

# Proyecto de Simulación y Programación Declarativa

## Agentes-Haskell

ALEJANDRO CAMPOS. C-411

Facultad de Matemática y Computación  
Universidad de la Habana  
2021

### I. INTRODUCCIÓN

Un agente es un sistema computacional situado dentro de un medio ambiente, dentro del cual es capaz de realizar acciones autónomas encaminadas a lograr sus objetivos. Un agente inteligente es aquel que es capaz de realizar autónomas flexibles para lograr sus objetivos, donde flexible significa: reactivo, pro-activo y sociable. Donde reactivo significa que capaz de percibir su ambiente y responder de un modo oportuno a los cambios que ocurren para lograr sus objetivos. Proactivo quiere decir que debe ser capaz de mostrar un comportamiento dirigido a objetivos tomando la iniciativa para lograr sus objetivos. Por último, un agente sociable debe ser capaz de interactuar con otros agentes para lograr sus objetivos.

El problema de agentes en el cual se centra el proyecto se detalla a continuación:

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 a no ser 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 por ninguna de las casillas ocupadas por cualquier otro elemento del ambiente.

**Suciedad:** La suciedad es por cada casilla del ambiente. Solo 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 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 esté vacía, y carga un niño, puede decidir si lo 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.

El objetivo del Robot de Casa es mantener la casa limpia. Se considera la casa limpia si el 60% de las casillas vacías no están sucias.

Para la implementación del problema anterior se usó como lenguaje de programación Haskell, que es un lenguaje de programación puramente funcional, estandarizado multi-propósito, con evaluación no estricta y memorizada, y fuerte tipificación estática. Los programas escritos en Haskell se representan siempre como funciones matemáticas, pero estas funciones nunca tienen efectos secundarios ni derivados. De este modo, cada función utilizada siempre devuelve el mismo resultado con la misma entrada, y el estado del programa nunca cambia. Por esto, el valor de una expresión o el resultado de una función dependen exclusivamente de los parámetros de entrada en el momento. En Haskell no pueden hacerse construcciones de lenguaje imperativo para programar una secuencia de declaraciones.

Por todo lo expuesto anteriormente, el objetivo de este informe es explicar el método de resolución utilizado para resolver el problema mencionado. Además, explicaremos los detalles de implementación y ejecución de la aplicación de consola que se ha creado, no sin antes abordar los distintos modelos de agentes utilizados y cómo se adaptaron a este problema en particular. Por último, haremos un análisis de los resultados obtenidos a partir de la ejecución de las simulaciones del problema.

## II. IDEAS PRINCIPALES PARA LA SOLUCIÓN DEL PROBLEMA

Para tratar este problema se siguieron las indicaciones de las conferencias de la asignatura Simulación y del libro "Temas de simulación", capítulo 6, los cuales, a muy groso modo, plantean que el agente recibe estímulos de su medio ambiente y realiza acciones sobre este. En secciones posteriores formalizaremos esta idea. Primeramente, en esta sección, realizaremos una interpretación del problema en cuestión y expondremos las ideas seguidas para su solución.

El problema en cuestión nos plantea que el ambiente es dinámico, es decir, que cada cierto tiempo  $t$  puede variar aleatoriamente. Esta variación del ambiente viene dada por el movimiento de los niños, los cuales pueden moverse a una casilla adyacente, empujar obstáculos y ensuciar el ambiente. Esto implica que el agente que se diseñe para controlar el robot no puede ser proactivo, pues cualquier estrategia planteada probablemente quedaría estropeada antes de cumplirse. En los modelos pro-activos se asume que el ambiente no va a cambiar mientras el procedimiento se ejecuta. Si el ambiente cambia y las precondiciones se vuelven falsas durante el proceso, entonces el comportamiento del procedimiento se indefiniría o simplemente falla. También se asume que durante la ejecución el objetivo sigue siendo válido hasta que este termine, pero si este deja de ser válido no hay razón para seguir ejecutando el procedimiento. En nuestro problema si seguimos el objetivo de capturar un niño, este puede dejar de ser visible para los agentes porque, por ejemplo, otro niño interpuso un obstáculo en el camino.

En nuestro ambiente el agente tiene información completa y actualizada sobre el estado del medio. Además, su determinismo garantiza que cualquier acción tiene un único efecto, es decir, no hay incertidumbre sobre el estado en el que quedará el ambiente después de realizar una acción. En este problema, el agente debe decidir solamente en base al episodio actual, no tiene que razonar las consecuencias en episodios futuros cualquier estrategia se verá condenada a fracasar en la mayoría de los casos por la aleatoriedad de cambio del ambiente.

En estos ambientes dinámicos, el agente debe ser reactivo, o bien una combinación entre reactivo y proactivo. Esto es, que debe ser sensitivo a los eventos que ocurran en el ambiente, donde estos eventos afectan los objetivos del agente o las suposiciones en las que se basa el proceso que el agente está ejecutando para lograr sus objetivos. Por otro lado, no queremos que nuestro agente esté constantemente reaccionando y por lo tanto nunca se enfoque en un objetivo el tiempo necesario para lograrlo. Es por ello que debemos construir sistemas que consigan un balance efectivo entre estos comportamientos.

Por lo tanto, las mejores estrategias tendrán que ser una combinación entre agente reactivo y proactivo, dado que el reactivo no podrá planificar a más de un paso y el proactivo verá sus planes atrofiados con el dinamismo del ambiente.

Finalmente, podemos afirmar que existen en nuestro ambiente un número fijo y finito de percepciones y acciones que lo pueden modificar. Ya sea por parte de los niños, como ya vimos, o por parte de los agentes, cuyas acciones y percepciones se definen en la siguiente sección.

## III. MODELOS DE AGENTES CONSIDERADOS

Como vimos en la sección anterior, nuestro ambiente está definido por un conjunto de estados  $E = \{s_1, s_2, s_3, \dots\}$ , que se generan producto de un cambio aleatorio dado por el movimiento de un niño (que genera suciedad o mueve obstáculos). También los agentes modifican el ambiente con las acciones que estos pueden realizar sobre él. Dado un estado  $s_i$  del ambiente, si aplicamos una acción del agente obtenemos un estado  $s_{i+1}$  producto de aplicar la acción  $a_i$  al estado  $s_i$ . Nuestros

agentes son capaces de realizar las siguientes acciones:

- Moverse a una casilla vacía, con niño, con corral vacío o con suciedad.
- Moverse a casillas adyacentes solo.
- Cargar niño.
- Moverse a casillas adyacentes, uno o dos pasos, con niño cargado.
- Dejar niño en cualquier casilla de las que puede moverse.
- Limpiar una casilla sucia.

Los agentes captan la habilidad de observar el ambiente, por lo que se definen las siguientes percepciones que los agentes tendrán sobre el tablero:

- Camino al niño más cercano.
- Camino a la suciedad más cercana.
- Camino al corral más cercano.
- Por ciento de limpieza del ambiente.

Luego, sabemos que el conjunto de estados del ambiente son aleatorios, por lo que no podemos trazar una estrategia a largo plazo. Los agentes cada vez que van a realizar una acción deben revisar las percepciones basados en el estado actual del ambiente para realizar una acción consecuente.

Definimos además en nuestros agentes dos estados, que cambian de acuerdo a las percepciones que el agente toma del ambiente:

- Cargando niño.
- Libre.

Las especificidades de cada modelo implementado se describen a continuación en las siguientes subsecciones:

## I. Robot Random