



Tecnológico de Monterrey

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: Alan Vega

Profesores:

Jorge Adolfo Ramírez Uresti

Sergio Ruiz Loza

Integrantes:

Alan Rodrigo Vega Reza A01750658

Análisis de la solución desarrollada

¿Por qué seleccionaron el modelo de multiagentes utilizado? Se tomaron en cuenta principalmente las cuestiones particulares del framework de multiagentes mesa, a partir de la actividad de las roombas de limpieza sobre todo fue que empezamos a tener una idea de la forma en la que trabajaba mesa y como podíamos traducir la ciudad y sus diferentes agentes a un modelo en mesa, del mismo modo, había varios requerimientos descritos previamente que tuvimos que checar para poder hacernos una idea de qué agentes íbamos a necesitar crear y las diferentes interacciones que tendrían entre ellos y su ambiente. En esta primera categorización, los agentes terminaron quedando divididos en las siguientes categorías:

- a. Vehículos (subdivididos en coches y camiones)
- b. Peatones
- c. Semáforos inteligentes

También dividimos el ambiente en los siguientes componentes:

- Calles (subdividido en calles normales, rotondas, pasos peatonales y banquetas (estas no tienen como tal su propio elemento en mesa sino que forman parte de los edificios))
- Edificios
- Estacionamientos
- Paradas de camiones

Entrando más en detalle, algunas de las cuestiones planteadas para resolver el problema de movilidad planteado en el reto fueron las siguientes:

- Los vehículos deben de poder saber los sentidos de tránsito válidos
- Los vehículos tienen que ser capaces ver e interpretar los semáforos de manera correcta
- Los vehículos tienen que poder tomar decisiones en la rotonda para poder continuar con su ruta.
- Los vehículos tienen que ser capaces de saber cómo llegar a su estacionamiento (el movimiento aleatorio no es una solución adecuada)
- Los semáforos tienen que poder comunicarse entre ellos, poder detectar presencia de vehículos y cambiar los colores de manera correcta para permitir el paso de los vehículos y peatones.
- Los agentes deben de poder identificar la presencia de otros agentes para evitar colisiones.
- Los vehículos deben de poder detectar que llegaron a su destino (estacionamientos para los autos, paradas de autobús para los autobuses).

Para mejorar la navegación de los agentes y optimizar el tránsito, se implementó una técnica de colaboración entre agentes, donde los coches, cada vez que llegan a su estacionamiento de destino, registran los pasos de su ruta en un elemento común al que todos los demás agentes pueden acceder para conocer rutas, a esto le llamamos Waze.

Internamente, Waze es un grafo direccionado con ponderación, su función es preguntar si hay una ruta directa entre los estacionamientos. En caso de que no, usa el algoritmo de Dijkstra para armar un grafo que conecte a los estacionamientos, siendo cada estacionamiento un nodo, y el costo asociado a cada unión el número de pasos para llegar de un estacionamiento a otro. En caso de que no exista una ruta, los vehículos tienen que seguir buscando y

registrando las rutas, la ruta que muestre el Waze siempre será la ruta con el mejor valor que haya sido descubierta. Si dos vehículos encuentran rutas diferentes de un estacionamiento a otro, la más corta será la que tome prioridad.

Los semáforos hablan con sus semáforos vecinos y deciden qué carril tiene prioridad basándose en la cantidad y cercanía de los vehículos.

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

1. Relación agentes – ambiente:
 - Normas de Tránsito: Cómo garantizar que los vehículos sigan las reglas de tránsito y tomen decisiones basándose en la información del entorno.
 - Comunicación entre agentes: Que los vehículos puedan compartir información entre ellos y con los semáforos.
 - Prevención de colisiones: Definir reglas para que los agentes puedan evitar conflictos con otros agentes (vehículos y peatones)
2. Optimización de tránsito
 - Rutas eficientes: Lograr que los vehículos lleguen a su destino con la menor cantidad de movimientos posibles.
 - Colaboración entre agentes: Uso de Waze para almacenar y actualizar rutas para que los agentes siempre tengan acceso a las rutas más cortas posibles.
 - Inteligencia en semáforos: Capacidad de los semáforos para evaluar y gestionar las prioridades de manera dinámica.
3. Rendimiento y escalabilidad
 - Número de agentes: Evaluar cómo se comporta el sistema con diferentes números de vehículos y peatones.
 - Puntos críticos: Identificación de zonas donde se acumulen más agentes y su impacto en el rendimiento.
 - Capacidad de adaptación: Respuesta del sistema ante cambios en el número de agentes, o en la densidad del tráfico.
4. Métodos de comunicación: Se establecieron mecanismos para que por un lado los vehículos puedan identificar y responder al entorno y a otros agentes y para que los semáforos puedan negociar entre ellos para dar prioridades en las intersecciones.

¿Cuál es la interacción de esas variables con respecto a los resultados de la simulación?

La simulación final posee las siguientes características:

- Los vehículos interactúan adecuadamente entre sí, no hay choques ni atropellamientos, los coches esperan a los camiones, las mejores rutas son descubiertas y comunicadas
- Los semáforos controlan de manera eficiente el tráfico, incluso con instancias de más de 100 agentes resuelven embotellamientos de manera rápida.
- La implementación para descubrir rutas mejoró considerablemente la eficiencia de la simulación, pasando de requerir de 800 a 1000 pasos para una simulación con 24 agentes (únicamente autos) a poder manejar hasta 400 agentes (únicamente autos) en menos de 1600 pasos.
- Los vehículos siguen los sentidos de las calles y respetan los semáforos.

Métodos de comunicación

- Vehículos y peatones pueden verse mutuamente y detenerse para evitar colisionar.
- Peatones pueden interactuar con los camiones
- Vehículos y semáforos se “comunican”; el vehículo le anuncia al semáforo cuando viene acercándose.
- Los vehículos checan los semáforos que tienen en frente y revisan el color para determinar si pueden avanzar o no

Normas del ambiente

- Respetar los semáforos es una regla implícita de los vehículos.
- Los vehículos siguen una serie de normas de tránsito tales como:
 - o Las calles en las que circulan tienen una dirección implícita que los agentes deben seguir
 - o Existen comportamientos permitidos y prohibidos, tales como cambiarse al carril en sentido contrario, empezar a ir en dirección opuesta respecto al sentido actual, estos comportamientos se evalúan con cada posición calculada.

¿Por qué seleccionaron el diseño gráfico presentado?

Principalmente fueron decisiones guiadas por la practicidad, por un lado la escala se hizo así para una integración más sencilla con el modelo de mesa. Por otro lado, se decidieron usar en su mayoría assets importados para poder construir la ciudad, esto para poder enfocarnos en la integración funcional con el modelo. Los vehículos tipo auto sin embargo, si fueron diseñados por cada uno de los miembros del equipo (nótese que esto excluye el autobús). Estos vehículos fueron diseñados a gusto individual de cada miembro con las intenciones de hacer el proyecto más divertido.

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

A nivel de modelo

- La implementación de técnicas de cooperación entre agentes para poder compartir información y contribuir al mejor funcionamiento del modelo a nivel global
- La implementación de una heurística de movimiento para cuando los agentes están descubriendo los estacionamientos al inicio y que permite reducir la distancia de manhattan junto con factores de aleatoriedad para evitar ciclos infinitos
- Implementación de semáforos inteligentes para gestionar el tráfico permite evitar embotellamientos y en caso de que ocurran, gestionarlos rápidamente.
- Integración de reglas de tránsito en el ambiente y en la toma de decisiones de los autos.

A nivel de Unity

- Integración efectiva con el proyecto de mesa, gracias al uso de archivos manejadores de agentes y un conjunto de semáforos predefinido, se puede acceder a un único JSON generado y enviado por el servidor de Flask para poder obtener posiciones y estados de cada paso del modelo.

- La lógica de movimiento implementada es eficiente a la hora de resolver el problema, se evitan choques a nivel visual, los vehículos realizan los giros de manera adecuada, similar a como lo harían en la vida real.
- Implementación de una cámara libre que usa técnicas de optimización para mejorar el rendimiento de la visualización en Unity.

¿Cuáles son las desventajas que existen en la solución presentada? Principalmente el comportamiento de los peatones, los cuales no tienen como tal un comportamiento inteligente, sino que utilizan un patrón de movimiento pseudoaleatorio, únicamente con las prevenciones adecuadas para evitar ciclos infinitos, del mismo modo existió un caso particular en el que en ocasiones en el modelo cuando había un obstáculo en la ruta más eficiente el modelo se saltaba un paso, lo cual si bien no causa ningún problema en el modelo, sí que puede afectar en la visualización de Unity, ya que esta requiere tener todos los puntos acomodados.

Existen casos en los que para evitar el ciclo o que el agente se atore, se requiere que el agente regrese a una ruta pseudoaleatoria para poder evadir los obstáculos. Del mismo modo, para evitar desincronizaciones en la visualización de Unity, la visualización espera a que todos los agentes alcancen su posición esperada, lo que hace que se vea un poco más lenta.

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

Los peatones podrían implementar algo similar al waze pero con referencias diferentes, quizás en lugar de buscar estacionamientos que los peatones busquen paradas de autobús y luego las actualicen en algo similar a waze, quizás se podrían marcar algunos edificios de interés común, como tiendas, centros comerciales, el estadio, parques, etc. Para que los peatones tengan más rutas disponibles para moverse por la ciudad de manera eficiente.

Por su parte, para la parte de movimientos, quizás se podría modificar el modelo en mesa para hacer un grid más grande que permita más grados de movimiento dentro de cada celda, ahorita mismo el movimiento en el grid es completamente discreto, mientras que Unity utiliza movimiento continuo, una solución posible que se me ocurre para mejorar la fluidez de movimiento sería que el modelo en mesa en lugar de que 1 celda de grid de mesa equivalga a 1m de grid de Unity, que cada metro en el grid de unity sea el equivalente a varias celdas en el mesa, esto no fue posible debido a que el tamaño del grid en el modelo estaba predefinido y no era posible cambiarlo, sin embargo algo que se me ocurre es que quizás el grid de mesa podría haber sido más grande, permitir que los agentes ocuparan más de un grid y así hacer un movimiento con más pasos en el grid del modelo, esto después sería más fácil de traducir en Unity para crear un movimiento más fluido debido a que en cada paso el coche requiere moverse menos.

Reflexión sobre el proceso de aprendizaje

Integrante	Fortalezas	Áreas de oportunidad	Expectativas	Compromisos personales
Alan Rodrigo	Autodidacta Perseverante	Organización	Quiero aprender sobre todo acerca de las cuestiones de	Me comprometo a prestar más atención a los diferentes

Vega Reza	Poder estar mucho tiempo resolviendo un problema hasta llegar a la solución	Hacer documentación Más atención a los aspectos visuales del proyecto	inteligencia artificial, pero también me llama mucho la atención la cuestión de gráficas computacionales y sobre todo cómo conectar ambas partes para que los datos del programa de multi agentes	aspectos del proyecto, a las cuestiones organizacionales y sobre todo a las cuestiones visuales del proyecto, y a poner especial atención y empeño a la escritura de documentos.
--------------	---	--	---	--

Como conclusión final a este bloque puedo decir que he aprendido mucho, sobre todo de la parte de gráficas computacionales, aunque de la parte de multiagentes también he aprendido bastante, sobre todo la parte de los cálculos matemáticos para poder hacer los diferentes movimientos de los autos, del mismo modo, un tema al que le tomé particular interés fue a la parte de la optimización, me pareció particularmente interesante la idea de poder definir qué está dentro del campo de visión de la cámara y decidir se renderizarlo o no.

Por otro lado la parte de sistemas multiagentes me pareció muy interesante sobre todo porque me dio ideas para usar en el equipo de robótica Quantum Robotics, dentro del equipo tengo una asignación para un algoritmo de evasión de obstáculos, y creo que estructurar toda la rutina del rover como un agente podría ayudar mucho a mejorar el comportamiento del rover en la competencia.

Considero que este bloque me dio muchos aprendizajes importantes y espero poder aplicarlos en proyectos futuros.