



UNIVERSIDAD DE GRANADA

Práctica 2: planificación clásica en PDDL

Juan Ocaña Valenzuela

5 de mayo de 2019

Índice

1. Diario de trabajo	3
2. Leyenda	4
3. Ejercicios	4
3.1. Ejercicio 1	4
3.1.1. Apartado A	4
3.1.2. Apartado B	4
3.1.3. Apartado C	5
3.1.4. Apartado D	6

1. Diario de trabajo

Domingo, 5 de mayo de 2019

Empecé la práctica hace unos días, pero no ha sido hasta hoy que he decidido comenzar a documentarla. Llevaba unos días bastante ocupado (la fiesta de la Escuela no se organiza sola), y tenía visita en casa, así que dedicaba relativamente poco espacio de mi sistema nervioso central a pensar en los apasionantes mundos de Belkan, más allá de la eterna duda: ¿es Belkan el personaje? ¿es el nombre del mundo? ¿o quizá un ser omnipresente que gobierna por aquellos lares?

Una vez mi visita se marchó, comencé a plantear el primer ejercicio de la relación, diseñando un mapa, poniendo cosas en él, definiendo las acciones y todo eso. Una vez creí haber terminado el modesto problema 1, en el cual cada personaje debe tener un objeto, ejecuté FF y...

no funcionaba.

No encontraba un plan. No lo encontraba y no sabía por qué. Estaba cansado y no me apetecía hacer nada, así que mandé a la mierda los estafalarios mundos de Belkan y me puse a jugar a Persona 5.

Hoy es un día nuevo, y nada más levantarme y desayunar he pensado que quizás podría hacer un pequeño esfuerzo por saber qué le pasaba a mi problema, y cómo arreglarlo. Y es que nuestro pequeño bicho no puede entregar objetos a los personajes si no lo hemos definido en el problema. Qué cosas, eh.

Bautizado como Patrick Dotimas (Patrick para los amigos), ahora que está en el problema encuentra una solución. Ya tenemos un problema resuelto... O no.

Observando el plan, resulta que, en un acto de amor incondicional, Patrick se entrega a sí mismo a los personajes, y no los objetos que coge. En lugar de algo como:

GIVE PATRICK ROSE R20 PRINCESS

tenemos algo así:

GIVE PATRICK PATRICK R20 PRINCESS

A ver cómo lo soluciono... De momento voy a comer.

[Actualización - 19:30]

Después de procrastinar un buen rato (y jugar otro rato a Persona 5), me he dado cuenta de que el dominio no estaba mal del todo. Patrick cogía sus objetos y los entregaba bien, pero no sabía qué objeto tenía ni daba. Lo he solucionado añadiendo un predicado más, *on hand*, que viene explicado en su sitio.

Acabo de caer en una cosa... Si Belkan es el personaje, y nuestro jugador se llama Patrick, ¿son estos los extraños mundos de Patrick? En fin, voy a ponerme con el generador de problemas, a ver si marcha.

Continuará...

2. Leyenda

TO DO.

3. Ejercicios

3.1. Ejercicio 1

Definir un dominio y problema de planificación considerando que el jugador podrá estar orientado al norte, sur, este u oeste y desplazarse de una zona a otra siempre que esté correctamente orientado. Por ejemplo, podrá desplazarse a una zona al norte de su zona actual, si está orientado al norte.

3.1.1. Apartado A

Representar en el dominio los objetos del mundo.

Para representar los objetos del mundo (jugador, personajes, objetos, habitaciones, caminos, etc.) se han establecido los siguientes tipos, con su jerarquía:

locatable: elemento que se puede localizar en una posición.

character: personaje.

player: jugador.

npc: personaje no jugador.

object: objeto.

orientation: orientación de un elemento.

room: habitación del dominio.

3.1.2. Apartado B

Representar predicados que permitan describir los estados del mundo.

Se han considerado los siguientes predicados:

at

(at ?r - room ?l - locatable)

Un elemento ?l se encuentra en la habitación ?r.

on_floor

(on_floor ?o - object)

Un objeto ?o se encuentra en el suelo.

compass

(compass ?o - orientation)

La orientación del personaje es ?o.

path

(path ?r1 ?r2 - room ?o - orientation)

Existe un camino entre ?r1 y ?r2, en el que la segunda habitación se encuentra con una orientación ?o respecto de la primera.

has_object

(has_object ?c - character)

Un personaje ?c tiene un objeto.

on_hand

(on_hand ?o - object)

El jugador tiene un objeto ?o en la mano.

3.1.3. Apartado C

Representar las siguientes acciones del jugador: girar a la izquierda, girar a la derecha, ir, coger, dejar, entregar.

Girar a la izquierda

(:action TURN_LEFT)

Dada una orientación, el jugador mirará a aquella a su izquierda:

- $N \rightarrow W$
- $W \rightarrow S$
- $S \rightarrow E$
- $E \rightarrow N$

Girar a la derecha

(:action TURN_RIGHT)

Dada una orientación, el jugador mirará a aquella a su derecha:

- $N \rightarrow E$
- $W \rightarrow N$
- $S \rightarrow W$

- $E \rightarrow S$

Girar 180 grados

(:action TURN_180)

Dada una orientación, el jugador mirará a aquella a su espalda:

- $N \rightarrow S$
- $W \rightarrow E$
- $S \rightarrow N$
- $E \rightarrow W$

Coger un objeto

(:action PICK)

Si el jugador se encuentra en la misma habitación que un objeto y este se halla en el suelo, el jugador podrá cogerlo. El objeto dejará de estar en la habitación y en el suelo.

Soltar un objeto

(:action DROP)

Si el jugador tiene un objeto, lo soltará. El objeto pasará a estar en la misma habitación que el jugador, y en el suelo.

Dar un objeto a un personaje

(:action GIVE)

Si el jugador tiene un objeto, el npc no, y ambos se encuentran en la misma habitación, el jugador le dará el objeto al npc. El jugador pasará a no tener objeto, y el npc sí lo hará.

Moverse a una habitación contigua

(:action GO)

El jugador se moverá a la habitación hacia la que esté orientado.

3.1.4. Apartado D

Plantear un problema de planificación con un estado inicial con 25 zonas conectadas arbitrariamente en el que aparezcan situados los 5 personajes en distintas zonas y al menos 5 objetos. El objetivo de este problema consistirá en conseguir que todos los personajes tengan al menos un objeto.

El mapa planteado en este problema es el siguiente:

TO DO

El objetivo es que todos los personajes dispongan de un objeto, es decir, que se cumpla:

```
(AND
  (has_object leonardo)
  (has_object prince)
  (has_object princess)
  (has_object professor)
  (has_object witch)
)
```

El plan generado por FF es el siguiente:

0: TURN_180 N 1: GO PATRICK R13 R18 S	21: GO PATRICK R20 R15 N	41: GO PATRICK R14 R13 W
2: TURN_180 S	22: TURN_180 N	42: GO PATRICK R13 R12 W
3: PICK PATRICK APPLE R18	23: PICK PATRICK GOLD R15	43: GO PATRICK R12 R11 W
4: TURN_LEFT N	24: GO PATRICK R15 R20 S	44: TURN_RIGHT W
5: GO PATRICK R18 R17 W	25: TURN_180 S	45: TURN_RIGHT N
6: GO PATRICK R17 R16 W	26: TURN_LEFT N	46: GIVE PATRICK ROSE R11 PRINCESS
7: TURN_RIGHT W	27: GO PATRICK R20 R19 W	47: GO PATRICK R11 R12 E
8: TURN_RIGHT N	28: TURN_180 W	48: TURN_LEFT E
9: GO PATRICK R16 R17 E	29: GIVE PATRICK GOLD R19 PROFESSOR	49: GO PATRICK R12 R7 N
10: TURN_RIGHT E	30: GO PATRICK R19 R20 E	50: GO PATRICK R7 R2 N
11: GO PATRICK R17 R22 S	31: TURN_RIGHT E	51: TURN_LEFT N
12: TURN_LEFT S	32: GO PATRICK R20 R25 S	52: GO PATRICK R2 R1 W
13: GIVE PATRICK APPLE R22 WITCH	33: TURN_180 S	53: TURN_LEFT W
14: GO PATRICK R22 R23 E	34: PICK PATRICK ROSE R25	54: PICK PATRICK OSCAR R1
15: GO PATRICK R23 R24 E	35: GO PATRICK R25 R20 N	55: GO PATRICK R1 R6 S
16: TURN_LEFT E	36: TURN_LEFT N	56: TURN_LEFT S
17: GO PATRICK R24 R19 N	37: GO PATRICK R20 R19 W	57: GO PATRICK R6 R7 E
18: TURN_RIGHT N	38: TURN_RIGHT W	58: TURN_RIGHT E
19: GO PATRICK R19 R20 E	39: GO PATRICK R19 R14 N	59: GIVE PATRICK OSCAR R7 LEONARDO
20: TURN_LEFT E	40: TURN_LEFT N	60: GO PATRICK R7 R12 S
		61: TURN_LEFT S

62: GO PATRICK R12 R13 E	67: TURN_LEFT E	73: GO PATRICK R3 R4 E
63: TURN_LEFT E	68: GO PATRICK R9 R4 N	74: TURN_RIGHT E
64: GO PATRICK R13 R8 N	69: TURN_LEFT N	75: GO PATRICK R4 R9 S
65: TURN_RIGHT N	70: GO PATRICK R4 R3 W	76: GIVE PATRICK ALGO-
66: GO PATRICK R8 R9 E	71: TURN_180 W	RITHM R9 PRINCE
	72: PICK PATRICK ALGO-	
	RITHM R3	