

Relación 2 - Espacios de estados y técnicas de planificación automática

Cuestiones

Cuestión 1. Describir diferencias entre el grafo que representa un espacio de estados y un árbol de búsqueda en dicho espacio.

Cuestión 2. ¿Son ciertas las siguientes afirmaciones? (justificar la respuesta):

- La lista de ABIERTOS usada en los procedimientos de búsqueda contiene en cada momento todas las hojas del árbol de búsqueda actual.
- Las listas de ABIERTOS y CERRADOS no tienen nodos en común.

Cuestión 3. En la representación de espacio de estados que se ha visto en el tema, la función ACCIONES devuelve la lista de acciones aplicables a un estado ¿Puede influir en algún algoritmo de búsqueda el orden en el que se coloquen esas acciones dentro de esa lista? Justificar la respuesta.

Cuestión 4. Dar un ejemplo de problema de espacios de estados en el que la solución más corta esté a profundidad p , pero que la búsqueda en profundidad acotada con cota p no encuentre la solución (usando la implementación que se ha visto en clase, que usa lista de CERRADOS).

Cuestión 5. Dar una ventaja de la búsqueda en profundidad frente a la búsqueda en anchura, y viceversa.

Cuestión 6. Especificar, razonando la respuesta, un problema de espacio de estados en el que la lista de CERRADOS en los algoritmos de búsqueda en anchura y en profundidad sea superflua.

Cuestión 7. Explicar en qué consiste (y para qué se usa) la técnica de “relajación de un problema de espacio de estados”.

Cuestión 8. Dar un ejemplo de problema de espacios de estados en los cuales aplicando el algoritmo primero el mejor con una heurística admisible no se obtenga solución óptima.

Cuestión 9. Responder si son ciertas o no la siguientes afirmaciones:

- En el lenguaje PDDL, todo estado es un objetivo.
- En el lenguaje PDDL, todo objetivo es un estado.

Cuestión 10. Supongamos que usamos el formalismo PDDL para representar un problema como espacio de estados. ¿Hay alguna diferencia entre el algoritmo de búsqueda por *primero el mejor* y la *búsqueda hacia adelante en profundidad con heurística*?

Cuestión 11. Definir con precisión la heurística Δ_0 usada en planificación hacia adelante y hacia atrás, y dar una idea intuitiva de la misma ¿Dónde y cómo se usaría dicha heurística en la búsqueda hacia atrás (suponiendo un objetivo sin variables)?

Cuestión 12. Consideremos un problema de espacio de estados y su representación mediante el formalismo PDDL. Sea g un objetivo formado únicamente por literales positivos con variables y sea x la única variable que ocurre en los literales de g . Sea $\sigma = \{x/A\}$ la sustitución de x por la constante A . Sea e un estado cualquiera y Δ_0 la heurística considerada en clase. ¿Es cierto que $\Delta_0(e, g) \leq \Delta_0(e, \sigma(g))$?

Cuestión 13. En el algoritmo de planificación de búsqueda hacia atrás se calculan los predecesores de un objetivo sólo para las acciones relevantes. Definir con precisión el concepto de acción relevante para un objetivo sin variables y cómo se calcula el predecesor del objetivo respecto de una acción relevante.

Cuestión 14. Consideremos la heurística Δ_0 definida para los algoritmos de búsqueda en los problemas de planificación con el formalismo PDDL.

- ¿Qué condiciones tiene que verificar una acción A en un paso concreto del algoritmo para que tenga sentido definir sobre ella la heurística Δ_0 ?
- ¿Qué valor se le asigna a la acción? Describe detalladamente cada uno de los elementos que intervienen en la definición.

Responder a las preguntas anteriores en **búsqueda hacia delante** y **búsqueda hacia atrás**.

Problemas

Problema 1. Ayuda a Bruce Willis y a Samuel L. Jackson a resolver el problema que se les plantea en la película *La Jungla de Cristal 3* modelando el problema como un problema de espacios de estados. Puedes ver la escena en http://www.dailymotion.com/video/xbcehl_el-problema-de-los-bidones_tech.

Problema 2. El problema de los misioneros y los caníbales se enuncia de la siguiente manera: “En la orilla izquierda de un río se encuentran M misioneros, C caníbales y una barca en la que caben como máximo P personas; se trata de determinar cómo pueden trasladarse los misioneros y los caníbales a la orilla derecha teniendo en cuenta que en la orilla en la que no esté la barca no debe haber más caníbales que misioneros y que *queremos minimizar la suma total de cambios de orilla realizados por cada una de las personas*”. Plantear el problema como un problema de espacio de estados e indicar qué algoritmo de búsqueda informada y qué heurística se usaría para encontrar una solución.

Problema 3. Dado el conjunto de valores enteros $I = \{i_1, i_2, i_3, \dots, i_N\}$, encontrar un subconjunto no vacío $S \subset I$ cuyos elementos sumen cero. Formaliza este problema como problema de espacio de estados indicando de forma precisa los distintos elementos que lo componen.

Problema 4. Explicar esquemáticamente (sin entrar en detalles de implementación) la representación como espacio de estados del siguiente problema. Definir también una heurística.

El problema es el siguiente: dados cuatro números naturales n , m , r y T , encontrar una secuencia mínima de operaciones aritméticas básicas (suma, resta, multiplicación y división) que usando n y m únicamente y partiendo de 0, obtenga como resultado T , con la restricción adicional de que ni n ni m se pueden usar más de r veces.

Por ejemplo, si $n = 2$, $m = 3$, $r = 3$ y $T = 28$, una posible solución (no necesariamente mínima) es $(((((0 + 3) \times 3) \times 2) - 3) \times 2) - 2$, ya que el resultado es 28 y ni 2 ni 3 se han usado más de tres veces cada uno.

Problema 5. Consideremos el siguiente puzzle: se dispone de 3 fichas negras (N), 3 fichas blancas (B) y una casilla vacía, en la siguiente posición inicial:

N N N B B B V

El objetivo del puzzle consiste en colocar todas las fichas blancas a la izquierda de las fichas negras, independientemente de la posición de la casilla vacía. Para ello, los movimientos permitidos son los siguientes:

- Una ficha puede moverse a una celda adyacente vacía, con coste 1.
- Una ficha puede desplazarse a una celda vacía saltando dos fichas como máximo, con un coste igual al número de fichas saltadas.

Definir una función heurística para este problema.

Problema 6. Consideremos el problema de espacios de estados con los estados A, B, C, D, E, F . La lista siguiente representa la relación sucesor de un estado:

$A \rightarrow B$	$A \rightarrow D$	$B \rightarrow C$	$C \rightarrow D$	$C \rightarrow E$	$C \rightarrow F$	$D \rightarrow B$	$D \rightarrow E$	$E \rightarrow F$
-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------

El estado inicial es A y el estado final es F . Tenemos también la siguiente heurística $h(A) = 5, h(B) = 4, h(C) = 3, h(D) = 2, h(E) = 1, h(F) = 0$. En caso de igualdad de condiciones, usamos el orden alfabético. Se pide:

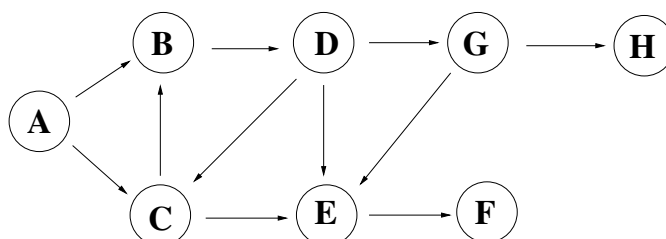
- (a) Aplicar el algoritmo de búsqueda en profundidad iterativa con cota inicial 2 (usando CERRADOS), rellenando la siguiente tabla y justificando cada uno de los pasos. Indicar *explícitamente* la **profundidad** de la solución encontrada.

PASO	ACTUAL	PROF(ACTUAL)	CERRADOS	ABIERTOS
0			\emptyset	[A]
1	A	0
...

- (b) Hacer lo mismo con el algoritmo de búsqueda por primero el mejor.

Comparar las soluciones encontradas.

Problema 7. El siguiente grafo representa un espacio de estados. El estado inicial es A y hay dos estados finales F y H . Tenemos también la siguiente heurística $h(A) = 4, h(B) = 3, h(C) = 3, h(D) = 2, h(E) = 3, h(F) = 0, h(G) = 1, h(H) = 0$. En caso de igualdad de condiciones, usamos el orden alfabético.



Se pide:

- Aplicar el algoritmo de búsqueda en profundidad (usando **CERRADOS**), rellenando la siguiente tabla y justificando cada uno de los pasos. ¿Cuál es la solución encontrada?

PASO	ACTUAL	PROF(ACTUAL)	CERRADOS	ABIERTOS
0			\emptyset	[A]
1	A	0
...

- Aplicar igualmente el algoritmo de búsqueda en profundidad acotada, con cota 3.
- Aplicar igualmente el algoritmo de búsqueda por primero el mejor.

Comparara las soluciones encontradas en cada apartado.

Problema 8. Calcular cuántos nodos se exploran (contando las repeticiones) en el algoritmo de profundidad iterativa en un problema de búsqueda donde el número de sucesores de cualquier nodo es b y la solución se alcanza en el *último* nodo a profundidad n .

Problema 9. Consideremos un problema de espacio de estados con factor de ramificación constante b y con una única solución que se encuentra a profundidad d .

- Calcular, tanto en el mejor como en el peor de los casos, el número de nodos que se necesitan analizar para encontrar la solución, al aplicar un algoritmo de búsqueda en anchura. Análogamente para la búsqueda en profundidad.
- Calcular, tanto en el mejor como en el peor de los casos, el máximo número de nodos que podrían estar a la espera de ser analizados en la lista de **ABIERTOS**, al aplicar el algoritmo de búsqueda en anchura. Lo mismo para el algoritmo de búsqueda en profundidad iterativa.

Problema 10. Un grupo de 5 personas quiere cruzar un viejo y estrecho puente. Es una noche cerrada y se necesita llevar una linterna para cruzar. El grupo sólo dispone de una linterna, a la que le quedan 5 minutos de batería.

1. Cada persona tarda en cruzar 10, 30, 60, 80 y 120 segundos, respectivamente.
2. El puente sólo resiste un máximo de 2 personas cruzando a la vez, y cuando cruzan dos personas juntas caminan a la velocidad del más lento.
3. No se puede lanzar la linterna de un extremo a otro del puente, así que cada vez que crucen dos personas, alguien tiene que volver a cruzar hacia atrás con la linterna a buscar a los compañeros que falten, y así hasta que hayan cruzado todos.

Se pide:

- a) Plantear el problema como búsqueda en espacio de estados, describiendo claramente todos los elementos necesarios de la representación.
- b) Definir una heurística.

Problema 11. Considérese el siguiente conjunto de predicados que describen el mundo en un problema de planificación de acciones de un camión T que transporta paquetes entre ciudades:

- $\text{PAQUETE}(x)$: el objeto x es un paquete.
- $\text{CIUDAD}(x)$: el objeto x es una ciudad.
- $\text{AUTOVÍA}(c1, c2)$: las ciudades $c1$ y $c2$ están conectadas por autovía.
- $\text{EN}(x, c)$: el objeto x (el camión o un paquete) está en la ciudad c .
- $\text{DENTRO-CAMIÓN}(x)$: el paquete x está cargado en el camión.
- $\text{DESCARGADO}()$: el camión está descargado.

Las acciones que se pueden realizar son las siguientes:

- $\text{CARGA}(p, c)$: el camión (que debe estar descargado) carga el paquete p en la ciudad c . Una vez cargado, el paquete ya no se considera que esté en la ciudad c .
- $\text{DESCARGA}(p, c)$: el camión descarga el paquete p en la ciudad c .
- $\text{IR}(c1, c2)$: el camión se desplaza por autovía desde la ciudad $c1$ a la ciudad $c2$.

Supongamos que deseamos encontrar la secuencia de acciones que a partir de un estado inicial en el que un paquete P1 está en Barcelona, un paquete P2 está en Madrid, y el camión T está en Sevilla, deja finalmente el paquete P1 en Sevilla, el paquete P2 en Barcelona y el camión descargado. Supondremos que existe una autovía entre Barcelona y Madrid y otra entre Madrid y Sevilla.

Representar el problema en el formalismo PDDL. Es decir, describir el estado inicial, el objetivo y las acciones.

Problema 12. Robi el robot se encuentra en una casa en cuyas habitaciones hay distintos objetos que puede transportar. Básicamente, Robi puede: trasladarse de una habitación a otra, coger un objeto que está en la misma habitación en la que está él, y soltar un objeto que tenga cogido, en la habitación en la que esté (con la restricción de que no puede coger más de un objeto a la vez). Si el objetivo es llevar todos los objetos a la cocina, plantear este problema como un problema de planificación, expresándolo al estilo PDDL (esto es, definiendo el lenguaje usado, las acciones, el estado inicial y el objetivo).

Problema 13. Consideremos un dominio que consta de un robot que puede desplazar cajas entre habitaciones conectadas.

Representamos este dominio con los siguientes símbolos:

- C1, C2 las dos cajas; H1, H2, H3 las tres habitaciones; P1, P2 las dos puertas.
- $\text{ABIERTA}(x)$: la puerta x está abierta
- $\text{EN}(x, y)$: la caja x está en la habitación y
- $\text{ROBOT-EN}(x)$: el robot está en la habitación x
- $\text{CONECTA}(x, y, z)$: la puerta x conecta las habitaciones y y z

Dar una representación, en el lenguaje PDDL, de las siguientes acciones:

- $\text{IR-VIA}(x, y, z)$: el robot va de la habitación y a la z a través de la puerta x
- $\text{DESPLAZA-VIA}(c, h1, h2, p)$: el robot desplaza la caja c de $h1$ a $h2$ vía la puerta p

- $CIERRA(x)$: el robot cierra la puerta x
- $ABRE(x)$: el robot abre la puerta x

Problema 14. Consideremos el problema del niño y el caramelo. En una habitación hay un niño que quiere un caramelo que está colgado de un árbol de navidad. El niño no alcanza al caramelo pero hay un taburete con el que puede alcanzarlo, puesto que si se sube en él está a la altura del caramelo. Inicialmente, el niño se encuentra en la habitación A, el caramelo en la B y el taburete en la habitación C.

Las acciones que se pueden realizar son las siguientes: IR de una habitación a otra; DESPLAZAR un objeto de una habitación a otra; SUBIRSE a un objeto; y AGARRAR un objeto (el niño puede agarrar un objeto si ambos están en la misma habitación y a la misma altura). El objetivo es, por supuesto, que el niño obtenga el caramelo. Se pide:

1. Declarar los símbolos de constantes y de predicados necesarios para representar el dominio del problema.
2. Dar una representación, en el lenguaje PDDL, de las acciones anteriores.
3. Especificar el estado inicial y el objetivo como conjunción de literales.

Problema 15.

Para los siguientes problemas de planificación, aplicar el algoritmo de **búsqueda en profundidad** (*backtracking*) para encontrar una solución. Indicar claramente el árbol de búsqueda que se genera, el orden en que se van analizando los estados, y la solución finalmente obtenida. (*Nota:* como orden entre los sucesores de un estado, considerar las acciones aplicables en orden alfabético).

- (a) • Acciones:

Acción	Precondiciones	Efectos
A	p_1	$\neg p_1, p_2$
B	p_1	$\neg p_1, p_3$
C	p_4	p_1
D	p_1	p_4

- Estado inicial: $\{p_1\}$
- Objetivo: $\{p_2, p_3\}$

- (b) • Acciones:

Acción	Precondiciones	Efectos
A	p_6	$\neg p_6, p_2$
B	p_3	$\neg p_4, p_5$
C	p_3	$\neg p_3, p_2$
D	p_2	$\neg p_2, \neg p_5, p_4$
E	p_1	p_3
F	p_3	$\neg p_1, p_5$

- Estado inicial: $\{p_1\}$
- Objetivo: $\{p_4, p_5\}$

Problema 16. En cada uno de los siguientes problemas de planificación, se da un estado e y un objetivo g . Se pide calcular $\Delta_0(e, g)$

- (a) • Acciones:
- | A | B | C |
|-------------|---------------|-----------------|
| ----- | ----- | ----- |
| Pre: P3, P4 | Pre: P5 | Pre: P3, P4 |
| Acc: P1, P2 | Acc: P1,P2,P6 | Acc: P1, P2, P5 |
- Estado e : {P3,P4}
 - Objetivo g : {P1,P2,P3}
-

- (b) • Acciones:
- | A | B | C | D |
|---------|---------|---------|-----------------|
| ----- | ----- | ----- | ----- |
| Pre: P2 | Pre: P4 | Pre: P6 | Pre: P1, P4, P5 |
| Acc: P4 | Acc: P5 | Acc: P5 | Acc: P3, P7 |
- Estado e : {P1,P2}
 - Objetivo g : {P3, P7}
-

- (c) • Acciones:
- | A | B | D |
|-------------|-------------|-------------|
| ----- | ----- | ----- |
| Pre: P6 | Pre: P3 | Pre: P3 |
| Efe: -P6,P2 | Efe: -P4,P5 | Efe: -P3,P2 |
-
- | E | F | G |
|-----------------|---------|-------------|
| ----- | ----- | ----- |
| Pre: P2 | Pre: P1 | Pre: P3 |
| Efe: -P2,-P5,P4 | Efe: P3 | Efe: -P1,P5 |
- Estado e : {P1}
 - Objetivo g : {P4,P5}