

Resumen-necesario-para-aprobar.pdf



MarcosPG



Inteligencia Artificial



3º Grado en Ingeniería Informática



**Escuela de Ingeniería y Arquitectura
Universidad de Zaragoza**



**Todas tenemos una amiga
experta en recorrer
kilómetros en discotecas.
Y si no la tienes eres tú.**

Para ser una experta en kilómetros
de verdad, déjate guiar por **coches.net**



Compra
o vende
tu coche

✓ fácil
✓ rápido

coches.net



Tu colega "el experto en Erasmus": se fue pensando en aprobarlo todo.
No aprobó nada. Lo probó todo.

PARTE 1

BÚSQUEDA

Nomenclatura:

- Nodo: estado de un árbol de búsqueda.
- Expandir estado: proceso de generar nuevos estados aplicando acciones permitidas.
- Nodo expandido: nodo que parte de los resultantes.
- Frontera/Lista abierta: conjunto de nodos sin hijos.
- Arco: transición de estados como resultado de acciones.
- Test objetivo: indica cuando se evalúa si un nodo es el objetivo de la búsqueda (dos tipos: al generar o al expandir).

La búsqueda se realiza sobre un espacio de estados.

Esta búsqueda se puede hacer de dos formas en árbol o en grafo:

- Árbol: en esta búsqueda solo hay frontera, por lo tanto se pueden repetir los nodos que exploremos.
- Grafo: ahora contamos con frontera y explorados, por lo que si un nodo ya se ha explorado no se vuelve a explorar.

Existen 3 tipos de búsqueda:

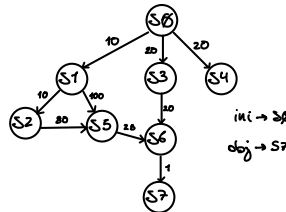
- No informada: aquellas que sólo usan la información disponible en la definición del problema.
- Informada: en esta si se usa el conocimiento extra que se tenga del problema (heurísticas).
- Local: se usan para buscar las soluciones más óptimas (les da igual que sea objetivo o no mientras sea el mejor).

BÚSQUEDA NO INFORMADA

Método general a seguir para hacer estos ejercicios:

- 1- Localizamos el estado inicial, el cual nos dibujamos y añadimos a la frontera.
- 2- Expandimos el estado que toque (cambiará según el tipo de búsqueda), añadiendo al árbol/graf de búsqueda los nuevos nodos que aparezcan al explorar. Estos nodos los introduciremos también en la frontera, además si la búsqueda es en grafo añadiremos el nodo que hemos expandido a la lista de explorados/expandidos.
- 3- Repetiremos el paso anterior hasta que lleguemos a evaluar el nodo objetivo. Esta evaluación dependerá del tipo de test objetivo que tenga la búsqueda, si es de expansión la búsqueda finaliza al expandir el nodo objetivo, si es de generación se finaliza en el momento que el nodo objetivo entra en la frontera.

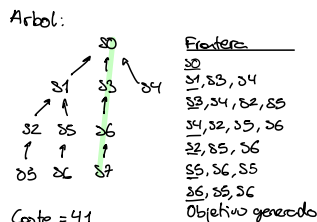
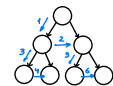
Para los ejemplos usaremos el siguiente espacio de estados:



Tipos de búsqueda no informada (hay mas pero estos son los que preguntan):

Anchura (BFS):

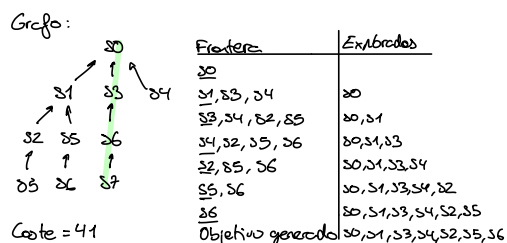
Es igual que en EDA, dado un nodo exploras sus hijos de izquierda a derecha, una vez los tengas, continuas explorando los nietos de la misma manera. En esta búsqueda en la lista de explorados se va generando una cola FIFO. Su test objetivo se realiza al generar los nodos.



Frontera:

S0
S1, S3, S4
S2, S5, S6, S7
S5, S6, S7
S6, S7, S7

Objetivo generado



Frontera:

S0
S1, S3, S4
S2, S5, S6, S7
S5, S6, S7
S6, S7, S7

Objetivo generado

Explorados:

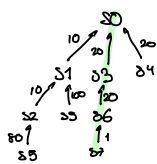
S0
S0, S1
S0, S1, S3
S0, S1, S3, S4
S0, S1, S3, S4, S2
S0, S1, S3, S4, S2, S5
S0, S1, S3, S4, S2, S5, S6



Coste uniforme (UC):

Expande el nodo que cuente con menor coste en la frontera. Test objetivo de expansión.

Arbol:

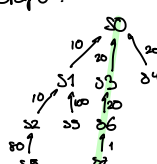


$C = 41$

Frontera:

$S0(0)$
 $S1(10), S3(20), S4(20)$
 $S2(10+10=20), S3(20), S4(20), S5(110)$
 $S3(20), S4(20), S5(110), S5(110)$
 $S4(20), S6(40), S5(110), S5(110)$
 $S6(40), S5(110), S5(110)$
 $S7(41)$
 Objetivo expandido

Grafo:



$C = 41$

Frontera:

$S0(0)$
 $S1(10), S3(20), S4(20)$
 $S2(20), S3(20), S4(20), S5(110)$
 $S3(20), S4(20), S5(110)$
 $S4(20), S6(40), S5(110)$
 $S6(40), S5(110)$
 $S7(41)$
 Objetivo expandido

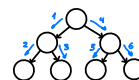
Explorados:

$S0(0)$
 $S0(0), S1(10)$
 $S0(0), S1(10), S2(20)$
 $S0(0), S1(10), S2(20), S3(20)$
 $S0(0), S1(10), S2(20), S3(20), S4(20)$
 $S0(0), S1(10), S2(20), S3(20), S4(20), S6(40)$
 $S0(0), S1(10), S2(20), S3(20), S4(20), S6(40), S7(41)$

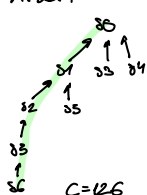
Profundidad (DFS):

Es igual que en EDA, vas expandiendo los nodos de la izquierda, hasta que este sea vacío, entonces comienzas a aplicar la búsqueda en profundidad a los de la derecha.

En la frontera se va generando una cola LIFO. Su test objetivo se realiza al generar el nodo.



Arbol:

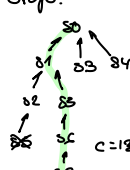


$C = 126$

Frontera:

$S0$
 $S1, S3, S4$
 $S2, S6, S3, S4$
 $S6, S5, S3, S4$
 $S6, S5, S3, S4$
 Objetivo

Grafo:



$C = 126$

Frontera:

$S0$
 $S1, S3, S4$
 $S2, S5, S3, S4$
 $S5, S3, S4$
 $S6, S3, S4$
 $S7, S3, S4$
 Objetivo

Explorados:

$S0$
 $S0, S1$
 $S0, S1, S2$
 $S0, S1, S2, S3$
 $S0, S1, S2, S3, S4$
 $S0, S1, S2, S3, S4, S6$
 $S0, S1, S2, S3, S4, S6, S7$

BÚSQUEDA INFORMADA

En este tipo de búsqueda si se hace uso del conocimiento que se tenga del problema. Se usa una función de evaluación de cada nodo, $f(n)$ que estima como de prometedor es el nodo.

Hay que distinguir función de evaluación, $f(n)$, de heurística, $h(n)$:

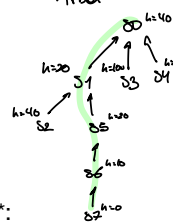
- $f(n)$: función de evaluación del coste. La elección de esta determina la estrategia.
- $h(n)$: función heurística, una estimación del coste del camino menos costoso desde el estado en el nodo "n" hasta el objetivo. Depende sólo del estado en "n".

Estrategias de búsqueda informada (hay más pero estas son las que preguntan):

Primero el mejor/Voraz:

Expande el nodo que cuente con menor función heurística $h(n)$. El tipo de t.o. lo especificaran en el enunciado.

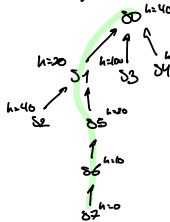
Arbol:



Frontera:

$S0(40)$
 $S1(20), S3(100), S4(110)$
 $S2(20), S2(40), S3(100), S4(110)$
 $S6(10), S2(40), S3(100), S4(110)$
 Objetivo S7 generado

Grafo:



Frontera:

$S0(40)$
 $S1(20), S3(100), S4(110)$
 $S2(20), S2(40), S3(100), S4(110)$
 $S6(10), S2(40), S3(100), S4(110)$
 Objetivo S7 generado

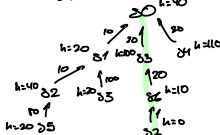
Explorados:

$S0(40)$
 $S0(40), S1(20)$
 $S0(40), S1(20), S2(20)$
 $S0(40), S1(20), S2(20), S3(100)$
 $S0(40), S1(20), S2(20), S3(100), S4(110)$

A*:

Expande el nodo que cuente con menor función de evaluación de coste $f(n)$. Test objetivo de expansión.

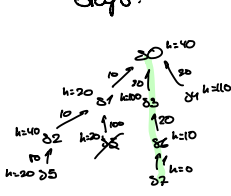
Arbol:



Frontera:

$S0(40)$
 $S1(20), S3(120), S4(130)$
 $S2(40), S3(120), S4(130), S5(130)$
 $S3(120), S5(120), S4(130), S5(130), S5(130)$
 $S6(50), S5(120), S4(130), S5(130)$
 $S7(20), S5(120), S4(130), S5(130)$
 Objetivo

Grafo:



Frontera:

$S0(40)$
 $S1(20), S3(120), S4(130)$
 $S2(40), S3(120), S4(130), S5(130)$
 $S3(120), S5(120), S4(130)$
 $S6(50), S5(120), S4(130)$
 $S7(20), S5(120), S4(130)$
 Objetivo

Explorados:

$S0(40)$
 $S0(40), S1(20)$
 $S0(40), S1(20), S2(20)$
 $S0(40), S1(20), S2(20), S3(120)$
 $S0(40), S1(20), S2(20), S3(120), S4(130)$
 $S0(40), S1(20), S2(20), S3(120), S4(130), S6(50)$
 $S0(40), S1(20), S2(20), S3(120), S4(130), S6(50), S7(20)$

BÚSQUEDA LOCAL

Se centran en encontrar el nodo más óptimo, encontrando el estado mejor de acuerdo a una función objetivo.

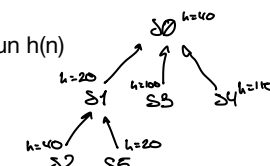
Estrategias de búsqueda local (hay más pero solo preguntan este):

Escalada/Hill-Climbing:

Avanza a la $h(n)$ menor y acaba cuando los nodos hijos NO tienen un $h(n)$ con valor menor (o el valor es el mismo).

A este nodo se le conoce como mínimo local y es lo que devolverá la búsqueda como resultado.

La frontera es local, es decir solo contiene los nodos resultantes de la última expansión.



Frontera:

$S0(40)$
 $S1(20), S3(100), S4(110)$
 $S2(40), S5(20)$

S1 mínimo local

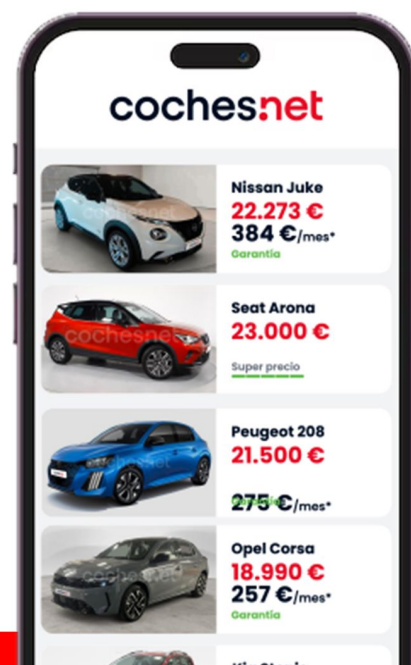
Nacida para sacar matrículas, experta en tener que matricularme tres veces de lo mismo.

Si quieres ser una verdadera experta, confía en coches.net



Vende o compra
tu coche

- ✓ Nuevos
- ✓ Renting
- ✓ Km 0
- ✓ Segunda Mano



Inteligencia Artificial



Comparte estos flyers en tu clase y consigue más dinero y recompensas



Banco de apuntes de la

WUOLAH

1 Imprime esta hoja

2 Recorta por la mitad

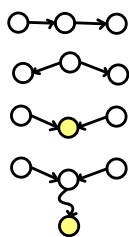
3 Coloca en un lugar visible para que tus compis puedan escanar y acceder a apuntes

4 Llévate dinero por cada descarga de los documentos descargados a través de tu QR

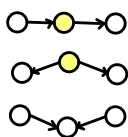


Reservados todos los derechos. No se permite la explotación económica ni la transformación de esta obra. Queda permitida la impresión en su totalidad.

Triplettes activos

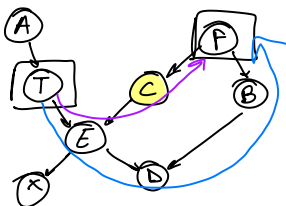


Tripletes inactivos



- 1- Sacar todos los caminos (sin hacer caso a las flechas) que unen los nodos que queremos saber si son independientes.
- 2- Evaluar las tripletas de cada camino (aquí si importan las flechas):
 - Si el coloreado no está en el del centro entonces no cuenta como coloreado.
 - Si TODAS las tripletas son activas entonces el camino es activo.
- 3- Una vez hemos evaluado todos los caminos, si todos son inactivos, entonces los nodos son independientes.

Example:



iT, F indep. dtdo C ?

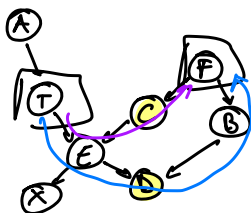
Genino 1:

$F \rightarrow E \leftarrow C$ Inac.
 $E \leftarrow C \leftarrow F$ Inac.

Cucino 2:

$\textcircled{1} \rightarrow \textcircled{E} \rightarrow \textcircled{D}$ Act.
 $\textcircled{E} \rightarrow \textcircled{D} \leftarrow \textcircled{B}$ Inac.
 $\textcircled{D} \leftarrow \textcircled{B} \leftarrow \textcircled{F}$ Act.

T y F son independientes ddo C



iT, F indep. dtdo C, D ?

Guano 1:

$(T \rightarrow E \leftarrow C)$ Inac. $\{ \text{Caus. 1 es independiente}$
 $(E \leftarrow C \leftarrow F)$ Inac.

Casino 2:

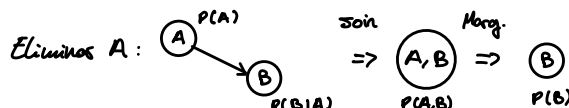
$T \rightarrow E \rightarrow D$ Act.
 $E \rightarrow D \leftarrow B$ Act
 $D \leftarrow B \leftarrow F$ Act.

} Gen. 2 es dependiente

T, F son dependientes dado C, D

Quando se pide la probabilidad de un caso (query) dadás una serie de condiciones (evidencias), no siempre es necesaria toda la información que se da en el problema, y para obtener el caso concreto que se pide se aplica el algoritmo de eliminación en aquellas probabilidades que sobren.

Este proceso se puede ver como ir metiendo probabilidades que sobran dentro de su hijo/hijos:

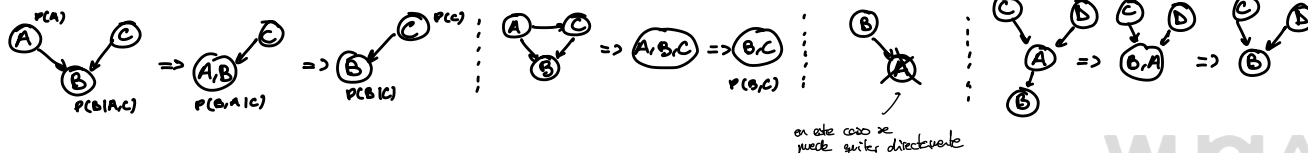


El algoritmo son dos pasos:

- Primero se hace un Join, que es básicamente multiplicar las diferentes posibilidades de las probabilidades, obteniendo así la prob. conjunta.
- Y en segundo lugar se Marginaliza, que consiste en eliminar de la prob. conjunta la prob. que no queremos, esto se hace sumando.

Nos encontraremos generalmente con las siguientes situaciones:

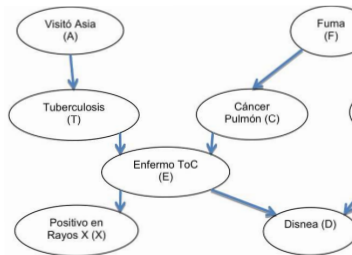
Eliminamos A:





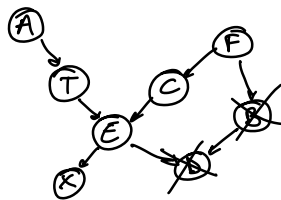
Tu colega "el experto en Erasmus": se fue pensando en aprobarlo todo.
No aprobó nada. Lo probó todo.

Ejemplo:



$P(a) = 0.01$	$P(c c, t) = 1$	$P(f) = 0.50$	$P(d e, b) = 0.90$	$P(b f) = 0.60$
$P(t a) = 0.05$	$P(c c, -t) = 1$	$P(c f) = 0.10$	$P(d e, -b) = 0.70$	$P(b f) = 0.30$
$P(t -a) = 0.01$	$P(e -c, t) = 1$	$P(c -f) = 0.01$	$P(d -e, b) = 0.80$	
	$P(e -c, -t) = 0$		$P(d -e, -b) = 0.10$	
$P(x e) = 0.98$				
$P(x -e) = 0.05$				

$P(T|+, +, f)$?



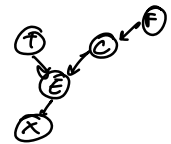
Eliminar A:

T A	$P(A) \cdot P(T A)$	$\rightarrow P(T, A)$	$\rightarrow P(T)$
++	0.01	0.05	0.0005
+-	0.04	0.01	0.0004
-+	0.01	0.05	0.0005
--	0.04	0.01	0.0004

$$P(+t) = 0.00104$$

$$P(-t) = 0.00096$$

hoy que son esas probabilidades en las que lo único que cambia es la que queremos eliminar



Eliminamos E:

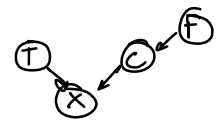
X C T E	$P(x E) \cdot P(E C, T)$	$\rightarrow P(x, E C, T)$	$\rightarrow P(x C, T)$
++ ++ ++	0.98	1	0.98
++ ++ -	0.05	0	0
++ ++ +	0.98	1	0.98
++ ++ -	0.05	0	0
++ - + +	0.98	1	0.98
++ - + -	0.05	0	0
++ - - +	0.98	1	0.98
++ - - -	0.05	0	0

$$P(+x|+, +t) = 0.98$$

$$P(+x|+, -t) = 0.98$$

$$P(+x|-, +t) = 0.98$$

$$P(+x|-, -t) = 0.05$$

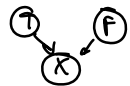


Eliminamos C:

F X T C	$P(+x C, T) \cdot P(C +, f)$	$\rightarrow P(+x, C T, f)$	$\rightarrow P(+x T, f)$
++ ++ ++	0.98	0.1	0.098
++ ++ -	0.05	0.9	0.045
++ - + +	0.98	0.1	0.098
++ - + -	0.05	0.9	0.045

$$P(+x|+, +f) = 0.098$$

$$P(+x|+, -f) = 0.045$$



Sumamos todos:

Structured Tables:

F X T	$P(T) \cdot P(+f) \cdot P(+x T, +f)$	$\rightarrow P(T, +f, +x)$
$++ ++ ++$	0.0104 0.5 0.198	$P(++ ++ ++)= 0.00850$
$++ ++ -$	0.0096 0.5 0.143	$P(++ ++ -)= 0.00713$

$$P(+t, +f, +x) = 0.0052$$

$$P(-t, +f, +x) = 0.0048$$

Normalizamos:

$$P(+t|+, +f, +x) = \frac{0.0052}{0.0052 + 0.0048} = 0.52$$

$$P(-t|+, +f, +x) = 1 - 0.52 = 0.48$$

WUOLAH

