



Tecnológico de Monterrey

Materia:

Modelación de Sistemas Multiagentes con gráficas
computacionales

M1. Actividad: Robot de Limpieza Reactivo

Realizado por:

Iván Ricardo Paredes Avilez

Matrícula:

A01705083

Fecha:

12 de noviembre de 2023

Para un espacio de 100x100, considera los siguientes escenarios:

- Escenario 1: 1 agente, 90% de celdas sucias.

100 pasos:

```
Porcentaje inicial de suciedad: 90.0  
Porcentaje final de suciedad: 89.46  
Tiempo tomado: 0.197 segundos  
Número de pasos: 100
```

1000 pasos:

```
Porcentaje inicial de suciedad: 90.0  
Porcentaje final de suciedad: 87.31  
Tiempo tomado: 0.197 segundos  
Número de pasos: 1000
```

10000 pasos:

```
Porcentaje inicial de suciedad: 90.0  
Porcentaje final de suciedad: 58.04  
Tiempo tomado: 1.346 segundos  
Número de pasos: 10000
```

- Escenario 2. 2 agentes, 90% de celdas sucias.

100 pasos

```
Porcentaje inicial de suciedad: 90.0  
Porcentaje final de suciedad: 89.07  
Tiempo tomado: 0.22 segundos  
Número de pasos: 100
```

1000 pasos

```
Porcentaje inicial de suciedad: 90.0  
Porcentaje final de suciedad: 83.28  
Tiempo tomado: 0.238 segundos  
Número de pasos: 1000
```

10000 pasos

```
Porcentaje inicial de suciedad: 90.0  
Porcentaje final de suciedad: 44.54  
Tiempo tomado: 1.67 segundos  
Número de pasos: 10000
```

¿Cuántos pasos de simulación toma limpiar todo el espacio?

Si cambiamos el bucle que ejecuta la simulación, por uno que solamente espera a que el porcentaje de suciedad llegue a 0, podemos saber cuántos pasos se necesitan. El nuevo código sería el siguiente:

```
while model.get_dirty_percentage() > 0:  
    model.step()
```

- Para 1 agente y el 90% de celdas sucias:

```
Porcentaje inicial de suciedad: 90.0  
Porcentaje final de suciedad: 0.0  
Tiempo tomado: 13.749 segundos  
Número de pasos: 266693
```

Se necesitan más de 260,000 pasos

- Para 2 agentes y el 90% de celdas sucias:

```
Porcentaje inicial de suciedad: 90.0  
Porcentaje final de suciedad: 0.0  
Tiempo tomado: 7.702 segundos  
Número de pasos: 123887
```

Se necesitan más de 123,000 pasos

Cantidad óptima de aspiradoras:

Considerando un máximo de 10 aspiradoras disponibles, podemos ejecutar el código nuevamente pero cada vez con más aspiradoras para determinar cuál es la cantidad óptima para tardar el menor tiempo posible.

- Para 1 agente y el 90% de celdas sucias:

```
Porcentaje inicial de suciedad: 90.0  
Porcentaje final de suciedad: 0.0  
Tiempo tomado: 13.749 segundos  
Número de pasos: 266693
```

Tiempo: 13.749s

- Para 2 agentes y el 90% de celdas sucias:

```
Porcentaje inicial de suciedad: 90.0  
Porcentaje final de suciedad: 0.0  
Tiempo tomado: 7.702 segundos  
Número de pasos: 123887
```

Tiempo: 7.702s

- Para 3 agentes y el 90% de celdas sucias:

```
Porcentaje inicial de suciedad: 90.0  
Porcentaje final de suciedad: 0.0  
Tiempo tomado: 6.794 segundos  
Número de pasos: 98643
```

Tiempo: 6.794s

- Para 4 agentes y el 90% de celdas sucias:

```
Porcentaje inicial de suciedad: 90.0  
Porcentaje final de suciedad: 0.0  
Tiempo tomado: 5.129 segundos  
Número de pasos: 55616
```

Tiempo: 5.129s

- Para 5 agentes y el 90% de celdas sucias:

```
Porcentaje inicial de suciedad: 90.0  
Porcentaje final de suciedad: 0.0  
Tiempo tomado: 4.132 segundos  
Número de pasos: 44885
```

Tiempo: 4.132s

- Para 6 agentes y el 90% de celdas sucias:

```
Porcentaje inicial de suciedad: 90.0  
Porcentaje final de suciedad: 0.0  
Tiempo tomado: 3.677 segundos  
Número de pasos: 36493
```

Tiempo: 3.677s

- Para 7 agentes y el 90% de celdas sucias:

```
Porcentaje inicial de suciedad: 90.0  
Porcentaje final de suciedad: 0.0  
Tiempo tomado: 3.137 segundos  
Número de pasos: 27250
```

Tiempo: 3.137s

- Para 8 agentes y el 90% de celdas sucias:

```
Porcentaje inicial de suciedad: 90.0  
Porcentaje final de suciedad: 0.0  
Tiempo tomado: 4.064 segundos  
Número de pasos: 34930
```

Tiempo: 4.064s

- Para 9 agentes y el 90% de celdas sucias:

```
Porcentaje inicial de suciedad: 90.0  
Porcentaje final de suciedad: 0.0  
Tiempo tomado: 3.408 segundos  
Número de pasos: 26548
```

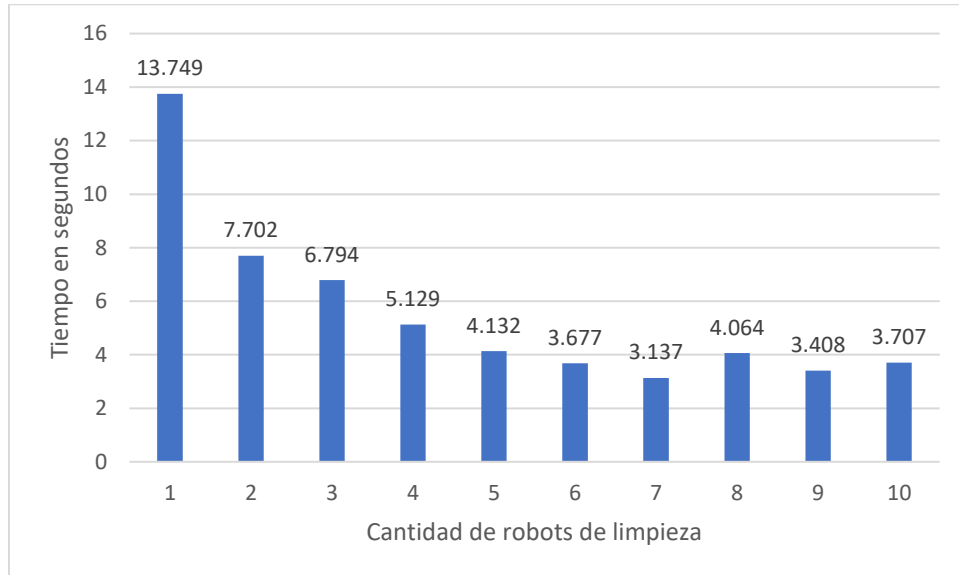
Tiempo: 3.408s

- Para 10 agentes y el 90% de celdas sucias:

```
Porcentaje inicial de suciedad: 90.0
Porcentaje final de suciedad: 0.0
Tiempo tomado: 3.707 segundos
Número de pasos: 26610
```

Tiempo: 3.707s

Si ponemos estos datos en una gráfica, obtenemos lo siguiente:



Después de hacer esta experimentación, podemos decir que la cantidad de robots óptima para limpiar todo el cuarto es de 7, pues fue la ejecución más rápida de todas. Cabe decir que, al tener igualmente un comportamiento aleatorio, es posible que que la cantidad de tiempo fluctúe entre ejecuciones y no siempre será la misma. Sin embargo, podemos observar la tendencia de que generalmente entre más robots haya, menor será el tiempo de ejecución. Aún así, aspectos como el tamaño de la cuadrícula o el porcentaje de suciedad pueden afectar el comportamiento de la simulación.

Conclusión

A través de esta actividad pude poner en práctica el conocimiento que he adquirido respecto a sistemas multiagentes y me permitió experimentar con lo que es una simulación basada en este tipo de programación. Lo que más me parece interesante es que podemos dotar de autonomía a estos agentes y observar un comportamiento que posiblemente pueda retratar lo que pasaría en la vida real a cierto nivel. En este caso pusimos a prueba un modelo que retrata el comportamiento que pueden tener los robots barredora cuando están colocados en el mismo espacio y determinar cuál es la cantidad óptima para limpiar en menor tiempo posible.

Repositorio en github: <https://github.com/IvanParedesA/Robot-de-Limpieza.git>