

Examen Parcial de IA

(3 de abril de 2019)

grupo 10

Duración: 55 min

1. (4 puntos) Los organizadores de la pasarela Gaudí nos piden un sistema inteligente que ayude a planificar los desfiles de la próxima edición. El sistema ha de organizar diez desfiles para diez diseñadores, dos por día en un total de cinco días, sabiendo que algunos diseñadores se niegan a desfilarse el mismo día que algún otro. Se han de asignar modelos a los desfiles (m por desfile) sin que la misma persona desfile más de tres veces en total durante la semana. Cada diseñador tiene también unas restricciones de altura y peso para sus modelos. Los organizadores de la pasarela han de pagar a los modelos y cada uno tiene su tarifa por participar en un desfile. Obviamente, los organizadores quieren pagar lo menos posible. Tras un análisis inicial del problema un compañero de nuestra empresa nos plantea dos estrategias distintas para resolverlo:

- a) Usar el algoritmo de A*. El estado es una asignación parcial de diseñadores y modelos a desfiles. Se definen dos operadores: **añadir_diseñador** a un desfile concreto, que comprueba que el diseñador no se haya añadido antes y que en caso de ser el segundo diseñador asignado del día, no haya incompatibilidad entre ambos diseñadores, el coste de este operador es siempre 1; y **añadir_modelo** a un desfile concreto, que comprueba que el desfile tenga ya diseñador asignado, que no haya más de m modelos en el desfile y que se cumplan las restricciones del diseñador, el coste de este operador es la tarifa del modelo asignado. Como función heurística usamos el número de diseñadores que falta incluir en los desfiles más el número de modelos que falta incluir en los desfiles multiplicado por la tarifa del modelo más caro.
- b) Usar un algoritmo de satisfacción de restricciones. El grafo de restricciones tendría tres variables por cada uno de los modelos que representan los tres posibles desfiles que se le pueden asignar a cada modelo ($desfile_1modelo_x$, $desfile_2modelo_x$, $desfile_3modelo_x$) y una variable por cada uno de los 10 diseñadores ($diseñador_y$). El dominio de cada una de esas variables sería el identificador del desfile a asignar (un número entero entre el 1 y el 10, asumiendo que 1 y 2 son los desfiles del primer día, 3 y 4 son los desfiles del segundo día, y así sucesivamente). Como restricciones binarias habría: desigualdad entre todo par de variables de un mismo modelo x ($desfile_zmodelo_x$ y $desfile_wmodelo_x$) de modo que tengan asignado un desfile diferente; desigualdad entre todo par de variables $diseñador_z$ y $diseñador_w$ (cada diseñador ha de tener asignado un desfile diferente), y para cada modelo x que no cumple las restricciones de altura y peso de un diseñador y , establecer también una restricción binaria de desigualdad entre la variable $diseñador_y$ y las tres variables del modelo x (el modelo x no puede coincidir en ninguno de sus desfiles asignados con el diseñador y).

Comenta cada una de las posibilidades indicando si resuelven o no el problema, qué errores te parece que tiene cada solución y cómo se podrían corregir, y qué ventajas e inconvenientes tienen cada una de ellas. Justifica la respuesta.

2. (6 puntos) Una empresa de transporte marítimo desea decidir la configuración de la carga de su próximo barco. La empresa recibe un conjunto de peticiones de envío de entre las que escoger. Cada petición tiene asociado un precio de transporte.

Cada petición va dentro de un tipo contenedor que está fijado por el tamaño, el peso del envío y las características de la carga. Existen solamente K tipos de contenedor. Los contenedores son propiedad de la empresa y su uso tiene un coste asociado que depende del tipo de contenedor.

Existen diferentes restricciones para la carga: la suma total de pesos de los contenedores no ha de sobrepasar la capacidad de carga del barco P_{max} , ni ha de ser inferior a cierto valor P_{min} . Cada contenedor tiene un peso asociado que depende de la carga que contiene. Las posibilidades de colocar los contenedores en el barco también imponen que haya un mínimo de C_{min} contenedores y un máximo de C_{max} contenedores de cada tipo.

El objetivo es encontrar la combinación de peticiones que maximicen el precio de transporte y minimicen el coste y que esté dentro de las restricciones impuestas.

En los siguientes apartados se proponen diferentes alternativas para algunos de los elementos necesarios para plantear la búsqueda (solución inicial, operadores, función heurística, ...). El objetivo es comentar la solución que se propone respecto a si es correcta, es eficiente, o es mejor o peor respecto a otras alternativas posibles. Justifica tu respuesta.

- a) Se plantea solucionarlo mediante Hill-climbing utilizando como solución inicial el barco vacío y como operadores añadir y quitar contenedores del barco. Queremos usar como función de evaluación de las soluciones lo siguiente:

$$h(n) = \sum_{i=0}^{Ncont} Precio_i \times \sum_{i=0}^{Ncont} Coste_i$$

Donde $Ncont$ es el número de contenedores que hay en una solución, $Precio_i$ es el precio de transporte del contenedor i de la solución y $Coste_i$ es el coste de un contenedor i de la solución.

- b) Se plantea solucionarlo mediante Hill-climbing utilizando como solución inicial escoger al azar C_{min} contenedores de cada tipo. Como operador utilizamos intercambiar un contenedor del barco por otro que no esté en él. Queremos usar como función de evaluación de las soluciones lo siguiente:

$$h(n) = \sum_{i=0}^{Ncont} \frac{Precio_i}{Coste_i}$$

- c) Se plantea utilizar algoritmos genéticos donde la representación de la solución es una tira de bits con tantos bits como peticiones haya, donde cada bit significa si la petición está o no en el barco. Para generar la población inicial usamos la misma técnica que en el apartado anterior. Usamos los operadores habituales de cruce y mutación y como función de evaluación usamos la del apartado anterior pero haciendo que valga cero si la solución incumple las restricciones del problema.