

|Instituto Tecnológico y de Estudios Superiores de Monterrey

Campus Puebla



**Modelación de Sistemas Multiagentes
con Gráficas Computacionales**

TC2008B.402

Evidencia 2

Profesor: Dr. Luciano García Bañuelos

Estudiantes:

Daniel Francisco Acosta Vázquez / A01736279

Diego García de los Salmones Ajuria / A01736106

Oskar Adolfo Villa López / A01275287

Raúl Díaz Romero / A01735839

Periodo Agosto - Diciembre 2023

5 / Octubre / 2023

Revisión 3

I. Github: https://github.com/RaulDiazR/MultiAgent_Intersection

II. Avances de Código

- A. Implementación de los agentes:** Se ha terminado prácticamente por completo la implementación de los agentes, pues el modelo actúa correctamente, generando carros inteligentes, los cuales actúan conforme a su entorno. Así mismo, los semáforos y las calles han sido implementados correctamente, por lo que cada agente ha sido implementado efectivamente en el modelo. Estos avances se pueden encontrar en la rama 'turn' del repositorio del proyecto.
- B. Implementación de la parte gráfica de la solución:** Se han recopilado los modelos necesarios para modelar las calles, edificios, semáforos, carros y demás objetos del sistema, además, se ha implementado un apartado lógico en OpenGL al implementar una matriz de control, la cual indican las acciones que deben de tomar los carros a través del sistema. En cuanto a la parte gráfica, falta principalmente realizar la conexión con Mesa a través del Rest Api, fuera de ello, se han llevado a cabo la mayoría de actividades. Estos avances se pueden encontrar en la rama 'OpenGL' del repositorio del proyecto.

III. Descripción del reto a desarrollar.

El proyecto consiste en la creación de un sistema multiagente que emule el comportamiento de múltiples carros al atravesar una intersección con semáforos. Este proyecto implica el desarrollo de algoritmos avanzados para coordinar la interacción entre los carros, ajustando sus velocidades con el propósito de evitar colisiones. Además, se requiere una gestión eficiente y efectiva de varios agentes, ya que se contempla la incorporación de varios carros que buscarán cruzar la intersección simultáneamente.

El proyecto engloba la creación del entorno donde interactúan los carros y semáforos, el apartado lógico. Para llevar a cabo tanto el desarrollo del sistema como la implementación de los carros y semáforos, se planea utilizar la librería 'Mesa' a través del lenguaje de programación 'Python', dicha librería facilita la creación, manipulación e interacción de los agentes del modelo.

Además, el apartado gráfico del proyecto se llevará a cabo utilizando la herramienta

OpenGL, en la cual se modelan los objetos utilizados en el proyecto, tales como: la ciudad, los carros, los semáforos y las calles.

Por último, se planea conectar el apartado lógico (Mesa) con el apartado gráfico (OpenGL) a través del uso de una arquitectura cliente-servidor, dada por el Rest Api. A través de dicha arquitectura, se enviarán datos desde Mesa hacia OpenGL en formato Json, estos datos definirán los estados y acciones de los agentes en el modelo en cada momento de la simulación.

IV. Descripción de Modelo y Agentes involucrados

A. Agentes y Modelo a utilizar:

1. **Car (Agente):** Este agente representa un carro en el sistema. Su principal función es simular el comportamiento de un carro en movimiento. Los carros deben ajustar sus velocidades y tomar decisiones para evitar colisiones con otros carros y respetar las señales de tráfico.
2. **TrafficLight (Agente):** Este agente representa un semáforo en la intersección. Su función es controlar el flujo de tráfico regulando cuándo los carros pueden avanzar o detenerse. Los carros deben responder a las señales de los semáforos.
3. **Street (Agente):** Este agente representa las calles de la ciudad. Las calles conectan diferentes partes de la ciudad. Los carros se desplazarán por estas calles.
4. **City (Modelo):** El modelo de la ciudad representa el entorno general donde interactúan los carros, los semáforos y las calles. Los carros se mueven dentro de esta ciudad, respetando las reglas de tráfico y respondiendo a los semáforos. Además, el modelo posee una matriz, la cual representa las diferentes acciones y limitaciones que tienen los carros a través de la ciudad, tales como: sentido de dirección, opciones de giro, posición de las calles y edificios, etc.

B. Relación entre agentes:

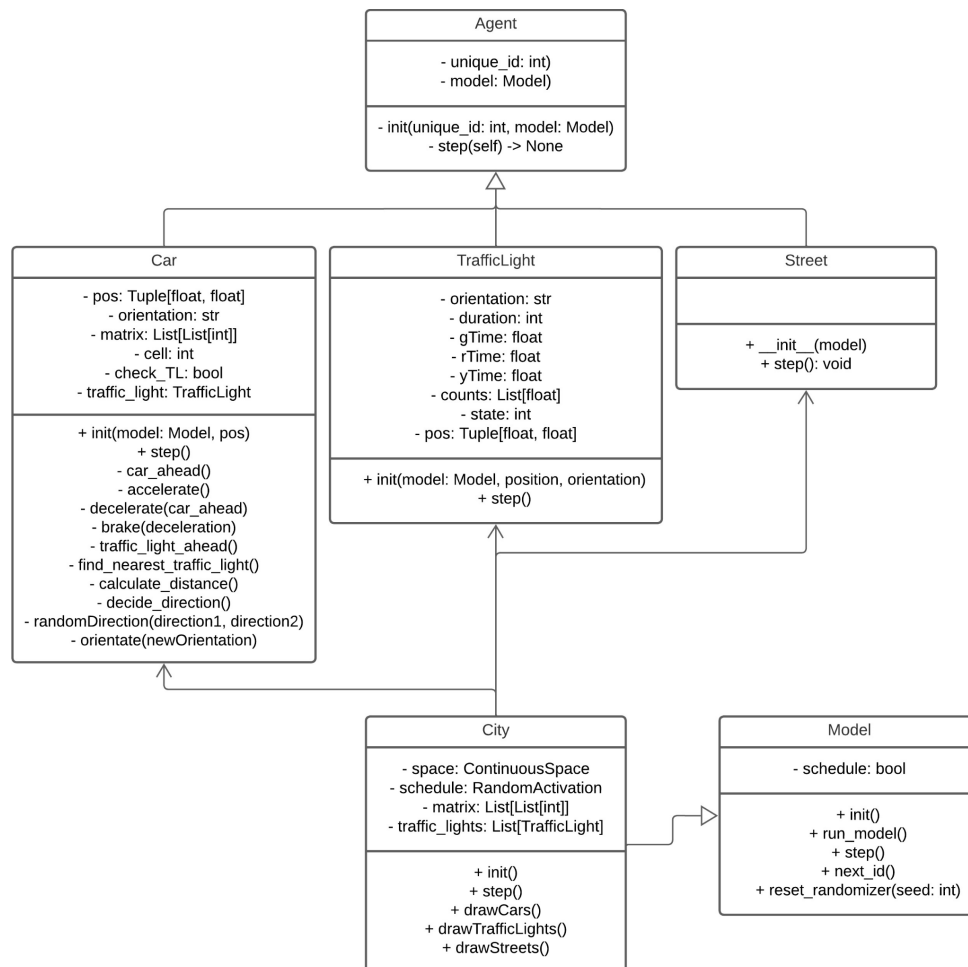
En este proyecto, los carros, semáforos y calles son agentes interdependientes que colaboran para simular el flujo de tráfico en una intersección. Los carros representan vehículos en movimiento y están sujetos a las señales de tráfico controladas por los semáforos. Los semáforos, por su parte, desempeñan un papel crucial al regular el tráfico en la intersección, alternando entre los estados de luz verde, amarilla y roja. Los carros deben interpretar estos estados y actuar en consecuencia, deteniéndose ante la luz roja y avanzando con precaución cuando la luz está en verde. Además, las

calles proporcionan el escenario para que los carros se desplacen y pueden tener características específicas, como direcciones permitidas. El modelo de ciudad, como entidad central, coordina y supervisa estas interacciones entre carros, semáforos y calles, asegurando que el tráfico fluya sin problemas y se eviten colisiones. En resumen, esta relación colaborativa entre agentes y el modelo de ciudad es esencial para crear una simulación realista y segura de carros cruzando una intersección con semáforos.

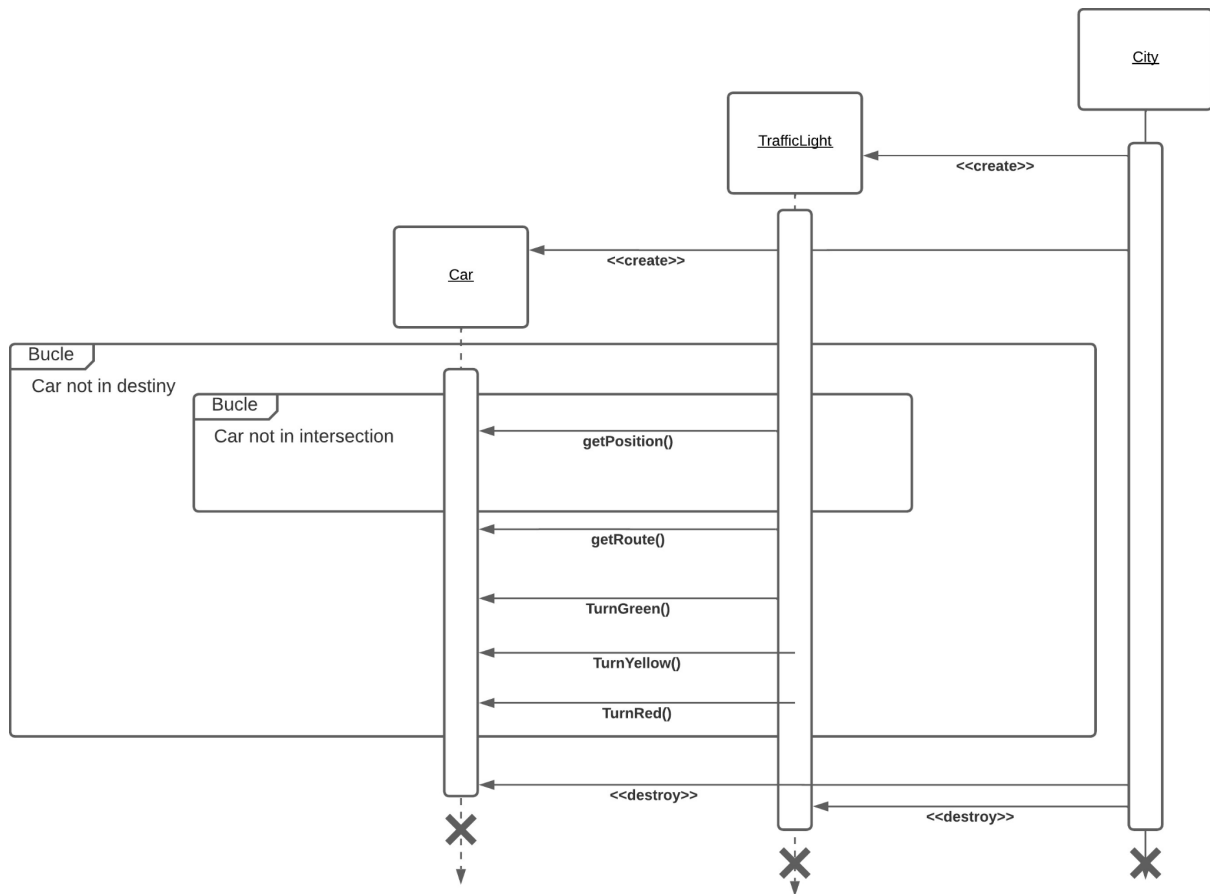
C. Diagrama de clase presentando los distintos agentes involucrados.

Link de LucidChart:

https://lucid.app/lucidchart/c94515f0-5335-46ee-aa3e-4d3b04cd8552/edit?view_items=rMypwTKdzM2C&invitationId=inv_0e690c61-fa72-413c-9408-77d80e37e626



D. Diagrama de Secuencia de protocolos de interacción.



V. Plan de trabajo

A. Actividades pendientes:

- Conectar el modelo con el ambiente. Semana 4
- Realizar la presentación del proyecto. Semana 4

B. Actividades planeadas para la tercera revisión (2 de octubre - 6 de octubre):

- Actualización del diagrama de secuencia. Responsable: Oskar. Intervalo de esfuerzo estimado: medio.
- Actualización del Diagrama de clases. Responsable: Raúl. Intervalo de esfuerzo estimado: medio.
- Construir el ambiente gráfico en OpenGL. Responsables: Diego y Daniel. Intervalo de esfuerzo estimado: medio.
- Actualizar Secciones de texto del documento. Responsable: Todo el equipo. Intervalo de esfuerzo estimado: Bajo.

C. Actividades completadas:

- Implementar las clases para cada agente en MESA (Car, Street y TrafficLight).
Responsable: Raúl. Intervalo de esfuerzo estimado: medio.
- Completar el modelo y el comportamiento correcto de los agentes en Mesa.
Responsables: Raúl y Oskar. Intervalo de esfuerzo estimado: medio.
- Obtener los modelos de los objetos utilizados en el sistema para OpenGL.
Responsables: Diego y Daniel. Intervalo de esfuerzo estimado: bajo.

VI. Aprendizaje adquirido.

- A. Programación y Algoritmos Avanzados:** En este proyecto hemos necesitado implementar algoritmos para coordinar el movimiento de múltiples agentes de manera eficiente, lo que incluye la optimización de rutas, la detección de colisiones, la sincronización de semáforos virtuales, etc, al igual que utilizar conceptos de programación para coordinar la lógica entre agentes y entre el apartado gráfico de la simulación.
- B. Simulación y Modelado:** En este proyecto hemos modelado el comportamiento de los carros, su interacción con la intersección y otros carros, al igual que un modelado de una vialidad simulando una ciudad.
- C. Coordinación y Colaboración:** Hemos aprendido sobre la importancia de la coordinación y colaboración entre agentes para evitar colisiones y maximizar la eficiencia en la intersección.
- D. Diseño de Sistemas Complejos:** En este proyecto hemos tenido que considerar cómo los diferentes componentes interactúan entre sí, cómo gestionar la comunicación y la toma de decisiones entre los agentes, y cómo mantener la estabilidad del sistema.
- E. Programación Orientada a Objetos:** Al modelar carros y la intersección como agentes, hemos trabajado con conceptos de programación orientada a objetos, como el diseño de clases, propiedades y métodos, y la lógica de los agentes en objetos.
- F. Optimización y Eficiencia:** Al trabajar con varios agentes al mismo tiempo hemos requerido trabajar en la optimización y la eficiencia de nuestra simulación con el fin de obtener una simulación correcta. De igual manera en el apartado gráfico hemos aprendido cuando es apto utilizar modelos de alta o baja cantidad de polígonos al igual que cuando conviene mejor utilizar texturas.

- G. Desarrollo de agentes inteligentes:** Hemos aprendido cómo crear agentes inteligentes e implementar algoritmos que permitan que estos puedan interactuar de manera efectiva en un entorno complejo como una intersección de tráfico.
- H. Modelado de objetos 3D:** La implementación de modelos para representar calles, edificios, semáforos y carros en OpenGL nos ha permitido aprender sobre modelado 3D y cómo importar y representar objetos tridimensionales de manera efectiva en una simulación.
- I. Importado de objetos 3D:** Al utilizar modelos pre creados y agregarlos a nuestra simulación, hemos aprendido a importar y trabajar con modelos y material externo y agregarlo al proyecto.
- J. Conexión entre lógica y gráficos:** Hemos aprendido acerca de la importancia de la comunicación entre la lógica y la parte gráfica de tu proyecto. Esto es fundamental para lograr una representación visual precisa de la simulación que refleje el estado actual de los agentes y el entorno.
- K. Escalamiento y Adaptabilidad:** Hemos aprendido a escalar y adaptar los datos obtenidos por la parte lógica del sistema y trasladarlos a la parte grafica del mismo manteniendo su integridad y asegurando una correcta representación de la simulación.
- L. Arreglo de Errores:** Hemos aprendido y reforzado nuestra habilidad de encontrar y arreglar errores cuando hay una conexión entre diferentes sistemas, al tener que reconocer si un error es lógico o gráfico y saber resolverlo de la manera adecuada.
- M. Trabajo Organizado:** Hemos aprendido a realizar el proyecto siguiendo el plan de trabajo al mismo tiempo que a organizar las tareas y juntar tanto el apartado gráfico como el lógico para poder ver una armonía entre ambos sistemas y tener una simulación eficaz.