

# Proyecto de Simulación de Agentes en Prolog

---

Rayniel Ramos González C-412

Luis Ernesto Martínez Padrón C-412

## Principales ideas seguidas

---

El ambiente en el que va a funcionar el agente a implementar es totalmente dinámico, los niños tienen consciencia propia de qué van a hacer y pueden mover los obstáculos o ensuciar casillas en cualquier momento, lo que podría cambiar la estrategia planeada por el agente del robot. 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. Las mejores estrategias tendrán que ser una combinación entre agente reactivo y proactivo, dado que el reactivo no podrá planificar a más de 1 paso y el reactivo verá sus planes atrofiados con el dinamismo del mapa.

## Agentes considerados

---

### Modelo agente Random(Reactivo)

---

Este es el caso más sencillo, simplemente en cada instante de tiempo el agente decide cuál va a ser su próximo movimiento a aleatoriamente. En el caso de los niños que son parte del ambiente, esta fue la idea implementada teniendo en cuenta los detalles de que los niños, además de moverse, ensucian el ambiente. En el caso del agente del robot, también contamos con una implementación de este tipo pero por propósitos experimentales, o sea, para poder comparar con agentes con ideas menos inestables. Se tuvo en cuenta que el robot, cuando esté encima de una casilla sucia, inmediatamente la limpia y, de estar en una casilla que no sea de corral y que tenga a un niño, lo continúa cargando en un próximo movimiento.

### Modelo de Agente Niñera(Pro-reactivo)

---

En este caso se siguió la idea de que en el caso de que los niños estén todos en los corrales, no se podrá ensuciar más y, a partir de ahí, solo queda limpiar el ambiente. Es una estrategia un tanto arriesgada pues, como no se limpia nada mientras se pone a los niños en los corrales, la cantidad de suciedad puede llegar al punto en que se despida al robot antes de que comience a limpiar. Se dice que es un agente Pro-reactivo(o proactivo-reactivo) porque aunque planea que su próximo movimiento es ir al niño más cercano, el corral más cercano o la casilla sucia más cercana, el dinamismo del ambiente obliga a que la ruta a tomar pueda cambiar a medida que pasa el tiempo. Esto obliga a que en cada instante de tiempo se recalcule la ruta a tomar por el robot.

### Modelo de Agente Aspiradora(Pro-reactivo)

---

Este modelo solo le interesa la cantidad de suciedad en el ambiente. Lo más probable es que nunca gane, pero busca siempre perder en el mayor tiempo posible. El único caso posible en el que este robot gana es un caso, que tiene MUY BAJAS probabilidades, y es el caso en el que los niños por pura aleatoriedad, caigan en las casillas de corral ellos solos antes de que se sature el ambiente de suciedad, caso en el que daría tiempo al robot de limpiar el resto de la casa sin ser despedido, si la cantidad de tiempo hasta que cambie el mapa es suficiente también.

## Ideas Seguidas para la Implementación

La asignación del tipo de casillas deberán ser hechas mediante hechos del tipo `casilla(X,Y,...)`. donde como otros argumentos se podría tener un identificador o cualquier otra propiedad de lo que sea que se encuentre en las coordenadas X e Y. Todos los hechos que se utilicen para guardar posiciones deben ser cláusulas dinámicas dado que podrían no estar definidas en código o nunca haberse llamado antes, sin embargo, es necesario que el intérprete las asimile como válidas, ya que serán insertadas y removidas de la base de datos con frecuencia.

## Resultados

Se realizaron numerosos experimentos con los 2 últimos modelos de agente de robot, se utilizaron 10 tipos de ambiente para cada uno de los modelos y dentro de cada uno se hicieron 30 simulaciones por agente. De estas simulaciones se obtuvieron los siguientes resultados:

### Ambiente 1:

- Dimensiones: 10x30
- 3 niños
- 10 % de suciedad inicial
- 10 % de obstáculos
- Tiempo total: 600 iteraciones
- Cambio de entorno: 150 iteraciones

Condiciones	Robot Niñera	Robot Aspiradora
No fue despedido	30	24
Fue despedido	0	7
Terminó de limpiar	0	0
Suciedad restante (% promedio)	15.11	44.50

### Ambiente 2:

- Dimensiones: 29x42
- 4 niños
- 6 % de suciedad inicial
- 5 % de obstáculos
- Tiempo total: 600 iteraciones
- Cambio de entorno: 150 iteraciones

Condiciones	Robot Niñera	Robot Aspiradora
No fue despedido	30	30
Fue despedido	0	0
Terminó de limpiar	0	0
Suciedad restante (% promedio)	17.67	32.62

#### Ambiente 3:

- Dimensiones: 18x48
- 3 niños
- 3 % de suciedad inicial
- 3 % de obstáculos
- Tiempo total: 600 iteraciones
- Cambio de entorno: 150 iteraciones

Condiciones	Robot Niñera	Robot Aspiradora
No fue despedido	30	30
Fue despedido	0	0
Terminó de limpiar	0	0
Suciedad restante (% promedio)	10.91	29.60

#### Ambiente 4:

- Dimensiones: 8x15
- 4 niños
- 5 % de suciedad inicial
- 5 % de obstáculos
- Tiempo total: 600 iteraciones
- Cambio de entorno: 150 iteraciones

Condiciones	Robot Niñera	Robot Aspiradora
No fue despedido	22	5
Fue despedido	6	25
Terminó de limpiar	15	1
Suciedad restante (% promedio)	15.11	61.52

#### Ambiente 5:

- Dimensiones: 25x9
- 2 niños
- 7 % de suciedad inicial

- 3 % de obstáculos
- Tiempo total: 600 iteraciones
- Cambio de entorno: 150 iteraciones

Condiciones	Robot Niñera	Robot Aspiradora
No fue despedido	30	25
Fue despedido	0	5
Terminó de limpiar	15	0
Suciedad restante (% promedio)	2.10	36.44

#### Ambiente 6:

- Dimensiones: 30x10
- 4 niños
- 15 % de suciedad inicial
- 10 % de obstáculos
- Tiempo total: 600 iteraciones
- Cambio de entorno: 150 iteraciones

Condiciones	Robot Niñera	Robot Aspiradora
No fue despedido	29	13
Fue despedido	1	17
Terminó de limpiar	0	0
Suciedad restante (% promedio)	22.0	56.43

#### Ambiente 7:

- Dimensiones: 25x22
- 3 niños
- 8 % de suciedad inicial
- 12 % de obstáculos
- Tiempo total: 600 iteraciones
- Cambio de entorno: 150 iteraciones

Condiciones	Robot Niñera	Robot Aspiradora
No fue despedido	30	27
Fue despedido	0	3
Terminó de limpiar	0	0
Suciedad restante (% promedio)	13.82	40.56

### Ambiente 8:

- Dimensiones: 35x30
- 3 niños
- 9 % de suciedad inicial
- 3 % de obstáculos
- Tiempo total: 600 iteraciones
- Cambio de entorno: 150 iteraciones

Condiciones	Robot Niñera	Robot Aspiradora
No fue despedido	30	30
Fue despedido	0	0
Terminó de limpiar	0	0
Suciedad restante (% promedio)	15.42	28.47

### Ambiente 9:

- Dimensiones: 27x14
- 4 niños
- 12 % de suciedad inicial
- 8 % de obstáculos
- Tiempo total: 600 iteraciones
- Cambio de entorno: 150 iteraciones

Condiciones	Robot Niñera	Robot Aspiradora
No fue despedido	30	14
Fue despedido	0	16
Terminó de limpiar	0	0
Suciedad restante (% promedio)	19.07	57.49

### Ambiente 10:

- Dimensiones: 25x25
- 4 niños
- 14 % de suciedad inicial
- 15 % de obstáculos
- Tiempo total: 600 iteraciones
- Cambio de entorno: 150 iteraciones

Condiciones	Robot Niñera	Robot Aspiradora
No fue despedido	30	25
Fue despedido	0	5
Terminó de limpiar	0	0
Suciedad restante (% promedio)	22.70	45.60

Como interpretación a estos datos obtenidos se puede decir que, como era de esperarse, el robot que no solo limpia tiene mayor efectividad en la totalidad de los escenarios. También es importante decir que si se aumenta la cantidad de tiempo que transcurre antes de un cambio de mapa, el robot niñera aumenta casi un 40% sus victorias.

También es importante decir que al aumentar la cantidad de niños a más de 5, el robot aspiradora pierde en la mayoría de los casos y escenarios.