



Tecnológico de Monterrey

Evidencia 1. Actividad Integradora

Karen Corona Espinoza A01642461

Diego Villanueva Terrazas A01568601

Mariela Quintanar De la Mora A01642675

Monica Escobedo Cota A01642647

Valeria Aispuro Gómez A01642923

27 de Agosto de 2024

Modelación de sistemas multiagentes con gráficas computacionales

Iván Axel Dounce Navarro

Propiedades de agentes:

- Reactividad: los robots recogen objetos en celdas de cuadrícula frente a ellos con sus manipuladores, los llevan a otra ubicación e incluso construyen pilas (de hasta 5 objetos).

Pueden distinguir si un campo está libre, es una pared, contiene una pila de objetos (cuántos objetos), está ocupado por otro robot, y si llevan un objeto en ese momento.

- Proactividad: los robots llevan consigo un mapa, el cual les ayuda a decidir qué objetos recoger.

Propiedades de ambiente:

- Accesible: los robots se pueden mover libremente, sobre cualquier terreno, y también girar en cualquier dirección.
- No determinista: los robots están en un almacén lleno de objetos en desorden, pueden tomar la decisión de avanzar a cualquier lugar (y agarrar objetos).
- Dinámico: hay más de un robot.
- Discreto: cada robot lleva consigo un mapa, el cual les indica qué ruta tomar.
- No episódico: el ordenar el almacén no se puede dividir por tiempos, al tener tantos agentes modificando el ambiente al mismo tiempo.

Métricas de éxito para cada agente:

- Tiempo final: en cuánto tiempo están todos los objetos apilados.
- Acciones realizadas por cada robot: número de acciones para completar la tarea.
- Colisiones: número de colisiones evitadas/ número de veces que el robot se detuvo.
- Identificación: porcentaje de objetos correctamente identificados.

Fórmula:

$$\text{Éxito} = (1/\text{tiempo final}) + (1/\text{acciones realizadas}) + (\text{Identificación}/100) - \text{Colisiones}$$

Diagramas de clases (agentes):

- Clase 'Warehouse Agent': Representa a un agente en el almacén (uno de los robots que realizan acciones).

Propiedades:

- percepts: Lista de percepciones del robot (lo que ve en las celdas adyacentes).
- carries: Lista de objetos que el robot está transportando.
- target: El objeto objetivo que el robot busca recoger.
- moves: Contador de movimientos realizados por el robot.
- rules: Lista de reglas que dictan el comportamiento del robot.
- actions: Lista de acciones que el robot puede realizar.
- direction: Dirección en la que el robot está orientado.

Métodos:

- see: Actualiza las percepciones del robot.
- next: Decide la próxima acción a realizar.
- step: Ejecuta un paso de simulación.
- move: Mueve al robot hacia una posición objetivo.
- rotate_left: Gira al robot hacia la izquierda.
- pickup: Recoge un objeto.
- drop: Deja caer una pila de objetos.
- find_nearest_object: Encuentra el objeto más cercano.
- move_towards_target: Mueve al robot hacia su objetivo.

- Clase 'WarehouseObject': los objetos que los robots deben recoger y apilar.

Propiedades:

- agentType: Indica el tipo de agente (un objeto).
- stacked: Indica si el objeto ya está apilado.

Métodos:

- setup: Inicializa el objeto.
 - step: Define el comportamiento del objeto en cada paso de la simulación.
- Clase 'WarehouseStack': pila de objetos en el almacén.

Propiedades:

- agentType: Indica el tipo de agente (una pila).
- content: Lista de objetos contenidos en la pila.

Métodos:

- setup: Inicializa la pila de objetos.
- Clase 'WarehouseModel': modelo de simulación del almacén, que coordina a todos los agentes.

Propiedades:

- grid: La cuadrícula en la que se mueven los robots y se encuentran los objetos.
- robots: Lista de robots en la simulación.
- objects: Lista de objetos en la simulación.
- stacks: Lista de pilas de objetos en la simulación.

- steps: Contador de pasos de la simulación.

Métodos:

- setup: Inicializa la simulación.
- step: Ejecuta un paso de la simulación.
- end: Finaliza la simulación y devuelve los resultados.

Relaciones en la simulación:

- ‘WarehouseAgent’ y ‘WarehouseObject’ (Interacción): Los WarehouseAgent (robots) interactúan con WarehouseObject (objetos) en la simulación. Los agentes pueden percibir, recoger y mover estos objetos.
- ‘WarehouseAgent’ y ‘WarehouseStack’ (Interacción): Los WarehouseAgent pueden crear WarehouseStack al apilar objetos en la cuadrícula.
- ‘WarehouseModel’ y las demás clases (Contención): WarehouseModel contiene y maneja instancias de WarehouseAgent, WarehouseObject, y WarehouseStack. Este modelo coordina la simulación, controlando los movimientos y las acciones de los robots, así como la disposición de los objetos y pilas.

Diagramas de clases (ontologías):

- Clase ‘Entity’.
- Clase ‘Robot’ subclase de ‘Entity’:

Propiedades:

- ‘carries’ : relación con ‘objeto’ (un robot puede llevar objetos/ un máximo de 5).
- Clase ‘Objeto’ subclase de ‘Entity’:

Propiedades:

- ‘has_place’: relación con ‘Place’ (indica que un objeto tiene un lugar asignado).

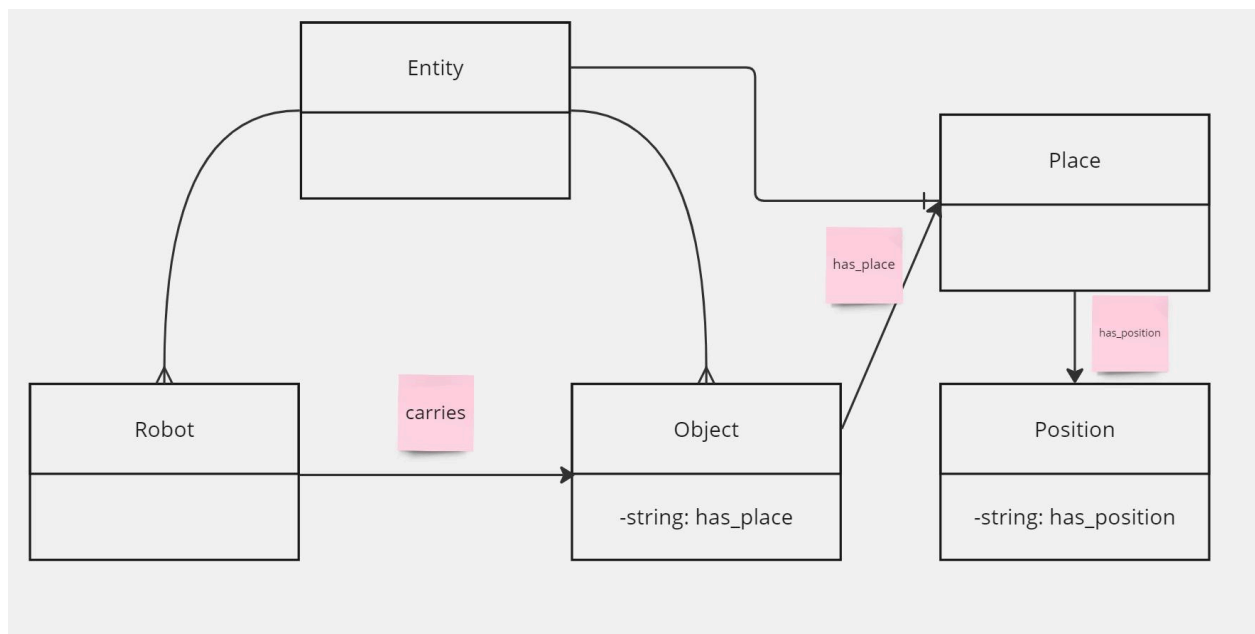
- Clase 'Place'.
- Clase 'Position' subclase de 'Place':

Propiedades:

- 'has_position': (string) propiedad de datos que asocia una posición con una cadena.

Relaciones:

- Entity: relación de herencia con Robot y Object.
- Place: relación de herencia con Position.
- Robot: relación de composición con 'object' mediante propiedad 'carries'.
- Object: relación de composición con 'Place' mediante la propiedad 'has_place'.
- Place: relación de composición con 'Position' mediante la propiedad 'has_position'.



Conclusión para mejorar la eficiencia de los agente:

En la simulación de robots para organizar un almacén, se destaca la importancia de métricas de éxito como el tiempo total, acciones realizadas, colisiones evitadas y precisión en la identificación de objetos. Para mejorar la eficiencia, proponemos:

- optimización de rutas con algoritmos especializados
- la distribución de tareas basadas en roles
- la cooperación entre robots mediante protocolos de comunicación
- el uso de algoritmos de aprendizaje automático y la gestión dinámica de colisiones
- estrategias para reducción de tiempos, evitando obstáculos, optimizando recursos y mejorando el desempeño de los agentes en la simulación

