Instituto Tecnológico y de Estudios Superiores De Monterrey

Campus Monterrey

Escuela de Ingeniería y Ciencias

Modelación de sistemas multiagentes con gráficas computacionales (Gpo 103)

Interconexión de dispositivos (Gpo 302)

Nombre del profesor(a):

Raul V. Ramírez Velarde, Alfredo Alan Flores Saldivar

Rogelio Jesús Villarreal De Ochoa - A00838563

Fecha de entrega

19 de Agosto de 2025

Explicación de código

Para entender cómo adaptamos el agente para cumplir la función de recoger la tierra, partimos del modelo base de los agentes visto en clase. A partir de ese modelo, agregamos comportamientos personalizados para que, en nuestro caso, el agente busque y limpie la tierra.

La entrega busca cumplir con los requerimientos discutidos previamente: lo importante era el rastreo y limpieza de tierra, por lo que ese fue el objetivo final del programa.

Elementos importados del trabajo previo en clase

- Bibliotecas: agentpy, numpy (random), matplotlib, IPython.display
- Módulos y clases: my_plot, DummyAgent, DummyModel

Base del modelo

Como base, tenemos un agente que se mueve en una dirección fija. Sobre este modelo se hicieron modificaciones y se agregaron nuevas funciones.

Clase DummyModel

setup

Se definen height y width como dimensiones de la matriz para la generación de tierra. Para representar los lugares con tierra se crea una matriz copia, agregando dos tierras iniciales. Todas las coordenadas con tierra se almacenan en una lista de tuplas para identificar los lugares sucios.

getDirtDirection

A partir de la posición actual del agente se determina la coordenada sucia más cercana. Esto se hace recorriendo la lista de coordenadas sucias y eligiendo la de menor distancia.

roombaMove

Se llama a getDirtDirection para obtener la diferencia entre la posición actual y la coordenada más cercana. El método devuelve 0, 1 o -1 dependiendo de la diferencia en cada eje. Este resultado se almacena en self.execute como comportamiento de movimiento.

generateRandomDirt

Usando random y las dimensiones de la grilla se generan posiciones aleatorias. Las posiciones resultantes se marcan como 1 en la cuadrícula y se agregan a dirty_coords.

cleanDirt

Si el agente está en una celda con tierra, se asigna valor 0 a esa posición en la cuadrícula.

step

Se definió el siguiente orden de ejecución: primero limpiar la tierra, después moverse y finalmente generar tierra aleatoria. Este orden hace que, si no hay tierra, el agente se detenga por un tick, aunque el comportamiento se puede personalizar según las necesidades.