

Algoritmo de Busca por Enxames de Partículas

Victor França

Recife, março de 2016

Introdução

- Particle Swarm Optimization
 - Desenvolvido por Kennedy (psicólogo social) e Eberhart (engenheiro elétrico)
 - Simular o comportamento social dos seres vivos
 - Inspirado na coreografia dos pássaros
 - Idéia básica: A interação social é capaz de encontrar soluções ótimas para problemas difíceis

Introdução

- Particle Swarm Optimization
 - Cada partícula representa uma possível solução
 - Cada partícula orienta sua busca através:
 - ▮ Própria experiência anterior em já conhecer locais onde costuma encontrar alimento
 - ▮ Informação obtida através de outros indivíduos de seu grupo social

Pseudocódigo

```
1 Inicializa as partículas com valores aleatórios;
2 enquanto (condição de parada não for atingida) faça
3   para (cada partícula) faça
4     avalie a partícula usando a função objetivo;
5     caso o novo fitness seja melhor do que seu pbest, atualize o pbest;
6   para (cada partícula) faça
7     atualize o valor de lbest;
8     atualize a velocidade da partícula;
9     atualize a posição da partícula;
10 fim.
```

Parâmetros

1	1	2	3	4	...	27	28	29	30
⋮					⋮				
30	1	2	3	4	...	27	28	29	30

- Criar 30 Partículas com 30 Dimensões
- Valores das Dimensões = Depende da Função
- Coeficiente Cognitivo $c1 = 2,05$
- Coeficiente Social $c2 = 2,05$
- Velocidade e Posição = Aleatórios
- Pbest e Gbest = Depende do Objetivo

Pseudocódigo

```
1 Inicializa as partículas com valores aleatórios;
2 enquanto (condição de parada não for atingida) faça
3   para (cada partícula) faça
4     avalie a partícula usando a função objetivo;
5     caso o novo fitness seja melhor do que seu pbest, atualize o pbest;
6   para (cada partícula) faça
7     atualize o valor de lbest;
8     atualize a velocidade da partícula;
9     atualize a posição da partícula;
10 fim.
```

Funções de Avaliação

Esfera

$$f_1(\vec{x}) = \sum_{i=1}^D x_i^2$$

Rastrigin

$$f(\mathbf{x}) = An + \sum_{i=1}^n [x_i^2 - A \cos(2\pi x_i)]$$

Rosenbrock

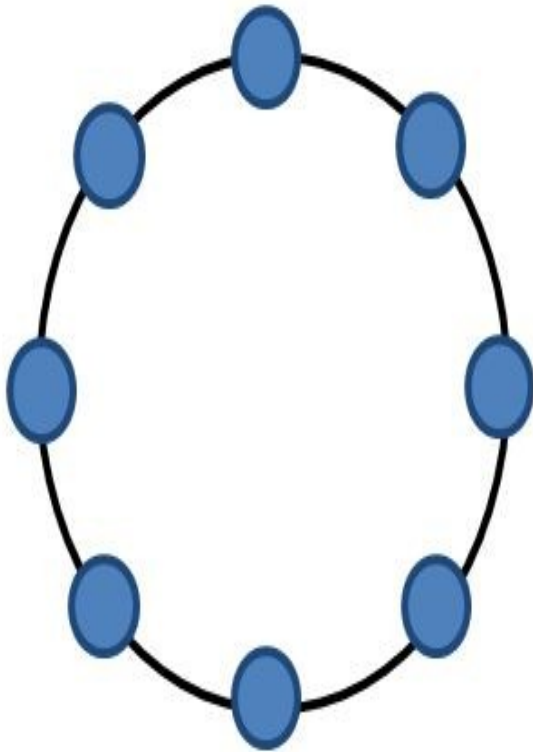
$$f(\mathbf{x}) = \sum_{i=1}^{d-1} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$$

Pseudocódigo

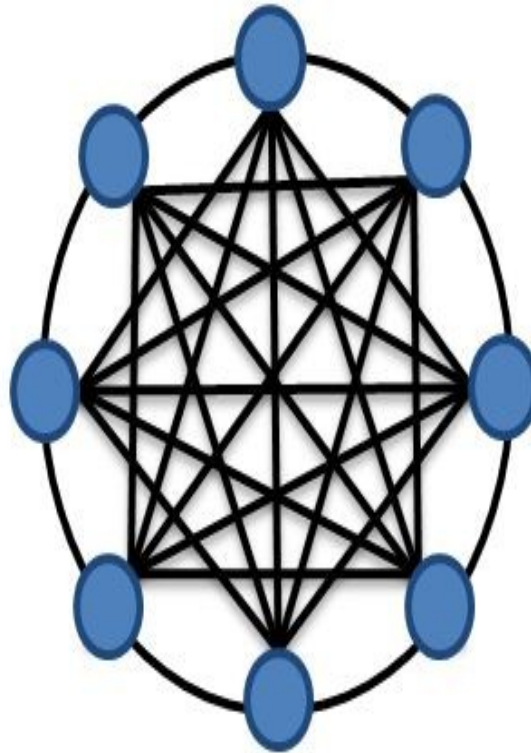
```
1 Inicializa as partículas com valores aleatórios;
2 enquanto (condição de parada não for atingida) faça
3   para (cada partícula) faça
4     avalie a partícula usando a função objetivo;
5     caso o novo fitness seja melhor do que seu pbest, atualize o pbest;
6   para (cada partícula) faça
7     atualize o valor de lbest;
8     atualize a velocidade da partícula;
9     atualize a posição da partícula;
10 fim.
```

Topologia de Comunicação

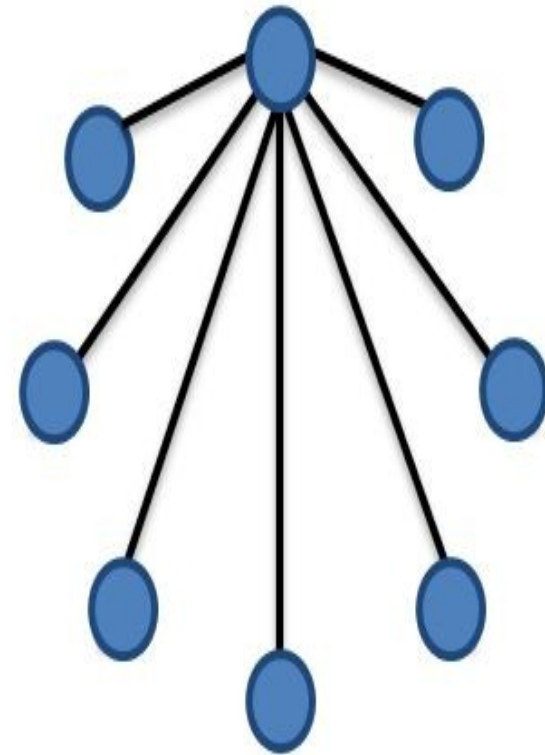
Local



Global



Focal



Pseudocódigo

```
1 Inicializa as partículas com valores aleatórios;
2 enquanto (condição de parada não for atingida) faça
3   para (cada partícula) faça
4     avalie a partícula usando a função objetivo;
5     caso o novo fitness seja melhor do que seu pbest, atualize o pbest;
6   para (cada partícula) faça
7     atualize o valor de lbest;
8     atualize a velocidade da partícula;
9     atualize a posição da partícula;
10 fim.
```

Cálculo da velocidade e da posição

- Particle Swarm Optimization

- Cálculo da velocidade

$$V_i(t+1) = V_i(t) + \\ c1 * \text{rand}(0,1) * [pbest(t) - x_i(t)] + \\ c2 * \text{rand}(0,1) * [gbest(t) - x_i(t)]$$

- Cálculo da posição

$$x_i(t+1) = x_i(t) + v_i(t+1)$$

Pseudocódigo

```
1 Inicializa as partículas com valores aleatórios;
2 enquanto (condição de parada não for atingida) faça
3   para (cada partícula) faça
4     avalie a partícula usando a função objetivo;
5     caso o novo fitness seja melhor do que seu pbest, atualize o pbest;
6   para (cada partícula) faça
7     atualize o valor de lbest;
8     atualize a velocidade da partícula;
9     atualize a posição da partícula;
10 fim.
```

Entendendo os parâmetros

- Particle Swarm Optimization

- Peso de Inércia (W)

$$Vi(t+1) = X * \{ Vi(t) + \\ c1 * rand(0,1) * [pbest(t) - xi(t)] + \\ c2 * rand(0,1) * [gbest(t) - xi(t)] \}$$

- Fator de Construção

$$\chi = \frac{2}{|2 - \varphi - \sqrt{\varphi^2 - 4\varphi}|}, \quad \varphi = c_1 + c_2 > 4$$

Obrigado