



UNIVERSIDAD DE GRANADA

GRADO INGENIERÍA INFORMÁTICA (2016 – 2017)

---

# TÉCNICAS SISTEMAS INTELIGENTES

Práctica 2 | Planificador FF-Metric

Trabajo realizado por: Antonio Miguel Morillo Chica

---

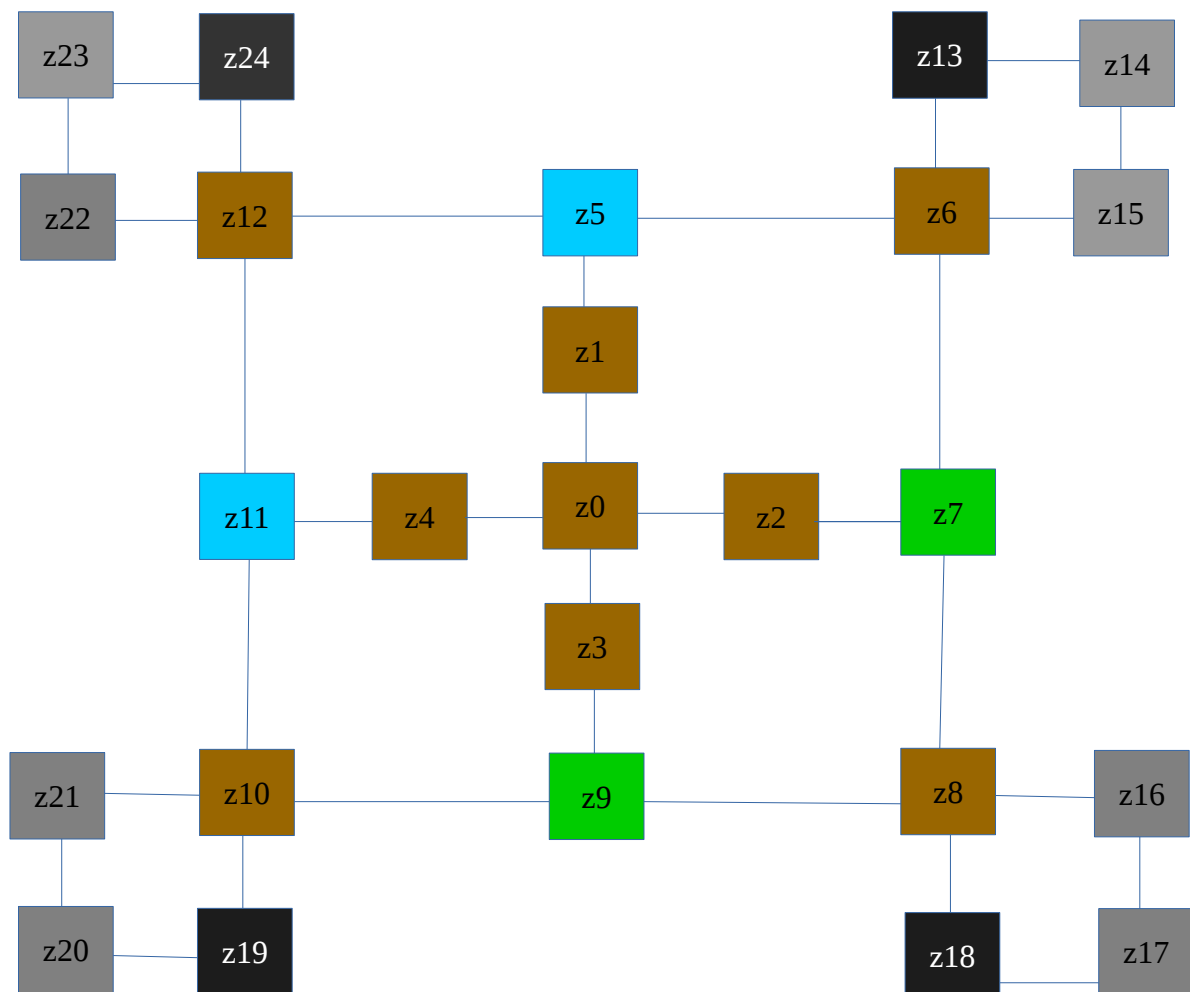
## 1. Introducción.

En esta practica usaremos el lenguaje pddl y el planificador global FF-Metic, un planificador diseñado para disparar reglas.

Supondremos que para definir el problema nos encontramos en la hipotesis del mundo cerrado. Todo lo que afirmemos será verdad y en caso de no hacerlo, falso.

## 2. Problema sobre el que se aplica el dominio.

Para todos los problemas realizo las pruebas en el mismo tipo de distribución de zonas-terreno que es la siguiente. Los objetos si los he distribuido distinto. Gis: piedra, marrón: tierra, azul: agua, negro: precipicios.



### 3. Ejercicios realizados.

La practica estaba compuesta de 6 ejercicios todos relativos al mismo tipo de problema basado en los mundos de belkan.

- **Ejercicio 1:** Definir el dominio y problema 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.

Este ejercicio ha sido muy simple una vez puestas las zonas para saber la orientación he jugado con dos elementos, un predicado que indica en todo momento la orientación actual del jugador, es como una brujula, dos predicados más llamados sig-giro-izq / sig- giro-der con dos orientaciones, con la que se pretende girar, la actual, y el estado resultante. Esto lo definiremos los objetos del problema. Por último la orientación está presente en las “conexiones” entre zonas, así una zona z1 está conectada con z2 con una orientación o1 y z2 está conectada con z1 con orientación o2 siendo o1 y o2 orientaciones opuestas.

```
step    0: PASAR MKXV Z0 Z1 NORTE
        1: GIRAR-DER MKXV NORTE ESTE
        2: COGER-OBJETO MKXV ROSA Z1
        3: GIRAR-DER MKXV ESTE SUR
        4: PASAR MKXV Z1 Z0 SUR
        5: PASAR MKXV Z0 Z3 SUR
        6: GIRAR-IZQ MKXV SUR ESTE
        7: GIRAR-IZQ MKXV ESTE NORTE
        8: DAR-OBJETO MKXV ROSA Z3 PRINCESA
        9: COGER-OBJETO MKXV MANZANA Z3
       10: PASAR MKXV Z3 Z0 NORTE
       11: PASAR MKXV Z0 Z1 NORTE
       12: GIRAR-IZQ MKXV NORTE OESTE
       13: DAR-OBJETO MKXV MANZANA Z1 BRUJA
       14: GIRAR-IZQ MKXV OESTE SUR
       15: PASAR MKXV Z1 Z0 SUR
       16: COGER-OBJETO MKXV ALGORITMO Z0
       17: DAR-OBJETO MKXV ALGORITMO Z0 PROFESOR
       18: GIRAR-DER MKXV SUR OESTE
       19: PASAR MKXV Z0 Z4 OESTE
       20: GIRAR-DER MKXV OESTE NORTE
       21: GIRAR-DER MKXV NORTE ESTE
       22: COGER-OBJETO MKXV ORO Z4
       23: PASAR MKXV Z4 Z0 ESTE
       24: PASAR MKXV Z0 Z2 ESTE
       25: DAR-OBJETO MKXV ORO Z2 PRINCIPE
       26: COGER-OBJETO MKXV OSCAR Z2
       27: GIRAR-DER MKXV ESTE SUR
       28: GIRAR-DER MKXV SUR OESTE
       29: PASAR MKXV Z2 Z0 OESTE
       30: PASAR MKXV Z0 Z4 OESTE
       31: DAR-OBJETO MKXV OSCAR Z4 LEONARDO
          Coste Total: 0.00

time spent:  0.00 seconds instantiating 947 easy, 0 hard action templates
            0.00 seconds reachability analysis, yielding 190 facts and 317 actions
            0.00 seconds creating final representation with 165 relevant facts, 0 relevant fluents
            0.00 seconds computing LNF
            0.00 seconds building connectivity graph
            0.46 seconds searching, evaluating 7829 states, to a max depth of 0
            0.46 seconds total time

mkxv@l00per:~/Documentos/TSI/Sesion 7/belkan/Metric-FFS
```

Además para el problema he descrito un predicado, a-entregar, para saber los objetos que he de entregar a cada personaje y saber si lo tengo o no en la mano. Esta idea va cambiando a lo largo los de los ejercicios como otras.

Las ejecución tanto con la optimización normal como la -g 1 y -h 1 es muy rápida, 0 seg y 0,46 respectivamente.

- **Ejercicio 2:** Considerar que la acción de desplazamiento entre zonas tiene un coste igual a la longitud del camino entre cada zona.

Lo único que he hecho ha sido definir dos funciones, coste coste, para el coste parcial entre zonas y coste-t que es el coste total del camino hasta el momento. En mi caso además del coste entre las zonas he añadido coste a las acciones, un coste muy bajo como es lógico.

```
ff: found legal plan as follows
step    0: COGER-OBJETO MKXV ALGORITMO Z0 PROFESOR
        1: DAR-OBJETO MKXV ALGORITMO Z0 PROFESOR
        2: GIRAR-IZO MKXV NORTE OESTE
        3: GIRAR-IZO MKXV OESTE SUR
        4: PASAR MKXV Z0 Z3 SUR
        5: GIRAR-DER MKXV SUR OESTE
        6: GIRAR-DER MKXV OESTE NORTE
        7: COGER-OBJETO MKXV MANZANA Z3 BRUJA
        8: PASAR MKXV Z3 Z0 NORTE
        9: PASAR MKXV Z0 Z1 NORTE
       10: GIRAR-DER MKXV NORTE ESTE
       11: DAR-OBJETO MKXV MANZANA Z1 BRUJA
       12: COGER-OBJETO MKXV ROSA Z1 PRINCESA
       13: GIRAR-DER MKXV ESTE SUR
       14: PASAR MKXV Z1 Z0 SUR
       15: PASAR MKXV Z0 Z3 SUR
       16: GIRAR-DER MKXV SUR OESTE
       17: DAR-OBJETO MKXV ROSA Z3 PRINCESA
       18: GIRAR-DER MKXV OESTE NORTE
       19: PASAR MKXV Z3 Z0 NORTE
       20: GIRAR-IZO MKXV NORTE OESTE
       21: PASAR MKXV Z0 Z4 OESTE
       22: GIRAR-IZO MKXV OESTE SUR
       23: GIRAR-IZO MKXV SUR ESTE
       24: COGER-OBJETO MKXV ORO Z4 PRINCIPE
       25: PASAR MKXV Z4 Z0 ESTE
       26: PASAR MKXV Z0 Z2 ESTE
       27: DAR-OBJETO MKXV ORO Z2 PRINCIPE
       28: COGER-OBJETO MKXV OSCAR Z2 LEONARDO
       29: GIRAR-DER MKXV ESTE SUR
       30: GIRAR-DER MKXV SUR OESTE
       31: PASAR MKXV Z2 Z0 OESTE
       32: PASAR MKXV Z0 Z4 OESTE
       33: DAR-OBJETO MKXV OSCAR Z4 LEONARDO
          Coste Total: 61.00

time spent:  0.00 seconds instantiating 1947 easy, 0 hard action templates
            0.00 seconds reachability analysis, yielding 190 facts and 797 actions
            0.00 seconds creating final representation with 165 relevant facts, 1 relevant fluents
            0.00 seconds computing LNF
            0.00 seconds building connectivity graph
            9.24 seconds searching, evaluating 40227 states, to a max depth of 0
            9.24 seconds total time

mkxv@l00per:~/Documentos/TSI/Sesion 7/belkan/Metric-FFS$ ./ff -p ../Ejercicio\ 2/ -o dominio.pddl -f problema.pd
dl -o -g 1 -h 1
```

Tanto las optimizaciones -O por defecto con  $g = 1$  y  $h = 5$  y la de -O -g 1 -h 1 funcionan perfectamente ante el problema dado, con tiempos de 12 y 9 seg respectivamente. También hay que decir que el problema expuesto al planificador era bastante simple.

- **Ejercicio 3:** Considerar ahora que (1) hay distintos tipos de zonas dependiendo del tipo de superficie que contengan, en concreto: Bosque, Agua, Precipicio, Arena y Piedra, y (2) hay dos nuevos tipos de objetos: Zapatilla y Bikini. Además, considerar también que el jugador, aparte de poder tener cogido un objeto, está dotado de una mochila donde puede guardar otro objeto (solo uno).

Para este ejercicio he modificado el dominio. Para controlar la mochila he seguido la misma metodología que con la mano, realmente es como tener dos manos. Así que he formados dos predicados para controlar cada cosa, mochila ?j (mochila del jugador, cuando esté este ocupada por un objeto) y tengo-en-mochila ?o (para saber que objeto está presente), misma metodología que la mano.

Para los terrenos he añadido 5 predicados, es-arena, es-piedra, es-agua, es-bosque y es-precipicio. Cada uno de estos relaciona cada zona con el hecho de ser un terreno u otro. Esta implicación en forma de predicado me ha llevado a formar 3 acciones distintas para la acción de desplazarse entre zonas. Una para pasar por zonas posibles sin objetos (piedra y arena) y otras dos para los casos en los que la zona es agua o bosque que podré pasar si tengo el objeto necesario para esta zona. Para saber el objeto necesario consulto el predicado necesito que relaciona una zona con el objeto necesario, zapatillas o bikini.

Además para controlar la mochila se necesitan dos nuevas acciones, guardar y sacar de la mochila. Para sacar de la mochila es necesario tener la mano libre y para guardar en mochila que no haya nada en la mochila, como es evidente.

```
mkxv@l00per: ~/Documentos/TSI/Sesion 7/belkan/Metric-FF
step 0: GIRAR-IZQ MKXV NORTE OESTE
1: COGER-OBJETO MKXV ALGORITMO Z0
2: DAR-OBJETO MKXV ALGORITMO Z0 PROFESOR
3: GIRAR-DER MKXV OESTE NORTE
4: PASAR-NORMAL MKXV Z0 Z1 NORTE
5: GIRAR-IZQ MKXV NORTE OESTE
6: GIRAR-IZQ MKXV OESTE SUR
7: COGER-OBJETO MKXV ZAPATILLAS Z1
8: GUARDA-MOCHILA MKXV ZAPATILLAS
9: COGER-OBJETO MKXV ORO Z1
10: PASAR-NORMAL MKXV Z1 Z0 SUR
11: GIRAR-IZQ MKXV SUR ESTE
12: PASAR-NORMAL MKXV Z0 Z2 ESTE
13: PASAR-BOSQUE MKXV Z2 Z7 ESTE ZAPATILLAS
14: GIRAR-IZQ MKXV ESTE NORTE
15: PASAR-NORMAL MKXV Z7 Z6 NORTE
16: DAR-OBJETO MKXV ORO Z6 PRINCIPE
17: GIRAR-IZQ MKXV NORTE OESTE
18: GIRAR-IZQ MKXV OESTE SUR
19: PASAR-BOSQUE MKXV Z6 Z7 SUR ZAPATILLAS
20: GIRAR-DER MKXV SUR OESTE
21: PASAR-NORMAL MKXV Z7 Z2 OESTE
22: GIRAR-DER MKXV OESTE NORTE
23: GIRAR-DER MKXV NORTE ESTE
24: COGER-OBJETO MKXV ROSA Z2
25: PASAR-BOSQUE MKXV Z2 Z7 ESTE ZAPATILLAS
26: GIRAR-DER MKXV ESTE SUR
27: PASAR-NORMAL MKXV Z7 Z8 SUR
28: DAR-OBJETO MKXV ROSA Z8 PRINCESA
29: GIRAR-DER MKXV SUR OESTE
30: PASAR-BOSQUE MKXV Z8 Z9 OESTE ZAPATILLAS
31: GIRAR-DER MKXV OESTE NORTE
32: PASAR-NORMAL MKXV Z9 Z3 NORTE
33: COGER-OBJETO MKXV OSCAR Z3
34: GIRAR-IZQ MKXV NORTE OESTE
35: GIRAR-IZQ MKXV OESTE SUR
36: PASAR-BOSQUE MKXV Z3 Z9 SUR ZAPATILLAS
37: GIRAR-DER MKXV SUR OESTE
38: PASAR-NORMAL MKXV Z9 Z10 OESTE
39: DAR-OBJETO MKXV OSCAR Z10 LEONARDO
Coste Total: 54.00

time spent: 0.01 seconds instantiating 1291 easy, 24 hard action templates
0.00 seconds reachability analysis, yielding 210 facts and 326 actions
0.00 seconds creating final representation with 174 relevant facts, 1 relevant fluents
0.00 seconds computing LNF
0.00 seconds building connectivity graph
1.88 seconds searching, evaluating 17378 states, to a max depth of 0
1.89 seconds total time

mkxv@l00per:~/Documentos/TSI/Sesion 7/belkan/Metric-FF$
```

Para este ejercicio el problema definido era un poco más complejo pero no mucho más allá, la optimización por defecto -O encuentra una solución buena en 0,32 segundos, al igual que la -g 1 -h 1

- **Ejercicio 4:** Se extiende el comportamiento del ejercicio 3 pero ahora los objetos deben de poder entregar a cualquiera siguiendo la tabla de valores aportada.

Para este problema lo único que hay que he tenido que hacer es añadir nuevos objetos, únicamente he puesto el doble de objetos, dos de cada tipo, con 5 personajes. Dejando 5 de los objetos en las mismas posiciones que antes.

He representado la tabla de entregas con con la función beneficio-parcial de entregar un posible objeto a los 5 personajes. Esta función me sirve para incrementar el beneficio total de las entregas, por ello, al dar un

objeto a un personaje ya no tengo en cuenta el hecho a-entregar, que relaciona a cada personaje con su objeto predilecto.

El goal que he ajustado es que el beneficio total de dar los objetos ha de ser 0, minimizandolo, ya que lo que hago es restar sobre el beneficio el valor del objeto. En el resultado de la ejecución se puede observar como tiende a dar objetos cuyo beneficio será 10 para alcanzar más rápido, en mi caso, de un beneficio de 0 ya que son las restas más fuertes.

```
mkxv@l00per: ~/Documentos/TSI/Sesion 7/belkan/Metric-FF
1: GIRAR-IZQ MKXV NORTE OESTE
2: DAR-OBJETO MKXV ALGORITMO Z0 PROFESOR
3: PASAR-NORMAL MKXV Z0 Z4 OESTE
4: GIRAR-IZQ MKXV OESTE SUR
5: COGER-OBJETO MKXV MANZANA Z4
6: GUARDA-MOCHILA MKXV MANZANA
7: GIRAR-IZQ MKXV SUR ESTE
8: COGER-OBJETO MKXV ALGORITMO Z4
9: PASAR-NORMAL MKXV Z4 Z0 ESTE
10: DEJAR-OBJETO MKXV ALGORITMO Z0
11: SACAR-MOCHILA MKXV MANZANA
12: GIRAR-IZQ MKXV ESTE NORTE
13: DAR-OBJETO MKXV MANZANA Z0 PROFESOR
14: COGER-OBJETO MKXV ALGORITMO Z0
15: GIRAR-DER MKXV NORTE ESTE
16: PASAR-NORMAL MKXV Z0 Z2 ESTE
17: GIRAR-DER MKXV ESTE SUR
18: DEJAR-OBJETO MKXV ALGORITMO Z2
19: COGER-OBJETO MKXV BIKINI Z2
20: GUARDA-MOCHILA MKXV BIKINI
21: GIRAR-DER MKXV SUR OESTE
22: COGER-OBJETO MKXV ALGORITMO Z2
23: PASAR-NORMAL MKXV Z2 Z0 OESTE
24: GIRAR-DER MKXV OESTE NORTE
25: DAR-OBJETO MKXV ALGORITMO Z0 PROFESOR
26: PASAR-NORMAL MKXV Z0 Z1 NORTE
27: COGER-OBJETO MKXV ORO Z1
28: PASAR-AGUA MKXV Z1 Z5 NORTE BIKINI
29: GIRAR-DER MKXV NORTE ESTE
30: PASAR-NORMAL MKXV Z5 Z6 ESTE
31: GIRAR-DER MKXV ESTE SUR
32: GIRAR-DER MKXV SUR OESTE
33: DAR-OBJETO MKXV ORO Z6 PRINCIPE
34: PASAR-AGUA MKXV Z6 Z5 OESTE BIKINI
35: GIRAR-IZQ MKXV OESTE SUR
36: PASAR-NORMAL MKXV Z5 Z1 SUR
37: COGER-OBJETO MKXV MANZANA Z1
38: PASAR-NORMAL MKXV Z1 Z0 SUR
39: GIRAR-DER MKXV SUR OESTE
40: DAR-OBJETO MKXV MANZANA Z0 PROFESOR
41: PASAR-NORMAL MKXV Z0 Z4 OESTE
42: COGER-OBJETO MKXV OSCAR Z4
43: PASAR-AGUA MKXV Z4 Z11 OESTE BIKINI
44: GIRAR-IZQ MKXV OESTE SUR
45: PASAR-NORMAL MKXV Z11 Z10 SUR
46: DAR-OBJETO MKXV OSCAR Z10 LEONARDO
Coste Total: 0.00

time spent: 0.01 seconds instantiating 1291 easy, 24 hard action templates
0.00 seconds reachability analysis, yielding 194 facts and 389 actions
0.00 seconds creating final representation with 189 relevant facts, 3 relevant fluents
0.00 seconds computing LNF
0.00 seconds building connectivity graph
16.34 seconds searching, evaluating 64607 states, to a max depth of 0
16.35 seconds total time

mkxv@l00per:~/Documentos/TSI/Sesion 7/belkan/Metric-FF$ ./ff -p ../Ejercicio\ 4/ -o dominio.pddl -f problema.pd
dl -o
```

En este ejercicio con optimización -O funciona bien y alcanza un tiempo de unos 16 segundos, en cambio, le es imposible con -O -g 1 -h 1.

- **Ejercicio 5:** Considerar ahora que el jugador tiene una mochila con una capacidad  $n > 1$ , (configurable en el estado inicial).

Para la solución de este ejercicio he añadido una función mochila que incrementa su valor cuando se introduce algún objeto al igual que decrementa cuando se saca. El límite del tamaño se establece en el init. Al al iniciar el problema el tam de la mochila es n, si metemos algo disminuye el “espacio libre” por lo que cuando el espacio “libre” llegue a 0 indicará que la mochila está completa. Por ello cuando introduzco un objeto lo que hago decrementar y cuando saco es aumentar. Además defino el tam-máximo con otra función.

Para ejemplificar los resultados lo he ejecutado con dos tamaños de mochila distintos, están todos los personajes en la misma casilla y en una opuesta todos los

◆ TAM 1 :

```
step    0: PASAR-NORMAL MKXV Z0 Z1 NORTE
        1: GIRAR-DER MKXV NORTE ESTE
        2: COGER-OBJETO MKXV ORO Z1
        3: GUARDA-MOCHILA MKXV ORO
        4: GIRAR-DER MKXV ESTE SUR
        5: COGER-OBJETO MKXV ROSA Z1
        6: PASAR-NORMAL MKXV Z1 Z0 SUR
        7: PASAR-NORMAL MKXV Z0 Z3 SUR
        8: DAR-OBJETO MKXV ROSA Z3 PRINCESA
        9: SACAR-MOCHILA MKXV ORO
       10: GIRAR-IZQ MKXV SUR ESTE
       11: GIRAR-IZQ MKXV ESTE NORTE
       12: DAR-OBJETO MKXV ORO Z3 PRINCIPE
       13: PASAR-NORMAL MKXV Z3 Z0 NORTE
       14: PASAR-NORMAL MKXV Z0 Z1 NORTE
       15: GIRAR-DER MKXV NORTE ESTE
       16: COGER-OBJETO MKXV OSCAR Z1
       17: GUARDA-MOCHILA MKXV OSCAR
       18: GIRAR-DER MKXV ESTE SUR
       19: COGER-OBJETO MKXV MANZANA Z1
       20: PASAR-NORMAL MKXV Z1 Z0 SUR
       21: PASAR-NORMAL MKXV Z0 Z3 SUR
       22: DAR-OBJETO MKXV MANZANA Z3 BRUJA
       23: SACAR-MOCHILA MKXV OSCAR
       24: DAR-OBJETO MKXV OSCAR Z3 LEONARDO
          Coste Total: 0.00

time spent:  0.03 seconds instantiating 1291 easy, 24 hard action templates
            0.01 seconds reachability analysis, yielding 197 facts and 389 actions
            0.00 seconds creating final representation with 187 relevant facts, 5 relevant fluents
            0.00 seconds computing LNF
            0.00 seconds building connectivity graph
           58.18 seconds searching, evaluating 90552 states, to a max depth of 0
           58.22 seconds total time

mkxv@l00per:~/Documentos/TSI/Sesion 7/belkan/Metric-FFS
```



◆ TAM 4 :

```
step 0: PASAR-NORMAL MKXV Z0 Z1 NORTE
1: GIRAR-DER MKXV NORTE ESTE
2: COGER-OBJETO MKXV ORO Z1
3: GUARDA-MOCHILA MKXV ORO
4: GIRAR-DER MKXV ESTE SUR
5: COGER-OBJETO MKXV ROSA Z1
6: GUARDA-MOCHILA MKXV ROSA
7: COGER-OBJETO MKXV OSCAR Z1
8: GUARDA-MOCHILA MKXV OSCAR
9: COGER-OBJETO MKXV MANZANA Z1
10: PASAR-NORMAL MKXV Z1 Z0 SUR
11: PASAR-NORMAL MKXV Z0 Z3 SUR
12: DAR-OBJETO MKXV MANZANA Z3 BRUJA
13: SACAR-MOCHILA MKXV OSCAR
14: DAR-OBJETO MKXV OSCAR Z3 LEONARDO
15: SACAR-MOCHILA MKXV ORO
16: DAR-OBJETO MKXV ORO Z3 PRINCIPE
17: SACAR-MOCHILA MKXV ROSA
18: DAR-OBJETO MKXV ROSA Z3 PRINCESA
Coste Total: 0.00

time spent: 0.02 seconds instantiating 1291 easy, 24 hard action templates
0.00 seconds reachability analysis, yielding 197 facts and 389 actions
0.00 seconds creating final representation with 187 relevant facts, 5 relevant fluents
0.00 seconds computing LNF
0.00 seconds building connectivity graph
9.06 seconds searching, evaluating 40324 states, to a max depth of 0
9.08 seconds total time

mkxv@l00per:~/Documentos/TSI/Sesion 7/belkan/Metric-FFS
```

Como podemos ver se realizan menos acciones cuando el tamaño de la mochila es más grande esto demuestra lo que el ejercicio dice. Además la optimización -g 1 -h 1 funciona perfectamente.

- **Ejercicio 6:** Con el dominio definido en el anterior ejercicio definir 3 problemas con distintas capacidades de la mochila y con distinto número de objetos a entregar, de forma que el planificador encuentre un plan para ellos maximizando el número de puntos a conseguir.

Para este ejercicio he tomado un punto de vista distinto, como hay que maximizar el número de entregas diremos que el goal es, entregar todos los objetos con un beneficio máximo por objeto. Al mismo tiempo, como maximize en el metric no funciona lo que he hecho ha sido minimizar el beneficio total máximo posible, es decir, si hay x objetos los repartirá de manera que el los menos valiosos se los dará a los personajes cuya relación con ellos es muy baja.

Con esta idea mis problemas para este ejercicio tiene distribuidos todos los objetos en la misma zona y todos los personajes en otra casilla, opuesta. De esta forma fuerzo a hacer uso de la mochila y veo si minimiza

al máximo el valor inicial del beneficio. La ejecución se divide en tres casos:

- ◆ Tam mochila 5, personajes 5, objetos 7 (1 oro , 1 oscar 1 rosa, 1 manzanas, 1 algoritmo, 1 zapatillas y bikini), entregas 5:

```
mkxv@l00per: ~/Documentos/TSI/Sesion 7/belkan/Metric-FF
99: GIRAR-DER MKXV NORTE ESTE
100: SACAR-MOCHILA MKXV ROSA
101: DAR-OBJETO MKXV ROSA Z1 LEONARDO
102: SACAR-MOCHILA MKXV MANZANA
103: GIRAR-IZQ MKXV ESTE NORTE
104: DAR-OBJETO MKXV MANZANA Z1 PRINCESA
105: COGER-OBJETO MKXV ZAPATILLAS Z1
106: GIRAR-IZQ MKXV NORTE OESTE
107: GIRAR-IZQ MKXV OESTE SUR
108: PASAR-NORMAL MKXV Z1 Z0 SUR
109: GUARDA-MOCHILA MKXV ZAPATILLAS
110: COGER-OBJETO MKXV ALGORITMO Z0
111: GUARDA-MOCHILA MKXV ALGORITMO
112: SACAR-MOCHILA MKXV ZAPATILLAS
113: DEJAR-OBJETO MKXV ZAPATILLAS Z0
114: SACAR-MOCHILA MKXV ALGORITMO
115: PASAR-NORMAL MKXV Z0 Z3 SUR
116: DEJAR-OBJETO MKXV ALGORITMO Z3
117: COGER-OBJETO MKXV OSCAR Z3
118: GUARDA-MOCHILA MKXV OSCAR
119: COGER-OBJETO MKXV ALGORITMO Z3
120: GUARDA-MOCHILA MKXV ALGORITMO
121: SACAR-MOCHILA MKXV OSCAR
122: GIRAR-IZQ MKXV SUR ESTE
123: GIRAR-IZQ MKXV ESTE NORTE
124: PASAR-NORMAL MKXV Z3 Z0 NORTE
125: PASAR-NORMAL MKXV Z0 Z1 NORTE
126: GIRAR-DER MKXV NORTE ESTE
127: DAR-OBJETO MKXV OSCAR Z1 PRINCIPE
128: SACAR-MOCHILA MKXV ALGORITMO
129: DAR-OBJETO MKXV ALGORITMO Z1 BRUJA
Coste Total: 5.00

time spent: 0.01 seconds instantiating 1293 easy, 24 hard action templates
0.00 seconds reachability analysis, yielding 193 facts and 405 actions
0.00 seconds creating final representation with 188 relevant facts, 6 relevant f
luents
0.00 seconds computing LNF
0.00 seconds building connectivity graph
0.02 seconds searching, evaluating 1074 states, to a max depth of 0
0.03 seconds total time

mkxv@l00per:~/Documentos/TSI/Sesion 7/belkan/Metric-FF$
```

**NOTA IMPORTANTE:** Tras mucho tiempo de experimentación no he sido capaz de hacer funcionar los distintos ejemplos. Hemos probado diferentes problemas pero no es posible resolverlo. A tal punto llega que si copio todo el ejemplo citado arriba y disminuyo en 1 todos los parametros, es decir, quito por ejemplo a leonardo, el oscar y un tam menos de mochila y entra en un optimo local. Lo doy por imposible tras muchas horas de ensayo y error.