## Práctica de búsqueda local

5 de Octubre de 2005

El objetivo es utilizar las clases java que se han visto en la clase de laboratorio para resolver el siguiente problema de búsqueda local.

El mayor problema con el que se encuentran las grandes instalaciones científicas es la asignación del uso de sus recursos a los grupos de investigación que quieren utilizarlas.

Un ejemplo es el supercomputador Mare Nostrum. El tiempo de cómputo ha de distribuirse entre las diferentes peticiones que se realizan, teniendo en cuenta el tiempo de máquina que necesitan y el momento en que se quieren obtener los resultados.

La práctica consistirá en resolver el problema de asignación del tiempo de computación de un supercomputador teniendo en cuenta los siguientes parámetros.

Supondremos que queremos asignar el tiempo de ejecución medido en horas de una semana completa (168 horas). Para simplificar, una ejecución tendrá una longitud que se medirá en horas completas (el tiempo de ejecución podrá ir de 1 a 5 horas) y se deberá ejecutar en una hora concreta (de la hora 0 a la 166) con una tolerancia específica para cada trabajo que estará dentro del intervalo [-5,+5] horas. El rango para cada trabajo puede ser diferente y no tiene por que ser simétrico. De esta manera un trabajo puede comenzar en cualquier hora que esté dentro del rango de tolerancia.

Evidentemente un trabajo no puede acabar su ejecución fuera de la hora 167.

Se dispondrá a priori de un número de peticiones a ejecutar. El objetivo es intentar asignar el máximo número de trabajos posible dentro de las horas disponibles.

Para generar los problemas podéis generar trabajos utilizando una distribución uniforme para cada uno de los parámetros (longitud del trabajo, hora de ejecución y tolerancia a la hora de ejecución). Hará falta generar más trabajos que los que caben realmente en el total de horas para poder encontrar soluciones que ocupen el máximo de horas.

- Implementad las clases necesarias para poder resolver el problema usando las clases del AIMA. Tendréis que incluir el código de las clases en el informe.
- Implementad el problema de manera que se puedan generar los problemas aleatoriamente y que se determine un estado inicial del cual partir. Deberéis decidir qué función heurística utilizar y los operadores de cambio de estado.
- Haced un informe con los siguientes experimentos para cada algoritmo (Hill climbing y Simulated annealing), justificando las decisiones que toméis:
  - Probad la efectividad de diferentes estrategias de inicialización de la búsqueda
  - Probad la efectividad de diferentes conjuntos de operadores
  - Probad el efecto que tiene sobre la solución diferentes funciones heurísticas (no vale sólo usar como función heurística el número de trabajos ejecutados)

- Efectividad de la búsqueda dependiendo del número de trabajos disponibles a ejecutar
- En el caso del simulated annealing probad el efecto de los parámetros sobre la bondad de las soluciones
- Comentad los resultados obtenidos respecto a las decisiones que habéis tomado (qué esperabais y qué ha sucedido)
- Comparad los dos algoritmos de búsqueda a partir de los resultados que habéis obtenido con los experimentos respecto a las soluciones obtenidas y el tiempo de ejecución

Se valorarán los experimentos realizados, la calidad del análisis y los comentarios sobre estos experimentos.

La práctica se ha de realizar por parejas.

La entrega de este informe se realizará el día **9 de noviembre**. No lo dejéis todo para el último día.