



Universidade Federal de São João del Rei  
Departamento de Ciência da Computação  
Curso de Ciência da Computação

## Algoritmo Genético 6 - PSO

Adélson de Oliveira Carmo Júnior  
212050019

### 1 Como executar o programa:

O programa possibilita a passagem de valores para o número de gerações e o tamanho da população como parâmetros, o que facilita a realização de novos testes para uma compreensão mais aprofundada do seu funcionamento. Para utilizar o programa, basta fornecer a sua versão do Python, o número de partículas, o número de gerações desejado e se deseja mostrar(0 para não e 1 para sim) o gráfico ou não seguindo a ordem exemplificada abaixo:

```
1 python3.11 ag1.py 100 100 1
```

Além disso, para o funcionamento adequado do programa, é essencial que o arquivo de entrada esteja localizado na mesma pasta do programa, devidamente nomeado e preenchido conforme demonstrado abaixo

- **instancia.txt:** Contém a matriz de distância das cidades;

Observação: Esse algoritmo está usado para resolver o caixeiro viajante nessa aplicação.

### 2 Resultados esperados

Após a execução do algoritmo, é gerado de saída: saída.txt. Este, salva os dados dos melhores indivíduos por iteração utilizando o método escolhido, respectivamente.

A figura 1 ilustra esse arquivo de saída. Se necessário, um gráfico adicional será gerado, conforme mostrado na figura 2, exibindo a curva de convergência para encontrar o melhor caminho.

instancia.txt	82	521.0
pso.py	83	521.0
saida.txt	84	521.0
	85	521.0
	86	521.0
	87	521.0
	88	521.0
	89	521.0
	90	521.0
	91	521.0
	92	521.0
	93	521.0
	94	521.0
	95	521.0
	96	521.0
	97	521.0
	98	521.0
	99	521.0
	100	521.0
	101	

Figura 1: Melhores resultados do método

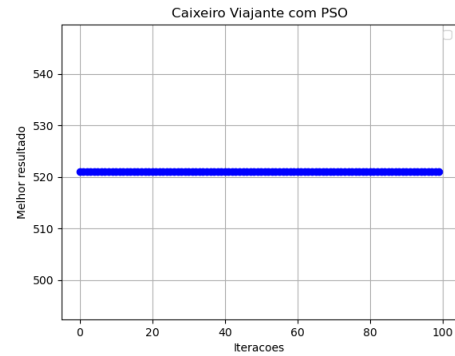


Figura 2: Gráfico do melhor indivíduo de cada geração

### 3 Testes e análise de resultados

Nesta documentação, são apresentados os resultados dos testes realizados com diferentes tamanhos de população de partículas em um algoritmo PSO (Particle Swarm Optimization). Foram conduzidos 100 testes, variando o tamanho da população entre 50, 100 e 250 partículas, enquanto o número de gerações foi mantido constante em 100 para todas as configurações, visando assegurar a precisão dos resultados.

A Figura 3 demonstra que os menores valores para o melhor indivíduo foram obtidos nas populações maiores. Por outro lado, a Figura 4 revela que todos os testes convergiram para soluções similares, independentemente do tamanho da população de partículas.

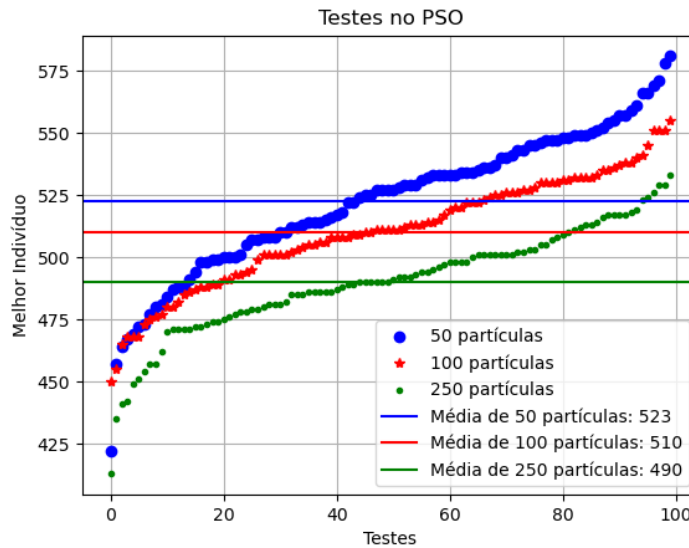


Figura 3: Melhores valores em cada teste

Assim, conclui-se que uma população maior de partículas tende a produzir resultados mais satisfatórios para o problema do caixeiro viajante. No entanto, é importante destacar que, ao comparar com algoritmos genéticos, o PSO ainda apresenta uma eficiência inferior.

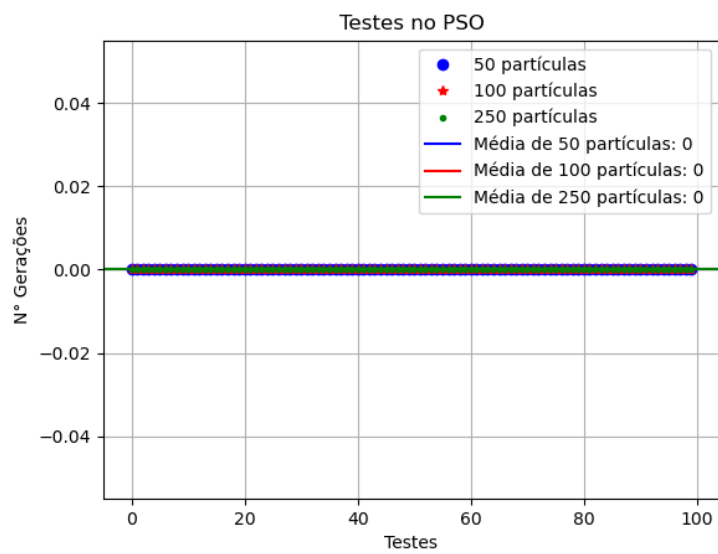


Figura 4: Gerações para encontra o melhor indivíduo