



Universidad
Carlos III de Madrid

Grado en Ingeniería Informática

Curso 2020/2021

Ingeniería del Conocimiento

Dominios de Planificación

Autores:

Alba Reinders Sánchez	100383444
Alejandro Valverde Mahou	100383383

Grupo 83	Leganés
----------	---------

Índice

1. Estudio de los dominios	3
1.1. Blocks World	3
1.2. Logistics	3
1.3. Rover	4
1.4. Satellite	6
1.5. Zeno Travel	7
2. Comparación de combustibles Satellite vs Zeno Travel	7
3. Experimentos	8
3.1. Blocks World	8
3.2. Logistics	8
3.3. Rover	8
3.4. Satellite	9
3.5. Zeno Travel	9

1. Estudio de los dominios

Se va a describir cada dominio brevemente y explicar sus predicados y acciones (precondiciones y efectos).

1.1. Blocks World

En este dominio tan solo se encuentra el tipo 'bloque'. Los **predicados** que tiene son:

- *(on ?x - block ?y - block)*: representa que el bloque x está sobre el bloque y.
- *(ontable ?x - block)*: representa que el bloque x está encima de la mesa.
- *(clear ?x - block)*: representa que el bloque x no tiene ningún bloque encima.
- *(handempty)*: representa que el gancho está libre.
- *(holding ?x - block)*: representa que el gancho sujeta al bloque x.

Sus **acciones** son:

- Pick-up:
Precondiciones: el bloque no tiene nada encima y está encima de la mesa y el gancho está libre.
Efectos: el bloque deja de estar encima de la mesa y el gancho pasa a sujetar el bloque.
- Put-down:
Precondiciones: el gancho sujeta el bloque.
Efectos: el bloque pasa a estar encima de la mesa, con nada encima, y el gancho pasa a estar libre.
- Stack:
Precondiciones: el gancho sujeta el bloque 1 y el bloque 2 no tiene nada encima.
Efectos: el bloque 2 pasa a tener encima el bloque 1 y el gancho pasa a estar libre.
- Unstack:
Precondiciones: el bloque 1 está sobre el bloque 2, el bloque 1 no tiene nada encima y el gancho está libre.
Efectos: el bloque 2 pasa a no tener nada encima y el gancho pasa a sujetar el bloque 1.

1.2. Logistics

En este dominio hay los siguientes tipos: 'camión', 'avión' (vehículo), 'paquete', 'vehículo' (objeto físico), 'aeropuerto', 'localización' (lugar), 'ciudad', 'lugar' y 'objeto físico' (objeto). Los **predicados** que tiene son:

- *(in-city ?loc - place ?city - city)*: representa que un lugar se encuentra en una ciudad.
- *(at ?obj - physobj ?loc - place)*: representa que un objeto físico está en un lugar.
- *(in ?pkg - package ?veh - vehicle)*: representa que un paquete está en un vehículo.

Sus **acciones** son:

- Load-truck:
Precondiciones: tener un paquete y un camión en el mismo lugar.
Efectos: se elimina el paquete del lugar y se mete dentro del camión.
- Load-airplane:
Precondiciones: tener un paquete y un avión en el mismo lugar.
Efectos: se elimina el paquete del lugar y se mete dentro del avión.
- Unload-truck:
Precondiciones: tener un paquete en un camión y el camión en un lugar.
Efectos: se elimina el paquete del camión y el paquete pasa a estar en el lugar.

- **Unload-airplane:**
Precondiciones: tener un paquete en un avión y el avión en un lugar.
Efectos: se elimina el paquete del avión y el paquete pasa a estar en el lugar.
- **Drive-truck:**
Precondiciones: tener un camión en el lugar 1, y que el lugar 1 y el lugar 2 estén en la misma ciudad.
Efectos: El camión deja de estar en lugar 1 y pasa a estar en el lugar 2.
- **Fly-airplane:**
Precondiciones: tener un avión en el lugar 1.
Efectos: el avión deja de estar en el lugar 1 y pasa a estar en el lugar 2.

1.3. Rover

En este dominio hay los siguientes tipos: 'rover', 'punto de ruta', 'almacenamiento', 'cámara', 'modo', 'aterrizador', y 'objetivo'. Los **predicados** que tiene son:

- *(at ?x - rover ?y - waypoint)*: representa que el rover se encuentra en un punto de ruta.
- *(at_lander ?x - lander ?y - waypoint)*: representa que el aterrizador se encuentra en un punto de ruta.
- *(can_traverse ?r - rover ?x - waypoint ?y - waypoint)*: representa que el rover puede pasar del punto 1 al punto 2.
- *(equipped_for_soil_analysis ?r - rover)*: representa que el rover está equipado para analizar sustrato.
- *(equipped_for_rock_analysis ?r - rover)*: representa que el rover está equipado para analizar rocas.
- *(equipped_for_imaging ?r - rover)*: representa que el rover está equipado para realizar imágenes.
- *(empty ?s - store)*: representa que el almacenamiento se encuentra vacío
- *(have_rock_analysis ?r - rover ?w - waypoint)*: representa que el rover posee información de las rocas de un punto de ruta.
- *(have_soil_analysis ?r - rover ?w - waypoint)*: representa que el rover posee información de los sustratos de un punto de ruta.
- *(full ?s - store)*: representa que el almacenamiento se encuentra lleno.
- *(calibrated ?c - camera ?r - rover)*: representa que la cámara del rover está calibrada.
- *(supports ?c - camera ?m - mode)*: representa que la cámara acepta el modo de toma de imagen.
- *(available ?r - rover)*: representa que el rover se encuentra disponible.
- *(visible ?w - waypoint ?p - waypoint)*: representa que el punto de ruta 2 es visible desde el punto de ruta 1.
- *(have_image ?r - rover ?o - objective ?m - mode)*: representa que el rover posee una foto tomada con un modo de un objetivo.
- *(communicated_soil_data ?w - waypoint)*: representa que la información del sustrato de un punto de ruta ha sido comunicada.
- *(communicated_rock_data ?w - waypoint)*: representa que la información de las rocas de un punto de ruta ha sido comunicada.
- *(communicated_image_data ?o - objective ?m - mode)*: representa que la información de la imagen tomada con un modo del objetivo ha sido comunicada.
- *(at_soil_sample ?w - waypoint)*: representa que un punto de ruta tiene muestras de sustratos.

- (*at_rock_sample ?w - waypoint*): representa que un punto de ruta tiene muestras de rocas.
- (*visible_from ?o - objective ?w - waypoint*): representa que el objetivo es visible desde el punto de ruta.
- (*store_of ?s - store ?r - rover*): representa que el almacenamiento pertenece al rover.
- (*calibration_target ?i - camera ?o - objective*): representa que la cámara está calibrada sobre el objetivo.
- (*on_board ?i - camera ?r - rover*): representa que el rover posee la cámara.
- (*channel_free ?l - lander*): representa que el canal de comunicación del aterrizador está disponible.
- (*in_sun ?w - waypoint*): representa que el punto de ruta es un punto de recarga.

Sus **acciones** son:

- **Navigate:**
Precondiciones: el rover está disponible, con más de 8 de energía, y se encuentra en el punto de ruta 1, y se puede viajar al punto de ruta 2.
Efectos: el rover deja de estar en el punto de ruta 1 y pasa a estar en el 2, y su energía decrementa.
- **Recharge:**
Precondiciones: el rover está en un punto de recarga y tiene de 80 o menos de energía.
Efectos: aumenta la energía del rover en 20 y aumenta el número de recargas hechas.
- **Sample_soil:**
Precondiciones: si el rover está equipado para tomar muestras de sustratos, su energía es mayor o igual a 3, está vacío y se encuentra sobre muestras de sustrato.
Efectos: el rover pasa a estar lleno con muestras de sustrato, y las muestras dejan de estar en el punto de ruta. La energía del rover decrece en 3 unidades.
- **Sample_rock:**
Precondiciones: si el rover está equipado para tomar muestras de rocas, su energía es mayor o igual a 5, está vacío y se encuentra sobre muestras de rocas.
Efectos: el rover pasa a estar lleno con muestras de rocas, y las muestras dejan de estar en el punto de ruta. La energía del rover decrece en 5 unidades.
- **Drop:**
Precondiciones: el almacenamiento del rover está lleno.
Efectos: el almacenamiento deja de estar lleno y pasa a estar vacío.
- **Calibrate:**
Precondiciones: el rover está equipado para hacer fotografías, tiene una cámara encima y se encuentra en un punto de ruta, su energía es mayor o igual a 2, hay un objetivo que fotografiar que es visible desde el punto de ruta.
Efectos: disminuye la energía en 2 y se calibra la cámara del rover.
- **Take_image:**
Precondiciones: la cámara del rover está calibrada, está equipado para hacer fotografías, y la cámara acepta el modo de fotografía, y el objetivo es visible desde la posición del rover, y su energías es mayor o igual a 1.
Efectos: se consigue una imagen del objetivo, la cámara deja de estar calibrada y se reduce la energía en 1 unidad.
- **Communicate_soil_data:**
Precondiciones: el rover está en el punto de ruta 1 y está disponible, el aterrizador está en el punto de ruta 2 y libre, el rover posee información de los sustratos del punto de ruta 3 y su energía es mayor o igual a 4 y desde el punto de ruta 1 es visible el punto de ruta 2.
Efectos: el rover deja de estar disponible y el aterrizador deja de estar libre, después vuelve a estar libre, se comunica la información del sustrato del punto de ruta 3 y el rover vuelve a estar disponible y su energía disminuye en 4.

- **Communicate_rock_data:**
Precondiciones: el rover está en el punto de ruta 1 y está disponible, el aterrizador está en el punto de ruta 2 y libre, el rover posee información de las rocas del punto de ruta 3 y su energía es mayor o igual a 4 y desde el punto de ruta 1 es visible el punto de ruta 2.
Efectos: el rover deja de estar disponible y el aterrizador deja de estar libre, después vuelve a estar libre, se comunica la información de las rocas del punto de ruta 3 y el rover vuelve a estar disponible y su energía disminuye en 4.
- **Communicate_image_data:**
Precondiciones: el rover está en el punto de ruta 1 y está disponible, el aterrizador está en el punto de ruta 2 y libre, el rover posee una foto tomada con un modo de un objetivo y su energía es mayor o igual a 6 y desde el punto de ruta 1 es visible el punto de ruta 2.
Efectos: el rover deja de estar disponible y el aterrizador deja de estar libre, después vuelve a estar libre, se comunica la información de la imagen tomada con un modo del objetivo y el rover vuelve a estar disponible y su energía disminuye en 6.

1.4. Satellite

En este dominio hay los siguientes tipos: 'satélite', 'dirección', 'instrumento' y 'modo'. Los **predicados** que tiene son:

- *(on_board ?i - instrument ?s - satellite)*: representa que el instrumento se encuentra en el satélite.
- *(supports ?i - instrument ?m - mode)*: representa que el instrumento acepta el mode.
- *(pointing ?s - satellite ?d - direction)*: representa que el satélite está mirando en la dirección.
- *(power_avail ?s - satellite)*: representa que el satélite tiene energía.
- *(power_on ?i - instrument)*: representa que el instrumento está encendido.
- *(calibrated ?i - instrument)*: representa que el instrumento está calibrado.
- *(have_image ?d - direction ?m - mode)*: representa que se tiene una imagen con el mode en la dirección.
- *(calibration_target ?i - instrument ?d - direction)*: representa que el instrumento está calibrado en la dirección.

Sus **acciones** son:

- **Turn_to:**
Precondiciones: el satélite está mirando a la dirección 1, hay una dirección 2 diferente a la 1, tiene al menos el combustible necesario para realizar el cambio de dirección.
Efectos: el satélite está mirando a la dirección 2 y ya no está mirando a la dirección 1, disminuye su combustible tanto como haya necesitado para hacer el cambio de dirección y se contabiliza el consumo de este combustible.
- **Switch_on:**
Precondiciones: el instrumento se encuentra en satélite y tiene energía.
Efectos: el instrumento pasa a estar encendido, sin calibrar y sin posibilidad de recibir más energía.
- **Switch_off:**
Precondiciones: el instrumento se encuentra en el satélite, y está encendido
Efectos: el instrumento pasa a estar apagado, y se vuelve a permitir que reciba energía.
- **Calibrate:**
Precondiciones: el instrumento está calibrado, encendido y se encuentra en el satélite y el satélite mira en la misma dirección que el instrumento.
Efectos: el instrumento está calibrado.

- Take_image:

Precondiciones: el instrumento se encuentra en el satélite, está calibrado, permite usar el modo, está encendido, el satélite está apuntando en la dirección, y todavía tiene capacidad de almacenaje.

Efectos: se disminuye la capacidad de almacenaje, se obtienen los datos y la imagen de la dirección con el modo.

1.5. Zeno Travel

En este dominio hay los siguientes tipos: 'aeronave', 'persona', 'ciudad' y 'nivel de combustible' (objeto). Los **predicados** que tiene son:

- (at ?x - (either person aircraft) ?c - city): representa que o una persona o una aeronave se encuentran en una ciudad.
- (in ?p - person ?a - aircraft): representa que la persona se encuentra en una aeronave.
- (fuel-level ?a - aircraft ?l - flevel): representa el nivel de combustible en una aeronave.
- (next ?l1 ?l2 - flevel): representa el cambio entre el nivel de combustible 1 y el nivel de combustible 2.

Sus **acciones** son:

- Board:

Precondiciones: la persona y la aeronave se encuentran en la misma ciudad.

Efectos: la persona deja de estar en la ciudad y se encuentra en la aeronave.

- Debark:

Precondiciones: la persona se encuentra en la aeronave y la aeronave está en una ciudad.

Efectos: la persona deja de estar en la aeronave y pasa a estar en la ciudad.

- Fly:

Precondiciones: la aeronave se encuentra en la ciudad 1 y su nivel de combustible puede pasar al anterior.

Efectos: la aeronave deja de estar en la ciudad 1 y pasa a estar en la ciudad 2, su nivel de combustible deja de ser el 1 y pasa a ser el 2.

- Zoom:

Precondiciones: la aeronave se encuentra en la ciudad 1, y existen 2 niveles anteriores de combustible.

Efectos: la aeronave pasa a tener el tercer nivel de combustible, y pasa a estar en la ciudad 2.

- Refuel:

Precondiciones: la aeronave se encuentra en la ciudad, y el nivel de combustible puede pasar al siguiente.

Efectos: el nivel de combustible de la aeronave pasa a ser el siguiente.

2. Comparación de combustibles Satellite vs Zeno Travel

A continuación se va a comparar cómo modelan la cantidad de combustible utilizado en los dominios de Satellite y Zeno Travel.

En **Satellite** se utiliza un valor numérico para representar la cantidad de combustible que tiene cada satélite y que disminuye su cantidad dependiendo del movimiento que realice.

En **Zeno Travel** se utilizan niveles de combustible de forma que un viaje normal consume un nivel, un viaje 'zoom' consume dos niveles y repostar recupera un nivel.

Estos dos dominios usan de forma distinta el mismo recurso (combustible) pero adaptado a las necesidades de cada dominio.

3. Experimentos

3.1. Blocks World

Con este dominio se intenta colocar un número de bloques en ciertas posiciones, que pueden ser o bien sobre la mesa o bien sobre otro bloque.

Problema	Tiempo	Longitud	Calidad	Nº objetos	Nº metas
<i>probBLOCKS-4-0</i>	0.00	5	5	4	3
<i>probBLOCKS-8-0</i>	0.00	25	25	8	7
<i>probBLOCKS-9-0</i>	0.20	47	47	9	8
<i>probBLOCKS-12-0</i>	0.77	69	69	12	11
<i>probBLOCKS-13-0</i>	0.73	85	85	13	12

Nuevo problema

Se tiene un total de 10 bloques, todos apilados uno encima de otro. El objetivo es que el bloque que se encuentra debajo del resto bloques, el que está encima de la mesa, pase a estar encima del bloque que se encuentra en la cima de la pila.

Problema	Tiempo	Longitud	Calidad	Nº objetos	Nº metas
<i>nuevo problema</i>	0.00	19	19	10	1

3.2. Logistics

Este dominio tiene como objetivo llevar paquetes de un lugar a otro. El transporte se puede hacer por camión o por avión.

Problema	Tiempo	Longitud	Calidad	Nº objetos	Nº metas
<i>probLOGISTICS-4-0</i>	0.00	19	19	15	4
<i>probLOGISTICS-9-0</i>	0.00	38	38	22	9
<i>probLOGISTICS-13-1</i>	0.06	67	67	37	13
<i>probLOGISTICS-14-1</i>	0.07	78	78	35	14
<i>probLOGISTICS-15-1</i>	0.09	73	73	37	15

Nuevo problema

Se tiene un único avión, 5 aeropuertos, 5 ciudades, 2 localizaciones en la ciudad 1, un camión y 5 paquetes repartidos cada uno en las ciudades. Inicialmente el avión está en la ciudad 5 y el camión en la ciudad 1. El objetivo es que el paquete 1 esté en la ciudad 4, el paquete 2 en la ciudad 3, el paquete 3 en la ciudad 1, el paquete 4 en la ciudad 5 y el paquete 5 en la ciudad 2.

Problema	Tiempo	Longitud	Calidad	Nº objetos	Nº metas
<i>nuevo problema</i>	0.00	18	18	19	5

3.3. Rover

El dominio tiene como objetivo comunicar la información de los datos recogidos por el rover de sustratos, rocas o imágenes en distintos puntos de ruta. Se intenta minimizar el combustible empleado.

Problema	Tiempo	Longitud	Calidad	Nº objetos	Nº metas
<i>pfile01</i>	7.17	14	0	13	3
<i>pfile10</i>	-	-	-	29	11
<i>pfile13</i>	-	-	-	30	12
<i>pfile15</i>	-	-	-	32	10
<i>pfile19</i>	-	-	-	51	17

A partir del problema *pfile13* se ejecuta sin el -O en el comando porque en caso contrario daba error.

Nuevo problema

Se tienen 2 aterrizadores, un sólo rover con su almacenamiento, 3 modos, 6 puntos de ruta, una cámara y 3 objetivos. La meta es conseguir una imagen en alta resolución del objetivo 0, del objetivo 1 y del objetivo 2.

Problema	Tiempo	Longitud	Calidad	Nº objetos	Nº metas
<i>nuevo problema</i>	1.15	12	0	17	3

3.4. Satellite

Con el dominio se intenta tomar distintas imágenes de diferentes objetivos, minimizando el consumo del combustible.

Problema	Tiempo	Longitud	Calidad	Nº objetos	Nº metas
<i>pfile01</i>	0.03	10	109.88	12	3
<i>pfile05</i>	-	-	-	25	8
<i>pfile10</i>	0.08	43	634.66	38	12
<i>pfile15</i>	74.88	71	806.31	57	24
<i>pfile18</i>	-	-	-	47	13

A partir del problema *pfile10* se ejecuta sin el -O en el comando porque en caso contrario daba error.

Nuevo problema

Se tiene un satélite con una capacidad de datos de 1200 y un sólo instrumento con 3 modos y 7 direcciones. El objetivo es que el satélite tome las imágenes de 4 direcciones con el primer modo.

Problema	Tiempo	Longitud	Calidad	Nº objetos	Nº metas
<i>nuevo problema</i>	0.07	11	108.59	12	4

3.5. Zeno Travel

Con el dominio se intenta llevar distintos pasajeros a distintas ciudades, usando aeronaves. También puede ser llevar ciertas aeronaves a ciertas ciudades.

Problema	Tiempo	Longitud	Calidad	Nº objetos	Nº metas
<i>pfile05</i>	0.01	10	10	17	4
<i>pfile10</i>	0.04	26	26	23	9
<i>pfile15</i>	11.40	43	43	38	14
<i>pfile17</i>	14.16	79	79	48	23
<i>pfile20</i>	108.95	104	104	58	24

Nuevo problema

Se tiene 2 aeronaves, 7 personas, 4 ciudades y 9 niveles de combustible. La primera aeronave se encuentra en la ciudad 1 con un nivel de combustible de 7 y la segunda en la ciudad 2 con un combustible de 2. Las personas están dispersas por las ciudades pero no de forma equitativa. El objetivo es llevar a la persona 1 y 5 a la ciudad 2, a las personas 2, 3 y 4 a la ciudad 3. A la persona 6 a la ciudad 1 y a la persona 7 a la 0. Además se quiere que el nivel de combustible de la aeronave 1 sea 3 y de la aeronave 2 sea 5.

Problema	Tiempo	Longitud	Calidad	Nº objetos	Nº metas
<i>nuevo problema</i>	0.04	28	28	23	9