

# Simulación

## Agentes

# Programación Declarativa

Leandro González Montesino

Grupo 411

Roberto Zahuis Benítez

Grupo 411

Facultad de Matemática y Computación

La Habana, Cuba

26 de abril 2019

### **1- Resumen:**

En el siguiente informe nos encontraremos con la descripción de la problemática a Simular. Métodos utilizados, breve explicación desde un enfoque más práctico, análisis de los resultados y un punto de inflexión para futuros trabajos, enmarcando temas de interés según lo obtenido.

### **2- Breve descripción del Problema:**

El ambiente en el cual intervienen los agentes es discreto y tiene la forma de un rectángulo de  $N \times M$ . Ningún agente u objeto del ambiente puede salir del terreno (chocan contra las paredes que lo rodean). El ambiente es de información completa, por tanto, todos los agentes conocen toda la información sobre el ambiente. El ambiente varía aleatoriamente cada  $t$  unidades de tiempo. El valor de  $t$  es conocido.

Los elementos que pueden existir en el ambiente son obstáculos, suciedad, niños, el corral y los agentes que son llamados Robots de Casa.

### 3- Definición objetiva:

La situación que se presentará posteriormente a modelar estará basada en los pasos siguientes, manteniendo todos los requisitos de la orientación.

1. Simular el comportamiento aleatorio de los niños.
2. Simular el comportamiento del agente Robot de manera Reactiva.
3. Simular el comportamiento del agente Robot de manera Proactiva.
4. Calcular un estimado según los resultados obtenidos.

La implementación del suceso fue concebida en el lenguaje de programación declarativa Prolog. Llevándose todo el proceso puramente en el mismo, con la posibilidad de correr en cualquier sistema operativo con SWI-Prolog u otro compilador de Prolog.

### 4- Ideas Principales:

En la simulación de dicha problemática, se le brindo un gran peso a las funciones `assert` y `retract`, donde de esa manera se mantenga la base de conocimiento en todo momento según los elementos necesarios. Teniendo en cuenta los casos de agentes empleados en el robot se procede a explicar en cada caso:

4.1- En una primera instancia se utilizó una lógica mayormente reactiva en el agente, aprovechando algunos de sus conocimientos del terreno. Donde el interés de dicho agente es resolver el problema de la casilla donde se encuentra estático, donde su prioridad uno es cargar al niño, de segundo orden es limpiar y un tercer caso moverse a otra casilla aleatoriamente.

Una vez el robot tiene en control a un niño, utilizamos el algoritmo conocido, BFS para encontrar el corral más cercano donde será encerrado y una vez ahí regresa a su comportamiento usual.

4.2- Como segunda propuesta se utilizó una lógica puramente proactiva en el agente, donde su objetivo principal es buscar al niño más cercano al robot, una vez atrapado será encerrado en el corral más próximo a la posición actual de coexistencias de los mismos, todo resuelto mediante BFS.

Es importante plantear que mientras no estén todos los niños previamente colocados en su corral el robot no procederá a limpiar ninguna casilla.

## 5- Método de uso:

- 1- Consultar el fichero llamado **main.pl**, en el mismo se encuentras todas las consultas a los otros archivos utilizados, ya una vez ahí entrar el comando **start**. Y se realizara todo el proceso.

- 2- En el fichero **main.pl** se puede escoger que tipo de robot se prefiere simular:

- a. Descomentar la siguiente línea para robot Proactivo:

```
%:-consult(robotZahuis).
```

- b. Descomentar la siguiente línea para robot Reactivo:

```
%:-consult(robotLeandro).
```

- 3- En el archive **utils.pl** se modifican los valores iniciales de la entrada de la simulación deseados de la siguiente manera:

```
initialize_variables:-  
    T is 60,      % Tiempo t  
    N is 12,      % Dimensión N  
    M is 12,      % Dimensión M  
    DIRTY is 25,  % Cantidad de espacios sucios  
    OBSTACLE is 25, % Cantidad de Obstáculos  
    CHILDREN is 9, % Cantidad de niños
```

## 6- Simulación:

La simulación fue llevada a cabo en 10 ambientes iniciales, donde se evaluó el proceso por cada cual un numero de 30 veces, obteniéndose los siguientes resultados:

#Ambiente	N	M	%S	#Niños	#Obstáculos	Moda de estados Finales	T	SuciedadF	#Simulaciones	Tipo Robot
1	9	10	0	7	18	Tablas	50	25-28	30	P
2	9	10	0	7	18	Despido	50	+60	30	R
3	9	10	20	7	18	Despido	50	45-X (X>60)	30	P
4	9	10	20	7	18	Despido	50	+60	30	R
5	5	5	2	3	5	Gano	40	0	30	P
6	5	5	2	3	5	Despido	40	+60	30	R
7	7	7	5	5	6	Gano=Despido	40	30	30	P
8	7	7	5	5	6	Despido	40	+60	30	R
9	12	12	25	9	25	Tablas	60	40-50	30	P
10	12	12	25	9	25	Despido	60	+60	30	R