

Reto: Movilidad Urbana



Tecnológico de Monterrey

Fecha de la Entrega: 29/11/2023

Alumnas

Fernanda Osorio A01026502

Lucia Barrenechea - A0178202

TC2038

Problemática

La movilidad urbana se define como:

“movimiento de las personas y bienes en las ciudades, independientemente del medio que utilicen para desplazarse, ya sea a pie, en transporte público, automóvil, bicicleta, etc” - CONUEE(2018)

Es decir, la movilidad urbana es como llegamos de un lugar a otro. Hoy en día las ciudades tienen que lidiar con un aumento exorbitante de automóviles en calles que no necesariamente fueron creadas para tales cantidades. En especial en una ciudad tan poblada como lo es la CDMX, sobra decir que una mala movilidad urbana es detrimental para nuestro desarrollo.

Los coches en CDMX son la manera en la que manejamos nuestra movilidad. Con ellos llegamos a los trabajos, a las escuelas, incluso a los doctores y hospitales. Entonces, ¿qué pasa si estos no pueden llegar a su destino? Nosotros tampoco podemos llegar al nuestro. Esto genera problemas económicos y de salud que, afortunadamente, pueden ser evitados con un correcto manejo y aplicación de la movilidad urbana que hay en nuestra ciudad y país.

Propuesta de Solución

Para esto nosotros proponemos una solución que utiliza agentes implementados en Python utilizando la librería Mesa. Un agente es un sistema de computadora que se sitúa en un ambiente y es capaz de realizar acciones autónomas dentro de ese ambiente para alcanzar sus objetivos. La librería *mesa* nos permite hacer justo eso, implementar un agente y un modelo que actuarán como coches y la ciudad que los contiene.

Diseño de Agentes

Esta propuesta de solución necesita diferentes agentes, no nada más existen los coches dentro de una ciudad. La ciudad está conformada por:

- coches
- caminos
- semáforos
- destinos

- obstáculos (Edificios)

Todos estos componentes, son agentes que tienen que existir dentro de la simulación de ciudad.

Los agentes deben de tener diferentes características.

- **Capacidad efectora:** los agentes de tipo coche tienen la capacidad de modificar su entorno al realizar una acción que se encuentre a su disposición, sin embargo estas solo se pueden hacer en ciertos escenarios que cumplan las condiciones. Por ejemplo, los agentes coche pueden modificar su entorno al moverse a una posición, sin embargo solo pueden realizar esta acción en caso de que un coche no se encuentre ya en esa posición.
- **Performance:** el agente es capaz de encontrar el mejor camino para ir hacia su destino, también es capaz de volver a encontrar este camino en caso de que el primero ya no esté disponible.
- **Environment:** el agente coche se enfrenta a un ambiente accesible, ya que conoce todos los caminos por los que se puede ir antes de seleccionar uno. Se enfrenta a un ambiente no determinista, ya que la selección de diferentes acciones puede modificar qué tan rápido llega el agente a su destino y por qué camino. Es un ambiente no episódico ya que el agente no necesita razonar sobre sus interacciones en futuros episodios. Es un ambiente dinámico, ya que a pesar de conocer todos los caminos, la efectividad de llegar al destino se ve comprometida por los agentes semáforos. Es un ambiente discreto ya que solo hay un número de acciones que el agente puede realizar.
- **Actuators:** el agente coche puede realizar las acciones de moverse, de evitar otro coche, hacer caso a un semáforo, recalcular el camino definido para su destino y seguir el camino definido
- **Sensors:** el coche checa sus alrededores para poder encontrar qué acciones puede realizar. Lo hace para checar en qué direcciones se puede mover y para saber qué es lo que lo rodea.
- **Proactividad:** los agentes coche pueden recalcular el camino que han decidido tomar hacia su destino en caso de que el primer camino ya no resulte factible o óptimo. Así es como toman iniciativa para cumplir su objetivo de llegar al destino seleccionado.
- **Reactividad:** los agentes coche pueden sensor cuando hay demasiados agentes coches a su alrededor, lo que indica que el camino que ha decidido recorrer puede tomarle más tiempo, así que en los lugares en los que es capaz de cambiar de camino (intersecciones y lugares de no

movimiento) el agente recalculará este camino. Así mismo los agentes pueden percibir cuando les es posible cambiarse de carril en la calle para poder acercarse más a su destino.

- **Habilidad social:** los agentes coche se relacionan entre ellos al no poder utilizar la posición que otro agente coche ya esté utilizando.
- **Métricas de desempeño:** el desempeño de un agente coche se medirá de acuerdo a si puede llegar o no a su destino.

Los agentes tienen diferentes características dependiendo de su funcionalidad en la simulación

Agentes Coche - Car(Agent):

- se inicializan con el modelo que se crea anteriormente, el grafo que contiene las posibles posiciones y destinos que puede tomar, el destino asignado y un estado de 0 o 1 que nos dice si todavía existe dentro de la simulación
- cada agente coche tiene su propio id que lo caracteriza

Agentes Semáforo - Traffic_Light(Agent):

- se inicializan con el modelo creado anteriormente, el tipo de tráfico que son, su estado que es un booleano y el tiempo por default que se tardan en cambiar.
- cada agente semáforo cuenta con su propio id que lo caracteriza

Agente Camino - Road(Agent):

- Este agente se inicializa con su id único, el modelo previamente creado y su dirección.
- La dirección de estos agentes es la que nos permitirá que el agente *Car* se mueva por toda la simulación siguiendo las reglas del sentido de la ciudad.

Los agentes: **Destino - Destination(Agent)**, **Obstáculo - Obstacle(Agent)**; reciben su id único y el modelo creado anteriormente.

El modelo que reciben todos los agentes se declara utilizando la librería *mesa*. Este modelo se encarga de inicializar los agentes(todos los tipos) en la simulación. También se encarga de “construir” la ciudad en base a un *txt* recibido. El grafo que se utiliza para construir los caminos posibles que un agente coche puede tomar también se genera en el modelo. Este grafo, es un grafo dirigido de acuerdo a los sentidos de las calles.

Arquitectura de Subsunción

trigger	condition	happening
El agente coche se mueve	El agente se encuentra en condición de destino	El agente desaparece.
El agente coche se mueve.	La posición futura es válida	El coche se mueve a la posición futura
El agente coche se mueve	Existe un agente de tipo semáforo en verde en posición futura	El coche se frena
El agente coche se mueve	Existe un agente de tipo semáforo en rojo en posición futura	El coche se frena
El agente coche se mueve	Existe un coche en su posición futura, en la posición de alado de la posición futura original no hay un coche y si existe un coche alado de la posición actual.	El coche no se mueve y espera.
El agente coche se mueve	Existe un coche en su posición futura, en la posición de alado de la posición futura original si hay un coche.	El coche no se mueve y espera.
El agente coche se mueve	Existe un coche en su posición futura, en la posición de alado de la posición futura original no hay un coche y no existe un coche alado de la posición actual.	El coche se mueve a la posición vecina de la posición original.
El agente coche se mueve	Existe un agente de tipo estrella derecha o izquierda ,no	Mantiene el mismo camino establecido.



	existe coche tanto 1 lugar adelante ni 2 lugares adelante.	
El agente coche se mueve	Existe un agente de tipo estrella arriba o abajo y no existe coche tanto 3 lugares adelante ni 4 lugares adelante.	Mantiene el mismo camino establecido.
El agente coche se mueve	Existe un agente de tipo estrella derecha o izquierda y existe coche tanto en 1 lugar adelante y 2 lugares adelante.	Se vuelve a recalcular el camino más cercano y se mueve a la nueva siguiente posición
El agente coche se mueve	Existe un agente de tipo estrella arriba o abajo y existe coche tanto 3 lugares adelante y 4 lugares adelante.	Se vuelve a recalcular el camino más cercano y se mueve a la nueva siguiente posición

Características del ambiente

El ambiente contiene 5 diferentes tipos de agentes: Car agent, traffic light agent, destination, obstacle y road. La ciudad está representada por un grid de 24x23. Todas las calles forman parte de un grafo dirigido donde las conexiones entre los nodos dependen de la dirección de las calles. Existen 16 destinos diferentes que son asignados de forma aleatoria a los agentes. Los agentes de coches son instanciados cada 4 pasos en las 4 esquinas de la ciudad. La simulación del ambiente se detiene cuando por cuarta vez un coche no puede ser instanciado ya que ya existe otro en el lugar de nacimiento.

Conclusiones

En conclusión, todos los elementos previamente mencionados hacen una propuesta de solución al mal manejo de la movilidad urbana en México. La simulación que se construye con estos elementos permite representar una mejor movilidad que sigue las reglas correctas como: los sentidos de las calles, respetar los semáforos y no permitir que un coche se encima a otro coche (choques).

Esta simulación nos demuestra que es posible lograr una mejor movilidad siempre y cuando sigamos las reglas establecidas y seamos capaces de tomar las mejores decisiones con respecto de dónde venimos y hacia dónde queremos llegar.

Aprendimos muchos conceptos a lo largo de realizar esta propuesta como lo son las arquitecturas de subsunción, los sistemas inteligentes, las características de un ambiente, entre muchas otras cosas.

Es gracias a actividades como esta que podemos observar los frutos de nuestra carrera aplicados en situaciones de la vida cotidiana. Nos permite observar la importancia de siempre ser éticos con nuestras acciones y decisiones. Además, nos fue posible apreciar la importancia de seguir las reglas y jerarquías al momento de realizar cualquier actividad, ya que eso nos llevó al mejor resultado que podemos obtener con nuestros conocimientos del momento.

Referencias:

CONUEE.(Abril,2018).*Movilidad Urbana Sostenible*.

<https://www.gob.mx/cms/uploads/attachment/file/313972/movilidadurbanasostenible.pdf>