

Técnicas de los Sistemas Inteligentes

Práctica 3

David Muñoz Sánchez
07256819C - Grupo A1

9 de junio de 2023

Índice

| | |
|--------------------------|---|
| 1. Tabla de resultados | 1 |
| 2. Preguntas | 2 |
| 2.1. Pregunta 1. | 2 |
| 2.2. Pregunta 2. | 2 |
| 3. Nota final. | 2 |

1. Tabla de resultados

| Problema | Ejecución | Longitud plan | Coste | Tiempo(s) |
|----------|--------------|---------------|-------|-----------|
| 1 | - | 5 | - | 0 |
| 2 | - | 13 | - | 0 |
| 3 | - | 21 | - | 7.18 |
| 4 | - | 30 | - | 0 |
| 5 | - | 28 | - | 0 |
| 6 | SIN MINIMIZE | 28 | - | 0 |
| | CON MINIMIZE | 28 | - | 0.01 |
| 7 | SIN MINIMIZE | 59 | - | 1.64 |
| | CON MINIMIZE | 58 | - | 59.10 |
| 8 | SIN MINIMIZE | 60 | 568 | 1.73 |
| | CON MINIMIZE | 59 | 558 | 479.61 |

Añadir también que, si se restringe al problema 6 para que encuentre una solución menor a 28, es **irresoluble** y metricFF nos lo hace saber en **1.48s**. Por tanto 28 es la solución óptima en términos de longitud del plan.

En cuanto al problema 7, si se emplea esta misma técnica, no se ha obtenido ningún resultado en un período de 45 minutos.

Finalmente, el problema 8, como puede parecer natural, tampoco ha arrojado resultados en un período de más de 45 minutos, minimizando en este caso la nueva métrica de tiempo, no la longitud del plan como en los ejercicios anteriores.

El minimize permite añadir otras métricas para que FF intente minimizarlas. Como en los tres últimos ejercicios se nos pedían métricas específicas para este problema, se optó por ejecutar cada uno de ellos sin minimizar la métrica e intentando minimizarla, para ver como cambiaba, lo cual se refleja en la tabla. Digo *intentando* porque MetricFF no garantiza el óptimo, si no que garantiza que el plan es válido e intenta que sea lo más cercano al óptimo posible. Es por ello, que se decía en el guion que había que ir probando a añadir un requisito al objetivo del problema que impusiera que el valor de la métrica fuera menor a un valor anterior encontrado, hasta que FF nos dijera que era insatisfacible siendo ese el óptimo. Para el 6, ha habido suerte, ya que la primera vez que se probó, se demostró este

hecho. Para el 7 y el 8 no tanta, ya que, a partir de los valores devueltos en la primera ejecución, la restricción de la métrica no terminó en un período de 45 minutos, por lo que se desistió de intentar demostrar el óptimo.

2. Preguntas

2.1. Pregunta 1.

En las distintas llamadas a MetricFF necesarias para resolver el Ejercicio 6 (o cualquier otro en donde se busque optimizar de modo efectivo el tamaño del plan o algún otro criterio), ¿MetricFF suele tardar aproximadamente el mismo tiempo en todas ellas? ¿A qué cree que se debe este fenómeno? Razone su respuesta.

Aunque las llamadas con MetricFF son generalmente rápidas, se ve como a medida que el problema se vuelve más complejo, MetricFF tarda más en contrar una solución. Además, para demostrar la irresolubilidad del 6, se necesita 100 veces más de tiempo. Para los dos últimos se paró la ejecución en algo más de 45 minutos sin demostrar nada.

Este resultado es comprensible, ya que es mucho más fácil encontrar una solución que demostrar que un problema no tiene solución. Al pedir explícitamente una solución insatisfacible, MetricFF tiene que explorar prácticamente todas las opciones y caminos posibles.

Aunque esto puede no ser un problema grave en el caso del ejercicio 6 (donde es más lento demostrar la optimalidad pero sigue siendo rápido), se nota de manera significativa en los ejercicios 7 y 8. En esos casos, los tiempos de ejecución para demostrar la insatisfacibilidad se vuelven computacionalmente inviables dado el crecimiento del espacio de búsqueda y de la complejidad del problema.

2.2. Pregunta 2.

En base a los tiempos de ejecución obtenidos, al comportamiento general observado en los distintos ejercicios, y teniendo en cuenta que el dominio de planificación planteado en esta práctica es de dificultad baja o moderada, ¿cuáles considera que pueden ser las principales limitaciones de la planificación automática en otros dominios? ¿Cómo considera que escalaría la resolución de los problemas si incorporásemos muchas más acciones, objetos y nodos del mapa?

En este dominio específico, parece que la planificación tiene una dificultad relativamente baja. Aunque se han aumentado gradualmente la complejidad de los problemas, los tiempos de ejecución no se han visto enormemente afectados. Sin embargo, a medida que aumentas la complejidad, especialmente al buscar soluciones óptimas, los tiempos de ejecución han aumentado significativamente.

Parece razonable considerar que los ejercicios 7 y 8 en este dominio ya presentan una dificultad un poco superior a los anteriores. Aunque estos problemas se resuelven en bastantes segundos, garantizar su funcionamiento óptimo puede llevar varias veces más de tiempo. En mi caso, después de 45 minutos en cada problema, no se ha llegado a ninguna conclusión.

Las principales limitaciones de la planificación automática en otros dominios pasarían por la complejidad de representación de esos dominios, ya que en los problemas de esta práctica, que no presentan mucha dificultad de planificación, a medida que se han ido añadiendo acciones, objetos y características, ha aumentado el tiempo para encontrar una solución (especialmente en los de optimización). Todo apunta, a que si siguiésemos esa tónica, los tiempos crecerían de forma exponencial y complicarían aún más la planificación.

3. Nota final.

Se añade también el fichero Python para calcular el coste, modificado, ya que he tenido que incluir la posibilidad de que haya `NAVEGAR_{1,2,3,4}` en el plan y no simplemente `NAVEGAR`.