



INSTITUTO TECNOLÓGICO Y DE ESTUDIOS SUPERIORES DE MONTERREY

Campus Estado de México

Modelación de sistemas multiagentes con gráficas computacionales (Gpo 301)

Revisión 4-Reflexión Individual: Iker Fuentes

Profesores:

Jorge Adolfo Ramírez Uresti

Sergio Ruiz Loza

Integrantes:

Carlos Iker Fuentes Reyes

A01749675

Análisis de la solución desarrollada

1. ¿Por qué seleccionaron el modelo multiagentes utilizado?

A la hora de diseñar el sistema multiagentes presentado, tomamos muchas cosas en cuenta, en primera instancia, usamos la actividad de los robots de limpieza (la introducción a mesa) como parteaguas de los requerimientos técnicos de la solución que queríamos presentar.

Asimismo, analizamos los requerimientos planteados en la descripción del reto, para hacernos una idea de los agentes que necesitaríamos crear y las interacciones que tendrían con su ambiente. En esta primera categorización de los agentes, hicimos uso de los conceptos vistos en clase sobre las características mínimas con las que un agente debería de contar para poder ser considerado de esta forma.

Así fue como acotamos nuestros agentes a los siguientes:

1. Vehículos [divididos en coches y camiones]
2. Peatones
3. Semáforos inteligentes

Mientras que definimos los siguientes componentes como elementos del ambiente con los que interactuarían:

1. Las calles [banquetas (aunque a nivel de mesa las banquetas cuentan como parte del edificio), rotondas o estándar]
2. Los edificios
3. Los estacionamientos
4. Las paradas de los camiones

Esto es desde la perspectiva de cómo decidimos construir el ambiente del modelo, ya entrando más de lleno en la solución planteada para el problema de movilidad dado por el reto, consideramos lo siguiente:

1. Los vehículos tienen que ser capaces de saber qué direcciones son válidas (simulando reglas de tránsito).
2. En las rotondas, los vehículos han de ser capaces de decidir cuál ruta tomar.
3. Los vehículos tienen que ser capaces de ver los semáforos e interpretar los colores para así tomar una decisión.
4. Los vehículos tienen que saber llegar a su estacionamiento de alguna forma. Movimiento aleatorio sería muy ineficiente y podría generar ciclos.
5. Los semáforos, para ser inteligentes deben de ser capaces de hablar entre ellos, interpretar la presencia de los vehículos y actuar conforme a ello.
6. Los agentes han de ser capaces de identificar la presencia de otros agentes, para evitar colisiones que entorpezcan el flujo vial.
7. Los vehículos han de saber que llegaron a su destino (los coches a sus estacionamientos y los camiones a sus paradas).

Asimismo, sabíamos que teníamos que hacer una optimización al flujo vial, por lo que implementamos una técnica de colaboración entre multiagentes, donde los coches, cada vez que llegan a su estacionamiento registran en un elemento común, al cual llamamos Waze, los pasos que siguieron en la ruta.

Internamente Waze se maneja como un grafo direccionado con ponderación, el cual implementa las siguientes funcionalidades:

1. Pregunta si hay una ruta directa entre estacionamientos. Si no, usa el algoritmo de Dijkstra para construir un grafo para conectar los estacionamientos (en la representación los estacionamientos son los nodos y el costo asociado a cada unión es el número de pasos para llegar de uno a otro).
2. En caso de que no exista una ruta, los vehículos siguen buscando y registrando rutas. El registro de ruta se hace con el mejor valor descubierto. Es decir, si dos vehículos encontraron una ruta de un estacionamiento a otro, pero una es más corta, la más corta toma prioridad.

Para los semáforos, usamos como guía la actividad de plantear semáforos inteligentes, y terminamos por hacer la implementación. Los semáforos hablan entre vecinos, hacen un juicio de quien tiene prioridad con base en la cercanía de vehículos y la cantidad de estos.

2. ¿Cuáles fueron las variables que se tomaron al momento de tomar la decisión?

Las variables consideradas abarcaron tanto aspectos técnicos como de comportamiento para asegurar un diseño funcional, escalable y eficiente:

1. Interacción entre agentes y ambiente:

- **Normas de tránsito:** Cómo garantizar que los vehículos sigan reglas y tomen decisiones informadas basándose en el entorno.
- **Comunicación entre agentes:** Métodos para que los vehículos y semáforos compartan información y actúen de manera coordinada.
- **Prevención de colisiones:** Definición de reglas para evitar conflictos entre agentes móviles (vehículos, peatones).

2. Optimización del flujo vial:

- **Rutas eficientes:** Cómo hacer que los vehículos encuentren estacionamientos de manera óptima, minimizando movimientos innecesarios.
- **Registro colaborativo:** Uso del sistema *Waze* para almacenar y mejorar rutas descubiertas, priorizando siempre las más cortas.
- **Inteligencia en semáforos:** Evaluación de la capacidad de los semáforos para gestionar prioridades dinámicamente.

3. Escalabilidad y rendimiento del sistema:

- **Número de agentes:** Evaluación del comportamiento del sistema con diferentes cantidades de vehículos, peatones y autobuses.
- **Puntos críticos:** Identificación de zonas donde tienden a reunirse más agentes y su impacto en el flujo.
- **Capacidad de adaptación:** Cómo el sistema responde a cambios, como un aumento en la densidad de tráfico.

4. Métodos de comunicación:

- Establecimos mecanismos específicos para que:
 - Los vehículos pudieran identificar y responder a las señales de tránsito y otros agentes.
 - Los semáforos pudieran negociar prioridades entre intersecciones.

3. ¿Cuál es la interacción de esas variables con respecto al resultado de la simulación?

La simulación final demostró las siguientes características:

1. Una interacción adecuada entre los vehículos (no se tuvieron coches, se esperaron los coches a los camiones, se comunicaron las mejores rutas descubiertas)
2. Un buen control de los semáforos. Instancias con más de 100 agentes llevaron a embotellamientos que se resolvieron rápido.
3. Las rutas descubiertas incrementaron considerablemente la eficiencia del proyecto. Pruebas previas a la implementación arrojaron resultados de 24 agentes tardando 800-1000 pasos en llegar a sus destinos, mientras que con la implementación (previo a peatones y camiones) hasta 400 agentes de tipo coche llegaron a sus estacionamientos en menos de 1600 pasos.
4. Los coches siguieron los sentidos correctos de las calles y respetaron los semáforos.

Más puntualmente se tuvieron las siguientes interacciones de las variables y los resultados finales:

1. Métodos de comunicación:

- Los vehículos y peatones eran capaces de verse mutuamente y detenerse para evitar colisiones.
- Los peatones interactuaron con los camiones.
- Los vehículos y los semáforos se “comunicaban” en el sentido de que el vehículo le anunciaba al semáforo que se acercaba (equivalente a un sensor).
- Los vehículos veían si tenían un semáforo en la dirección de su movimiento y revisaban su estado para determinar su siguiente acción.

2. Normas implícitas en el ambiente

- Respetar los semáforos es un comportamiento implícito en los agentes de tipo vehículo. Si ves un semáforo enfrente tuyo y éste se encuentra en rojo, esperas a que se ponga en verde para avanzar.
- Los vehículos seguían las normas de tránsito de dos formas:
 - Las calles en las que circulaban tenían la dirección implícita que los agentes conocían al interactuar con ella (moverse al grid).
 - Tenían una serie de comportamientos permitidos y prohibidos (como cambiarse a una calle en sentido contrario, o empezar a ir en dirección opuesta con respecto a su camino actual). Estos comportamientos son evaluados/decididos a la hora de decidir la siguiente posición.

4. ¿Por qué seleccionaron el diseño gráfico presentado?

Dos cuestiones importantes surgieron a la hora de decidir el diseño:

1. Practicidad (escala) para una integración más sencilla con respecto a las dimensiones de la simulación realizada en mesa. Tanto para los vehículos como para la construcción del ambiente.
2. Se utilizaron, en su mayoría, assets importados a la hora de construir la ciudad, para así enfocarnos en la integración funcional más que la gráfica. Lo que fue diseñado, fueron los vehículos utilizados para la simulación. La justificación de diseño de cada uno fue por el gusto individual de cada miembro del equipo y sus intenciones de hacer de este proyecto más divertido.

5. ¿Cuáles son las ventajas que encuentras en la solución final presentada?

Considero que nuestra solución final presenta las siguientes ventajas:

1. A nivel de mesa [modelo]

- a. La implementación de una técnica de cooperación entre agentes para compartir información y para construir un plan global parcial (de cómo ir entre estacionamientos).
- b. La implementación de una heurística simple de movimiento para cuando los agentes intentan descubrir su estacionamiento por primera vez (reducir la distancia manhattan), junto con un factor de aleatoriedad que evita que se ciclen los agentes al seguir la heurística (como la evaluación es la misma para un punto determinado, puede llegar a un punto donde se queda siempre yendo en la misma dirección en vez de intentar rutas nuevas).
- c. La implementación de semáforos inteligentes para la gestión del tráfico es algo que permite evitar embotellamientos o bien, resolverlos rápidamente.
- d. Integración implícita de las leyes de tránsito en las direcciones de las calles y en la toma de decisiones de los autos.

2. A nivel de Unity

- a. Una integración efectiva con el proyecto de mesa, al tener un manejador de agentes en Unity (con manejador de agentes, me refiero a los códigos que instancian a los vehículos, peatones y camiones) y el conjunto de semáforos predefinido, se puede acceder al JSON generado del servidor de Flask y mandado a la clase de conexión, para así dar las posiciones y estados (información sobre la simulación) a cada uno.
- b. Asimismo, la lógica de movimiento implementada considero que es eficiente a la hora de resolver el problema presentado. Yo me enfoqué más en la parte de mesa y en la integración con Unity, pero considero que esta parte del movimiento de los vehículos, haciendo uso de las transformaciones por medio de matrices para representar el movimiento, le da una apariencia natural a la simulación.

6. ¿Cuáles son las desventajas que existen en la solución presentada?

Considero que la mayor debilidad de la simulación podría ser el comportamiento de los peatones, puesto que no es una planeación inteligente, más que nada es una caminata pseudoaleatoria, pero planeada de tal forma que no se cicle. Asimismo, existió un caso particular que se dio a la hora de seguir la ruta más eficiente, en el que si se tenía un obstáculo que podía frenar el objeto, se saltaba un paso, algo que no afectaba tal cual en mesa, pero sí podría haberlo hecho en unity, pues era importante el tener todos los puntos acomodados.

La solución llegó a trabar la posición en algunos casos, por lo que tuvo que hacerse un caso de particular en el que si llevaba mucho tiempo atorado, regresara a su ruta pseudoaleatoria para evadir los obstáculos. Además, el que todos los objetos se esperen a que todos lleguen a su posición original hace que la simulación se vea un poco más lenta.

7. ¿Qué modificaciones podrías hacer para reducir o eliminar las desventajas mencionadas?

Con respecto a los peatones, sería implementar algo similar al waze, pero con destinos diferentes (tal vez las paradas de autobús) o bien, marcar los diferentes conjuntos de edificios como destinos para ellos, además de generar un sistema de rutas que tome en cuenta las rutas de los camiones y que los peatones puedan acceder a ellos y hacer cierta planificación.

Para el problema del salto de posiciones (el cual ya no sucede), sería hacer más pruebas y optimizaciones a la construcción de la ruta, tal vez reemplazando la queue por una lista normal [similar a la implementación actual de los camiones] y llevando un índice para hacer el avance, en vez de hacer un pop a la queue para obtener la nueva posición.

Para la parte de los movimientos, considero que la solución planteada es la mejor tomando en cuenta las limitaciones de tiempo y experiencia. Puede que no se vea dinámico, pero esto evita desfases que lleven a que los valores de la simulación no cuadren con lo esperado (o que llamen al servidor en momentos diferentes y esto lleve a desfases en la simulación con respecto al mesa). Podría ser que generamos un contenedor global de la información y que los agentes la llamasen para actualizar su posición. Aunque esta opción tendría que trabajarse más y probar diferentes escenarios.

Reflexión sobre mi proceso de aprendizaje

Recuperando lo planteado en la primera revisión:

Integrante	Fortalezas	Áreas de oportunidad	Expectativas	Compromisos personales
Carlos Iker Fuentes Reyes	Disciplina. Compromiso con el trabajo. Perfeccionismo. Capacidad de aprender rápido.	Organización. Formalización de la documentación. Paciencia y estandarización de código.	Deseo aprender tanto como me sea posible del modelaje e implementación de agentes. La IA es un área de interés para mí y desearía poder dedicarme a ello en un futuro.	Realizar el autoestudio necesario para llevar a cabo el proyecto de manera adecuada. Ir a las asesorías necesarias para asegurarme de llegar a un nivel que me permita resolver los problemas que surjan. Preguntar en clase, participar, entender y asegurarme de hacer un trabajo de alta calidad.

Habiendo llegado al final del bloque, y dándome el tiempo de reflexionar, creo fervientemente que he aprendido mucho, en particular en la parte de multiagentes (ya que fue de lo que principalmente me encargué yo), pues hice numerosas pruebas a la hora de crear los agentes, definir sus interacciones, y delimitar sus comportamientos. A lo largo del presente bloque, implementé varias soluciones para modelar su comportamiento y construí sus interacciones con el ambiente siguiendo los principios vistos en clase, junto con investigaciones propias sobre el tema y de mesa. Asimismo, creo que el haber implementado el Waze (el algoritmo de grafos para las rutas haciendo uso de la colaboración entre los agentes de tipo coche) me forzó a pensar de forma diferente en la colaboración de los agentes para construir una solución.

Fue mi primer acercamiento tan profundo con una rama de la IA (previamente había trabajado con LLM y visión por computadora, pero nunca al grado de hacer implementaciones así de profundas), por lo que siento que me hice de una mejor idea sobre lo que en realidad es la IA y cómo es que funciona, por lo menos en la rama de multiagentes. Agradezco bastante esta introducción al tema y espero poder seguir utilizando los temas aquí aprendidos para desarrollar proyectos futuros y, tal vez, estudiarlos en un grado de educación superior en el futuro. Me resulta muy interesante esta rama de las ciencias computacionales y considero que dan un marco de solución interesante para diversos problemas que son de mi interés, como el modelaje de sistemas delimitados por cierto conjunto de reglas y características de los agentes, o simulaciones de escenarios basados en información que permiten generar modelos de respuesta.

Asimismo, con la parte de Unity, aprendí mucho sobre la forma en la que se modela el movimiento a través de matrices y transformaciones. Sinceramente, jamás le había dado demasiado pensamiento a cómo es que las computadoras calculan de manera interna los movimientos que vemos en unity, por lo que el tener que hacer implementaciones de cero, además de aprender sobre conceptos de texturas, iluminación y técnicas de optimización fue una experiencia bastante enriquecedora. Considero que este grado de detalle visto en las implementaciones de Unity, me permitió adquirir un conocimiento más fundamental sobre las gráficas computacionales y muchos conceptos que me serán de utilidad en el futuro. En particular los conceptos de optimización me parecieron interesantes, espero seguir aprendiendo de esto en el futuro y aplicarlos en mi carrera profesional o en mis proyectos personales.

