

Algoritmo Graphplan

La construcción de un árbol de planificación permite estimar el número de pasos necesarios para alcanzar un objetivo, g , a partir de un determinado estado, s_o . Si el grafo indica que g no es alcanzable, está garantizado que no existe un plan con las acciones disponibles que permita llegar a g desde s_o . Además el grafo tampoco sobrestima el número de pasos.

1 Representación clásica

Los estados son conjuntos de átomos, fórmulas de la forma $P(o_1, \dots, o_n)$ donde P es un símbolo de predicado y cada o_i es una constante, que son considerados ciertos. El resto de los átomos (los que no aparecen en el estado, son considerados falsos).

Los objetivos se representan también con un conjunto de átomos, los que deben ser ciertos, es decir, los que deben aparecer en el estado.

Las acciones añaden y/o eliminan átomos a los estados. Las precondiciones son conjuntos de átomos y los efectos como los efectos se representan con dos conjuntos de átomos: efectos+ y efectos-.

Una acción es aplicable a un estado, s , si $\text{precond} \subseteq s$. El estado que resulta al aplicar la acción es $(S \setminus \text{efectos-}) \cup \text{efectos+}$.

Pueden definirse operadores o esquemas de acciones utilizando variables. Las distintas formas de instanciar estas variables darán lugar a un conjunto de acciones concretas.

1.1 Equivalencia con la representación mediante variables de estado

Desde el punto de vista teórico ambos sistemas de representación tienen una expresividad equivalente. Cualquier problema representado utilizando uno de los dos sistemas puede ser traducido fácilmente al otro con un coste lineal.

Clásica		Variables
$P(o_1, \dots, o_n)$	\implies	$P(o_1, \dots, o_n) = 1$
$-P(o_1, \dots, o_n)$	\implies	$P(o_1, \dots, o_n) = 0$
$P(o_1, \dots, o_n, v)$	\Leftarrow	$P(o_1, \dots, o_n) = v$
precond : $P(o_1, \dots, o_n, w)$	\Leftarrow	$P(o_1, \dots, o_n) \leftarrow v$
$P(o_1, \dots, o_n, v)$		
$-P(o_1, \dots, o_n, w)$		

2 Grafos de planificación

Es un grafo dirigido dividido en niveles. Los arcos sólo son posibles entre nodos de un nivel y el siguiente nivel. Los nodos del nivel P_0 se corresponden con el conjunto de átomos que definen el estado inicial s_0 . Los restantes niveles $1 \leq i$ contienen dos capas de nodos, una correspondiente a acciones, A_i y otra a átomos P_i .

- Los nodos de la capa A_i se corresponden con acciones cuyas precondiciones aparecen en la capa P_{i-1} . Cada precondición está conectada con la correspondiente acción.
- la capa P_i es la unión de la capa P_{i-1} y el conjunto de efectos positivos de las acciones de la capa A_i . Cada acción está conectada con cada uno de sus efectos

Entre las acciones hay incompatibilidades, es decir, hay acciones que no pueden suceder de forma independiente; por ejemplo, si la precondición de una acción es un efecto negativo de otra.

Para facilitar la detección de incompatibilidades se introducen para cada átomo una acción de persistencia que tiene como única precondición y efecto positivo dicho átomo.

Las relaciones de incompatibilidad entre las acciones de una capa, A_i , dan lugar a incompatibilidades entre los átomos de la siguiente capa P_i ; y estas incompatibilidades producen incompatibilidades adicionales entre las acciones de la siguiente capa A_{i+1} .

Estas incompatibilidades o *mutex* se definen como sigue:

- Dos acciones, **a** y **b**, son mutex si:
 - un efecto negativo de **a** (resp. **b**) es una precondición de **b** (resp. **a**)
ó
 - un efecto negativo de **a** (resp. **b**) es un efecto positivo de **b** (resp. **a**)
ó
 - una precondición de **a** es mutex con una precondición de **b**
- Dos átomos, **p** y **q**, son mutex si:
 - todos los pares de acciones que los (incluidas las acciones de persistencia) son mutex
y
 - ninguna acción los produce a ambos

Los átomos de la capa P_0 no son mutex entre sí y las acciones de persistencia de la capa A_1 tampoco. Estas últimas ya que sus precondiciones no lo son y no tienen efectos negativos.

Los conjunto de acciones independientes de un nivel (ningún par de las acciones del conjunto es mutex) pueden aplicarse, sobre los estados que contengan todas las precondiciones de las mismas, en cualquier orden.

3 Algoritmo GRAPHPLAN

El algoritmo GRAPHPLAN construye sucesivamente las capas del grafo de planificación.

El primer intento de extracción de un plan se inicia, en la última capa de acciones construida, en el momento en el la capa de átomos contenga al objetivo. Para ello se selecciona un conjunto de acciones (sin mutex entre sí) de dicha capa que contenga al objetivo entre la unión de sus efectos. Las precondiciones de dichas acciones pasan a ser el “nuevo objetivo” para la extracción de un plan en la capa de acciones anterior. Si en algún momento no se encuentra ningún conjunto de acciones con las condiciones exigidas se escoge un nuevo conjunto en el nivel posterior. Si se alcanza el nivel inicial, los conjuntos de acciones obtenidos constituirán la solución.

```
1: procedure GRAPHPLAN( $s_0, g$ )
2:    $P_0 \leftarrow s_0$ 
3:    $i \leftarrow 0$ 
4:   repeat
5:     if  $g \subseteq P_i$  then                                     ▷ sin mutex entre ellos
6:        $RES \leftarrow$  extraer plan
7:       if  $RES \neq$  fallo then
8:         return  $RES$ 
9:       end if
10:    else
11:       $i \leftarrow i + 1$ 
12:       $A_i \leftarrow$  acciones cuyas precondiciones están en  $P_{i-1}$       ▷ sin mutex entre ellas
13:      Calcular mutex en  $A_i$ 
14:       $P_i \leftarrow$  efectos positivos de  $A_i$ 
15:      Calcular mutex en  $P_i$ 
16:      Añadir un arco entre cada acción de  $A_i$  y, cada una de sus precondiciones en  $P_{i-1}$ 
        y cada uno de sus efectos positivos en  $P_i$ . Añadir un arco discontinuo entre cada acción de  $A_i$ 
        y cada uno de sus efectos negativos en  $P_i$ .
17:    end if
18:    until  $P_i \equiv P_{i-1}$                                           ▷ mismos átomos y mutex
19:    return fallo
20: end procedure
```

El grafo de planificación que se obtiene con el algoritmo GRAPHPLAN (deteniendo el mismo en el paso 6 sin necesidad de intentar la extracción de un plan) permite definir una heurística admisible (sin sobreestimar el número de acciones necesarias para alcanzar el objetivo). El valor vendrá dado por la distancia entre el primer nivel que contenga al estado considerado y el último nivel generado.

4 Objetivos del trabajo

Implementar parte del algoritmo GRAPHPLAN y realizar un análisis comparativo realizando búsquedas a distintas versiones del problema de los robots estibadores (ver versión generalizada

del boletín de problemas). Para ello:

- Elegir al menos seis instancias distintas del problema con varias localizaciones, pilas de contenedores, gruas, ...).
- Comparar los resultados obtenidos para resolver las instancias anteriores utilizando los algoritmos:
 - de búsqueda en profundidad
 - de búsqueda primero el mejor. Comparar el resultado utilizando varias heurísticas.
 - Utilizar la construcción del grafo de planificación que se obtiene al aplicar el algoritmo GRAPHPLAN para el cálculo de la heurística, tal y como se ha descrito.
 - de descomposición hacia adelante

Para la comparativa es interesante medir entre otros, tiempo y espacio necesarios (nodos analizados, generados, ...), calidad de la solución obtenida, ... La construcción del grafo de planificación no debe tenerse en cuenta en el análisis ya que la complejidad de GRAPHPLAN reside en el proceso de extracción del plan.

4.1 Representación

Puede utilizarse la representación clásica descrita, una representación mediante variables de estado o una extensión de la representación clásica que permite el uso de literales (átomo y negaciones de átomos) tanto en las precondiciones de las acciones como en el objetivo. Debe utilizarse el mismo método de representación para todos los algoritmos.

En cualquier caso es necesario incluir la descripción del sistema utilizado y las modificaciones, en caso de que sean necesarias, sobre los distintos algoritmos.

4.2 Evaluación

Algunas consideraciones que se tendrán en cuenta en la evaluación de este trabajo:

- La eficiencia de la implementación, el buen estilo de programación en Python y la documentación del código.
- La claridad con que se incluyan los resultados obtenidos.
- Cualquier aportación original que se incluya en el trabajo además de las descritas.
- Posibilidad de reproducir los resultados presentados (salvo los de tiempo y espacio, ya que dependen del ordenador utilizado).

5 Documentación

Puede encontrarse una descripción sobre el sistema de representación descrito y el algoritmo GRAHPLAN en:

S. J. Russell y P. Norvig. *Artificial Intelligence: A Modern Approach*. 3ª ed. Prentice Hall Series in Artificial Intelligence. Pearson, 2009. 1132 págs. ISBN: 978-0-13-604259-4. URL: <http://aima.cs.berkeley.edu/>.

M. Ghallab, D. Nau y P. Traverso. *Automated Planning. Theory and Practice*. Morgan-Kaufmann, 2004. 635 págs. ISBN: 978-1-55860-856-6. URL: <http://projects.laas.fr/planning/>.

Apéndice: Ejemplo

Se incluye como ejemplo el problema del mundo de los bloques tal y como se describe en la presentación del tema de Planificación

Problema del mundo de los bloques

- Átomos:

Símbolo de predicado `bloque_encima`: dados $b \in Bloques \cup \{MESA\}$ y $b' \in Bloques \cup \{NINGUNO\}$ (distinto de b), tenemos el átomo `bloque_encima(b, b')`.

Símbolo de predicado `bloque_cogido`: dados $b \in Bloques \cup \{NINGUNO\}$, tenemos el átomo `bloque_cogido(b)`

- Estado inicial:

$$s_0 = \{\text{bloque_encima}(\text{MESA}, A), \text{bloque_encima}(A, \text{NINGUNO}), \\ \text{bloque_encima}(\text{MESA}, B), \text{bloque_encima}(B, C), \\ \text{bloque_encima}(C, \text{NINGUNO}), \text{bloque_cogido}(\text{NINGUNO})\}$$

- Objetivo:

$$g = \{\text{bloque_encima}(\text{MESA}, C), \text{bloque_encima}(C, B), \text{bloque_encima}(B, A)\}$$

- Operadores:

Para cada $b \in Bloques$

`coger_bloque_mesa(b)`:

- * precondition: `bloque_encima(MESA, b), bloque_encima(b, NINGUNO), bloque_cogido(NINGUNO)`
- * efectos+: `bloque_cogido(b)`
- efectos−: `bloque_encima(MESA, b), bloque_encima(b, NINGUNO), bloque_cogido(NINGUNO)`

`poner_bloque_mesa(b)`:

- * precondition: `bloque_cogido(b)`
- * efectos+: `bloque_encima(MESA, b), bloque_encima(b, NINGUNO), bloque_cogido(NINGUNO)`
- efectos−: `bloque_cogido(b)`

Para cada $b, b' \in Bloques$ ($b \neq b'$)

`coger_bloque_pila(b, b')`:

- * precondition: `bloque_encima(b', b), bloque_encima(b, NINGUNO), bloque_cogido(NINGUNO)`
- * efectos+: `bloque_cogido(b), bloque_encima(b', NINGUNO),`

```

efectos-: bloque_encima(b', b), bloque_encima(b, NINGUNO),
          bloque_cogido(NINGUNO)

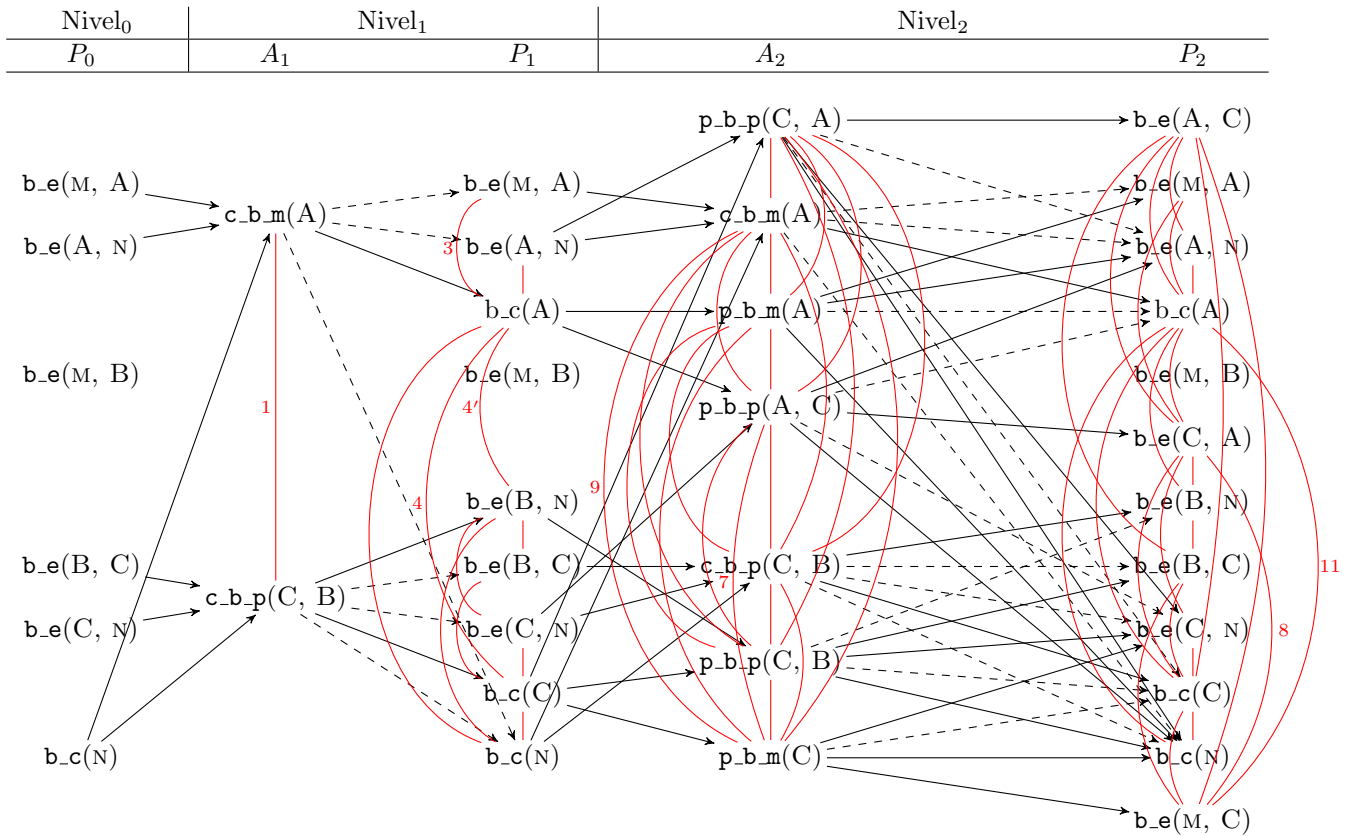
poner_bloque_pila(b, b'):

* precondition: bloque_cogido(b), bloque_encima(b', NINGUNO),
* efectos+: bloque_encima(b', b), bloque_encima(b, NINGUNO),
            bloque_cogido(NINGUNO)
efectos-: bloque_cogido(b), bloque_encima(b', NINGUNO),

```

Primeros niveles del grafo de planificación

Por simplificar utilizaremos sólo las iniciales de los símbolos de predicado, de las constantes y de los nombres de las acciones. Por el mismo motivo no se han incluido las acciones de persistencia.



Las acciones están conectadas a través de líneas discontinuas con sus efectos negativos. Los enlaces mutex entre acciones y átomos se han marcado con líneas rojas.

Comentarios sobre algunos mutex (numerados en la imagen):

1. Como las acciones $c_b_m(A)$ y $c_b_p(C, B)$ tienen como precondition y efecto negativo el átomo $b_c(N)$, son mutex
2. La acción $c_b_m(A)$ es mutex con la acción de persistencia que mantiene el átomo $b_e(M, A)$, ya que el efecto negativo $b_e(M, A)$ de la primera es efecto positivo de la segunda.

3. Como las dos únicas acciones que dan lugar a la aparición de $b_e(M, A)$ y $b_c(A)$ en la capa P_1 son, respectivamente, la correspondiente acción de persistencia y la acción $c_b_m(A)$ y éstas son mutex (ver comentario anterior), los átomos mencionados son mutex.
4. Como las dos únicas acciones que dan lugar a la aparición de los átomos $b_c(A)$ y $b_c(C)$ (ó $b_e(B, N)$) en la capa P_1 son, respectivamente, la acción $c_b_m(A)$ y $c_b_p(C, B)$, y éstas son mutex (ver un comentario anterior), los átomos mencionados son mutex.
5. La acción $p_b_p(C, A)$ y la acción de persistencia del átomo $b_e(M, A)$ (ambas en la capa A_2) no son mutex ya que ninguna de las precondiciones de la primera y $b_e(M, A)$ son mutex, y ninguno de los efectos negativos de la primera es $b_e(M, A)$.
El mismo razonamiento se tiene para la acción $p_b_p(C, A)$ y la acción de persistencia del átomo $b_e(M, B)$
6. Como entre las acciones que dan lugar a la aparición de los átomos $b_e(A, C)$ y $b_e(M, A)$ en la capa P_2 están, respectivamente, la acción $p_b_p(C, A)$ y la acción de persistencia del segundo, y estas no son mutex (ver comentario anterior), los átomos mencionados no son mutex.
El mismo razonamiento se tiene para los átomos $b_e(A, C)$ y $b_e(M, B)$.
7. Como las acciones $p_b_p(A, C)$ y $p_b_m(C)$ tienen como precondición, respectivamente, los átomos $b_c(A)$ y $b_c(C)$ que son mutex en P_1 (ver comentario anterior); las acciones mencionadas son mutex en A_2
8. Como las dos únicas acciones que dan lugar a la aparición de los átomos $b_e(C, A)$ y $b_e(M, C)$ en la capa P_2 son, respectivamente, la acción $p_b_p(A, C)$ y $p_b_m(C)$, y éstas son mutex (ver comentario anterior), los átomos mencionados son mutex.
9. La acción $c_b_m(A)$ y la acción $p_b_m(C)$ son mutex, ya que el efecto negativo $b_c(N)$ de la primera es efecto positivo de la segunda.
10. Como las acciones $c_b_m(A)$ y la acción de persistencia del átomo $b_c(A)$ tienen como precondición, respectivamente, los átomos $b_e(M, A)$ y $b_c(A)$ que son mutex en P_1 (ver un comentario anterior); las acciones mencionadas son mutex en A_2
11. Los átomos $b_c(A)$ y $b_e(M, C)$ son mutex ya que los dos pares de acciones, $c_b_m(A)$ y $p_b_m(C)$, y la acción de persistencia del átomo $b_c(A)$ y $p_b_m(C)$, son mutex (ver los dos comentarios anteriores). Dichos pares de acciones son los únicos que dan lugar a los átomos mencionados.

Entre otras, la acción $p_b_p(A, B)$ no se añade a la capa A_2 ya que, aunque sus precondiciones $b_c(A)$ y $b_e(B, N)$ están en la capa P_1 , éstas son mutex.