A New Multi-objective Optimization Algorithm inspired by Lichtenberg Figures Applied to Constrained Mechanical Engineering Problems

Pereira, J. L. J. [1], Francisco, M. B.[1], Cunha Jr, S. S. [1], Gomes, G. F. [1]

[1]*Mechanical Engineering Institute, University Federal of Itajubá*
1303, Av. BPS, 37500-903, Itajubá, Minas Gerais, Brazil
joaoluizjp@unifei.edu.br, Matheus_brendon@yahoo.com.br, guilhermefergom@gmail.com, sebas@unifei.edu.br.

Abstract. The optimization problems that must meet more than one objective are called multi-objective optimization problems and may present several optimal solutions. Classic optimization methods had their importance in the past, but they lost space for new algorithms that emerged with the advancement of computing, better able to deal with a greater number of variables, objectives and nonlinearities. Evolutionary and meta-heuristic algorithms are exponents in the literature as the main tool for solving complex multi-objective problems. This work presents the first multi-objective meta-heuristic registered as a computer program in Brazil. The Multi-objective Lichtenberg Algorithm is an optimizer inspired by the physical phenomenon of radial intra-cloud lightning and Lichtenberg Figures. The algorithm is tested and compared with other renowned algorithms (MOPSO and NSGA-II) in some Zitzler-Deb-Thiele test functions and is finally applied to constrained and unconstrained engineering problems: Welded Beam Design, Disc Brake and 4-bar-truss. MOLA proved to be a promising multi-objective optimization algorithm surpassing today's most renowned algorithms.

Keywords: Optimization, Meta-heuristics, Lichtenberg Figures, Diffusion Limited Aggregation

1 Introduction

Optimization is a highly desired engineering tool in human history. The first algorithms were what are now called classical methods, based on gradients, and this is how many engineers around the world are taught in undergraduate courses. However, with the advancement of technological and computing development, optimization methods have emerged and overcome challenges that classical methods could never solve. Optimization of non-linear, non-convex, multimodal, multi-dimensional and multi-objective problems has gained prominence with the application of meta-heuristics [1].

The meta-heuristics are an exponent of Artificial Intelligence in optimization problems and are currently the most used tools in complex engineering problems [2]. When the problem has more than one objective, it is called Multi-objective and the number of algorithms capable of solving them is even smaller. Most of the works found in the literature using multi-objective meta-heuristics use evolutionary or swarm algorithms. The most popular are NSGA-II and MOPSO [3].These types of algorithms are inspired by behaviors found in nature and each one had general parameters, such as population and number of iterations, and specific parameters that control these algorithms according to their way of creation. According to the no-free-lunch (NFL) theorem [4], none of them can be excellent at solving any type of problem.

This paper shows the first multi-objective meta-heuristic registered in Brazil at *Instituto Nacional da Propriedade Industrial* (INPI), the multi-objective version of the Lichtenberg Algorithm (LA). This algorithm is inspired by the physical phenomena of lightning storms and Lichtenberg figures. The details of the creation, validation and some applications can be found in Pereira *et al* [5]. The optimizer has been successfully applied in the identification of cracks [6], damage in composites [7] and in the design optimization of carbon fiber isogrid lower limb prosthesis considering single-objective optimization [8]. One of the reasons for LA's success is that it combines in a single optimizer a stochastic search based on population and trajectory, with shots of Lichtenberg Figures (LF) in the best solution from the previous iteration. At each iteration, this figure (known as a peculiar natural pattern - fractals) is fired with different sizes and rotations, being able to sweep from tiny regions to the entire search space.

The MOLA will be compared with MOPSO and NSGA-II in three functions of the set called Zitzler-Deb-Thiele (ZDT) test functions [9] and then applied to complex constrained and unconstrained engineering problems: Welded Beam Design, Disc Brake and 4-bar-truss. The manuscript is organized as follows: Section 2 brings a general theoretical background about Lichtenberg Algorithm. Section 3 presents the methodological procedure. Section 4 brings the results and discussions and section 5 brings the conclusions.

2 Lichtenberg Algorithm

A new meta-heuristic inspired by the physical phenomena of lightning storms and Lichtenberg Figures was recently created by Pereira *et al*. [5], which presents all the details of the creation of the Lichtenberg Algorithm (LA). Lichtenberg was the first to study the phenomenon of propagation of electric discharges in dielectric material, which leads the figure to have branched and tortuous aspects. According to Merrill [10], the impossibility of determining the resistances at each point, having the heterogeneities of the material, determines the random growth of the figure for each case, even for the same material and electrostatic conditions. Turner [11] suggests that LF can be built through a random growth process with many particles, forming a cluster. Due to its stochastic model, each execution of the algorithm can generate different figures. Therefore, the construction of the Lichtenberg Figure is entirely numerical.

Among some growth models found in the literature, the Diffusion Limited Aggregation (DLA) theory, proposed by Witten and Sander [12] [13], was used. A binary matrix (0 and 1) is built as a map and, in the center, a particle represented by the number one is fixed. The cluster is built by the values of the matrix that is one and the empty spaces have zero value. Each matrix element of value one is a particle of the cluster and the number of them (N_p) in the cluster is defined at the beginning of the program. The space for construction of the figure is defined by the creation radius (R_c) and from it the matrix is generated with line and column numbers equal to twice R_c (diameter).

Particles are randomly released across the matrix and if they reach the cluster that in the beginning was just a particle in the center, they have an S probability of fixing, also called the stickiness coefficient. This parameter controls the density of the cluster. The particle walks are plotted randomly, radially, and as if they were on a Cartesian plane from the center and settling down anywhere on the map, rounding off the position to a matrix element with line and column. At this point, it can be added only if there is another particle next to it confirmed by a lateral check. If it reaches a radius slightly larger than the R_c, it is exterminated and another one starts the random walk again. This happens until all the particles determined at the entrance N_p are contained in the cluster or until it reaches its limit of construction.

Each particle in the cluster can be transformed into locations on a Cartesian plane and the LF can be plotted at any size, slope or starting point. Then, the extracted figure is plotted in the exact size of the search space and its center in the center of it. At each iteration, this figure can be plotted with different sizes and rotations, selected at random. This is done as a measure to improve the exploration and exploitation capabilities of the algorithm, in addition to preventing a flawed reading of the search space.

Another parameter of the optimizer is the refinement (*ref*), an input parameter that can be from 0 to 1 and is a creator of a second LF (red) every iteration from 0 to 100% with the same size of the main LF (blue) (see Figure 3). This smaller scale figure improves local search. If *ref* = 0, only the global LF (blue) acts on the optimizer every iteration, the global one.

Not all LF points are used to compute the objective function(s), as the number of points used for this purpose or population (*pop*) is defined at the beginning of the algorithm. The LF points (that represents the population) are chosen throughout the LF structure and are represented graphically by black dots, all of which are always within the search space by means of a check. This form makes LA a hybrid algorithm as it merges two types of algorithms found in the literature: population and trajectory. This hybrid routine, not found in any meta-heuristics, brought to the algorithm a great capacity for exploitation and exploration.

The sixth parameter is switching factor (M), a parameter for changing the LF in the optimizer input data. This parameter can be set as zero, one or two. If one, a figure is generated when starting the program and the same figure is used in all the iterations of the execution. If it is 2, a new figure is generated at each iteration. If M is worth 0, a previously saved figure is used in the optimizer and no figure is generated. It is important to note that the generation of a Lichtenberg Figure can take about two minutes. However, if no figure is generated, the optimizer generates final solutions in less than a second. Finally, the number of iterations (N_{iter}).

All points evaluated in the search space generate solutions in the objective space and these solutions are compared using the Pareto dominance relationship, where the non-dominated solutions are kept in the solution space and the non-dominated ones are excluded. The set of non-dominated solutions for each iteration forms the current Pareto front of the problem, which tends to approach the real one through the iterations. The algorithm works considering all points of the current Pareto front as candidate points to plot LF's. At each iteration, one of these points are selected at random to plot a LF, generating in the variable space a forced search in the regions that present better values of objective functions.

Table 1 shows some recommendations for the parameter ranges to be used. Figure 1 shows the LA acting in the search space for a two-dimensional problem. Note that *d* is the number of variables in the problem.

Table 1. Recommended LA Parameters [5].

Parameter	Value
R_c	50 to 200
N_p	10^3 to 10^6
S	0 to 1
Pop	(10 to 40) × d
ref	0 to 1
M	0, 1 or 2
N_{iter}	10^2 to 10^3

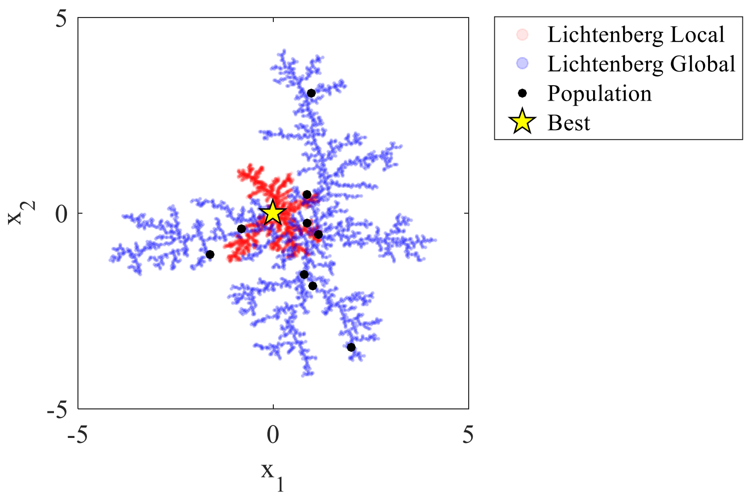


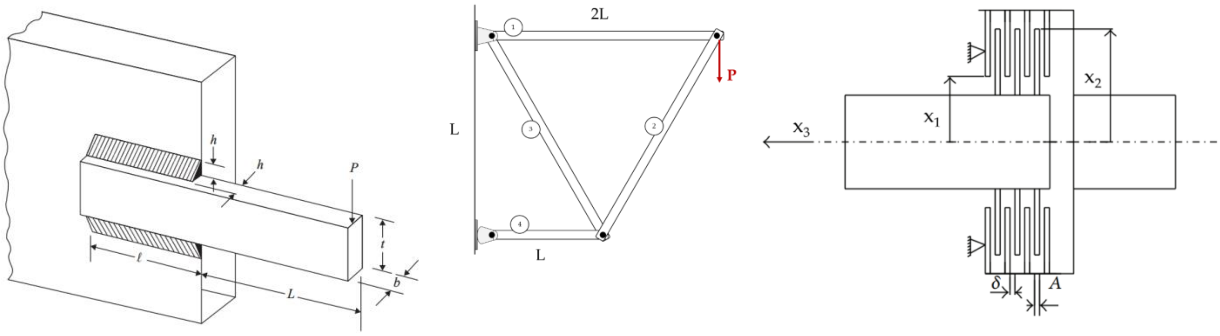
Figure 1. Population distribution in the Lichtenberg Figures (pop = 10 and ref = 0.3) [5]

3 Methodology

The algorithm will be tested on three test functions from the ZDT suite. The first (ZDT1) is a convex Pareto front; The second has a concave and the third has a Pareto front with several convex segments (discontinuous and disconnected). These functions are represented in Table 2. After being compared, the meta-heuristic will be applied to three design problems in mechanical engineering. The three can be found in Ray & Liew [14] and are represented in Figure 2.

The welded beam design works with four variables: width (x_1=h) and length (x_2=l) of the welded area, the depth (x_3=t) and the thickness (x_4=b) of the beam. The objectives are to minimize the total manufacturing cost (f_1) (Equation 1) and the deflection of the beam (f_2) (Equation 2) under the appropriate constraints (Equation 3) and search space (Equation 4).

In the 4-bar truss design problem the structural volume (f_1) (Equation 5) and displacement (f_2) (Equation 6) should be minimized subject only to the search space (Equation 7). There are four design variables related to cross sectional area of members 1, 2, 3 and 4.



a) Welded Beam Design b) Four bar truss design c) Disk Brake Design

Figure 2. Multi-objective Design Problems (Rao, 2009)

The disk brake design has two objectives: to minimize the stopping time (f_1) (Equation 8) and mass of a brake (f_2) (Equation 9). There are four design variables: the inner radius of the disk (x_1), the outer radius of the disk (x_2), the engaging force (area between the brake disc and the brake blocks) (x_3) and the number of friction surfaces (x_4) as well five constraints (Equation 10) and the search space in Equation 11.

Table 2. ZDT Test Functions

ZDT1	ZDT2	ZDT3
Minimize $f_1(x) = x_1$	Minimize $f_1(x) = x_1$	Minimize $f_1(x) = x_1$
$f_2(x) = g(x) \times h(f_1(x), g(x))$	$f_2(x) = g(x) \times h(f_1(x), g(x))$	$f_2(x) = g(x) \times h(f_1(x), g(x))$
Where $g(x) = 1 + \frac{9}{N-1} \sum_{I=2}^{N} x_i$	Where $g(x) = 1 + \frac{9}{N-1} \sum_{I=2}^{N} x_i$	Where $g(x) = 1 + \frac{9}{N-1} \sum_{I=2}^{N} x_i$
$h(f_1(x), g(x)) = 1 - \sqrt{f_1(x) / g(x)}$	$h(f_1(x), g(x)) = 1 - (f_1(x) / g(x))^2$	$h(f_1(x), g(x)) = 1 - \sqrt{\frac{f_1(x)}{g(x)}} - (\frac{f_1(x)}{g(x)}) \times \sin(10\pi f_1(x))$
$0 \le x_i \le 1, 1 \le i \le 30$	$0 \le x_i \le 1, 1 \le i \le 30$	$0 \le x_i \le 1, 1 \le i \le 30$

The parameters of the MOLA for these applications are: *Pop* = 50*;* N_{iter}=100*;* R_c = 200; N_p = 10^6*; S* = 1*; ref* = 0.4 *e M* = 0. The chosen values were found after an analysis of Design of Experiments (DOE) and Analysis of Variance (ANOVA) using weighted least-squares and considering 10 complex functions. The Model Indicator fit (R^2) of the analysis was more than 90%. Details and a full discussion of ways to tune the Lichtenberg Algorithm will be published coming soon.

$$\min f_1(x) = 1.1047 x_1^2 x_2 + 0.04811 x_3 x_4 (14 + x_2) \tag{1}$$

$$\min f_2(x) = 2.1952 / x_3^3 x_4 \tag{2}$$

$$g_1(X) = \tau(X) - 13600 \le 0 \tag{3.a}$$

$$g_2(X) = \sigma(X) - 30000 \le 0 \tag{3.b}$$

$$g_3(X) = x_1 - x_4 \le 0 \tag{3.c}$$

$$g_4(X) = 6000 - P_C(X) \le 0 \tag{3.d}$$

$$0.125 \le x_i \le 5, i = 1,4;\ 0.125 \le x_i \le 10.0, i = 2,3 \tag{4}$$

where:

$$\tau(X)=\sqrt{(\tau')^2+2\tau''\tau'\frac{x_2}{2R}+(\tau'')^2};\ \tau'(X)=\frac{P}{\sqrt{2}x_1x_2};\ \tau''(X)=\frac{MR}{J};\ M=P(L+\frac{x_2}{2});$$

$$R=\sqrt{\frac{x_2{}^2}{4}+(\frac{x_1+x_3}{2})^2};\ J=2\left\{\frac{x_1x_2}{\sqrt{2}}\left[\frac{x_2{}^2}{12}+(\frac{x_1+x_3}{2})^2\right]\right\};\ \sigma(X)=\frac{504000}{x_4x_3{}^2};$$

$$\delta(X)=\frac{4PL^3}{Ex_3{}^3x_4};$$

$$P_C(x)=\frac{4.013}{L^2}\left(1-\frac{x_3}{2L}\sqrt{\frac{E}{4G}}\right)\sqrt{(\frac{EGx_3{}^2x_4{}^6}{36})}$$

$$\min\ f_1(x)=200(2x_1+\sqrt{2x_2}+\sqrt{x_3}+x_4) \tag{5}$$

$$\min\ f_2(x)=0.01((2/x_1)+(2\sqrt{2}/x_2)-(2\sqrt{2}/x_3)+(2/x_4)) \tag{6}$$

$$1\le x_i\le 5, i=1,4;\ \ 1.4142\le x_i\le 3, i=2,3 \tag{7}$$

$$\min\ f_1(x)=4,9(10^{-5})(x_2^2-x_1^2)(x_4-1) \tag{8}$$

$$\min\ f_2(x)=(9,82(10^6)(x_2^2-x_1^2))/((x_2^3-x_1^3)x_4x_3) \tag{9}$$

$$g_1(X)=20+x_1-x_2 \tag{10.a}$$

$$g_2(X)=2,5(x_4+1)-30 \tag{10.b}$$

$$g_3(X)=x_3/(3,14(x_2^2-x_1^2)^2)-0,4 \tag{10.c}$$

$$g_4(X)=2,22(10^{-3})(x_3)(x_2^3-x_1^3)/(x_2^2-x_1^2)^2-1 \tag{10.d}$$

$$g_5(X)=900-(2,66(10^{-2})x_3x_4(x_2^3-x_1^3))/((x_2^2-x_1^2)) \tag{10.e}$$

$$55\le x_1\le 80;\ 75\le x_2\le 110;\ 1000\le x_3\le 3000;\ 2\le x_4\le 20 \tag{11}$$

4 Results and Discussions

The results for the ZDT functions can be seen in Figure 3. NSGA-II and MOPSO results were found in [3]. When it comes to meta-heuristics, their efficiency is measured by the exploration and exploitation capacity. The first refers to the ability of the algorithm to escape local minimums and better search other regions of the search space. The last one refers to the capacity of the meta-heuristic to improve the solution already found in terms of accuracy [15]. However, when it comes to multi-objective algorithms, in addition to these characteristics, the algorithms are evaluated for their convergence and coverage capacity. Convergence is to bring the Pareto front closer to the individual minima of each objective function. Coverage details the algorithm's ability to find very diverse and scattered solutions [03]. It can be seen in Figure 3 that the Lichtenberg Algorithm obtained excellent results in these aspects, since the solutions found practically overlapped the true Pareto front of the test functions.

The algorithm found better solutions than one of the most used algorithms in the literature for multi-objective optimization, the NSGA-II. And it had similar results to MOPSO, another widely used algorithm.

However, when the problem presented a disconnected Pareto front, MOLA fared better.

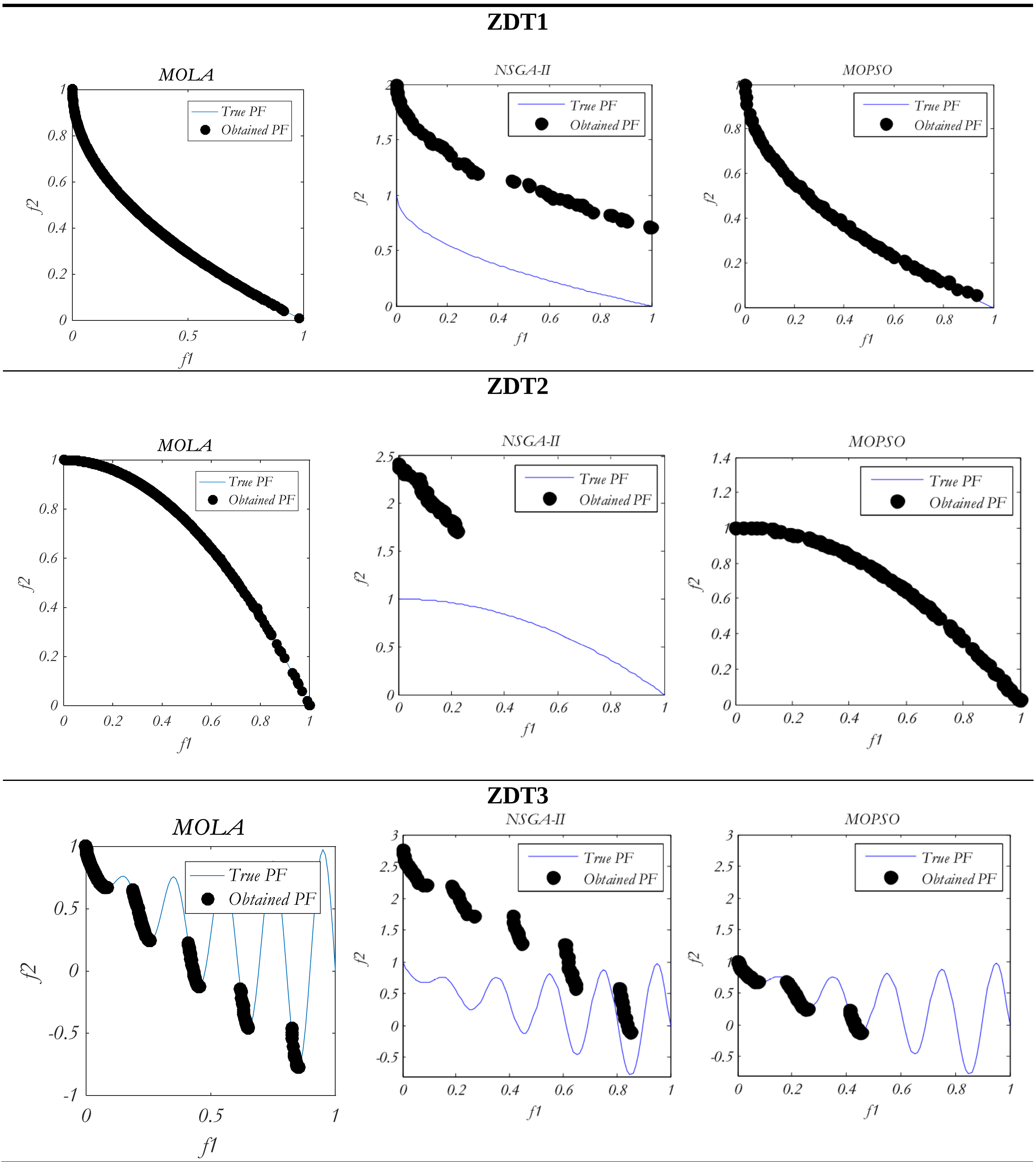
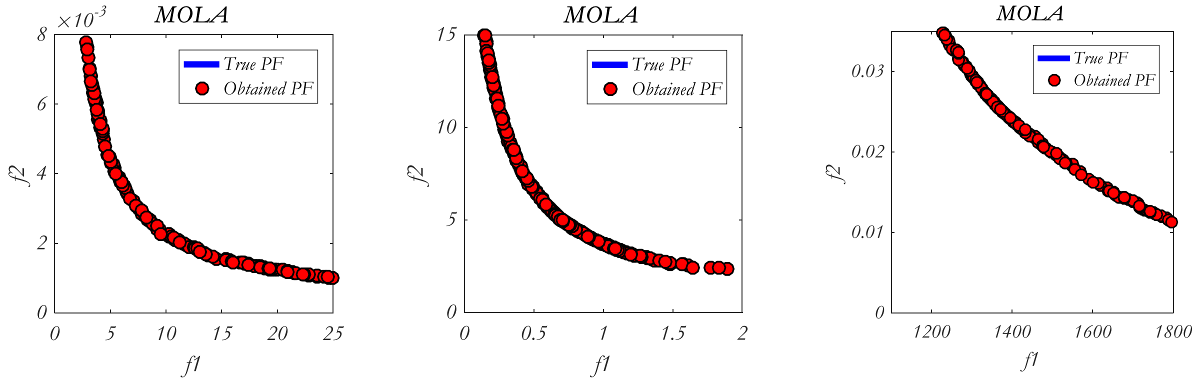


Figure 3. Pareto front found by MOLA, NSGA-II and MOPSO in ZDT test functions

In addition to these functions where the Multi-objective Lichtenberg Algorithm was tested, in an international paper in the process of publication it faced a larger battery of other meta-heuristics and more complex problems, with excellent results. Validated, the algorithm was applied to the three engineering problems mentioned above and the results are shown in Figure 4. It is noticed that the algorithm found a Pareto front that is practically coincident with the true one, with many solutions, making it suitable for solving multi-objective problems.



a) Welded Beam Design b) Disc Brake c) 4-bar-truss

Figure 4. Pareto front found in complex design problems

5 Conclusion

This paper discloses the creation of one of the first Brazilian meta-heuristics. In a tropical country where lightning is common, the Lichtenberg Algorithm is born inspired by the physical phenomenon of lightning and Lichtenberg figures. In this work it was applied to three common and varied multi-objective test functions and to 3 test problems in engineering.

It can be observed that the algorithm found solutions in all cases with good convergence and coverage, with a Pareto front that almost overlaps the Pareto front expected for the problem. The algorithm proved to be effective and with equivalent and even superior results than the NSGA - II and MOPSO in these results.

In this way, the Brazilian algorithm that honors Lichtenberg is able and prepared to face challenges and complex optimization applications. The algorithm code in the mono-objective version is available at [16].Where will soon have a multi-objective version as well.

Acknowledgements. The authors would like to acknowledge the financial support from the Brazilian agency CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico), CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior) and FAPEMIG (Fundação de Amparo à Pesquisa do Estado de Minas Gerais - APQ-00385-18). The authors would also like to acknowledge of the Tutorial Education Program (PET – Programa de Educação Tutorial).

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] Yang, Xin-She. Nature-inspired optimization algorithms. *Elsevier*, 2014.

[2] Mirjalili, S., Saremi, S., Mirjalili, S. M., & Coelho, L. dos S. (2016). Multi-objective grey wolf optimizer: A novel algorithm for multi-criterion optimization. *Expert Systems with Applications*, 47, 106–119.

[3] Mirjalili, S. Z., Mirjalili, S., Saremi, S., Faris, H., & Aljarah, I. (2018). Grasshopper optimization algorithm for multi-objective optimization problems. *Applied Intelligence*, 48(4), 805–820.

[4] Joyce T., Herrmann J.M. (2018) A Review of No Free Lunch Theorems, and Their Implications for Metaheuristic Optimisation. In: Yang XS. (eds) Nature-Inspired Algorithms and Applied Optimization. *Studies in Computational Intelligence*, vol 744. Springer, Cham.

[5] Pereira JLJ, Francisco MB, Diniz CA, Oliver GA, Cunha SS Jr, Gomes GF (2021). Lichtenberg Algorithm: A Novel Hybrid PHYSICS-Based Meta-Heuristic For Global Optimization. *Expert Systems with Applications*. https://doi.org/10.1016/j.eswa.2020.114522

[6] Pereira JLJ, Chuman M, Cunha SS Jr, Gomes GF (2020) Lichtenberg optimization algorithm applied to crack tip identification in thin plate-like structures. *Engineering & Computations*. https://doi.org/10.1108/ EC-12-2019-0564

[7] Pereira JLJ, Francisco MB, Cunha SS Jr, Gomes GF (2021b). A powerful Lichtenberg Optimization Algorithm: A damage identification case study. *Engineering Applications of Artificial Intelligence*. https://doi.org/10.1016/j.engappai.2020.104055

[8] Francisco, M. B., Pereira, J. L. J., Oliver, G. A., Da Silva, F. H. S., Cunha, S. S., Gomes, G. F. (2021). Multi-objective Design Optimization of CRP Isogrid Tubes Using SunFlower Multi-Objective Optimization Based on Metamodel. *Computers and Structures*.

[9] Zitzler, E., Deb, K., & Thiele, L. Comparison of Multi-objective Evolutionary Algorithms: Empirical Results. *Evolutionary Computation*, 8(2):173–195, Summer 2000.

[10] Merrill, F. H., and Von Hippel, A. (1939). The atomphysical Interpretation os Lichtenberg Figures and Their Application to the study of Gas Discharge. *Journal of Applied Physics*, 10(12), 873-887.

[11] Turner, A. From Lichtenberg to Lightning: Understanding Random Growth. *Newsletter of the London* Mathematical Society, 2019.

[12] Witten, T. A., Sander, L. M. Diffusion-limited aggregation, *Phys. Rev*. Lett. 27, 1983. p. 5687-5697.

[13] Witten, T. A., Sander, L. M. Diffusion-limited aggregation: a kinetic critical phenomenon, *Phys. Rev*. Lett. 47, 1981. p. 1400-1403.

[14] Ray T, Liew KM (2002) A swarm metaphor for multiobjective design optimization. *Eng Optim* 34:141–153

[15] Olorunda O, Engelbrecht AP. Measuring exploration/exploitation in particle swarms using swarm diversity. (2008) IEEE Congress on Evolutionary Computation (*IEEE World Congress on Computational Intelligence*), 2008. IEEE, pp 1128- 1134.

[16] Pereira, João Luiz Junho, et al (2021). Lichtenberg Algorithm (LA) (https://www.mathworks.com/matlabcentral/fileexchange/84732-lichtenberg-algorithm-la), MATLAB Central File Exchange. Retrieved June 17, 2021