



Tecnológico de Monterrey

M5. Revisión de avance 2

Modelación de sistemas multiagentes con gráficas computacionales

(Gpo 570)

Equipo 6:

Miguel Ponce Zertuche | A01383176

Maximiliano Flores Moreno | A00836019

Jorge Emiliano Bustamante Montenegro | A00834924

Maestros:

Raul Valente Ramirez Velarde

Ivan Axel Dounce Nava

Fecha de entrega:

12 de Marzo del 2024

Situación problema

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.

Este reto te permitirá contribuir a la solución del problema de movilidad urbana en México, mediante un enfoque que reduzca la congestión vehicular al simular de manera gráfica el tráfico, representando la salida de un sistema multi agentes.

Reto a desarrollar

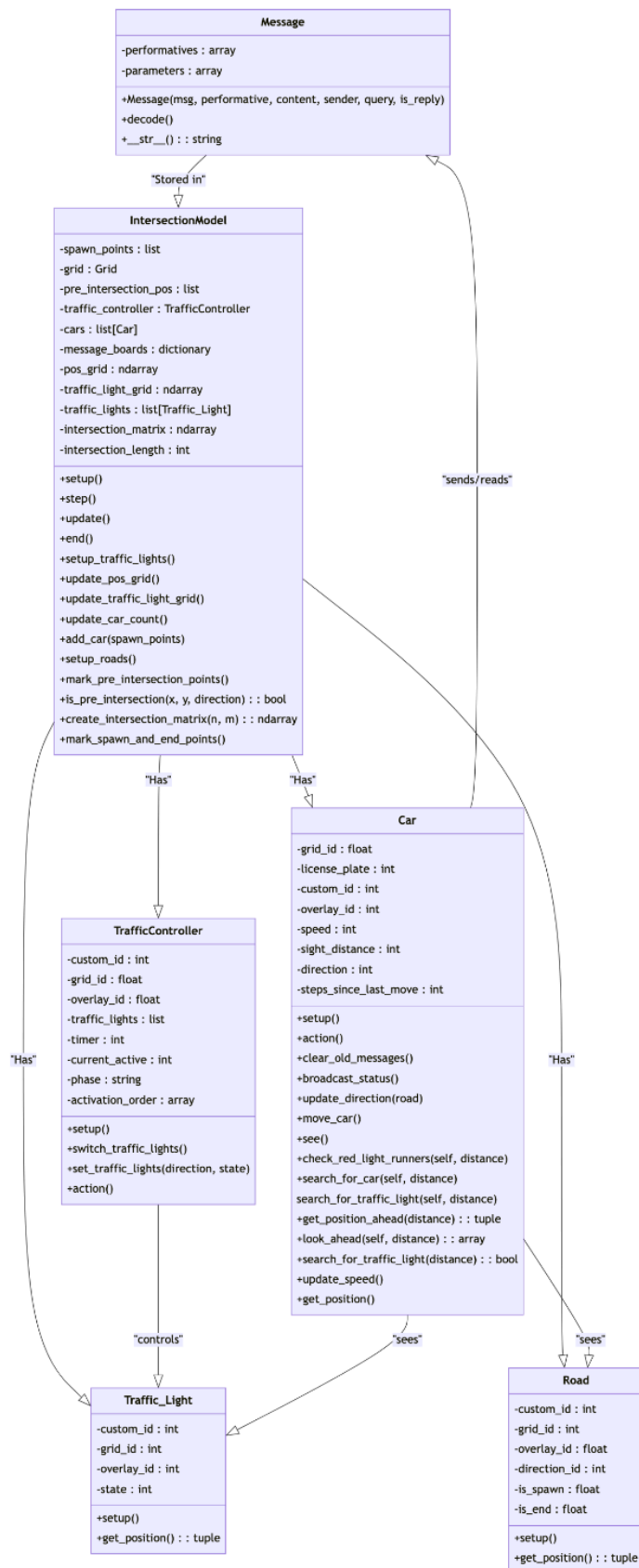
Tráfico vehicular en la ciudad:

Es importante para las personas que residen o trabajan en una ciudad llegar a su destino de manera eficiente y cómoda. Por eso, en una ciudad grande, la cantidad de vehículos transportándose y encontrándose en intersecciones (algunas de éstas con semáforos) lleva a un problema de alto tráfico, llevando a las personas a frustrarse por los largos tiempos de trayecto, impidiendo incluso llegar puntualmente a sus respectivas citas.

Identificación de agentes

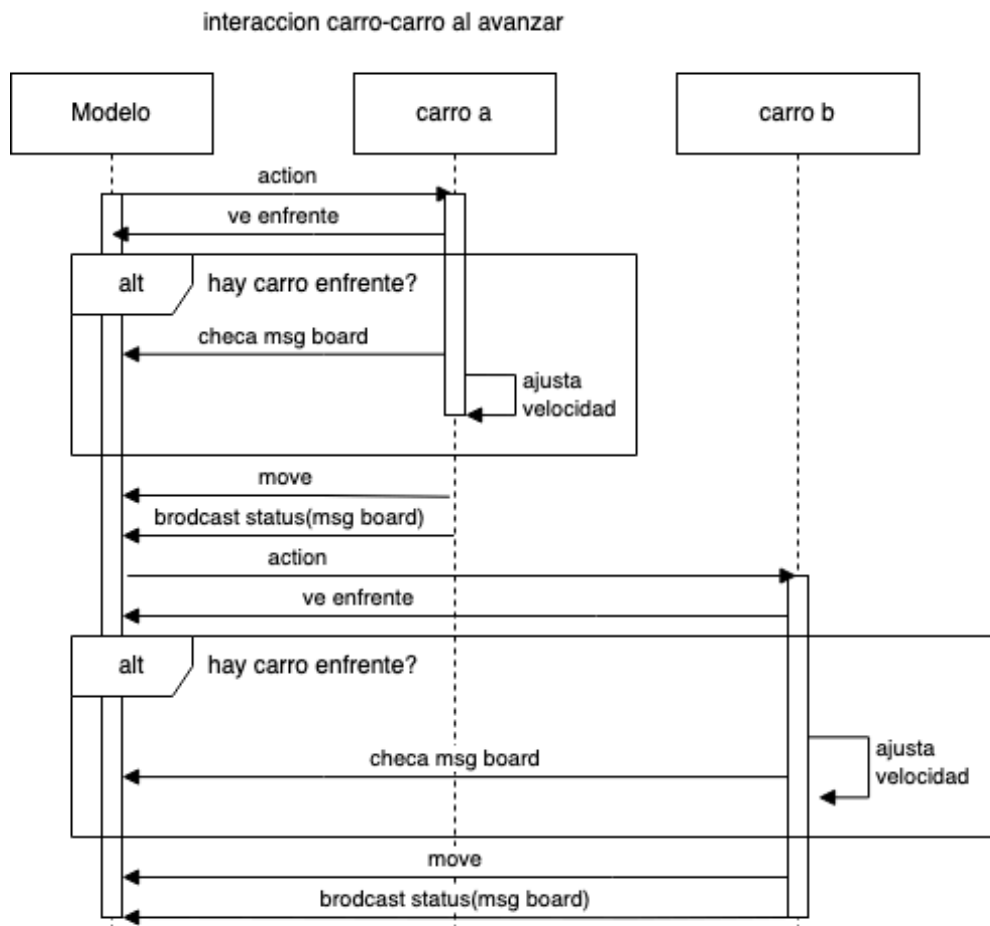
- **Carros:** Agente principal de modelo sus acciones y estado están determinados en base a los carros a su alrededor y su ambiente
- **Semáforos:** Agente sin acciones cuya finalidad es que determine las acciones de los carros
- **Controlador de Semaforos:** Controla los estados de los semaforos
- **Calle:** Agente cuya sola función es definir la dirección de movimiento

Diagrama de clase

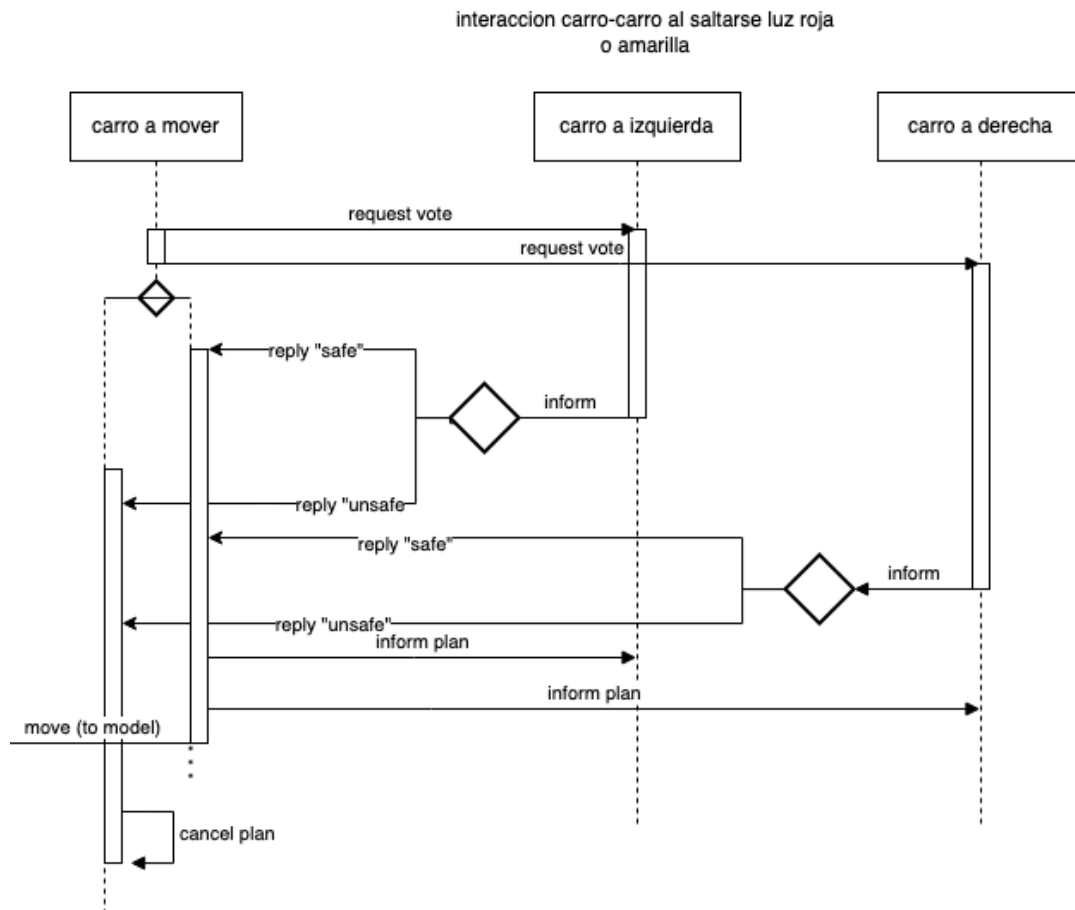


Diagramas de protocolos de interacción

- interaccion al avanzar



-
- interaccion al avanzar en luz amarilla o roja



Tipo de Razonamiento:

En nuestra implementación de un modelo para una intersección, la forma de razonamiento es **deductiva**, ya que todos los agentes actúan dependiendo de un set de reglas a diferencia de los goles. El agente principal y el cual actúa dependiendo a su ambiente y hace decisiones basado en este, es el carro. Este cada paso llama su función `action()`, la cual entiende y analiza su entorno, como la dirección de la calle en la que está, su velocidad, y agentes en un rango de distancia definido frente a este para definir qué hará, con acciones como reducir o incrementar su velocidad, moverse, e incluso una posibilidad de romper las reglas al no respetar la luz roja o amarilla. Pero dentro de todo esto sigue estando controlado por reglas que determinan su próxima acción.

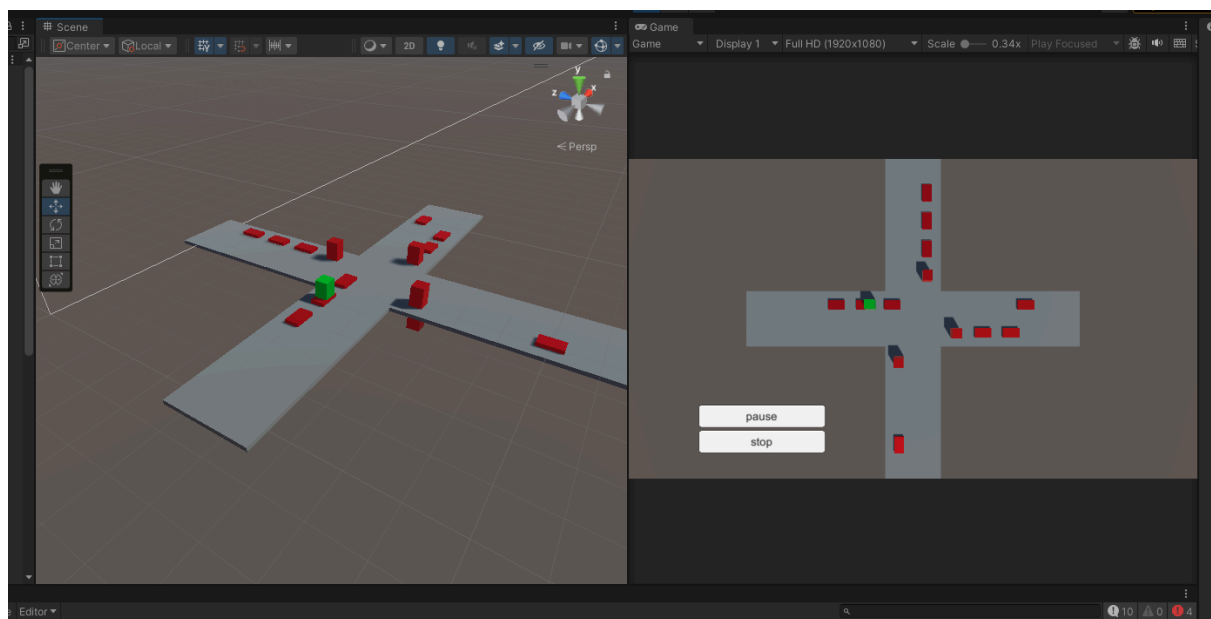
Protocolos de interacción:

Al ingresar a una intersección todo conductor tiene que estar consciente de su ambiente, incluyendo no solo semáforos sino otros conductores de otras direcciones, ya que en el mundo real siempre hay posibilidad de accidentes, ya sea a causa de distracción, ignorancia,

u otras causas. Es por esto que los agentes están equipados con la capacidad de Analizar el estado de otros agentes haciendo requests de información sobre sus intenciones y características como velocidad. Esto aplica ambos para conductores que simplemente siguen la ley y manejan de manera segura, y conductores que checan sus alrededores para ver si pueden hacer movimientos riesgosos, como avanzar en luz amarilla o incluso luz roja. Para esto se puede implementar un sistema de **votación**, donde los conductores relevantes votan si una acción es segura y factible sin llevar a una colisión, y estos votos deciden si la acción se lleva a cabo.

Progreso Unity

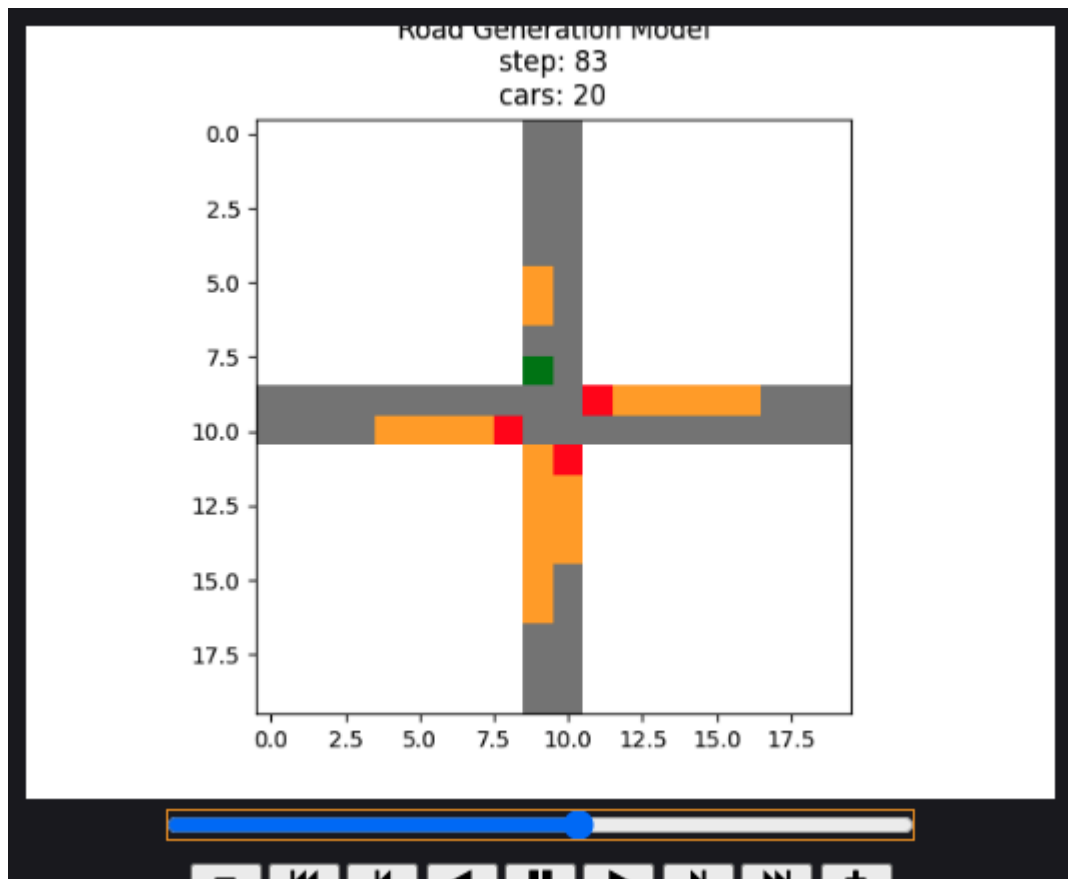
- UI para modificar parametros
- conexion completa a Python
- visualizacion funcional de modelo
- movimientos fluidos
- FALTANTE:
- mejoras esteticas (assets)
- decoraciones dinamicas
- estadisticas en vivo





Progreso Python

- Modelo completo
- Comunicacion entre agentes implementada
- exportacion de datos
- FALTANTE:
- refinacion de comunicacion



```
127.0.0.1 - - [12/Mar/2024 18:36:00] "GET /sum?x=5&y=10 HTTP/1.1" 200 -
Completed: 302 steps
Run time: 0:00:01.405727
Simulation finished
127.0.0.1 - - [12/Mar/2024 18:36:08] "POST /run_model HTTP/1.1" 200 -
127.0.0.1 - - [12/Mar/2024 18:36:09] "GET /intersection_matrix HTTP/1.1" 200 -
127.0.0.1 - - [12/Mar/2024 18:36:10] "GET /frames HTTP/1.1" 200 -
127.0.0.1 - - [12/Mar/2024 18:36:18] "GET /frames HTTP/1.1" 200 -
```

Link de las tareas divididas para el

equipo: <https://tasks.office.com/tecmx.onmicrosoft.com/Home/PlanViews/hHeXCMXGqUuJIGApd7kx1mQAGwCW?Type=PlanLink&Channel=Link&CreatedTime=638448156811620000>

Link para repositorio: <https://github.com/jorgebustmontec/reto-TC2008b-agentpy>