



Tecnológico de Monterrey

Modelación de sistemas multiagentes con gráficas computacionales

TC2008B.523

Dr. Sergio Ruiz Loza
Dr. Christopher David Balderas Silva

Actividad Integradora

Annya Paulina Verduzco Meza - A01650668

Fecha de entrega: 23 de noviembre 2021

Introducción:

En esta actividad somos el propietario de 5 robots nuevos y un almacén lleno de cajas. El dueño anterior del almacén lo dejó en completo desorden, por lo que depende de tus robots organizar las cajas en algo parecido al orden y convertirlo en un negocio exitoso.

Todos los robots están equipados con la tecnología de sensores más nueva que les permite recibir datos de sensores de las cuatro celdas adyacentes. Por tanto, es fácil distinguir si un campo está libre, es una pared, contiene una pila de cajas (y cuantas cajas hay en la pila) o está ocupado por otro robot.

Los robots también tienen sensores de presión equipados que les indican si llevan una caja en ese momento.

Para esta tarea, debemos enseñar a sus robots cómo ordenar el almacén. La organización de los agentes depende de ti, siempre que todas las cajas terminen en pilas ordenadas de cinco.

Ejecución:

Para dar solución a la parte de multiagentes lo primero será declarar los agentes necesarios:

- Cajas
- Robots

Y el ambiente:

- Almacén

Inicializamos (init) las posiciones iniciales de todas las K cajas en el ambiente, todas se encuentran a nivel del piso, lo que significa que al iniciar la simulación no hay ninguna pila de cajas construidas.

Todos los agentes (robots) empiezan en una posición de celda aleatoria y vacía. A continuación se ejecuta la simulación en el tiempo máximo establecido

Durante la simulación se recopila la siguiente información:

- El tiempo final necesario para que todas las cajas que se encuentran en el ambiente estén apiladas en montones de máximo 5 cajas.
- El número total de movimientos realizados por todos los robots del ambiente (5).

Explicación de los diagramas

Diagrama de protocolos de interacción:

En el diagrama de protocolos de interacción como su nombre lo indica se muestra la interacción de los agentes con el ambiente. En este caso nuestro ambiente es el almacén y los agentes en él son nuestros 5 robots y las K cajas que se encuentran desordenadas en el piso.

Lo primero es inicializar la posición de las cajas, los robots reciben la información de las celdas adyacentes, encontrando si estas están ocupadas o libres, si el robot detecta una caja cerca avanza y la recoge y luego la deja para crear una pila.

Si de lo contrario el robot avanza y se encuentra con una pila de 5 cajas que está completa se mueve a una celda diferente aleatoria.

Finalmente se imprime la data recolectada necesaria:

- El tiempo final necesario para que todas las cajas que se encuentran en el ambiente estén apiladas en montones de máximo 5 cajas.
- El número total de movimientos realizados por todos los robots del ambiente (5).
- Analiza si existe una estrategia que podría disminuir el tiempo dedicado a apilar y organizar las cajas así como la cantidad de movimientos realizados.

Mostrando así todas las interacciones existentes entre los agentes y el ambiente.

Diagrama de clases:

En el diagrama de clases se muestran las 3 clases que se utilizarían en la implementación de la solución de este problema: el ambiente, el agente robot y agente caja.

En este diagrama la clase “Almacén” funciona como el ambiente de la simulación por lo tanto contiene a los agentes “robot” y “caja”. El ambiente tendrá 5 robots y K número de cajas. El objetivo es que los robots recojan las cajas y las coloquen en pilas de 5 cajas máximo.

El almacén lo simulamos con Multigrid, esto construye un tablero que permite la existencia de muchos agentes en la misma posición, schedule permite que los agentes avancen y den un paso cuando sea necesario; al ser de activación aleatoria activa a los agentes de manera aleatoria. Asumiendo que para la solución de la actividad utilizamos la librería de mesa se incluyen los atributos de schedule y printCollectedData, este último es responsable de guardar la información requerida para la solución.

La clase Robot_Agent simula a los robots en el almacén, tiene un atributo “hasBox” que ayuda a indicar si en ese momento el robot está sosteniendo una caja o si necesita apilarla. También tenemos pickAbox, dropBox y moveForward, acciones que el robot puede realizar.

Finalmente Box_Agent represent las cajas en la simulación, estas solo sirven para ser recogidas y apiladas.

Reflexión

Recordando la actividad del robot que aspiraba y limpiaba las celdas del grid, pienso que una estrategia para optimizar tiempo y movimientos sería correr varias veces la simulación variando el número de robots buscando una cantidad de agentes en donde apilar las cajas resulte más rápido.

Diagrama de protocolos de interacción:

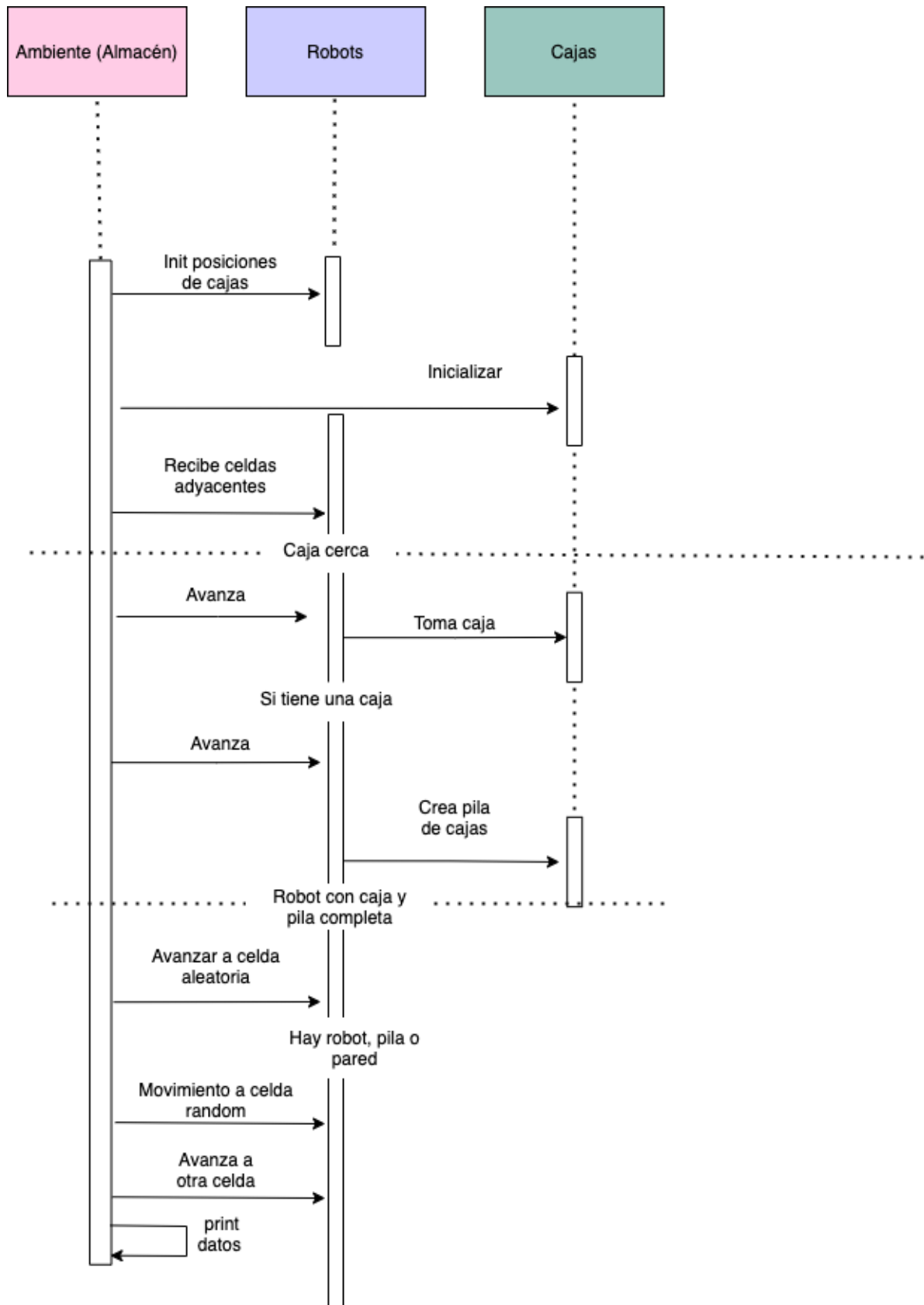


Diagrama de clases

