

# Algorítmica II. Algoritmos Metaheurísticos

Joaquín Roiz Pagador<sup>1</sup>

Universidad Pablo de Olavide, Carretera Utrera s/n, Sevilla, España,  
quiniroiz@gmail.com

**Abstract.** El siguiente documento pertenece a una serie de documentos que pretende servir a modo de resumen para el temario de Algorítmica II para el Grado en Ingeniería Informática en Sistemas de Información.

**Keywords:** algoritmos, metaheurística, optimización, particle, swarm

## Optimización Nube de Partículas.

Haciendo uso del concepto de banco de peces, bandadas de pájaros o similares, la **Optimización Nube de Partículas** es una *Metaheurística poblacional* inspirada en el **comportamiento social** del vuelo de bandadas de aves y el movimiento de los bancos de peces.

Una población se compone de:

- Partículas (**swarm**).
- Se mueven (**vuelan**).
- **Espacio de búsqueda**.

El movimiento de la partícula p depende de:

- Su posición desde que comenzó el algoritmo(pBest).
- La mejor posición de las partículas de su entorno (IBest) o toda la nube (gBest) desde el comienzo del algoritmo.

En cada iteración, se cambia aleatoriamente la velocidad de p para acercarla a las posiciones pBest y IBest/gBest.

**Se tomará como líder al que tenga mejor puntuación y le siguen.**

*Utilización* Optimización continua. Fue desarrollado en USA, en 1995.

### Características

- Intercambio de información entre agentes de búsqueda.
- Guardar información del mejor propio y del global.
- Implementación sencilla.
- Convergencia rápida y buenas soluciones.

**NOTA:** Los pájaros no saben dónde está la comida, pero sí conocen su distancia a la misma (sintético). Se sigue al ave más cercano a ella.

## PSO.

Particle Swarm Optimization es un sistema multiagente, basado en operaciones simples en las que se comunican los agentes.

Cada partícula se compone de fitness, posición y vector velocidad:

- Tres vectores.
  - Vector  $X \rightarrow$  posición actual.
  - Vector  $pBest \rightarrow$  la mejor solución actual.
  - Vector  $V \rightarrow$  gradiente (dirección).
- Dos valores de fitness.
  - X-Fitness  $\rightarrow$  fitness de la solución actual (vector  $x$ )
  - P-Fitness  $\rightarrow$  fitness de la mejor solución local (vector).  $pBest$ ).

$$\begin{array}{c}
 \overline{P_i} \\
 \\
 X_i = \langle X_{i1}, \dots, X_{in} \rangle \\
 pBest_i = \langle P_{i1}, \dots, P_{in} \rangle \\
 v_i = \langle v_{i1}, \dots, v_{in} \rangle \\
 X_{fitness} = ? \\
 pBest_{fitness} = ?
 \end{array}$$

Se generan posiciones aleatoriamente, velocidades también aleatoriamente, pero distinto a 0 (Porque no salen buenos resultados).  $\rightarrow V_{max}$  será la velocidad máxima.  $[-V_{max}, V_{max}]$

### Partículas y nube.

$$\begin{array}{l}
 V_{id} = V_{id} + \phi_1 \cdot rnd() \cdot (pBest_{id} - X_{id}) + \phi_2 \cdot rnd() \cdot (g_{id} - X_{id}) \\
 X_{id} = X_{id} + V_{id}
 \end{array}$$

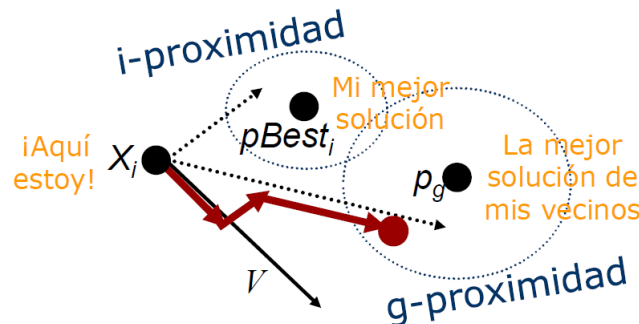
Donde:

- $p_i$  es la partícula en cuestión,  $pBest_{id}$  es la mejor solución encontrada por la partícula.
- $\phi_1, \phi_2$  son ratios de aprendizaje (pesos) que controlan los componentes **cognitivo** y **social**.
- $g$  representa el índice de la partícula con el mejor  $pBest_{fitness}$  del entorno de  $p_i$  (IBest) o de toda la nube (gBest).
- $rnd()$  es un número aleatorio en  $[0, 1]$ .
- $d$  es la  $d$ -ésima dimensión del vector.

### Tipos de PSO.

Encontramos cuatro tipos de PSO.

1. Modelo completo:  $\phi_1, \phi_2 > 0$ .
2. Sólo Cognitivo:  $\phi_1 > 0$  y  $\phi_2 = 0$ .
3. Sólo Social:  $\phi_1 = 0$  y  $\phi_2 > 0$ .
4. Sólo Social exclusivo:  $\phi_1 = 0$ ,  $\phi_2 > 0$  y  $g \neq i$  (la partícula en sí no puede ser la mejor de su entorno).



**Fig. 1.** Representación Gráfica del comportamiento de PSO.

### Pseudocódigo.

```
t = 0
for (i = 1 hasta  $N_{Particulas}$ ) do
  | inicializar  $X_i$  y  $V_i$ ;
end
while (nosecumplalacondiciondeparada) do
  | t  $\leftarrow$  t + 1
  | for (i = 1 hasta  $N_{Particulas}$ ) do
  | | evaluar  $X_i$ 
  | | if ( $F(X_i)$  es mejor que  $F(pBest)$ ) then
  | | |  $pBest \leftarrow X_i$ 
  | | |  $F(pBest_i) \leftarrow F(X_i)$ 
  | | end
  | end
  | for (i = 1 hasta  $N_{Particulas}$ ) do
  | | Escoger  $Ibest$ , la partícula con mejor fitness del entorno de  $X_i$ 
  | | Calcular  $V_i$ , la velocidad media de  $X_i$ , de acuerdo a  $pBest_i$  y  $Ibest_i$ 
  | | Calcular la nueva posición  $X_i$ , de acuerdo a  $X_i$  y  $V_i$ 
  | end
end
Devolver mejor solución encontrada
```

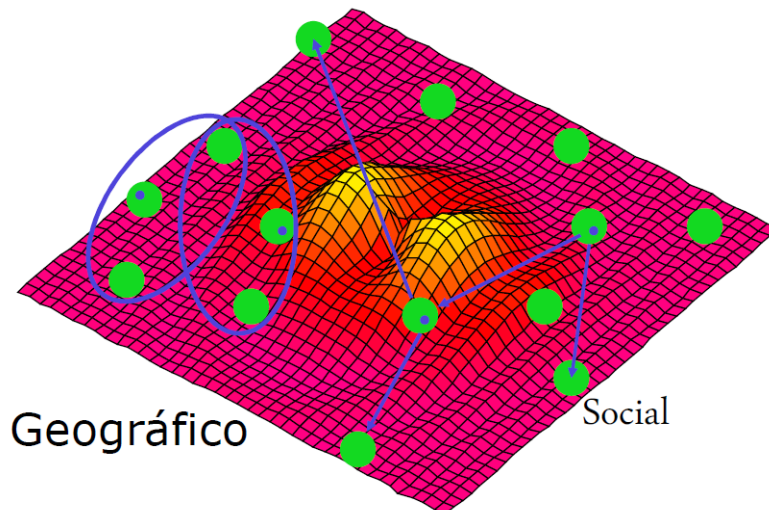
**Algorithm 1:** Algoritmo Optimización Nube de Partículas.

### Valor de los parámetros

- Tamaño de la nube: entre 20 y 40 partículas (problemas simples, 10; problemas muy complejos, 100-200).
- Velocidad máxima:  $V_{max}$  se suele definir a partir del intervalo de cada variable.
- Ratios de aprendizaje: habitualmente,  $\phi_1 = \phi_2 = 2$ .
- PSO Global vs. PSO Local: la versión global converge más rápido pero cae más fácilmente en óptimos locales y viceversa.

Entornos:

1. Geográficos: distancia mínima al resto.
2. Sociales: Se define una lista de vecinas a priori.



**Fig. 2.** Representación gráfica de un Entorno Geográfico para una Optimización Nube de Partículas.

## References

1. Material asignatura, *Metaheurísticas*, UGR. <http://sci2s.ugr.es/graduateCourses/Metaheuristics>
2. E.-G. Talbi, Metaheuristics. From design to implementation, Wiley, 2009 Material.
3. C. Blum, A Roli, Metaheuristics in Combinatorial Optimization: overview and conceptual comparison. ACM Computing Surveys, 35(3), 2003, 268-308.
4. B. Melián, F. Glover. Introducción a la búsqueda tabú, 2006. [http://leeds-faculty.colorado.edu/glover/fred%20pubs/329%20-%20Introduccion%20a%20la%20Busqueda%20Tabu%20TS\\_Spanish%20w%20Belen\(11-9-06\).pdf](http://leeds-faculty.colorado.edu/glover/fred%20pubs/329%20-%20Introduccion%20a%20la%20Busqueda%20Tabu%20TS_Spanish%20w%20Belen(11-9-06).pdf)