



Professor Bruno Henrique Groenner

# Computação Evolucionária

## Análise de algoritmos

André de Aguiar Braga  
Andrey Gustavo de Souza

Lavras  
2017



## Sumário

<b>Introdução .....</b>	<b>2</b>
<b>Objetivo .....</b>	<b>5</b>
<b>Metodologia.....</b>	<b>6</b>
<b>Resultados e discussão .....</b>	<b>7</b>
<b>Função Ackley .....</b>	<b>7</b>
<b>Função Eggholder .....</b>	<b>8</b>
<b>Função Langermann .....</b>	<b>9</b>
<b>Função Schaffer #2.....</b>	<b>10</b>
<b>Conclusão .....</b>	<b>12</b>
<b>Referências Bibliográficas .....</b>	<b>13</b>

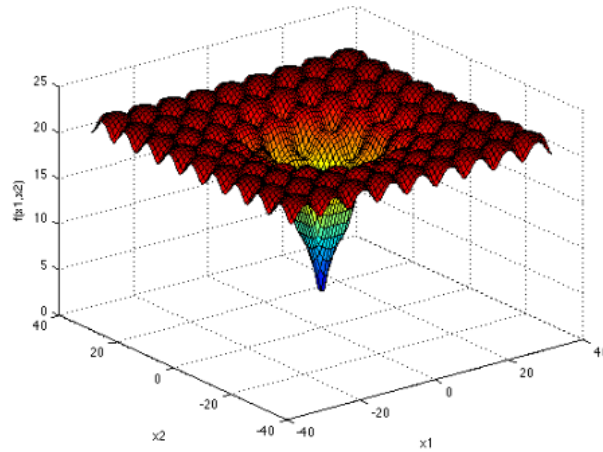
## Introdução

Neste trabalho é feita a análise de diferentes algoritmos de otimização bioinspirados. Os algoritmos a serem analisados foram apresentados nas aulas teóricas da disciplina. Por esse motivo, não é feita uma introdução teórica aprofundada a respeito dos mesmos, sendo apresentadas somente sua inspiração biológica.

- Algoritmo genético (GA)- Inspirado na Teoria da Evolução de Darwin, em que mutações genéticas geram indivíduos mais aptos de forma a passarem as características a suas crias.
- Particle Swarm Optimization (PSO)- A inspiração desse método vem da movimentação de grupos de animais, tais como a revoada de pássaros que tendem a seguir seus caminhos por boas áreas do ambiente.
- Teaching Learning Based Optimization (TLBO)- Baseado na interação aluno-professor presente nas salas de aula
- Sistemas Imunológicos Artificiais (CLONALG)- A ideia de sua implementação é baseada na Teoria da Seleção Clonal de imunidade adquirida em relação às definições de infecção, anticorpos, antígenos e etc.
- Algoritmo de Evolução Diferencial (DE)- Trabalha com a ideia de mutação e evolução.

No trabalho são utilizadas 4 diferentes funções de otimização utilizadas para analisar a solução encontrada por cada um dos 5 algoritmos citados.

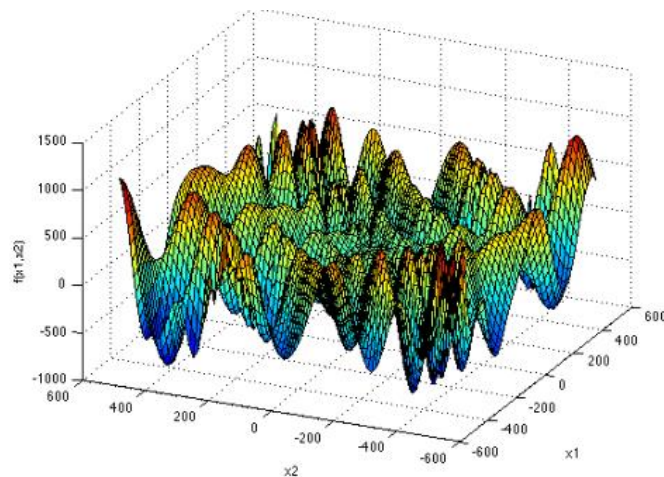
- Ackley- Apresenta mínimo global igual a 0, nos pontos (0,0).



$$f(\mathbf{x}) = -a \exp \left( -b \sqrt{\frac{1}{d} \sum_{i=1}^d x_i^2} \right) - \exp \left( \frac{1}{d} \sum_{i=1}^d \cos(cx_i) \right) + a + \exp(1)$$

Figura 1- Função Ackley

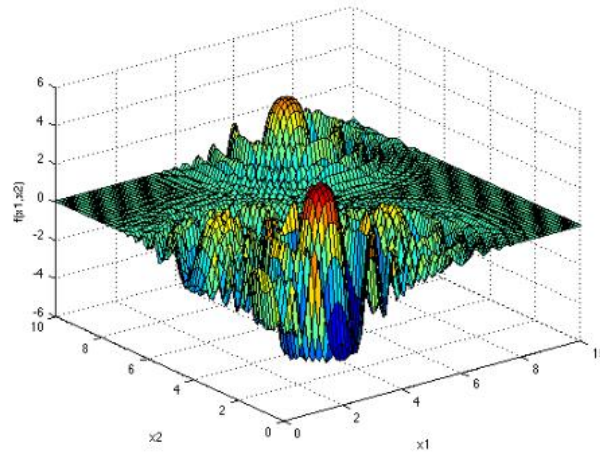
- Eggholder- De difícil otimização, apresenta mínimo global igual a - 959.6407 em (512,404.2319).



$$f(\mathbf{x}) = -(x_2 + 47) \sin \left( \sqrt{\left| x_2 + \frac{x_1}{2} + 47 \right|} \right) - x_1 \sin \left( \sqrt{|x_1 - (x_2 + 47)|} \right)$$

Figura 2- Função Eggholder

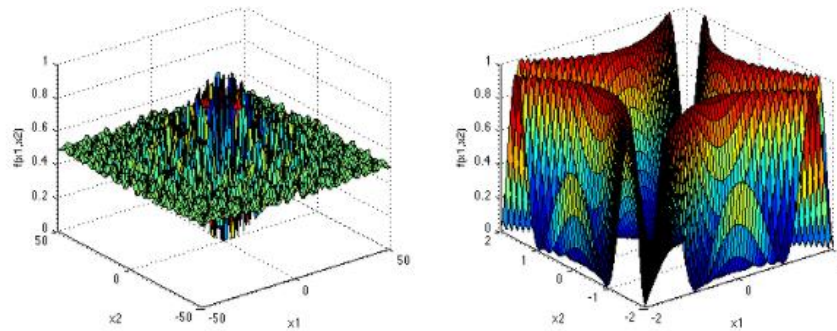
- Langermann- Seu mínimo global não é informado.



$$f(\mathbf{x}) = \sum_{i=1}^m c_i \exp \left( -\frac{1}{\pi} \sum_{j=1}^d (x_j - A_{ij})^2 \right) \cos \left( \pi \sum_{j=1}^d (x_j - A_{ij})^2 \right)$$

Figura 3- Função Langermann

- Schaffer- A segunda função Schaffer tem mínimo global igual a 0 em (0,0).



$$f(\mathbf{x}) = 0.5 + \frac{\sin^2(x_1^2 - x_2^2) - 0.5}{[1 + 0.001(x_1^2 + x_2^2)]^2}$$

Figura 4- Função Schaffer

Nas seções seguintes são apresentados os objetivos, metodologia e também os resultados e conclusão a respeito do trabalho.

## **Objetivo**

O presente trabalho tem como objetivo analisar os diferentes algoritmos bioinspirados, citados na seção anterior, quanto à sua capacidade de encontrar baixos custos para as funções de otimização utilizadas.

Além disso, objetiva-se discutir a respeito da particularidade de cada algoritmo em relação a cada uma das funções, fazendo uma comparação entre os mesmos.

## **Metodologia**

Os algoritmos bioinspirados foram programados em Python de maneira que a cada iteração fossem guardados os resultados obtidos. Dessa forma, ao final de cada repetição da execução do algoritmo é possível analisar a evolução dos resultados encontrados pelo mesmo.

Uma vez executadas 50 vezes cada algoritmo, cada um com 500 gerações e com população de 100 indivíduos. Realizou-se a análise estatística dos resultados de maneira a possibilitar a comparação da eficiência dos mesmos de acordo com cada uma das funções custo utilizadas.

Os resultados obtidos, juntamente com os gráficos comparativos são apresentados posteriormente neste documento.



## Resultados e discussão

### Função Ackley

A Tabela 1 apresenta os valores médios, máximos e mínimos do custo da função Ackley, cujo objetivo foi alcançar o valor mínimo. Todas as funções, com a exceção do Algoritmo Genético atingiram o mesmo patamar mínimo, alguns em todas as tentativas.

O desempenho inferior do algoritmo genético pode ser explicado pela representação binárias dos valores de entrada, uma vez que um número reduzido de bits pode reduzir o alcance dos valores, em valores tendendo à zero. A Figura 5 apresenta o comportamento de cada função em cada geração, com destaque para os algoritmos Clonalg e TLBO que apresentaram convergência rápida.

Tabela 1 - Valores de Custo para Função Ackley

Algoritmo	Média $\pm$ Desvio Padrão	Mínimo	Máximo
Clonalg	$4.4409 \times 10^{-16} \pm 0$	$4.4409 \times 10^{-16}$	$4.4409 \times 10^{-16}$
Evolução Diferencial	$4.4409 \times 10^{-16} \pm 0$	$4.4409 \times 10^{-16}$	$4.4409 \times 10^{-16}$
Algoritmo Genético	$0.2282 \pm 0.78$	0.0020	3.5745
PSO	$1.6520 \times 10^{-15} \pm 1.86 \times 10^{-15}$	$4.4409 \times 10^{-16}$	$7.5495 \times 10^{-15}$
TLBO	$4.4409 \times 10^{-16} \pm 0$	$4.4409 \times 10^{-16}$	$4.4409 \times 10^{-16}$

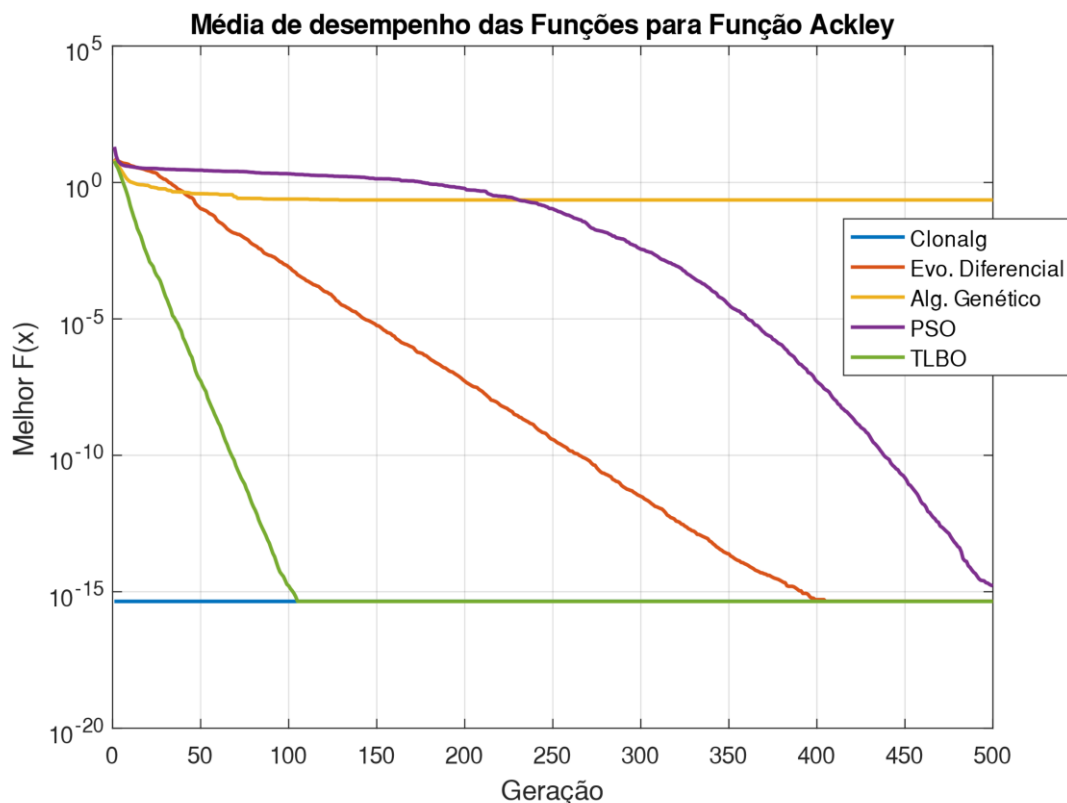


Figura 5 - Custo por Geração

## Função *Eggholder*

A Tabela 2 apresenta os valores médios, máximos e mínimos do custo da função *Eggholder*, cujo objetivo foi alcançar o valor mínimo. Todas as funções atingiram o mesmo patamar mínimo, alguns em todas as tentativas. O desempenho com maior variação na resposta foi a do algoritmo genético, apesar de apresentar resultados satisfatórios.

A Figura 6 apresenta o comportamento de cada função em cada geração. Para esta função, todos os algoritmos analisados obtiveram boa convergência.

Tabela 2 - Valores de Custo para Função *Eggholder*

Algoritmo	Média $\pm$ Desvio Padrão	Mínimo	Máximo
Clonalg	$-958.097 \pm 0.9494$	-959.432	-955.66
Evolução Diferencial	$-959.6407 \pm 0$	-959.6407	-959.6407
Algoritmo Genético	$-911.4271 \pm 46.2887$	-959.6407	-75.9713
PSO	$-959.6407 \pm 0$	-959.6407	-959.6407
TLBO	$-959.6407 \pm 0$	-959.6407	-959.6407

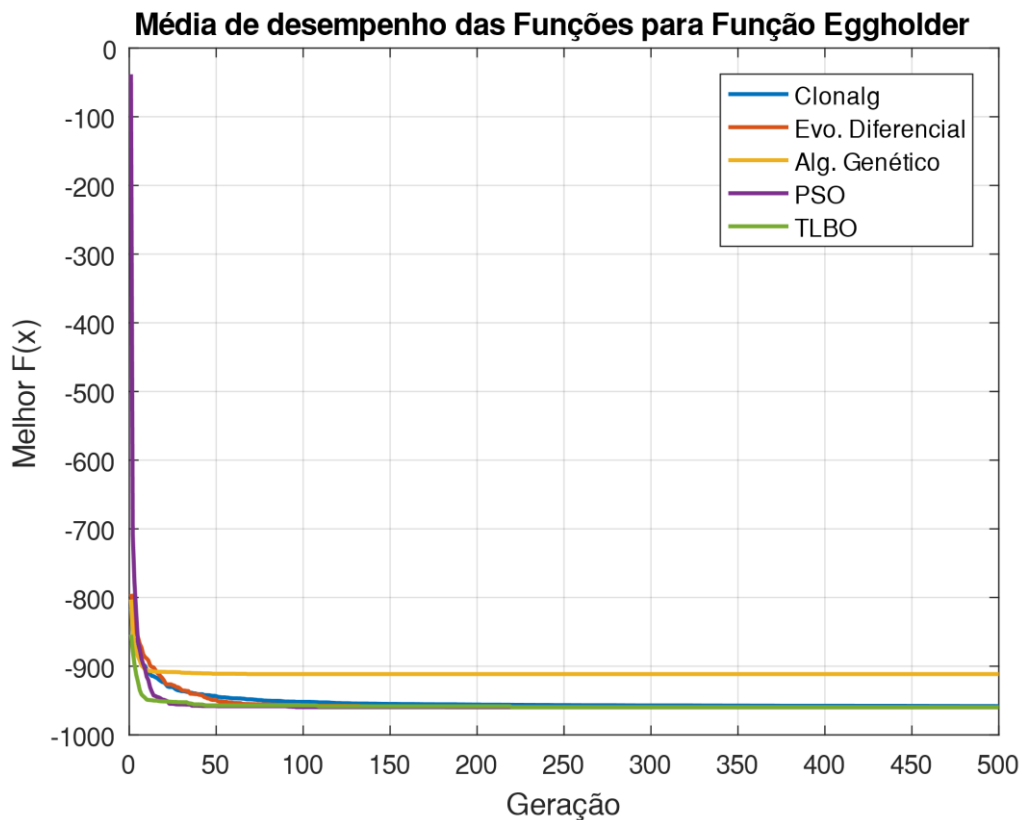


Figura 6 - Custo por Geração

## Função *Langermann*

A Tabela 3 apresenta os valores médios, máximos e mínimos do custo da função *Langermann*, obtidos por meio dos algoritmos abaixo mostrados. Todas as funções atingiram o mesmo patamar mínimo, alguns em todas as tentativas. O desempenho com maior variação na resposta foi a do algoritmo genético, apesar de apresentar resultados satisfatórios.

A Figura 7 apresenta o comportamento de cada função em cada geração. Para esta função, todos os algoritmos analisados obtiveram boa convergência.

Tabela 3 - Valores de Custo para Função *Langermann*

Algoritmo	Média $\pm$ Desvio Padrão	Mínimo	Máximo
Clonalg	$-5.1180 \pm 0.0429$	-5.1621	-4.9510
Evolução Diferencial	$-5.1621 \pm 0$	-5.1621	-5.1621
Algoritmo Genético	$-4.7451 \pm 0.7923$	-5.1621	-2.6679
PSO	$-5.1621 \pm 0$	-5.1621	-5.1621
TLBO	$-5.1621 \pm 0$	-5.1621	-5.1621

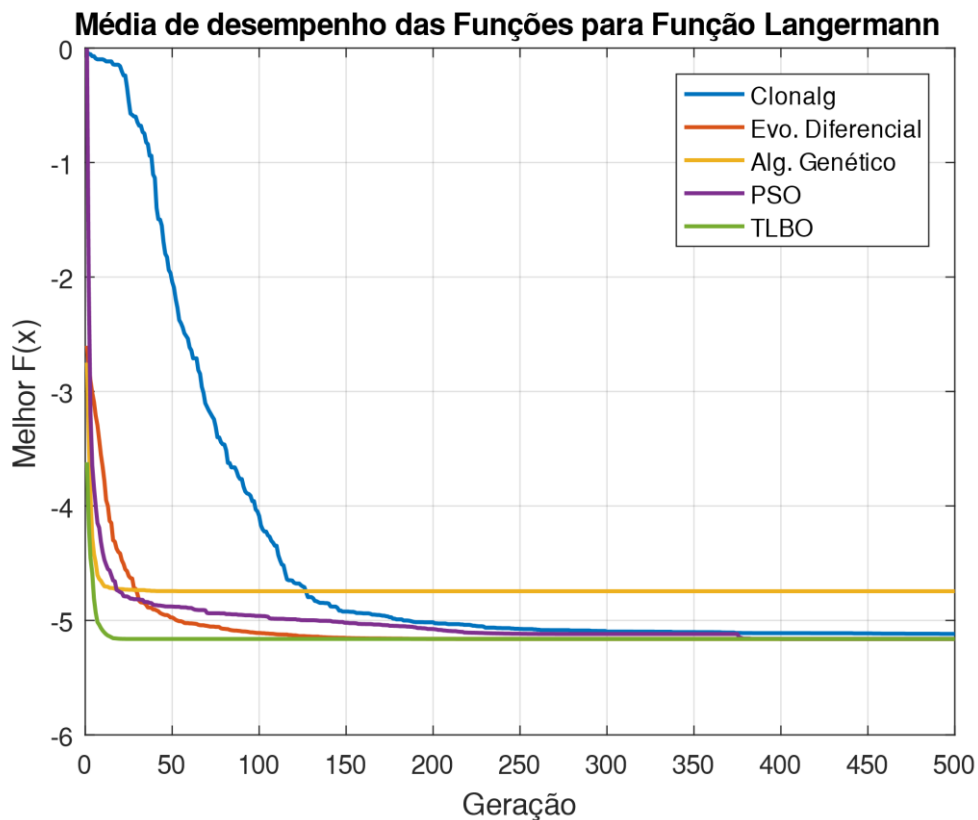


Figura 7 - Custo por Geração

## Função Schaffer #2

A Tabela 4 apresenta os valores médios, máximos e mínimos do custo da função *Schaffer #2*, obtidos por meio dos algoritmos abaixo mostrados. Todas as funções atingiram o mesmo patamar mínimo, alguns em todas as tentativas. O desempenho com maior variação na resposta foi a do algoritmo genético, apesar de apresentar resultados satisfatórios.

A Figura 8 apresenta o comportamento de cada função em cada geração num gráfico semilog, onde é possível observar que os algoritmos TLBO e PSO atingiram o valor mínimo de 0. Na Figura 9 observa-se que o algoritmo Clonalg obtém uma convergência rápida, assim como o TLBO e genéticos, este segundo, porém tem o maior desvio padrão dentre os algoritmos testados, o que resulta em uma média dos valores mínimos inferior aos demais.

Tabela 4 - Valores de Custo para Função Schaffer #2

Algoritmo	Média $\pm$ Desvio Padrão	Mínimo	Máximo
Clonalg	$0 \pm 0$	0	0
Evolução Diferencial	$3.627 \times 10^{-8} \pm 5.325 \times 10^{-8}$	$9.277 \times 10^{-11}$	$2.738 \times 10^{-7}$
Algoritmo Genético	$0.0185 \pm 0.0272$	$4.657 \times 10^{-9}$	0.1700
PSO	$0 \pm 0$	0	0
TLBO	$0 \pm 0$	0	0

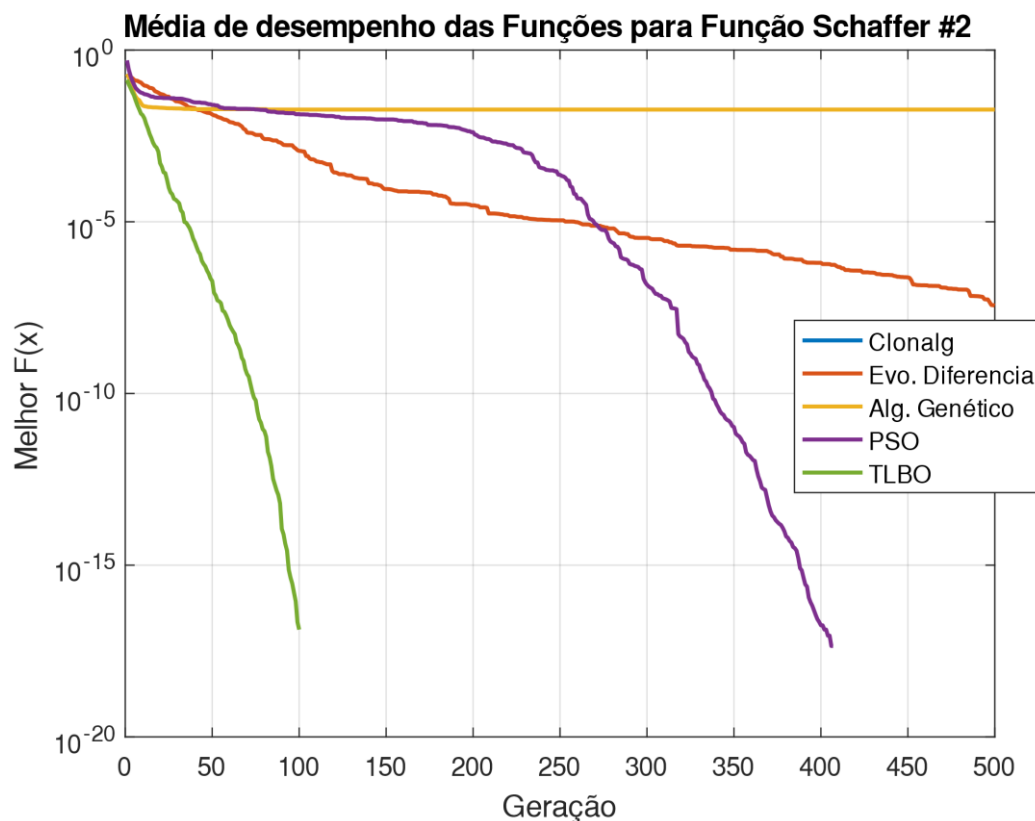


Figura 8 - Custo por Geração - Semilog

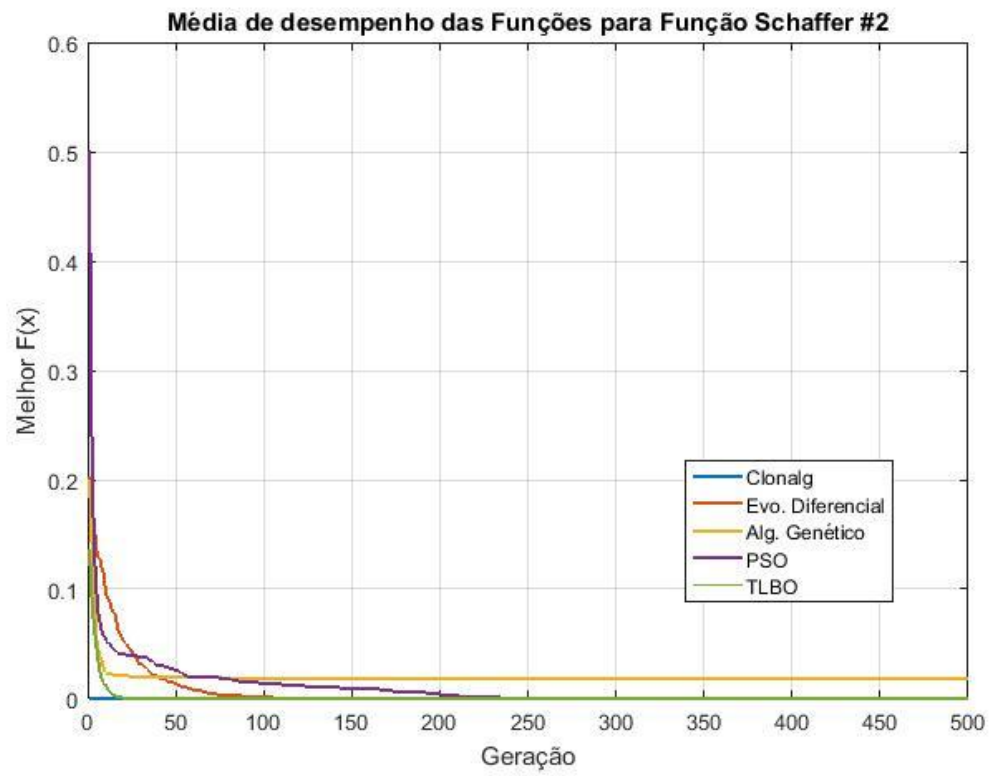


Figura 9 - Custo por Geração

## **Conclusão**

O presente trabalho proporcionou aos envolvidos a possibilidade de analisar de maneira mais aprofundada o funcionamento de diferentes algoritmos bioinspirados. Observa-se que os resultados encontrados por eles são similares, de maneira que a escolha do melhor algoritmo deve, também, levar em consideração o custo computacional de cada um deles. Essa similaridade das respostas faz com que a visualização da análise de variância (ANOVA) seja comprometida, sendo esse o motivo da não utilização das imagens desse teste no presente documento. Destaca-se que a análise a respeito do custo computacional não foi feita nesse trabalho, uma vez que se objetivou a obtenção do menor custo computacional possível, desconsiderando o tempo para que esse custo fosse alcançado.

De maneira geral, a análise dos resultados obtidos por cada método de otimização utilizado foi realizado de maneira satisfatória, sendo capaz de proporcionar aos envolvidos a visualização das particularidades e dificuldades de implementação de cada um dos algoritmos bioinspirados utilizados.

## **Referências Bibliográficas**

- [1] Brownlee, J. Clever Algorithms. Nature-Inspired Programming Recipes, 2011. Disponível em <<http://cleveralgorithms.com/nature-inspired/index.html>>
- [2] <<https://www.sfu.ca/~ssurjano/optimization.html>> Acesso em 26/11/2017
- [3] Eiben, A. E.; Smith, J. E. Introduction to Evolutionary Computing. New York: Springer, 2007. 300p.