

Técnicas de los Sistemas Inteligentes

Enunciado de la Práctica 3

Introducción

El objetivo de la tercera práctica de la asignatura de Técnicas de los Sistemas Inteligentes consiste en diseñar e implementar un dominio de planificación clásico basado en el juego StarCraft¹. Para el desarrollo de la práctica los estudiantes deberán confeccionar un conjunto de ficheros PDDL y ejecutarlos en el planificador Metric-FF para obtener un plan válido. Este documento describe el mundo del juego, así como las tareas a realizar para superar la práctica, e información sobre la entrega y su evaluación.

Descripción del mundo

El mundo sobre el que se va a desarrollar la práctica está inspirado en juegos de estrategia militar y gestión de recursos, tales como StarCraft o Age of Empires. En StarCraft, un grupo de vehículos de construcción espacial (VCEs) deben recolectar una serie de recursos para construir nuevas estructuras, entre otras posibles acciones. Para la realización de esta práctica usaremos una versión simplificada del juego StarCraft 1 de Blizzard Entertainment (Figura 1).



Figura 1: Ejemplo de entorno en el que se desarrolla la acción en StarCraft.

¹ Videojuego de estrategia en tiempo real de ciencia ficción militar desarrollado por Blizzard Entertainment, y lanzado en 1998. Se trata de uno de los juegos para ordenador más vendidos de la historia, con más de 11 millones de copias vendidas (a fecha de Febrero de 2009). Más información en [https://en.wikipedia.org/wiki/StarCraft_\(video_game\)](https://en.wikipedia.org/wiki/StarCraft_(video_game))

En nuestro mundo tenemos una pequeña base Terran², en la que podemos construir edificios y reclutar unidades. Para poder realizar estas acciones necesitaremos extraer recursos del entorno. Los recursos que podremos extraer serán de dos tipos:

- Minerales



- Gas Vespeno



Ambos recursos deben ser recolectados por un tipo de unidad: el VCE. Los VCE son los trabajadores encargados recolectar recursos y construir edificios. Los nuevos edificios que se construyan desbloquean nuevas capacidades en la base, ya sea habilitando el reclutamiento de nuevos tipos de unidades, o permitiendo el desarrollo de investigaciones que a su vez expandirán aún más dichas capacidades. Para poder extraer recursos, los VCE deben asignarse a un “nodo de recursos” (el emplazamiento donde habrá o bien minerales o bien gas vespeno) y se encargarán de, periódicamente, obtener/extraer el recurso. Un trabajador asignado a un recurso no podrá moverse por el mapa ni construir edificios hasta que sea desasignado del nodo. A un nodo de recursos se pueden asignar tantos VCE como haya libres. Por simplicidad, será suficiente asignar un único VCE a un nodo de recursos para tener ilimitados recursos de ese tipo (o bien minerales o bien gas vespeno), los cuales podrán ser usados para construir o reclutar cualquier elemento que requiera dicho recurso.



Figura 2: Ejemplo de VCE para recolección de recursos

Para representar el mapa usaremos un grid. Cada celda del grid representará una “localización”, en la que puede haber edificios y unidades. No hay límite en cuanto al número de unidades que pueden encontrarse en una localización determinada, pero solo puede haber un edificio construido en cada localización. Cada nodo de recursos se encuentra en alguna localización. En una localización solo puede haber un nodo de recursos, aunque puede construirse un edificio en una localización que contenga un nodo de recurso. Es decir, una misma localización puede contener al mismo tiempo un nodo de recursos, un edificio y número ilimitado de unidades. Cada localización está conectada con aquellas localizaciones adyacentes en el grid que representa el mapa (es decir, casillas de arriba, abajo, izquierda y derecha; excepto en bordes y esquinas).

Descripción de la tarea a realizar

Los estudiantes deberán definir un dominio de planificación escrito en PDDL que represente el mundo presentado en la sección anterior y, a medida que se realicen los ejercicios propuestos, ir añadiéndole nuevas funcionalidades.

² Una de las tres razas del videojuego StarCraft, junto con los Protoss y los Zerg. Los Terran son humanos exiliados de la Tierra y altamente evolucionados, con una gran capacidad de adaptación a cualquier situación.



Ejercicio 1: Diseñar y programar una versión inicial del dominio de planificación propuesto. Los dominios diseñados deben cumplir con los requisitos propuestos en las siguientes líneas.

- Representar en el dominio los tipos: **Unidades, Edificios, y Localizaciones**. Usar constantes para representar el tipo de unidad **VCE**, los tipos de edificio **CentroDeMando** y **Barracones**, y los tipos de recursos **Minerales** y **Gas**.
- Definir los **predicados** necesarios para:
 - Determinar si un edificio o unidad están en una localización concreta.
 - Representar que existe un camino entre dos localizaciones.
 - Asignar un nodo de un recurso concreto a una localización concreta.
 - Indicar si un VCE están extrayendo un recurso.
 - Definir qué recurso necesita cada edificio para ser construido.
- El dominio debe contener las siguientes 3 acciones definidas:
 - **Navegar**: Mueve a una unidad entre dos localizaciones.
 - **Asignar**: Asigna un trabajador a un nodo de recurso. Por simplicidad, un trabajador asignado a un nodo de mineral ya no podrá hacer nada más el resto de la ejecución.
 - **Construir**: Ordena a un trabajador libre que construya un edificio en una localización. En esta implementación, un edificio solo requerirá de un único tipo de recurso para ser construido.

En caso de que el estudiante considere que necesita más tipos, constantes o predicados para realizar correctamente el ejercicio tiene libertad total para crearlos.

Definir un fichero de problema con un mapa de, al menos, 5x5 localizaciones de tamaño. En una de las localizaciones deben encontrarse un CentroDeMando y al menos tres VCE. En el mapa debe haber al menos tres nodos de mineral y dos de gas. Para poder construir los barracones hace falta tener minerales, mientras que para poder construir nuevos Centros de Mando hace falta gas. El objetivo (:goal) de este ejercicio es la construcción de un edificio de Barracones en una localización.

Nota: Aunque un edificio no haya sido construido, en la lista de objetos del problema debe aparecer un objeto para representarlo.

Ejercicio 2: Incluir una restricción nueva a la extracción de recursos, de forma que, para poder obtener Gas, primero deberá construirse un extractor sobre el nodo. Añadir el edificio Extractor. Este edificio solo puede construirse en una localización que tenga un nodo de gas y necesitará minerales. Modificar la acción *Asignar* para que solo pueda asignarse un VCE a un nodo de gas si tiene construido un extractor. El objetivo de este ejercicio es la construcción de un edificio de Barracones en una localización.

Ejercicio 3: Modificar la acción de *Construir* para que tenga en cuenta que un edificio puede requerir más de un tipo de recurso. La acción de construir debe inferir por sí misma si se tiene el tipo de recursos necesarios para poder ejecutarse. Para los siguientes ejercicios es muy importante definir esta restricción de forma que solo se tenga que instanciar una única vez por tipo de edificio. Ejemplo: Para incluir información sobre los recursos necesarios para los Barracones solo debe escribirse una única vez (necesita Barracones Minerales) independientemente del número de barracones que existan en el problema. Modificar las restricciones de construcción nuevos Centros de Mando para que requieran tanto Gas como Minerales. El objetivo de este ejercicio es la construcción de un edificio de Barracones en una localización.

Ejercicio 4: Crear una acción *Reclutar* para crear nuevas unidades. Incluir dos nuevos tipos de unidades, el Marine y el Segador. Al igual que los edificios, la creación de cada unidad requiere



distintos recursos. VCE y Marines requieren mineral. Segadores requieren mineral y gas. Cada edificio podrá generar unas unidades u otras: El CentroDeMando podrá reclutar nuevos VCE y los barracones podrán generar Marines y Segadores. Modificar el fichero de problema para que el punto de inicio sea tener sobre el mapa solo un trabajador y un CentroDeMando. El objetivo de este ejercicio es disponer de un marine en una localización, y tener otro marine y un segador en otra.

Ejercicio 5: Incluir en el dominio todos los elementos necesarios para usar una nueva acción *Investigar*. Esta acción permitirá realizar nuevas investigaciones para la base. Las investigaciones se desbloquean en un nuevo edificio “Bahía de Ingeniería”. Al igual que los edificios y unidades las investigaciones requieren de varios recursos. Crear una investigación “Impulsor Segador”. Modificar la acción *Reclutar* para que no se puedan crear Segadores hasta que no se haya obtenido esta investigación. El objetivo de este ejercicio es disponer de un marine en una localización, y tener otro marine y un segador en otra.

Ejercicio 6: Modificar todos los elementos necesarios del dominio para que este sea capaz de usar información numérica.

1. Incluir una nueva acción para recolectar recursos de un nodo y almacenarlos. Cada vez que se llame a esta acción se actualizará el número de minerales o gas guardado. Por cada trabajador asignado a un nodo se incrementará en 10 el recurso³. Incluir una nueva acción para desasignar VCE de un nodo.
2. Modificar las acciones de *Construir*, *Reclutar* e *Investigar* para para consumir un cierto número de recursos. Estos recursos deben estar almacenados y ser suficientes para poder ejecutar la acción.
3. Poner un límite a la cantidad de recursos de un tipo que se puede almacenar. Incluir un nuevo edificio “Depósito”. Cada depósito incrementará el límite de almacenamiento en cierta magnitud. Empezando por un límite de 100 unidades de cada recurso, cada depósito incrementará el límite en otras 100 unidades.
4. Los costes de cada elemento (edificio, unidad o investigación) son los siguientes:

Elemento	Minerales	Gas
Centro de Mando	150	50
Barracones	150	0
Extractor	75	0
Bahía	125	0
Deposito	75	25
VCE	50	0
Marine	50	0
Segador	50	50
Impulsor Segador	50	200

El objetivo de este ejercicio es disponer de un marine en una localización, y tener otro marine y un segador en otra.

³ En caso de que un factor de 10 no permita al alumno encontrar una solución en un tiempo razonable, puede emplearse un factor superior (20 o 25, por ejemplo), siempre y cuando los dominios devuelvan un plan correcto y en la memoria se explique/justifique la decisión.



Ejercicio 7: Incluir una nueva función para controlar el tiempo que tarda cada acción del plan.

1. Asignar una duración fija a la acción *Recolectar*. Asignar y desasignar son acciones que no cuestan tiempo. El resto de acciones dependerán del edificio, unidad o investigación con la que se esté trabajando.
2. Determinar una distancia para cada pareja de puntos conectados en el mapa. Modificar la acción *Navegar* para que su duración dependa de la relación entre distancia y la velocidad de la unidad que se va a mover. Por su naturaleza los Segadores deben ser el doble de rápidos que los Marines. Las distancias y velocidades deberán determinarlas los alumnos ellos estimen oportuno.
3. Incluir una métrica para encontrar un plan que minimice el tiempo del plan.
4. Los tiempos de creación de cada elemento (edificio, unidad o investigación) son los siguientes:

Elemento	Tiempo
Centro de Mando	100
Barracones	46
Extractor	21
Bahía	25
Depósito	21
VCE	12
Marine	18
Segador	32
Impulsor Segador	120

El objetivo de este ejercicio es disponer de un marine en una localización, y tener otro marine y un segador en otra.

Evaluación y Entrega

Los estudiantes deben entregar en la actividad creada en PRADO, **antes del 31 de Mayo de 2020 (a las 14:00)**, una carpeta comprimida en formato ZIP con el siguiente contenido:

- **Una memoria en formato PDF** en la que se describan pormenorizadamente las decisiones tomadas para realizar los ejercicios de la relación de problemas. Es importante que en cada apartado se comente qué modificaciones se ha hecho a los ficheros de problemas para cumplir los requisitos de cada ejercicio. Esta memoria deberá tener un apartado por ejercicio y no podrá exceder las 10 páginas. En caso de contener código PDDL para ilustrar algo en un apartado, éste debe ser breve y bien motivado. Es decir, la memoria no es el lugar donde repetir el código creado, sino donde se explica qué se ha hecho y por qué.
- **Una carpeta “código” con 7 ficheros de dominio y 7 ficheros de problema escritos en PDDL.** Por cada ejercicio debe haber un fichero de dominio y otro de problemas.

La puntuación para la entrega será la siguiente:

- 1 punto por cada ejercicio correcto.
- 3 puntos la memoria, en la que se valorará la claridad expositiva, la corrección de los argumentos, y la buena presentación.