

# Trabalho Computacional 2: Ajuste de Hiperparâmetros do *Particle Swarm Optimization*

Pedro Vinícius A. B. Venâncio

**Resumo**—A busca por valores de hiperparâmetros apropriados para algoritmos evolutivos e algoritmos de aprendizado de máquina é um dos desafios mais persistentes na área de Inteligência Computacional. Já é de conhecimento comum que bons valores de hiperparâmetros são essenciais para um bom desempenho dos modelos, no entanto, tais valores ainda são selecionados na prática por convenções ou pelo método de tentativa e erro, cujas comparações experimentais são realizadas em uma escala bastante limitada. Este trabalho propõe um estudo acerca do ajuste de hiperparâmetros no contexto da Computação Evolucionária a partir do algoritmo *Particle Swarm Optimization* (PSO) em duas versões distintas. A fim de investigar a sensibilidade do algoritmo perante as alterações nos hiperparâmetros, a análise foi conduzida em variantes de duas funções mono-objetivos clássicas: *Sphere Function* (unimodal) e *Rotated Rosenbrock's Function* (multimodal).

**Palavras-chave**—Ajuste de hiperparâmetros, enxame de partículas, otimização mono-objetivo.

## I. INTRODUÇÃO

Nos últimos anos, com a popularidade das técnicas de *Deep Learning*, a dicotomia entre os termos "parâmetro" e "hiperparâmetro" se tornou ainda mais evidente. Parâmetros estão relacionados às variáveis internas do modelo, como é o caso dos filtros que são aprendidos em redes neurais convolucionais. Os hiperparâmetros, por sua vez, tratam-se de variáveis externas e, portanto, podem ser definidas a priori pelo usuário, como o tamanho da população e as taxas de mutação e cruzamento em algoritmos genéticos.

A importância da otimização dos hiperparâmetros vem sendo ressaltada principalmente em competições e desafios de algoritmos de Inteligência Computacional, como os disponíveis na plataforma Kaggle. Casos em que um ajuste eficiente dos hiperparâmetros traria mudanças consideráveis no ranking de classificação de uma competição são cada vez mais recorrentes e reivindicam uma maior atenção em prol dessa temática.

No que tange o *Particle Swarm Optimization* (PSO), algoritmo evolutivo bastante utilizado para casos em que os métodos baseados em gradiente são inviáveis ou apresentam déficit de desempenho, diversos são os hiperparâmetros que requerem um ajuste adequado. Desde a proposta inicial do PSO por Kennedy e Eberhart em 1995 [1], inúmeras outras versões foram sendo elaboradas para sintonizar os hiperparâmetros existentes e ainda propor novos, de modo a melhorar o desempenho frente a aplicações mais específicas.

Sob essa perspectiva, o presente trabalho tem como objetivo implementar o algoritmo *Particle Swarm Optimization* em duas de suas vertentes de busca: local (*lbest PSO*) e global (*gbest PSO*) e, a partir dessas configurações, otimizar seus

respectivos hiperparâmetros para realizar análises de sensibilidade e comparações de desempenho. Os dois problemas mono-objetivos escolhidos para os experimentos originaram do desafio *Competition on Niching Methods for Multimodal Function Optimization* do *Congress on Evolutionary Computation* de 2013 (CEC2013) [2].

## II. PARTICLE SWARM OPTIMIZATION (PSO)

O *Particle Swarm Optimization* (PSO) é um método evolutivo para resolver problemas de otimização computacionalmente difíceis e foi inspirado no comportamento de um grande bando de pássaros à procura de um alvo, no qual é atingido por um esforço mútuo, competitivo e cooperativo. Em virtude desse princípio, o conjunto de soluções candidatas (população) é comumente denominado de nuvem ou enxame e as soluções candidatas (indivíduos) de partículas.

Seja uma população com  $m$  indivíduos, tem-se que a posição de uma dada partícula  $i$  no tempo  $t$  é denotada por  $\mathbf{x}_i(t) = [x_{i,1}, x_{i,2}, \dots, x_{i,D}]$ , cuja inicialização é  $\mathbf{x}_i(0) \sim U(\mathbf{x}_{\min}, \mathbf{x}_{\max})$  [3]. Em toda iteração a posição de cada partícula é atualizada conforme a Equação (1):

$$\mathbf{x}_i(t+1) = \mathbf{x}_i(t) + \mathbf{v}_i(t+1) \quad (1)$$

onde  $\mathbf{v}_i(t+1)$  indica a velocidade da partícula  $i$  no instante de tempo  $t+1$ . A velocidade é representada por  $\mathbf{v}_i(t) = [v_{i,1}, v_{i,2}, \dots, v_{i,D}]$  e sua atualização no instante seguinte é dada pela Equação (2):

$$\mathbf{v}_i(t+1) = \chi[w\mathbf{v}_i(t) + c_1\mathbf{r}_1 \circ (\mathbf{p}_i(t) - \mathbf{x}_i(t)) + c_2\mathbf{r}_2 \circ (\mathbf{g}_i(t) - \mathbf{x}_i(t))] \quad (2)$$

onde  $\chi$  é o parâmetro de restrição,  $w$  é o peso de inércia,  $\mathbf{p}_i(t)$  é a melhor posição que a partícula  $i$  adquiriu até o instante  $t$ ,  $\mathbf{g}_i(t)$  é a melhor posição encontrada na vizinhança da partícula  $i$  até o instante  $t$  e  $c_1$  e  $c_2$  são os coeficientes de aceleração, cujas proporções no instante  $t$  são definidas pelos vetores aleatórios  $\mathbf{r}_1(t), \mathbf{r}_2(t) \sim U(0, 1)$ . As inicializações das velocidades e posições foram, respectivamente,  $\mathbf{v}_i(t=0) = 0$  e  $\mathbf{p}_i(t=0) = \mathbf{x}_i(0) \forall i \in [1, D]$ .

O primeiro termo da Equação (2) ( $w\mathbf{v}_i(t)$ ) se refere à componente inercial e, portanto, serve como uma memória da direção anterior, evitando que a partícula mude drasticamente de direção. O segundo termo ( $c_1\mathbf{r}_1 \circ (\mathbf{p}_i(t) - \mathbf{x}_i(t))$ ), por sua vez, atua como uma componente cognitiva, de modo que as partículas tendam às suas melhores posições. Por fim, o terceiro termo ( $c_2\mathbf{r}_2 \circ (\mathbf{g}_i(t) - \mathbf{x}_i(t))$ ) quantifica o desempenho da partícula  $i$  em relação a um grupo de partículas e, assim, cada partícula também é atraída para a melhor posição encontrada no grupo (componente social) [3].

Diversas modificações no PSO clássico foram desenvolvidas para melhorar a velocidade de convergência e a qualidade das soluções encontradas, inclusive, a versão descrita é denominada *Constricted PSO*. Dentre essas sugestões de melhorias, considerou-se ainda a técnica *Velocity Clamping* na implementação do algoritmo para o presente trabalho [4]. Tal estratégia consiste no controle das velocidades, visto que a atualização da posição pode atingir valores extremamente grandes em um período curto de tempo, tornando o comportamento da partícula mais oscilatório e, portanto, mais propenso a saltar regiões ótimas. Assim, se uma atualização da posição de uma partícula exceder uma velocidade máxima especificada, essa velocidade é limitada através da Equação (3) [4]:

$$v_{ij}(t+1) = \begin{cases} v_{ij}(t+1) & \text{se } v_{ij}(t+1) < V_{\max,j} \\ V_{\max,j} & \text{se } v_{ij}(t+1) \geq V_{\max,j} \end{cases} \quad (3)$$

tal que  $V_{\max,j} = \delta \times (x_{\max,j} - x_{\min,j})$ , onde  $\delta$  corresponde a proporção das diferenças entre os extremos de cada coordenada  $j$  que será definida como velocidade limite e está compreendido no intervalo  $\delta \in [0, 2; 1, 0]$ .

#### A. Global Best

No caso em que a vizinhança é definida como a população inteira, todas as partículas são atraídas para a melhor posição encontrada pelo enxame, o que implica, consequentemente, em uma convergência mais rápida. Essa versão foi nomeada de *Global Best Particle Swarm Optimization (gbest PSO)* e sua topologia normalmente é representada por uma estrela, onde todas as partículas compõem a vizinhança umas das outras (Figura 1). Por conseguinte, a melhor posição encontrada na vizinhança de cada uma das  $m$  partículas no instante  $t$ , isto é,  $g_i(t) \forall i \in [1, m]$ , será a mesma e, portanto, igual a  $g(t)$ .

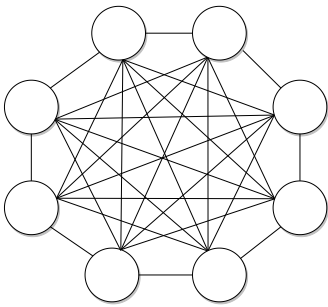


Figura 1. Estrutura de vizinhança em estrela. Fonte: Elaborado pelo autor.

#### B. Local Best

Outra versão popular é o *Local Best Particle Swarm Optimization (lbest PSO)*, cuja vizinhança de cada partícula é dada por  $\mathcal{N}_i = \{\mathbf{p}_{\text{mod}(i-1,m)}, \mathbf{p}_i, \mathbf{p}_{\text{mod}(i+1,m)}\}$  e definida por uma topologia em anel (Figura 2). Nesse caso, a melhor posição encontrada na vizinhança  $\mathcal{N}_i$  pela partícula  $i$  em problemas de minimização é definida pela Equação (4):

$$g_i(t) = \arg \min_{\{\mathbf{p}_k \in \mathcal{N}_i\}} f(\mathbf{p}_k) \quad (4)$$

ou seja, a melhor posição encontrada pela partícula  $i$  no instante  $t$  será a posição da partícula de menor função objetivo que esteja contida na vizinhança  $\mathcal{N}_i$ . Esse tipo de estrutura de vizinhança local baseada em índice não demanda tanto esforço computacional e ainda promove maior diversidade às soluções candidatas durante o processo de otimização [5].

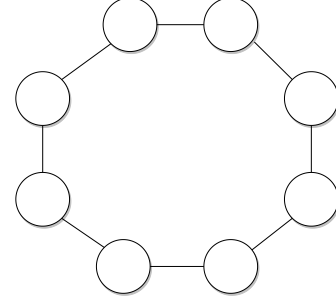


Figura 2. Estrutura de vizinhança em anel. Fonte: Elaborado pelo autor.

### III. PROBLEMAS MONO-OBJETIVOS

Dentre as funções mono-objetivo que compuseram o *benchmark* de problemas do desafio *Competition on Niching Methods for Multimodal Function Optimization* [2], uma função unimodal e uma multimodal foram escolhidas para o planejamento e a análise dos experimentos. Ambas funções são problemas  $\min f(x)$  e cada solução candidata  $\mathbf{x} = [x_1, x_2, \dots, x_D]^T$  de dimensão  $D$  estará restrita a um espaço de busca  $[-100, 100]^D$ .

A função unimodal selecionada foi uma versão da *Sphere Function* com deslocamento, cuja expressão matemática é dada pela Equação (5):

$$f_{\text{sphere}}(\mathbf{x}) = \sum_{i=1}^D z_i^2 + f_{\text{sphere}}^* \quad (5)$$

tal que  $\mathbf{z} = \mathbf{x} - \mathbf{o}$ , onde  $\mathbf{o}$  é um deslocamento distribuído aleatoriamente em  $[-80, 80]^D$ . O valor da função objetivo no ótimo global deslocado é  $f_{\text{sphere}}^* = -1400$  e os valores de  $\mathbf{o}$  foram padronizados e definidos no arquivo `shift_data.txt` pela comissão organizadora da competição [2]. A Figura 3 ilustra a superfície para o caso bidimensional da função descrita.

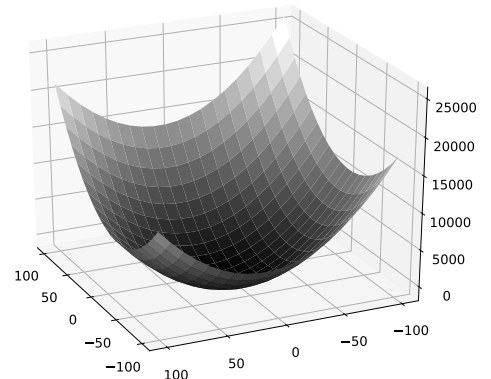


Figura 3. Função *Sphere* da Equação (5) para  $D = 2$ . Fonte: Elaborado pelo autor.

A função multimodal escolhida, por sua vez, foi uma versão rotacionada e deslocada da *Rosenbrock's Function*, que é dada por (6):

$$f_{\text{rosenbrock}}(\mathbf{x}) = \sum_{i=1}^{D-1} (100(z_i^2 + z_{i+1})^2 + (z_i - 1)^2) + f_{\text{rosenbrock}}^*(\mathbf{x}) \quad (6)$$

tal que  $z = \mathbf{M}_1 \left( \frac{2,048(\mathbf{x}-\mathbf{o})}{100} \right) + 1$ , onde  $\mathbf{M}_1$  é a matriz de rotação gerada a partir de entradas padrão normalmente distribuídas por ortonormalização de Gram-Schmidt e cujos valores foram disponibilizados pela comissão organizadora no arquivo `M_D10.txt` [2]. O valor da função objetivo no ótimo global é  $f_{\text{rosenbrock}}^*(\mathbf{x}) = -900$ . A Figura 4 apresenta a superfície bidimensional da função descrita e ressalta sua principal propriedade que é ter um vale estreito do ótimo local ao ótimo global.

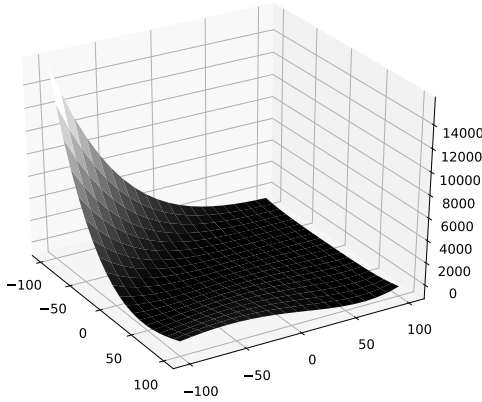


Figura 4. Função Rosenbrock da Equação (6) para  $D = 2$ . Fonte: Elaborado pelo autor.

#### IV. EXPERIMENTOS E RESULTADOS

Para cada uma das duas funções de teste escolhidas, o *gbest PSO* e o *lbest PSO* foram executados em 4 configurações de hiperparâmetros distintas (Tabela I). Nas configurações em que se deseja verificar  $w$  e  $\chi$  no domínio  $[0, 1]$ , os valores foram definidos no meio do intervalo, isto é,  $w_{\text{cfg}_2} = \chi_{\text{cfg}_3} = w_{\text{cfg}_4} = \chi_{\text{cfg}_1} = 0,5$ . Assim, de modo a evocar o Teorema Central do Limite (TLC), foram realizadas 31 execuções de cada uma das configurações do PSO em suas duas versões e a proposta de avaliação se baseou nos valores de função objetivo da melhor solução em cada execução.

TABELA I  
CONFIGURAÇÕES DE HIPERPARÂMETROS.

cfg <sub>1</sub> :	$\chi_1 = 1$	$w = 1$
cfg <sub>2</sub> :	$\chi = 1$	$0 < w < 1$
cfg <sub>3</sub> :	$0 < \chi < 1$	$w = 1$
cfg <sub>4</sub> :	$0 < \chi < 1$	$0 < w < 1$

O processo iterativo do algoritmo foi interrompido após um número máximo de avaliações da função objetivo  $\text{eval}_{\text{max}}$  ser

atingido ou após a solução ótima global ser encontrada, isto é, uma solução cujo valor da função objetivo é  $f(\mathbf{x}) = f_{\text{função}}^*(\mathbf{x})$ . Os demais hiperparâmetros se mantiveram fixos durante todos os experimentos, conforme a Tabela II [3].

TABELA II  
HIPERPARÂMETROS FIXOS.

$D$	$m$	$c_1$	$c_2$	$\delta$	$\text{eval}_{\text{max}}$
10	100	2.05	2.05	0.8	100.000

No que se refere ao desempenho do *Global Best PSO* para a minimização da função *Sphere*, é possível constatar que as configurações  $\text{cfg}_2$  e  $\text{cfg}_3$  atingiram o ótimo global em todas as execuções e, portanto, obtiveram desvio nulo e médias próximas de  $f_{\text{sphere}}^* = -1400$  (Figura 5). A pior configuração foi a  $\text{cfg}_1$ , cujos valores de função objetivo foram praticamente todos positivos. Uma possível justificativa para esse desempenho ruim é que valores de constrição  $\chi$  e inércia  $w$  próximos de 1 priorizam a investigação de áreas novas e ainda desconhecidas no espaço de busca (*exploration*), estratégia similar a de uma busca aleatória. A configuração  $\text{cfg}_2$ , por sua vez, abdica um pouco da diversidade e prioriza o compartilhamento de informações de soluções visitadas anteriormente (*exploitation*), o que justifica os valores razoáveis de média ( $\mu_{\text{cfg}_2} = -1089,31$ ) e de desvio ( $\sigma_{\text{cfg}_2} = 285,32$ ).

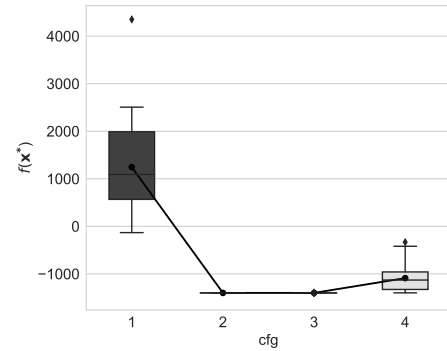


Figura 5. Desempenho do *Global Best Particle Swarm Optimization (gbest PSO)* na minimização da função *Sphere* para quatro configurações de hiperparâmetros  $\chi$  e  $w$  (Tabela I).

A função *Rosenbrock*, em virtude de sua multimodalidade, ocasionou maiores dificuldades ao *Global Best PSO*. Dentre as configurações testadas, nenhuma delas foi capaz de obter média igual a  $f_{\text{rosenbrock}}^* = -900$  e desvio nulo. A configuração que mais se aproximou dessa perspectiva de robustez foi a  $\text{cfg}_2$ , cujos valores de média e desvio foram, respectivamente,  $\mu_{\text{cfg}_2} = -879,04$  e  $\sigma_{\text{cfg}_2} = 27,13$ . Ainda que a área do diagrama de caixa da configuração  $\text{cfg}_3$  seja visualmente maior, tanto a média ( $\mu_{\text{cfg}_3} = -872,24$ ) quanto o desvio ( $\sigma_{\text{cfg}_3} = 29,09$ ) foram bem próximos da configuração  $\text{cfg}_2$ , que apresenta um número significativo de *outliers* (Figura 6). Novamente, tanto a configuração  $\text{cfg}_1$  quanto a  $\text{cfg}_4$ , que não combinam os processos de *exploration* e *exploitation*, apresentaram, em média, as piores soluções.

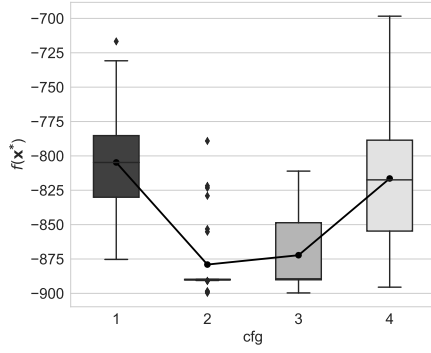


Figura 6. Desempenho do *Global Best Particle Swarm Optimization (gbest PSO)* na minimização da função *Rosenbrock* para quatro configurações de hiperparâmetros  $\chi$  e  $w$  (Tabela I).

A fim de realizar comparações estatísticas de desempenho a posteriori, o *Local Best PSO* foi testado sob as mesmas condições que o *Global Best PSO*. No problema de minimização da função *Sphere*, as configurações  $cfg_2$ ,  $cfg_3$  e  $cfg_4$  foram visualmente idênticas entre si, porém apenas as configurações  $cfg_2$  e  $cfg_3$  obtiveram média  $f_{sphere}^* = -1400$  e desvio nulo (Figura 7). No caso da  $cfg_4$ , a qualidade das soluções encontradas também foi bastante satisfatória. A média foi  $\mu_{cfg_4} = -1399,97$  e o desvio foi  $\sigma_{cfg_4} = 0,0572$ , sugerindo que a topologia em anel cumpre bem o papel de diversidade (*exploration*) perdido ao definir a constrição  $\chi$  e a inércia  $w$  significativamente menores que 1. Já a  $cfg_1$ , cuja configuração prioriza a exploração do espaço em busca de soluções novas, tem tal propriedade ainda mais intensificada pela estrutura de vizinhança esparsa e, portanto, comporta-se como um método bastante oscilatório, justificando sua média positiva ( $\mu_{cfg_1} = 1340,01$ ) e seu desvio padrão acentuado ( $\sigma_{cfg_1} = 540,09$ ).

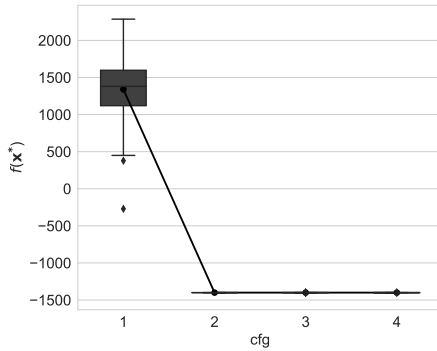


Figura 7. Desempenho do *Local Best Particle Swarm Optimization (lbest PSO)* na minimização da função *Sphere* para quatro configurações de hiperparâmetros  $\chi$  e  $w$  (Tabela I).

Por fim, assim como no *Global Best PSO*, nenhuma das quatro configurações examinadas do *Local Best PSO* foram capazes de alcançar o ótimo global da função *Rosenbrock* em todas as 31 execuções. Novamente, as melhores foram  $cfg_2$  e  $cfg_3$ , cujas médias foram, respectivamente,  $\mu_{cfg_2} = -893,87 \pm 4,27$  e  $\mu_{cfg_3} = -897,29 \pm 4,08$ . A configuração  $cfg_1$  também tornou a ser a pior, bem como a  $cfg_4$  continuou com resultados razoáveis. É interessante ressaltar ainda que,

para todos os quatro conjuntos de experimentos (Figuras 5, 6, 7, 8) as configurações  $cfg_2$  e  $cfg_3$  se destacaram, apresentando médias próximas dos mínimos globais das funções e desvios substancialmente pequenos.

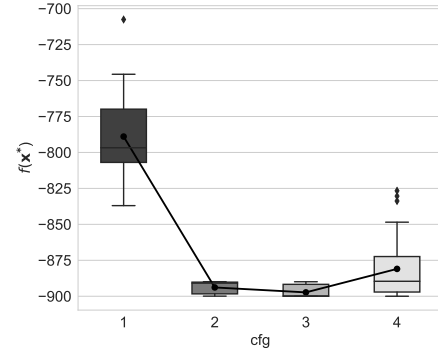


Figura 8. Desempenho do *Local Best Particle Swarm Optimization (lbest PSO)* na minimização da função *Rosenbrock* para quatro configurações de hiperparâmetros  $\chi$  e  $w$  (Tabela I).

De modo a verificar se existem diferenças estatísticas entre as configurações  $cfg_2$  e  $cfg_3$  segundo a qualidade média das melhores soluções em duas topologias de vizinhança distintas, o teste de *Welch* bilateral para 2 amostras independentes foi realizado, uma vez que a premissa de homocedasticidade (igualdade de variâncias) das amostras para alguns pares de comparações não é válida. Tal teste requer a normalidade das amostras, garantida pelo TLC, e assume como hipótese nula ( $H_0$ ) que a média é igual nas duas populações.

TABELA III

ESTATÍSTICAS DO TESTE T DE WELCH.

Comparação	Versão	Função	$p$ -valor
$cfg_2 \times cfg_3$	<i>gbest</i>	<i>Sphere</i>	0,2833
$cfg_2 \times cfg_3$	<i>gbest</i>	<i>Rosenbrock</i>	0,3453
$cfg_2 \times cfg_3$	<i>lbest</i>	<i>Sphere</i>	1,0
$cfg_2 \times cfg_3$	<i>lbest</i>	<i>Rosenbrock</i>	0,0020

A partir dos resultados dos testes estatísticos presentes na Tabela III, é possível afirmar que as configurações  $cfg_2$  e  $cfg_3$  para a versão *Global Best PSO* são estatisticamente equivalentes em ambas funções objetivo, uma vez que não se pode rejeitar a hipótese nula ao nível de confiança de 95% ( $p > 0,05$ ). No que tange a versão *Local Best PSO*, tem-se situações distintas para os problemas considerados. A qualidade média das soluções para a função *Sphere* não só foram estatisticamente equivalentes como foram idênticas, conforme indica o  $p$ -valor unitário. Em contrapartida, foram detectadas diferenças estatísticas entre os desempenhos médios das configurações  $cfg_2$  e  $cfg_3$  na minimização da função *Rosenbrock*, visto que a hipótese nula é rejeitada ao nível de confiança de 95% ( $p < 0,05$ ). Assim, é possível concluir a partir das médias que a configuração  $cfg_2$  supera a  $cfg_3$  na minimização da função multimodal com a versão *lbest*.

## V. CONCLUSÕES

Mediante a realização do presente trabalho, é possível evidenciar a versatilidade do *Particle Swarm Optimization* (PSO) em problemas mono-objetivos de alta dimensão. Dada a vasta literatura acerca das sugestões de melhorias no comportamento do algoritmo para determinadas aplicações, foram consideradas duas versões do PSO com topologias de vizinhança distintas, uma de natureza global e outra de natureza local.

Contudo, nota-se que a robustez desses algoritmos está altamente condicionada a escolha dos hiperparâmetros. Pequenas variações nesses valores prejudicam bastante a qualidade das soluções, uma vez que afetam diretamente na exploração do espaço de busca. Sob essa perspectiva e de acordo com os resultados apresentados, fica ainda mais evidente a necessidade de se fazer ajustes eficientes de hiperparâmetros para se alcançar bons desempenhos.

## REFERÊNCIAS

- [1] J. Kennedy and R. Eberhart, "Particle Swarm Optimization," IEEE, pp. 1942–1948, 1995.
- [2] J. Liang, B. Qu, P. Suganthan, and A. G. Hernández-Díaz, "Problem Definitions and Evaluation Criteria for the CEC 2013 Special Session on Real-Parameter Optimization," *Computational Intelligence Laboratory, Zhengzhou University, Zhengzhou, China and Nanyang Technological University, Singapore, Technical Report*, vol. 201212, no. 34, pp. 281–295, 2013.
- [3] F. G. Guimarães, "Lecture Notes on Evolutionary Computation," 2019, Acessado em 7 de Setembro de 2020.
- [4] R. D. Russ Eberhart, Pat Simpson, *Computational Intelligence PC Tools*. San Diego, CA, USA: Academic Press Professional, 1996.
- [5] A. P. Engelbrecht, "Particle Swarm Optimization: Global Best or Local Best?" in *2013 BRICS Congress on Computational Intelligence and 11th Brazilian Congress on Computational Intelligence*. IEEE, 2013, pp. 124–135.