

Instituto Tecnológico y de Estudios Superiores de Monterrey



Modelación de sistemas multiagentes con gráficas computacionales

Grupo 104

Actividad integradora 1 – Sistemas multiagentes con GC

Equipo:

Luis Ángel Guzmán Iribe - A01741757

Cesar Galvez Cruz- A01252177

Antonio López Chávez - A01741741

Sebastián Gálvez Trujillo - A01251884

Profesores:

María Angélica Barreda Beltrán

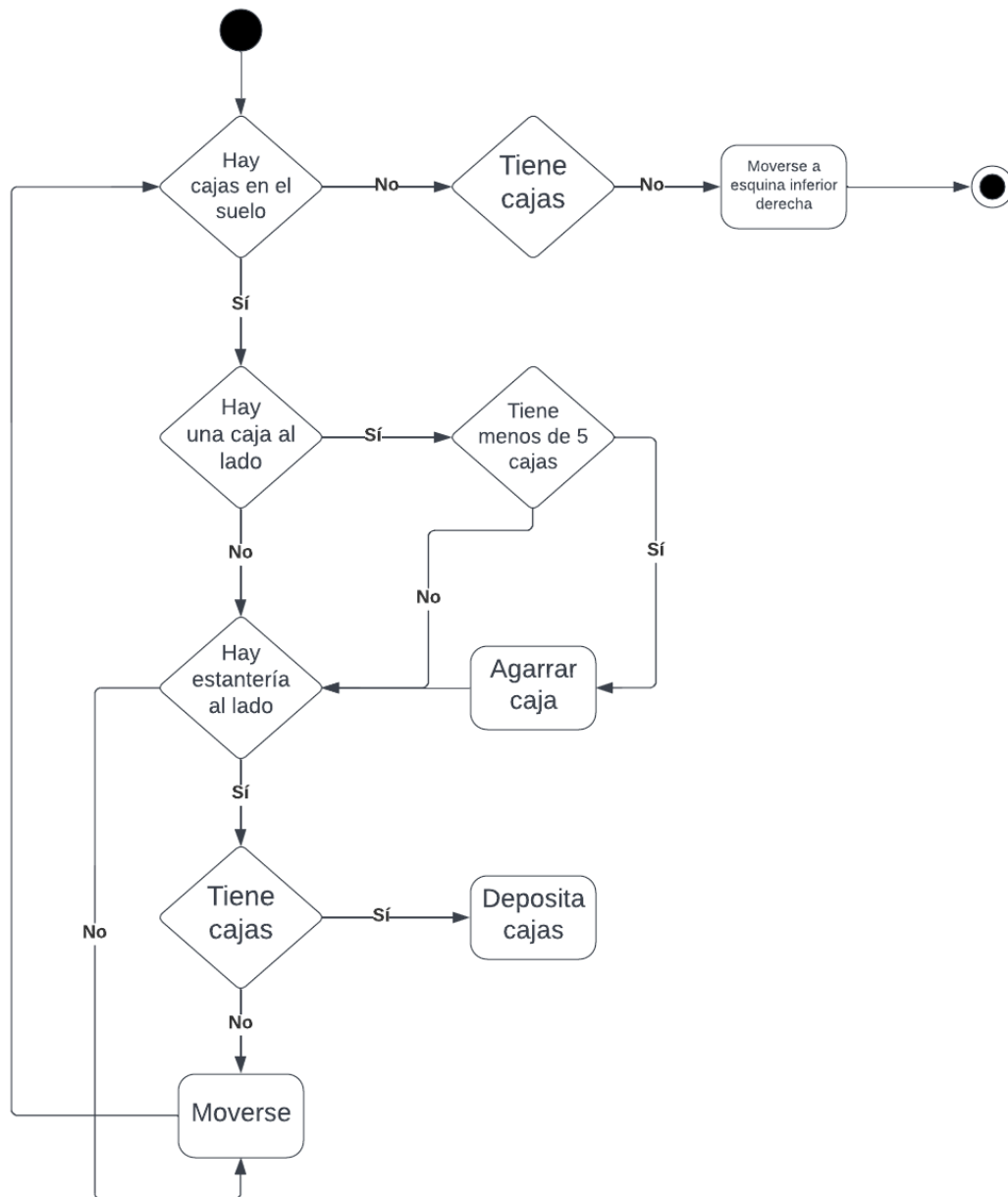
Jorge Mario Cruz Duarte

Fecha de entrega: 6 de septiembre del 2022

## Diagramas de clase:

Los agentes que merecían tener un diagrama de estados son los “Robots”, pues tienen diversas decisiones por tomar a lo largo del recorrido, mientras que las cajas y estantería solo reciben acciones de los “Robots”, ya sea moverles o recibir objetos.

Robots:



## Protocolos de movimiento/acción:

### Robots:

- Se inician en posiciones vacías aleatorias dentro de la cuadrícula
- Únicamente pueden moverse hacia las casillas directamente adyacentes a su posición actual (excluyendo diagonales)
- Únicamente pueden moverse hacia casillas vacías, y no pueden coexistir con otro agente en la misma casilla
- Pueden cargar un máximo de 5 cajas al mismo tiempo
- Cuando ya portan 5 cajas o no hay más cajas en el suelo de la cuadrícula, la llevan a la estantería
- Cuando ya no hay cajas en la cuadrícula, los robots se estacionan en la esquina inferior derecha

### Cajas:

- Se inician en posiciones vacías aleatorias dentro de la cuadrícula
- Pueden coexistir en el mismo espacio que otros agentes, aunque dada nuestra implementación este caso no se presenta.

### Estanterías:

- Se inician desde la esquina superior izquierda, abarcando la primera fila.
- Cada estantería puede contener hasta 5 cajas.
- Siempre que haya espacio en la primera fila, se generarán suficientes estanterías para el total de cajas.

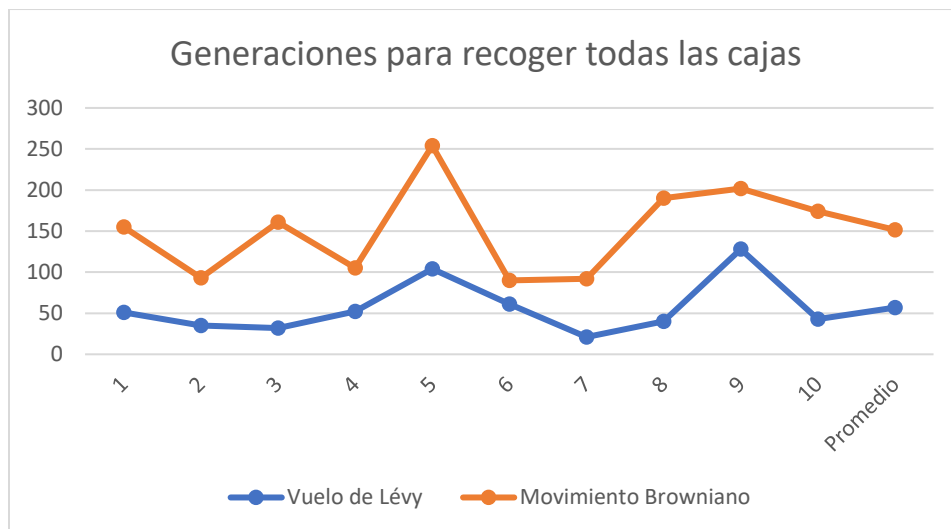
### Vuelo de Lévy:

La implementación del vuelo de Lévy para este programa funciona seleccionando una casilla aleatoria en la cuadrícula y moviendo al agente hasta esa dirección sin interrupciones. Esta implementación, reduce considerablemente el número de movimientos de los agentes y la cantidad de generaciones necesarias para recoger todas las cajas del piso. El movimiento browniano o aleatorio, en comparación, presenta un mayor número de movimientos y obtiene tiempos de finalización más largos.

Cantidad de movimientos necesarios para recoger todas las cajas (Vuelo de Levy VS Movimiento Browniano)



Cantidad de generaciones necesarias para recoger todas las cajas (Vuelo de Levy VS Movimiento Browniano)



## Estrategia cooperativa:

Para resolver la situación problema se necesitó analizar las reglas y movimientos exactos de cada robot, creando así el diagrama de estados/flujo correspondiente, con el cual se empezó a programar un comportamiento fiel a la teoría en Python. La creación de los agentes de cajas y estanterías fue sencilla pues son casi estáticos, pero los robots necesitaron de más tiempo.

Para el Unity no fue necesario seguir el comportamiento completo, solo que se represente de forma más llamativa y “sencilla”, sin mucho trabajo en la inteligencia de los agentes, utilizando el concepto de “waypoints”, los cuales permitieron crear los caminos necesarios para que los robots consigan las cajas y las almacenen en las estanterías.

La mayor parte del trabajo fue pulir detalles para que el movimiento sea correcto y evitar errores que puedan comprometer la entrega.