



Tecnológico de Monterrey

Instituto Tecnológico de estudios superiores de Monterrey

Campus Estado de México

Departamento de Ingeniería

TC2008B

Modelación de sistemas multiagentes con gráficas computacionales

Grupo: 302

Profesor: Jorge Adolfo Ramírez Uresti

Revisión 3

“Modelación Agentes”

Fecha: 21/11/2024

Alumnos:

Santiago Villazón Ponce de León

A01746396

Juan Antonio Figueroa Rodríguez

A01369043

Iván Alexander Ramos Ramírez
A01750817
Sebastián Antonio Almanza
A01749694

Índice

Modelación de sistemas multiagentes con gráficas computacionales.....	1
I. IMPLEMENTACIÓN DEL SMA.....	3
II. IMPLEMENTACIÓN GRÁFICA.....	3
III. DEFINICIÓN DEL MEDIO AMBIENTE.....	3
III.I. Topología y Configuración de la Intersección.....	4
III.II. Elementos del Ambiente Físico.....	4
III.III. Componentes Inteligentes y Tecnológicos.....	4
III.IV. Elementos Dinámicos y Reglas de Comportamiento en el Tráfico.....	4
III.V. Objetivo del Ambiente y Condiciones de Operación.....	5
Mayormente inaccesible.....	6
Mayormente determinista.....	6
No episódico.....	7
Dinámico.....	7
Discreto.....	7
Explicación detallada de los agentes.....	8
Agente "Carro".....	8
Agentes "Semáforo".....	8
Tipos de Semáforos.....	8
Funcionalidad Proyectada.....	8
V. DIAGRAMAS DE AGENTES USANDO AUML Y DIAGRAMA SMA.....	10
VI. DIAGRAMA DE INTERACCIÓN.....	11

I. IMPLEMENTACIÓN DEL SMA

Para la implementación del SMA, se utilizó Python en conjunto con la librería de agentes Mesa, lo que facilitó la definición modular de los agentes y su interacción dentro del modelo. La simulación, gestionada mediante el framework MESA, emplea activación aleatoria de agentes y una cuadrícula que representa la estructura de una ciudad.

El modelo cityClass simula un entorno urbano donde los agentes, representados como vehículos, se desplazan entre garajes y destinos asignados aleatoriamente. La cuadrícula incluye celdas restringidas que simulan obstáculos, como edificios, y direcciones permitidas que guían los trayectos de los agentes. Además, semáforos sincronizados regulan el flujo vehicular en cruces estratégicos, promoviendo un comportamiento ordenado y dinámico.

La clase CarAgent define a los vehículos, los cuales respetan las restricciones del entorno y las señales de tráfico. Su comportamiento se basa en el algoritmo A* para calcular rutas óptimas, garantizando una interacción fluida con el entorno y otros agentes. Este diseño permite analizar dinámicas de tráfico y optimizar rutas en escenarios urbanos complejos.

El enfoque modular del modelo asegura una representación eficiente de la red urbana, donde cada componente interactúa de forma controlada, proporcionando una herramienta robusta para el análisis del tráfico en ciudades simuladas.

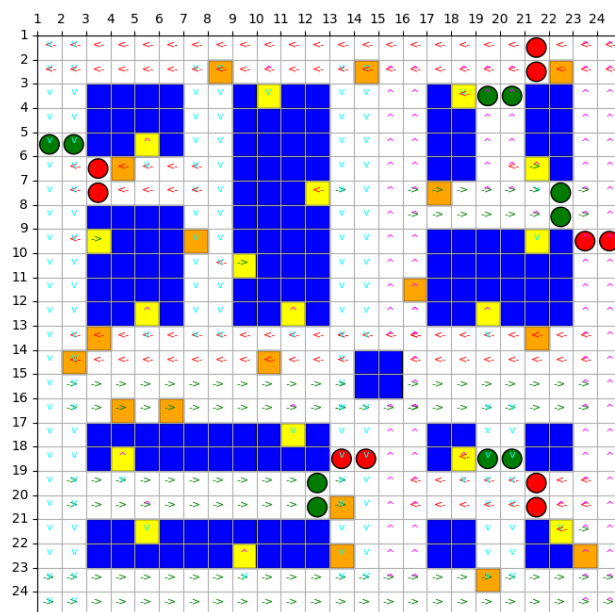


Imagen 1. Simulación visualizada en matplotlib

Considerando lo comentado, el avance actual representa aproximadamente el 65% de la implementación final del proyecto, esto ya que ya se ha logrado implementar de forma correcta el comportamiento de los agentes principales (vehículo y semáforo) dentro de la simulación de matplotlib y su correcto funcionamiento, como lo es el que los vehículos respeten los semáforos, no colisionen entre sí los vehículos, las direcciones de las calles, y que estos puedan ir de un estacionamiento al otro apoyándose del algoritmo de A* para calcular los caminos y distancias para llegar de un punto a otro de forma eficiente.

De igual forma quedan como pendientes la implementación completa del agente del peatón así como establecer una conexión total entre la simulación ya construida con la visualización en Unity.

II. IMPLEMENTACIÓN GRÁFICA

Para la representación gráfica de la ciudad proporcionada (imagen 1) se utilizó el motor Unity para crear la visualización en 3D.

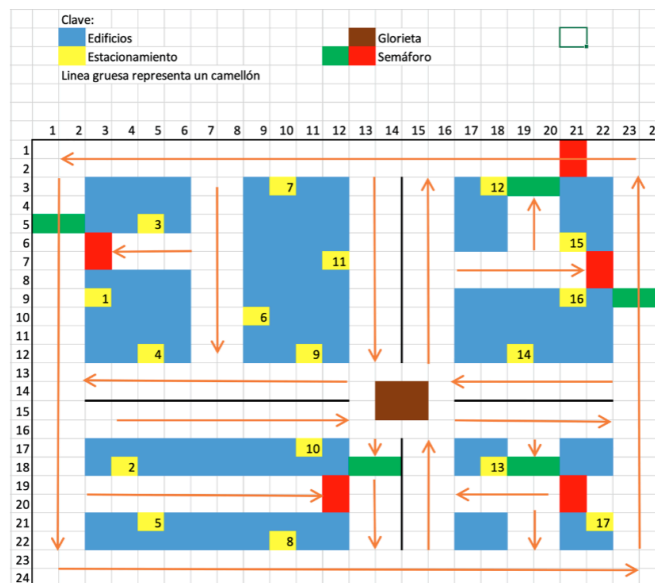


Imagen 2. Distribución de la ciudad

En Unity la ciudad quedó representada de la siguiente manera siguiendo el mapa dado para la representación de la ciudad:

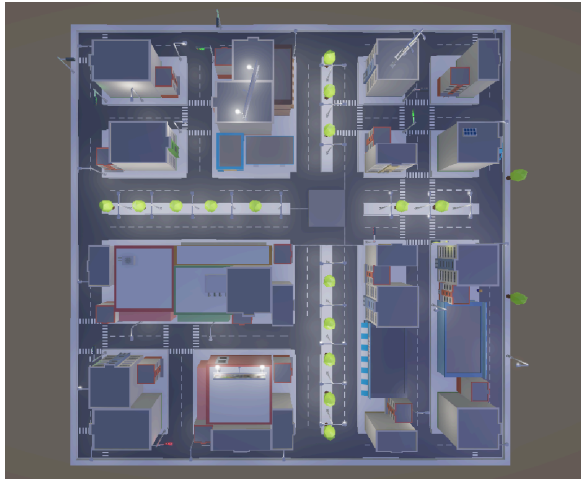


Imagen 3. Vista Cenital de la ciudad

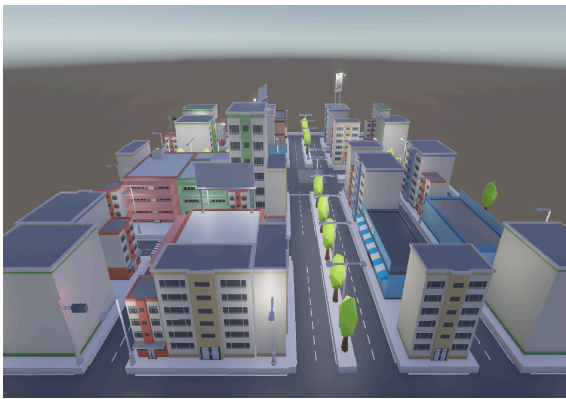


Imagen 4. Vista al horizonte de la ciudad



Imagen 5. Vista de los edificios



Imagen 6. Vista de auto y peatón a escala de la ciudad

Para la representación gráfica de la ciudad se utilizaron:

- Modelos y assets del paquete LowPoly proveniente de la Unity Asset Store
- Modelos y assets creados por el equipo utilizando Blender

Consideramos que el avance entregado corresponde a un 70% en el apartado gráfico ya que esta ya incluye la distribución pedida para el mapa de la ciudad y cuenta con sus edificios en su totalidad siguiendo el mapa de la distribución, así como las calles alrededor de todo el mapa y con los semáforos en cada una de las intersecciones indicadas. De igual forma ya se han incorporado elementos decorativos como lo son los árboles, elementos de iluminación (luz del área, luces en las luminarias y en los espectaculares), y los vehículos diseñados y el peatón. Como trabajo futuro se tiene contemplado la conexión con el modelo de mesa para la funcionalidad completa de la visualización.

III. DEFINICIÓN DEL MEDIO AMBIENTE

El ambiente del sistema es una intersección urbana controlada, situada en un entorno donde se encuentran múltiples carriles de circulación provenientes de cuatro direcciones cardinales. Este entorno incluye vehículos (automóviles) y semáforos inteligentes que interactúan entre sí para optimizar el flujo de tráfico y reducir el tiempo de espera en la intersección.

Conforme a los requerimientos de la situación problema, el ambiente que se tomó en cuenta para desarrollar simula el tránsito de vehículos en una ciudad creada de manera ficticia en México. Los principales atributos de este pequeño modelo son una calle principal, la cual rodea todo el perímetro de la misma ciudad, al igual que una rotonda casi al centro de todas las calles lo cual permite que el flujo sea más eficiente. A continuación se presentan las características de la ciudad.

- 1 cuadrícula inicial de 24 x 24
- 12 edificios (124 celdas)
- 17 destinos / garajes (celdas de inicio / destino)
- 1 rotonda
- 10 semáforos (20 celdas)

A continuación, se presentan las características detalladas de este ambiente:

III.I. Topología y Configuración de la Intersección

Intersección de Cuatro Vías: La simulación representa una intersección donde se cruzan dos avenidas de doble sentido, generando cuatro vías de acceso y salida. Cada avenida tiene dos carriles de entrada y dos de salida en cada dirección, permitiendo el tráfico en ambas direcciones en todas las vías.

Líneas de Visibilidad y Frenado: En cada entrada de la intersección, existen líneas de visibilidad marcadas, que son zonas donde los vehículos pueden observar la luz del semáforo con anticipación y comenzar a frenar en caso de necesitar detenerse. Estas líneas son también puntos clave en los que los semáforos pueden detectar la presencia de vehículos que se aproximan.

III.II. Elementos del Ambiente Físico

Carriles y Señalización Horizontal: La intersección está equipada con señalización horizontal que incluye líneas de frenado y flechas de dirección. Los carriles están delimitados para el tránsito vehicular.

Zonas de Detección y Control: La intersección cuenta con sensores en el pavimento y cámaras de tráfico en puntos estratégicos que permiten a los semáforos detectar la llegada de vehículos y estimar el flujo vehicular.

III.III. Componentes Inteligentes y Tecnológicos

Semáforos Inteligentes: Cada semáforo está equipado con un sistema de control inteligente que permite la detección de vehículos mediante sensores y el ajuste de la programación de luces de acuerdo con la demanda en cada dirección.

- *Luz Amarilla Predeterminada:* En ausencia de tráfico, los semáforos permanecen en luz amarilla, lo que indica precaución. Este modo ahorra energía y asegura que el semáforo esté en un estado de espera hasta que se detecte la llegada de un vehículo.
- *Cambio a Luz Verde:* Cuando se detecta la llegada de un vehículo, el semáforo cambia a verde en la dirección de arribo, facilitando el flujo del tráfico y minimizando tiempos de espera.
- *Coordinación de Luces:* Cuando más de un vehículo se aproxima desde diferentes direcciones, el sistema coordina un ciclo de luces entre los semáforos de la intersección, de modo que cada vía tenga una oportunidad de pasar en un orden optimizado para reducir la acumulación de tráfico.

III.IV. Elementos Dinámicos y Reglas de Comportamiento en el Tráfico

Flujo de Vehículos: El ambiente está diseñado para simular un flujo de tráfico realista, incluyendo variaciones en la densidad de vehículos a lo largo del tiempo, dependiendo de factores como horas pico, emergencias simuladas, y patrones de conducción. Los vehículos tienen comportamientos basados en reglas predeterminadas, como detenerse en rojo, avanzar en verde, y disminuir la velocidad al acercarse a la luz amarilla.

Comunicación entre Vehículos y Semáforos: Cada vehículo que se aproxima envía un mensaje al semáforo con su tiempo estimado de llegada a la intersección. Este mensaje se utiliza para calcular el tiempo óptimo de cambio de luces y

maximizar la eficiencia en la intersección. Los vehículos también detectan el cambio en el semáforo para ajustar su velocidad y garantizar un flujo continuo.

III.V. Objetivo del Ambiente y Condiciones de Operación

Reducción del Tiempo de Espera: El principal objetivo del ambiente es reducir el tiempo de espera para cada vehículo en la intersección. El sistema de control de semáforos inteligentes ajusta dinámicamente los tiempos de luz verde para reducir la congestión y evitar la formación de largas filas de vehículos.

Optimización del Flujo de Tráfico: A través de la detección y comunicación entre los vehículos y los semáforos, el ambiente se enfoca en mantener un flujo continuo de tráfico, reduciendo las interrupciones y optimizando el uso de la infraestructura vial.

Seguridad Vial: Al incluir zonas de frenado, un sistema de comunicación coordinado entre los agentes (vehículos y semáforos), el ambiente está diseñado para minimizar el riesgo de accidentes en la intersección. La prioridad en la luz verde se da a los vehículos en movimiento, mientras que la luz amarilla en espera permite una transición segura cuando no hay tráfico.

En resumen, el ambiente para esta simulación multiagente representa una intersección altamente optimizada para el flujo de tráfico en una ciudad moderna, con una infraestructura de semáforos inteligentes que detecta, comunica, y responde a las condiciones de tráfico en tiempo real. Los sensores, las cámaras y los semáforos inteligentes trabajan juntos para mejorar la eficiencia de la circulación y garantizar la seguridad de los conductores.

Nuestro ambiente está diseñado específicamente para fomentar la interacción entre los agentes "Carro" y "Semáforo". Esto se debe a que la lógica principal que buscamos simular y representar es el comportamiento de los embotellamientos o el tráfico vehicular intenso que ocurre de forma recurrente entre estos agentes. Además, estos elementos nos permiten crear un sistema capaz de implementar y analizar dichas dinámicas de manera efectiva.

Aunado a lo anterior es que, además de los dos agentes mencionados con anterioridad, hay diversos agentes de MESA los cuales no tienen la funcionalidad como tal de un agente gracias a su carencia de autonomía entre otros aspectos.

- Es posible añadir o generar hasta un máximo de 17 agentes "Carro" antes de iniciar la simulación.

Esto se debe a que el diseño del ambiente que hemos implementado solo permite la creación de agentes "Carro" dentro de las áreas designadas como estacionamientos o destinos. Este enfoque busca representar de manera más

realista los escenarios en los que los vehículos salen de un estacionamiento en condiciones de tráfico pesado.

Los agentes "Carro" pueden ser añadidos a la cuadrícula siempre que haya al menos un espacio de estacionamiento disponible para completar esta acción. Sin embargo, esta funcionalidad podría modificarse en el futuro con base en la retroalimentación recibida para esta entrega. De igual manera, la cantidad de vehículos que se pueden agregar en intervalos de n tiempo podría ajustarse si fuera necesario.

Aunque más adelante se detallan las características específicas de ambos tipos de agentes, es relevante mencionar que su interacción es completamente mutua y no está influenciada por variables externas. En particular, las intersecciones controladas por semáforos inteligentes juegan un papel clave en el sistema.

A diferencia de otras celdas de tránsito, las cuatro celdas correspondientes a las intersecciones con semáforos facilitan que los agentes "Carro" tomen decisiones sobre el camino más adecuado para alcanzar su destino de la manera más eficiente posible. Por otro lado, la lógica y funcionalidad de los semáforos se describe detalladamente más adelante en este documento. A continuación, se exponen las características principales del ambiente.

Mayormente inaccesible

El ambiente tiene una accesibilidad limitada, ya que los agentes no pueden percibir el estado completo del entorno. Esto requiere que cada agente mantenga un estado interno para decidir sus acciones. En nuestra simulación, esta restricción se debe a las limitaciones de los sensores de los agentes principales. El agente "Carro", por ejemplo, solo puede observar las celdas vecinas, lo que deja ocultos otros elementos del entorno que podrían influir en su toma de decisiones. Aunque el agente tiene una ruta predefinida hacia su destino, su percepción está restringida a lo que sus sensores pueden captar en su proximidad inmediata.

El agente "Semáforo", por otro lado, tiene acceso a más información, ya que puede comunicarse tanto con otros semáforos como con los agentes "Carro". Sin embargo, incluso para este agente, algunos elementos del entorno, como secciones de calles e intersecciones, permanecen ocultos, lo que limita su capacidad para razonar sobre ciertas situaciones. En general, la accesibilidad del ambiente se estima entre el 45% y el 55%.

Mayormente determinista

El ambiente es mayormente determinista, ya que las acciones actuales de los agentes determinan en gran medida el estado siguiente del entorno. Por ejemplo, en el caso de los agentes "Carro", su posición actual y la presencia de otros vehículos cercanos influyen directamente en su próximo movimiento. Factores como su destino, la ruta planificada y las reglas de prioridad en las intersecciones son determinantes en su comportamiento.

Para los agentes "Semáforo", el número de carros en su fila impacta su lógica interna para cambiar de estado, buscando reducir el tráfico vehicular. Aunque el ambiente tiene un grado de incertidumbre debido a la creación o eliminación de agentes "Carro" sin considerar el estado global del entorno, la mayoría de las interacciones siguen patrones predecibles y causales.

No episódico

El ambiente no es episódico, ya que las acciones de los agentes no se dividen en ciclos independientes. Cada acción tiene un impacto continuo y acumulativo en el entorno. Los agentes necesitan considerar el estado previo y planificar sus movimientos futuros. Por ejemplo, un agente "Carro" debe evaluar constantemente su entorno inmediato para determinar cómo avanzar, evitando colisiones o seleccionando la mejor ruta hacia su destino.

En el caso de los agentes "Semáforo", su estado actual (como el color) depende directamente del estado anterior y de factores como la cantidad de carros en espera. Este comportamiento no episódico permite que las decisiones y acciones de los agentes sean dinámicas y basadas en condiciones previas.

Dinámico

El ambiente es dinámico, ya que puede cambiar mientras los agentes están deliberando sus próximas acciones. Esto se debe a que otros agentes en el sistema también están realizando decisiones y acciones en paralelo. Por ejemplo, mientras un agente "Carro" evalúa si avanzar o no, los semáforos pueden cambiar su estado, afectando las decisiones y movimientos de los carros.

Esta naturaleza dinámica es clave para representar situaciones de tráfico realistas, donde los cambios en el entorno ocurren constantemente y requieren que los agentes adapten su comportamiento en tiempo real.

Discreto

El ambiente es discreto, ya que tiene un número limitado de acciones y percepciones disponibles para los agentes. Esto significa que es posible enumerar

todas las opciones y estados del sistema. Los agentes "Carro", por ejemplo, pueden realizar acciones específicas como avanzar, detenerse, entrar o salir de un estacionamiento.

Los agentes "Semáforo" también tienen un conjunto definido de acciones, principalmente cambiar su estado (color). Esta discreción permite una simulación más manejable y facilita el análisis del comportamiento de los agentes en un entorno bien estructurado.

Explicación detallada de los agentes

Agente "Carro"

El agente "Carro" simula vehículos que comienzan su recorrido en un estacionamiento y buscan llegar a su destino. Solo pueden aparecer en estacionamientos vacíos y calculan su ruta más corta utilizando el algoritmo A* (Astar). Este algoritmo les permite determinar el camino más eficiente considerando la distancia y las restricciones del entorno.

Los carros siguen ciertas reglas, como respetar prioridades al avanzar y detenerse en semáforos rojos. Además, deben adaptarse a cambios en el entorno, como intersecciones congestionadas, donde pueden tomar decisiones alternativas para aliviar el tráfico y continuar hacia su destino.

Agentes "Semáforo"

Los agentes "Semáforo" (o simplemente semáforos) son los responsables de regular y mejorar el flujo de tráfico en nuestra simulación. Su objetivo principal es reducir los embotellamientos y mantener un tránsito continuo en las intersecciones. Aunque su funcionalidad actual es básica debido al enfoque inicial en los agentes "Carro", a continuación se detalla su comportamiento y las mejoras proyectadas para futuras entregas.

Estos agentes cuentan con sensores avanzados que les permiten una mayor percepción del entorno y facilitan una comunicación activa con otros semáforos. Desde el inicio de la simulación, los semáforos se colocan en posiciones predeterminadas según el diseño del mapa y están orientados en una dirección específica, lo que determina qué vías o calles regulan. Esta orientación es clave para coordinar el tráfico de manera eficiente.

Tipos de Semáforos

1. **Semáforo horizontal**
 - Controla el flujo de vehículos que se desplazan hacia la izquierda o derecha de su posición.
2. **Semáforo vertical**

- Regula el tráfico de vehículos que avanzan hacia arriba o abajo de su posición.

Al inicio de la simulación, los semáforos verticales se configuran en verde, mientras que los horizontales comienzan en rojo. Este esquema inicial garantiza un balance del 50% de semáforos en verde y rojo, lo que es útil para mantener un flujo de tráfico equilibrado. Aunque esta configuración no es definitiva, se utiliza como un punto de partida funcional para la simulación.

Funcionalidad Proyectada

Para la entrega final, se espera implementar una lógica más avanzada en los semáforos. Esto incluirá la capacidad de detectar la cantidad de vehículos esperando en sus respectivas filas y ajustar su estado (verde o rojo) en función de esta información. Este comportamiento permitirá optimizar aún más el flujo vehicular en las intersecciones y adaptarse dinámicamente a las condiciones del tráfico en tiempo real.

Los semáforos, junto con los agentes "Carro", forman un sistema colaborativo que busca modelar situaciones de tráfico urbano de manera realista y eficiente.

IV. DIAGRAMAS PEAS

Tomando en cuenta los diagramas de PEAS, La descripción de estos en base a sus siglas. (Performance, Environment, Actuators, Sensors) define cómo cada agente opera en un sistema. *Performance* establece los objetivos del agente, *Environment* es el contexto donde interactúa, *Actuators* son sus mecanismos de acción (como cambios de luces en semáforos) y *Sensors* son los dispositivos que capturan información del entorno para guiar sus decisiones.

Semáforos:

- *P (Performance)*: Controlar el paso de vehículos, minimizando el tiempo de espera en la intersección y optimizando el flujo en ambas direcciones.
- *E (Environment)*: Intersección con flujo vehicular.
- *A (Actuators)*: Cambios en la luz (amarillo, verde, y eventualmente rojo si es necesario).
- *S (Sensors)*: Detectores de proximidad y sensores de comunicación para recibir tiempos de llegada de los vehículos.

Vehículos:

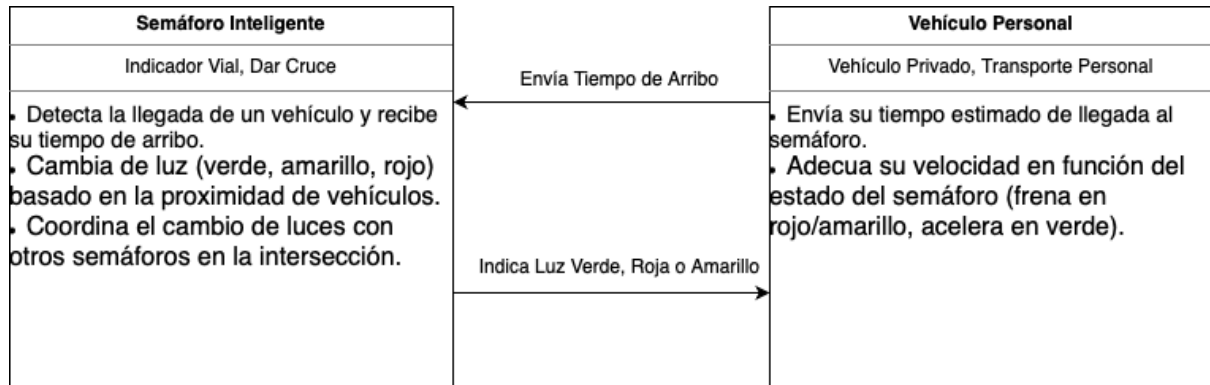
- *P (Performance)*: Llegar al destino con el menor tiempo de espera.
- *E (Environment)*: Vías de la intersección.
- *A (Actuators)*: Aceleración, frenado y cambio de dirección.
- *S (Sensors)*: Detectores de otros vehículos y comunicación con semáforos.

V. DIAGRAMAS DE AGENTES USANDO AUML Y DIAGRAMA SMA

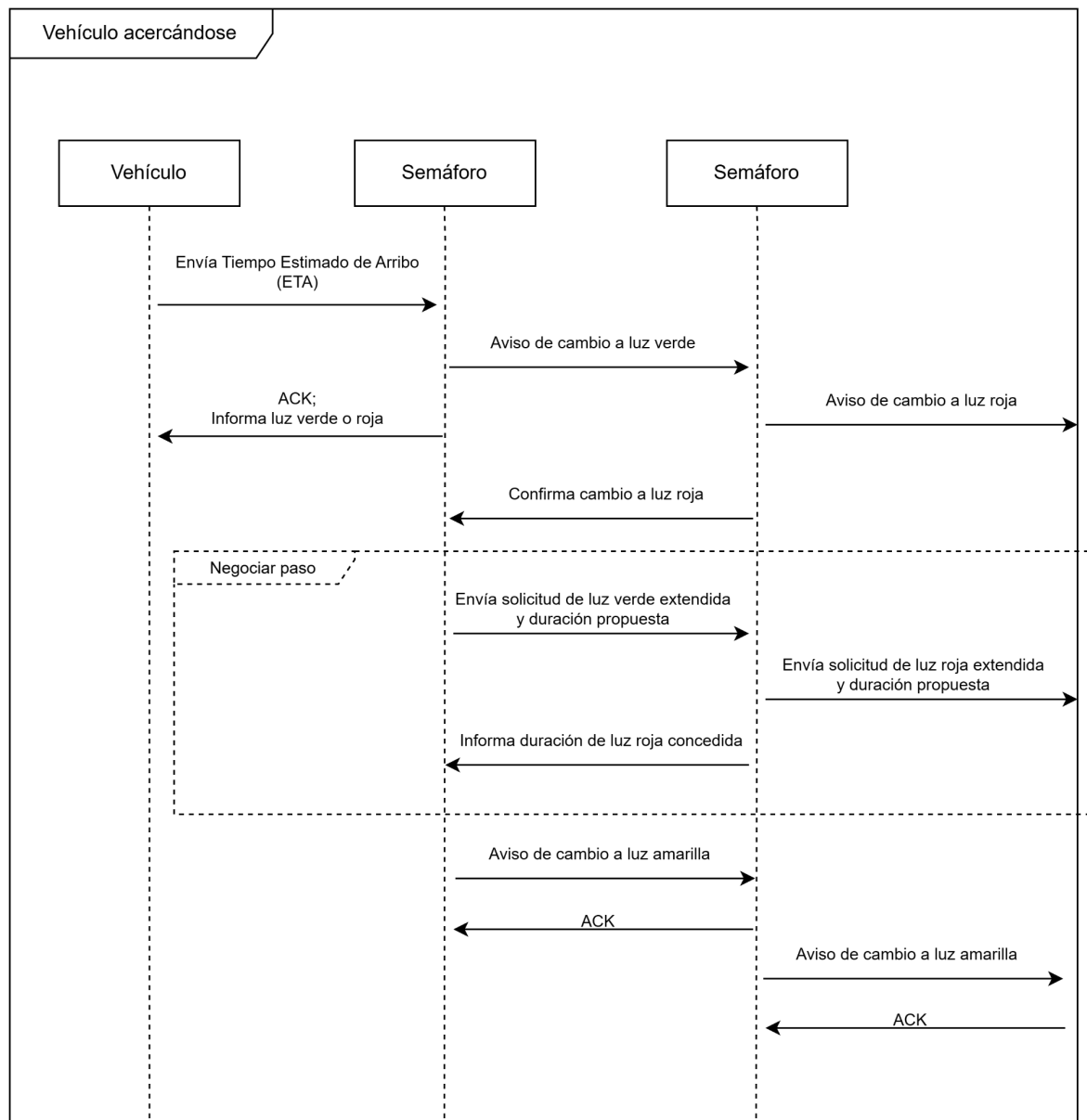
Agentes:

Semáforos Inteligentes: Controla la señalización de la intersección y responde a los coches que se acercan.

Vehículo: Representa a los coches que se aproximan a la intersección.



VI. DIAGRAMA DE INTERACCIÓN



VII. PLAN DE TRABAJO Y APRENDIZAJE ADQUIRIDO

Plan de trabajo

A continuación se presentan las actividades que ya han sido completadas en cuanto al trabajo realizado así como las actividades que hacen falta por completar para tener el proyecto terminado en su totalidad:

Actividad	Responsable	Fecha límite de realización	Intervalo de Esfuerzo Estimado.	Intervalo de Esfuerzo final	% Completado
Desarrollo de diagrama de agentes	Santiago Villazón Ponce de León	5 de Noviembre	2 horas	1 hora	100%
Definición de plan de trabajo	Juan Antonio Figueroa Rodríguez	5 de Noviembre	30 minutos	30 min	100%
Definición de tecnologías	Iván Alexander Ramos Ramírez	5 de Noviembre	45 minutos	20 min	100%
Investigación sobre el desarrollo de los agentes.	Sebastián Antonio Almanza	5 de Noviembre	3 horas	2 horas	100%
Ideación de los elementos de la visualización	Juan Antonio Figueroa Rodríguez y Sebastián Antonio	5 de Noviembre	2 horas	30 min	100%
Modelación de elementos de la visualización utilizando Blender	Juan Antonio Figueroa Rodríguez	8 de Noviembre	2 Semanas	3 Semanas	100%
Desarrollo de diagrama de agentes con AUML	Santiago Villazón Ponce de León	9 de Noviembre	1 horas	1 hora	100%
Desarrollo de la definición del medio ambiente	Santiago Villazón Ponce de León	9 de Noviembre	1 horas	1 hora y media	100%

Desarrollo de Diagramas PEAS	Santiago Villazón Ponce de León	9 de Noviembre	20 min	30 min	100%
Desarrollo de agentes (vehículo y semáforo) usando Mesa	Sebastián Antonio Almanza, Iván Alexander y Santiago Villazón Ponce de León	11 de Noviembre	2 Días	2 Días	100%
Desarrollo de modelo usando Mesa	Sebastián Antonio Almanza, Iván Alexander y Santiago Villazón Ponce de León	11 de Noviembre	2 Días	2 Días	100%
Desarrollo de visualización utilizando Matplotlib	Iván Alexander y Santiago Villazón Ponce de León	11 de Noviembre	3 Días	1 Día	100%

Actividades pendientes por terminar

Actividad	Responsable(s)	Fecha límite de realización	Intervalo de tiempo estimado	% Completado
Creación de visualización en 3D en Unity	Sebastián Antonio Almanza	13 de Noviembre	1 Día	70%
Desarrollo de agente del peatón	Santiago Villazón Ponce de León	22 de Noviembre	2 Días	20%

Terminar todos los modelados de Blender	Juan Antonio Figueroa	22 de Noviembre	1 Día	80%
Incorporar todos los modelados dentro de Unity	Sebastián Antonio Almanza y Juan Antonio Figueroa	22 de Noviembre	20 Min	10%
Conectar simulación de Mesa a Unity	Sebastián Antonio Almanza	23 de Noviembre	3 horas	50%
Rectificar que la simulación se ejecuta correctamente en Unity	Todo el equipo	25 de Noviembre	20 min	0%

Aprendizaje Adquirido

Sebastián:

Durante el desarrollo de este 60% del proyecto de gráficas con sistemas de multiagentes considero que los aprendizajes más importantes que he adquirido en la parte de agentes es el cómo se conforman los sistemas de multiagentes así como cuáles son las mejores prácticas para desarrollar agentes que se comporten de forma adecuada y que puedan comunicarse con otros agentes, así mismo el cómo estos se implementan en la programación utilizando python. En la parte de gráficas considero que los mayores aprendizajes han sido el aprender a modelar objetos utilizando Blender, así como el saber como colocar y ajustar correctamente la iluminación para que las escenas en Unity tengan un aspecto correcto sin tomas quemadas o con poca iluminación.

Santiago:

En esta fase del proyecto, he desarrollado un conocimiento integral que abarca tanto los sistemas multiagentes como las habilidades en diseño gráfico. En el área de los agentes, aprendí a diseñar sistemas que permiten la interacción eficiente entre múltiples agentes, asegurando comportamientos adecuados y comunicación fluida, todo implementado mediante Python. Por otro lado, en el aspecto gráfico, he fortalecido mis habilidades en modelado 3D utilizando Blender, creando

elementos visuales que enriquecen el proyecto. Asimismo, adquirí experiencia en el manejo de iluminación dentro de Unity, logrando escenas equilibradas que evitan problemas como sobreexposición o falta de luz. Este proceso ha sido clave para consolidar mi capacidad de integrar soluciones técnicas y estéticas, contribuyendo de manera efectiva al desarrollo del proyecto.

Antonio:

En esta etapa del proyecto, he alcanzado aprendizajes clave en dos áreas principales: los sistemas multiagentes y el diseño gráfico. En el ámbito de los agentes, he logrado entender cómo estructurar estos sistemas para que los agentes interactúen de manera eficiente y desarrollen comportamientos adecuados, utilizando herramientas como Python para su implementación. Sin embargo, el mayor énfasis de mi avance está en el uso de Blender. El aprendizaje de modelado 3D ha sido fundamental, ya que me ha permitido crear objetos como coches, estructuras etc. que ayudan al proyecto. Además, he aprendido a trabajar con iluminación en Unity, asegurando que las escenas tengan un equilibrio visual adecuado, evitando tomas quemadas o poco iluminadas. Este enfoque en Blender ha sido crucial para darle al proyecto un aspecto visual, sólido y profesional.

Iván:

En esta etapa del proyecto, he tenido la oportunidad de familiarizarme con conceptos y herramientas clave que no conocía previamente. He aprendido a manejar control de versiones con GitHub, lo que me ha permitido trabajar de manera más eficiente en equipo, gestionando cambios y manteniendo el código organizado. En el ámbito del desarrollo multiagentes, he adquirido experiencia en la creación e implementación de agentes que interactúan dentro de un sistema, lo que me ha permitido entender cómo coordinar comportamientos y gestionar su comunicación. Además, he perfeccionado mis habilidades para depurar código, especialmente en lo relacionado con la lógica de interacción entre los agentes, lo que ha sido crucial para resolver errores y optimizar el rendimiento del sistema. Este proceso me ha permitido mejorar mi capacidad para desarrollar soluciones ágiles y eficientes en un entorno colaborativo y dinámico.