



## AAD (Versión 2021, 1.0)

# Algoritmos de Optimización Basados en Colonias de Hormigas

### Objetivos

El objetivo de esta actividad dirigida es estudiar el funcionamiento de los *Algoritmos de Optimización Basados en Colonias de Hormigas (OCH)*. Para ello, se requerirá que el alumno implemente distintas variantes de estos algoritmos, correspondientes al *Sistema de Hormigas (SH)*, *Sistema de Hormigas Elitista (SHE)* y *SCH* (Sistema de colonias de hormigas), para resolver el problema del *Viajante de Comercio*. El comportamiento de los algoritmos de OCH implementados deberá compararse con un *algoritmo greedy*.

### Enunciado de la Actividad

El Problema del Viajante de Comercio (TSP) es uno de los problemas de optimización combinatoria más conocidos. En su formulación más general, dadas una serie de ciudades, el objetivo consiste en encontrar el circuito de menor coste que parta de una ciudad concreta, pase por todas las demás una sola vez y retorne a la ciudad de origen.

En nuestro caso trabajaremos con tres instancias estándar del problema (dos *data sets* modelo), obtenidas de la biblioteca TSPLIB, todas ellas correspondientes al TSP simétrico. Éstas son:

- Ch130: Tamaño 130 ciudades. Coste de la solución óptima: 6.110
- A280: Tamaño 280 ciudades. Coste de la solución óptima: 2.579

Todos estos ficheros presentan el mismo formato, una lista con dos valores para cada ciudad que representan sus coordenadas en el plano. Para componer la matriz de costes se deberá calcular la *distancia euclídea* entre cada par de ciudades  $(i, j)$ . Los costos obtenidos han de ser números enteros, es decir, no se considerarán decimales. Así, la matriz de distancias se calcularía en lenguaje C de la siguiente forma:

```
xd = x[i] - x[j];  
yd = y[i] - y[j];  
dij = rint( sqrt( xd*xd + yd*yd ) );
```

donde `rint` es la función de redondeo y `sqrt` es la raíz cuadrada.

El alumno implementará el SH (Sistema de Hormigas) SHE (Sistema de Hormigas Elitista) y SCH (Sistema de colonias de hormigas) para resolver el problema, con los distintos *data sets*.

A continuación, mostramos los valores de los parámetros a considerar en las ejecuciones de estos algoritmos, así como el número de ejecuciones que habría que hacer en cada caso, partiendo siempre de una semilla distinta en cada ejecución para el generador de números aleatorios:

SH, SHE y SCH:

- Número de ejecuciones a realizar: 5.
- Número de hormigas:  $m = 10$
- Tiempo de ejecución (criterio de parada):  $10000 \cdot n$
- Valor de los parámetros de la regla de transición:  $\alpha = 1$  y  $\beta = 2$
- Número de hormigas elitistas para SHE será:  $e = 15$ .
- Valor del parámetro de evaporación en la actualización de feromona:  $\rho = 0,1$ .
- Cantidad inicial de feromona  $\tau_0 = 1 / (n \cdot L)$ , siendo  $n$  el tamaño de la instancia del problema y  $L$  el coste del circuito devuelto por la heurística *greedy* implementada.
- Aporte de feromona en el mecanismo de actualización:

$$f(C(S_k)) = \frac{1}{C(S_k)}$$

- Para SCH, Regla de actualización local (SCH):  $\varphi=0.1$
- Regla de transición:  $\alpha=1$ ,  $\beta=2$ ,  $q_0=0.98$  (SCH)

El alumno tendrá que contabilizar el número de evaluaciones (llamadas realizadas a la función de coste) producidas por los distintos algoritmos, que será empleado como una medida adicional de comparación de su calidad.

A partir de la experimentación efectuada, se obtendrán varias tablas en las que se incluirán los resultados obtenidos por cada algoritmo en cada instancia, así como la media y desviación típica de estos, con la forma de la Tabla 1.1.

	<i>Caso 1</i>		<i>Caso 2</i>		<i>Caso 3</i>	
	<i>Coste</i>	<i>Ev.</i>	<i>Coste</i>	<i>Ev.</i>	<i>Coste</i>	<i>Ev.</i>
Ejecución 1	x	x	x	x	x	x
Ejecución 2	x	x	x	x	x	x
Ejecución 3	x	x	x	x	x	x
Ejecución 4	x	x	x	x	x	x
Ejecución 5	x	x	x	x	x	x
Media	x	x	x	x	x	x
Desv. Típ.	x	x	x	x	x	x

Tabla 1.1

Finalmente, se construirá una tabla global de resultados con la estructura mostrada en la Tablas 1.2. El dato entre paréntesis debajo del nombre de la instancia muestra el coste de la solución óptima (o la mejor encontrada) de la instancia, valor que debe ser considerado en el análisis de resultados. La columna etiquetada con *Mej* indica el valor de la mejor solución encontrada, la columna  $\sigma$  refiere a la desviación típica y la columna etiquetada con *Ev* indica el número medio de evaluaciones realizadas por el algoritmo en las 5 ejecuciones.

A partir de los datos mostrados en estas tablas, **el alumno realizará un análisis** de los resultados obtenidos, **que influirá en la calificación de la actividad**. En dicho análisis, se deben comparar, para cada instancia, las distintas variantes de los algoritmos en términos de: número de evaluaciones, mejor resultado individual obtenido y mejor resultado medio (robustez del algoritmo), y cualquier otro tipo de consideración de interés que descubra en el análisis.

Las prácticas se realizarán individualmente

### **Fecha y Método de Entrega;**

El día del examen final de la asignatura. Debe entregar 1 fichero comprimido ZIP ó RAR, que contenga:

- Documento DOC (MS Word) ó PDF con las tablas de resultados y descripción somera de la estructura del código y listado de las soluciones encontradas conjuntamente con las semillas utilizadas.
- Ficheros de código fuente completo ejecutable utilizado.
- Scripts o cualquier otra información necesaria para la ejecución.