

Lista de Algoritmos Heurísticos

Carlos Antonio Macías Duarte

17 de marzo de 2019

1. Recocido Simulado

Algoritmo 1: Recocido Simulado (SA)

Entrada:

n : Tamaño de vecindad

c : Control de temperatura

t : Número de iteraciones

Salida:

g : Mejor solución

1 $i \leftarrow 1$: Contador de iteraciones;

2 $g \leftarrow generarSolucionInicial()$: Solución inicial;

3 **mientras** $i \leq t$ **hacer**

4 $p \leftarrow generarVecindad(n, g)$: Generar nueva vecindad;

5 $j \leftarrow 1$: Contador de la vecindad;

6 **mientras** $j \leq n$ **hacer**

7 **si** p_j mejor que g **entonces**

8 $g \leftarrow p_j$;

9 **si no, si** $\exp(\frac{f(g) - f(p_j)}{c}) > rand(0, 1)$ **entonces**

10 $g \leftarrow p_j$;

11 $i \leftarrow i + 1$;

12 $c \leftarrow generarNuevoTamanoVecindad()$: Genera el nuevo tamaño de la vecindad siguiente;

13 $n \leftarrow generarNuevaTemperatura()$: Genera el nuevo valor de control de temperatura;

14 **devolver** g

2. Algoritmo Genético

Algoritmo 2: Algoritmo Genético (GA)

Entrada:

n : Tamaño de la población

t : Número de generaciones

Salida:

g : Mejor individuo

```
1  $p \leftarrow inicializaPoblación(n)$  : Inicialización la primera generación;  
2  $i \leftarrow 1$  : Contador de iteraciones;  
3  $g \leftarrow mejorIndividuo(p)$  : Mejor mejor individuo de la generación;  
4  $q \leftarrow seleccionarParejas(p)$  : Selección de las parejas;  
5 mientras  $i \leq t$  hacer  
6    $p \leftarrow cruza(q)$  : Proceso de cruza de parejas;  
7    $p \leftarrow mutacion(p)$  : Proceso de Mutación de individuos;  
8    $g \leftarrow mejorIndividuo(p)$  : Obtener el mejor individuo;  
9    $i \leftarrow i + 1$ ;  
10 devolver  $g$ 
```

3. Optimización por Enjambre de Partículas

Algoritmo 3: Optimización por Enjambre de Partículas (PSO)

Entrada:

n : Tamaño del enjambre

t : Número de iteraciones

Salida:

g : Partícula con la mejor posición

```
1  $p \leftarrow inicializaEnjambre(n)$  : Inicialización del enjambre;  
2  $i \leftarrow 1$  : Contador de iteraciones;  
3  $g \leftarrow mejorParticula(p)$  : Mejor partícula;  
4 mientras  $i \leq t$  hacer  
5    $nuevaVelocidad(p, g)$  : Actualiza la velocidad de las partículas;  
6    $nuevaPosicion(p)$  : Calcular nuevas posiciones;  
7    $g \leftarrow mejorParticula(p)$  : Obtener la Mejor partícula;  
8    $i \leftarrow i + 1$ ;  
9 devolver  $g$ 
```
