

Particle Swarm Optimization (PSO)

Carolina Ribeiro Xavier

Junho de 2025

Inspiração Biológica

- ▶ O **Particle Swarm Optimization (PSO)** foi desenvolvido por James Kennedy e Russel Eberhart em 1995.
- ▶ Originalmente, projetado para resolver problemas de otimização com **variáveis contínuas**.
- ▶ Inspirado pelo **comportamento social e cooperativo** de diversas espécies em busca de recursos.
- ▶ O algoritmo guia-se pela **experiência pessoal** (P_{best}) e pela **experiência geral** (G_{best}).

Como o PSO Funciona?

- ▶ Cria uma população de **soluções candidatas** (partículas).
- ▶ Move as partículas no espaço de busca usando **fórmulas matemáticas** para posição e velocidade.
- ▶ O movimento de cada partícula é influenciado por:
 - ▶ Sua **melhor posição conhecida** (P_{best}).
 - ▶ A **melhor posição conhecida pelo grupo** (G_{best}).

Analogia: Bando de Pássaros

- ▶ O PSO pode simular o comportamento de um bando de pássaros procurando alimento.
- ▶ O bando encontra seu alvo por **esforço conjunto**, compartilhando informações.



Figura: Bando de pássaros

PSO: Enxame e Partículas

- ▶ É um **algoritmo populacional**.
- ▶ A população é chamada de "**nuvem**" ou "**enxame**".
- ▶ Os indivíduos são chamados de "**partículas**".
- ▶ O enxame evolui por **cooperação e competição**.
- ▶ Partículas se beneficiam da **própria experiência** e da **experiência dos outros** membros.

Atualização de Posição e Velocidade

Equação de Velocidade

A atualização da velocidade é dada pela Equação 1:

$$v_{ij}^{k+1} = \textcolor{red}{w}v_{ij}^k + c_1r_{1j}(p_{best_{ij}}^k - x_{ij}^k) + c_2r_{2j}(g_{best_j}^k - x_{ij}^k) \quad (1)$$

para $i = [1, \dots, m]$ e $j = [1, \dots, n]$.

- ▶ **Parte em vermelho:** responsável pela **diversificação** da solução.
- ▶ **Parte em azul:** responsável pela **intensificação**.

Atualização de Posição e Velocidade

Equação de Posição

A posição de cada partícula depende da velocidade e é dada pela Equação 2:

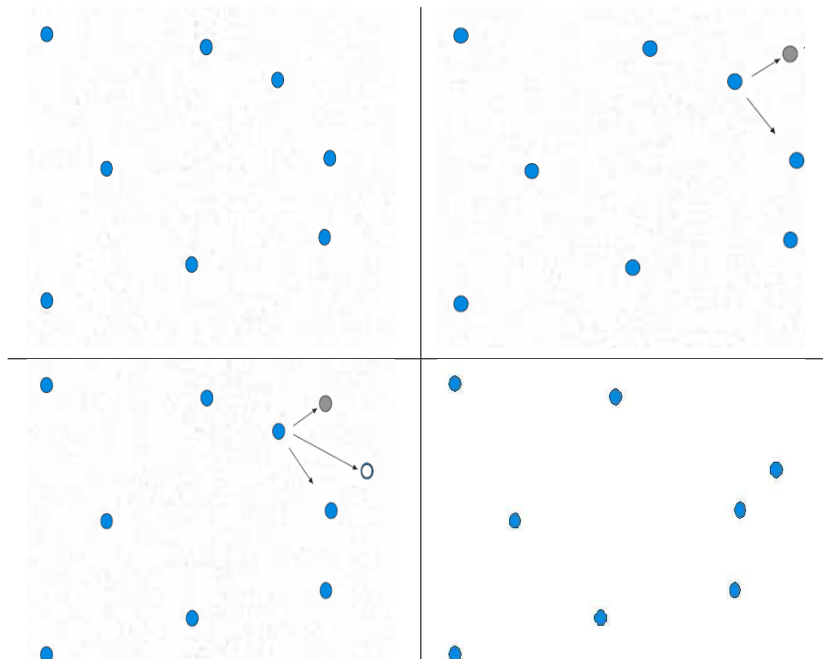
$$x_i^{k+1} = x_i^k + v_i^{k+1} \quad (2)$$

A nova velocidade da partícula é influenciada por três fatores:

- ▶ **Velocidade anterior.**
- ▶ **Distância entre sua posição atual e melhor posição alcançada (cognitivo).**
- ▶ **Distância entre sua posição atual e a melhor posição do grupo (social).**

A partir do cálculo dessa nova velocidade, a partícula “voa” para sua nova posição.

Exemplo de Mudança de Posição



Interpretação Geométrica

A interpretação geométrica dessa mudança de posição é um conjunto de operações simples sobre vetores.

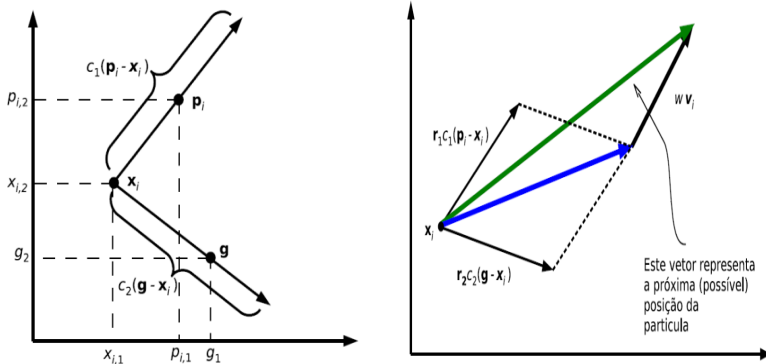


Figura: Interpretação geométrica da mudança de posição de uma partícula

Pseudocódigo

Dada a dimensão da função que se deseja otimizar, teremos os vetores para representar as soluções do problema:

inicialize a nuvem de partículas

repita

para $i = 1$ **até** m

se $f(\mathbf{x}_i) < f(\mathbf{p}_i)$ **então**

$\mathbf{p}_i = \mathbf{x}_i$

se $f(\mathbf{x}_i) < f(\mathbf{g})$ **então**

$\mathbf{g} = \mathbf{x}_i$

fim se

fim se

para $j = 1$ **até** n

$r_1 = \text{rand}()$, $r_2 = \text{rand}()$

$\mathbf{v}_{ij} = w\mathbf{v}_{ij} + c_1 r_1 (\mathbf{p}_i - \mathbf{x}_{ij}) + c_2 r_2 (\mathbf{g}_j - \mathbf{x}_{ij})$

fim para

$\mathbf{x}_i = \mathbf{x}_i + \mathbf{v}_i$

Funções Benchmark

- ▶ Usaremos o PSO para otimizar **funções benchmark** de variáveis reais e contínuas.

Questões de Projeto do PSO

As principais questões para o PSO são:

- ▶ **Configuração inicial**
- ▶ **Função objetivo**
- ▶ **Representação**
- ▶ **Parâmetros**
- ▶ **Atualização de posição**
- ▶ **Topologia**

Configuração Inicial

- ▶ Cada partícula deve ter uma **localização inicial**.
- ▶ Esta pode ser **aleatória** ou **distribuída** de forma dirigida pelo espaço de busca.

Função Objetivo

- ▶ Mapeia a **qualidade do indivíduo** diante da solução.
- ▶ Nosso objetivo é **minimizar** uma função de parâmetros reais.

Representação da Partícula

Cada partícula i terá três vetores de n dimensões:

$$X_i = \begin{bmatrix} x_0 \\ x_1 \\ \dots \\ x_n \end{bmatrix} \quad V_i = \begin{bmatrix} v_0 \\ v_1 \\ \dots \\ v_n \end{bmatrix} \quad P_{best_i} = \begin{bmatrix} p_0 \\ p_1 \\ \dots \\ p_n \end{bmatrix}$$

- ▶ X : vetor de coordenadas da **posição atual**.
- ▶ V : vetor da **velocidade**.
- ▶ P_{best_i} : guarda as coordenadas da **melhor posição conhecida pela partícula**.
- ▶ Cada partícula possui: valor de **aptidão** $f(x_i)$ e o **índice da melhor partícula** de sua vizinhança.

Representação Global

- ▶ Globalmente tem-se o vetor $G_{best_i} = \begin{bmatrix} g_0 \\ g_1 \\ \dots \\ g_n \end{bmatrix}$.
- ▶ Guarda as coordenadas da **melhor solução visitada pelo enxame** ou pela vizinhança da partícula (depende da topologia).
- ▶ Podemos ter somente o índice da partícula que acessou o melhor valor e acessar seu atributo P_{best} .

Parâmetros do PSO

- ▶ m : **tamanho do enxame** (número de partículas).
- ▶ c_1 : parâmetro de **aprendizado cognitivo** (taxa de aprendizado pessoal).
- ▶ c_2 : parâmetro de **aprendizado social** (taxa de aprendizado do grupo).
- ▶ w : **ponderação de inércia** (fator de diversificação).

Todos são parâmetros definidos pelo usuário.

Topologia

- ▶ Diferentes topologias podem gerar diferentes soluções.
- ▶ Quanto mais vizinhos em comum, mais rápida a convergência.

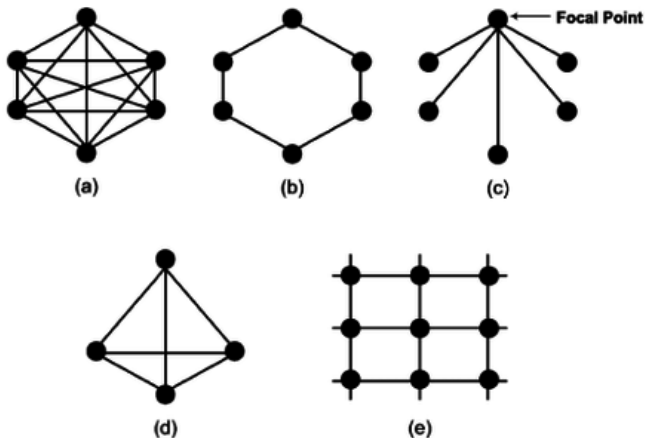


Figura: Exemplos de Topologias de Swarm

Critério de Parada

Quando parar o algoritmo?

- ▶ **Número máximo de iterações.**
- ▶ **Estagnação** (atingiu um mínimo local).
- ▶ **Convergência** (atingiu um mínimo global ou solução satisfatória).

Detalhes para Implementação

- ▶ **Limites de Posição:** De acordo com a função objetivo, estabelecer limites para as posições de uma partícula:
 $x_{ij} \in [x_{min}, x_{max}]$.
- ▶ **Recuo de Limites:** Se x_{ij} sair do intervalo:
 - ▶ Se $x_{ij} < x_{min}$, então $x_{ij} = x_{min}$.
 - ▶ Se $x_{ij} > x_{max}$, então $x_{ij} = x_{max}$.
- ▶ **Limites de Velocidade:** Pode-se estabelecer
 $v_{min} \leq v \leq v_{max}$.

Desafio de Implementação

- ▶ Escolha uma função do artigo para otimizar.
- ▶ Prefira funções com mais de 2 dimensões para um desafio maior.
- ▶ Defina a **estrutura de dados** para armazenar partículas, soluções, vizinhanças e valores de fitness.
- ▶ Explore bibliotecas para operações com vetores.

Experimento Fatorial

Defina 3 dos valores listados a seguir com um experimento fatorial para analisar seu impacto:

- ▶ w (**fator de diversificação**)
- ▶ c_1 (**intensificação**, fator cognitivo)
- ▶ c_2 (**intensificação**, fator social)
- ▶ fixe m (**número de partículas**)
- ▶ **Topologias**
- ▶ fixe k (**número máximo de iterações**)