

Agentes Integilentes

Proyecto de Simulación

MatCom - UH

Autor: Leynier Gutiérrez González

Grupo: C412

Correo: l.gutierrez@estudiantes.matcom.uh.cu

Tutor: Dr. Yudivian Almeida, Universidad de La Habana

Código: github.com/leynier/agents

Descripción del Problema

El ambiente en el cual intervienen los agentes es discreto y tiene la forma de un rectángulo de $N \times M$. El ambiente es de información completa, por tanto todos los agentes conocen toda la información sobre el agente. El ambiente puede variar aleatoriamente cada t unidades de tiempo. El valor de t es conocido.

Las acciones que realizan los agentes ocurren por turnos. En un turno, los agentes realizan sus acciones, una sola por cada agente, y modifican el medio sin que este varíe, que cambie por una acción de los agentes. En el siguiente, el ambiente puede variar. Si es el momento de cambio del ambiente, ocurre primero el cambio natural del ambiente y luego la variación aleatoria.

En una unidad de tiempo ocurren el turno del agente y el turno de cambio del ambiente.

Los elementos que pueden existir en el ambiente son obstáculos, suciedad, niños, el corral y los agentes que son llamados Robots de Casa. A continuación se precisan las características de los elementos del ambiente:

Obstáculos

Estos ocupan una única casilla en el ambiente. Ellos pueden ser movidos, empujándolos, por los niños, una única casilla. El Robot de Casa sin embargo no puede moverlo. No pueden ser movidos ninguna de las casillas ocupadas por cualquier otro elemento del ambiente.

Suciedad

La suciedad es por cada casilla del ambiente. Sólo puede aparecer en casillas que previamente estuvieron vacías. Esta, o aparece en el estado inicial o es creada por los niños.

Corral

El corral ocupa casillas adyacentes en número igual al del total de niños presentes en el ambiente. El corral no puede moverse. En una casilla del corral solo puede coexistir un niño. En una casilla del corral, que esté vacía, puede entrar un robot. En una misma casilla del corral pueden coexistir un niño y un robot solo si el robot lo carga, o si acaba de dejar al niño.

Niño

Los niños ocupan solo una casilla. Ellos en el turno del ambiente se mueven, si es posible (si la casilla no está ocupada: no tiene suciedad, no está el corral, no hay un Robot de Casa), y aleatoriamente (puede que no ocurra movimiento), a una de las casilla adyacentes. Si esa casilla está ocupada por un obstáculo este es empujado por el niño, si en la dirección hay más de un obstáculo, entonces se desplazan todos. Si el obstáculo está en una posición donde no puede ser empujado y el niño lo intenta, entonces el obstáculo no se mueve y el niño ocupa la misma posición.

Los niños son los responsables de que aparezca la suciedad. Si en una cuadrícula de 3 por 3 hay un solo niño, entonces, luego de que él se mueva aleatoriamente, una de las casillas de la cuadrícula anterior que esté vacía puede haber sido ensuciada. Si hay dos niños se pueden ensuciar hasta 3. Si hay tres niños o más pueden resultar sucias hasta 6.

Los niños cuando están en una casilla del corral, ni se mueven ni ensucian.

Si un niño es capturado por un Robot de Casa tampoco se mueve ni ensucia.

Robot de Casa

El Robot de Casa se encarga de limpiar y de controlar a los niños. El Robot se mueve a una de las casillas adyacentes, las que decida. Solo se mueve una casilla sino, carga un niño. Si carga un niño puede moverse hasta dos casillas consecutivas.

También puede realizar las acciones de limpiar y cargar niños. Si se mueve a una casilla con suciedad, en el próximo turno puede decidir limpiar o moverse. Si se mueve a una casilla donde está un niño, inmediatamente lo carga. En ese momento, coexisten en la casilla Robot y niño.

Si se mueve a una casilla del corral que está vacía, y carga un niño, puede decidir si deja esta casilla o se sigue moviendo. El robot puede dejar al niño que carga en cualquier casilla. En ese momento cesa el movimiento del Robot en el turno, y coexisten hasta el próximo turno, en la misma casilla, Robot y niño.

Objetivos

El objetivo del Robot de Casa es mantener la casa (a.k.a el ambiente) limpia. Se considera la casa limpia si el 60 % de las casillas vacías no están sucias. Se sabe que si la casa llega al 60 % de casillas sucias el Robot es despedido e inmediatamente cesa la simulación. Si el Robot ubica a todos los niños en el corral y el 100 % de las casillas están limpias también cesa la simulación. Estos son llamados estados finales.

Debe programar el comportamiento del robot por cada turno así como las posibles variaciones del ambiente.

Ambiente inicial

Como ambiente inicial se especifica el tamaño del ambiente, el porcentaje de casillas que aparecen sucias, el porcentaje de obstáculos y el número de niños. El Robot de Casa parte de una posición aleatoria y es el que realiza el primer turno. Igual, se especifica el valor del tiempo de unidades de cambio (t). Con estos datos se genera un ambiente inicial que cumpla las restricciones previamente planteadas en el Marco General. El ambiente inicial debe ser factible.

En caso de que no se logre uno de los estados finales del ambiente, la simulación debe detenerse cuando hayan transcurrido 100 veces t .

Principales Ideas

Para la implementación y resolución del problema se utilizó el lenguaje de programación [Python](#) en su versión [3.8.5](#) y se alojó en la plataforma de desarrollo colaborativo [GitHub](#). Se utilizaron varias bibliotecas como [Poetry](#) (para el manejo de entornos de Python y dependencias), [Material Mkdocs](#) (para la generación de la documentación basada en Markdown), [Typer](#) (para la implementación de la interfaz de línea de comandos), entre otras. El editor de texto que se utilizó fue [Visual Studio Code](#). Todo lo anterior de software de código abierto y gratuito.

El entorno de la habitación se decidió modelar como una matriz. Las casillas pueden estar libres, con suciedad, con un obstáculo o ocupada por un corral. Las acciones de los bebés consisten en moverse arriba, a la derecha, a la izquierda, abajo o quedarse en el lugar. Las acciones de los robots consisten en las mismas antes mencionadas para los bebés además de limpiar, soltar al bebé, moverse en las cuatro diagonales y moverse más deprisa (avanzando dos casillas, solo cuando tiene un bebe cargado).

Modelos de Agentes

Se modelaron tres agentes, uno para el concepto de bebé y los dos restantes para el concepto de robot.

Agente Random Bebé

Este agente se comporta de manera aleatoria, como su nombre indica. En los casos en que se encuentre en un corral o esté siendo cargado por un robot la acción correcta es no realizar ninguna acción, en las restantes circunstancias elegirá una de entre las 5 acciones de manera aleatoria en la que cada acción tiene la misma probabilidad de ser seleccionada.

Agente Random Robot

Al igual que el Agente Random Beb , el Agente Random Robot escoger  de manera aleatoria una de las acciones posibles a realizar en el momento dado seg n como se encuentre el entorno.

Agente Robot Basado en Modelo

La implementaci n del agente pretende seguir la reactiva basada en modelos. Esto quiere decir que dado los estados del entorno y del modelo, tomar  una decisi n que ejecutar  en ese momento.

El modelo de este agente cuenta con los siguientes datos:

- **time:** Entero indicador de la cantidad de turnos que el agente ha estado realizando acciones.
- **visited:** Diccionario que indica para cada casilla el  ltimo turno en que fue visitada por el agente. De no haber sido visitada el valor de esta casilla es 0.

Luego de realizar la acci n el agente modifica los datos del modelo. De haberse movido a una casilla este actualizar  el tiempo con el time actual. La variable time incrementar  en uno su valor al finalizar el turno. Ya que el agente conoce el tiempo de variaci n aleatoria del ambiente al ocurrir esta el agente limpiar  todos sus registros de casillas visitadas marc ndolas como no visitadas. Esto se hace para reflejar que el ambiente que ten a recorrido antes ya no es el mismo.

Las reglas b sicas de comportamiento de este agente son las siguientes:

1. Si el agente se encuentra cargando un beb  y no est  en un corral entonces intentar  llevarlo al corral m s cercano.
2. Si el agente se encuentra cargando un beb  y est  en un corral entonces soltar  al beb .
3. Si el agente se encuentra en una casilla sucia entonces la limpiar .
4. Si el agente se encuentra en una casilla limpia o de corral entonces se mover  a la casilla marcada con el menor tiempo en su diccionario visited. La revisi n de estas casillas ser  siguiendo el orden horario.

Experimentos

Usando los modelos de agentes anteriores se confeccionaron varios escenarios en los cuales evaluar a los agentes. Se confeccionaron diez escenarios en total. Por cada escenario se corre la simulaci n 30 veces y se toman los resultados finales. En todos los escenarios se prob  con un  nico agente. Los escenarios son generados de forma aleatoria a partir de los par metros siguientes:

- A. Cantidad de filas
- B. Cantidad de columnas
- C. Cantidad de beb s
- D. Cantidad de obst culos
- E. Cantidad de casillas sucias iniciales
- F. Tiempo de variaci n aleatoria

Las configuraciones de parámetros por escenarios usados fueron las siguientes:

Escenario	A	B	C	D	E	F
1	5	5	2	5	5	10
2	7	7	3	15	10	10
3	10	10	4	20	20	10
4	12	12	5	25	30	15
5	14	14	3	15	10	15
6	16	16	4	20	20	15
7	18	18	5	25	30	20
8	20	20	3	15	10	20
9	22	22	4	20	20	20
10	25	25	5	25	30	25

Agente Random Robot

Escenario	Victorias	Empates	Derrotas	Promedio de casillas sucias
1	0	0	30	8.33
2	0	0	30	13.63
3	0	0	30	30.4331.7
4	0	0	30	46.80
5	0	0	30	51.63
6	0	0	30	85.66
7	0	0	30	118.46
8	0	0	30	62.50
9	0	0	30	144.45
10	0	0	30	224.50

Agente Robot Basado en Modelo

Escenario	Victorias	Empates	Derrotas	Promedio de casillas sucias
1	6	0	24	6.66
2	1	0	29	12.96
3	1	0	29	31.70
4	0	0	30	46.63
5	4	6	20	72.53
6	0	3	7	93.26
7	0	0	30	118.73
8	1	24	5	153.40
9	0	9	21	184.26
10	0	3	27	238.46

Conclusiones

Analizando los resultados de los experimentos se evidencio que el agente basado en una estrategia aleatoria no es una solución efectiva, sin embargo en agente reactivo basado en modelo tuvo un mejor desempeño, pero tampoco se puede calificar de bueno. Otras alternativas analizadas fueron utilizar un sistema de inferencia difusa para describir las reglas y estrategias que debía seguir un agente, se decidió dejar para futuras versiones del estudio.