Solución Tarea 2

IIC2613 – Inteligencia Artificial

16 de junio de 2021

Primer Semestre 2021 - Profesores J. Baier - A. Villa

Parte 1

Pregunta 1

Tenemos el caso de que n_1 y n_2 son tales que $f(n_1) = f(n_2)$, donde además son los nodos con mayor prioridad dentro de la *open*.

Como señala el enunciado, al presentarse este empate deberíamos elegir aquel nodo con mayor valor de g o, equivalentemente, menor valor de h.

Intuitivamente es lógica esta opción, ya que es razonable pensar que el nodo que debemos sacar de la *open* sea aquel que esté más próximo al objetivo. Dicho esto, sabemos que con una heurística **admisible y consistente**, se esperaría que mientras más cerca estoy del objetivo entonces menor es mi heurística h (recordemos que $h(s^*) = 0$ donde s^* es el objetivo), y por consecuencia mayor el costo del camino g. Por lo tanto, en caso de empate entre dos nodos, aquel cuyo valor de h sea menor y cuyo valor de g sea mayor, entonces seguramente se encontrará más cerca del objetivo y por eso debería tener una prioridad mayor en la *Open*.

Aspectos a considerar

- Se explica correctamente porque la regla tiene sentido (a mayor valor de g, o menor de h, más cerca estoy del objetivo).

Pregunta 2

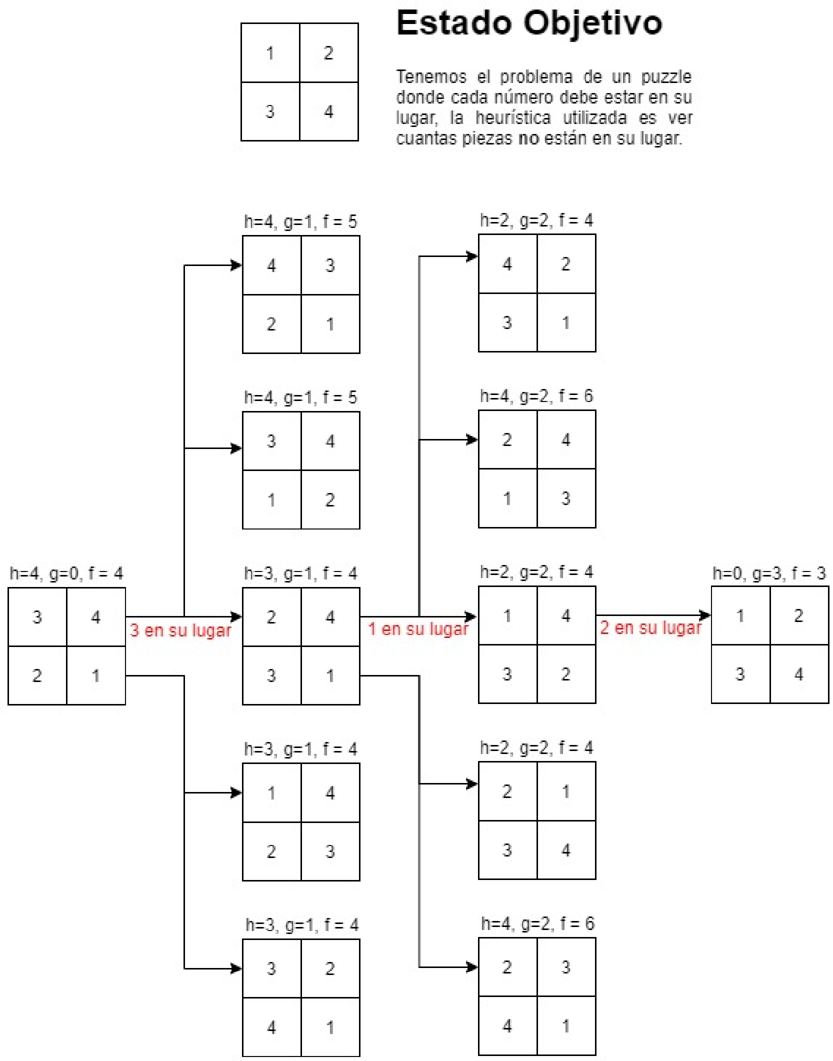
Con la regla de desempate Weighted A* sigue siendo w-suboptimo debido a que el valor de f de los nodos dentro de la OPEN no cambia. En particular, el hecho de que se establezca una regla de desempate sobre los valores de g o h no rompe con la demostración del vídeo, ya que de todas formas para cualquier nodo s extraído de la open se seguirá cumpliendo que $f(s) \leq f(s^*)$ debido a que el nodo extraído sigue correspondiendo al menor valor de f, el cual "en el peor de los casos" fue elegido por la regla de desempate.

Aspectos a considerar

- Explicar que Weighted A* sigue siendo w-suboptimo porque el valor de f no cambia
- Se indica explícitamente porque la demostración del video se sigue cumpliendo.

Pregunta 3

A continuación se muestra un ejemplo de un problema exponencialmente más ineficiente cuando se ocupa la regla de empates de manera contraria. [1]:



Este problema consiste en un puzzle de $2x2$, donde en una acción yo puedo intercambiar una pieza

[1] Imagen muestra funcionamiento de A^* con la correcta regla de empates.

con la otra. Por lo tanto, el objetivo del problema es que a partir de un estado dado, debo llegar al estado final donde todas las piezas están ordenadas. Hay que señalar también que esto cumple con la condición del enunciado, donde se pide buscar un problema que al desordenar las acciones de la solución obtengo otra solución, por ejemplo en la solución mostrada en la imagen (coloco la pieza 3 en su lugar $\rightarrow$ coloco 1 en su lugar $\rightarrow$ coloco 2 en su lugar), yo podría realizar estas acciones en cualquier orden y seguiría obteniendo una solución.

Justificación: La heurística h utilizada para este ejemplo se define como la cantidad de piezas que no están en su lugar (ver imagen), además el valor de g se incrementa con cada acción realizada, siendo f la suma entre h y g. Con esto, podemos notar que en el problema descrito en la imagen, donde hay un correcto funcionamiento de la regla de empates, la eficiencia esta dada por la cantidad de nodos expandidos que es, en el peor de los casos, $B \cdot P$, donde B es el factor de ramificación promedio y P es la profundidad de la solución (Esto se puede ver claramente en la imagen).

Por otro lado, si quebramos la regla de empates y le ponemos prioridad a aquellos nodos cuya heurística h es mayor (o su g es menor), notamos que al expandir un nodo, el valor de f en el mejor de los nodos hijo puede ser igual a los del padre e igual a los de los hermanos del nodo padre, lo que me obligará a expandir primero a **todos** los nodos hermanos cuyo valor de f sea igual al del mejor hijo, debido a que el valor de g de los nodos hermanos es menor al de los hijos. Con esto, sabiendo que la solución está a una profundidad P, nos dice que en el peor de los casos deberé expandir B^P nodos antes de encontrar la solución. Por lo tanto, podemos notar que la implementación quebrando la regla de empates es *exponencialmente más ineficiente* que la que no la quiebra (B^P vs $B \cdot P$).

Aspectos a considerar

- Se muestra un problema concreto que es exponencialmente más ineficiente al ocupar la regla de empates de manera contraria (de preferencia es un problema que al desordenar las acciones de la solución, se obtiene otra solución). El problema señalado arriba sólo es una referencia, no es necesario que sea igual o similar a ese.
- La justificación de porque el problema es exponencialmente más ineficiente debe ser correcta.

Pregunta 4

Se reporta en un gráfico o tabla el rendimiento (medido en expansiones) de la implementación sin regla de empates vs la implementación con regla de empates.

```
#prob      #exp      #gen      |sol|     tiempo maxsubopt
    1    151817    290844         45      18.86      0.00
    2    170564    320461         42      21.83      0.00
    3    198327    384013         42      38.36      0.00
    4    191259    362370         41      33.22      0.00
    5    620168   1162970         46      86.31      0.00
    6    440524    829358         47      56.62      0.00
    7    410133    786993         44      50.70      0.00
    8    148509    287219         53      17.78      0.00
    9    301944    575488         46      62.50      0.00
   10    614465   1177611         46     139.34      0.00
Tiempo total:          525.53
Expansiones totales: 3247710
Costo total:           452
```

Figura 1: Sin regla de empates

```
#prob      #exp      #gen      |sol|     tiempo maxsubopt
    1     32470     62702         45       4.42      0.00
    2     48443     91053         42       7.41      0.00
    3     66296    129512         42      10.41      0.00
    4    142928    272445         41      21.53      0.00
    5    154019    291486         46      24.25      0.00
    6    179269    335095         47      32.91      0.00
    7    191088    363345         44      29.61      0.00
    8    273541    516079         53      43.67      0.00
    9    330838    629125         46      48.52      0.00
   10    486106    918183         46      66.95      0.00
Tiempo total:          289.67
Expansiones totales: 1904998
Costo total:           452
```

Figura 2: Con regla de empates

Parte 2

Pregunta 1

Pregunta a)

Para que una heurística sea consistente debe cumplir con las siguientes dos condiciones:

1. $h(s) = 0, \forall s \in G$
2. $h(s) \leq c(s, s') + h(s'), \forall s$

- Para 1: Si nos encontramos en un estado s, donde s es el estado *goal*, entonces $h(s) = \sum^{15} (x_{goal} - x_{actual}) + (y_{goal} - y_{actual})$ en donde claramente el resultado de la sumatoria es 0 ya que para cada coordenada el valor de *goal* será igual al *actual*.
- Para 2: Para este caso debemos notar que hay dos posibilidades:
  - Estado s' tiene una heurística mayor a s (nos alejamos del objetivo): Al aumentar la heurística y al saber que el costo es siempre mayor o igual a 1, es trivial señalar que se cumple la condición.
  - Estado s' tiene una heurística menor a s (nos acercamos al objetivo): Recordemos que el nodo s es el padre del nodo s', por lo tanto al movernos del nodo s al s' estamos incurriendo en un costo de 1. Luego, el hecho de que s' tenga una menor heurística nos indica que a lo más una pieza del puzzle de 15 se acercó al objetivo a lo más en una posición, por lo que la heurística h(s') es a lo sumo menor en una unidad a h(s), donde en ese caso $h(s) = c(s, s') + h(s)$.

Aspectos a considerar

- Se deben demostrar ambas condiciones de consistencia.
- Demostración no necesariamente tiene que ser igual a la de arriba pero tiene que seguir una lógica correcta y no puede caer en contradicciones.
- No pueden partir asumiendo lo que tienen que demostrar.

Pregunta b)

Tomando la definición de consistencia tenemos lo siguiente:

$$h(s) \leq c(s, s') + h(s')$$
$$h(s) + g(s) \leq c(s, s') + g(s) + h(s')$$
$$h(s) + g(s) \leq g(s') + h(s')$$
$$f(s) \leq f(s')$$

Esto nos señala que los valores de f de los nodos en la open son monótonamente no decrecientes. Esto implica que si ya expandí un nodo s, si este vuelve a aparecer en la open necesariamente es por un camino con un costo mayor, por lo que no será considerado. De esto se concluye que al expandir un nodo ya se ha encontrado el mejor camino hasta él.

Aspectos a considerar

- Se debe mostrar que los valores de f de los nodos en la open son monótonamente no decrecientes (no es necesario que lo digan explicito).
- Explican la razón de porque esto implica que un nodo puede ser expandido a lo más una vez con una heurística consistente.

Pregunta 2

Se deben reportar los resultados (light_manhattan vs manhattan_div_15) en número de expansiones según los pesos especificados en el enunciado.

Pregunta 3

Se debe indicar un link al video.

Aspectos a considerar

- Explicación de la heurística debe ser correcta y no contener contradicciones ni afirmaciones incorrectas.
- Debe cumplir con los objetivos especificados en el enunciado.

Bonus: Pregunta 5

Se debe contar con una explicación de la implementación de la heurística *linear conflicts*.

En lineas generales la explicación debería señalar los aspectos particulares de la implementación. Esta debe corresponder efectivamente a la heurística de *linear conflicts*.

Parte 3

Pregunta 1

Se debe realizar una combinación entre **num_pref y out_of**, para determinar la mejor configuración en términos de expansiones totales.

Aspectos a considerar

- Se tienen al menos 6 combinaciones de valores y a lo más 8.
- Reporte es razonablemente criterioso con los limites de tiempo para ejecuciones largas, granularidades y valores elegidos.
- Se señala cual es la mejor configuración.
- Se argumenta si es suficiente con los experimentos que se realizaron para llegar a la mejor configuración. En general en este punto uno esperaría que no debido a la complejidad del problema y la falta de configuraciones por comprobar, pero de igual forma se podría considerar correcto en caso contrario si dan una explicación lógica según los resultados obtenidos.

Pregunta 3

En esta pregunta se debería incluir una explicación similar a la demostración de suboptimalidad para *astar* con la particularidad que ahora los nodos no son sacados directamente de la open sino que entre la unión entre open y preferred.

La demostración se basa en el lema de A* que dice que siempre hay un estado s* en la open que está en un camino óptimo y es tal que $f(s*) \leq c*$. Ahora bien, en este caso no estamos seguros si ese estado s* está en open o preferred, pero si sabemos que está en la unión entre open y preferred, por lo que se debe discutir que el lema se sigue cumpliendo en esta forma modificada. Una forma de mostrarlo es la siguiente:

Partiendo de la demostración de A*:

$$f(s) \leq c*$$

Siendo s un estado que puedo haber estado en open o preferred tenemos.

$$\frac{c(\pi)}{c^*} \leq \frac{c(\pi)}{f(s)}$$

Ahora según el lema de A* sabemos que $f(s*) \leq c*$, siendo s* un estado en un camino óptimo que esta en la union entre open y preferred:

$$\frac{c(\pi)}{c^*} \leq \frac{c(\pi)}{min_{s \in Open \cup Preferred} g(s) + h(s)}$$

Aspectos a considerar

- Se debe incluir la explicación de suboptimalidad y de la cota entregada en el código.

Pregunta 5

Para esta pregunta se debería incorporar una explicación de la modificación propuesta en el punto 4 y porqué esta lleva a que el algoritmo satisfaga la cota de suboptimalidad.

Básicamente se debe mostrar que en todo momento de la ejecución del algoritmo la condición $g(t) + h(t) > w \cdot m$ evita que se expandan nodos que infrinjan la cota de suboptimalidad, de este modo la solución retornada (si es que existe) en todo momento satisface la cota ya que esta corresponde a los nodos que fueron expandidos.

Otra forma de verlo es que sólo se expanden aquellos nodos que cumplen con $g(t) + h(t) \leq w \cdot m$ lo que es equivalente a $\frac{g(t)+h(t)}{m} \leq w$, lo que precisamente representa la cota de suboptimalidad.

Pregunta 6

Se especifica la comparación entre el algoritmo Weighted A* y Preferred A* para distintos valores para la cota de suboptimalidad y peso.

Bonus: Pregunta 7