyzed cases.

An analysis was conducted to evaluate the Pareto set using an MTD method to allow the DM to indicate his preferences. Additionally, the values of the design variables and their objective functions obtained by the MTD were presented, showing the best solutions according to some criteria.

Future works intend to apply the algorithm for solving other test problems considering large-scale optimization problems.

Acknowledgements. The authors thank the Postgraduate Program in Civil Engineering (UFJF and UFRJ), Brazilian Agencies CNPq (grant 306186/2017-9), FAPEMIG (Grant TEC PPM-00174-18), and CAPES for the financial support.

References

[1] R. Eberhart and J. Kennedy. A new optimizer using particle swarm theory. In *Micro Machine and Human Science, 1995. MHS'95., Proceedings of the Sixth International Symposium on*, pp. 39–43. IEEE, 1995.

[2] L. S. d. Oliveira and S. F. Saramago. Multiobjective optimization techniques applied to engineering problems. *Journal of the brazilian society of mechanical sciences and engineering*, vol. 32, n. 1, pp. 94–105, 2010.

[3] K.-J. Bathe. *Finite element procedures*. Prentice Hall, Pearson Education, Inc., 2006.

[4] E. C. R. Carvalho. *Um algoritmo por enxame de partículas para a solução de problemas de otimização estrutural multiobjetivo considerando frequências naturais de vibração*. PhD thesis, Programa de Pós-graduação em Modelagem Computacional, Universidade Federal de Juiz de Fora, 2018.

[5] X. Zhang, X. Zheng, R. Cheng, J. Qiu, and Y. Jin. A competitive mechanism based multi-objective particle swarm optimizer with fast convergence. *Information Sciences*, vol. 427, pp. 63–76, 2018.

[6] Q. Lin, J. Li, Z. Du, J. Chen, and Z. Ming. A novel multi-objective particle swarm optimization with multiple search strategies. *European Journal of Operational Research*, vol. 247, n. 3, pp. 732–744, 2015.

[7] Q. Lin, S. Liu, Q. Zhu, C. Tang, R. Song, J. Chen, C. A. C. Coello, K.-C. Wong, and J. Zhang. Particle swarm optimization with a balanceable fitness estimation for many-objective optimization problems. *IEEE Transactions on Evolutionary Computation*, vol. 22, n. 1, pp. 32–46, 2016.

[8] R. Kar, D. Mandal, S. Mondal, and S. P. Ghoshal. Craziness based particle swarm optimization algorithm for fir band stop filter design. *Swarm and evolutionary computation*, vol. 7, pp. 58–64, 2012.

[9] C. R. Raquel and P. C. Naval Jr. An effective use of crowding distance in multiobjective particle swarm optimization. In *Proceedings of the 7th annual conference on Genetic and evolutionary computation*, pp. 257–264. ACM, 2005.

[10] S. Kukkonen and J. Lampinen. GDE3: The third evolution step of generalized differential evolution. In *2005 IEEE congress on evolutionary computation*, volume 1, pp. 443–450. IEEE, 2005.

[11] D. E. C. Vargas. *Um algoritmo de evolução diferencial com penalização adaptativa para otimização estrutural multiobjetivo*. PhD thesis, Programa de Pós-graduação em Modelagem Computacional, Universidade Federal de Juiz de Fora, 2015.

[12] J. S. Angelo, H. S. Bernardino, and H. J. Barbosa. Ant colony approaches for multiobjective structural optimization problems with a cardinality constraint. *Advances in Engineering Software*, vol. 80, pp. 101–115, 2015.

[13] A. C. Lemonge and H. J. Barbosa. An adaptive penalty scheme for genetic algorithms in structural optimization. *International Journal for Numerical methods in engineering*, vol. 59, n. 5, pp. 703–736, 2004.

[14] E. Zitzler and L. Thiele. Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach. *IEEE Transactions on Evolutionary Computation*, vol. 3, n. 4, pp. 257–271, 1999.

[15] C. M. Fonseca and P. J. Fleming. On the performance assessment and comparison of stochastic multiobjective optimizers. In *International Conference on Parallel Problem Solving from Nature*, pp. 584–593. Springer, 1996.

[16] C. M. Fonseca, L. Paquete, and M. López-Ibánez. An improved dimension-sweep algorithm for the hypervolume indicator. In *IEEE Congress on Evolutionary Computation, 2006 - CEC 2006*, pp. 1157–1163. IEEE, 2006.

[17] C. M. Fonseca, A. P. Guerreiro, M. López-Ibáñez, and L. Paquete. On the computation of the empirical attainment function. In *International Conference on Evolutionary Multi-Criterion Optimization*, pp. 106–120. Springer, 2011.

[18] C. A. C. Coello, G. B. Lamont, D. A. Van Veldhuizen, and others. *Evolutionary algorithms for solving multi-objective problems*, volume 5. Springer, 2007.

[19] Y. Tian, R. Cheng, X. Zhang, and Y. Jin. PlatEMO: A MATLAB platform for evolutionary multi-objective optimization. *IEEE Computational Intelligence Magazine*, vol. 12, n. 4, pp. 73–87, 2017.

[20] T. J. Hughes. *The finite element method: linear static and dynamic finite element analysis*. DoverPublications.com, 2012.

[21] R. A. Gellatly and L. Berke. Optimal structural design. Technical report, DTIC Document, 1971.

[22] R. Parreiras and J. Vasconcelos. Decision making in multiobjective optimization aided by the multicriteria tournament decision method. *Nonlinear Analysis: Theory, Methods & Applications*, vol. 71, n. 12, pp. e191–e198, 2009.