

Práctica 3 - Técnicas de los sistemas inteligentes

1. Tabla de resultados

	Número de acciones	Tiempo invertido
Ejercicio 1	4	0.00 segundos
Ejercicio 2	11	0.00 segundos
Ejercicio 3	17	0.06 segundos
Ejercicio 4	31	0.03 segundos
Ejercicio 5	52	3.81 segundos
Ejercicio 6	52	536.43 segundos
Ejercicio 7	34	121.53 segundos

A medida que avanzamos en la resolución de los ejercicios, aumenta el tiempo invertido en esta. Sobre todo, aumenta notablemente en los ejercicios en los que se lleva un control del número de acciones completadas (ejercicios 6 y 7), pues esta actividad parece muy costosa. En el ejercicio 7, aún así, baja el tiempo invertido (y el número de acciones) en comparación con el ejercicio anterior. Esto parece deberse a que los teletransportadores comienzan estando ya contruidos, y su construcción (en el 6) es una de las acciones que más tiempo lleva, pues tiene más precondiciones (tener Investigación de Teletransporte) que al construir otro tipo de unidades.

En los primeros ejercicios, los tiempos son prácticamente despreciables, pues las acciones que se invocan tienen muy pocas precondiciones. Esto nos hace pensar que el número de precondiciones (así como su complejidad) influye notablemente en el tiempo invertido en realizar las acciones.

También he notado que al introducir más objetivos en los problemas, aunque parezcan redundantes, se consigue reducir un poco el tiempo de ejecución del problema, aunque no su número de acciones requeridas.

2. Pregunta #1

Sí, el tiempo que tarda MetricFF es aproximadamente el mismo para distintos valores del número máximo de acciones, aunque sube un poco, cuanto menor fijemos este valor. Esto puede ser porque al construir los planes, tiene que descartar los que tienen más acciones que el máximo, cuando con un valor máximo mayor, no los hubiese descartado. Es decir, se siguen buscando planes nuevos con menor número de pasos, cuando antes se hubiese aceptado uno con más pasos, y esto conlleva más tiempo.

Por otro lado, cuando comparamos el tiempo de resolución minimizando una métrica y sin minimizarla, este no es ni remotamente parecido. El ejemplo lo vemos en el tiempo del ejercicio 5 vs el del ejercicio 6. Son el mismo ejercicio, pero añadiendo la minimización de la métrica, lo que conlleva mucho más tiempo. Esto puede deberse a que, tras realizar cada acción, tiene que comprobar que no se haya superado el máximo de acciones permitido, en lugar de quedarse con otra solución cercana al óptimo.

3. Pregunta #2

Apenas varía el tiempo de ejecución introduciendo los cambios propuestos. A medida que voy añadiendo más operadores (hasta llegar a 7), los tiempos van oscilando siempre entre 0.2 y 0.5, pero no se aprecia una relación de incremento o decremento directa. Con 7 operadores, tarda concretamente 0.02 segundos, un poco menos que con menor número de operadores.

Considero que el hecho de añadir más acciones, debería suponer un incremento en el tiempo de ejecución, pues el programa tardaría más tiempo en comprobar las posibles combinaciones de acciones para llegar a un mismo resultado. O también porque se requerirán unas acciones previas a otras y eso también sube el tiempo.

Subir el número de objetos creo que podría ayudar a decrementar este tiempo, pues quizás se pueden utilizar dos operadores distintos para hacer lo mismo que antes hacía uno. Por ejemplo, un operador en lugar de tener que recorrer muchas posiciones del mapa para llegar a hacer algo, se puede utilizar otro operador que se coloca en otro lado y reducir el número de acciones de movimiento. Y si no hacen falta, no se utilizan estos operadores extra, por lo que sólo podrían beneficiar al plan.