

House-Cleaner

- **Luis Alejandro Lara Rojas (C-412)**

Orden del problema asignado

Se quiere simular un ambiente rectangular donde conviven varios elementos:

- **Obstáculos:** se encuentran en cualquier posición del ambiente y son movibles solo por los niños.
- **Corrales:** Constituyen casillas donde se puede ubicar a los niños para evitar que se muevan o ensucien.
- **Suciedad:** Son ocasionadas por los niños en cada turno en posiciones aleatorias a su alrededor. Pueden ser eliminadas por algún robot que se encuentre sobre ella.
- **Niños:** En cada turno ensucian y se mueven si no están siendo cargados por un robot, o se encuentran en un corral.
- **Robots:** Son los encargados de limpiar y llevar a los niños a los corrales de tal manera que se mantenga la casa lo más limpia posible.

Las acciones se realizan por turnos. En cada turno todos los robots ejecutan sus acciones, y varía el ambiente. La variación del ambiente consiste en las acciones aleatorias de los niños.

Principales ideas seguidas para la solución del problema

Para la solución de este problema se ha implementado una ejecución de acciones por turno donde en cada turno se realizan por orden las siguientes acciones:

- Los robots deciden cual va a ser su proxima acción y la realizan inmediatamente.
- Aparecen nuevas casillas con suciedad resultado de los niños.
- Los niños se mueven.

Para la toma de decisión de los robots existen funcionalidades como el algoritmo de BFS para la búsqueda de caminos sin obstáculos. En dependencia del modelo de agente utilizado para el robot, este toma una decisión respecto a cuál será su próxima acción. Esta acción puede ser limpiar, moverse o cargar a un niño.

Para las nuevas casillas con suciedad, se realiza un recorrido por cada niño que no esté siendo cargado por un robot o que no esté encima de un corral, y se genera una suciedad en alguno de los 6 espacios adyacentes.

En el caso del movimiento del niño, este primero elige a donde moverse, y en caso de ser posible el movimiento, lo realiza empujando la cadena de obstáculos que pueda existir en esa dirección.

Modelos de Agentes Considerados

Para la ejecución de la simulación se utilizan 3 modelos de agentes:

- Uno más proactivo, el cual se traza objetivos específicos: intenta primeramente alcanzar algún niño, si ya está cargando uno intenta llevarlo para un corral, y si no hay niños libres intenta limpiar la suciedad.

- Uno más reactivo, que ejecuta las acciones en dependencia de como esté el ambiente en ese momento. Si se encuentra en una casilla que está sucia, la limpia. Si se encuentra en una casilla con un niño, lo recoge. Si tiene a un niño cargado, intenta dejarlo en algún corral. De otra forma, se mueve aleatoriamente o permanece en el lugar en caso de no poder moverse.
- Uno equilibrado. Este último se traza objetivos a cumplir en el momento, en dependencia de cómo se encuentre el ambiente. Si se encuentra encima de una suciedad, la limpia. Luego, si hay niños sueltos busca la ruta más corta a un niño y se mueve por esa dirección. Si no hay ruta posible hacia ningún niño suelto, busca la ruta más corta posible a una casilla sucia, y se mueve en esa dirección. En otro caso, permanece en su posición.

Ideas seguidas para la implementación

El proyecto se encuentra dividido en los siguientes módulos:

- Corral: contiene la función findCorral que retorna True si existe un corral en una posición indicada.
- Dirt: contiene todas las funcionalidades referentes a las casillas con suciedad. Se encuentran las funcionalidades de crear suciedades aleatorias, encontrar suciedades en alguna posición, etc.
- Environment: en este módulo se encuentra lo referente a la ejecución de las acciones de los agentes y las variaciones del ambiente por cada turno.
- EnvironmentCases: contiene 9 casos de prueba de simulación. Son 3 objetos de tipo Environment, y cada uno tiene 3 casos de prueba: uno por cada modelo de Agente. Si se desea agregar otro caso de prueba se puede agregar a este módulo.
- Kid: contiene las funcionalidades relacionadas con los niños. Es aquí donde se mueven los niños aleatoriamente en cada turno, donde se busca si hay un niño en cierta posición, etc.
- Lib: en este módulo se comienza la ejecución de la simulación, y es donde se ejecutan todos los turnos. En la función start se puede cambiar el número 5, que representa la cantidad de turnos que se van a ejecutar.
- Obstacle: contiene fundamentalmente las funcionalidades que se utilizan a la hora de empujar obstáculos.
- Robot: es el módulo que contiene todos los procedimientos realizados por los Agentes. Es aquí donde se hacen todas las búsquedas y se ejecutan todas las posibles acciones de estos.
- Types: contiene todos los tipos usados en el programa.
- Utils: encierra todas las funcionalidades complementarias usadas en los módulos principales.

En este proyecto existen varios tipos:

- El tipo Position: es una tupla (fila, columna).
- Los tipos de objetos: Corral, Dirt, Obstacle, Kid y Robot. Corral, Dirt y Obstacle están formados por una posición(fila, columna), mientras que Kid y Robot tienen además un valor Bool que dice si está cargado o con carga, respectivamente.
- El tipo EnvironmentState: es una tupla de 5 elementos que contiene las 5 listas de objetos del ambiente: corral, suciedad, obstáculo, niño y robot.
- El tipo Direction: es un String que indica una dirección(norte, sur, este u oeste).
- El tipo Activity: que denota el modelo de agente utilizado(reactivo, proactivo o pro-reactivo).

Consideraciones obtenidas

Luego de ejecutar la simulación en repetidas ocasiones de cada uno de los diferentes escenarios con cada modelo de Agente, se puede llegar a la conclusión de que el modelo más eficiente fue el que equilibra entre modo reactivo y proactivo. En la gran mayoría de las veces, este modelo brindó una solución más eficiente que los otros dos modelos para el mismo ambiente inicial.

El nivel de limpieza obtenida al terminar la simulación casi todos los casos fue superior al 60%. No fue así para el caso de los agentes proactivos y reactivos, en los cuales para una cantidad considerable de turnos, el nivel de limpieza quedaba por debajo del 40%.