# Respuestas Escritas Tarea 2 IA

Tomás Couso

Mayo 2022

# 1 Demostración admisibilidad heurísticas cubo rubik

Una heurística $h$ se dice admisible si para todo estado $s$ se cumple la siguiente desigualdad con respecto a un estado objetivo $s_g$:

$$h(s) \leq \delta(s, s_g)$$

donde $\delta(s, s_g)$ es el costo de un camino óptimo entre los estados $s$ y $s_g$.

- Max Manhattan Corners ($h_1$) y Min Manhattan Corners ($h_2$):

  Demostremos admisibilidad por casos

  <u>Escenario 1</u>

  El primer escenario posible es tener 1 o más esquinas en posiciones incorrectas y a distancia manhattan de 6 de su posición objetivo. Notamos que en dicha situación, el escenario de menos movimientos para llegar a un estado objetivo sería que todas las esquinas quedaran en su posición correcta al ejecutar 3 movimientos, de manera tal que $\delta(s, s_g) = 3$. En caso de que todas las esquinas se encontraran a distancia 6, se tendría el siguiente escenario:

$$h_1(s) = \frac{max\{6, 6, 6, 6, 6, 6, 6, 6\}}{2} = 3 \leq \delta(s, s_g)$$

$$h_2(s) = \frac{min\{6, 6, 6, 6, 6, 6, 6, 6\}}{2} = 3 \leq \delta(s, s_g)$$

  Notamos que si alguna de las piezas está a una distancia menor que 6, $h_1$ mantiene su valor, mientras que $h_2$ lo disminuye, de modo que las desigualdades siguen siendo válidas. Notamos también que si son necesarios más de 3 movimientos para llegar a un estado objetivo, las desigualdades también siguen siendo válidas.

  <u>Escenario 2</u>

  El segundo escenario posible es tener 1 o más esquinas en posiciones incorrectas y a distancia manhattan de no más de 4 de su posición objetivo. Notamos que en dicha situación, el escenario de menos movimientos para llegar a un estado objetivo sería que todas las esquinas quedaran en su posición correcta al ejecutar 2 movimientos, de manera tal que $\delta(s, s_g) = 2$. En caso de que todas las esquinas se encontraran a distancia 4, se tendría el siguiente escenario:

$$h_1(s) = \frac{max\{4, 4, 4, 4, 4, 4, 4, 4\}}{2} = 2 \leq \delta(s, s_g)$$

$$h_2(s) = \frac{min\{4, 4, 4, 4, 4, 4, 4, 4\}}{2} = 2 \leq \delta(s, s_g)$$

Notamos que si alguna de las piezas está a una distancia menor que 4, $h_1$ mantiene su valor, mientras que $h_2$ lo disminuye, de modo que las desigualdades siguen siendo válidas. Notamos también que si son necesarios más de 2 movimientos para llegar a un estado objetivo, las desigualdades también siguen siendo válidas.

Escenario 3

El tercer escenario posible es tener 1 o más esquinas en posiciones incorrectas y a distancia manhattan de no más de 2 de su posición objetivo. Notamos que en dicha situación, el escenario de menos movimientos para llegar a un estado objetivo sería que todas las esquinas quedaran en su posición correcta al ejecutar 1 movimiento, de manera tal que $\delta(s, s_g) = 1$. En caso de que todas las esquinas se encontraran a distancia 4, se tendría el siguiente escenario:

$$h_1(s) = \frac{max\{2,2,2,2,2,2,2,2\}}{2} = 1 \leq \delta(s, s_g)$$

$$h_2(s) = \frac{min\{2,2,2,2,2,2,2,2\}}{2} = 1 \leq \delta(s, s_g)$$

Notamos que si alguna de las piezas está a una distancia menor que 2, $h_1$ mantiene su valor, mientras que $h_2$ lo disminuye, de modo que las desigualdades siguen siendo válidas. Notamos también que si son necesarios más de 1 movimiento para llegar a un estado objetivo, las desigualdades también siguen siendo válidas.

Escenario 4

El cuarto escenario es que todas las esquinas se encuentren en su posición objetivo. En dicho caso, todas se encuentran a distancia manhattan 0 de su objetivo, y son necesarios 0 movimientos o más para llegar al estado objetivo, de modo que se cumple:

$$h_1(s) = \leq \delta(s, s_g)$$

$$h_2(s) = \leq \delta(s, s_g)$$

Los cuatro escenarios anteriores clasifican todos los posibles escenarios que pueden tenerse en un cubo rubik. Como para todos estos se cumple la desigualdades

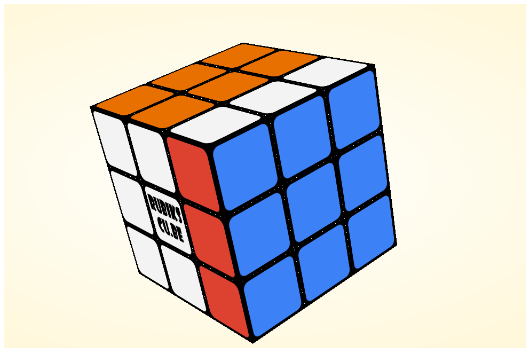
$$h_1(s) = \leq \delta(s, s_g)$$

$$h_2(s) = \leq \delta(s, s_g)$$

se concluye que $h_1$ y $h_2$ son heurísticas admisibles.

- Sum Manhattan Corners ($h_3$):

  Considérese la siguiente configuración $s$ del cubo rubik, obtenida a partir de ejecutar un movimiento F a un cubo resuelto $s_g$:



Podemos notar que:

$$\delta(s, s_g) = 1$$

pues con el movimiento F anticlokwise podemos resolver el cubo con el mínimo de movimientos posibles.

Notamos que tenemos cuatro esquinas a una distancia manhattan con su posición objetivo de dos unidades cada una. Si calculamos el valor de la heurística $h_3$, se tiene el siguiente valor:

$$h_3(s) = 2 + 2 + 2 + 2 = 8$$

Podemos constatar en este caso que se cumple que:

$$h_3(s) > \delta(s, s_g)$$

lo que constituye un contraejemplo para la admisibilidad de $h_3$. Por lo tanto, se concluye que $h_3$ es una heurística no admisible.

## 2 Demostración admisibilidad en caso de consistencia

Sea $h$ heurística consistente, $S$ conjunto de estados de un problema de búsqueda, $s_g$ estado objetivo del mismo problema de búsqueda, $c(s, s')$ el costo de transitar desde un estado $s \in S$ a otro estado $s' \in S$, $\delta(s, s')$ el costo de un camino óptimo entre los estados $s \in S$ y $s' \in S$ y $Succ(s)$ el conjunto de estados sucesores de $s \in S$.

Por demostrar

$$h(s) \leq \delta(s, s_g) \quad \forall s \in S$$

Se asumirá para efectos de esta demostración que todos los estados posibles de un problema de búsqueda tienen algún camino hacia el estado objetivo, de modo que para cada estado $s_n$ existe una sucesión numerable desde el estado objetivo $s_g$ hasta $s_n$:

$$s_g, s_1, s_2, ..., s_n$$

La anterior estructura permite una demostración por inducción que, en base a lo recién asumido, permite concluir admisibilidad para todo estado posible del problema de búsqueda.

Como caso base, considérese el estado $s_1 \in \{s : s_g \in Succ(s)\}$. Por consistencia de $h$:

$$h(s_1) \leq c(s_1, s_g) + h(s_g)$$

Como $h$ es consistente, sabemos que $h(s_g) = 0$. Al mismo tiempo, sabemos que $c(s_1, s_g) = \delta(s_1, s_g)$. Luego, se tiene:

$$h(s_1) \leq \delta(s_1, s_g)$$

lo que demuestra admisibilidad para el caso base.

Considérese ahora la siguiente hipótesis inductiva concerniente al estado $s_{n+1} \in \{s : s_n \in Succ(s)\}$:

$$h(s_n) \leq \delta(s_, s_g) \rightarrow h(s_{n+1}) \leq \delta(s_{n+1}, s_g)$$

Asumamos $s_n$ que cumple con el antecedente de la hipótesis de inducción y $s_{n+1} \in \{s : s_n \in Succ(s)\}$:

$$h(s_n) \leq \delta(s_, s_g)$$

$$h(s_n) + c(s_{n+1}, s_n) \leq \delta(s_, s_g) + c(s_{n+1}, s_n)$$

Como h es consistente:

$$h(s_{n+1}) \leq c(s_{s+1}, s_n) + h(s_n)$$

Luego:

$$h(s_{n+1}) \leq c(s_{s+1}, s_n) + \delta(s_n, s_g)$$

$$h(s_{n+1}) \leq \delta(s_{n+1}, s_g)$$

En base a la demostración del caso base y la hipótesis inductiva, es posible concluir:

$$h(s) \leq \delta(s, s_g) \quad \forall s \in S.$$

# 3 Comparación de las heurísticas para el cubo rubik

Para efectos de comparar las heurísticas, se emplearon las siguientes seis instancias del conjunto de ejemplo:

```
OOOOOOOOOYYYWWWGGGBBBYYYWWWGGGBBBGGGBBBYYYWWWRRRRRRRRR
RRBOOOYYYGYRWWWOGYOBBGYRWWWOGRYBBGWWOGRBBGRBBGRYGRYOOW
WOOWOOGGGYYOWWWRGGBBOYYOWWWRGGBBOYYWRRRBGGBBOYYYBRRBRR
BBROOROOROWWGGBGGGWYYYYYOWBGGGWBRYYYOWYRBBGBRWRBWROWRO
WGWWOOWOOOYYRYYWWRGBGOYYRWWGGRGBGYBOBYYRWRGOOGBBRRRBBB
BOYGOWGOGYBBOYRBGWOGROYYWWRBGWOBBOWWGGWRRWOBBRRYRRYYYG
```

Se utilizaron dichas instacias de cubos debido a que no todas las heurísticas fueron capaces de resolver las instancias de cubos del conjunto de ejemplo en un tiempo razonable. Como Max Manhattan Corners (Max) fue la única heurística capaz de resolver el conjunto de cubos de ejemplo por completo, se procedió a seleccionar instancias de cubos que fueron resueltas en poco tiempo (menos de un segundo) por Max Manhattan Corners, asumiendo que dichas instancias podrían ser resueltas dentro de un tiempo razonable por Min Manhattan Corners (Min) y Sum Manhattan Corners (Sum). Las instancias seleccionadas en base al anterior criterio pudieron ser resueltas por los tres algoritmos. El detalle de los tiempos de ejecución y nodos expandidos se muestra en las tres tablas de a continuación:

Tabla 1: Ejecución con Min Manhattan Corners.

| | expanded-nodes | time |
|---|---:|---:|
| 0 | 53 | 0.755033 |
| 1 | 4812 | 67.741854 |
| 2 | 97 | 1.421339 |
| 3 | 1460 | 21.386039 |
| 4 | 5099 | 74.767006 |
| 5 | 4258 | 62.455686 |
| Total time | | 228.53 |
| Mean expanded | | 2629.84 |

Tabla 2: Ejecución con Max Manhattan Corners.

| | expanded-nodes | time |
|---|---|---|
| 0 | 3 | 0.046549 |
| 1 | 45 | 0.699442 |
| 2 | 3 | 0.048728 |
| 3 | 63 | 0.949137 |
| 4 | 39 | 0.607914 |
| 5 | 37 | 0.573581 |
| Total time | | 2.93 |
| Mean expanded | | 31.67 |

Tabla 3: Ejecución con Sum Manhattan Corners.

| | expanded-nodes | time |
|---|---|---|
| 0 | 2 | 0.033037 |
| 1 | 24432 | 389.265682 |
| 2 | 2 | 0.032844 |
| 3 | 19 | 0.316695 |
| 4 | 682 | 10.497295 |
| 5 | 4 | 0.063975 |
| Total time | | 400.21 |
| Mean expanded | | 4019.17 |

Los resultados observados son altamente consistentes con el análisis teórico de las heurísticas, pues el peor desempeño en ambos indicadores se observa en Sum, la única heurística inadmisible. Esto perjudica la ejecución en tanto cuando una heurística es inadmisible, puede sobrestimar la distancia a una solución en términos de costo, de modo que en ocaciones no reconocerá cuando se encuentra en un estado cercano a una solución y seguirá explorando el espacio de búsqueda. Las heurísticas admisibles rindieron considerablemente mejor precisamente debido a que no sobrestiman la distancia a una solución. Pese a que tanto Max como Min son admisibles, se pudo observar que el desempeño de Max fue superior en aproximadamente dos órdenes de magnitud al de Min. Dicha tendencia se puede explicar en tanto si bien ambas heurísticas son admisibles, Max es un estimador mucho más informativo de la distancia existente entre un estado actual y un estado solución.

Para ilustrar lo anterior, tomemos el escenario donde se llega a un cubo que tiene al menos una esquina en la posición correcta, pero otras en posiciones incorrectas. En tal caso Min será igual a cero independientemente de las distancias de las demás esquinas a sus respectivos objetivos, lo que implica que una vez que se llega a un estado con una esquina en posición correcta, $h = 0$, de modo que $f = g$. Es decir, apenas llegamos a un estado con una esquina correcta con la heurística Min, comenzamos a ejecutar búsqueda por amplitud en vez de A*. Lo anterior es equivalente a decir que apenas tenemos una esquina correcta, Min deja de ser informativo sobre el problema. También existe un estado en el que Max deja de ser informativo para la búsqueda, pero esto sucede cuando todas las esquinas están en su posición correcta. De tal modo, Max es un estimador más informativo que Min en tanto nos permite llegar de manera informada a un estado más cercano a la solución que Min, lo que explica las diferencias en ejecución observadas entre ambos algoritmos.