



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



Master Universitario en Inteligencia Artificial, Reconocimiento de Formas e
Imagen Digital
Universidad Politécnica de Valencia

**Simulación de Evacuación mediante Sistemas
Multi-Agentes con MESA**
TRABAJO SISTEMAS MULTI-AGENTES

Máster Universitario en Inteligencia Artificial, Reconocimiento de Formas e
Imagen Digital

Autor: Miquel Gómez

Resumen

Este trabajo presenta el desarrollo e implementación de un sistema de simulación multi-agente para el estudio de dinámicas de evacuación en situaciones de emergencia.

Utilizando el framework MESA en Python, se ha creado un modelo discreto basado en malla que permite analizar el impacto de comportamientos individuales en procesos de evacuación de grupos de personas.

Se han implementado tres tipologías de agentes (educados, agresivos y lentos) que interactúan en diferentes escenarios. Estos escenarios representan espacios y situaciones reales. Los resultados experimentales revelan conclusiones contraintuitivas respecto a la literatura clásica, demostrando que en modelos discretos las simulaciones no siempre se comportan como se espera dadas sus limitaciones.

A su vez, se valida la utilidad del framework MESA para este tipo de simulaciones y el poder parametrizar escenarios o situaciones para su posterior evaluación. Lo que confirma que se puede, hasta cierto punto de realismo, estudiar el comportamiento emergente de las multitudes en situaciones de evacuación.

Se presenta el sistema con todos sus componentes, la metodología de evaluación y los resultados obtenidos, destacando las implicaciones prácticas para el diseño de espacios seguros y eficientes.

Palabras clave: Sistemas Multi-Agentes; Agentes; Simulación; Evacuación; Simulación de evacuación; Multitudes.

Índice general

Índice general	IV
1 Introducción	1
1.1 Alcance y Objetivos	1
1.1.1 Enfoque en Evacuación de Emergencia	1
1.2 Uso de MESA	2
1.3 Experimentación	2
1.4 Estructura del Documento	3
2 Estado del Arte	5
2.1 Objetivos de la Simulación de Evacuación	5
2.1.1 Diseño Basado en Prestaciones (PBD) y Seguridad	5
2.1.2 Nota sobre Nivel de Servicio (LOS)	6
2.2 Marcos Regulatorios y Validación	6
2.3 Arquitecturas de Navegación y Micro-comportamientos	6
2.3.1 Modelos de Espacio Continuo	6
2.3.2 Modelos Discretos y Grid-Based	6
3 Análisis y definición del problema	7
3.1 Modelo: Discrete Grid-Based	7
3.2 Agentes: Comportamiento en Evacuación	8
4 Implementación	9
4.1 Entorno y Navegación (Floor Fields)	9
4.1.1 Heurística A*	9
4.2 Comportamiento de los Agentes	10
4.2.1 1. Intención de Movimiento (Velocidad y Densidad)	10
4.2.2 2. Resolución de Prioridad	10
4.2.3 3. Gestión de Deadlocks (Pánico)	10
4.3 Visualización y Herramientas	11
4.3.1 Representación de los entornos	12
5 Evaluación	13
5.1 Definición de Variables e Hipótesis	13
5.1.1 Variables del Estudio	13
5.1.2 Hipótesis de Trabajo	13
5.2 Métricas de Evaluación	14
5.3 Diseño de Experimentos	14
5.3.1 Escenarios	14
5.4 Configuración y Protocolo Experimental	16
5.4.1 Valores por Defecto	16
5.4.2 Definición de Experimentos	17
6 Resultados Experimentales	19
6.1 Análisis de Resultados Agregados	19
6.1.1 Densidad de Agentes	19
6.1.2 Comportamiento Agresivo	20
6.1.3 Agentes Lentos	21

6.1.4	Algoritmos de Pathfinding	22
6.1.5	Preferencias de Salida	22
6.1.6	Ancho de Salida	22
6.1.7	Número de Salidas	23
6.2	Análisis de Evolución Temporal	23
6.2.1	Dinámica de las Preferencias de Salida	23
6.2.2	Dinámica del Ancho de Salida	24
6.3	Discusión y Limitaciones	24
7	Conclusiones y Trabajo Futuro	25
7.1	Conclusiones	25
7.1.1	La Geometría como Factor Determinante	25
7.1.2	Paradoja del Conocimiento y Costo de la Rigidez	25
7.1.3	Impacto de la Heterogeneidad de la Multitud	26
7.1.4	Importancia de la Re-planificación Dinámica	26
7.1.5	Dinámica Temporal	26
7.2	Trabajo Futuro	26

CAPÍTULO 1

Introducción

La gestión de evacuaciones y la seguridad en situaciones de emergencia son preocupaciones crecientes en la planificación urbana y arquitectónica. Entender cómo las personas evacuan espacios durante emergencias, cómo reaccionan bajo presión y cómo interactúan entre sí en procesos de evacuación, es fundamental para diseñar espacios seguros.

La simulación informática surge como una alternativa ética, económica y segura a los experimentos con personas reales, permitiendo probar escenarios hipotéticos de riesgo sin poner en peligro vidas humanas.

Este trabajo surge de la asignatura de Sistemas Multi-Agente, con el objetivo de experimentar con el framework MESA en Python para crear un entorno de experimentación sobre evacuación de multitudes. Se centra en el desarrollo y análisis del sistema, así como en la evaluación de su comportamiento bajo diferentes configuraciones y escenarios. Todo esto, para observar la viabilidad de MESA como herramienta para este tipo de simulaciones.

1.1 Alcance y Objetivos

El objetivo principal de este trabajo es desarrollar un entorno de simulación multi-agente (MAS) flexible utilizando la librería MESA en Python, que permita estudiar el impacto de los comportamientos individuales heterogéneos en la dinámica de evacuación de grupos de personas.

- **Abstracción frente a Precisión Física:** Este proyecto **no pretende** crear una herramienta de ingeniería civil certificable bajo la norma ISO 20414 (estándar del que se habla más tarde) para validación legal de planos de evacuación, ni replicar con exactitud milimétrica unidades físicas reales.
- **Enfoque en Tendencias y Comportamiento de Evacuación:** El objetivo es simular tendencias emergentes y relaciones causales en procesos de evacuación (por ejemplo, "¿Cómo afecta un 10 % de agentes agresivos al tiempo total de evacuación?") en lugar de predecir tiempos exactos de evacuación.

1.1.1. Enfoque en Evacuación de Emergencia

El sistema se centra exclusivamente en el estudio de situaciones de evacuación de emergencia, donde la rapidez y eficiencia del proceso de evacuación son críticas para la supervivencia. Se pretende analizar cómo diferentes configuraciones de agentes y esce-

narios de evacuación afectan métricas clave como el tiempo total de evacuación, la tasa de flujo hacia las salidas y la congestión durante la evacuación.

Estos sistemas de evacuación son relevantes en contextos como:

- **Eventos Masivos:** Evacuación de conciertos, festivales y eventos deportivos en situaciones de emergencia.
- **Infraestructuras Públicas:** Planes de evacuación para aeropuertos, estaciones de tren y centros comerciales ante emergencias.
- **Diseño Urbano:** Planificación de rutas de evacuación en espacios públicos y minimización de riesgos durante evacuaciones.

Aunque el sistema también tiene la capacidad técnica de simular **flujos continuos de personas** en situaciones no urgentes, este aspecto no se ha explorado en profundidad en este trabajo. El enfoque principal es la evacuación de emergencia.

El enfoque principal y único de este trabajo es la evacuación de emergencia, validando si un espacio permite una evacuación segura y eficiente bajo diferentes condiciones y composiciones de agentes durante situaciones críticas. El sistema permite modelar de forma sencilla distintos escenarios de evacuación y composiciones de agentes en emergencias.

1.2 Uso de MESA

MESA es un framework de código abierto en Python diseñado para facilitar la creación, simulación y visualización de modelos basados en agentes. Proporciona una estructura modular que permite a los desarrolladores definir agentes, entornos y reglas de interacción de manera sencilla.

Además, una vez el sistema está implementado, MESA ofrece herramientas integradas para la recolección de datos y la visualización en tiempo real, lo que facilita el análisis de los resultados de las simulaciones.

En este trabajo, MESA se utiliza como la base para construir el modelo de simulación de multitudes, aprovechando sus capacidades para gestionar agentes heterogéneos y sus interacciones en un entorno discreto basado en malla.

Se ha implementado todo el sistema siguiendo la documentación oficial de MESA [1] y adaptando sus componentes a las necesidades específicas del estudio de evacuación.

1.3 Experimentación

El trabajo incluye un notebook de Jupyter que permite ejecutar simulaciones con diferentes configuraciones y escenarios, cambiarlos y visualizar los resultados y las estadísticas en tiempo real.

Este notebook facilita la experimentación al permitir modificar parámetros como la densidad de agentes, la proporción de tipos de agentes y las características del entorno. Además, incluye gráficos y tablas que resumen las métricas clave de cada simulación, permitiendo un análisis comparativo entre diferentes configuraciones.

Cualquier persona interesada puede clonar el repositorio del proyecto, instalar las dependencias necesarias y ejecutar el notebook para replicar los experimentos o realizar nuevas pruebas con diferentes parámetros.

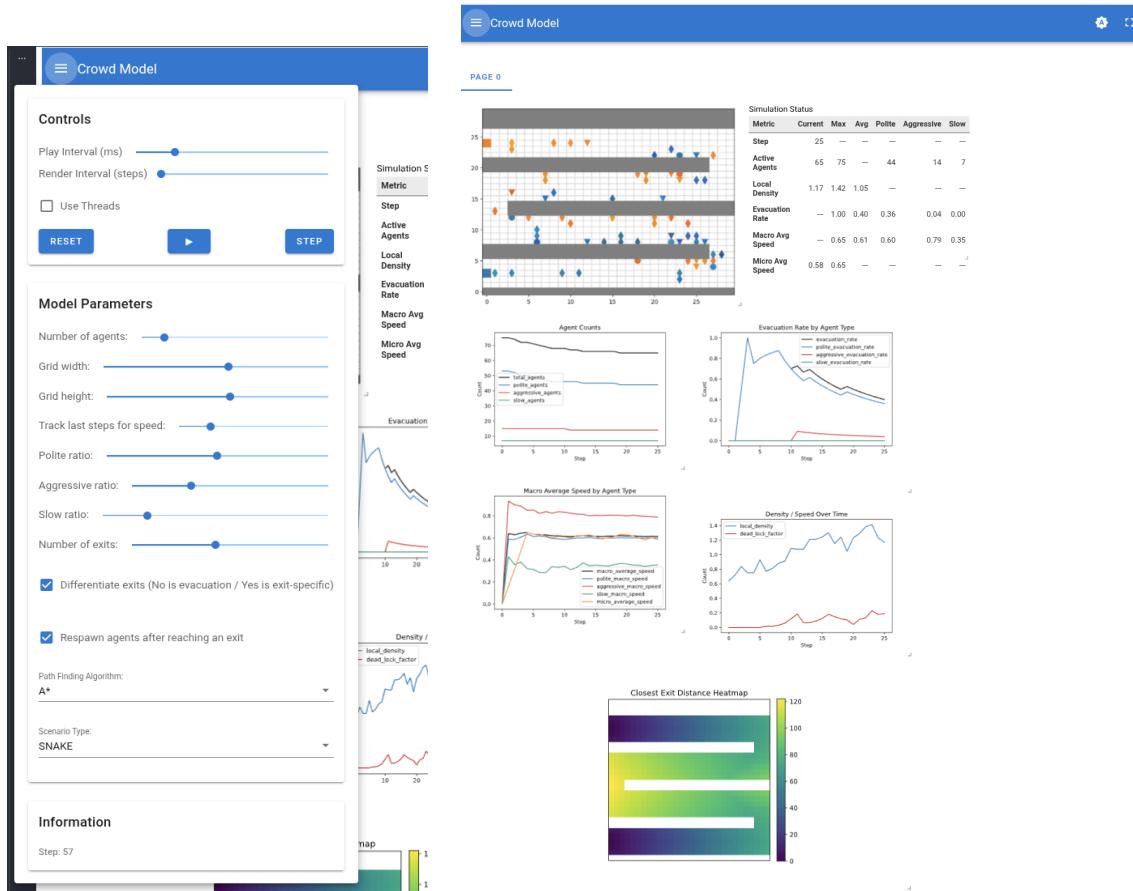


Figura 1.1: Capturas de pantalla del sistema de simulación final.

1.4 Estructura del Documento

El documento se estructura en: estado del arte, donde se revisan las tecnologías y usos actuales de estos sistemas; análisis del problema, donde se define la estrategia de modelado; evaluación experimental, detallando métricas y escenarios; implementación técnica; y finalmente los resultados y conclusiones obtenidos.

El desarrollo del código y la redacción de este trabajo se ha realizado totalmente a mano. Se ha hecho uso de asistentes como Copilot o Gemini como ayuda para la revisión y análisis de resultados, pero sin generar ningún contenido automáticamente. La única excepción es la generación automática gráficas y tablas a partir de los datos experimentales y referencias en el estado del arte para entender la situación actual de los sistemas. Todo lo que se ha generado ha sido añadido al proyecto bajo la supervisión y control de los autores.

CAPÍTULO 2

Estado del Arte

La simulación de evacuación ha evolucionado de ser una disciplina académica minoritaria, a convertirse en una herramienta crítica para garantizar la seguridad en situaciones de emergencia en infraestructuras modernas.

Este capítulo evalúa el estado del arte actual en simulación de evacuación, dividiendo el análisis en los objetivos que persiguen estos sistemas: los marcos regulatorios de seguridad en evacuaciones y las arquitecturas de micro-comportamiento utilizadas para modelar la navegación de agentes durante evacuaciones.

2.1 Objetivos de la Simulación de Evacuación

En el panorama actual de seguridad en edificaciones, la simulación de evacuación responde principalmente a regulaciones de seguridad y Diseño Basado en Prestaciones para garantizar evacuaciones efectivas en situaciones de emergencia.

2.1.1. Diseño Basado en Prestaciones (PBD) y Seguridad

Históricamente, los códigos de edificación eran toscos, dictando reglas rígidas sobre anchos de escalera y distancias de evacuación. El enfoque PBD permite diseños flexibles siempre que se demuestre, mediante simulación, que se cumplen los criterios de seguridad [2].

La métrica fundamental en PBD es la relación entre el Tiempo Disponible para la Evacuación Segura (ASET) y el Tiempo Requerido para la Evacuación Segura (RSET). Para que un diseño sea válido, debe cumplirse que:

$$ASET > RSET + \text{Margen_de_Seguridad} \quad (2.1)$$

Mientras que el ASET depende de la dinámica del fuego (toxicidad, calor, humo), el RSET es calculado por el simulador de multitudes y se compone de:

- **Tiempo de Detección y Tiempo de Notificación:** Darse cuenta del peligro y alertar.
- **Tiempo de Pre-evacuación:** El intervalo entre la alarma y el primer movimiento.
- **Tiempo de Viaje:** La fase física del movimiento desde el origen hasta una zona segura. Este es el componente principal que modela el sistema propuesto.

2.1.2. Nota sobre Nivel de Servicio (LOS)

Aunque existen metodologías como el Nivel de Servicio (LOS) para evaluar el confort peatonal en situaciones no urgentes [3], este trabajo se centra exclusivamente en evacuación de emergencia. Por tanto, las métricas de confort no son consideradas.

2.2 Marcos Regulatorios y Validación

Para que un modelo de simulación sea considerado una herramienta válida y no un mero ejercicio teórico, debe adherirse a estándares de validación internacionales.

- **ISO 20414:2020:** Esta norma establece protocolos rigurosos de verificación y validación para modelos de evacuación. Exige pruebas de componentes (por ejemplo, verificar que un agente se mueve a la velocidad asignada), verificación funcional (capacidad de flujo en puertas, contraflujo en pasillos) y validación cualitativa de comportamientos emergentes como la formación de arcos en las salidas [4].
- **NIST y SFPE:** El National Institute of Standards and Technology y la Society of Fire Protection Engineers proporcionan los datos demográficos y de comportamiento estándar (velocidades según edad, dimensiones corporales) que deben utilizarse para configurar los agentes [5].

Dado el alcance académico de este trabajo, no se pretende certificar el modelo bajo ISO 20414, pero se han seguido sus directrices para asegurar una base sólida y reproducible.

2.3 Arquitecturas de Navegación y Micro-comportamientos

La lógica de un simulador de multitudes reside en sus algoritmos de navegación local y resolución de conflictos. Existen dos tipos principales en la literatura:

2.3.1. Modelos de Espacio Continuo

Dominantes en la industria de la animación y la robótica por su fidelidad visual.

- **Social Force Model (SFM):** Propuesto por Helbing [6], trata a los agentes como partículas sometidas a fuerzas newtonianas. Combina una fuerza impulsora hacia la meta con fuerzas repulsivas para mantener la distancia personal.
- **Optimal Reciprocal Collision Avoidance (ORCA):** Opera en el espacio de velocidades para garantizar matemáticamente trayectorias libres de colisiones [7]. Aunque eficiente, puede resultar en comportamientos 'perfectos' o robóticos.

2.3.2. Modelos Discretos y Grid-Based

Los modelos basados en Autómatas Celulares (CA) o mallas discretas dividen el espacio en celdas. Son la base del sistema implementado en este trabajo. El desafío principal es la resolución de conflictos (cuando dos agentes compiten por la misma celda). El estado del arte actual emplea *Floor Fields* (campos de potencial) estáticos y dinámicos para guiar a los agentes, y utiliza teoría de juegos o heurísticas de "paciencia" para resolver bloqueos, superando las reglas simples de exclusión [8].

Ya que este es el elegido para el sistema, se profundiza en él más adelante.

CAPÍTULO 3

Análisis y definición del problema

El desarrollo de simuladores de evacuación implica equilibrar el realismo físico con la complejidad computacional. Dado el objetivo de estudiar comportamientos emergentes durante evacuaciones de grandes grupos, se requiere una aproximación que permita simular cientos de personas evacuando en tiempo real o cercano al tiempo real, por lo que se prioriza la ejecución del trabajo para la asignatura antes que la precisión milimétrica.

Además, el framework MESA en Python, aunque flexible y modular, no está optimizado para simulaciones de alta fidelidad física. Por lo tanto, esto nos deja como mejor opción un modelo discreto basado en rejilla que simplifica la representación espacial y las interacciones entre agentes.

3.1 Modelo: Discrete Grid-Based

Para llevar a cabo las simulaciones, se ha optado por un **Modelo Discreto basado en Rejilla (Grid-Based)**, implementado sobre el framework mencionado anteriormente.

Comparado con los modelos planteados en el estado del arte, esta elección presenta las siguientes cualidades:

1. **Espacio Discreto:** El entorno se divide en celdas cuadradas, donde cada celda se configura para contener únicamente un agente, una pared (obstáculo) o una salida. Esto simplifica la detección de colisiones y elimina los costosos cálculos de geometrías complejas realistas.
2. **Tiempo Discreto:** La simulación avanza en pasos discretos o 'steps' y la velocidad de los agentes se modelará probabilísticamente: un agente con 'mayor velocidad' simplemente tiene una mayor probabilidad de moverse en cada step y viceversa.
3. **Navegación:** La estructura de malla facilita la implementación directa de algoritmos de búsqueda de caminos en grafos como BFS y A*, esenciales para la planificación de rutas de los agentes. Con estos algoritmos, se pueden modelar las rutas óptimas hacia las salidas teniendo en cuenta los obstáculos.

BFS es simple y representa NO tener conocimiento previo del entorno (seguir solo las señales de evacuación), mientras que A* permite incorporar heurísticas para simular conocimiento previo o preferencias de ruta (sé que por aquí había una salida). Esto permite estudiar como afecta el conocimiento propio sobre él hecho de que las personas hagan caso a las señales de evacuación y no a su instinto.

3.2 Agentes: Comportamiento en Evacuación

Si se observan análisis reales de evacuaciones, las personas exhiben comportamientos heterogéneos bajo situaciones de emergencia. Para obtener resultados significativos sobre evacuación, el sistema debe ser capaz de modelar distintos perfiles de comportamiento de evacuación que interactúen entre sí.

En concreto, para este trabajo se han definido tres tipos de agentes con comportamientos diferenciados durante evacuaciones. Estos pretenden agrupar las características más relevantes observadas en emergencias:

- **Polite (Educados):** Serían la gente por defecto, educados y cooperativos. Deben ser capaces de ceder el paso y esperar, evitando conflictos activos y no imponerse ante los demás.
- **Aggressive (Agresivos):** Modelan la competencia y el pánico. Sería la gente que corre y se pone por delante. Su lógica ha de priorizar su movimiento sobre el de los demás, ocupando espacios libres sin consideración.
- **Slow (Lentos):** Modelan la diversidad física, gente con dificultades, niños o personas mayores. Actúan como obstáculos dinámicos y permiten estudiar el impacto de los usuarios más vulnerables en el flujo general.

La lógica e interacción de estos agentes será muy similar, pero con pequeñas variaciones que los cualifican. Esto permitirá observar dinámicas emergentes interesantes.

En concreto, se evaluarán comportamientos en choques de flujo, cuellos de botella y congestión en pasillos estrechos.

CAPÍTULO 4

Implementación

El sistema ha sido implementado utilizando Python y la librería **MESA**. Como se ha mencionado, MESA es un framework modular para el modelado basado en agentes y es bastante popular porque permite una rápida prototipación, una fácil extensibilidad de comportamientos y la recolección eficiente de métricas estadísticas.

La solución propuesta combina un entorno discreto (grid) eficiente computacionalmente, con agentes heterogéneos que exhiben comportamientos complejos a la hora de evacuar a partir de reglas simples (navegación y evasión de bloqueos entre otros).

Todos los parámetros del sistema son fácilmente configurables (distintas densidades de agentes, composiciones de tipos de agentes y topologías de escenarios). En `./app/src/config/config.py` y `./app/src/definitions` se encuentran todos los parámetros ajustables del sistema.

4.1 Entorno y Navegación (Floor Fields)

El espacio se modela mediante una malla ortogonal (`OrthogonalMooreGrid`) donde cada celda puede contener en cada momento un obstáculo, una salida o un único agente.

Para la navegación, se evita el recálculo de rutas a cada paso que tendría en cuenta los agentes moviéndose. En su lugar, se pre-calculan **Mapas de Distancia Estáticos**: al inicio, se inunda la malla (BFS/A*) desde las salidas, asignando a cada celda su distancia a una salida, creando un 'mapa en el suelo' o *Floor field*.

Usando los valores asignados a cada celda de la malla, los agentes realizan una especie de descenso de gradiente discreto, eligiendo la celda vecina con menor coste.

$$d(c_{actual}) \geq d(c_{vecino}) \quad \text{Se mueve de } c_{actual} \text{ a } c_{vecino}$$

Diferentes mapas pueden generarse para distintas salidas si se requieren rutas específicas, gestionando así flujos complejos (Fig. 4.2). Por defecto, cada salida creará su propio mapa de distancia, y los agentes elegirán la salida más cercana usando el menor valor de entre todos estos mapas si no se especifica lo contrario.

4.1.1. Heurística A*

Para simular un conocimiento del escenario, y no solo seguir posibles indicaciones sobre la salidas que pueda tener este, se integra una heurística Euclíadiana ponderada (factor 0.5) en la generación del mapa. Esto suaviza las rutas en espacios abiertos, reduciendo el movimiento en zigzag típico de las mallas Manhattan pero da algo de realismo al comportamiento y simula este 'conocimiento'.

4.2 Comportamiento de los Agentes

Como ya se ha mencionado en el análisis, las masas de gente son heterogéneas y están compuestas por diferentes perfiles. Para recrear este hecho, se definen tres perfiles cuya lógica de decisión en cada ciclo (*step*) sigue tres etapas:

4.2.1. 1. Intención de Movimiento (Velocidad y Densidad)

La probabilidad de moverse en un turno depende de:

- **Velocidad Base:** Probabilidad p_{mov} intrínseca del agente que se elige en un rango según el tipo (alta para Agresivos, baja para Lentos). A cada tick se mueve si un número aleatorio $R \in [0, 1]$ es menor que esta probabilidad.
 - **Polite:** $p_{mov} \in [0,65, 1,0]$
 - **Aggressive:** $p_{mov} \in [0,8, 1,0]$
 - **Slow:** $p_{mov} \in [0,5, 0,65]$
- **Fricción Social (Crowd Slowdown):** Si la densidad local es alta (> 3 vecinos), hay una probabilidad de que el siguiente movimiento NO se produzca, simulando la dificultad de moverse en aglomeraciones. Para los agentes *Polite* este efecto es mayor, los *Aggressive* son los menos afectados, mientras que los *Slow* se quedarían entre medias.

4.2.2. 2. Resolución de Prioridad

Para simular esta agresividad / agentes que no dudan respecto al resto de agentes, el sistema resuelve los movimientos por fases: primero actúan los **Agresivos**, seguidos aleatoriamente por **Polite** y **Slow**.

Esta estrategia simple dota a las multitudes de una cualidad de imposición física, de forma eficiente, y que a la vez que crea la ilusión de diferentes niveles de agresividad. Esto se hace aplicando `shuffle_do` a los agresivos antes que al resto.

4.2.3. 3. Gestión de Deadlocks (Pánico)

Si un agente desea moverse pero ninguna celda al rededor cumple el criterio de movimiento ($d_{vecino} \leq d_{propia}$), incrementa un contador interno de frustración (C_{dl}) en uno. Este contador se usa para relajar la condición de movimiento, ponderando el contador con un factor distinto por tipo de agente (menor en agresivos que en el resto).

$$d_{vecino} \leq d_{propia} + C_{dl} \times U_{tipo}$$

El valor C_{dl} se contagia a los vecinos inmediatos, simulando la propagación del pánico en situaciones de congestión. Esto crea dos estados de comportamiento:

- **Estado Normal:** Solo se mueve a celdas que mejoran la distancia a la salida $d_{vecino} \leq d_{propia}$. Es ' \leq ' y no ' $<$ ' para permitir movimientos laterales.
- **Estado de Pánico:** Al saturarse, el agente 'rompe' la regla de optimización, aceptando más movimientos laterales o incluso de retroceso para desbloquear la situación. Además, 'contagia' esta frustración a sus vecinos, forzando una reorganización local del grupo.

4.3 Visualización y Herramientas

Para el análisis, se ha usado la interfaz web interactiva que ofrece mesa, para mostrar información en tiempo real, como los mapas de calor de navegación o gráficas de métricas de flujo en tiempo real (Fig. 4.1)

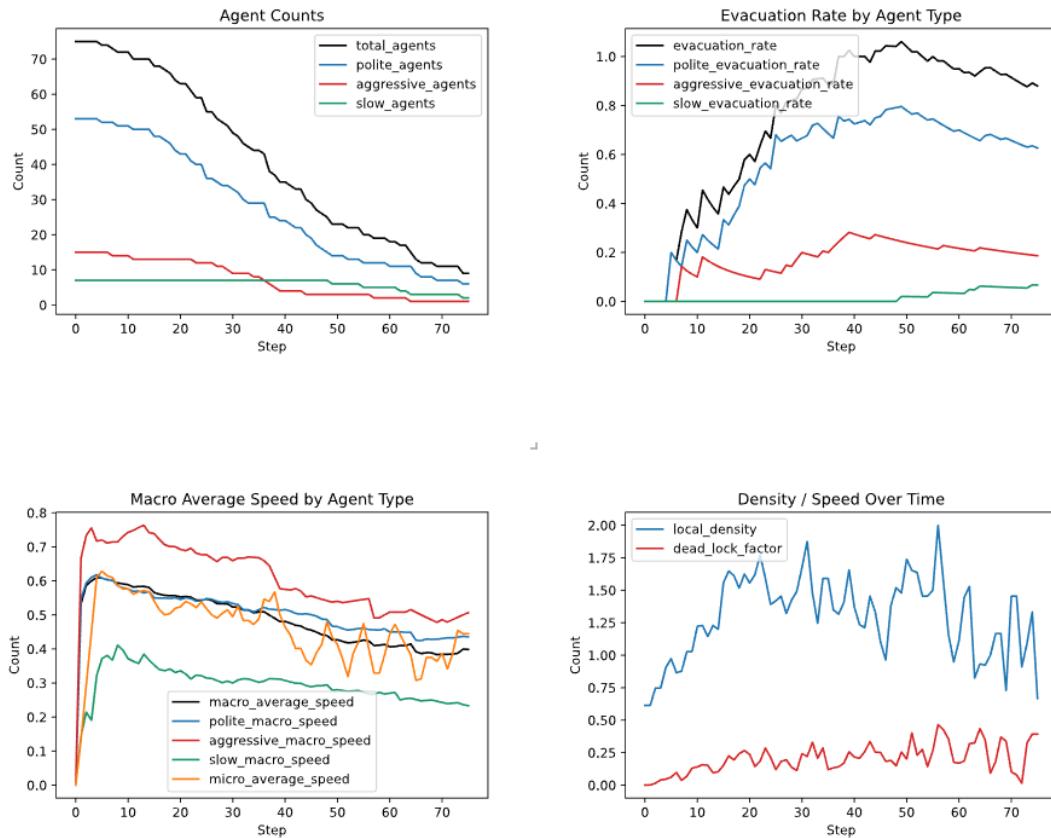


Figura 4.1: Interfaz de visualización con métricas en tiempo real.

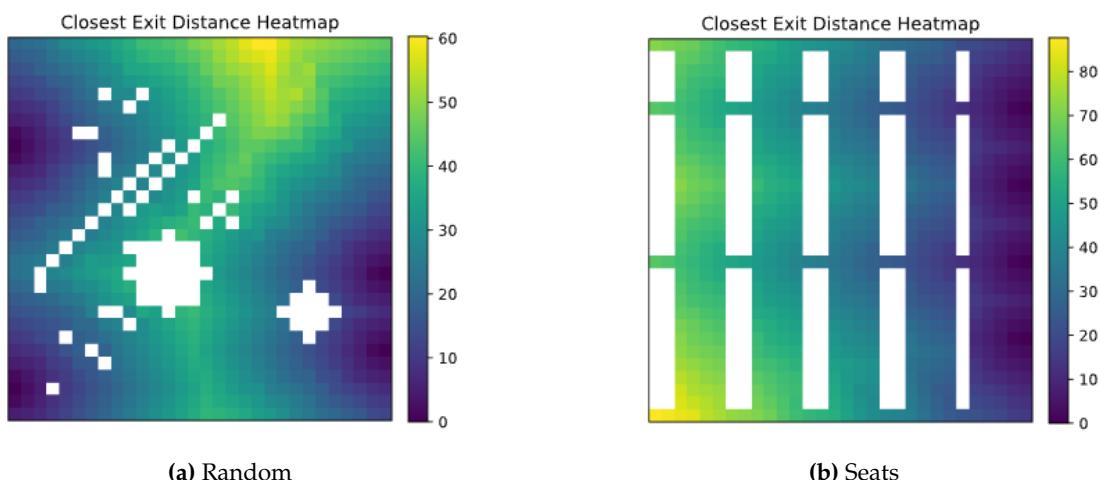


Figura 4.2: Mapas de calor de distancias de navegación (Floor Fields). Morado indica zonas cercanas a la salida, amarillo zonas lejanas.

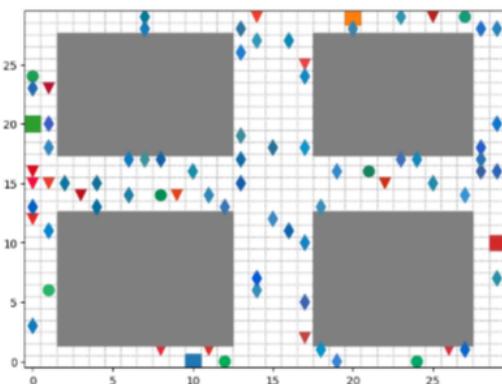
4.3.1. Representación de los entornos

Usando la visualización de MESA, se ha implementado un sistema que nos permite ver en tiempo real el estado de la simulación. Cada celda de la malla tendrá una figura con un color distinto según su contenido:

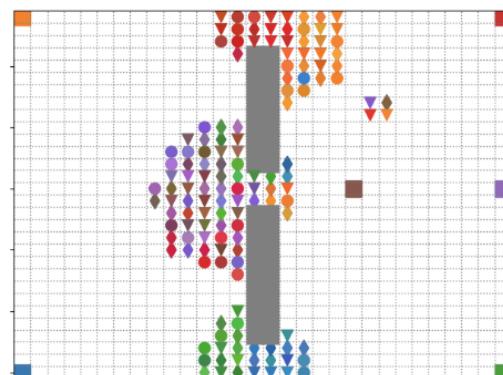
- **Paredes/Obstáculos:** Celdas con cuadrados grises.
- **Salidas:** Celdas con cuadrados de diferentes colores.
- **Agentes:** Según el tipo de agentes, se usarán diferentes geometrías y colores. Por defecto, cuando la simulación sea de evacuación sin preferencia de salida:
 - **Polite:** Rombos azules.
 - **Aggressive:** Triángulos rojos.
 - **Slow:** Círculos verdes.

Cuando la simulación sea de evacuación con preferencia de salida, los agentes cambiarán su color según la salida a la que se dirijan.

En todos los casos los agentes tendrán un tono de color ligeramente distinto al base para poder distinguirlos mejor y seguir su movimiento durante la simulación.



(a) Agentes rombos azules (Polite), triángulos rojos (Aggressive), círculos verdes (Slow).



(b) Ejemplo de agentes con distintos colores según la salida a la que se dirigen.

Figura 4.3: Visualización de entornos de simulación en MESA: tipología de agentes en evacuación por puerta más cercana (izquierda-derecha) y por puerta de preferencia (derecha).

CAPÍTULO 5

Evaluación

La validación y análisis del modelo propuesto se realiza mediante una serie de experimentos controlados. En este capítulo se detallan las métricas diseñadas para cuantificar el comportamiento de la multitud y la metodología de los experimentos.

5.1 Definición de Variables e Hipótesis

Para estructurar la evaluación, se identifican las variables del estudio y se plantean las hipótesis de partida.

5.1.1. Variables del Estudio

- **Variables Independientes:** Parámetros controlados que definen los escenarios (Densidad, Composición de la Multitud, Topología, Algoritmo de Navegación, Estrategia de Salida).
- **Variables Dependientes:** Métricas observadas para medir el desempeño (Tiempo Total, Flujo de Salida, Factor de Bloqueo, Velocidad Media).

5.1.2. Hipótesis de Trabajo

Se plantean las siguientes hipótesis a validar mediante la simulación:

1. **H1 - Saturación no lineal:** El tiempo de evacuación crecerá exponencialmente con la densidad en entornos cerrados.
2. **H2 - Costo de la Agresividad:** Se parte de la premisa clásica de que la agresividad, al reducir la cooperación (esperar turno), aumentará los conflictos y, por tanto, el tiempo total de evacuación.
3. **H3 - Costo de la Heterogeneidad:** La presencia de agentes lentos afectará al flujo desproporcionadamente respecto a su cantidad.
4. **H4 - Las indicaciones son recomendables:** La navegación que sigue las indicaciones de emergencia (BFS) superará a la planificada (A^*) en entornos con alta densidad de obstáculos desconocidos.

5.2 Métricas de Evaluación

Se han implementado monitores de datos (Data Collectors, clase de MESA) que registran paso a paso el estado de la simulación.

Aunque las unidades son abstractas (celdas, ticks), estas métricas son análogas a las utilizadas en estudios de seguridad profesional:

- **Tiempo de Evacuación (Total Steps):** Cantidad de ticks necesarios para que el último agente abandone el escenario. Representa el RSET (Required Safe Egress Time) y es la métrica principal de eficiencia.
- **Tasa de Evacuación (Evacuation Rate):** Cantidad de agentes que alcanzan una salida por tick. Permite visualizar la constancia del flujo e identificar cuellos de botella (caídas en el flow rate).
- **Densidad Local (Local density):** Se calcula para cada agente la ocupación de sus 8 celdas vecinas. El promedio global de esta métrica indica el nivel de congestión del sistema.
- **Factor de Bloqueo (Deadlock Factor):** Métrica específica diseñada para este sistema discreto. Mide la proporción de agentes que *intentaron* moverse pero no han podido debido a que sus celdas objetivo estaban ocupadas. Un alto factor de bloqueo indica fricción ineficiente o colapso del flujo.

Como ya se ha mencionado, esta métrica se usa para gestionar el comportamiento de los agentes en situaciones de congestión y alterar su lógica de movimiento.

- **Velocidad Macro y Micro:**

- *Velocidad Macro:* Velocidad promedio de todo el conjunto de agentes en el sistema a lo largo de toda la simulación.
- *Velocidad Micro:* Velocidad individual efectiva 'actual'. Mide la velocidad media pero solo teniendo en cuenta los últimos 4 ticks.

Luego, también se recogen métricas de cada una por cada tipo de agente (agresivos, educados y lentos) además de las globales.

5.3 Diseño de Experimentos

Se han diseñado escenarios específicos para poner a prueba distintas hipótesis sobre la dinámica de multitudes. Aunque el sistema que se ha creado soporta simulación de flujo continuo, este estudio se centra exclusivamente en el **Escenario de Evacuación**, donde el objetivo es vaciar estos recintos simulados.

5.3.1. Escenarios

Se han definido 6 escenarios (ver Figura 5.1) con topologías distintivas para evaluar diferentes aspectos de la navegación:

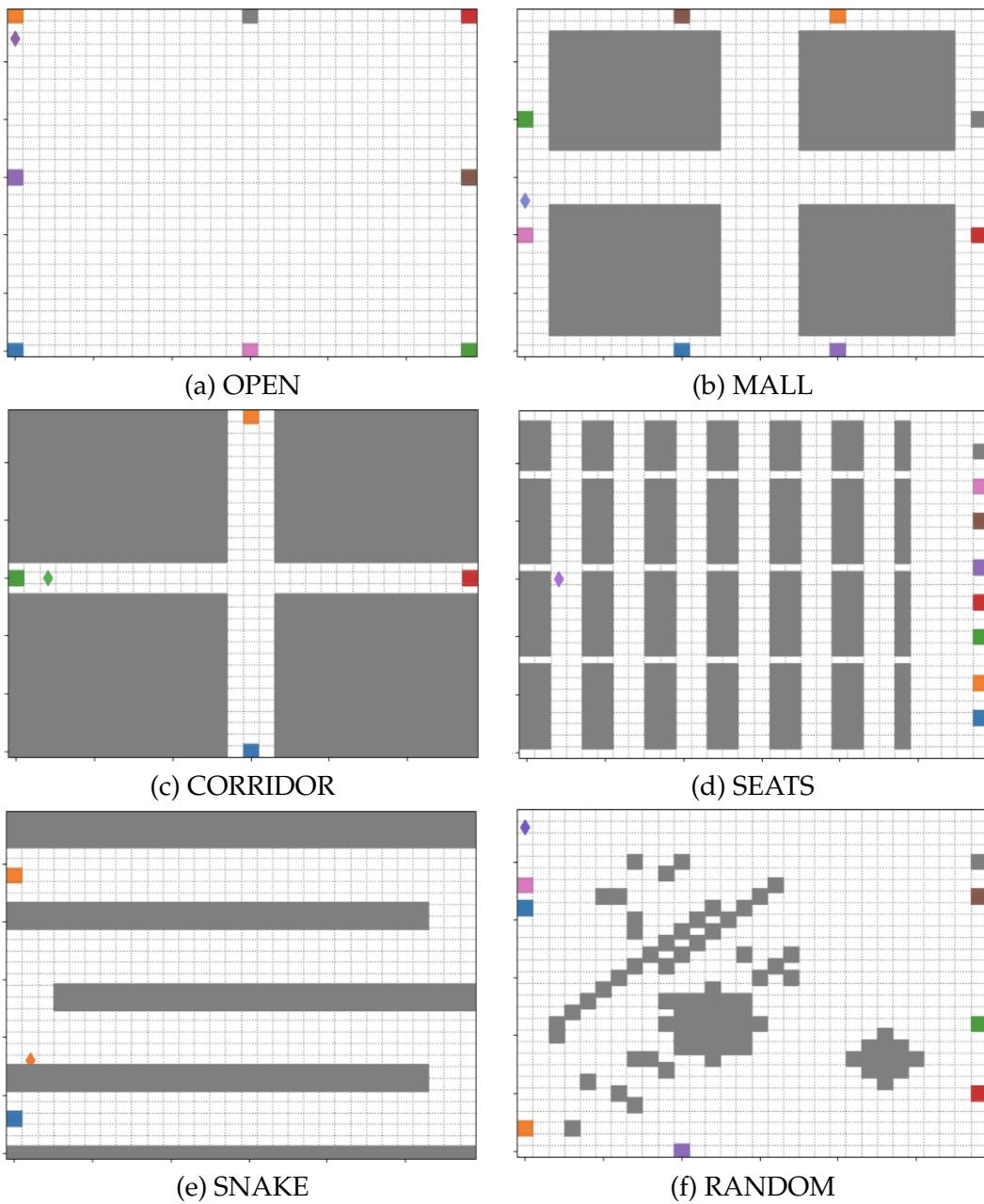


Figura 5.1: Escenarios implementados: (a, f) Control; (b, c) Pasillos y cruces; (d) Alta densidad de obstáculos; (e) Geometría compleja; Todos con sus respectivas salidas en cuadrados de colores

- **OPEN:** Un espacio abierto sin obstáculos. Hay salidas en los bordes de la malla. No es de mucho interés para este trabajo, pero sirve como referencia y espacio de experimentación.
 - **MALL:** Pretende simular un centro comercial donde se juntan varios pasillos con tiendas al rededor. Los agentes pueden moverse por los pasillos y por el anillo exterior, pero no pueden atravesar las tiendas. Las salidas están de dos a dos en los bordes de la malla.
 - **CORRIDOR:** Similar a MALL, pretende simular la intersección de varios pasillos con poco espacio para moverse. Las salidas están de dos a dos en los finales de cada pasillo.

- **SEATS:** Pretende simular un auditorio, cine o concierto, en el que todas las salidas están en un lado y el resto de la sala está llena de filas de asientos con 'pasillos' entre medias.
- **SNAKE:** Simula un pasillo zigzagueante con una o dos salidas en los extremos. La idea detrás de este escenario es ver el comportamiento de los agentes en espacios estrechos y con giros. Podrían simular una cola de personas en entradas a eventos.
- **RANDOM:** Por último, este escenario genera obstáculos (paredes, círculos y cuadrados) de forma aleatoria en la malla. Las salidas están en los bordes de la malla. Al igual que OPEN, este escenario no es de mucho interés para el trabajo, pero sirve como referencia y espacio de experimentación.

Es importante mencionar que las salidas NO se modelan incrustadas en las paredes, sino como **celdas libres en los bordes** de la malla. Los agentes pueden salir del escenario al llegar a una celda de salida (cualquiera de su vecindad) y esto se puede hacer de 5 en 5 (dado el vecindario de Moore). Esto quiere decir que el flujo máximo que cada salida puede soportar es de 5 agentes por tick.

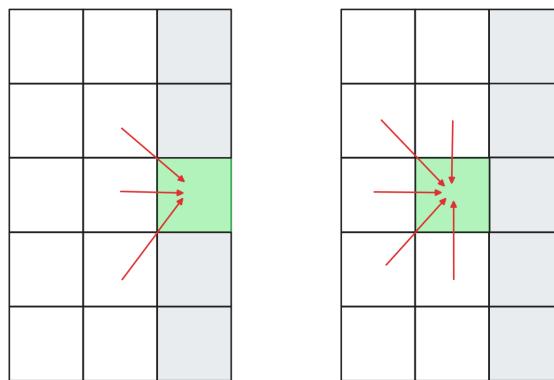


Figura 5.2: Posición de las salidas importa y puede afectar a la cantidad de agentes que pueden evacuar rápidamente.

En caso de haberlas incrustado, el flujo máximo sería de tres. Es una decisión de diseño a destacar, ya que con este modelo 'generoso' cualquier agente al lado de una salida puede evacuar instantáneamente. Esto puede no ser totalmente realista en escenarios de alta densidad, ya que las puertas físicas tienen un ancho limitado, pero evita cuellos de botella artificiales en el borde de la simulación.

5.4 Configuración y Protocolo Experimental

Para garantizar la reproducibilidad de los resultados, a continuación se detallan los valores por defecto utilizados en todas las simulaciones, así como las modificaciones específicas realizadas para cada conjunto de experimentos.

5.4.1. Valores por Defecto

Salvo que se especifique lo contrario en la descripción de cada experimento, las simulaciones parten de una configuración base establecida (Tabla 5.1). Cada simulación se repite **1000 veces** para asegurar la significancia estadística de los datos.

Parámetro	Valor por Defecto
Dimensiones de la Malla	30 x 30 celdas
Población Inicial	125 agentes
Algoritmo de Pathfinding	A* (con heurística Euclídea)
Escenario Base	MALL
Número de Salidas	4 salidas (distribuidas en los bordes)
Comportamiento de Salida	Oportunista (Salida más cercana)
Reaparición	Desactivada (Respawn = False)
Distribución de Agentes	70 % Polite (Educados), 20 % Aggressive (Agresivos), 10 % Slow (Lentos)

Tabla 5.1: Configuración base para las simulaciones, elegidas tras experimentación no documentada.

5.4.2. Definición de Experimentos

Se han llevado a cabo las siguientes baterías de experimentos, modificando parámetros específicos respecto a la configuración base (resumen en Tabla 5.2):

1. **Impacto de la Densidad:** Evalúa la saturación en distintos escenarios con poblaciones de 10 a 300 agentes (puros y mixtos).
2. **Impacto de la Agresividad:** Analiza en el escenario Seats cómo varía el flujo al aumentar el porcentaje de agentes agresivos (0 % a 100 %).
3. **Impacto de Agentes Lentos:** Analiza en el escenario Seats el efecto de obstáculos dinámicos que crean los agentes lentos (0 % a 100 % lentos).
4. **Comparativa de Pathfinding:** Contrasta navegación informada (A*) vs evacuación ciega (BFS).
5. **Preferencia de Salidas:** Mide el coste de imponer salidas fijas frente a la elección oportunista. Analiza el impacto de la rigidez en la toma de decisiones con cosas como 'salir por donde has entrado' o 'es que mi coche está por ahí'.
6. **Número de Salidas:** Busca el punto de saturación del flujo de evacuación según la cantidad de puertas disponibles.
7. **Ancho de Salida:** Analiza el impacto de ampliar supuestos 'cuellos de botella' (pasillos entre asientos de 1 a 4 celdas) en el escenario Seats. Decimos supuestos porque es lo primero que se piensa al ver el escenario.

Experimento	Variaciones	Objetivo
1. Densidad	10-300 agentes	Saturación del entorno.
2. Agresividad	0-100 % agresivos	Efecto en bloqueos.
3. Lentos	0-100 % lentos	Impacto obstáculos dinámicos.
4. Pathfinding	A* vs BFS	Heurística vs Reactivo.
5. Preferencias	Asignada vs Cercana	Impacto rigidez decisión.
6. Num. Salidas	1-8 salidas	Capacidad geometría.
7. Ancho Salida	1-4 celdas	Capacidad puertas.

Tabla 5.2: Resumen de variables experimentales y objetivos.

CAPÍTULO 6

Resultados Experimentales

Este capítulo presenta un análisis de los experimentos de simulación realizados para evaluar el impacto de diversos parámetros sobre la eficiencia de evacuación y la dinámica de multitudes. El análisis se ha llevado a cabo siguiendo la configuración experimental detallados en la Sección 5.4.

El análisis se estructura en dos partes complementarias:

1. **Análisis de Resultados Totales:** Evaluación de los tiempos totales de evacuación y métricas de comportamiento promedio a lo largo de toda la simulación.
2. **Análisis de Evolución Temporal:** Estudio de cómo las métricas clave evolucionan durante el transcurso de la evacuación, revelando patrones de congestión y dinámicas de flujo.

Las simulaciones se han centrado en recopilar los **Steps** totales requeridos para que todos los agentes evacuen, junto con **Métricas de Comportamiento** detalladas (velocidad, densidad, bloqueo, flujo) registradas en cada paso de la simulación para entender tanto la *eficiencia final* como la *calidad del proceso* de evacuación.

Se analizan a continuación los resultados obtenidos para las 7 variables experimentales definidas en la Tabla 5.2.

6.1 Análisis de Resultados Agregados

Esta sección examina los **resultados finales** de las simulaciones, analizando tiempos totales de evacuación y promedios de métricas de comportamiento a lo largo de toda la ejecución en cada experimento.

6.1.1. Densidad de Agentes

Observación: El tiempo de evacuación aumenta de manera no lineal con el número de agentes, mostrando una relación que indica saturación del sistema.

Resultados: Incrementar la población de 10 a 300 agentes causa aumentos en el tiempo de evacuación que varían significativamente según el escenario. En particular, el aumento en steps promedio es:

- **Corridor:** +80 % de steps (de ~50 a ~90 steps)
- **Mall:** +150 % de steps (de ~45 a ~113 steps)

- **Snake:** +200 % de steps (de ~60 a ~180 steps)
- **Seats:** +370 % de steps (de ~50 a ~235 steps)



Figura 6.1: Impacto del número de agentes en el tiempo total de evacuación en diferentes escenarios.

Interpretación: Los entornos altamente estructurados con obstáculos internos y cuellos de botella (Seats) son extremadamente sensibles a la densidad poblacional y se saturan rápidamente. En contraste, los espacios con flujo más directo (Corridor) manejan el incremento de carga de manera más lineal, permitiendo que los agentes se distribuyan mejor a lo largo de los pasillos.

La métrica de **densidad local promedio** confirma esta observación, aumentando de 0.18 vecinos (10 agentes) a 2.10 vecinos (300 agentes), lo que indica que los agentes tienden a estar más agrupados y por lo tanto ir más lentos.

Hay que mencionar que por el tamaño elegido, en corridor NO cabían más agentes a partir de 175, por lo que no se han podido simular densidades mayores en este escenario. Aún teniendo todo el espacio ocupado, los agentes son capaces de evacuar en un tiempo razonable.

6.1.2. Comportamiento Agresivo

Observación: En contra de la hipótesis inicial (H2) que predecía un mayor conflicto, los resultados muestran que la agresividad puede mejorar la eficiencia global del sistema.

Resultados: En el escenario Seats con 200 agentes:

- 0 % agresivos: 238 steps promedio
- 25 % agresivos: 225 steps (-5.5 %)
- 50 % agresivos: 215 steps (-9.7 %)
- 100 % agresivos: 205 steps (-13.9 %)

El **factor de bloqueo** disminuye del 0.73 (en educados) al 0.61 (en agresivos).

Interpretación: Este resultado rechaza H2, en lugar de entorpecer, los agentes agresivos mejoran el flujo. Al priorizar su propio movimiento y ocupar espacios disponibles sin vacilación, resuelven situaciones de bloqueo más rápidamente que los agentes educados, que tienden a esperar y bloquearse por querer ceder el paso.

Además, hay que mencionar que los parámetros de movimiento de estos agentes agresivos (mayor probabilidad de moverse y menor sensibilidad a la densidad) les permiten aprovechar mejor las oportunidades de movimiento, reduciendo la congestión.



Figura 6.2: Impacto de la proporción de agentes agresivos en Seats (200 agentes).

6.1.3. Agentes Lentos

Observación: Similar a lo sucedido con los agentes agresivo pero al revés. Los agentes con movilidad reducida actúan como obstáculos dinámicos que empeoran significativamente la evacuación.

Resultados: En escenario Seats con 200 agentes:

- 0 % lentos: 218 steps promedio vs 100 % lentos: 316 steps (+44.9 %)

Interpretación: La velocidad macro promedio cae de 0.37 (sin lentos) a 0.26 (100 % lentos), una reducción del 30 %. La alta variabilidad sugiere que la *posición* de los agentes lentos (especialmente en cuellos de botella) es más determinante que su cantidad.

De forma similar, estos agentes tienen parámetros de movimiento peores que el resto, lo que provoca que el flujo general se ralentice al tener que esperar a estos 'obstáculos móviles'.



Figura 6.3: Impacto de la proporción de agentes lentos en Seats (200 agentes).

6.1.4. Algoritmos de Pathfinding

Observación: El rendimiento relativo de A* frente a BFS es altamente dependiente de la topología del entorno.

Resultados:

- **Mall** (espacio abierto): A* 33 % más rápido que BFS (105 frente a 140 steps)
- **Seats** (obstáculos densos): BFS 8 % más rápido que A* (228 frente a 248 steps)

Interpretación: A* optimiza rutas en espacios abiertos, pero en entornos complejos puede llevar a *mínimos locales* haciendo que los agentes tiendan a acercarse a esquinas o, al haber congestión, a esquivarse 'peor'. BFS, al explorar uniformemente, resulta más robusto en escenarios complejos, sugiriendo que "no saber nada" es a veces más seguro que una heurística imperfecta.

Hay que recordar que el peso de la heurística es 0.5, lo que crea un balance entre exploración y explotación. Aún así, en escenarios complejos, este balance no es suficiente para evitar que los agentes queden atrapados en rutas subóptimas si están siendo bloqueados por otros.

6.1.5. Preferencias de Salida

Observación: La asignación rígida de salidas preferidas es catastrófica comparada con elegir la salida más cercana.

Resultados: El tiempo aumenta un 120 % en Mall y hasta un 1,000 % en Snake. El **factor de bloqueo** se dispara de 0.13 a 3.33.

Interpretación: El problema fundamental es el **tráfico cruzado**: agentes cruzando el centro de la sala para ir a una salida lejana bloquean a los que van a una cercana. Es inviable que los agentes NO evacuen por la puerta más próxima en escenarios densos, las ideas de 'salgo por donde he entrado' y simulares matan más que cualquier otro efecto.

6.1.6. Ancho de Salida

Observación: Ampliar el pasillo de salida En Seats, satura la mejora del rendimiento al pasar de 1 a 2. Pasar de 1 a 2 celdas mejora un 30 %, pero de 3 a 4 es despreciable (<2 %).

Interpretación: Ver que la mejora se satura tan rápido, descarta este 'supuesto' cuello de botella como principal limitante. Con un mínimo de pasillo, se da que otro factor está tomando más relevancia a la hora de limitar las salidas.



Figura 6.4: Impacto asintótico del ancho de salida en Seats (125 agentes).

6.1.7. Número de Salidas

Observación: En todo los espacios existe un punto de saturación en el que, añadir más salidas, no mejora significativamente el tiempo de evacuación.

Resultados: En Mall (200 agentes), pasar de 1 a 4 salidas mejora el tiempo un 56 %, pero de 4 a 8 solo un 5 % adicional. El cuello de botella se desplaza de las puertas a la capacidad de los agentes de llegar a estas.

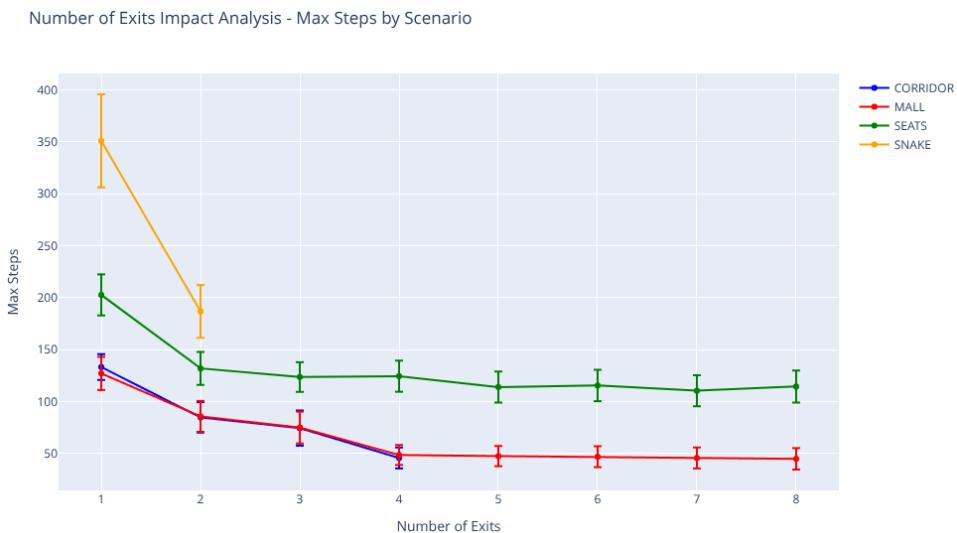


Figura 6.5: Saturación del beneficio al aumentar el número de salidas (Mall, 200 agentes).

Interpretación: Tanto en este caso como en el del ancho de salida, el sistema permite ver que las limitaciones y cuellos de botella, no siempre están donde uno espera. En entornos complejos, si bien mejoras esto supuesto cuello de botella (más salidas, más anchas) ayuda, el verdadero limitante se encuentra en otro punto, y es ahí donde hay que centrar los esfuerzos de investigación, en ver qué partes hay que optimizar y no solo centrarse en una.

6.2 Análisis de Evolución Temporal

Para comprender bien las simulaciones y no solo quedarse con los promedios globales, vamos a estudiar la evolución de estas.

Para poder hacerlo de forma adecuada, se ha normalizado el tiempo de las simulaciones ($t_{\text{norm}} \in [0, 1]$) dividiendo entre la duración total de la simulación.

Esto permite comparar la evolución de métricas como el bloqueo y la velocidad entre simulaciones de distinta duración, ya que de otra forma no tendría sentido ver en qué tick ocurre un pico de bloqueo si unas simulaciones duran 50 ticks y otras 300.

6.2.1. Dinámica de las Preferencias de Salida

El análisis nos permite ver por qué las preferencias no permiten evacuaciones fluidas. Mientras que el escenario sin preferencias mantiene el factor de bloqueo bajo hasta el final, vemos que en el caso con preferencias, el bloqueo crece exponencialmente en la fase de congestión (10 – 50 % del tiempo), alcanzando un pico **722 % mayor** antes de tiempo ($t_{\text{norm}} \approx 0,42$).

La velocidad promedio cae drásticamente un 20 % del tiempo *antes* de que ocurra el bloqueo máximo, lo que encima sirve como un indicador del atasco que crea el tráfico cruzado.

Por lo que se puede sacar la conclusión de que el aumento tan dramático del tiempo de evacuación, se debe a que los agentes crean bloqueos que han de resolver, lo que no les permite avanzar a las salidas.

6.2.2. Dinámica del Ancho de Salida

En escenarios complicados como Seats, ampliar las salidas de 1 a 4 celdas apenas reduce el tiempo total, pero altera cuándo ocurre la congestión.

Salidas más anchas *adelantan* el pico de bloqueo (del 75 % al 45 % del tiempo), aunque no reducen su magnitud. Esto confirma que el cuello de botella es el movimiento entre asientos, y que la velocidad constante de los agentes (0.28) refleja movimiento lento, que seguramente se deba a esperar a otros agentes a dejarles espacio.

Ampliar la salida mejora el flujo, pero no acelera el llegar a ellas, por lo que este escenario necesitaría de mejor optimización.

6.3 Discusión y Limitaciones

Como se ha visto, el modelo captura dinámicas emergentes complejas, pero presenta ciertas limitaciones:

- **Ausencia de Colas en Salidas:** No se observaron aglomeraciones densas o largas colas en las puertas mismas, tal como ocurren en la realidad (cuellos de botella). Esto es atribuible al modelado "generoso" de las salidas descrito en la Sección 5.3, donde el flujo permitido es de hasta 5 agentes/tick, disipando la presión en el borde.
- **Falta de Re-planificación Dinámica:** Los agentes no reaccionan a la congestión más que paniqueando. Una mejor solución podría ser redirigirse a salidas menos saturadas. Esta rigidez empeora los bloqueos en escenarios de preferencias, comportamiento que se podría usar para recrear el "balance de carga" natural que hacen los humanos.

CAPÍTULO 7

Conclusiones y Trabajo Futuro

7.1 Conclusiones

Este trabajo valida el uso de sistemas multi-agente discretos (MESA) para simular evacuaciones, demostrando que comportamientos complejos emergen de reglas simples de interacción local.

El sistema desarrollado ha permitido aislar y cuantificar variables críticas que determinan la seguridad de un recinto, arrojando luz sobre la relación entre el diseño arquitectónico y el comportamiento humano.

Las principales conclusiones derivadas de la experimentación se detallan a continuación:

7.1.1. La Geometría como Factor Determinante

La topología del entorno es el factor más influyente en la eficiencia de la evacuación, validando la hipótesis **H1**.

En espacios abiertos (Open, Mall), el flujo depende de la capacidad de las salidas; sin embargo, en entornos complejos (Seats), los obstáculos internos crean cuellos de botella previos a la salida que marcan el ritmo de evacuación, causando un crecimiento exponencial del tiempo contra más agentes participen.

Cabe mencionar que, debido al modelado de salidas como celdas libres (no incrustadas), no se observaron las largas colas características en los bordes de salida, desplazándose los bloqueos principales hacia el interior de la sala. Aún con esta ventaja, se concluye que **ninguna optimización del comportamiento de los agentes compensa un diseño físico mal calculado**.

7.1.2. Paradoja del Conocimiento y Costo de la Rígidez

Algo de lo más sorprendente confirma la hipótesis **H4** sobre la robustez de los sistemas reactivos. Los experimentos demuestran que:

- La navegación inteligente.^o informada (A^*) es propensa a caer en mínimos locales y llevar a agentes por caminos no tan óptimos, mientras que la evacuación que sigue 'el camino marcado por las señales' (BFS) resulta más robusta, superando a 'saber por donde estaba la salida' en escenarios complejos.
- La **preferencia de salidas** genera flujos cruzados catastróficos, aumentando el tiempo de evacuación hasta un 1000 % en algunos escenarios.

Esto sugiere que, en pánico, la **flexibilidad y el comportamiento oportunista** son estrategias superiores.

7.1.3. Impacto de la Heterogeneidad de la Multitud

La composición de la multitud altera fundamentalmente la dinámica del flujo:

- **Agresividad como Catalizador:** Se **refuta la hipótesis H2**. Contrario a la intuición de que la falta de cooperación empeoraría la evacuación, un porcentaje de agentes agresivos mejora la eficiencia en sistemas discretos, evacuando más rápido.

Habría que ver el comportamiento de estos agentes con flujos que chocan para ver si el mismo efecto se mantiene, o por otro lado esta agresividad impide resolver Deadlocks.

- **Fragilidad ante Agentes Lentos:** Se confirma la hipótesis **H3**. El sistema es desproporcionadamente sensible a los agentes lentos. Un pequeño porcentaje actúa como obstáculos móviles que pueden empeorar la evacuación.

A la hora de diseñar espacios, es crucial considerar la presencia de individuos con movilidad reducida y su impacto en la dinámica general.

7.1.4. Importancia de la Re-planificación Dinámica

Una limitación observada, que sirve como conclusión, es el impacto de la falta de re-planificación. Los agentes, al no re-evaluar sus rutas en función de la congestión, tienden a unirse a bloqueos en lugar de buscar salidas alternativas.

Esto confirma que la **percepción de densidad en tiempo real** es un componente esencial para una evacuación eficiente, más allá del conocimiento estático y de las indicaciones que pueda tener un escenario.

7.1.5. Dinámica Temporal

El análisis temporal revela que el colapso del flujo (deadlock masivo) es predecible. La caída en la velocidad media sucede justo antes que a la creación de esto, actuando como un indicador.

Además, se observa que ciertas mejoras físicas (como ampliar pasillos de salida en entornos densos) no resuelven el problema, sino que simplemente desplazan el momento en que ocurre la congestión.

7.2 Trabajo Futuro

- **Re-planificación Dinámica:** Implementar mapas de distancias dinámicos (que los agentes actualicen por densidad) para que los agentes eviten zonas congestionadas proactivamente.
- **Movimiento Continuo:** Usar una implementación continua y no basada en celdas (sub-grid movement) para permitir adelantamientos y flujos sin las limitaciones del modelo de celda discreta.
- **Validación Empírica:** Comparar los resultados y los comportamientos emergentes del modelo con datos reales de evacuaciones para calibrar y validar las reglas de comportamiento.

Bibliografía

- [1] Jackie Kazil, David Masad y Andrew Crooks. *MESA: Agent-based Modeling in Python*. Python framework for agent-based modeling. 2020. URL: <https://mesa.readthedocs.io/>.
- [2] Morgan J. Hurley et al. *SFPE Handbook of Fire Protection Engineering*. Oct. de 2015. DOI: [10.1007/978-1-4939-2565-0](https://doi.org/10.1007/978-1-4939-2565-0). URL: <https://doi.org/10.1007/978-1-4939-2565-0>.
- [3] *Client challenge*. URL: <https://www.scribd.com/document/371131651/PedestrianPlanningAndDesignJJFruin-1971>.
- [4] ISO 20414:2020. URL: <https://www.iso.org/standard/78348.html>.
- [5] *Publications | NIST*. URL: https://www.nist.gov/publications/search_by_author/1148671?k=&t=&n=&d%5Bmin%5D=&d%5Bmax%5D=&page=1.
- [6] Dirk Helbing y Péter Molnár. «Social force model for pedestrian dynamics». En: *Physical review. E, Statistical physics, plasmas, fluids, and related interdisciplinary topics* 51.5 (mayo de 1995), págs. 4282-4286. DOI: [10.1103/physreve.51.4282](https://doi.org/10.1103/physreve.51.4282). URL: <https://doi.org/10.1103/physreve.51.4282>.
- [7] Jur Van den Berg et al. *Optimal Reciprocal Collision Avoidance (2D)*. URL: <https://gamma.cs.unc.edu/ORCA/>.
- [8] M. Davidich y G. Köster. «Towards automatic and robust adjustment of human behavioral parameters in a pedestrian stream model to measured data». En: *Safety Science* 50.5 (feb. de 2012), págs. 1253-1260. DOI: [10.1016/j.ssci.2011.12.024](https://doi.org/10.1016/j.ssci.2011.12.024). URL: <https://arxiv.org/abs/0802.1620>.