

Técnicas de los Sistemas Inteligentes

Práctica 3: Planificación con PDDL

Curso 2021-2022

Pedro Bedmar López - 75935296Z

pedrobedmar@correo.ugr.es

Grado en Ingeniería Informática

1. Tabla de resultados

Tras la implementación de los ejercicios de la práctica, obtengo los resultados que aparecen en la siguiente tabla:

Ejercicio	Número de acciones (y coste si procede)	Tiempo de ejecución (s)
1	3	0.00
2	11	0.01
3	16	0.73
4	28	0.18
5	28	27.47
6	24	594.65
7	45	43.31
8	45 (coste: 493)	2.72

El desarrollo de los ejercicios ha sido incremental: esto significa que hemos comenzado con un dominio básico al que se han ido añadiendo componentes.

Del ejercicio 1 al 6 hemos seguido este enfoque progresivo. Como se puede observar en la mayoría de casos, el tiempo de ejecución aumenta de un ejercicio al siguiente. Esto es lógico, ya que al implementar nuevas acciones y precondiciones el espacio de búsqueda aumenta, de la misma forma que la complejidad del problema. Esta complejidad también conlleva un aumento del número de acciones necesarias a ejecutar (el camino hasta la solución es más largo).

Destacar el tiempo de ejecución del ejercicio 6, donde casi se superan los 600 segundos. Esto se debe a que es un problema complejo, donde se optimiza el resultado del ejercicio 5 evitando superar un número de acciones. Este número coincide con el mínimo posible, 24 acciones, por lo que PDDL tiene que realizar una búsqueda mucho más exhaustiva al existir una restricción tan fuerte.

Los ejercicios 7 y 8 se desarrollan a partir del ejercicio 4. El hecho de utilizar métricas numéricas hace que el tiempo de ejecución aumente con respecto al ejercicio 4.

2. Preguntas

En las distintas llamadas a MetricFF necesarias para resolver el Ejercicio 6, ¿MetricFF tarda aproximadamente el mismo tiempo en todas ellas? ¿A qué cree que se debe este fenómeno? Razone su respuesta.

No tarda lo mismo. Cuanto más alto es el número de acciones permitidas, menos tarda la ejecución. Esto se debe a que si el número de acciones permitidas es alto, el planificador tiene mayor libertad para elegir posibles soluciones. Si limitamos el número de acciones, hay que llevar a cabo una búsqueda mucho más exhaustiva hasta dar con un plan que satisface las restricciones. En el momento en que restringimos el número de acciones hasta tal punto que no existe solución, la ejecución será especialmente lenta, ya que se recorre todo el árbol de búsqueda para confirmar que el problema es insatisfacible.

En el ejercicio 5, donde no existe ningún número máximo de acciones permitidas, el tiempo de ejecución es de 27.47 segundos devolviendo un plan de 28 acciones. Al restringir la búsqueda para que el plan sea menor a 28 acciones, obtenemos 24 acciones en 594.65 segundos. Si intentamos restringirla aún más (a un número de acciones menor a 24), obtenemos que el problema es insatisfacible. En realizar esta comprobación tarda 689.20 segundos.

En base a los tiempos de ejecución obtenidos, ¿cree que el dominio de planificación planteado en esta práctica es de dificultad moderada/media/alta? ¿Cuáles son las limitaciones de la planificación automática en otros dominios?

Creo que depende del ejercicio. En los primeros la dificultad es baja, pero conforme vamos avanzando en la práctica la dificultad aumenta al tener un mayor número de precondiciones y acciones. El máximo se alcanza con la minimización del número de acciones realizadas.

En dominios donde existen muchas variables a tener en cuenta (expresadas en forma de un gran número de precondiciones y acciones), la planificación automática puede dar problemas debido al alto coste computacional que supone buscar un plan que cumpla todas las restricciones. Existe el mismo problema en dominios donde existen variables que pueden tomar un gran número de valores, ya que sentencias como *forall* sufren enormemente.

También factores como el determinismo/no determinismo de las acciones o el uso de variables discretas o continuas influyen enormemente en la dificultad de resolución.

Finalmente comentar que a nivel teórico los problemas de planificación clásica son problemas muy complejos, englobándose dentro de la clase de complejidad PSPACE.