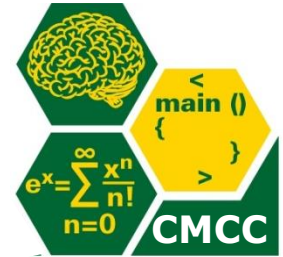




Universidade Federal do ABC



ENXAME DE PARTÍCULAS

Prof. Fabrício Olivetti de França



Universidade Federal do ABC

HISTÓRICO

Comportamento Coordenado

Em 1987, Reynolds se interessou pelo comportamento coordenado de alguns animais como:

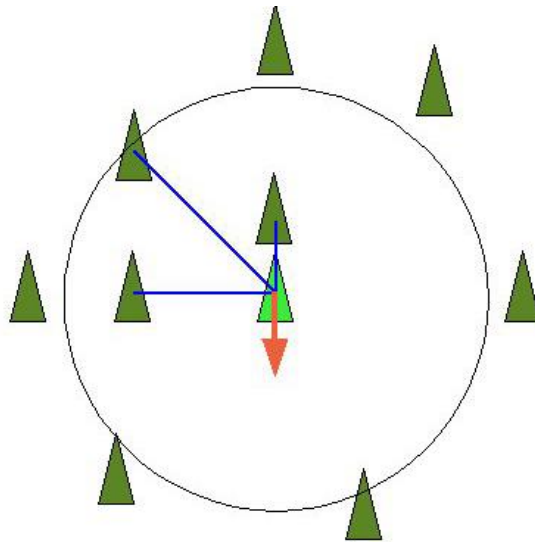
- voo de bando de pássaros
- nado sincronizado de cardume de peixes
- dentre outros.



Comportamento Coordenado

Ele propôs o seguinte modelo comportamental baseado em 3 regras:

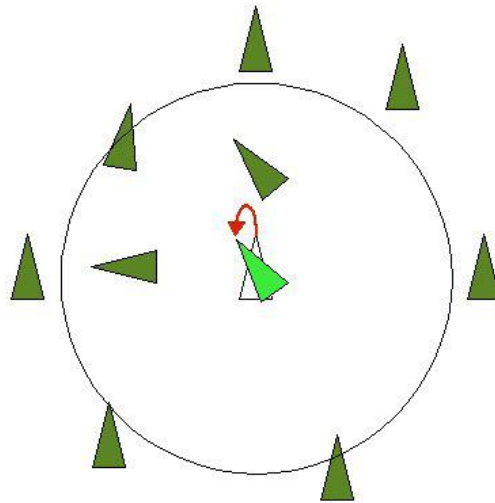
Separação: cada agente tenta se distanciar de seu vizinho se estiver muito próximo.



Comportamento Coordenado

Ele propôs o seguinte modelo comportamental baseado em 3 regras:

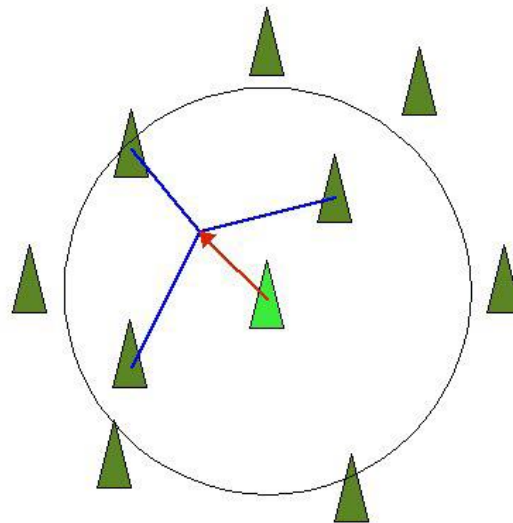
Alinhamento: cada agente segue a direção média de seus vizinhos.



Comportamento Coordenado

Ele propôs o seguinte modelo comportamental baseado em 3 regras:

Coesão: cada agente tenta seguir na posição média de seus vizinhos.



Comportamento Coordenado

Kennedy e Eberhart, em 1995, incluíram um alvo (ex.: ninho) no modelo de Reynolds de tal forma que:

- ❑ Cada agente fosse atraído para esse alvo
- ❑ O agente tivesse memória do local onde ele ficou mais próximo ao alvo
- ❑ Cada agente compartilhasse essa informação com seus vizinhos



Comportamento Coordenado

Com esse simples modelo, eventualmente todos os agentes atingiram o alvo.

E se a métrica de distância até o alvo fosse substituída por uma função de minimização (com alvo desconhecido)?

Os agentes chegariam até o mínimo?





Universidade Federal do ABC

ESTRUTURA BÁSICA

Pseudo-Algoritmo

O algoritmo Particle Swarm Optimization (PSO) pode ser descrito da seguinte maneira:

inicializa()

Para it de 1 até t:

 Para cada p em P:

 atualiza posição

 atualiza velocidade

 atualiza melhor partícula



Pseudo-Algoritmo

Nesse algoritmo trabalhamos com:

$P \rightarrow$ matriz com a posição de cada partícula

$P_{best} \rightarrow$ matriz com a melhor posição já encontrada por cada partícula

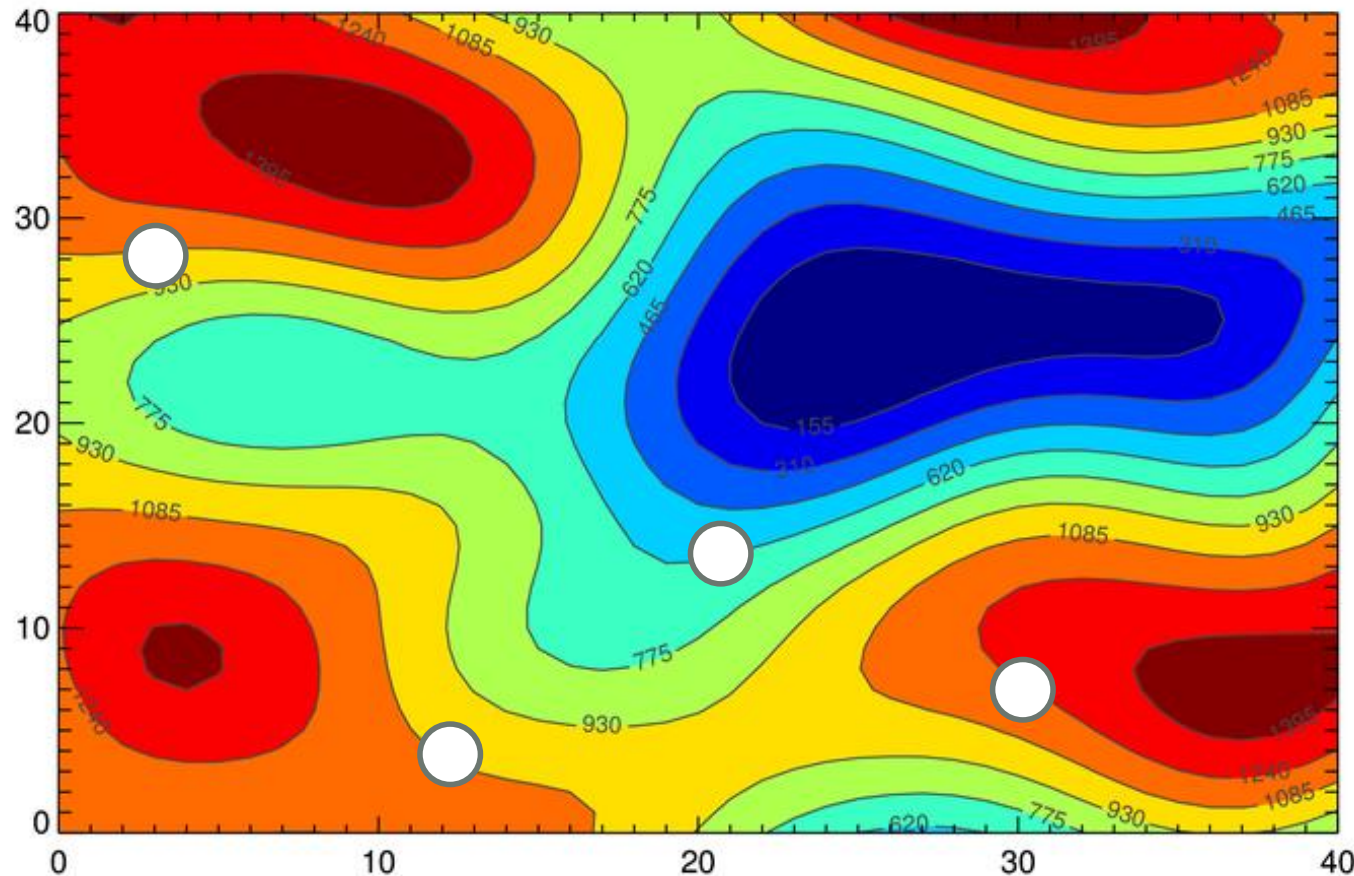
$V \rightarrow$ velocidade de cada partícula

$p_{gbest} \rightarrow$ melhor posição encontrada até então



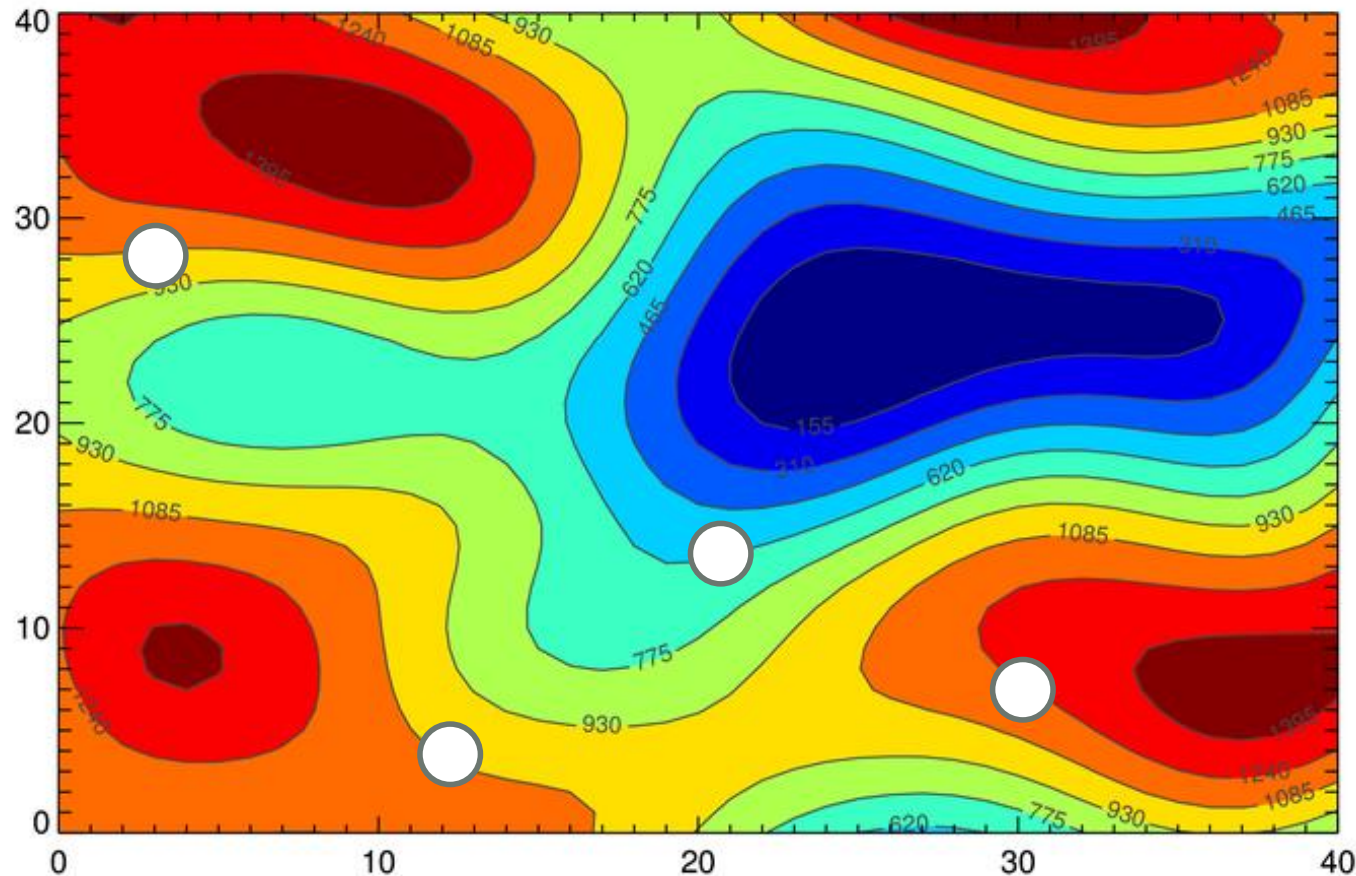
Algoritmo Passo a Passo

Cria uma população de partículas aleatoriamente.



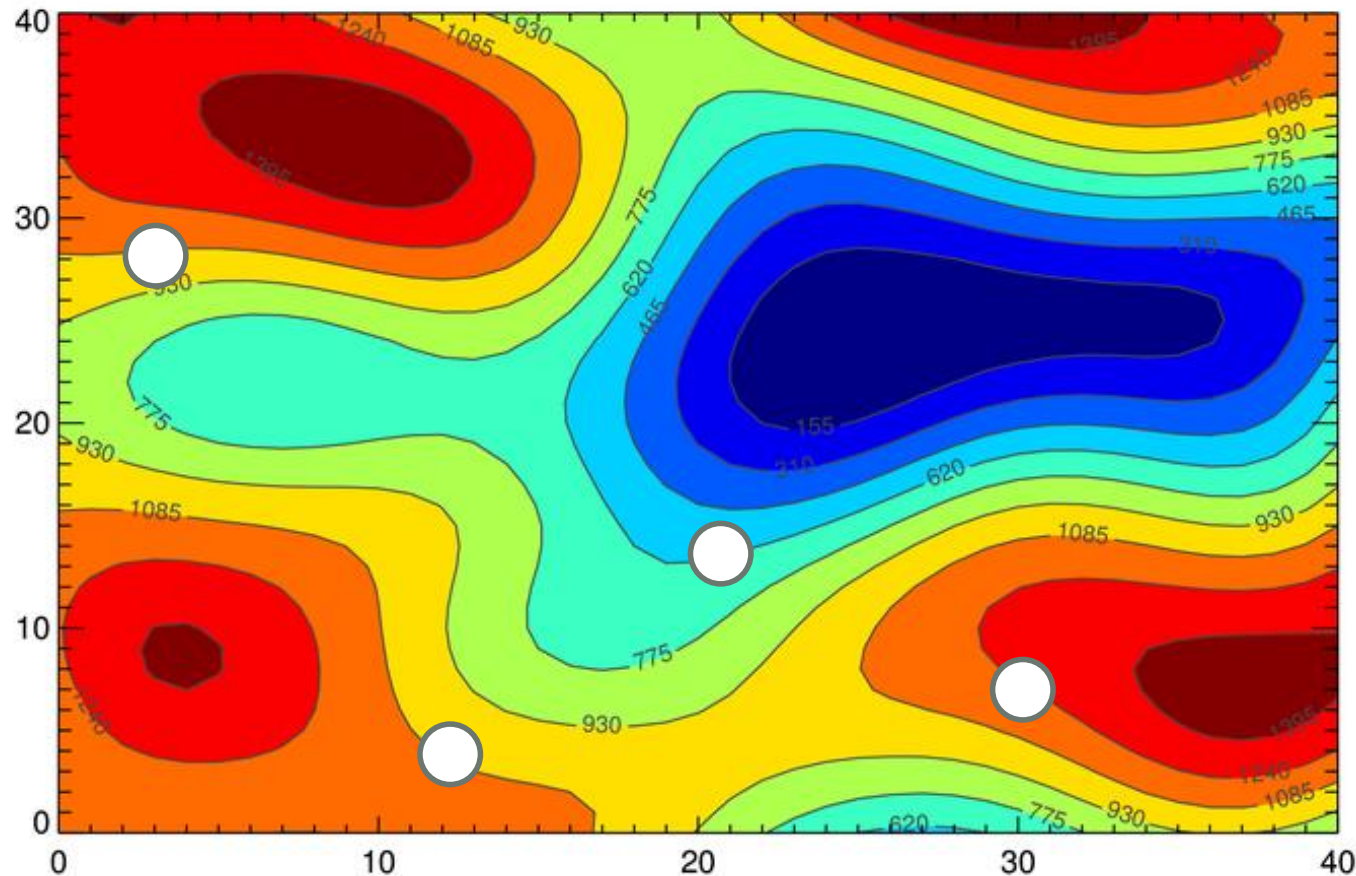
Algoritmo Passo a Passo

Avalia cada partícula com a função de fitness.



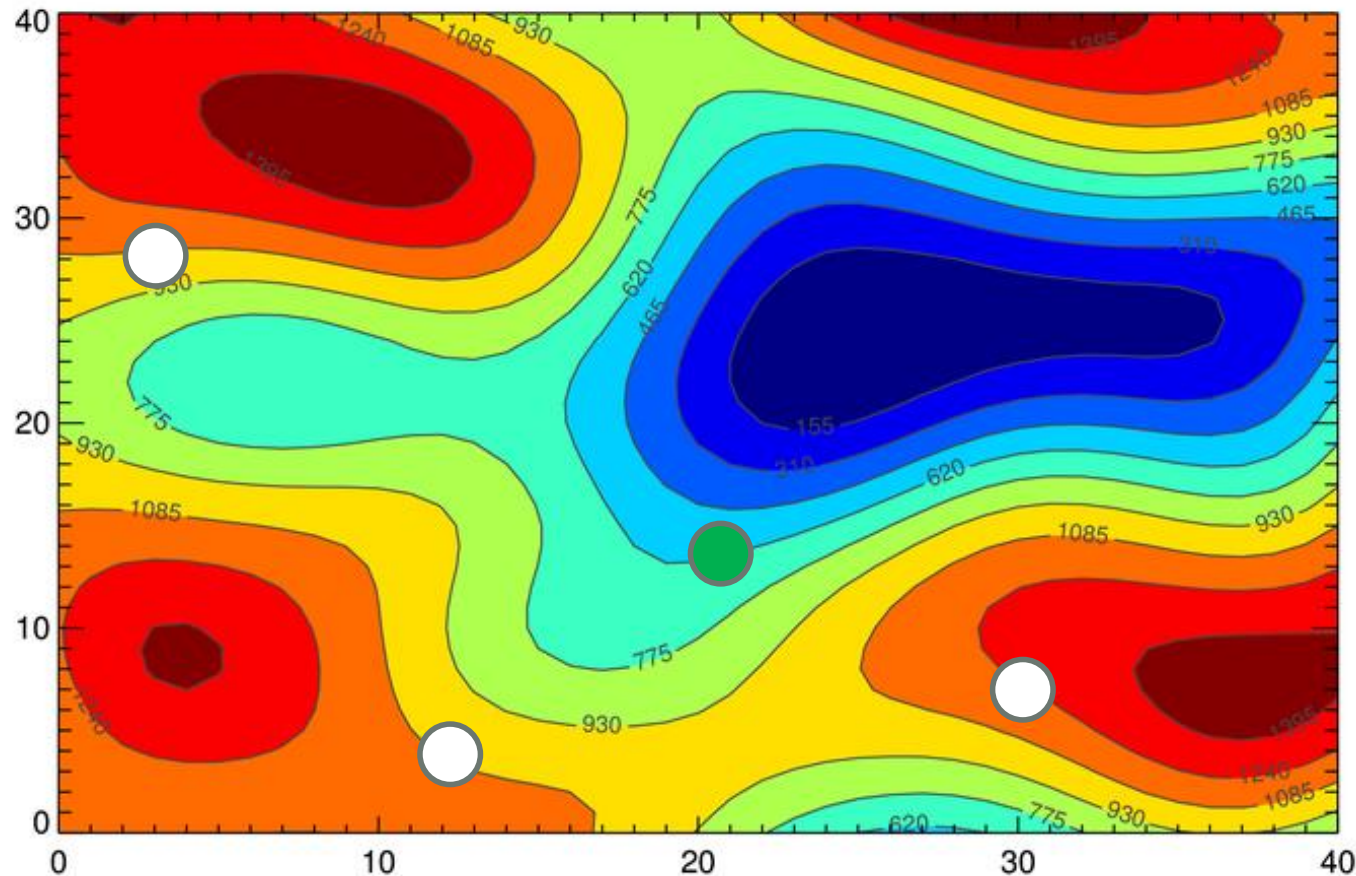
Algoritmo Passo a Passo

Pbest será a posição atual das partículas.



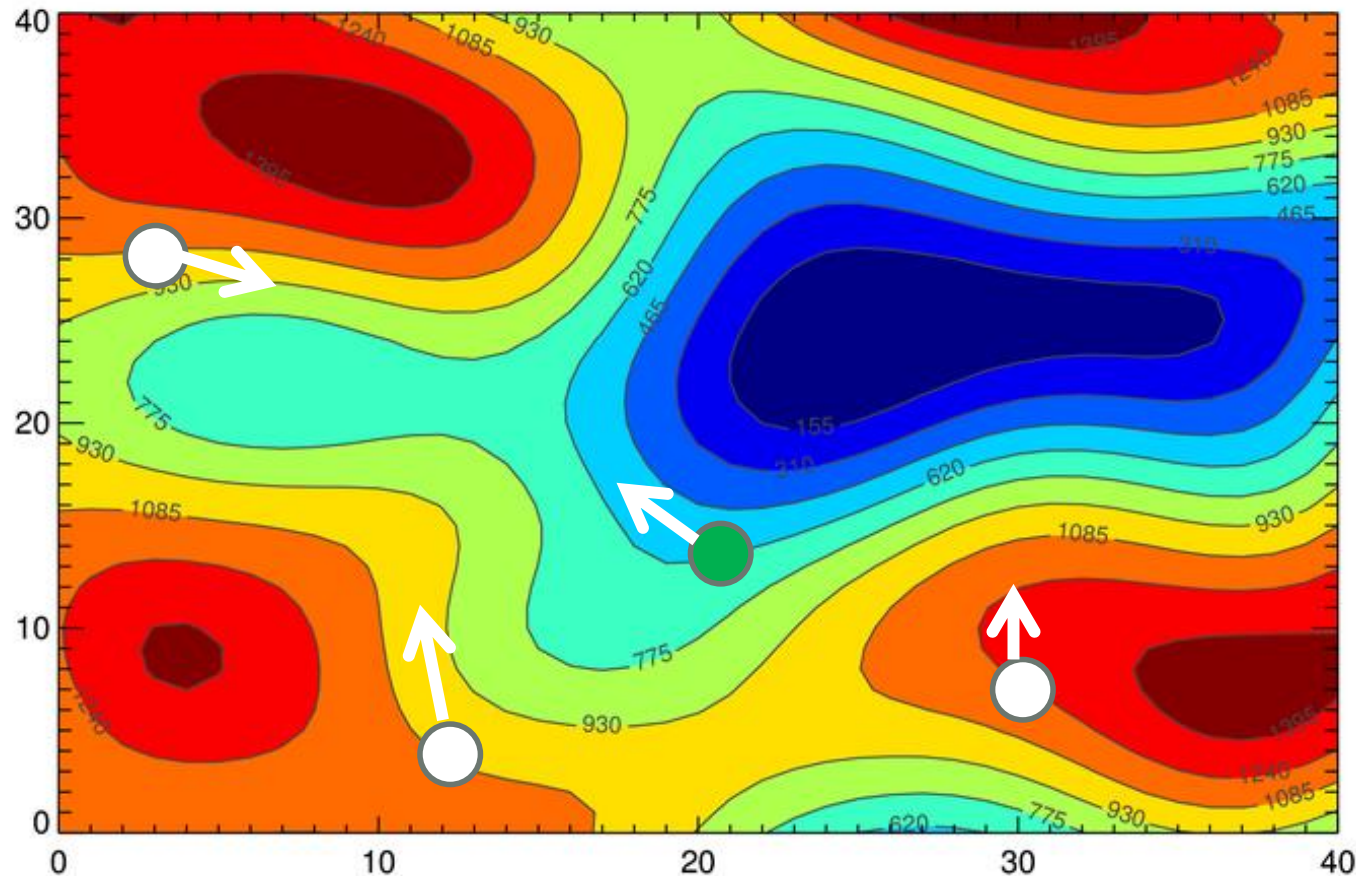
Algoritmo Passo a Passo

Determina a **melhor** partícula.



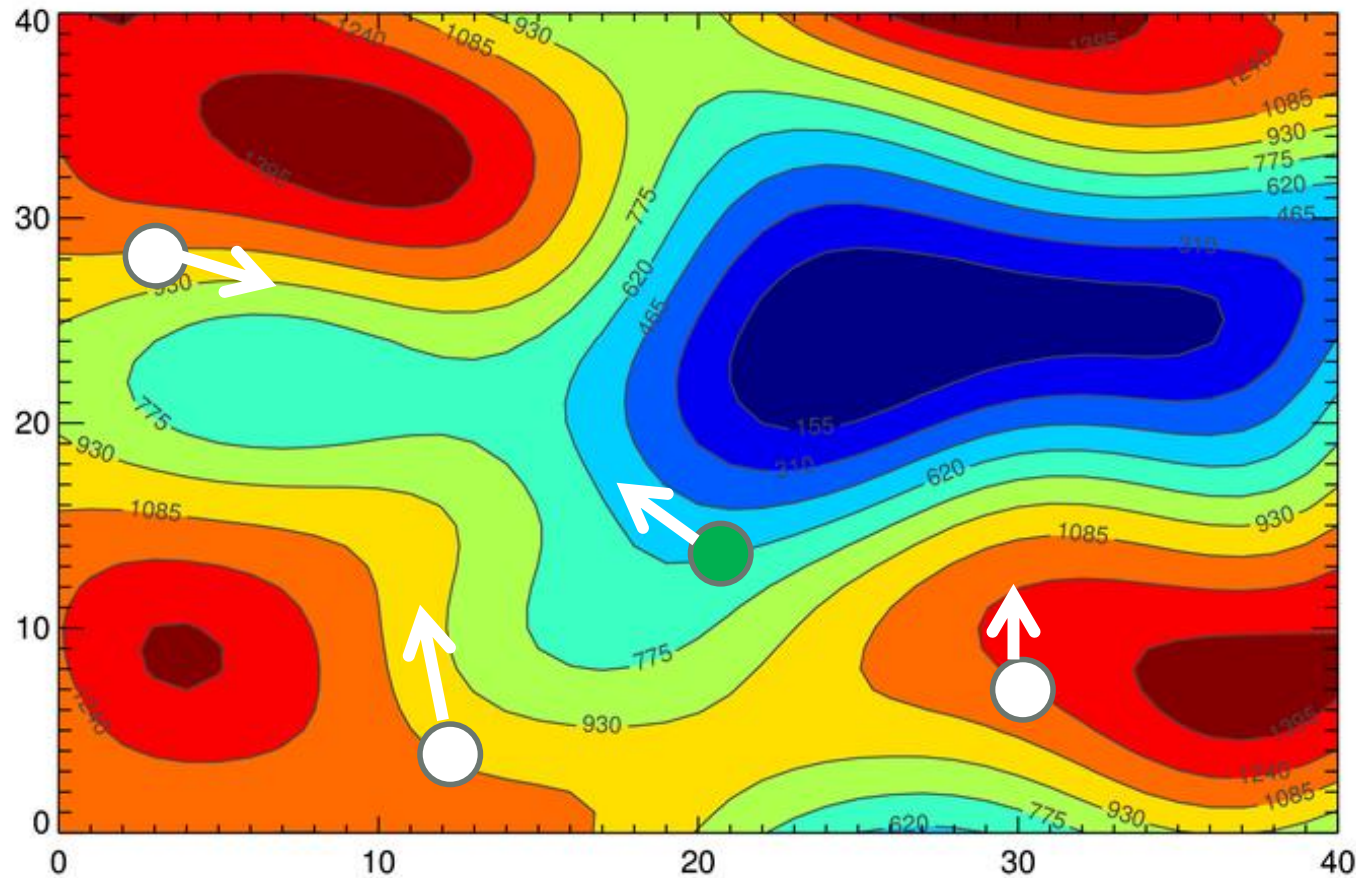
Algoritmo Passo a Passo

Inicializa a velocidade aleatoriamente.



Algoritmo Passo a Passo

Para cada partícula faça:

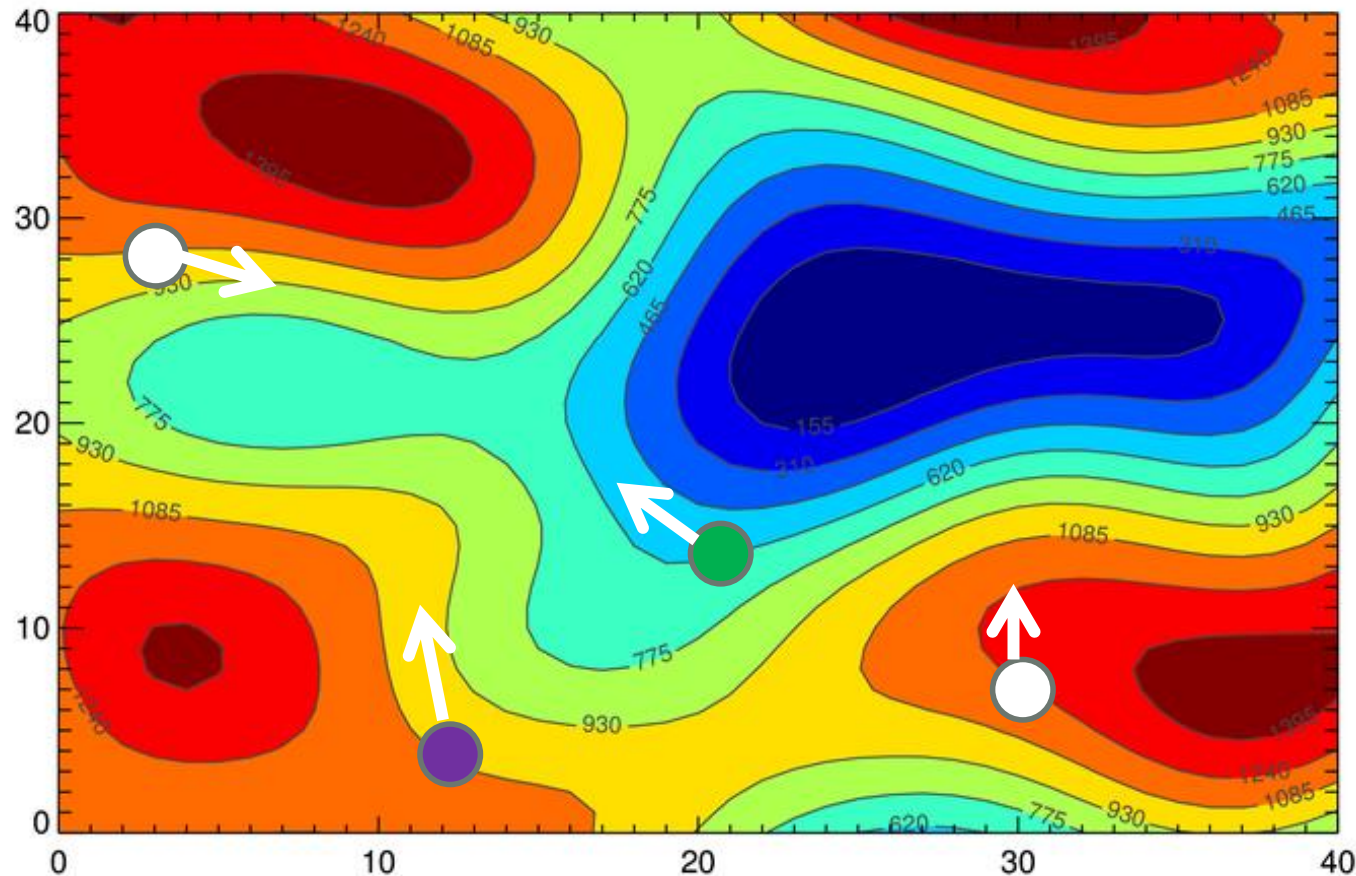


sorteie dois números aleatórios r_1 e r_2



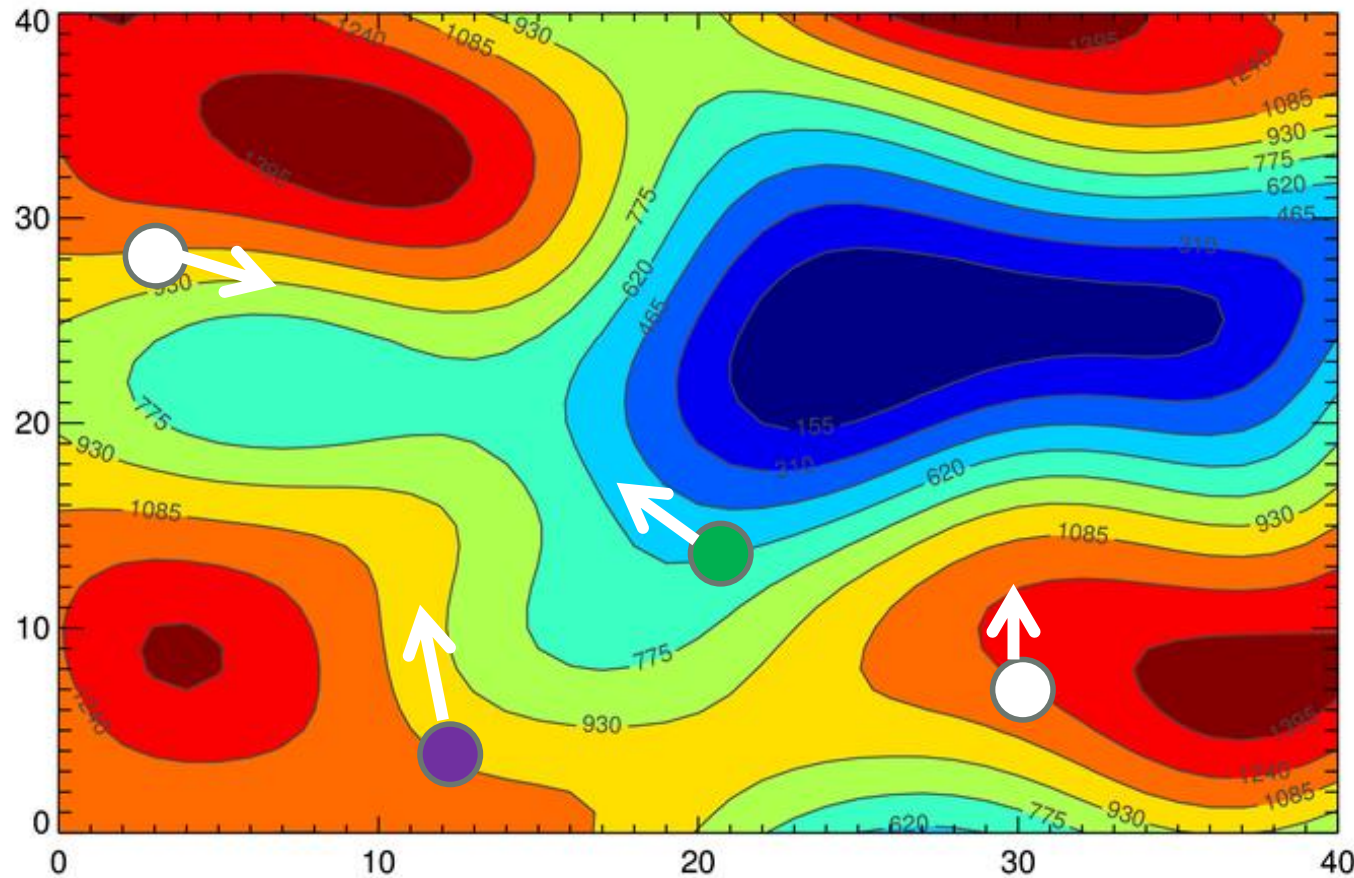
Algoritmo Passo a Passo

Atualiza a velocidade dela como:



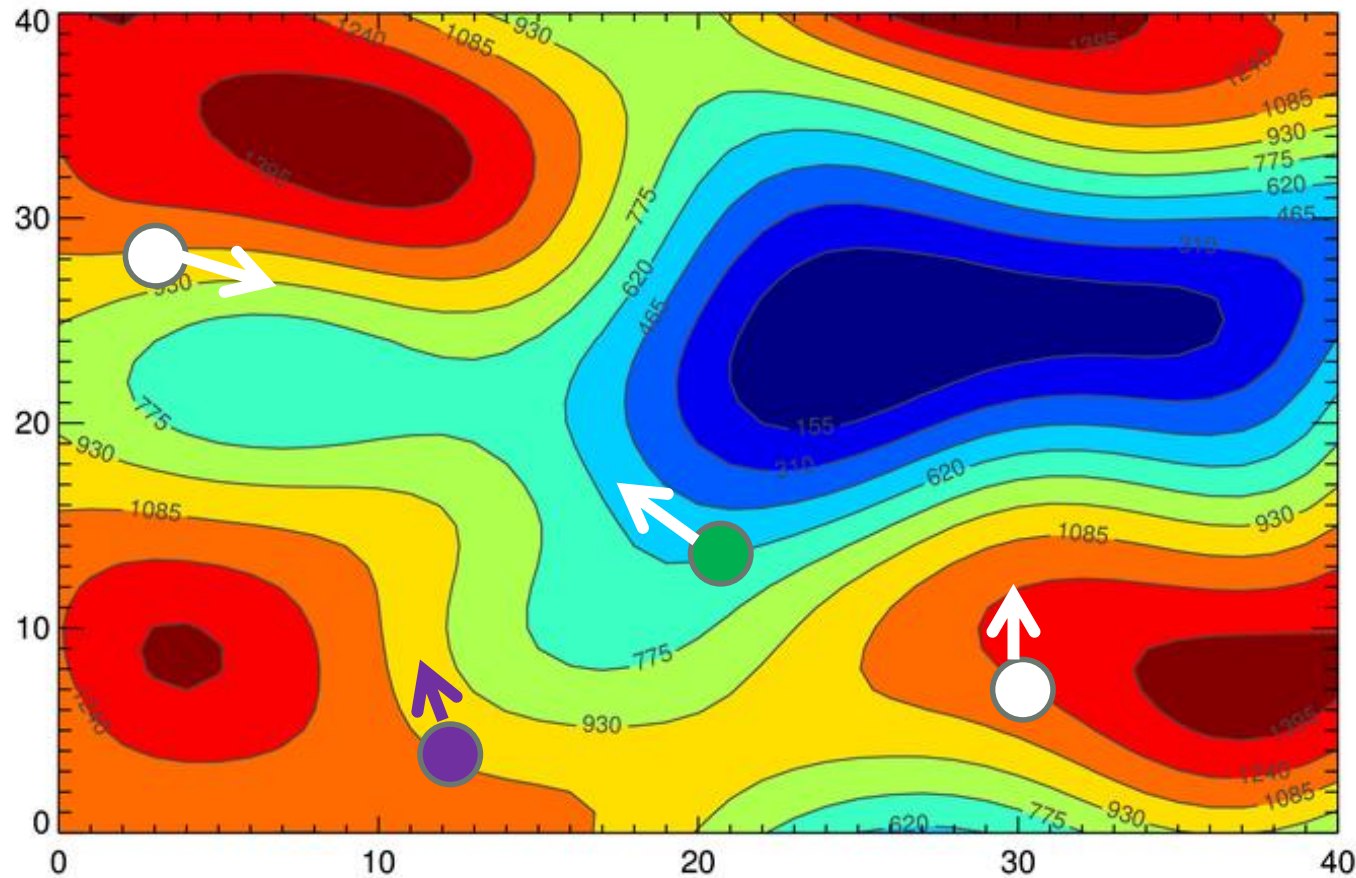
Algoritmo Passo a Passo

$$v[i] = w.v[i] + \phi_1.r_1.(pbest[i] - p[i]) + \phi_2.r_2.(pgbest - p[i])$$



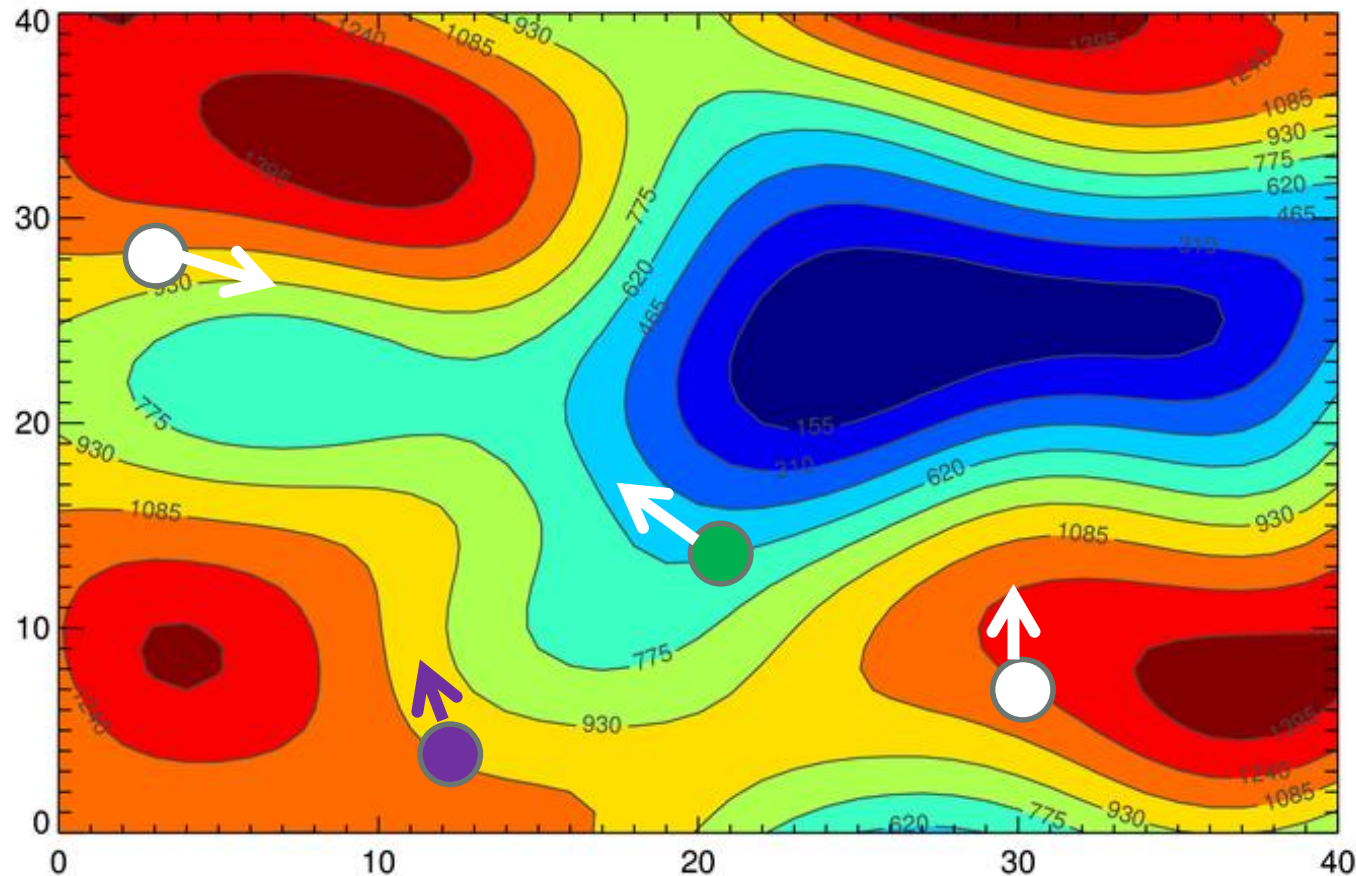
Algoritmo Passo a Passo

$w.v[i]$



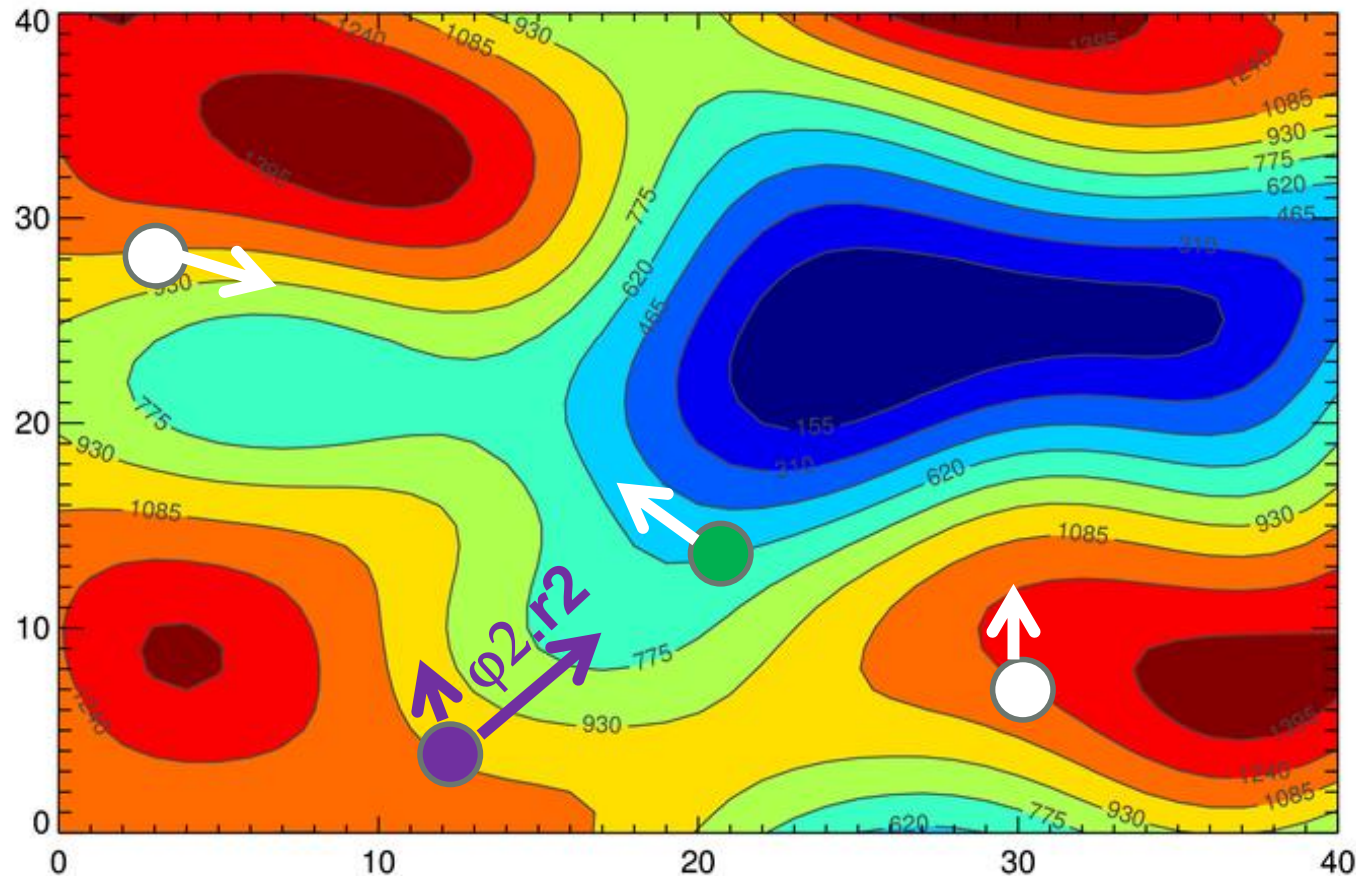
Algoritmo Passo a Passo

$\phi_{1.r1}(\text{pbest}[i] - p[i])$ (inicialmente = 0)



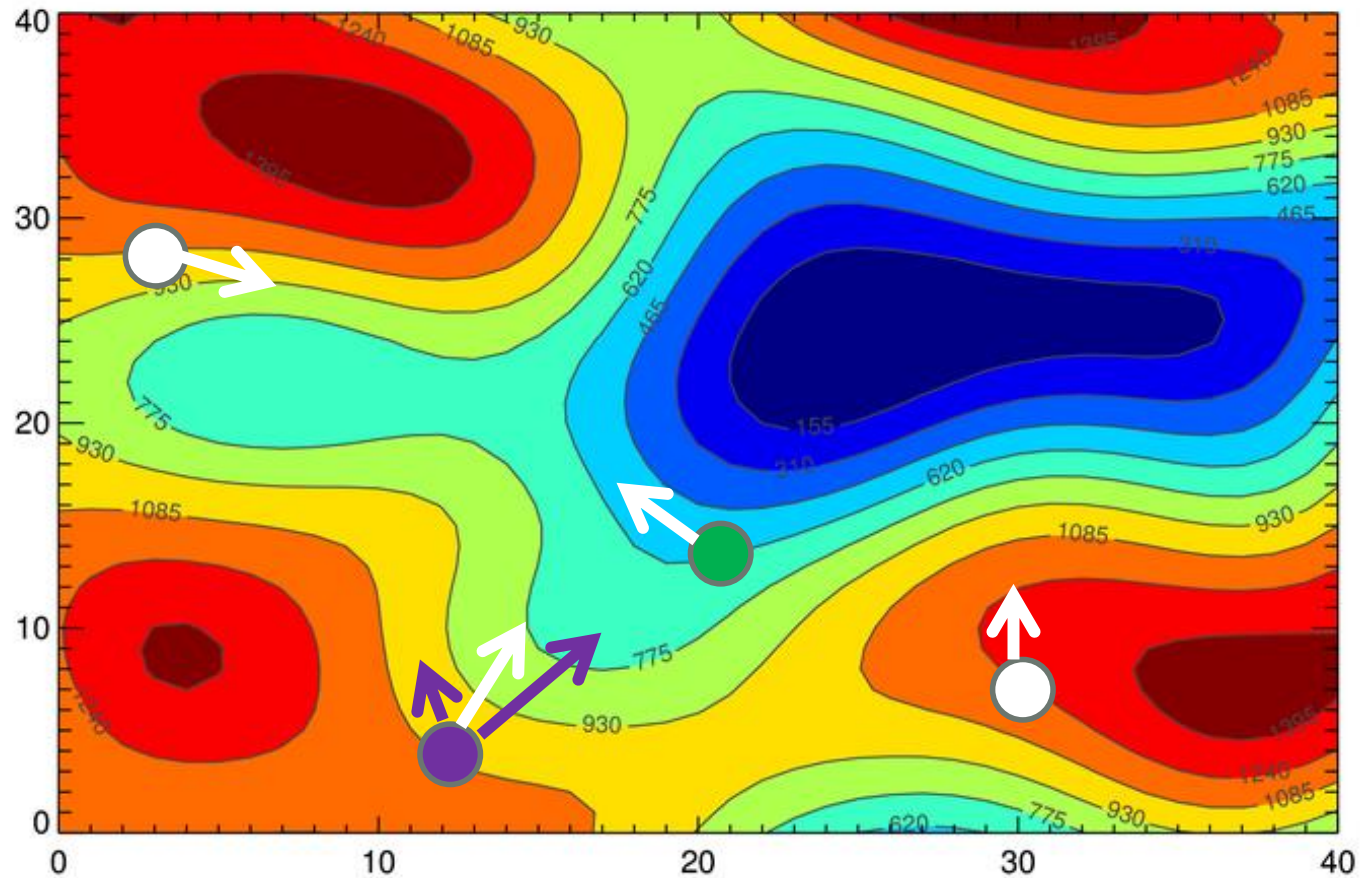
Algoritmo Passo a Passo

$\phi_{2,r2}(\text{pgbest} - p[i])$



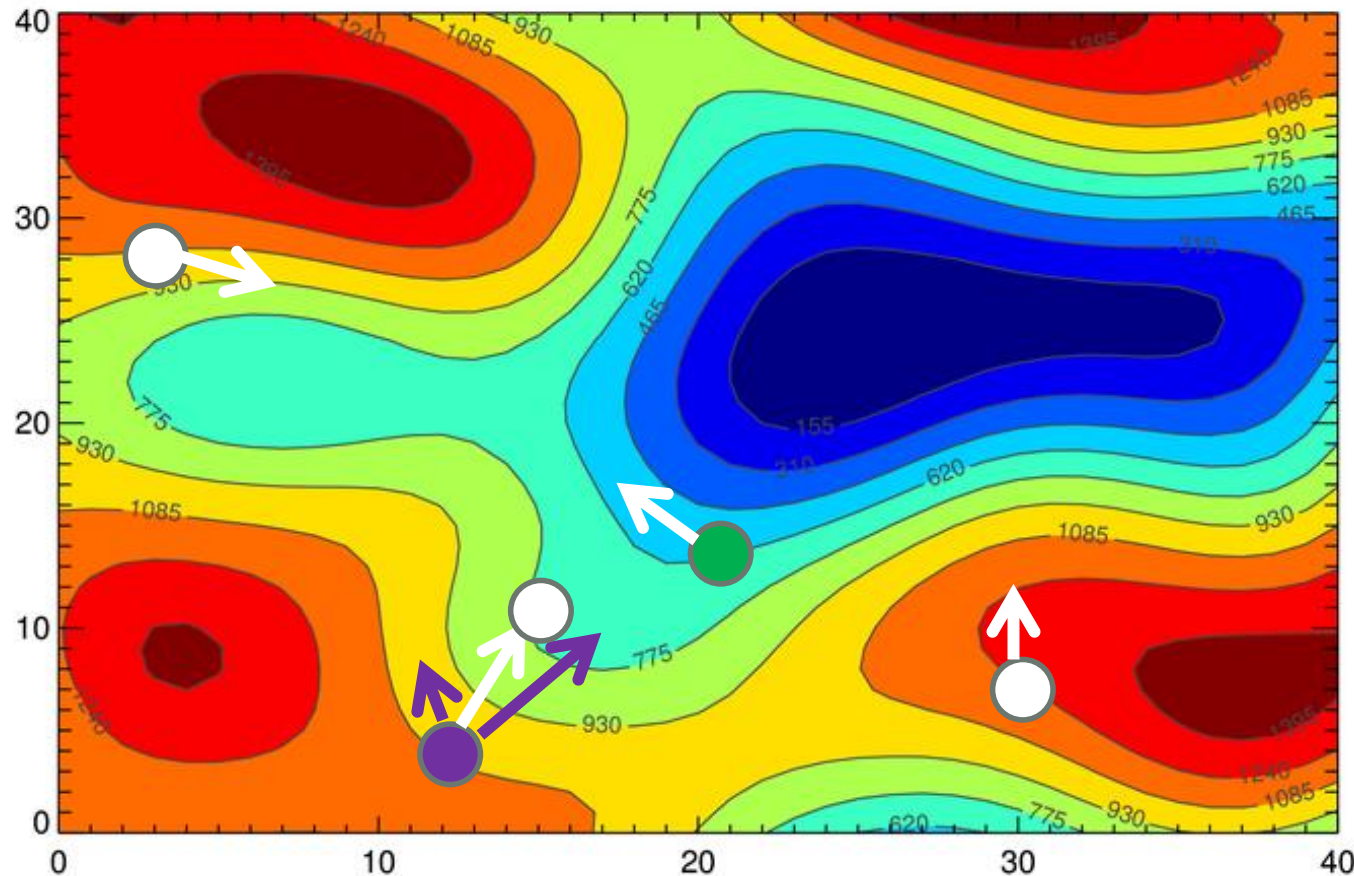
Algoritmo Passo a Passo

Resultado:



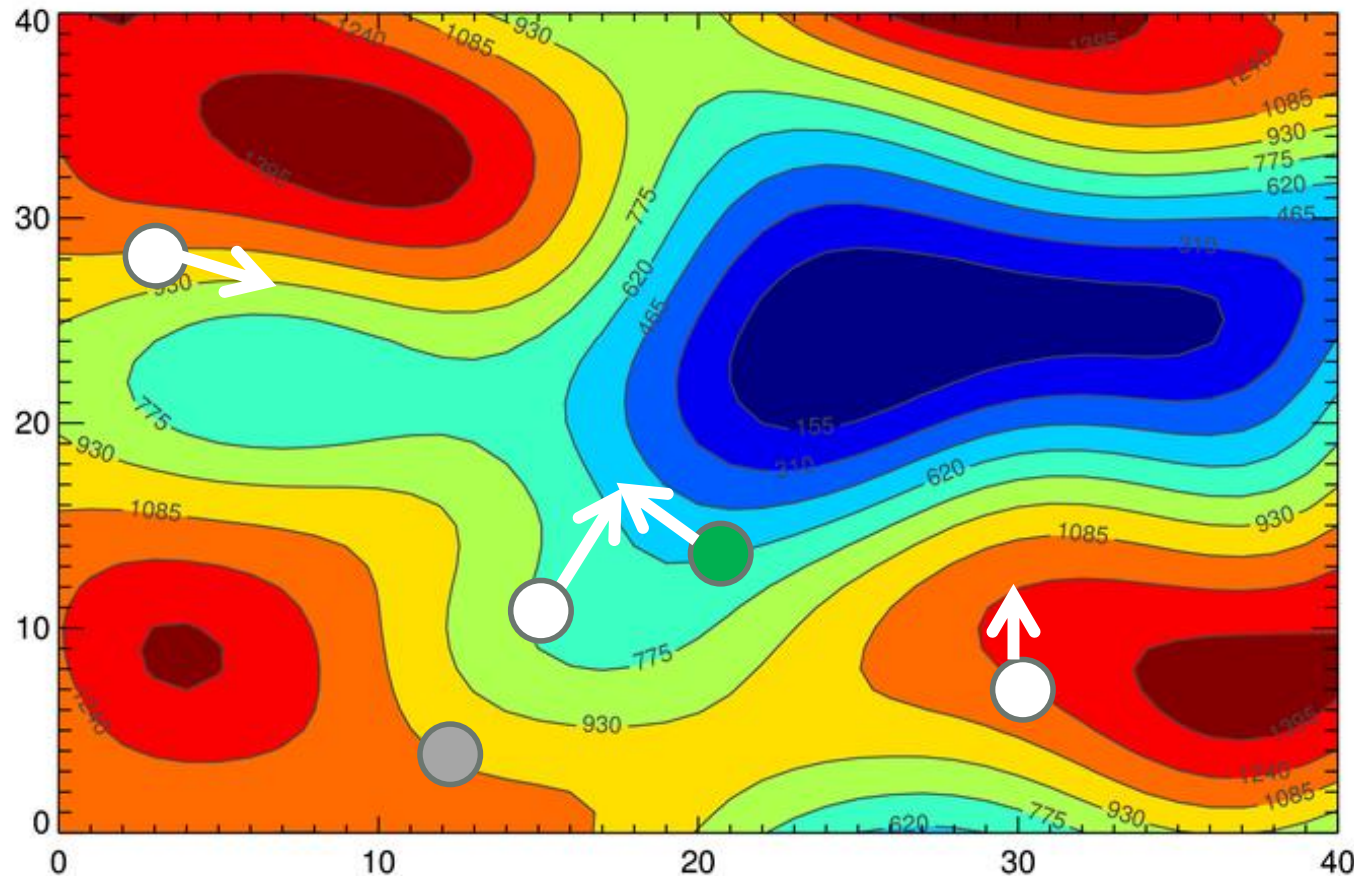
Algoritmo Passo a Passo

Atualiza a posição da partícula com $p[i] = p[i] + v[i]$



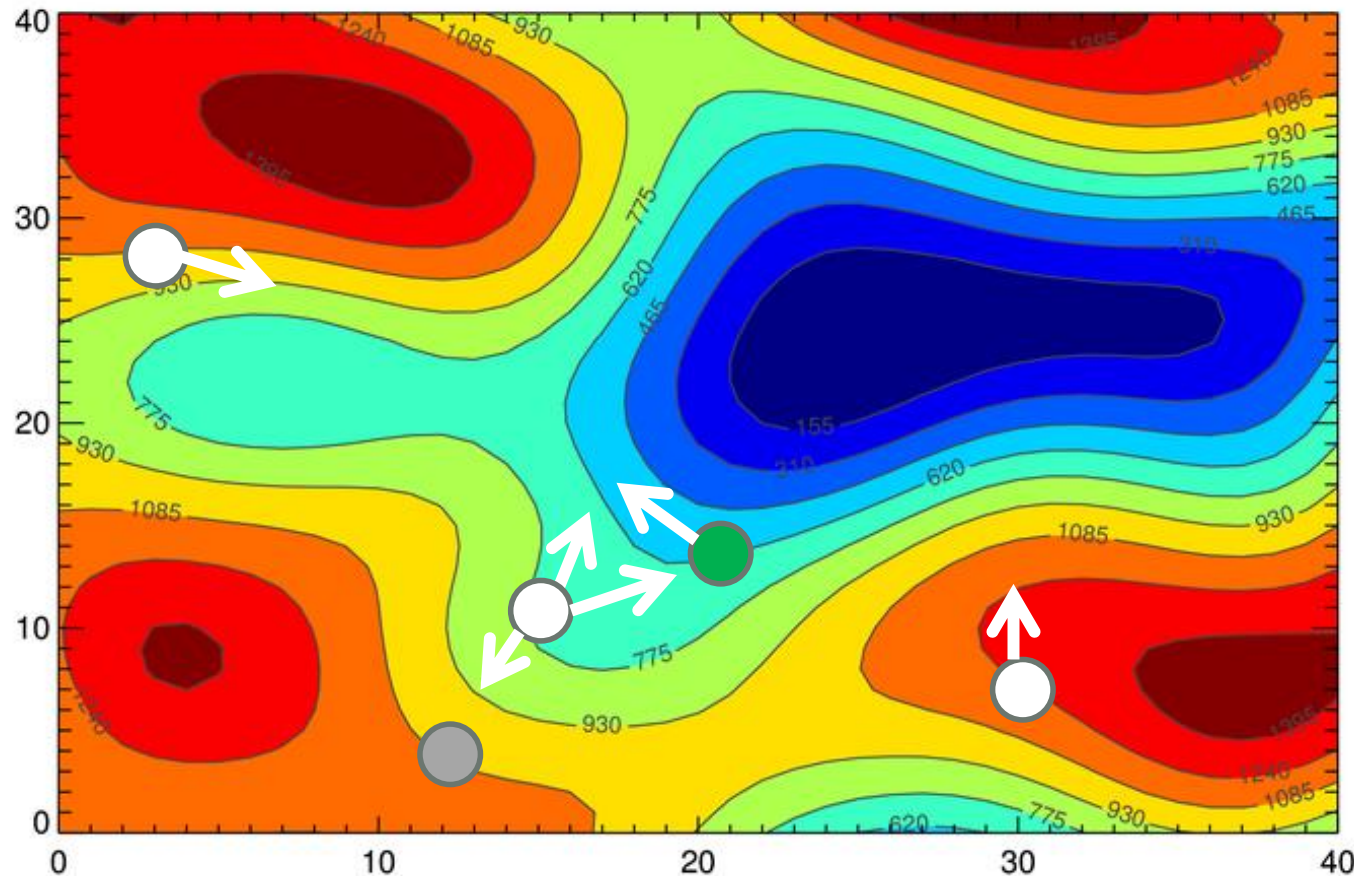
Algoritmo Passo a Passo

Atualiza a **melhor posição da partícula** e **global** se for o caso



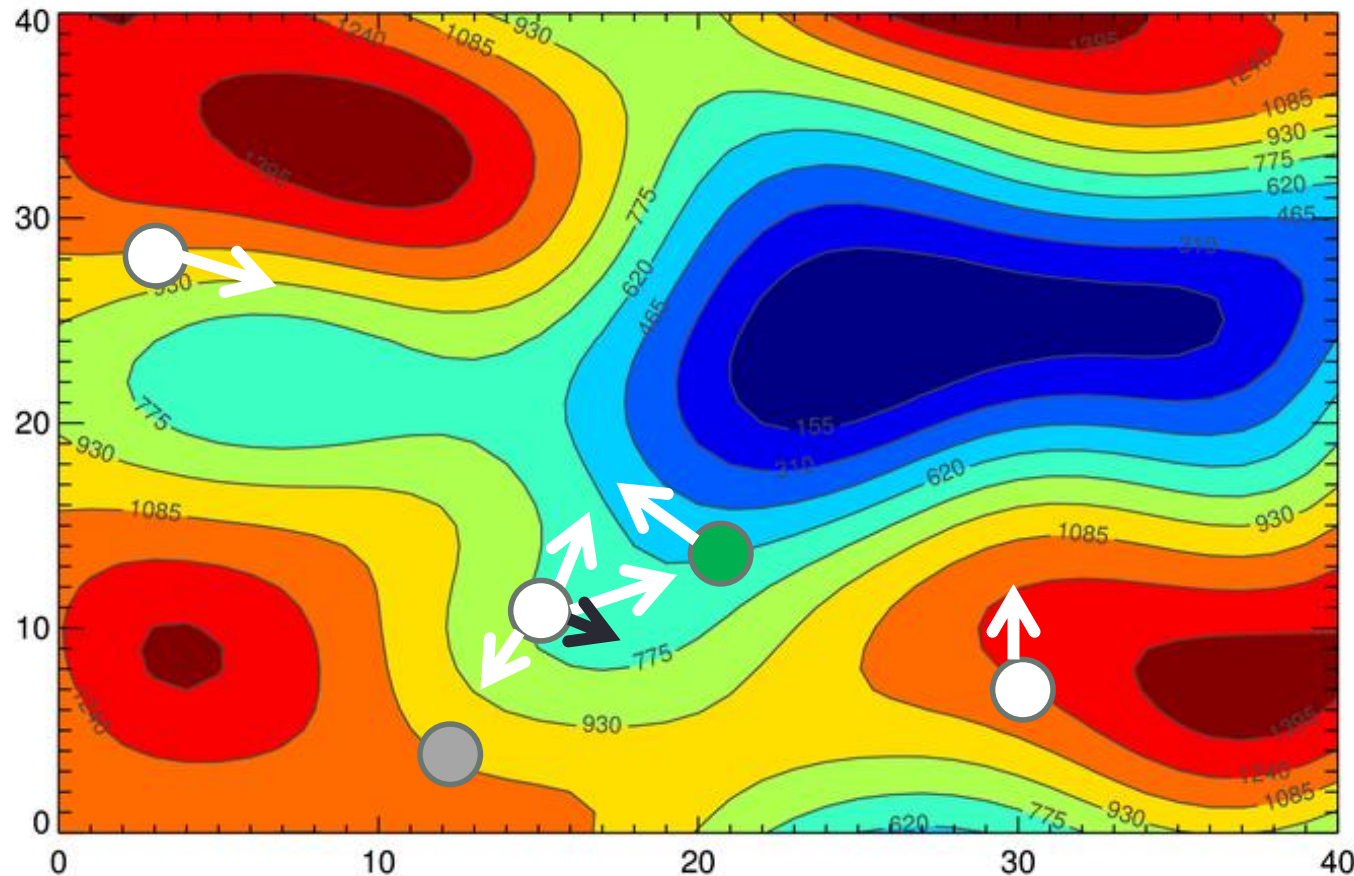
Algoritmo Passo a Passo

Em uma próxima iteração, para essa mesma partícula teremos:



Algoritmo Passo a Passo

Em uma próxima iteração, para essa mesma partícula teremos:



Pseudo-Algoritmo

$$v[i] = w.v[i] + \phi_1.r_1.(pbest[i] - p[i]) + \phi_2.r_2.(pgbest - p[i])$$

$w \rightarrow$ inércia $[0,1]$

$r_1, r_2 \rightarrow$ valores aleatórios do tamanho do passo $(0,1]$

ϕ_1 e $\phi_2 \rightarrow$ pesos da importância entre influência pessoal e social.



Pseudo-Algoritmo

Vamos ver como funciona:

<http://www.macs.hw.ac.uk/~dwcorne/mypages/apps/ps.html>





Universidade Federal do ABC

VARIAÇÕES

Vizinhança Local x Global

Alterar:

$$v[i] = w.v[i] + \phi_1.r_1.(pbest[i] - p[i]) + \phi_2.r_2.(pgbest - p[i])$$

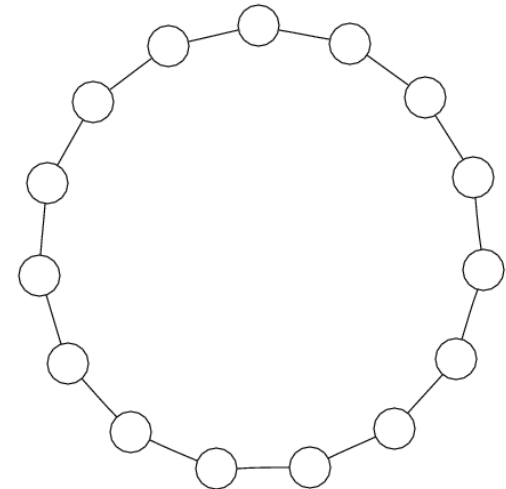
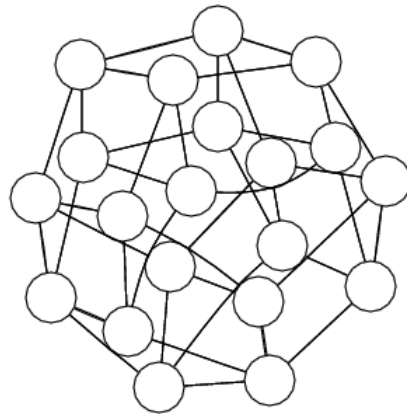
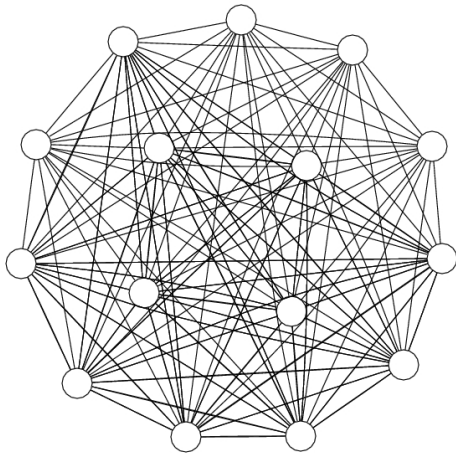
para

$$v[i] = w.v[i] + \phi_1.r_1.(pbest[i] - p[i]) + \phi_2.r_2.(plbest - p[i])$$



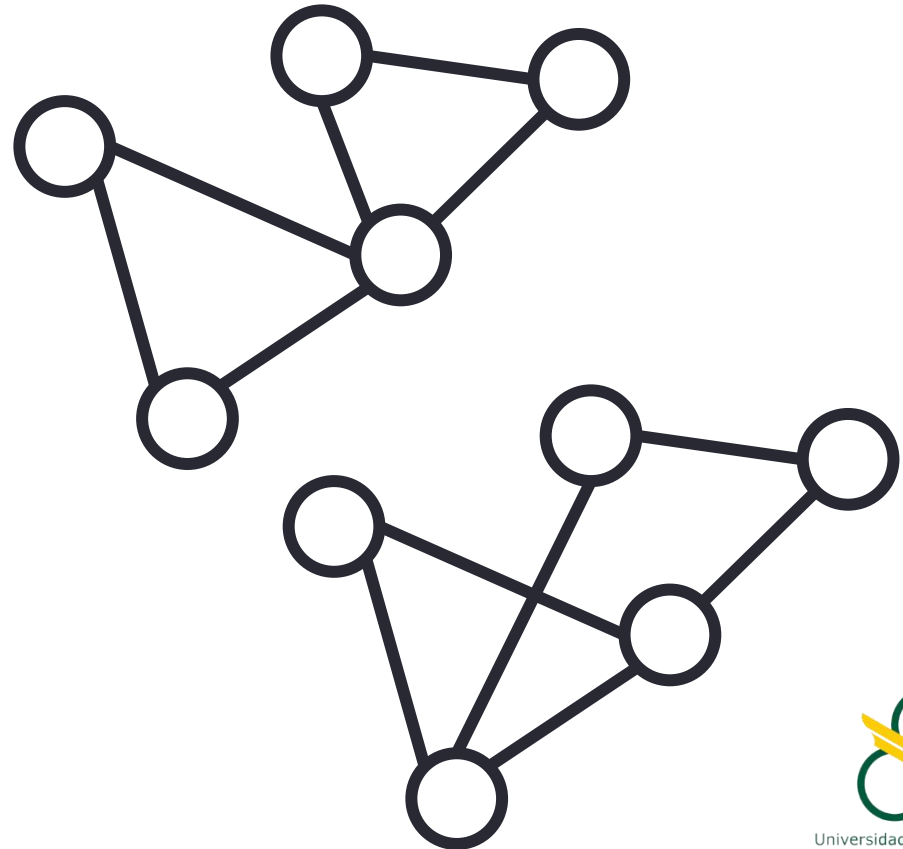
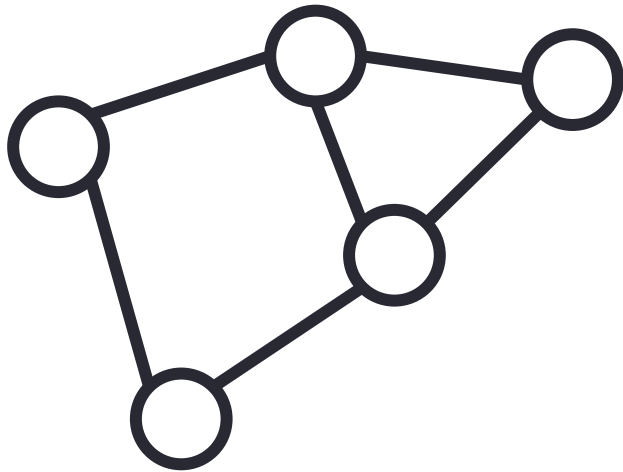
Vizinhança Local x Global

A definição de l_{best} pode ser definida como as N partículas mais próximas ou conectando inicialmente as partículas em rede:



Vizinhança Local x Global

Essa vizinhança pode se adaptar com as iterações:

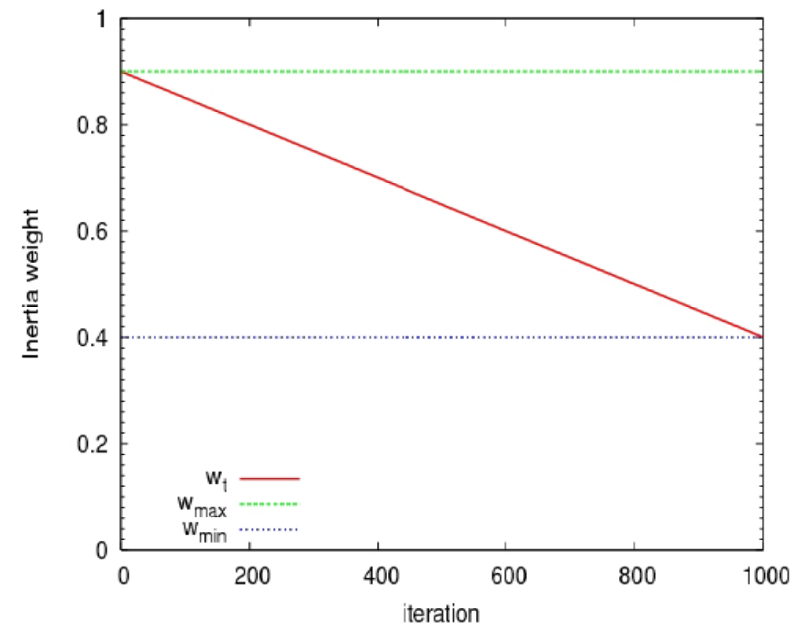


Inércia adaptativa

$$v[i] = w.v[i] + \phi_1.r1.(pbest[i] - p[i]) + \phi_2.r2.(pgbest - p[i])$$

O valor de w pode ser reduzido com o passar das iterações:

- ☐ Linearmente
- ☐ Função não-linear
- ☐ Adaptativo (ES)



PSO Fully Informed

$$v[i] = w.v[i] + \sum_{k \in Ni} \varphi_1.r_1.(pbest[k] - p[i])$$

A velocidade é atualizada de acordo com todos os vizinhos de i .



População Adaptativa

Partículas que não contribuem para a melhoria da solução por um tempo: MATE-AS!!!!

A melhor solução não mudou por um certo tempo:
CRIE NOVAS PARTÍCULAS A PARTIR DA(S)
MELHOR(ES).



Subpopulação

- ❑ Mantenha várias sub-populações em paralelo
- ❑ Comunicação entre populações é restrita
- ❑ Cada população tenta otimizar uma região



Evolutionary PSO

- ❑ A otimização da função-objetivo é tratada pelo PSO
- ❑ A otimização dos parâmetros do PSO é tratada por um Algoritmo Evolutivo

