Examen Parcial de IA

(14 de abril de 2008)
grupo 20
Duración: 1 hora

1. (6 puntos) Una de las labores de los equipos de comunicaciones en una red es distribuir el flujo de paquetes que le llegan entre los diferentes equipos con los que están conectados, optimizando la calidad de la distribución de los paquetes.

   Una posible estrategia de distribución de paquetes sería decidir a priori cuantos de los paquetes que se reciben en cierto momento se envían por cada conexión, sin preocuparnos exactamente a donde deben ir.

   Un equipo de comunicaciones recibe cierto número de paquetes (P) por segundo que tiene que distribuir entre sus conexiones de salida. Para cada una de las N conexiones de salida se conocen tres informaciones, su capacidad en número de paquetes por segundo, el tiempo de retraso que introduce la conexión (tiempo medio adicional que añade el nodo de salida al tiempo de llegada a destino de cada paquete) y el número medio de saltos que hará cada paquete hasta llegar a su destino.

   Deseamos calcular el número de paquetes que debemos enviar por cada conexión de salida para optimizar la distribución de paquetes de manera que el retraso y número medio de saltos de los paquetes sea el mínimo posible. Evidentemente se han de enviar todos los paquetes que llegan.

   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 tomando como solución inicial asignar 0 paquetes a cada conexión. Como operador utilizamos aumentar en cierta cantidad el número de paquetes de una conexión sin pasarnos de su capacidad. Usamos como función heurística la suma del producto entre el número de paquetes asignados a cada conexión y el retraso de la conexión.

   Dado que una solución debe enviar todos los paquetes la solución inicial propuesta no es solución.
   *El factor de ramificación del operador será bastante grande (número de conexiones * paquetes que caben en la conexión).*
   La función heurística es incompleta ya que solo tenemos en cuenta el retraso de las conexiones, pero no el número de saltos. Además tal como esta definida la solución inicial y el objetivo del problema, esta solución sería la mejor para esta función heurística (es su valor mínimo) ya que cualquier paquete que asignemos hará aumentar el retraso.

   b) Se plantea solucionarlo mediante Hill-Climbing usando como solución inicial el repartir los paquetes a partes iguales entre todas las conexiones. Como operador utilizamos aumentar o disminuir en cierta cantidad el número de paquetes de una conexión. Como función heurística usamos el sumatorio para todas las conexiones con paquetes asignados del producto entre el retraso de la conexión y el número de saltos.

La solución es válida (es una asignación completa de paquetes a conexiones) y su coste es bajo. Lo que podemos decir sobre la calidad de la solución inicial es que equilibra el retraso de todas las conexiones, no será la mejor solución inicial, pero tampoco será la peor.
Los operadores son correctos ya que nos permiten generar todas las soluciones posibles, pero deberían comprobar las restricciones del problema (no superar los limites de capacidad)
La función heurística es aparentemente correcta ya que al minimizarla llegamos a soluciones mejores, el problema que tiene es que para distintas distribuciones de paquetes dará el mismo valor, por lo que no es capaz de distinguir bien entre soluciones mejores o peores. Seria mas adecuado incluir el número de paquetes por conexión en el heurístico.

c) Se plantea solucionarlo mediante Hill-Climbing tomando como solución inicial el repartir los paquetes entre las conexiones asignando aleatoriamente a una conexión el máximo de su capacidad repitiendo el proceso hasta haber repartido todos los paquetes. Como operador se utilizaría mover cualquier paquete de una conexión a otra siempre que la operación sea válida.

La solución inicial es correcta (es una asignación completa de paquetes a conexiones), su coste es bajo pero no tenemos ninguna garantía sobre su calidad.
*El operador es válido, pero su factor de ramificación es grande (paquetes por conexión * conexión^2). Teniendo en cuenta que todos los paquetes de una conexión son iguales solo tiene sentido hacer la operación una vez por conexión. Al ser un operador de granularidad tan pequeña hará también que la busqueda sea mucho mas larga que con los operadores de las otras opciones*

d) Se plantea solucionarlo mediante algoritmos genéticos donde la representación del problema es una tira de bits compuesta por la concatenación de la representación en binario del número de paquetes asignados a cada conexión (evidentemente usando el mismo número de bits para cada conexión). Como solución inicial ordenamos las conexiones por su retraso y asignamos paquetes en ese orden llenando la capacidad de cada conexión hasta haber asignado todos los paquetes. Usamos los operadores de cruce y mutación habituales.

La codificación permite representar correctamente una solución
La solución inicial es solución, su coste no será demasiado grande asumiendo que el número de conexiones no es demasiado alto. Esta solución intenta optimizar uno de los criterios (el retraso) esto no significa que la solución inicial sea buena respecto al objetivo del problema, no tenemos garantías de estar cerca de la solución aunque probablemente sea mejor que las alternativas de los otros apartados. El problema es que solo podemos generar una solución.
Los operadores de cruce y mutación pueden dar lugar a soluciones invalidas ya que no se comprueban las restricciones del problema.

2. (4 puntos) Tenemos un camión que puede llevar cierta carga máxima y tenemos que recoger y dejar una serie de paquetes en diferentes puntos de una ciudad haciendo el recorrido más corto posible, sin que se sobrepase en ningún momento la carga máxima del camión. Partimos de cierto punto de origen y volvemos a él, habiendo dejado todos los paquetes. Para obtener el recorrido se dispone de un mapa de la ciudad que indica la longitud mínima entre cada par de puntos por los que ha de pasar el camión.

   Puedes resolverlo mediante:

   a) El algoritmo de A*. El estado es el camino recorrido. Utilizamos como coste la longitud del camino actual. La función heurística vale infinito si el camión en el estado actual supera el peso máximo y, en caso contrario, es la suma de las distancias de los puntos por recorrer al origen. El operador aplicable es pasar del punto actual a otro no visitado.

Se plantea la busqueda como un camino, lo que es adecuado para resolverlo mediante A^*.
Utilizar la longitud del camino como coste de la solución también es adecuado.
La función heurística no es admisible ya que la suma de las distancias de los puntos por recorrer al punto de origen sera mas grande que la distancia del camino que queda por recorrer, por lo que no tenemos garantía de obtener el óptimo.
El hacer infinita la distancia cuando se supere el peso máximo es una alternativa correcta a hacer esa comprobación antes de aplicar un operador.
El operador de busqueda también es adecuado, dado lo que se propone para la función heurística.
Con otra función heurística habría que comprobar las restricciones del problema
En este caso para solucionar el problema correctamente se debería encontrar un heurístico admisible. En este problema concreto es difícil encontrar un heurístico admisible que sea una buena estimación del camino.

b) Satisfacción de restricciones, donde las variables son todas las aristas del grafo de conexiones entre los puntos a recorrer, éstas son variables booleanas e indican si pertenecen al camino a recorrer o no. Las restricciones son que debe haber exactamente dos aristas de un mismo vértice en la solución y que no se sobrepase el peso del camión en el recorrido formado por las aristas.

La elección de variables y dominios para representar el problema es aparentemente correcta aunque un poco excesiva (un numero cuadrático de variables)
Cumplir las restricciones nos impondría que las aristas que elijamos formen un camino y que en ningún punto del recorrido se supere el peso del camión. No obstante es imposible cumplir las restricciones explorando las asignaciones mediante ningún algoritmo de satisfacción de restricciones que hemos visto, no podríamos encontrar una asignación valida para la primera variable, es imposible tener dos aristas para un mismo vértice cuando estamos asignando la primera arista.
Además, como el objetivo del problema es minimizar el camino, plantearlo como SR es incorrecto ya que no encontraremos el camino de longitud mínima sino uno cualquiera. Esta circunstancia no se puede corregir, por lo que no podríamos resolver el problema de esta manera.

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.

Examen Parcial de IA

(15 de abril de 2008)
grupo 30
Duración: 1 hora

1. (6 puntos) La empresa "Fish-homes" se dedica a la preparación y venta de acuarios siguiendo ciertos criterios estratégicos. Se dispone de P especies de peces a usar y para cada posible tipo de pez a introducir en el acuario dispone de dos informaciones relevantes: el espacio vital (número de cms. cúbicos necesario para una correcta supervivencia del pez) y el beneficio (euros de ganancia obtenidos por la inclusión de ese pez en un acuario). Dado un modelo específico de acuario que tendrá un volumen determinado, la empresa desea generar la configuración de peces más adecuada teniendo en cuenta las siguientes restricciones: la suma del espacio vital de todos los peces incluidos no debe ser inferior a 1/3 del volumen del acuario, ni superior a los 2/3, debe haber un mínimo de 6 especies representadas y el beneficio debe ser el máximo posible.

   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) Usar Hill-climbing. Como estado inicial consideramos el acuario vacío. El operador disponible es `aumentar-pez (tipo, cantidad)`. Como función heurística usamos la suma de beneficio de cada uno de los peces incluido en el acuario.

   La solución inicial no es valida ya que incumple las restricciones para ser solución (suma del espacio vital de los peces no inferior a 1/3 del volumen). No obstante con el operador propuesto podríamos llegar al espacio de soluciones durante la busqueda, pero no esta garantizado.
   *El operador no comprueba las restricciones del problema (no pasarse de 2/3 del volumen), su factor de ramificación es bastante alto (numero de tipos de peces * peces que caben de un tipo).*
   La función heurística es correcta ya que es precisamente eso lo que queremos optimizar

   b) Usar Hill-climbing. Como estado inicial consideramos el acuario con una asignación aleatoria de 6 especies de peces distintas, escogiendo la misma cantidad de peces para cada especie que llenen el acuario hasta 1/3 de su volumen. Disponemos del operador `aumentar-pez (tipo, cantidad)` que comprueba el posible exceso de ocupación en el acuario y del operador `disminuir-pez (tipo, cantidad)` que comprueba la posible infra-ocupación del acuario. Como heurística usamos el producto para cada especie dentro del acuario del número de unidades de la especie, el volumen que necesita cada especie y su beneficio.

   La solución inicial es correcta (cumplen las restricciones), hacer la selección de manera aleatoria no nos garantiza nada sobre la calidad de la solución inicial.
   Los operadores nos permiten explorar todo el espacio de soluciones aunque tienen un factor de ramificación grande. El operador disminuir también tendría que comprobar que queden por lo menos seis tipos de peces en la pecera. Hay que notar que el operador disminuir peces no se podrá aplicar porque siempre disminuirá la función heurística, por lo que estos sucesores nunca se elegiran.
   El volumen que ocupa el pez en la pecera no es un criterio que tengamos que optimizar, por lo que no debería formar parte de la función heurística. Además si multiplicamos por el volumen estaremos dando mas importancia los peces mas grandes y distorsionando el beneficio que obtenemos por tipo de pez.

   c) Usar Hill-climbing. Como solución inicial ordenamos las especies de peces por beneficio y vamos añadiendo un pez de cada tipo en ese orden hasta llenar el acuario a 2/3 de su volumen. Como operador utilizamos cambiar un pez de una especie a otra. Como función heurística usamos h'(n) = (2/3 volumen total acuario - suma espacio vital de peces incluidos) + suma de beneficio de cada uno de los peces incluido en el acuario.

La solución inicial cumple las restricciones. El coste de hallarla no es muy grande ya que podemos suponer que el numero de especies disponibles no sera muy grande. Esta manera de repartir los peces no nos garantizara demasiado respecto a la bondad de la solución.
El operador seria correcto si comprobara las restricciones del problema, su factor de ramificación es pequeño (considerando que solo apliquemos un cambio por combinación de especies), pero su granularidad es pequeña, por lo que el camino hasta la solución sera largo
En la función heurística seguimos incluyendo el volumen y no es un criterio que tengamos que optimizar. Podríamos suponer que nos evitaría elegir soluciones en las que nos pasaramos del volumen máximo de peces, pero la función no nos lo garantiza ya puede haber no soluciones que den un valor mejor que soluciones para este heurístico.

d) Usar algoritmos genéticos. Un individuo es una tira de $\sum_{i=1}^{K} L_i$ bits, siendo K el número de especies distintas disponibles y L_i el número de peces de cada especie necesarios para ocupar los 2/3 del acuario. Por tanto, para cada especie i, el valor L_i puede ser distinto. Como población inicial se generan aleatoriamente n individuos donde en cada uno hay exactamente 6 bits a 1. Como operadores se usan los habituales de cruce y mutación.

La codificación de la solución es adecuada para representarla (numero de peces de cada especie en la pecera).
La solución inicial es incorrecta ya que no se asegura que se ocupe 1/3 del volumen.
Los operadores de cruce y mutación no comprueban las restricciones del problema por lo que no nos serviran

2. (4 puntos) Para hacer presión ante TMB, los sindicatos de conductores de autobuses en Barcelona quieren encontrar la forma de cumplir los servicios mínimos que se les imponen sobre el número de autobuses que han de circular por línea y hora, pero afectando al mayor número de usuarios, para que sus reclamaciones les den a ellos mayor poder en las negociaciones. Para ello han construido una tabla que, para cada hora y línea de autobús, les dice el número de usuarios que viajan a esa hora por esa línea, y el número mínimo de autobuses que según TMB han de pasar durante esa hora por esa línea para cumplir los servicios mínimos. Sabemos que cada autobús puede llevar hasta p pasajeros. Hay además otra regla que han de cumplir, y es que TMB también impone un número mínimo M total de autobuses que han de circular al día, siendo este número algo mayor que la suma de los autobuses de los servicios mínimos antes mencionados.

Se nos plantean las siguientes alternativas:

a) Queremos utilizar satisfacción de restricciones donde tenemos una variable por cada línea de autobuses y cada hora, y los valores son el número de autobuses asignados a cada línea y cada hora. Las restricciones son el número mínimo de autobuses para cada línea y hora, el número mínimo de autobuses que han de circular durante el día y que el número total de usuarios que se queden sin autobús sea mayor que un cierto valor U.

La representación del problema mediante variables y dominios es correcta. Dar una asignación de autobuses a producto cartesiano de lineas y horas es una solución.
Las restricciones que se plantean son correctas y son las que debería cumplir una solución (mínimo de autobuses por linea y hora y mínimo de autobuses total).
Dado que el problema nos pide maximizar el numero de usuarios sin servicio esta aproximación no seria correcta ya que SR no optimiza, lo único que obtendríamos seria la primera solución que dejara mas de U usuarios sin servicio.

b) Queremos utilizar A*. El estado sería la asignación de autobuses a horas y líneas, partiendo de la asignación vacía. Como operador utilizaríamos asignar un autobús a una línea y hora, las condiciones de aplicabilidad del operador serían no superar el número M de autobuses en la solución. El coste del operador es p. La función heurística sería la siguiente:

$$\begin{aligned} h'(n) &= (M - \sum_{\forall hora,linea} autobuses_{hora,linea}) \times p \\ &\quad + (\sum_{\forall hora,linea} serv\_min_{hora,linea} - autobuses_{hora,linea}) \end{aligned}$$

Donde $autobuses_{hora,linea}$ es el número de autobuses asignados a una línea a cierta hora y $serv\_min_{hora,linea}$ es el número de autobuses que exigen los servicios mínimos.

El planteamiento del estado es correcto, tendremos un estado inicial que sera la asignación vacía y pasaremos por estados con diversas asignaciones hasta llegar a la asignación que tenga M autobuses.
El factor de ramificación es muy grande (todas las posibles asignaciones de autobuses a lineas y horas). Además, garantizar no pasarnos de M autobuses no nos garantiza el cumplir el resto de restricciones del problema (servicios mínimos por linea) al completar la asignación.
La función heurística no esta relacionada con el objetivo, que seria dejar el mayor numero de pasajeros sin servicio. De hecho el coste de la solución sera siempre el mismo (el numero de pasajeros que somos capaces de servir con M autobuses). El segundo sumando además nos hará la función no admisible ya que el coste de la solución (que es fijo) se vera incrementado con lo que nos falte para llegar los servicios mínimos.
Para arreglar el planteamiento el coste del operador no debería ser p, sino el numero de pasajeros que acabaremos transportando en el autobus (esperamos no llenarlo del todo). Además habría que garantizar cumplir las restricciones de servicios mínimos de manera diferente que añadiéndolo a la función heurística y habría que preservar la admisibilidad del heurístico estimando de manera optimista (por lo bajo) el numero de pasajeros que realmente serviremos con los autobuses que nos quedan por asignar.

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.

Examen Parcial de IA

(16 de abril de 2008)
grupo 10
Duración: 1 hora

1. (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 va dentro de un tipo contenedor que está fijado por el tamaño y peso del envío. 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. Cada petición tiene asociada un precio de transporte.

   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}. 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. Cada contenedor tiene un peso asociado que depende de la carga que contiene.

   El objetivo es encontrar la combinación de peticiones que den el máximo precio de transporte y tengan el coste mas pequeño 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:

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

   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.

   La solución vacía no cumple las restricciones del problema (un numero mínimo de contenedores de cada tipo) por lo que no es válida.
   Los operadores serian correctos si comprobaran las restricciones del problema. Habría que tener cuidado si alguno de los operadores empeora siempre la solución ya que no serviría para nada.
   Respecto a la función heurística deberiamos considerar si la minimizamos o la maximizamos.
   Si la minimizamos, el precio del transporte ira en el sentido contrario al que pretendemos ya que cuanto mas pequeño sea peor es la solución para nosotros. Si maximizamos el problema lo encontraremos con el coste.
   Además se pueden obtener soluciones iguales con un precio grande y un coste pequeño o al revés, por lo que no distingue correctamente entre soluciones buenas y malas. De hecho esta función busca un equilibrio entre estas dos cantidades.

   b) Se plantea solucionarlo mediante Hill-climbing utilizando como solución inicial llenar el barco de contenedores de un solo tipo hasta llegar al peso mínimo y como operadores añadir un contenedor sin pasar del peso máximo y quitar un contenedor del barco sin bajar del peso mínimo. Queremos usar como función de evaluación de las soluciones lo siguiente:

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

La solución tampoco cumple las restricciones de la solución, debe haber un mínimo numero de contenedores de cada tipo.
Los operadores comprueban las restricciones del peso del barco, pero no el numero máximo y mínimo de contenedores.
La función la deberiamos maximizar ya que cuanto mejor es la solución mayor es el cociente, de todas maneras estamos maximizando el ratio entre coste y precio y podemos obtener el mismo ratio de maneras diferentes de manera que no obtendremos la solución con máximo precio posible, sino la solución en la que obtengamos el máximo ratio precio respecto coste.
Además el operador quitar hace que disminuya siempre la función así que nunca se elegira ese operador si maximizamos.

c) 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:

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

La solución inicial no nos garantiza que cumplamos las restricciones de peso que impone la solución.
El operador no es correcto, no comprueba las restricciones del problema y además no permite modificar el numero de contenedores de la solución, por lo que no permitiría explorar todo el espacio de busqueda.
Esta función heurística debería maximizarse, optimiza precisamente el objetivo que pide el problema, por lo que es correcta.

d) 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.

La representación de la solución es correcta, representa la asignación de peticiones al barco.
La forma de generar soluciones iniciales del apartado anterior no genera soluciones validas, por lo que no nos serviría.
Los operadores de cruce y mutación no nos garantizan que se cumplan las restricciones del problema.
Hacer que la función heurística del apartado anterior valga cero si no se cumplen las condiciones del problema podría paliar el defecto de los operadores de cruce y mutación, pero no nos aseguran de que en algún momento la población se llene de soluciones no validad y que la busqueda o no converja o sea demasiado lenta.

2. (4 puntos) Los dueños de cierto hipódromo desean configurar las lista de participantes de cada carrera de una jornada. Cada día se realizan 10 carreras con 10 caballos cada una. Para confeccionar las carreras pueden usar caballos de 8 caballerizas profesionales que ponen 7 caballos cada una a su

disposición. Un caballo puede correr en varias carreras siempre que tenga como mínimo tres carreras de descanso. Tampoco pueden correr más de dos caballos de la misma caballeriza en una carrera.

Que un caballo corra en una competición tiene un coste para el hipódromo que depende del ranking del caballo. Buscamos minimizar el coste total de las carreras respetando las restricciones. Se plantean dos estrategias distintas para resolver el problema:

a) Usar A* tomando como estado la asignación de caballos a carreras. El estado inicial sería la asignación vacía, el estado final sería la asignación completa. Usaríamos como operador de cambio de estado el asignar un caballo a una carrera, su coste sería el coste del caballo asignado. Como función heurística usaríamos el número de caballos que faltan por asignar.

   Usar la asignación de caballos a carreras es una representación válida del estado del problema, con estado inicial la asignación vacía y estado final la asignación completa.
   El operador debería comprobar las restricciones del problema para obtener soluciones parciales validas, aunque también se podría hacer que la función heurística valiera infinito para soluciones no validas. Su coste es correcto ya que el coste de los caballos es lo que pretendemos minimizar.
   El factor de ramificación es muy grande ya que a cada paso deberemos contemplar cualquier asignación posible de caballos a puestos disponibles.
   La función heurística es admisible dependiendo de las unidades en que se cuente el coste de los caballos. En el caso de que sea admisible es poco informativa ya que es una estimación demasiado optimista del coste de lo que queda de solución y no tiene ninguna relación con el objetivo. De hecho el algoritmo seguramente degenerara a una busqueda en anchura guiada por el coste de la solución parcial.
   Para arreglar el problema deberiamos estimar mejor la función heurística, aunque en este caso es difícil estimar de manera informativa el coste de lo que falta en la solución, e imponer alguna manera para restringir el factor de ramificación como por ejemplo ordenar el modo en el que hacemos las asignaciones de cada carrera y solo permitir las asignaciones en ese orden.

b) Usar un algoritmo de satisfacción de restricciones. Supondríamos que un caballo solo puede estar en un máximo de tres carreras. Crearíamos un grafo de restricciones cuyas variables serían las tres posibles carreras de cada caballo. El dominio de cada variable sería la carrera en la que corre el caballo o vacío si el caballo no corre. Tendríamos una restricción entre las carreras consecutivas de un caballo que no permitiera que su diferencia fuera menor que tres y una restricción entre todas las carreras de los caballos de una caballeriza para que no pudiera haber más de dos caballos en la misma carrera.

   La forma de representar el problema mediante las variables y dominios es correcta, decidir en que tres carreras como máximo corre un caballo.
   Las restricciones entre carreras de un mismo caballo y caballos de una misma caballeriza son correctas, pero debería haber una restricción adicional que no permitiera que hubiera mas de 100 asignaciones en total, no mas de 10 caballos por carrera y que al final haya 100 asignaciones exactamente.
   El handicap de este planteamiento es que deseamos optimizar el coste de los caballos y mediante un algoritmo de SR es imposible.

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.