



Campus Santa Fe

Reporte del reto

Modelación de sistemas multiagentes con gráficas computacionales (Gpo 301)

Andrés Jaramillo Barón A01029079

Rodrigo Sosa Rojas A01027913

Octavio Navarro Hinojosa

Gilberto Echeverría Furió

29/11/2024

Problema que se está resolviendo

La movilidad urbana, se define como la habilidad de transportarse de un lugar a otro y es fundamental para el desarrollo económico y social y la calidad de vida de los habitantes de una ciudad. Desde hace un tiempo, asociar la movilidad con el uso del automóvil ha sido un signo distintivo de progreso. Sin embargo, esta asociación ya no es posible hoy. El crecimiento y uso indiscriminado del automóvil —que fomenta políticas públicas erróneamente asociadas con la movilidad sostenible— genera efectos negativos enormes en los niveles económico, ambiental y social en México.

Durante las últimas décadas, ha existido una tendencia alarmante de un incremento en el uso de automóviles en México. Los Kilómetros-Auto Recorridos (VKT por sus siglas en Inglés) se han triplicado, de 106 millones en 1990, a 339 millones en 2010. Ésto se correlaciona simultáneamente con un incremento en los impactos negativos asociados a los autos, como el smog, accidentes, enfermedades y congestión vehicular.

Para que México pueda estar entre las economías más grandes del mundo, es necesario mejorar la movilidad en sus ciudades, lo que es crítico para las actividades económicas y la calidad de vida de millones de personas.

Propuesta de solución

Debido a los problemas que el tráfico genera en la vida real, creamos un sistema de búsqueda de rutas efectivas para un entorno lleno de autos. El objetivo principal de la simulación es optimizar el tráfico que se genera en ciertas rutas de un punto a otro, trazando alternativas mientras se respetan las direcciones de tránsito y se respetan señales de alto como semáforos. Los autos se generan en las 4 esquinas del tablero cada cierto intervalo, todos con un destino determinado aleatoriamente de una lista. Los autos activamente evitan colisiones y tratan de evitar rutas congestionadas.

El diseño de los agentes

Performance:

El objetivo del auto es llegar a un destino aleatorio de una lista de destinos. Esto se consigue calculando una ruta óptima por la ciudad y después siguen esta ruta calculada, siguiendo las reglas de la ciudad y evitando el tráfico en caso de ser necesario.

Environment:

El carro se enfrenta a las calles que determinan la dirección a la que se pueden mover los autos, los edificios funcionan como obstáculos para los carros, y los semáforos que en caso de estar en rojos detienen a los autos.

Actuators:

El carro no tiene el poder para cambiar el ambiente.

Sensors:

El auto puede percibir lo que tiene a su alrededor y tomar decisiones en base a ello. Cuando detecta un semáforo, el agente revisa si está en rojo o en verde para detenerse o continuar con su camino. Si detecta un auto enfrente, se detiene hasta que ya no se encuentre en su campo de visión. Además, puede detectar las direcciones que cada una de las calles tiene, modificando su movimiento.

Proactividad

La principal proactividad que podemos destacar es la generación de una ruta utilizando A* (A estrella). En el momento de generarse un auto, se le otorga un destino aleatorio e inmediatamente traza una ruta que seguirá hasta que algo lo detenga por su reactividad.

Reactividad

Si el auto se encuentra en una situación donde la ruta trazada ya no es viable debido a un embotellamiento, el auto reacciona y trazará una nueva ruta para evitar el tráfico. De la misma forma, si un auto detecta una luz roja, este le dará prioridad sobre el seguimiento de su ruta y se detendrá hasta que esté en verde.

Métricas de desempeño

Existen dos principales métricas que permiten determinar la efectividad de nuestros autos:

- **Cantidad de autos que llegaron a su destino:** Con esta métrica, podemos analizar si estamos evitando embotellamientos, así como determinar si nuestro algoritmo está funcionando de forma correcta. Entre mayor sea este valor, mejor será nuestra simulación
- **Cantidad de autos generados:** Esta métrica determina cuántos autos existen al mismo tiempo en la simulación. Entre mayor sea este número en cada step, nuestra implementación podría considerarse menos efectiva, ya que eso sugiere que los autos no están llegando rápidamente a su destino.

Restricciones

Los autos se rigen por reglas que limitan sus movimientos o acciones dependiendo del step. Entre estas restricciones podemos destacar los semáforos y otros autos que pueden frenar el movimiento de los autos y la dirección de las calles que pueden determinar hacia dónde se pueden dirigir

Arquitectura de subsunción

Menor prioridad

- Si no hay nada que que prive el movimiento → Moverse
- Si se encuentra un semáforo en verde → Moverse
- Si se encuentra un semáforo en rojo → Detenerse
- Si se encuentra con un auto → Detenerse
- Si se encuentra en una intersección → Ver la ruta y tomar un camino
- Si termina la ruta → Moverse hacia el Destino
- Si detecta la celda Destino a su lado → Moverse hacia la celda Destino
- Si se encuentra en la posición de Destino → Desaparecer
- Si no tiene ruta → Crear una ruta hacia su destino

Mayor prioridad

Características del ambiente

En este ambiente coexisten numerosos factores como los semáforos, edificios o la calle. Si bien en base de código estos componentes son considerados “agentes”, no cumplen con un propósito determinado, así que son considerados parte del ambiente de forma teórica.

El ambiente es **medianamente accesible**. Los autos son capaces de percibir lo que hay a su alrededor, alterando su comportamiento. No es totalmente accesible debido a que los autos

no guardan donde han estado, incluso si atraviesan por todo el mapa antes de llegar a su destino.

El ambiente es **no determinista**. Al moverse a, por ejemplo, una intersección, un agente puede tomar varias decisiones de movimiento dependiendo de si detecta tráfico o no. Esta múltiple cantidad de decisiones sugieren que no siempre se llegará a un resultado con cierta acción, volviéndolo no determinista

El ambiente es **no episódico**. El hecho de que el auto pueda actualizar su ruta si detecta algo que la afecte sugiere que conoce estados pasados y los sustituye por estados futuros posibles. Esto no elimina su capacidad episódica de tomar una decisión de acuerdo a lo que ve en ese turno.

El ambiente es **dinámico**. Autos podrán aparecer y desaparecer, moverse o cambiar de ruta, pero los semáforos siempre cambiarán constantemente de color sin importar qué hagan los autos.

El ambiente es **discreto**. Existe un número aleatorio y finito de destinos a los que un auto puede llegar. Además, existe la condición de que, si existen 4 autos en las esquinas cuando sea el momento de invocar autos nuevos, la simulación terminará.

WebGL

Para poder mostrar una simulación más completa, utilizamos WebGL, la cual es una librería que nos permite realizar animaciones 3D. Para esto, tuvimos que incluir nuevos conceptos como las matrices de transformación para poder rotar y trasladar los objetos a través del plano. Asimismo, implementamos el uso de luces y sombras y la inclusión de modelos 3D complejos de autos para que la simulación se viera mucho más realista.

Movilidad Urbana



APIs

Nuestra simulación es capaz de existir tanto en 2D usando mesa y en 3D usando WebGL. Para poder mostrarla en 3D, necesitamos un API que conecte ambos lados. Esta API es suministrada por la librería de Flask, la cual recibe los datos de la simulación y los envía a WebGL para su interpretación en su modelo 3D. Esta API utiliza los métodos de POST y GET para recibir los datos y enviarlos.

Conclusiones

Al realizar esta actividad, logramos determinar numerosas posibles soluciones a los problemas de tráfico. Si bien estas simulaciones suponen un entorno muy utópico, no podemos negar la utilidad de algoritmos de búsqueda para rutas óptimas. En la vida real, es probable que los sistemas de generación de rutas rápidas como Google Maps o Waze utilicen estos algoritmos para darnos la ruta más rápida hacia nuestros destinos.