



Tecnológico de Monterrey

Reto: Movilidad Urbana

Rafael Blanga Hanono | A01781442

Leon Blanga Hasbani | A01784008

27 de Noviembre del 2024

Índice

Problemática:	3
Solución propuesta:	4
Diseño de los agentes	5
Agente Coche.....	5
La arquitectura de subsunción de los agentes	6
Arquitectura del agente inteligente “Coche”.....	6
Características del ambiente	7
No Determinista:.....	7
Dinámico:.....	7
Episódico:.....	7
Inaccesible:.....	7
Discreto:.....	7
Conclusiones:	8

Problemática:

En este proyecto, se desarrolló una simulación del tráfico vehicular en una ciudad utilizando el framework MESA. El objetivo principal es modelar el comportamiento de los coches en una ciudad compleja que incluye calles con direcciones específicas, semáforos que regulan el flujo vehicular y obstáculos como edificios. Los coches deben moverse desde su posición inicial hasta un destino determinado (el cuál se les asigna de manera aleatoria), respetando las reglas de tránsito, como los sentidos de las calles y el estado de los semáforos, y evitando colisiones con otros vehículos y obstáculos.

La simulación debe ser modelada en 3D, con el uso de objetos en formato .obj, mismos que deben ser procesados para que se adapten al contexto de manejo utilizado por WebGL, siendo esta la herramienta con la que se llevó a cabo la implementación del frontend. Así mismo, se busca que tanto la escena de la simulación, como los objetos presentes en ella, tengan un aspecto realista. Para lograr esto, se tiene que implementar el uso de iluminación -ambiental, especular y difusa-, e interpolación del movimiento, logrando de esta manera un increíble acercamiento a la realidad a través de gráficas computacionales. También, es importante recalcar que los agentes dinámicos (como coches y semáforos) presenten un comportamiento el cual siga las reglas predefinidas para cada uno en el backend con el uso del framework MESA, logrando finalmente, una simulación realista de la ciudad con agentes inteligentes interactuando en su ambiente.

Solución propuesta:

En nuestra solución, implementamos un modelo basado en agentes utilizando el framework MESA para simular el tráfico vehicular en una ciudad. Comenzamos por definir los agentes principales de la simulación: coches, semáforos, calles, destinos y edificios. Cada agente tiene un comportamiento específico que permite modelar una ciudad compleja y dinámica. Por ejemplo, los coches están diseñados para moverse hacia su destino siguiendo reglas de tránsito como los sentidos de las calles y el estado de los semáforos. Además, incorporamos una lógica de colisión, basada en un sistema de “reservas de espacio” para garantizar que los vehículos eviten ocupar la misma celda que otro coche.

Para integrar la simulación con un entorno 3D, utilizamos WebGL como herramienta para la visualización del frontend. Adaptamos el modelo diseñado en MESA (2D) del backend para que fuera compatible con WebGL, lo que nos permitió representar visualmente los agentes en un espacio tridimensional. Esto implicó convertir las posiciones del grid (x,y) en coordenadas tridimensionales y modelar sus movimientos con transformaciones (traslación y rotación). Además, procesamos distintos objetos en formato .obj a través de funciones desarrolladas por nosotros en javascript y el uso de los shaders, para representar a los coches, edificios y otros elementos estáticos y dinámicos del entorno.

Para lograr una conexión integral del proyecto, implementamos un servidor Flask que actúa como middleware entre el backend y el frontend. Este servidor cuenta con 5 endpoints los cuales hacen lo siguiente:

- Inicializar el modelo con la información recibida del frontend.
- Obtener los agentes estáticos del modelo (edificios, destino y calles) junto con sus respectivas variables y pasarlos al frontend para ser renderizados.
- Obtener los agentes dinámicos del modelo (semáforo y coches) junto con sus datos actualizados y pasarlos al frontend para ser renderizados.
- Avanzar un paso la simulación.
- Obtener las estadísticas recopiladas por el modelo y enviarlas al frontend.

Finalmente, para lograr una experiencia visual más realista, configuramos WebGL, junto con el fragment shader, de tal manera que en la escena donde se visualiza la simulación tenga iluminación -desde los semáforos, hasta el “sol”-, procesando tanto las normales de los objetos modelados, como la posición dinámica de la fuente de luz y de la cámara, con las cuales logramos obtener la suma de la luz ambiental, especular y difusa de todos los componentes en la escena. Así mismo, se implementó el uso de interpolación, para lograr asemejar lo más posible, el movimiento de los coches al que observamos en la vida real.

Diseño de los agentes

(objetivo, capacidad efectora, percepción, proactividad, métricas de desempeño, etc.).

Agente Coche

- Objetivo:

- El objetivo principal del coche es desplazarse desde su posición inicial hasta un destino específico, el cual se le asigna de manera aleatoria al inicio de la simulación. Durante este recorrido, el coche debe cumplir con distintas restricciones, como respetar las direcciones de las calles y el estado de los semáforos, así como la posición de otros coches en celdas vecinas para evitar colisiones,

- Capacidad Efectora:

- El coche tiene la capacidad de moverse de una celda a otra dentro del grid, pero su movimiento está condicionado por diversas restricciones. Puede desplazarse únicamente en la dirección permitida por las calles y debe detenerse si encuentra un semáforo en rojo en su camino. Además, si la celda de destino está ocupada por otro coche o un obstáculo, el agente buscará alternativas.

- Percepción:

- La capacidad del coche se diseñó para que sea capaz de percibir el estado de su entorno inmediato. Esto incluye detectar agentes y objetos en las celdas vecinas, como calles y su dirección, el estado (verde o rojo) de los semáforos, y la presencia de otros coches u obstáculos. También monitorea continuamente su posición actual dentro del grid y lo evalúa con respecto a la ubicación de su destino con un cálculo de la distancia euclidiana. Esta percepción le permite tomar decisiones informadas sobre su siguiente movimiento.

- Proactividad:

- El agente coche muestra su proactividad al planear y adaptarse mientras se mueve hacia su destino. Siempre busca el camino más corto, ajustándose si encuentra semáforos, calles en sentido contrario o lugares ocupados, y evita repetir rutas que ya recorrió para no quedarse atrapado. Si hay bloqueos, espera y vuelve a evaluar las opciones para seguir avanzando. De esta forma, se asegura de moverse de manera eficiente y respetando las reglas del tráfico en la simulación.

- Métricas de desempeño:

- En cuanto a las métricas de desempeño, identificamos y recopilamos las siguientes:
 - Accidentes que sufrió el automóvil: Representando la situación en que más de un agente de tipo Coche se encontrara en la misma celda en un mismo paso
 - Pasos que le tomó llegar a su destino.
 - Cantidad de infracciones de las leyes de tránsito: Representando el movimiento hacia celdas no permitidas por las limitantes del modelo, como pueden ser circular en sentido contrario, ocupar el espacio de un edificio o pasarse un semáforo.

La arquitectura de subsunción de los agentes.

Arquitectura del agente inteligente “Coche”

ARQUITECTURA DE SUBSUNCIÓN



Nuestra arquitectura de subsunción permite que el agente se enfoque primero en las tareas más importantes, como evitar choques. Al mismo tiempo, puede adaptarse y planear su movimiento de forma más estratégica cuando no hay tantas restricciones. La organización en capas asegura que las prioridades más altas, siempre sean atendidas antes que las menos importantes, logrando así un comportamiento eficiente y confiable.

Características del ambiente.

No Determinista:

- El ambiente es no determinista ya que los agentes no conocen exactamente el desenlace o efecto de sus acciones.

Dinámico:

- El ambiente es dinámico, porque está en constante cambio debido al movimiento de los coches, el cambio de los semáforos y las interacciones entre los diferentes elementos del entorno. Esto obliga a los agentes a adaptarse en tiempo real, tomando decisiones basadas en las condiciones actuales para cumplir sus objetivos y respetar las reglas del sistema.

Episódico:

- El ambiente es episódico porque el comportamiento de los agentes se evalúa y actualiza en pasos discretos, donde cada paso es independiente del siguiente en términos de decisiones. Esto significa que los agentes toman decisiones basadas únicamente en la información disponible en el momento actual, sin anticipar ni planificar explícitamente los efectos a largo plazo de sus acciones.

Inaccesible:

- El ambiente es inaccesible porque los agentes solo tienen información de sus celdas vecinas, sin conocer el mapa completo. Esto los obliga a tomar decisiones basadas únicamente en la información local que perciben.

Discreto:

- El ambiente es discreto porque tanto las acciones como las percepciones de los agentes están limitadas a un conjunto definido y finito. Los agentes pueden realizar un número específico de movimientos, como avanzar en las direcciones permitidas o detenerse en semáforos.

Conclusiones:

A lo largo de este proyecto, logramos cumplir nuestro objetivo principal de simular un entorno de tráfico vehicular en una ciudad compleja, integrando agentes inteligentes con comportamientos realistas y un sistema de visualización tridimensional en WebGL. Cada paso del desarrollo nos permitió enfrentar y superar retos significativos, como diseñar un ambiente dinámico, y asegurar una conexión fluida entre el backend y el frontend.

Uno de los aprendizajes más valiosos fue comprender la importancia de descomponer problemas complejos en componentes manejables, como los agentes y sus reglas específicas. Además, la implementación de conceptos teóricos, como el manejo de prioridades y la percepción limitada, nos permitió observar cómo estas ideas se traducen en comportamientos emergentes dentro del sistema. Este proyecto no solo reforzó nuestras habilidades técnicas en programación y diseño de sistemas, sino que también nos enseñó a colaborar de manera eficiente para desarrollar una solución integral a un problema práctico para distintas situaciones.