

**UNIVERSIDAD DE GRANADA**  
**E.T.S. DE INGENIERÍA INFORMÁTICA**



**Departamento de Ciencias de la  
Computación e Inteligencia Artificial**

# **Bioinformática**

<http://sci2s.ugr.es/docencia/bioinformatica>

## **Guión de Prácticas**

**Práctica 1:**  
**Algoritmos de Optimización Basados en**  
**Colonias de Hormigas**

Curso 2011-2012

Quinto Curso de la Ingeniería en Informática

# Práctica 1

## Algoritmos de Optimización Basados en Colonias de Hormigas

### 1. Objetivos

El objetivo de esta práctica es estudiar el funcionamiento de los *Algoritmos de Optimización Basados en Colonias de Hormigas (OCH)*<sup>1</sup>. Para ello, se requerirá que el alumno utilice distintas variantes de estos algoritmos, correspondientes al *Sistema de Hormigas*, el *Sistema de Colonias de Hormigas*, *Sistema de Hormigas Max-Min* y *Sistema de Hormigas Mejor-Peor*, para resolver seis instancias del *Viajante de Comercio* con y sin búsqueda local.

La práctica se califica sobre 2,5 puntos. La fecha de entrega es el Jueves 19 de abril de 2012.

### 2. El Problema del Viajante de Comercio

El Problema del Viajante de Comercio (TSP) es uno de los problemas de optimización combinatorial 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 seis instancias del problema, obtenidas de la biblioteca TSPLIB<sup>2</sup>, todas ellas correspondientes al TSP simétrico. Serán las siguientes:

- eil76: Tamaño 76 ciudades. Coste de la solución óptima: 538
- kroA100: Tamaño 100 ciudades. Coste de la solución óptima: 21282

---

<sup>1</sup><http://iridia.ulb.ac.be/~mdorigo/ACO/ACO.html>

<sup>2</sup><http://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95>

- d198: Tamaño 198 ciudades. Coste de la solución óptima: 15780
- lin318: Tamaño 318 ciudades. Coste de la solución óptima: 42029
- att532: Tamaño 532 ciudades. Coste de la solución óptima: 27686
- rat783: Tamaño 783 ciudades. Coste de la solución óptima: 8806

Dichas instancias (junto con algunas soluciones óptimas útiles para estudiar los resultados obtenidos) se encuentran en la página web de la asignatura. Todos estos ficheros presentan el mismo formato, una lista con dos valores para cada ciudad que representan sus coordenadas en el plano. La matriz de costes sería calculada utilizando la *distancia euclídea* entre cada par de ciudades  $(i, j)$ :

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

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

### 3. Algoritmos de Optimización Basados en Colonias de Hormigas

El alumno tendrá que utilizar distintas variantes de algoritmos de OCH (dos por algoritmo), tomando como base cuatro algoritmos concretos: el *Sistema de Hormigas* (SH), el *Sistema de Colonias de Hormigas* (SCH), *Sistema de Hormigas Max-Min* (SHMM) y *Sistema de Hormigas Mejor-Peor* (SHMP) para resolver el problema del Viajante de Comercio en las seis instancias propuestas. A continuación se indican dichas variantes:

- *Sistema de Hormigas* — El alumno aplicará el SH con y sin búsqueda local para resolver el problema. El algoritmo de búsqueda local a emplear será el 3-opt, que se ha explicado en las clases de teoría (ver página 65 del Tema 3 de teoría). A partir de este momento, nos referiremos a estos dos algoritmos como *SH* y *SH-BL*.
- *Sistema de Colonias de Hormigas* — Al igual que en el caso anterior, el alumno aplicará dos variantes del SCH, sin y con búsqueda local 3-opt (*SCH* y *SCH-BL* respectivamente).
- *Sistema de Hormigas Max-Min* — Igualmente con dos variantes del SHMM, sin y con búsqueda local 3-opt (*SHMM* y *SHMM-BL* respectivamente).
- *Sistema de Hormigas Mejor-Peor* — También con dos variantes del SHMP, sin y con búsqueda local 3-opt (*SHMP* y *SHMP-BL* respectivamente).

Todos los algoritmos mencionados se encuentran implementados en la página web de la asignatura.

## 4. Aplicación de los Algoritmos Conocidos al Viajante de Comercio (TSP)

Esta práctica comprende tanto la descripción y aplicación de los algoritmos propuestos para la resolución del TSP:

- SH y SH-BL.
- SCH y SCH-BL.
- SHMM y SHMM-BL.
- SHMP y SHMP-BL.

como el análisis de resultados derivado de la ejecución de todos ellos sobre las seis instancias del problema:

- eil76    • kroA100    • d198    • lin318    • att532    • rat783

Como se ha indicado anteriormente, la práctica se calificará sobre un total de 2,5 puntos, correspondiendo 1 punto a la descripción de los algoritmos y 1,5 puntos a la experimentación y al análisis de los resultados. Los algoritmos deben ser descritos muy brevemente haciendo referencia a la forma en que los pasos más importantes están implementados en los códigos disponibles (estructuras de datos y aspectos fundamentales del código). Es decir, debe demostrarse que se han estudiado antes de su uso, chequeando el código disponible y relacionándolo con la teoría.

## 5. Compilación, Ejecuciones, Valores de Parámetros y Tablas de Resultados

Los códigos de los distintos algoritmos que se van a utilizar están disponibles en la web de la asignatura en el siguiente fichero Zip: ACOTSP.V1.0.zip<sup>3</sup>. En el fichero README

---

<sup>3</sup>Este código es el proporcionado por Thomas Stützle en la página web de Marco Dorigo sobre los algoritmos OCH (<http://www.aco-metaheuristic.org/>). Hay disponible una nueva versión publicada recientemente, la ACOTSP.V1.02, en el enlace <http://www.aco-metaheuristic.org/aco-code/public-software.html>, pero no hemos trabajado con ella.

incluido en el paquete ACOTSP.V1 se puede encontrar una descripción del contenido e indicaciones sobre como compilar y ejecutar las distintas versiones de los algoritmos.

El código ha sido desarrollado en ANSI C bajo Linux. Lo primero que el alumno debe hacer es compilar dicho código. Para compilar bajo Linux debe ejecutarse ‘make’ en la línea de comandos. Se generará el ejecutable ‘acotsp’ que podrá ser utilizado con los parámetros adecuados para realizar toda la experimentación. Puesto que ha sido implementado en ANSI C no debe ser difícil compilarlo en Windows teniendo en cuenta la información contenida en el fichero ‘Makefile’.

A diferencia de los determinísticos, los algoritmos probabilísticos se caracterizan por la toma de decisiones aleatorias a lo largo de su ejecución. Este hecho implica que un mismo algoritmo probabilístico aplicado al mismo caso de un problema pueda comportarse de forma diferente en cada ejecución.

Cuando se analiza el comportamiento de un algoritmo probabilístico en un caso de un problema, se desea que el resultado obtenido no esté sesgado por una secuencia aleatoria concreta que pueda influir positiva o negativamente en las decisiones tomadas durante su ejecución. Por tanto, resulta necesario efectuar varias ejecuciones con distintas secuencias probabilísticas y calcular el resultado medio de todas las ejecuciones para representar con mayor fidelidad su comportamiento. En nuestro caso se realizarán **5 ejecuciones independientes**.

A continuación, mostramos los valores de los parámetros a considerar en las ejecuciones de los algoritmos considerados en esta práctica:

■ **SH y SH-BL:**

- Número de ejecuciones a realizar: 5. Al llamar al ejecutable se puede indicar con `-tries`.
- Número de hormigas:  $m = 15$ . Al llamar al ejecutable se puede indicar con `-ants`.
- Tiempo de ejecución (criterio de parada): Cada algoritmo de OCH se ejecutará durante un tiempo determinado que supondrá el criterio de parada. Este tiempo dependerá de la instancia del problema: 5 min. (eil76 y kroA100), 10 min. (d198 y lin318) y 15 min. (att532 y rat783). Al llamar al ejecutable se puede indicar con `-time` (en segundos).
- Valor de los parámetros de la regla de transición:  $\alpha = 1$  y  $\beta = 2$ . Al llamar al ejecutable se puede indicar con `-alpha` y `-beta` respectivamente.
- Valor del parámetro de evaporación en la actualización de feromona:  $\rho = 0,2$ . Al llamar al ejecutable se puede indicar con `-rho`.
- Búsqueda local y número de vecinos para la búsqueda local: 3-opt y 40 vecinos. Al llamar al ejecutable se puede indicar con `-localsearch` y `-nnls` respectivamente.

Teniendo en cuenta todos estos parámetros y que la instancia se indica con `-tsplibfile`, que el valor del óptimo se puede indicar con `-optimum` y es usado para los cálculos de resultados y que SH se indica con `-as`, tendríamos la siguiente llamada para el caso de SH aplicado en eil76:

```
./acotsp -tsplibfile eil76.tsp -tries 5 -as -localsearch 0 -alpha 1 -beta 2 -rho 0.2 -time 300 -optimum 538 -ants 15
```

Y la siguiente llamada para el caso de SH-BL:

```
./acotsp -tsplibfile eil76.tsp -tries 5 -as -localsearch 3 -nnls 40 -alpha 1 -beta 2 -rho 0.2 -time 300 -optimum 538 -ants 15
```

(Nota: no hacer copiar-pegar del pdf para evitar problemas con los juegos de caracteres, es mejor escribirlo a mano para evitar sorpresas.)

- **SCH y SCH-BL:** Se considerarán los mismos valores que en el SH, además de los siguientes:
  - Valor del parámetro  $q_0$  en la regla de transición: 0,8. Al llamar al ejecutable se puede indicar con `-q0`.
  - Valor del parámetro de evaporación en la actualización local de feromona:  $\varphi = 0,2$ . No hace falta indicarlo, lo tiene fijado por defecto.

Teniendo en cuenta que SCH se indica con `-acs`, tendríamos la siguiente llamada para el caso de SCH aplicado en eil76:

```
./acotsp -tsplibfile eil76.tsp -tries 5 -acs -localsearch 0 -alpha 1 -beta 2 -rho 0.2 -q0 0.8 -time 300 -optimum 538 -ants 15
```

Y la siguiente llamada para el caso de SCH-BL:

```
./acotsp -tsplibfile eil76.tsp -tries 5 -acs -localsearch 3 -nnls 40 -alpha 1 -beta 2 -rho 0.2 -q0 0.8 -time 300 -optimum 538 -ants 15
```

- **SHMM y SHMM-BL:** Se considerarán los mismos valores que en el SH. Teniendo en cuenta que SHMM se indica con `-mmas`, tendríamos la siguiente llamada para el caso de SHMM aplicado en eil76:
 

```
./acotsp -tsplibfile eil76.tsp -tries 5 -mmas -localsearch 0 -alpha 1 -beta 2 -rho 0.2 -time 300 -optimum 538 -ants 15
```

Y la siguiente llamada para el caso de SHMM-BL:

```
./acotsp -tsplibfile eil76.tsp -tries 5 -mmas -localsearch 3 -nnls 40 -alpha 1 -beta 2 -rho 0.2 -time 300 -optimum 538 -ants 15
```
- **SHMP y SHMP-BL:** Se considerarán los mismos valores que en el SH, además de los siguientes:

- Probabilidad de mutación:  $P_m = 0,3$ . No hace falta indicarlo, lo tiene fijado por defecto.
- Iteraciones sin mejora: 0,3. No hace falta indicarlo, lo tiene fijado por defecto.

Teniendo en cuenta que SHMP se indica con `-bwas`, tendríamos la siguiente llamada para el caso de SHMP aplicado en `eil76`:

```
./acotsp -tsplibfile eil76.tsp -tries 5 -bwas -localsearch 0 -alpha 1 -beta 2 -rho 0.2
-time 300 -optimum 538 -ants 15
```

Y la siguiente llamada para el caso de SHMP-BL:

```
./acotsp -tsplibfile eil76.tsp -tries 5 -bwas -localsearch 3 -nnls 40 -alpha 1 -beta 2
-rho 0.2 -time 300 -optimum 538 -ants 15
```

A partir de la experimentación efectuada, se obtendrán varias tablas en las que se incluirán los resultados obtenidos por cada algoritmo en todas las instancias, así como la media y desviación típica de estos, con la forma de la tabla 1.1. Estas tablas (una por algoritmo) se incluirán en un apéndice, nunca en la sección de análisis de resultados.

Tabla 1.1: Resultados del algoritmo X en los distintos casos del problema

	<i>eil76</i>	<i>...</i>	<i>rat783</i>
	<i>Coste</i>	<i>Coste</i>	<i>Coste</i>
Ejecución 1	x	x	x
Ejecución 2	x	x	x
Ejecución 3	x	x	x
Ejecución 4	x	x	x
Ejecución 5	x	x	x
Media	x	x	x
Desv. Típ.	x	x	x

Finalmente, se construirán dos tablas globales de resultados (una con los tres problemas más sencillos y otra con los tres más complicados) con la estructura mostrada en las tablas 1.2 y 1.3. El dato entre paréntesis debajo del nombre de la instancia muestra el coste de la solución óptima 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 y la columna  $\sigma$  se refiere a la desviación típica. Éstas tablas si se incluirán en la sección de análisis de resultados.

A partir de los datos mostrados en estas tablas, el alumno realizará un análisis de los resultados obtenidos, **que influirá significativamente en la calificación de la práctica**. En dicho análisis, se deben comparar, para cada instancia de cada problema,

Tabla 1.2: Resultados globales de los algoritmos del Bloque 1 (más simples)

<i>Modelo</i>	<i>eil76</i> (538)			<i>kroA100</i> (21282)			<i>d198</i> (15780)		
	<i>Med</i>	<i>Mej</i>	$\sigma$	<i>Med</i>	<i>Mej</i>	$\sigma$	<i>Med</i>	<i>Mej</i>	$\sigma$
<b>SH</b>	x	x	x	x	x	x	x	x	x
<b>SCH</b>	x	x	x	x	x	x	x	x	x
<b>SHMM</b>	x	x	x	x	x	x	x	x	x
<b>SHMP</b>	x	x	x	x	x	x	x	x	x
<b>SH-BL</b>	x	x	x	x	x	x	x	x	x
<b>SCH-BL</b>	x	x	x	x	x	x	x	x	x
<b>SHMM-BL</b>	x	x	x	x	x	x	x	x	x
<b>SHMP-BL</b>	x	x	x	x	x	x	x	x	x

Tabla 1.3: Resultados globales de los algoritmos del Bloque 1 (más complejos)

<i>Modelo</i>	<i>lin318</i> (42029)			<i>att532</i> (27686)			<i>rat783</i> (8806)		
	<i>Med</i>	<i>Mej</i>	$\sigma$	<i>Med</i>	<i>Mej</i>	$\sigma$	<i>Med</i>	<i>Mej</i>	$\sigma$
<b>SH</b>	x	x	x	x	x	x	x	x	x
<b>SCH</b>	x	x	x	x	x	x	x	x	x
<b>SHMM</b>	x	x	x	x	x	x	x	x	x
<b>SHMP</b>	x	x	x	x	x	x	x	x	x
<b>SH-BL</b>	x	x	x	x	x	x	x	x	x
<b>SCH-BL</b>	x	x	x	x	x	x	x	x	x
<b>SHMM-BL</b>	x	x	x	x	x	x	x	x	x
<b>SHMP-BL</b>	x	x	x	x	x	x	x	x	x

las distintas variantes de los algoritmos de OCH entre sí en términos de: mejor resultado individual obtenido y mejor resultado medio (robustez del algoritmo).

En la documentación entregada se deberá indicar el modelo de ordenador empleado durante la experimentación.

## 6. Estructura de la Documentación a Entregar

La documentación debe ser breve y concisa. Será entregada en papel e incluirá las siguientes secciones:

1. Índice.
2. Breve descripción del problema empleando como **máximo 1 página** (es decir, una cara de un folio).



3. Breve descripción de los algoritmos de OCH considerados (haciendo referencia a las implementaciones):
  - a) El Sistema de Hormigas (**máximo 2 páginas**).
  - b) El Sistema de Colonias de Hormigas (**máximo 2 páginas**).
  - c) El Sistema de Hormigas Max-Min (**máximo 2 páginas**).
  - d) El Sistema de Hormigas Mejor-Peor (**máximo 2 páginas**).
  - e) Uso de la búsqueda local en cada algoritmo de OCH (**máximo 2 páginas**).
4. Experimentos y análisis de resultados:
  - a) Instancias del problema estudiadas.
  - b) Valores de los parámetros considerados en las ejecuciones de los algoritmos.
  - c) Resultados obtenidos (según el formato de las tablas de resultados globales).
  - d) Análisis de resultados más allá de la mera lectura de las tablas obtenidas. Para completar el análisis se podrán añadir representaciones de las soluciones, gráficas de convergencia de la función objetivo de la mejor solución encontrada hasta el momento y/o de otros valores interesantes para observar la evolución del algoritmo, etc.
5. Referencias bibliográficas en caso de haber sido empleadas.

#### NOTAS IMPORTANTES:

- Para facilitar la organización de la documentación, se recomienda comenzar cada descripción o sección en una página nueva.
- Superar el máximo de páginas establecido para cada sección o emplear un formato ilegible podrá influir negativamente (hasta en un 50 %) en la evaluación final de la práctica.

## 7. Componentes de la Práctica

- **Documentación:** La documentación incluirá como mínimo las secciones indicadas en el apartado anterior. Aunque lo esencial es el contenido, también debe cuidarse la presentación y la redacción.

El documento llevará una portada en la que se incluirá el título de la práctica, el curso académico actual, el nombre y DNI del alumno, su correo electrónico, el grupo de prácticas, su horario y el nombre del profesor de prácticas correspondiente.

- **Ficheros:**

Se entregará un disco que contenga una versión ejecutable de los programas y los ficheros de datos/resultados correspondientes a las instancias. La versión ejecutable de los programas junto con los ficheros de datos se incluirán en un subdirectorío del raíz de nombre BIN. En este mismo directorio se adjuntará un pequeño fichero de texto de nombre LEEME que contendrá breves reseñas sobre cada fichero incluido en el directorio, es decir, indicará a qué instancia corresponde cada fichero de datos.

**Es importante que se pueda reproducir la misma experimentación realizada por el alumno usando los ejecutables generados tras compilación directa de los códigos disponibles en la *Web*** (en caso de modificar dichos códigos también se deben incluir los correspondientes ficheros fuente, describiendo los cambios realizados y el por qué en los apartados correspondientes de la documentación entregada).

**La práctica se entregará en dos formatos distintos:**

- **En formato electrónico a través de la web del departamento CCIA.** Se subirá un fichero zip que incluya un pdf con la documentación y los ficheros mencionados.
- **En formato impreso entregado en mano al profesor/a de prácticas.** Se le proporcionará una copia impresa de la documentación y un CD-ROM con los ficheros. Los discos irán etiquetados con la misma información descrita para la portada de la documentación **y deberán estar pegados de algún modo a la documentación, no sueltos.**