



Universidade Federal  
de São João del-Rei

UNIVERSIDADE FEDERAL DE SÃO JOÃO DEL-REI  
DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO  
ALGORITMOS BIOINSPIRADOS

# Particle Swarm Optimization

Gustavo Henriques da Cunha

São João del-Rei  
2024

## Lista de Figuras

1	Gráfico da Função de Rastrigin . . . . .	3
2	Tabela com comparação entre os testes com diferentes parâmetros . . . . .	4
3	Gráfico com comparação entre as execuções com topologia do Melhor Global	5
4	Gráfico com comparação entre as execuções com topologia do Vizinho Adjacente . . . . .	6
5	Gráfico com detalhes da população em execução com topologia do Melhor Global . . . . .	7
6	Gráfico com detalhes da população em execução com topologia do Vizinho Adjacente . . . . .	7

# Sumário

<b>1</b>	<b>INTRODUÇÃO</b>	<b>1</b>
1.1	Objetivo . . . . .	1
<b>2</b>	<b>O ALGORITMO</b>	<b>1</b>
<b>3</b>	<b>ABORDAGEM DO PROBLEMA</b>	<b>2</b>
<b>4</b>	<b>ANÁLISE DE DESEMPENHO</b>	<b>3</b>
<b>5</b>	<b>CONCLUSÃO</b>	<b>8</b>
	<b>REFERÊNCIAS</b>	<b>8</b>

# 1 INTRODUÇÃO

Este é um trabalho prático da disciplina de Algoritmos Bioinspirados no curso de Ciência da Computação na UFSJ.

## 1.1 Objetivo

Este trabalho têm como objetivo aprender a construir e testar o algoritmo *Particle Swarm Optimization* (PSO) aplicado a uma função *benchmark*.

Além disso, é posto em foco a análise e calibragem dos parâmetros, buscando melhorar a eficiência do algoritmo.

## 2 O ALGORITMO

Concebido por James Kennedy e Russell Eberhart em 1995, o algoritmo PSO se originou simplesmente como uma tentativa de simulação simplificada de uma revoada de pássaros (Kennedy e Eberhart, 1995), onde mais tarde foi observado o potencial do modelo para otimizar problemas, inicialmente de variáveis contínuas.

Para simular o aspecto cooperativo que pode ser encontrado nos pássaros, o PSO guarda a experiência individual de cada indivíduo, mais uma experiência comunitária, que guia outros indivíduos para um estado melhor, como se fossem pássaros se comunicando entre si.

O PSO começa ao criar uma população de  $m$  indivíduos de forma aleatória. Esses indivíduos podem ser chamados de partículas, e são movidos pelo espaço de busca, usando as experiências pessoal e comunitária, tentando convergir para uma posição mais vantajosa. A forma como a partícula se movimentará pelo espaço será influenciada por sua velocidade, que será calculada pela fórmula abaixo:

$$v_{ij} = wv_{ij}^k + c_1r_1(p_{best_{ij}}^k - x_{ij}^k) + c_2r_2(g_{best_j}^k - x_{ij}^k)$$

para  $i = [1, \dots, m]$  e  $j = [1, \dots, n]$ , onde  $m$  é o número de partículas na população e  $n$  o número da dimensão de cada solução.

Note que o fator  $p_{best_{ij}}^k - x_{ij}^k$  da equação representa a experiência pessoal ou cognitiva e o fator  $(g_{best_j}^k - x_{ij}^k)$  representa a experiência comunitária. O fator  $w$ , chamado de fator

de inércia, atua com o intuito de diversificar a solução.

Em particular, tem-se o fator de experiência comunitária pode ser obtido por diferentes maneiras, dependendo da topologia usada na versão do algoritmo. No caso deste trabalho, é usado duas dessas versões, que são comparadas na análise de parâmetros. Uma delas consegue o fator comunitário em  $g_{best_j}^k$  com um melhor global, conhecido por todas as partículas, enquanto a outra abordagem consegue diferentes valores para cada partícula, onde ele é achado através dos vizinhos adjacentes da partícula.

Com a velocidade calculada, tem-se a seguinte equação para atualização da partícula:

$$x_i^{k+1} = x_i^k + v_i^{k+1}$$

Assim, em cada iteração é feita uma atualização em cada partícula, que caminham em direção a melhor solução até convergirem em algum ponto.

### 3 ABORDAGEM DO PROBLEMA

Neste trabalho, por ser o PSO bom em problemas de natureza contínua, foi escolhido uma função de *benchmark* para fazer o teste do algoritmo. Essas funções são muito usadas para testar os mais diversos algoritmos de otimização.

Assim, para testar os parâmetros que melhor funcionam com o algoritmo foi escolhida a Função de Rastrigin:

$$f(x) = 10d + \sum_{i=1}^d [x_i^2 - 10 \cos(2\pi x_i)]$$

Onde  $d$  é o numero de dimensões usado na função. O domínio das entradas fica no intervalo  $x_i \in [-5.12, 5.12]$ , para todo  $i = 1, \dots, d$ .

A função de Rastrigin é boa pois permite escolher o número de dimensões, o que faz possível dificultar o algoritmo a achar o ótimo global caso funcione bem com poucas dimensões, além de apresentar vários mínimos locais, como podemos ver na figura 1.

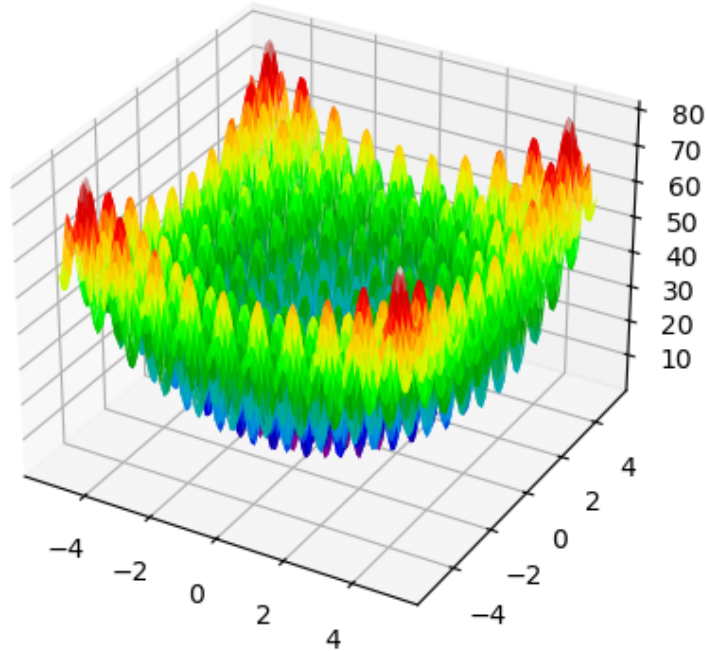


Figura 1: Gráfico da Função de Rastrigin

O mínimo global é conhecido, onde  $f(x^*) = 0$ , em  $x^* = (0, \dots, 0)$ .

## 4 ANÁLISE DE DESEMPENHO

O algoritmo *Particle Swarm Optimization* é probabilístico e depende da qualidade de seus parâmetros para gerar soluções melhores. Assim, para verificar qual o melhor conjunto de parâmetros que geram os melhores resultados, são feitos diferentes testes com diferentes conjuntos de parâmetros e gerado uma análise estatística em cima dos resultados obtidos.

Assim, pode ser variado os parâmetros como número de iterações, número de partículas, as constantes  $c_1$ ,  $c_2$  e  $w$  e o tipo de topologia. Para cada conjunto de parâmetros, é executado 10 vezes o algoritmo, sendo coletado o melhor indivíduo de cada execução, para ser feito o cálculo da média e o desvio padrão dos melhores resultados por conjunto de parâmetros, com o objetivo de achar quem têm a melhor média e que está consistente-

mente produzindo os melhores resultados.

Com isso, foi gerado a tabela da figura 2, que mostra, ordenadamente, os resultados desses testes com a função de Rastrigin com 5 dimensões:

Iterações	N. Partículas	N. Dimensões	$C_1$	$C_2$	W	Topologia	Média	Desvio
1000	100	5	1.2	4.1	0.02	M. Global	0.00	0.00
1000	150	5	1.5	4.1	0.02	M. Global	0.00	0.00
1000	200	5	1.5	4.1	0.01	M. Global	0.00	0.00
1000	200	5	1.5	4.1	0.02	M. Global	0.00	0.00
500	300	5	1.5	4.1	0.03	M. Global	0.00	0.00
1000	200	5	1.5	4.1	0.03	M. Global	0.00	0.00
200	300	5	1.5	4.1	0.03	M. Global	0.00	0.00
200	300	5	1.5	3.1	0.03	M. Global	0.20	0.40
100	500	5	0.8	3.1	0.03	M. Global	0.30	0.46
200	500	5	1.5	3.1	0.03	M. Global	0.40	0.49
100	300	5	0.8	3.1	0.03	M. Global	0.73	1.28
200	500	5	0.8	3.1	0.03	M. Global	0.80	0.60
3000	500	5	2.1	1.2	0.01	V. Adjacente	0.83	0.51
200	300	5	0.8	3.1	0.03	M. Global	1.45	2.16
300	100	5	0.8	3.1	0.03	M. Global	1.49	0.80
3000	500	5	2.1	1.2	0.01	V. Adjacente	1.53	0.77
3000	300	5	2.1	1.2	0.02	V. Adjacente	1.66	0.77
3000	300	5	3.9	1.0	0.05	V. Adjacente	1.69	0.77
3000	300	5	2.1	1.2	0.01	V. Adjacente	1.94	1.26
3000	300	5	3.5	1.0	0.0	V. Adjacente	2.07	1.17
3000	300	5	3.5	1.0	0.01	V. Adjacente	2.46	0.79
50000	50	5	0.8	4.1	0.03	M. Global	2.49	7.46
1000	200	5	1.5	4.1	0.05	M. Global	2.49	7.46
3000	300	5	3.5	1.0	0.1	V. Adjacente	2.57	0.88
3000	300	5	3.5	1.0	0.08	V. Adjacente	2.82	1.64
3000	300	5	4.6	0.5	0.01	V. Adjacente	2.84	1.01
1000	100	5	1.5	4.1	0.02	M. Global	2.89	8.68
3000	300	5	3.5	1.0	0.001	V. Adjacente	2.93	1.60
3000	300	5	1.3	1.2	0.01	V. Adjacente	2.93	1.76
3000	300	5	4.1	1.2	0.02	V. Adjacente	3.01	1.73

Figura 2: Tabela com comparação entre os testes com diferentes parâmetros

É possível notar que o algoritmo conseguiu constantemente a solução ótima com a topologia do melhor global, com um  $C_1$  menor que o  $C_2$  e com uma inércia baixa. Mesmo com um número baixo de partículas o algoritmo consegue com facilidade achar uma solução boa.

Entretanto, com a topologia do vizinho adjacente não foi possível repetir os resultados obtidos com o melhor global.

Note que a forma que os parâmetros são arranjados depende da topologia. Com o uso do melhor global os resultados são melhores com um fator comunitário maior, onde o melhor atua como um líder efetivamente levando todas as partículas para uma posição melhor. Já com os vizinhos adjacentes, o fator cognitivo maior produz melhores resultados, onde pode ser interpretado como cada partícula levando em consideração as experiências dos seus vizinhos, mas priorizando sua própria experiência pessoal enquanto viaja pelo espaço de busca.

Com os resultados da tabela, foi produzido gráficos que mostram a convergência das soluções usando o melhor conjunto de parâmetros encontrados para cada topologia.

Os gráficos das imagens 3 e 4 mostram, respectivamente, a topologia do melhor global e a topologia do vizinho adjacente, depois de dez execuções usando esses parâmetros, onde foram traçados retas que mostram o progresso da *qualidade* da melhor solução por geração.

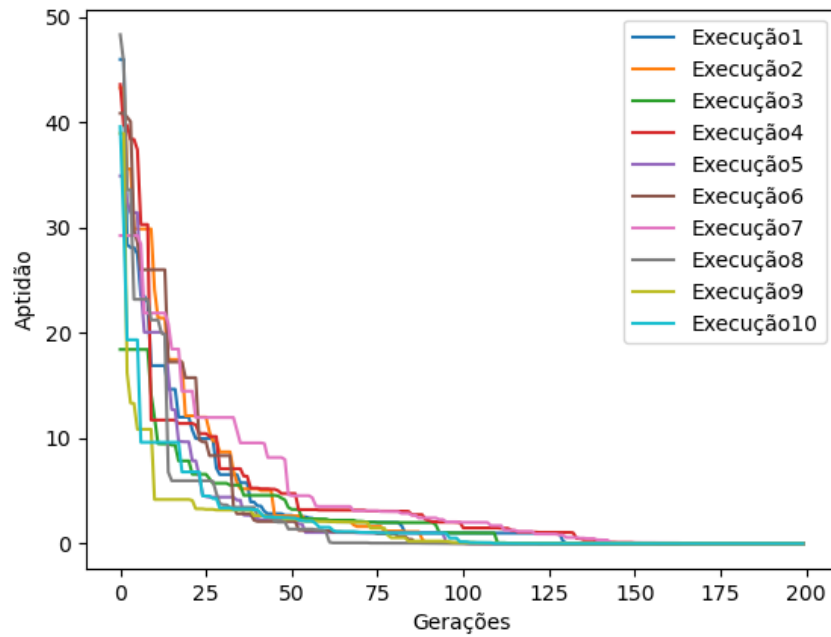


Figura 3: Gráfico com comparação entre as execuções com topologia do Melhor Global



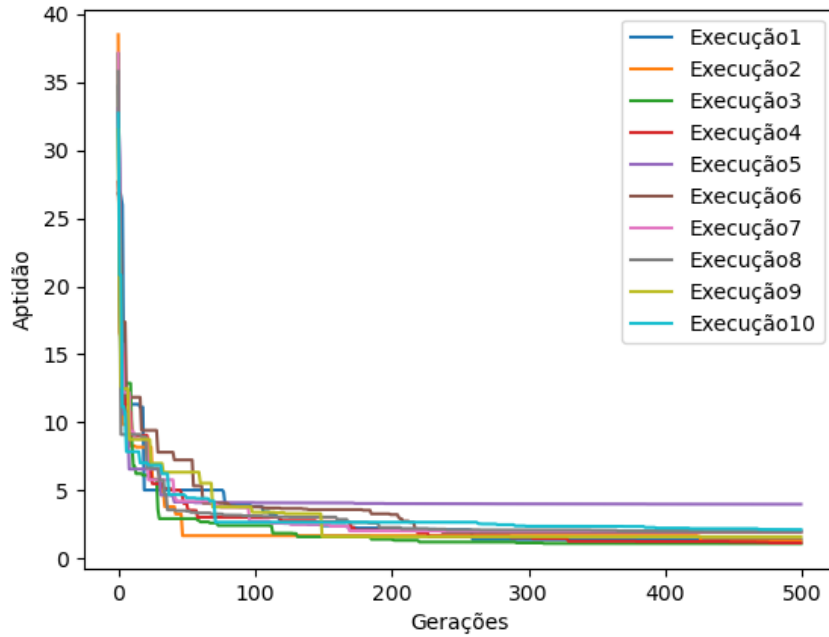


Figura 4: Gráfico com comparação entre as execuções com topologia do Vizinho Adjacente

Consegue-se notar que em ambos os gráficos as execuções tiveram uma convergência rápida, encontrando o ótimo local ou global logo nas primeiras gerações. Especificamente no caso da topologia do melhor global, a solução ótima foi encontrada rapidamente em todas as execuções. No caso da topologia do vizinho adjacente, é possível observar que apesar do ótimo global não ser encontrado, em boa parte das execuções um ótimo local próximo ao ótimo global é encontrado, e em algumas execuções ainda é possível ver uma melhora mínima na aptidão.

Nas imagens 5 e 6 abaixo, foi feita, para cada topologia e com os parâmetros usados anteriormente, a análise, em uma execução, do melhor e pior de cada geração, além da média e mediana da população durante a execução.

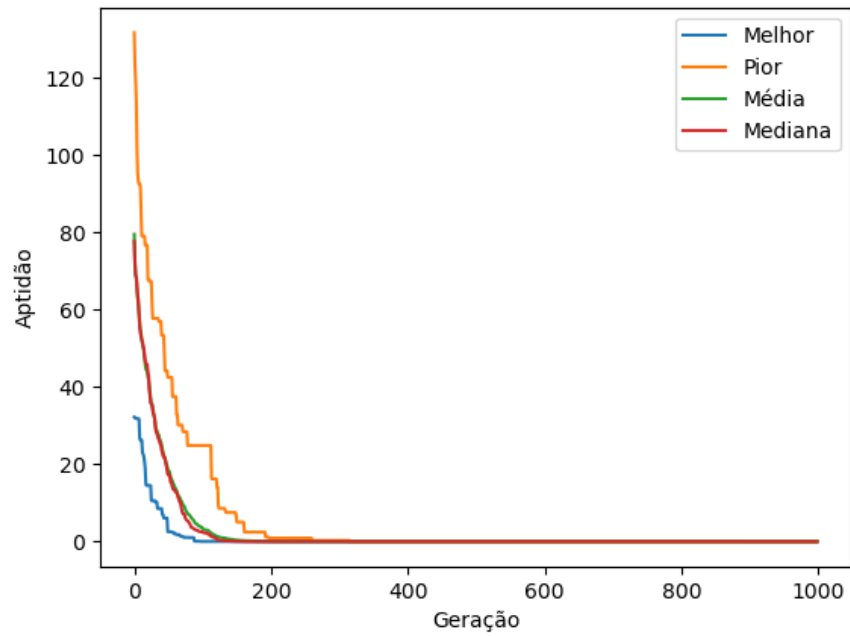


Figura 5: Gráfico com detalhes da população em execução com topologia do Melhor Global

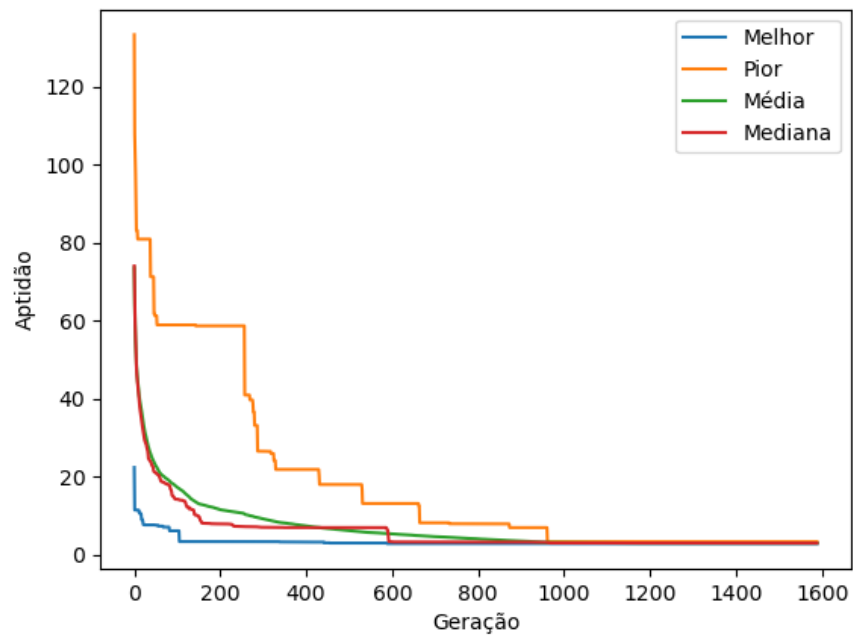


Figura 6: Gráfico com detalhes da população em execução com topologia do Vizinheiro Adjacente

Em ambos os casos é possível observar que toda a população tende a convergir para o melhor global encontrado depois de algumas iterações. No caso da segunda topologia isso é mais demorado, o que permite uma procura em um espaço maior por mais tempo, o que pode fazer com que o algoritmo fuja de um ótimo local, apesar de que isso não acontece nessa execução. Na primeira topologia, como é utilizado o melhor global como fator de experiência comunitária, assim que o ótimo global é achado, as outras partículas seguem ao mesmo ponto rapidamente.

## 5 CONCLUSÃO

Em todos os testes realizados, é possível ver que com os parâmetros certos o algoritmo consegue achar várias soluções que se aproximam ou chegam no ótimo global da função. Destaca-se o desempenho melhor da topologia com melhor global nos testes feitos com dimensão cinco, onde conseguiu o melhor resultado possível com facilidade e velocidade.

## Referências

[KENNEDY, 1995] KENNEDY, J.; EBERHART, R. (1995). *Particle Swarm Optimization*. Proceedings of the IEEE International Conference on Neural Networks, v. 4, Perth, Australia, 1942-1948.