

Práctica 4: Aplicación de ACO para el TSP

Convocatoria de junio 2013 (curso académico 2012/2013)

Asignatura: Metaheurísticas
3º Grado Ingeniería Informática (Universidad de Córdoba)

3 de mayo de 2013

Resumen

Esta práctica tiene como objetivo concienciar al alumno de la aplicabilidad del algoritmo de optimización basado en colonias de hormigas (*Ant Colony Optimization*, ACO) al problema del viajante de comercio (*Travelling Salesman Problem*, TSP). Para ello se presentará un *software* que aplica ACO a TSP y se pedirá la comparación de resultados con los obtenidos en la práctica 2 de esta asignatura. La entrega se hará utilizando la tarea en Moodle habilitada al efecto. Se deberá subir un único fichero comprimido todos los entregables indicados en este guión. El día tope para la entrega es el **18 de mayo**. En caso de que dos alumnos entreguen prácticas copiadas, no se puntuarán ninguna de las dos.

1. Introducción

El trabajo a realizar en la práctica consiste en utilizar el algoritmo ACO para obtener los mejores resultados posibles en las distintas instancias del TSP utilizadas durante la práctica 2 de esta asignatura. Para ello, se hará uso del *software* ACOTSP disponible en Moodle y que puede descargarse también en la web <http://www.aco-metaheuristic.org/aco-code/public-software.html>. El alumno deberá manejar este *software* en la búsqueda del mejor resultado posible para todas las instancias. Para esta práctica, por tanto, no será necesario programar el algoritmo (como en prácticas anteriores), sino que el objetivo es que el alumno sea capaz de entender el funcionamiento del *software* ACOTSP y obtener los mejores resultados para las instancias evaluadas del TSP.

El *software* proporciona valores por defecto para todos los parámetros del algoritmo. Aunque existen formas sistemáticas de realizar la tarea de optimización del valor de los parámetros, la más simple es la estrategia de *prueba y error*. Por tanto, el alumno deberá probar distintos valores de los parámetros hasta obtener los mejores para las instancias evaluadas del TSP.

Para utilizar el *software* ACOTSP, descomprime el fichero que hay subido en Moodle sobre la carpeta que quieras. Para compilarlo, utiliza el *Makefile* incluido en el mismo. Para ejecutarlo, lanza el ejecutable `acotsp`. Observa las distintas opciones que ofrece el programa ejecutando la orden `./acotsp --help`.

La sección 2 describe consideraciones que hay que tener en cuenta sobre el formato de los ficheros de datos antes de empezar a utilizar el *software*. La sección 3 describe algunas cuestiones a tener en cuenta durante la optimización de los parámetros del algoritmo. Finalmente, la sección 4 especifica los ficheros a entregar para esta práctica.

2. Consideraciones sobre el formato de las instancias

ACOTSP considera un formato específico para las instancias de TSP, que es el especificado por la TSPLIB¹, un paquete con distintos problemas TSP utilizados para probar la eficiencia de

¹<https://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95/>

nuevos algoritmos de optimización. En Moodle, tenéis disponibles las instancias de los problemas de la segunda práctica en dicho formato². Como se puede observar en los ficheros, aparecen las coordenadas de las distintas ciudades.

Por otro lado, la propia web de TSPLIB ofrece los caminos óptimos para cada un de los problemas TSP. La distancias de los caminos óptimos se encuentra en el fichero `best_values_TSP.txt`.

3. Consideraciones para realizar la búsqueda de los parámetros

Lo primero que debes hacer es consultar el archivo `README`, en concreto las secciones `USAGE` y `OUTPUT`. Las consideraciones a tener en cuenta para realizar la búsqueda de los parámetros son las siguientes:

- Vamos a utilizar el algoritmo `Ant Colony System (ACS)`, que es uno de los más básicos.
- Los parámetros más importantes a tener en cuenta son:
 - `--time`: tiempo máximo que se permite por cada ejecución del algoritmo. Tomaremos siempre un tiempo máximo de **5 segundos**, que supondrá una de las condiciones de parada.
 - `--optimum`: valor óptimo para la instancia TSP, que supone la otra condición de parada (si llegamos a ese valor, dejaremos de realizar ejecuciones). Este valor deberás de extraerlo del fichero de mejores valores.
 - `--tries`: el número de intentos máximo a realizar. Realizaremos un máximo de **3 intentos** por cada problema TSP.
 - `--ants`: número de hormigas a utilizar.
 - `--beta`: exponente que controla la influencia que tiene la información heurística a la hora de elegir la próxima ciudad.
 - `--rho`: factor que controla la evaporación de la feromona.
 - `--q0`: probabilidad de que la siguiente ciudad que se elija sea siempre la mejor posible.
 - `--localsearch`: aplicación de procedimientos de búsqueda local.
 - `--tsplibfile`: fichero con la instancia de TSP a ejecutar (en formato TSPLIB).
- En el fichero `README` puedes ver un ejemplo de ejecución del algoritmo tipo `acs`.

Para realizar las pruebas, se recomienda utilizar el *script* de `bash` que se adjunta en Moodle. Este *script* genera los informes que ya se han utilizado en otras prácticas y que son compatibles con la hoja Excel que también se encuentra en Moodle. De esta forma solo tenéis que modificar los valores de los parámetros en el *script* hasta encontrar los más adecuados. Como siempre, el valor de los parámetros deberá ser constante para todas las instancias o, en todo caso, depender del tamaño de la instancia. Los valores que se recomienda modificar son: `--ants`, `--beta`, `--rho`, `--q0` y `--localsearch`. Se valorará positivamente si el alumno consigue reducir el tiempo computacional necesario para la ejecución (que ahora mismo son 15 segundos por instancia).

Debido a que la condición de parada depende del tiempo de ejecución en segundos, **es necesario que todas las pruebas se hagan en la máquina `ts.uco.es`**.

4. Entregables

Los ficheros a entregar serán los siguientes:

²Seis de los ficheros no han sido tenidos en cuenta para esta práctica, por no estar disponible su especificación en el formato necesario para ACOTSP (en concreto, las instancias `fri26`, `gr17`, `gr96`, `gr202`, `p01` y `dantzig42`).

- Memoria de la práctica en un fichero pdf que describa el algoritmo considerado, las tablas de resultados y los parámetros elegidos.
- Versión final del *script* que se ha considerado.

4.1. Memoria de la práctica

La memoria de la práctica deberá incluir, al menos, el siguiente contenido:

- Portada con el número de práctica, título de la práctica, asignatura, titulación, escuela, universidad, curso académico, nombre, DNI y correo electrónico del alumno.
- Índice del contenido de la memoria con numeración de las páginas.
- Descripción del funcionamiento general del algoritmo ACS (**máximo 2 carillas**).
- Experimentos y análisis de resultados:
 - Descripción de los valores utilizados para los distintos parámetros y del razonamiento que se ha seguido para su elección.
 - Resultados obtenidos según el formato utilizado en otras prácticas.
 - Comparación de los resultados contra los obtenidos en la práctica 2.
 - Análisis de la mejor solución obtenida para **dos** instancias del TSP y comparación con la mejor obtenida en la práctica 2.
- Referencias bibliográficas u otro tipo de material distinto del proporcionado en la asignatura que se haya consultado para realizar la práctica (en caso de haberlo hecho).

Aunque lo importante es el contenido, se valorará también la presentación, incluyendo formato, estilo y estructuración del documento. La presencia de demasiadas faltas ortográficas puede disminuir la nota obtenida.