



Tecnológico
de Monterrey

Campus Santa Fe

MA. Actividad: Roomba

Modelación de sistemas multiagentes con gráficas computacionales (Gpo 301)

Miranda Urban Solano A01752391

Sábado 22 de noviembre de 2025

Índice

<i>Introducción</i>	3
<i>Problemática</i>	3
<i>Desarrollo</i>	3
<i>Análisis del ambiente y agentes</i>	3
Descripción ambiente	3
Descripción de los agentes	3
Características generales.....	3
PEAS de los agentes	4
Arquitectura de subsunción de los agentes.....	4
Propuesta de solución	5
<i>Análisis de resultados</i>	5
Simulación 1	5
Simulación 2	6
<i>Conclusiones</i>	7
<i>Anexo</i>	9
<i>Evidencias: Simulación 1</i>	9
Capturas de pantalla.....	9
<i>Evidencias: Simulación 2</i>	13
Capturas de pantalla.....	13
Estadísticas recolectadas por agente	17

Introducción

Problemática

El presente trabajo tiene como objetivo desarrollar y evaluar un sistema de agentes autónomos capaces de realizar tareas de limpieza dentro de un entorno simulado. Específicamente, se busca que los agentes tomen decisiones que les permita maximizar el nivel de limpieza alcanzable dentro de un tiempo de ejecución limitado, mientras gestionan eficientemente su energía. Se analiza el impacto del número de agentes en el desempeño global, considerando métricas como tiempo requerido, cantidad de movimientos realizados y eficiencia energética.

Desarrollo

Análisis del ambiente y agentes

Descripción ambiente

El ambiente corresponde a una cuadrícula de tamaño $M \times N$, la cual se inicializa con:

- 15% de celdas sucias distribuidas aleatoriamente.
- 2% de obstáculos que impiden el paso de los agentes.
- Estaciones de carga asignadas aleatoriamente, en donde aparecen por primera vez los agentes.

Descripción de los agentes

Los agentes tienen como objetivo principal limpiar el cuarto lo más que pueda en el tiempo máximo de ejecución, sin quedarse sin batería. Para ello se consideran las siguientes características:

- Inicia con 100% de batería.
- Cada acción consume 1% de energía.
- Permanecer sobre una estación de carga durante un episodio recupera 5% de batería.
- Debe ser capaz de planificar su regreso a la estación de carga antes de agotarse.

Características generales

Con toda la información anterior, es posible determinar que el ambiente presenta las siguientes características:

- Inaccesible: El agente únicamente puede acceder a la información de sus vecinos inmediatos.
- No determinista: Una acción no tiene un efecto garantizado en todas sus ejecuciones, pues se pueden ver afectados por los obstáculos.

- No episódico: El desempeño del agente no depende del episodio anterior.
- Estático: Permanece sin cambios excepto por la realización de acciones de los agentes.
- Discreto: El número de acciones y percepciones es fijo y finito.

Por otro lado, las características de los agentes son las siguientes:

- Reactivos: Tienen la capacidad de percibir su entorno y responden a cambios que hay dentro de este, por ejemplo, saben qué hacer si se encuentran con celdas bloqueadas, sucias, o con estaciones de carga.
- Proactivos: Planifican rutas de retorno usando BFS.
- Habilidades sociales: Al encontrarse con otro agente, comparten entre ellos información de las estaciones de carga y áreas exploradas.

PEAS de los agentes

- Performance: Los agentes son capaces de percibir la simulación, tomar acciones para sobrevivir (cargarse), y tomar decisiones para cumplir con su objetivo de limpieza (evitar obstáculos y limpiar celdas sucias).
- Ambiente: Su ambiente es una cuadrícula con zonas sucias, limpias y con estaciones de carga generadas de manera aleatoria.
- Actuadores: Para poder interactuar con su entorno, los agentes pueden moverse a cuadrículas vecinas sin obstáculos y preferentemente con suciedad para poder limpiar y cumplir con su objetivo, además conoce su zona de carga (o zonas de carga en general si ya tiene noción de otra) para poder cargarse en caso de tener poca batería.
- Sensores: Sus sensores son la “vista” que tiene de la cuadrícula que la rodea (celdas aledañas) y las características de estas.

Arquitectura de subsunción de los agentes

Los agentes deben de obtener información de su ambiente y evaluar para la toma de decisiones. Para ello se usa una jerarquía por capas para elegir acciones. A continuación, se muestran las posibles acciones de los agentes y su orden de menor a mayor prioridad:

5. Explorando
4. Comunicándose
3. Limpiando
2. Regresando
1. Cargando

Propuesta de solución

La solución propuesta consiste en un sistema multiagente que utiliza la arquitectura de subsunción para gestionar su comportamiento. Cada agente opera con una máquina de estados que prioriza: cargar batería cuando está por debajo del 20%, regresar a estaciones de carga usando BFS para calcular rutas óptimas, limpiar celdas sucias cuando están disponibles, comunicarse con otros agentes para compartir conocimiento sobre estaciones y áreas exploradas, y explorar estratégicamente el ambiente priorizando celdas no visitadas con suciedad. Además, los agentes utilizan percepción local para navegar evitando obstáculos y tomar decisiones autónomas, además incorpora mecanismos de colaboración donde los agentes intercambian información cuando coinciden en una celda, compartiendo ubicaciones de estaciones de carga y mapas de áreas exploradas.

Análisis de resultados

Para poder comprobar la funcionalidad de la solución propuesta se realizaron dos simulaciones con diferentes consideraciones.

Simulación 1

Se toman en cuenta las consideraciones generales, pero se inicializa un agente Roomba en la celda [1,1] junto a su respectiva estación de carga. El agente conoce únicamente su estación de carga inicial, debe limpiar el ambiente completo y gestionarse sin colaboración.

Se realizaron 10 iteraciones y se obtuvieron los siguientes resultados (véase el anexo para las evidencias).

Número de iteración	Grid	Steps	Celdas limpiadas	Movimientos totales	Batería restante
1	8x8	162	15/15	130	35%
2	8x8	73	15/15	57	28%
3	10x10	209	15/15	169	36%
4	10x10	73	15/15	57	28%
5	16x16	1000	37/38	809	14%
6	13x13	1000	23/25	819	44%
7	11x11	123	18/18	96	26%
8	6x7	26	6/6	19	75%
9	9x9	55	12/12	42	46%
10	12x12	270	21/21	216	23%

Con base en los resultados obtenidos, es posible observar lo siguiente:

- Eficiencia en diferentes escalas: El algoritmo es altamente eficiente en ambientes pequeños, observado en la relación de movimientos por celda limpia. Sin embargo, esta

eficiencia disminuye en espacios grandes, donde se observa que el tiempo de limpieza crece a mayor velocidad que el área del grid, por lo que demuestra que no es un sistema con escalabilidad.

- Buena gestión de batería: Los roombas cerca del límite mínimo de batería sin llegar a agotarse, es decir, los agentes recargan cuando es necesario. Por lo que los agentes no mueren durante las operaciones, haciendo la limpieza más rápida al no perder elementos.
- Confidabilidad en el retorno a estaciones: Los roombas pueden regresar exitosamente para recargar y reanudan sus operaciones una vez recuperadas la energía suficiente.
- Efectividad al limpiar: El sistema alcanza un 100% de efectividad en grids de tamaño pequeño y mediano, logrando limpieza completa en espacios usuales para este tipo de dispositivos.
- Limitaciones al no tener compañeros de limpieza: Al operar con un solo agente y no contar con mecanismos de comunicación, el agente no puede ejercer su mejor desempeño en espacios más grandes.

En conclusión, la estrategia de exploración implementada es óptima para espacios pequeños pero no es funcional con ambientes extensos, podría mejorar al optimizarla o al añadir otros agentes. El sistema no es escalable para grandes extensiones, pero ofrece una buena cobertura y completitud de limpieza en espacios pequeños. Esta característica lo hace perfecto para las áreas en donde típicamente operan máquinas de limpieza autónomas, como habitaciones u oficinas.

Simulación 2

Se toman en cuenta las consideraciones generales para inicializar el modelo. Los agentes empiezan en posiciones aleatorias con sus respectivas estaciones de carga. Importante considerar que los agentes conocen solamente la posición de su estación de carga inicial, pero pueden cargarse en cualquier estación de carga.

- Deberás de recopilar los datos solicitados por cada agente (tiempo de limpieza, cantidad de celdas limpiadas, número de movimientos realizados por el agente)

Se realizaron 10 iteraciones y se obtuvieron los siguientes resultados generales (véase el anexo, sección simulación 2, capturas de pantalla para más detalles).

Número de iteración	Grid	Número de roombas	Steps	Celdas limpiadas	Movimientos totales
1	30x30	14	211	135/135	2467
2	30x30	10	160	117/117	1313
3	30x30	21	209	117/117	3747
4	30x30	5	511	117/117	2071
5	30x30	24	93	117/117	1898

6	30x30	15	249	117/117	3122
7	30x30	17	111	117/117	1617
8	30x30	19	124	117/117	2068
9	30x30	26	72	117/117	1729
10	30x30	3	100	105/117	2436

Además, se evaluaron las métricas individuales de cada agente (véase el anexo, sección simulación 2, estadísticas recolectadas por agente para ver los datos reabados). Con base en esta información, se realizó el siguiente análisis:

- Relación entre número de agentes y eficiencia: Se identificó una relación directa entre la cantidad de agentes y el tiempo de limpieza. Mientras que 24 agentes completaron la limpieza en tan solo 93 steps, 3 agentes requirieron 1000 steps sin alcanzar la limpieza completa. Esto muestra que, a mayor número de agentes, menor es el tiempo necesario para finalizar la tarea. El punto óptimo se sitúa entre 10 y 15 agentes, ya que configuraciones superiores a 21 agentes no mejoran significativamente el tiempo, pero incrementan los movimientos totales del sistema.
- Distribución de carga de trabajo: La cantidad de agentes influye directamente en la carga de trabajo individual. En grupos reducidos, cada agente realiza una mayor cantidad de trabajo individual, mientras que en configuraciones con más agentes la carga se distribuye más equitativamente.
- Gestión energética eficiente: Los roombas demostraron una gestión óptima de energía, operando consistentemente cerca del límite mínimo de batería sin llegar a agotarse por completo, esto confirma que los agentes recargan oportunamente cuando es necesario. Cabe destacar que, en configuraciones con mayor número de agentes, se observó una batería residual más elevada, indicando una mejor gestión colectiva de energía.
- Ventajas de la colaboración multiagente: El sistema multiagente demostró una reducción significativa en el tiempo de limpieza comparado con configuraciones de pocos agentes.

La simulación confirma definitivamente que el enfoque social implementado en los agentes es significativamente más efectivo que configuraciones individuales o con pocas unidades, logrando mejoras sustanciales tanto en tiempo de limpieza como en completitud de la tarea.

Conclusiones

Al analizar ambas simulaciones se pudo observar claras ventajas en el enfoque multiagente sobre las configuraciones de agentes individuales. Mientras que el agente único mostró limitaciones en escalabilidad y crecimientos de tiempos de limpieza desproporcionados en grids de gran tamaño, el sistema multiagente logró optimizar el tiempo de limpieza mediante la distribución colaborativa de tareas. Específicamente, se encontró que, en la simulación programada, la

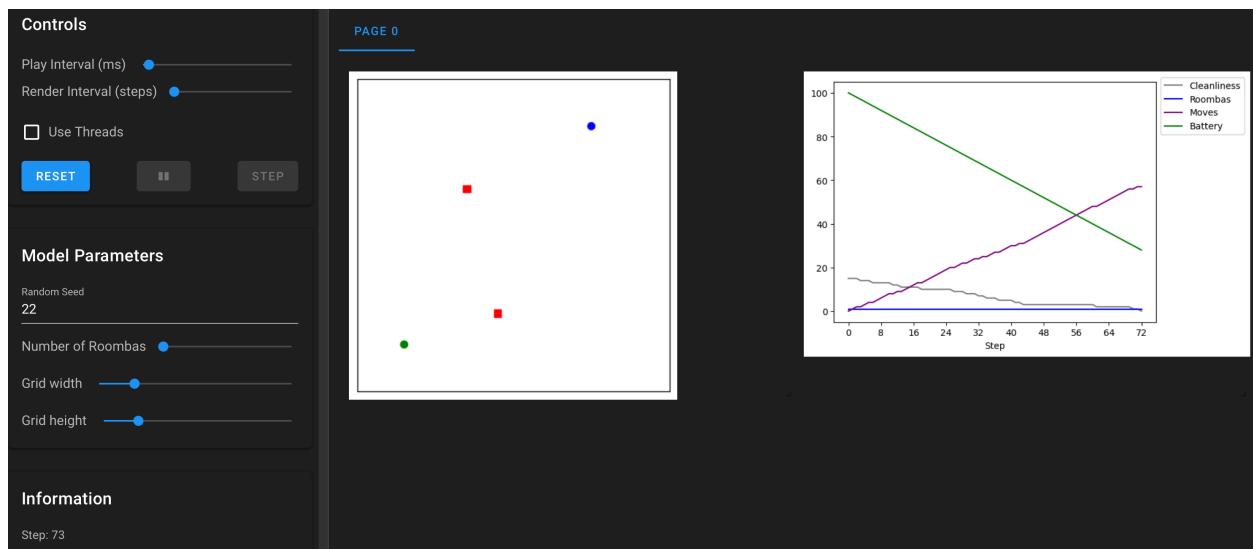
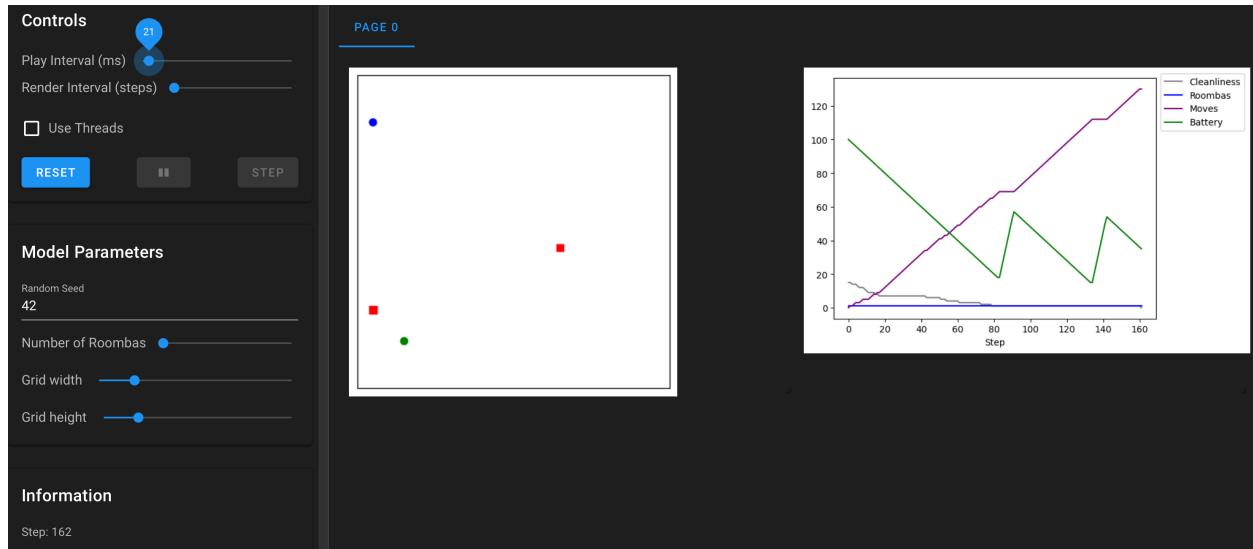
cantidad de agentes ideal es de entre 10 a 15 agentes, pues esta cantidad permite encontrar un equilibrio en la eficiencia operativa, reduciendo el tiempo requerido para limpieza total comparado con otras configuraciones.

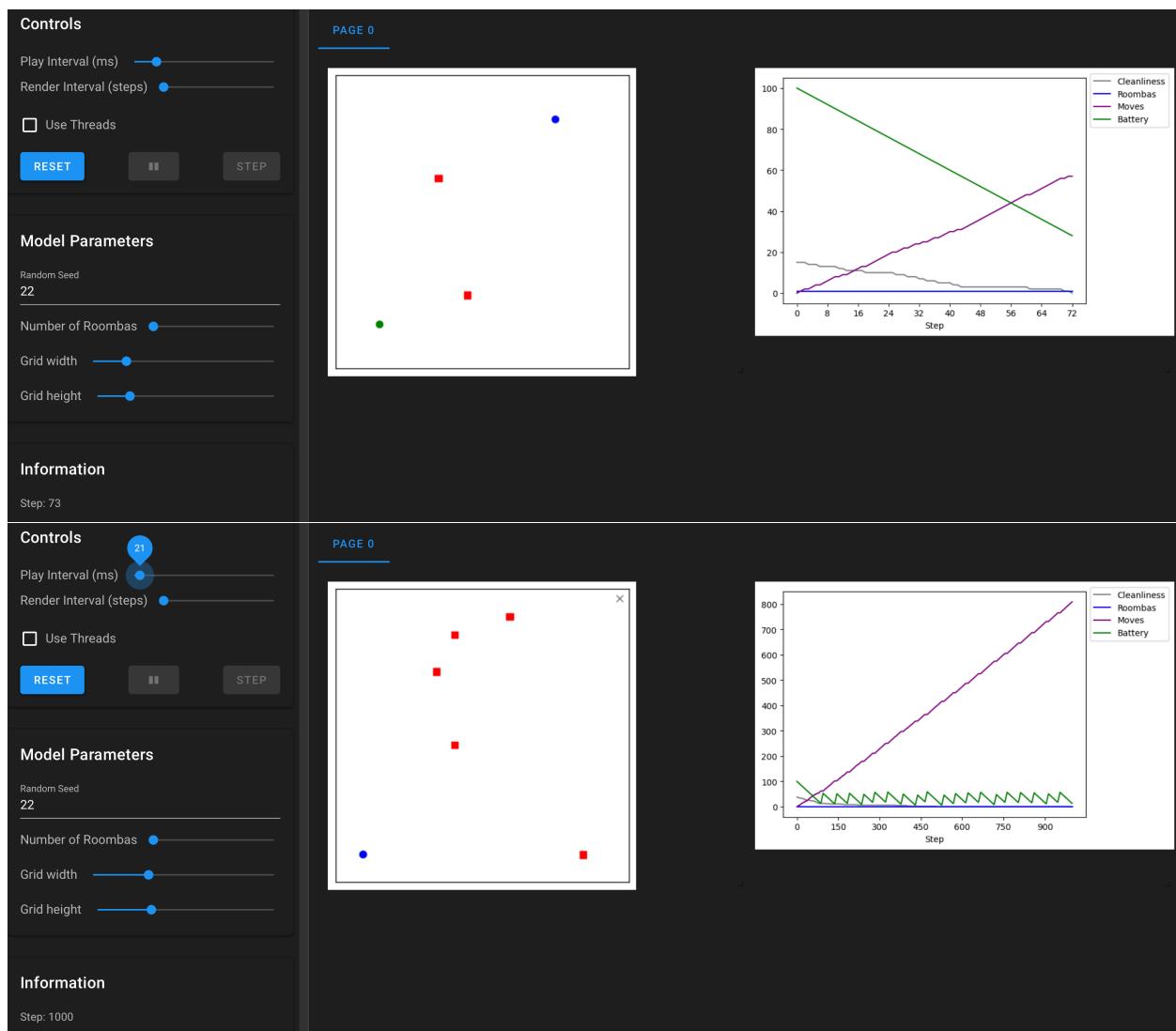
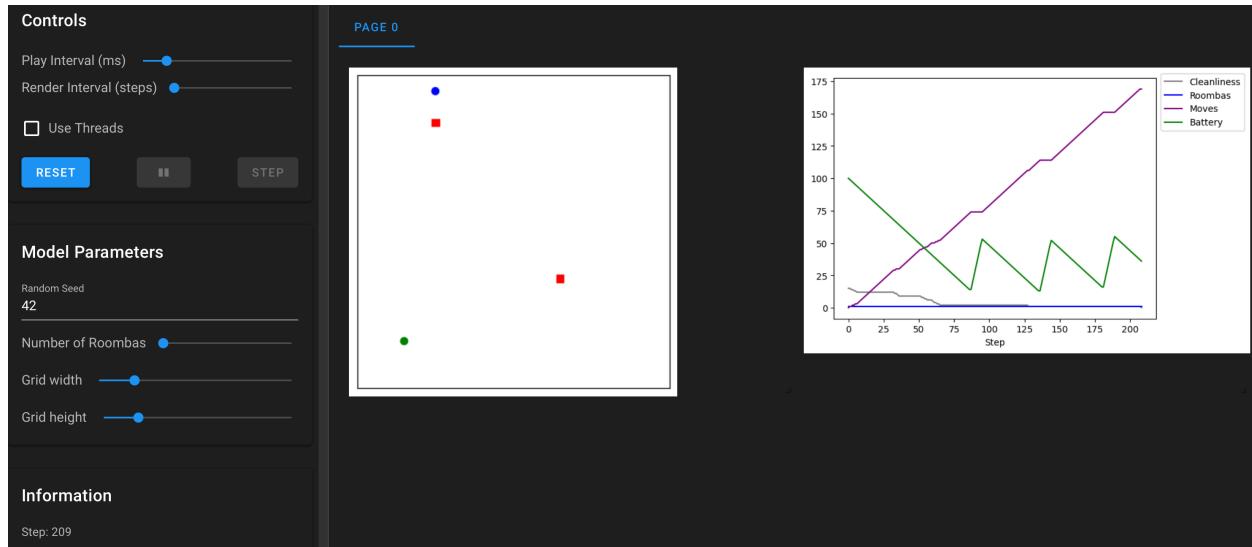
En definitiva, la implementación de mecanismos de comunicación y coordinación entre agentes resultó crucial para el éxito del sistema, permitiendo observar que la inteligencia colectiva supera significativamente las capacidades individuales en tareas de limpieza extensas. Esto se ve reflejado, no solamente en la completitud del objetivo, sino que también en la gestión energética y la distribución inteligente de la carga de trabajo.

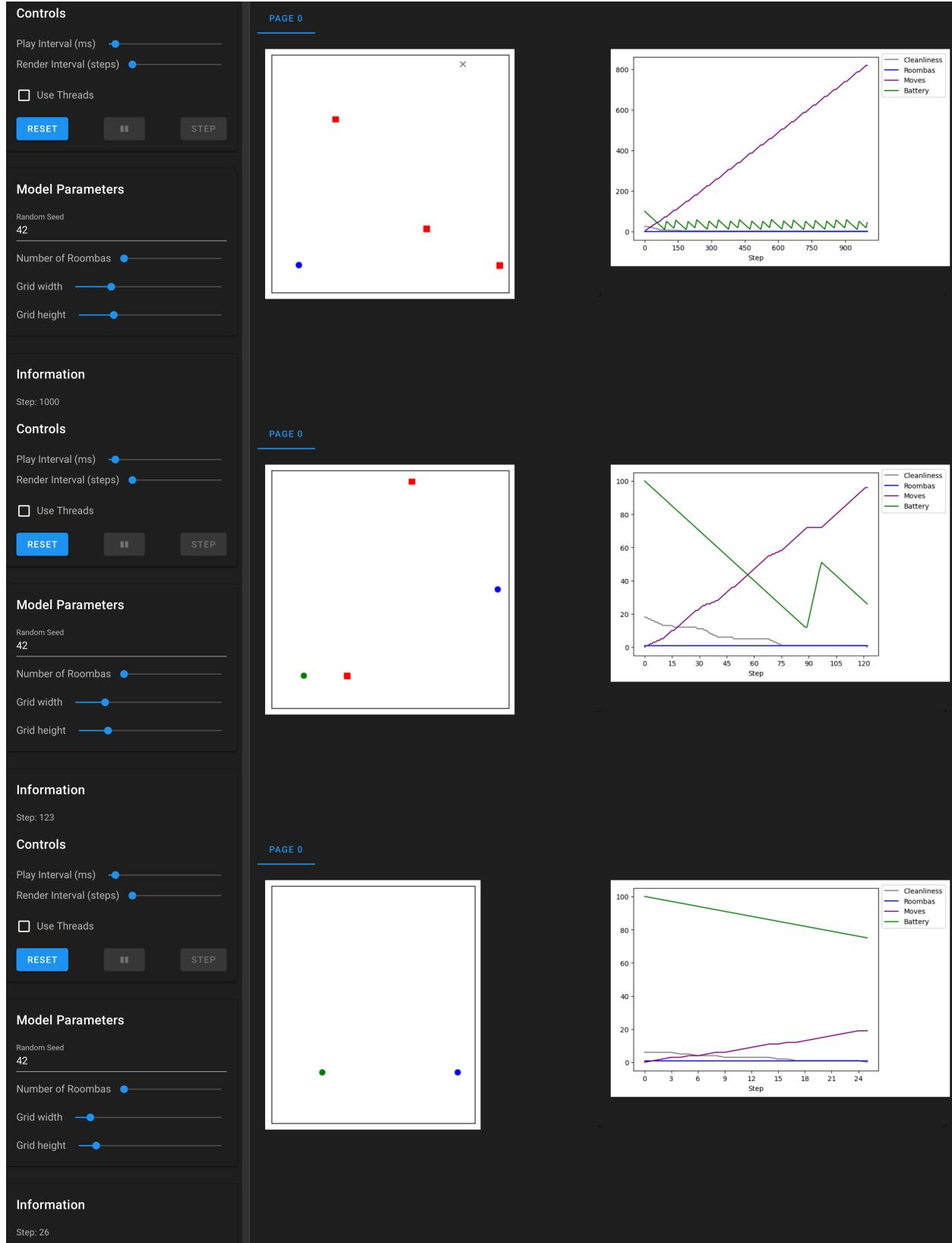
Anexo

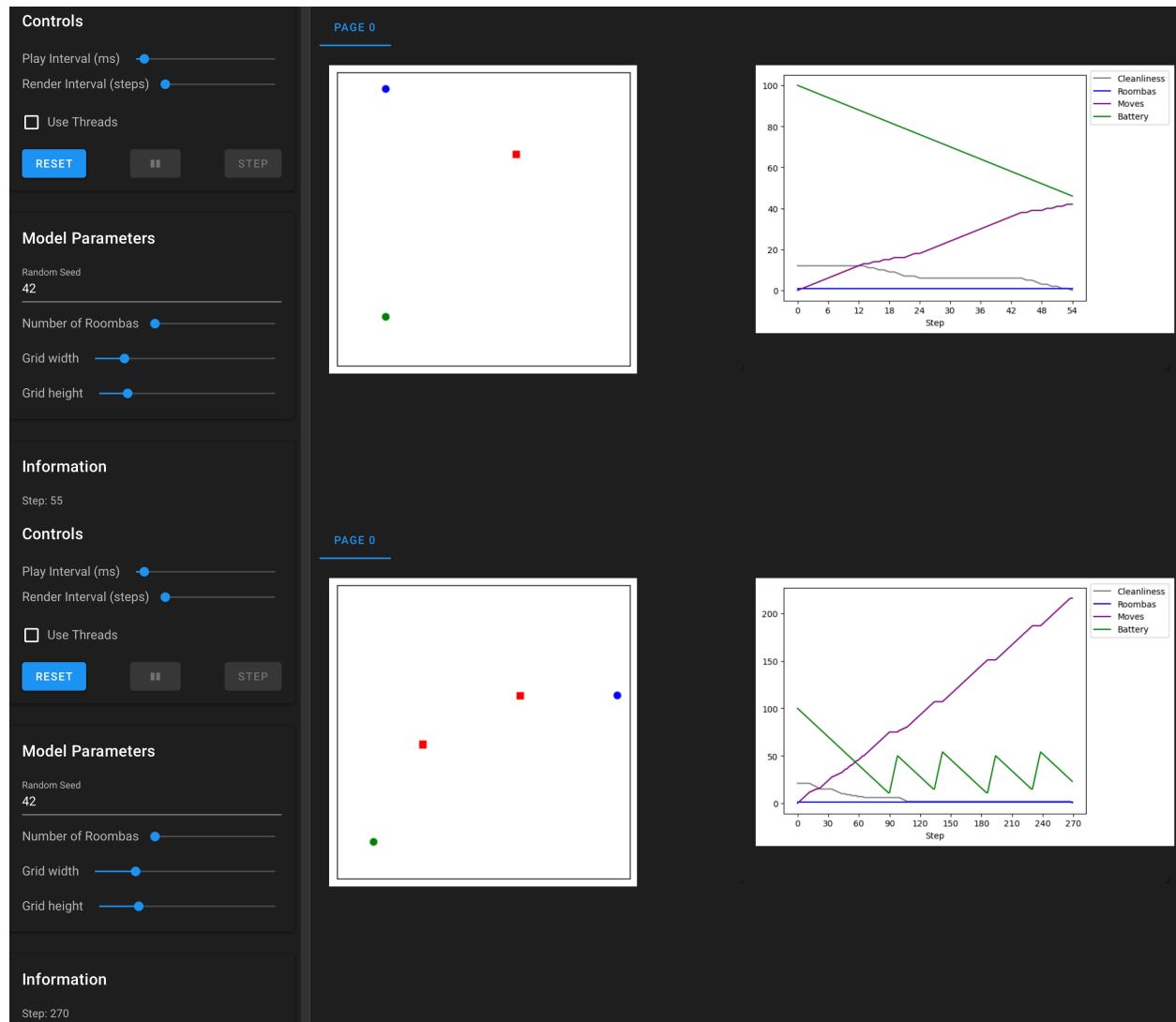
Evidencias: Simulación 1

Capturas de pantalla



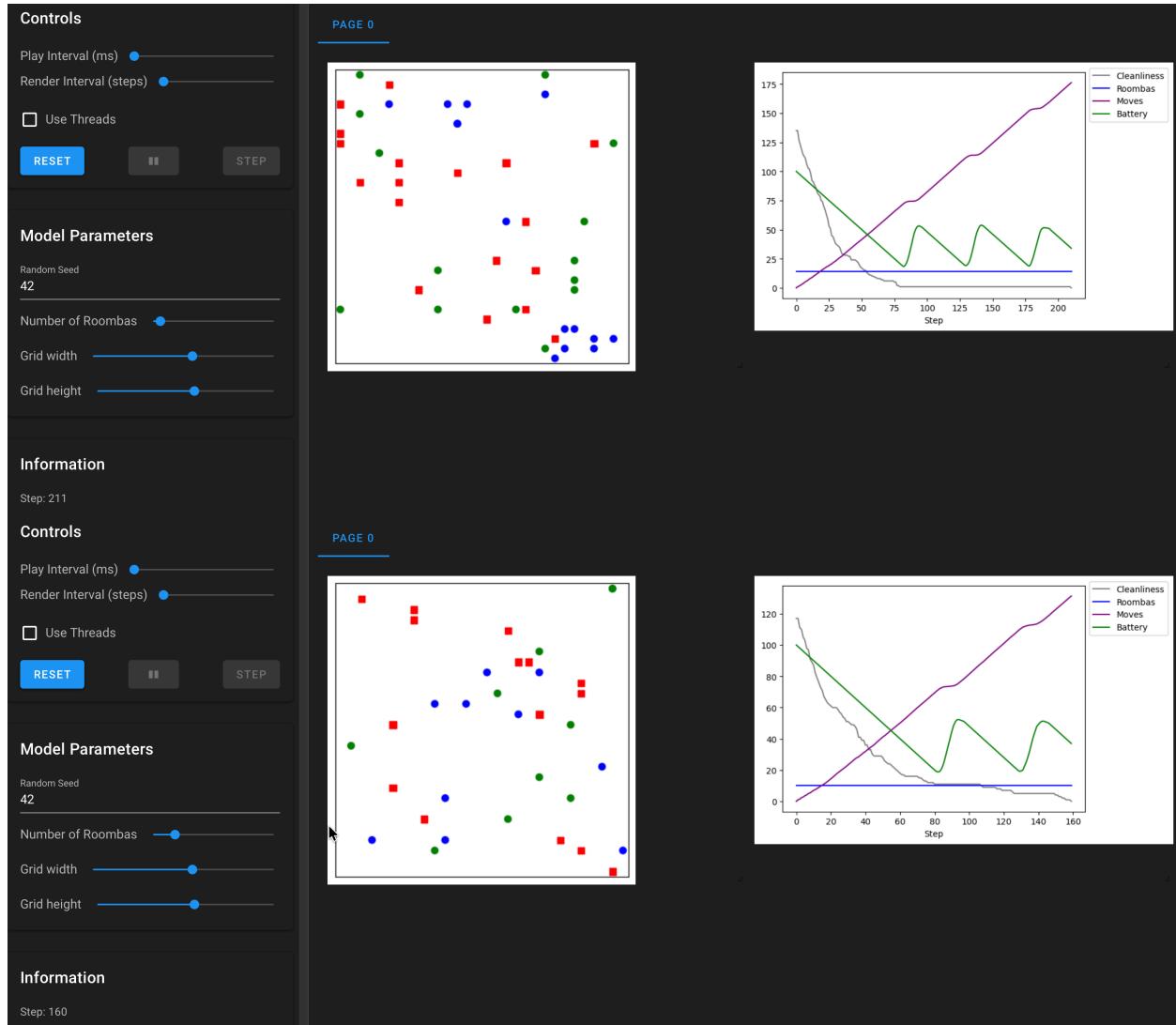


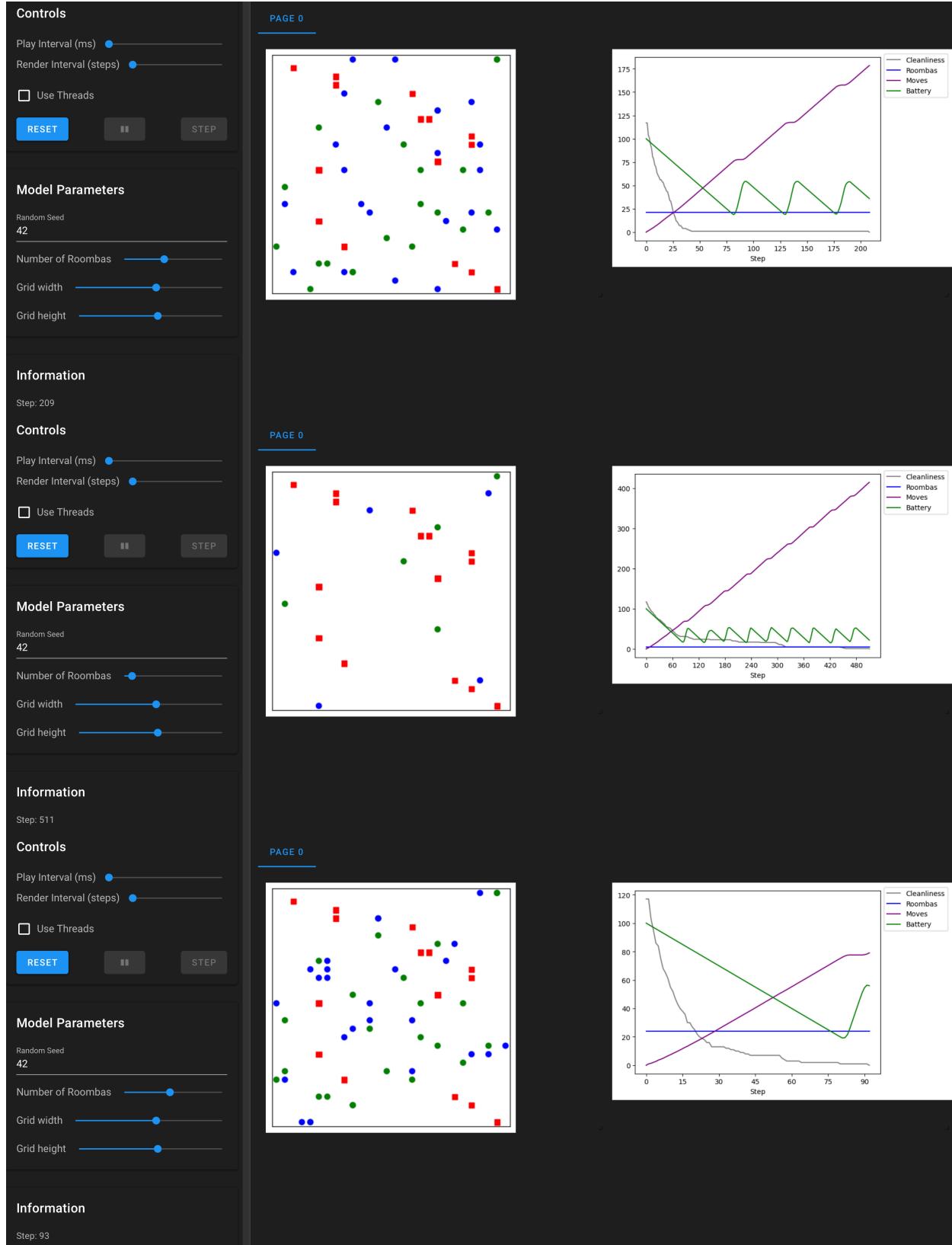


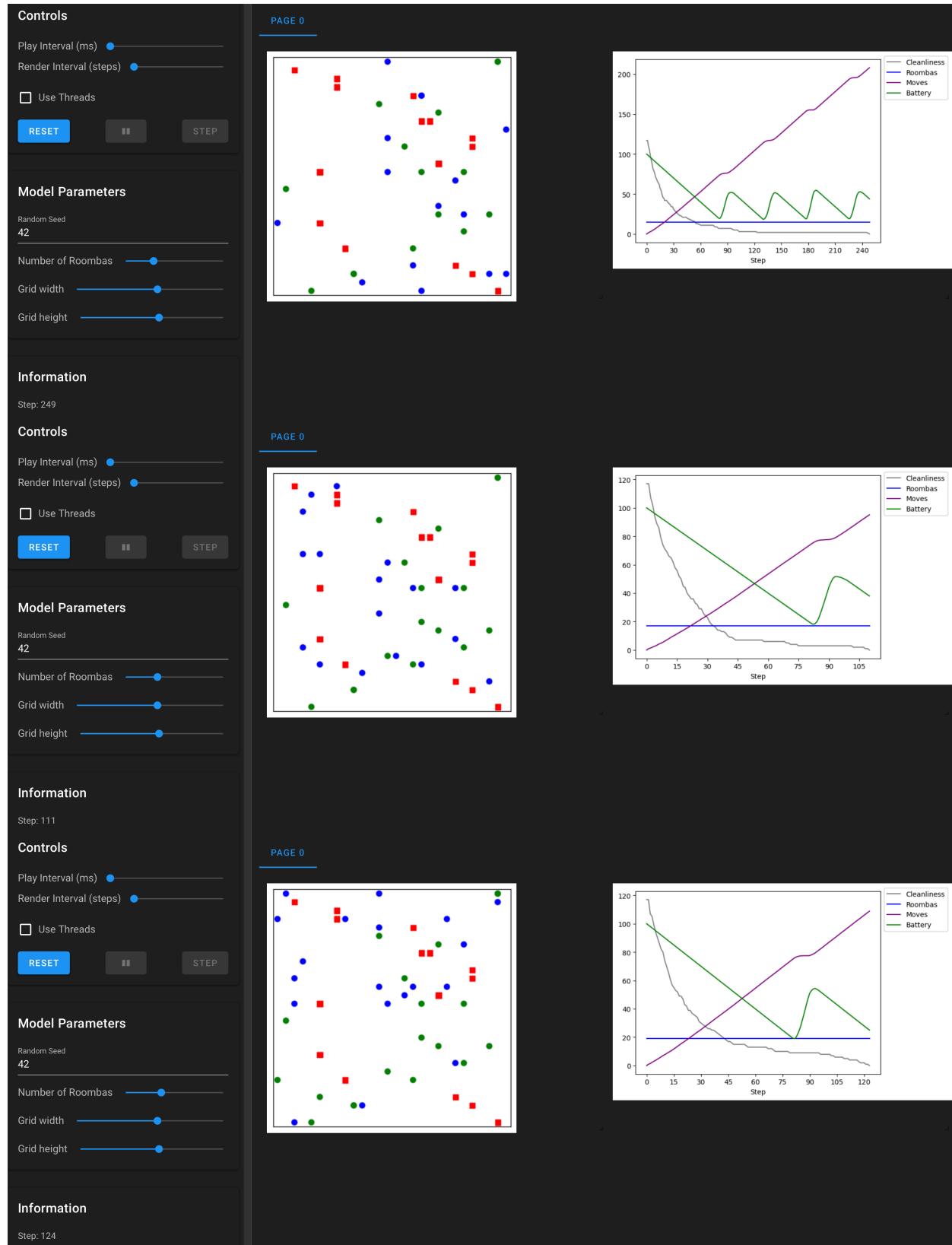


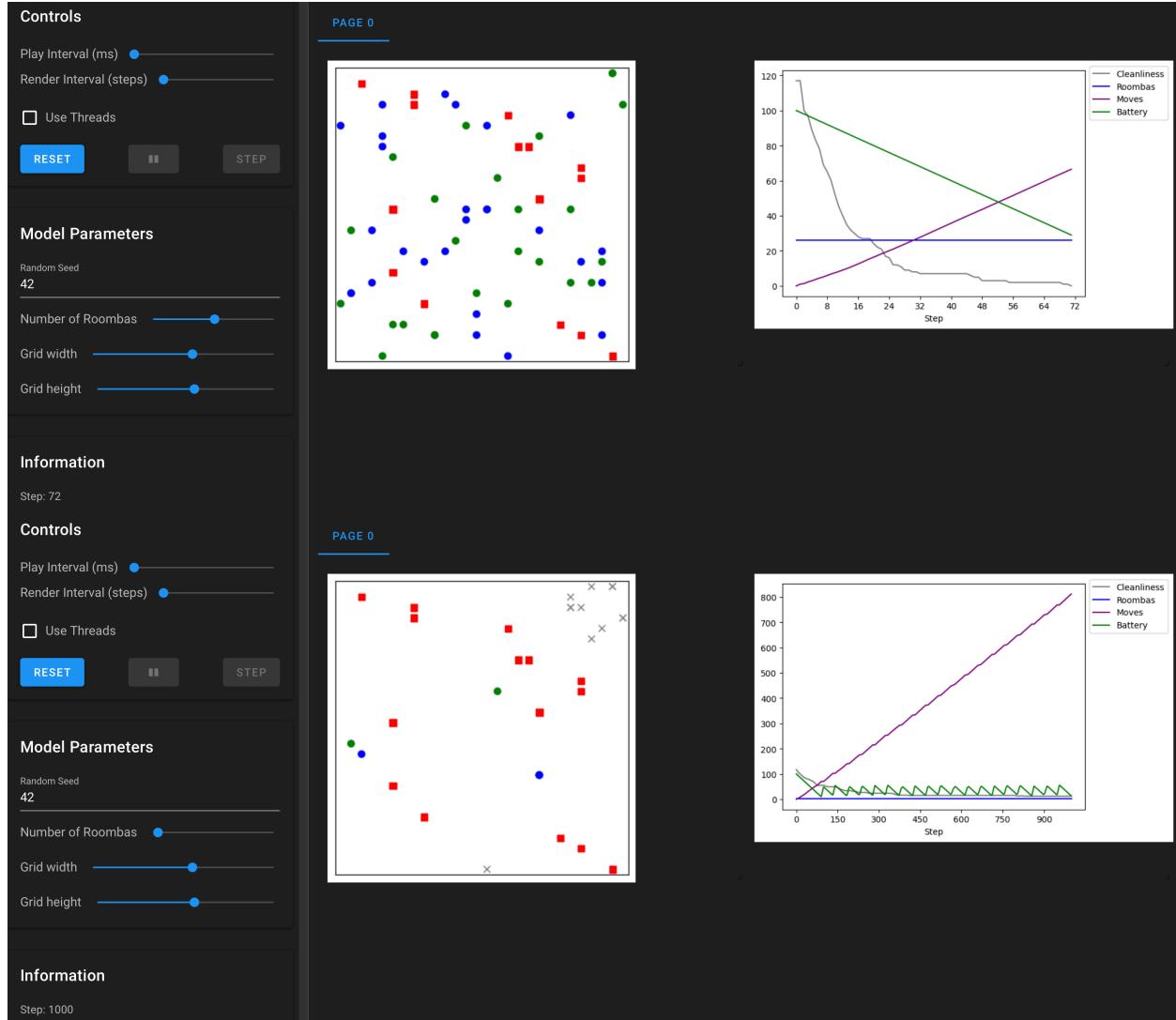
Evidencias: Simulación 2

Capturas de pantalla









Estadísticas recolectadas por agente

- Estadísticas por agente de la simulación 1:

Agente 33: 6 celda(s) limpiada(s), 180 movimientos, 34% de batería restante
 Agente 34: 9 celda(s) limpiada(s), 177 movimientos, 34% de batería restante
 Agente 35: 6 celda(s) limpiada(s), 180 movimientos, 34% de batería restante
 Agente 36: 7 celda(s) limpiada(s), 179 movimientos, 34% de batería restante
 Agente 37: 13 celda(s) limpiada(s), 173 movimientos, 34% de batería restante
 Agente 38: 10 celda(s) limpiada(s), 176 movimientos, 34% de batería restante
 Agente 39: 7 celda(s) limpiada(s), 179 movimientos, 34% de batería restante
 Agente 40: 10 celda(s) limpiada(s), 175 movimientos, 34% de batería restante
 Agente 41: 9 celda(s) limpiada(s), 177 movimientos, 34% de batería restante
 Agente 42: 10 celda(s) limpiada(s), 176 movimientos, 34% de batería restante
 Agente 43: 12 celda(s) limpiada(s), 174 movimientos, 34% de batería restante
 Agente 44: 9 celda(s) limpiada(s), 176 movimientos, 34% de batería restante
 Agente 45: 10 celda(s) limpiada(s), 176 movimientos, 34% de batería restante
 Agente 46: 17 celda(s) limpiada(s), 169 movimientos, 34% de batería restante

- Estadísticas por agente de la simulación 2:

Agente 26: 11 celda(s) limpiada(s), 132 movimientos, 37% de batería restante
 Agente 27: 17 celda(s) limpiada(s), 126 movimientos, 37% de batería restante
 Agente 28: 22 celda(s) limpiada(s), 121 movimientos, 37% de batería restante
 Agente 29: 7 celda(s) limpiada(s), 136 movimientos, 37% de batería restante
 Agente 30: 5 celda(s) limpiada(s), 138 movimientos, 37% de batería restante
 Agente 31: 7 celda(s) limpiada(s), 136 movimientos, 37% de batería restante
 Agente 32: 11 celda(s) limpiada(s), 132 movimientos, 37% de batería restante
 Agente 33: 11 celda(s) limpiada(s), 132 movimientos, 37% de batería restante
 Agente 34: 10 celda(s) limpiada(s), 133 movimientos, 37% de batería restante
 Agente 35: 16 celda(s) limpiada(s), 127 movimientos, 37% de batería restante

- Estadísticas por agente de la simulación 3:

Agente 37: 10 celda(s) limpiada(s), 174 movimientos, 36% de batería restante
 Agente 38: 5 celda(s) limpiada(s), 179 movimientos, 36% de batería restante
 Agente 39: 8 celda(s) limpiada(s), 176 movimientos, 36% de batería restante
 Agente 40: 9 celda(s) limpiada(s), 175 movimientos, 36% de batería restante
 Agente 41: 9 celda(s) limpiada(s), 175 movimientos, 36% de batería restante
 Agente 42: 5 celda(s) limpiada(s), 179 movimientos, 36% de batería restante
 Agente 43: 4 celda(s) limpiada(s), 180 movimientos, 36% de batería restante
 Agente 44: 4 celda(s) limpiada(s), 180 movimientos, 36% de batería restante
 Agente 45: 0 celda(s) limpiada(s), 184 movimientos, 36% de batería restante
 Agente 46: 5 celda(s) limpiada(s), 179 movimientos, 36% de batería restante
 Agente 47: 3 celda(s) limpiada(s), 181 movimientos, 36% de batería restante
 Agente 48: 3 celda(s) limpiada(s), 181 movimientos, 36% de batería restante
 Agente 49: 3 celda(s) limpiada(s), 181 movimientos, 36% de batería restante
 Agente 50: 5 celda(s) limpiada(s), 179 movimientos, 36% de batería restante
 Agente 51: 5 celda(s) limpiada(s), 179 movimientos, 36% de batería restante
 Agente 52: 10 celda(s) limpiada(s), 174 movimientos, 36% de batería restante
 Agente 53: 7 celda(s) limpiada(s), 177 movimientos, 36% de batería restante

Agente 54: 2 celda(s) limpiada(s), 182 movimientos, 36% de batería restante
 Agente 55: 6 celda(s) limpiada(s), 178 movimientos, 36% de batería restante
 Agente 56: 2 celda(s) limpiada(s), 182 movimientos, 36% de batería restante
 Agente 57: 12 celda(s) limpiada(s), 172 movimientos, 36% de batería restante

- Estadísticas por agente simulación 4:

Agente 21: 31 celda(s) limpiada(s), 407 movimientos, 22% de batería restante
 Agente 22: 26 celda(s) limpiada(s), 412 movimientos, 22% de batería restante
 Agente 23: 18 celda(s) limpiada(s), 420 movimientos, 22% de batería restante
 Agente 24: 24 celda(s) limpiada(s), 414 movimientos, 22% de batería restante
 Agente 25: 18 celda(s) limpiada(s), 420 movimientos, 22% de batería restante

- Estadísticas por agente simulación 5:

Agente 40: 11 celda(s) limpiada(s), 73 movimientos, 56% de batería restante
 Agente 41: 6 celda(s) limpiada(s), 78 movimientos, 56% de batería restante
 Agente 42: 6 celda(s) limpiada(s), 78 movimientos, 56% de batería restante
 Agente 43: 7 celda(s) limpiada(s), 77 movimientos, 56% de batería restante
 Agente 44: 3 celda(s) limpiada(s), 81 movimientos, 56% de batería restante
 Agente 45: 5 celda(s) limpiada(s), 79 movimientos, 56% de batería restante
 Agente 46: 6 celda(s) limpiada(s), 78 movimientos, 56% de batería restante
 Agente 47: 6 celda(s) limpiada(s), 78 movimientos, 56% de batería restante
 Agente 48: 2 celda(s) limpiada(s), 82 movimientos, 56% de batería restante
 Agente 49: 4 celda(s) limpiada(s), 80 movimientos, 56% de batería restante
 Agente 50: 8 celda(s) limpiada(s), 76 movimientos, 56% de batería restante
 Agente 51: 1 celda(s) limpiada(s), 83 movimientos, 56% de batería restante
 Agente 52: 9 celda(s) limpiada(s), 75 movimientos, 56% de batería restante
 Agente 53: 1 celda(s) limpiada(s), 83 movimientos, 56% de batería restante
 Agente 54: 7 celda(s) limpiada(s), 77 movimientos, 56% de batería restante
 Agente 55: 5 celda(s) limpiada(s), 79 movimientos, 56% de batería restante
 Agente 56: 5 celda(s) limpiada(s), 79 movimientos, 56% de batería restante
 Agente 57: 2 celda(s) limpiada(s), 82 movimientos, 56% de batería restante
 Agente 58: 1 celda(s) limpiada(s), 83 movimientos, 56% de batería restante
 Agente 59: 4 celda(s) limpiada(s), 79 movimientos, 56% de batería restante
 Agente 60: 9 celda(s) limpiada(s), 75 movimientos, 56% de batería restante
 Agente 61: 3 celda(s) limpiada(s), 81 movimientos, 56% de batería restante
 Agente 62: 5 celda(s) limpiada(s), 79 movimientos, 56% de batería restante
 Agente 63: 1 celda(s) limpiada(s), 83 movimientos, 56% de batería restante

- Estadísticas por agente simulación 6:

Agente 31: 4 celda(s) limpiada(s), 212 movimientos, 44% de batería restante
 Agente 32: 10 celda(s) limpiada(s), 206 movimientos, 44% de batería restante
 Agente 33: 10 celda(s) limpiada(s), 206 movimientos, 44% de batería restante
 Agente 34: 20 celda(s) limpiada(s), 196 movimientos, 44% de batería restante
 Agente 35: 7 celda(s) limpiada(s), 209 movimientos, 44% de batería restante
 Agente 36: 3 celda(s) limpiada(s), 213 movimientos, 44% de batería restante
 Agente 37: 15 celda(s) limpiada(s), 201 movimientos, 44% de batería restante
 Agente 38: 11 celda(s) limpiada(s), 205 movimientos, 44% de batería restante
 Agente 39: 4 celda(s) limpiada(s), 212 movimientos, 44% de batería restante

Agente 40: 9 celda(s) limpiada(s), 207 movimientos, 44% de batería restante
 Agente 41: 4 celda(s) limpiada(s), 211 movimientos, 44% de batería restante
 Agente 42: 2 celda(s) limpiada(s), 214 movimientos, 44% de batería restante
 Agente 43: 11 celda(s) limpiada(s), 205 movimientos, 44% de batería restante
 Agente 44: 5 celda(s) limpiada(s), 211 movimientos, 44% de batería restante
 Agente 45: 2 celda(s) limpiada(s), 214 movimientos, 44% de batería restante

- Estadísticas por agente simulación 7:

Agente 33: 7 celda(s) limpiada(s), 95 movimientos, 38% de batería restante
 Agente 34: 4 celda(s) limpiada(s), 98 movimientos, 38% de batería restante
 Agente 35: 11 celda(s) limpiada(s), 91 movimientos, 38% de batería restante
 Agente 36: 5 celda(s) limpiada(s), 97 movimientos, 38% de batería restante
 Agente 37: 6 celda(s) limpiada(s), 96 movimientos, 38% de batería restante
 Agente 38: 5 celda(s) limpiada(s), 97 movimientos, 38% de batería restante
 Agente 39: 9 celda(s) limpiada(s), 93 movimientos, 38% de batería restante
 Agente 40: 10 celda(s) limpiada(s), 92 movimientos, 38% de batería restante
 Agente 41: 10 celda(s) limpiada(s), 92 movimientos, 38% de batería restante
 Agente 42: 4 celda(s) limpiada(s), 98 movimientos, 38% de batería restante
 Agente 43: 9 celda(s) limpiada(s), 93 movimientos, 38% de batería restante
 Agente 44: 5 celda(s) limpiada(s), 97 movimientos, 38% de batería restante
 Agente 45: 8 celda(s) limpiada(s), 94 movimientos, 38% de batería restante
 Agente 46: 7 celda(s) limpiada(s), 95 movimientos, 38% de batería restante
 Agente 47: 4 celda(s) limpiada(s), 98 movimientos, 38% de batería restante
 Agente 48: 4 celda(s) limpiada(s), 98 movimientos, 38% de batería restante
 Agente 49: 9 celda(s) limpiada(s), 93 movimientos, 38% de batería restante

- Estadísticas por agente simulación 8:

Agente 35: 1 celda(s) limpiada(s), 114 movimientos, 25% de batería restante
 Agente 36: 6 celda(s) limpiada(s), 109 movimientos, 25% de batería restante
 Agente 37: 14 celda(s) limpiada(s), 101 movimientos, 25% de batería restante
 Agente 38: 7 celda(s) limpiada(s), 108 movimientos, 25% de batería restante
 Agente 39: 8 celda(s) limpiada(s), 107 movimientos, 25% de batería restante
 Agente 40: 3 celda(s) limpiada(s), 112 movimientos, 25% de batería restante
 Agente 41: 5 celda(s) limpiada(s), 110 movimientos, 25% de batería restante
 Agente 42: 10 celda(s) limpiada(s), 105 movimientos, 25% de batería restante
 Agente 43: 11 celda(s) limpiada(s), 104 movimientos, 25% de batería restante
 Agente 44: 9 celda(s) limpiada(s), 106 movimientos, 25% de batería restante
 Agente 45: 2 celda(s) limpiada(s), 113 movimientos, 25% de batería restante
 Agente 46: 3 celda(s) limpiada(s), 112 movimientos, 25% de batería restante
 Agente 47: 7 celda(s) limpiada(s), 108 movimientos, 25% de batería restante
 Agente 48: 1 celda(s) limpiada(s), 114 movimientos, 25% de batería restante
 Agente 49: 6 celda(s) limpiada(s), 109 movimientos, 25% de batería restante
 Agente 50: 4 celda(s) limpiada(s), 111 movimientos, 25% de batería restante
 Agente 51: 5 celda(s) limpiada(s), 110 movimientos, 25% de batería restante
 Agente 52: 2 celda(s) limpiada(s), 113 movimientos, 25% de batería restante
 Agente 53: 13 celda(s) limpiada(s), 102 movimientos, 25% de batería restante

- Estadísticas por agente simulación 9:

Agente 42: 10 celda(s) limpiada(s), 61 movimientos, 29% de batería restante

Agente 43: 2 celda(s) limpiada(s), 69 movimientos, 29% de batería restante
Agente 44: 4 celda(s) limpiada(s), 67 movimientos, 29% de batería restante
Agente 45: 3 celda(s) limpiada(s), 68 movimientos, 29% de batería restante
Agente 46: 5 celda(s) limpiada(s), 66 movimientos, 29% de batería restante
Agente 47: 2 celda(s) limpiada(s), 69 movimientos, 29% de batería restante
Agente 48: 1 celda(s) limpiada(s), 70 movimientos, 29% de batería restante
Agente 49: 4 celda(s) limpiada(s), 67 movimientos, 29% de batería restante
Agente 50: 8 celda(s) limpiada(s), 63 movimientos, 29% de batería restante
Agente 51: 4 celda(s) limpiada(s), 67 movimientos, 29% de batería restante
Agente 52: 6 celda(s) limpiada(s), 65 movimientos, 29% de batería restante
Agente 53: 1 celda(s) limpiada(s), 70 movimientos, 29% de batería restante
Agente 54: 5 celda(s) limpiada(s), 66 movimientos, 29% de batería restante
Agente 55: 1 celda(s) limpiada(s), 70 movimientos, 29% de batería restante
Agente 56: 3 celda(s) limpiada(s), 68 movimientos, 29% de batería restante
Agente 57: 4 celda(s) limpiada(s), 67 movimientos, 29% de batería restante
Agente 58: 6 celda(s) limpiada(s), 65 movimientos, 29% de batería restante
Agente 59: 3 celda(s) limpiada(s), 68 movimientos, 29% de batería restante
Agente 60: 1 celda(s) limpiada(s), 70 movimientos, 29% de batería restante
Agente 61: 3 celda(s) limpiada(s), 68 movimientos, 29% de batería restante
Agente 62: 10 celda(s) limpiada(s), 61 movimientos, 29% de batería restante
Agente 63: 7 celda(s) limpiada(s), 64 movimientos, 29% de batería restante
Agente 64: 7 celda(s) limpiada(s), 64 movimientos, 29% de batería restante
Agente 65: 1 celda(s) limpiada(s), 70 movimientos, 29% de batería restante
Agente 66: 7 celda(s) limpiada(s), 64 movimientos, 29% de batería restante
Agente 67: 9 celda(s) limpiada(s), 62 movimientos, 29% de batería restante

- Estadísticas por agente simulación 10:

Agente 19: 36 celda(s) limpiada(s), 811 movimientos, 13% de batería restante
Agente 20: 39 celda(s) limpiada(s), 808 movimientos, 13% de batería restante
Agente 21: 30 celda(s) limpiada(s), 817 movimientos, 13% de batería restante