



Tecnológico de Monterrey

Revisión 3 - Avance al 60%

Arturo Montes González A01798012

Ares Ortiz Botello A01747848

Andrés Iván Rodríguez Méndez A01754650

Rosa Itzel Figueroa Rosas A01748086

23 de noviembre, 2023

Modelación de sistemas multiagentes con gráficas computacionales

Jorge Adolfo Ramírez Uresti

Oriam Renan De Gyves López

Revisión 3 - Avance del 60%

Descripción del medio ambiente

El medio ambiente de la simulación de congestión vial en la Ciudad de México presenta características distintivas que impactan directamente en el rendimiento de los agentes involucrados (automóviles, metrobuses, camiones y peatones). Estas son esenciales para comprender cómo los agentes interactúan y toman decisiones en su entorno dinámico. A continuación se brindará una descripción y justificación detallada de las cinco características de nuestro entorno:

1. Accesible vs Inaccessible

El medio ambiente es 100% accesible dado que los sensores de los agentes permiten recopilar información relevante del entorno. Los agentes pueden percibir semáforos, señales de tránsito, otros vehículos y peatones, lo que facilita la toma de decisiones informadas.

2. Deterministic vs Non-deterministic

Nuestro entorno es en gran medida no determinista (70%) debido a la naturaleza dinámica del tráfico urbano. Aunque los agentes pueden anticipar y reaccionar según las señales y eventos detectados, las acciones de otros agentes y las condiciones del tráfico pueden cambiar rápidamente, lo que introduce cierta incertidumbre en el sistema.

3. Episodic vs Non-episodic

Nuestro medio ambiente es 75% episódico, puesto que cada agente experimenta episodios que incluyen percepciones y acciones específicas. Los eventos, como semáforos que cambian de color, señales de tránsito y movimientos de otros agentes, definen los episodios. No obstante, los episodios no son totalmente independientes, ya que las interacciones entre agentes y las condiciones del tráfico pueden influir en eventos futuros.

4. Static vs Dynamic

El entorno es 100% dinámico debido a que las condiciones del tráfico, la presencia de peatones y otros vehículos, así como los cambios en las señales de tránsito y semáforos, se ven modificadas constantemente. Aunque el entorno puede no cambiar a un ritmo extremo, la dinámica del tráfico implica una constante adaptación por parte de los agentes.

5. Discrete vs Continuous

El medio ambiente tiene aspectos continuos (80%), ya que las variables como la velocidad de los vehículos y la posición de los peatones, pueden variar de manera continua. Las acciones, aunque discretas en ejecución (avanzar, detenerse, cambiar de carril), pueden ser influenciadas por factores continuos, como la velocidad actual del vehículo o la posición relativa de otros agentes en el entorno.

Diagramas de agente usando AUMML

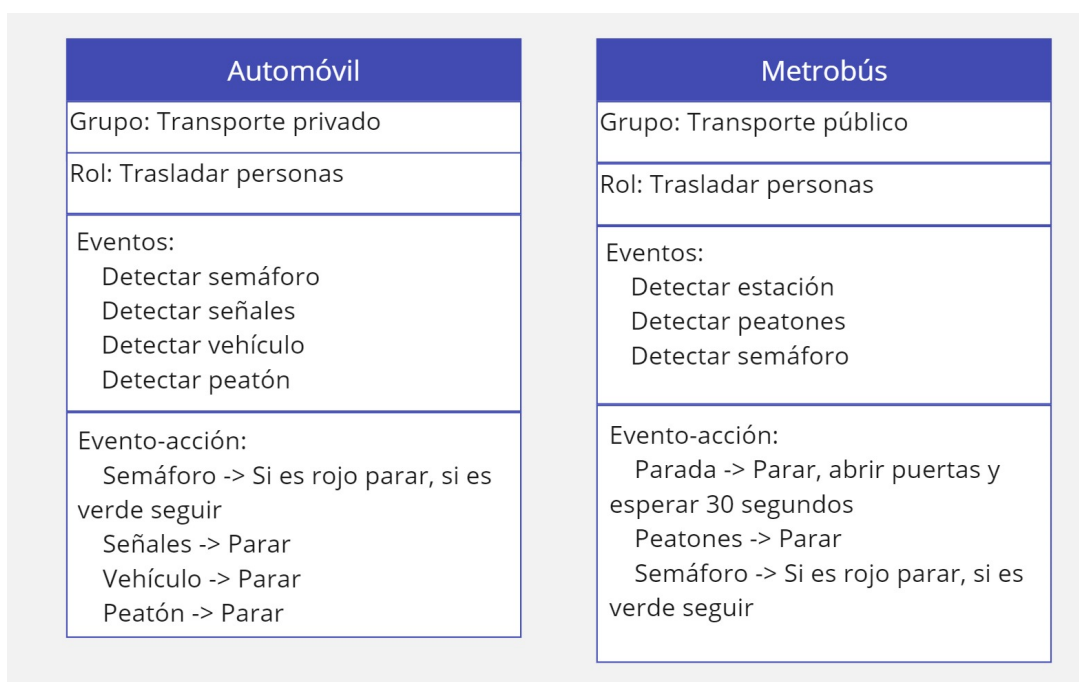


Figura 1. Diagrama de clase de automóvil y metrobús

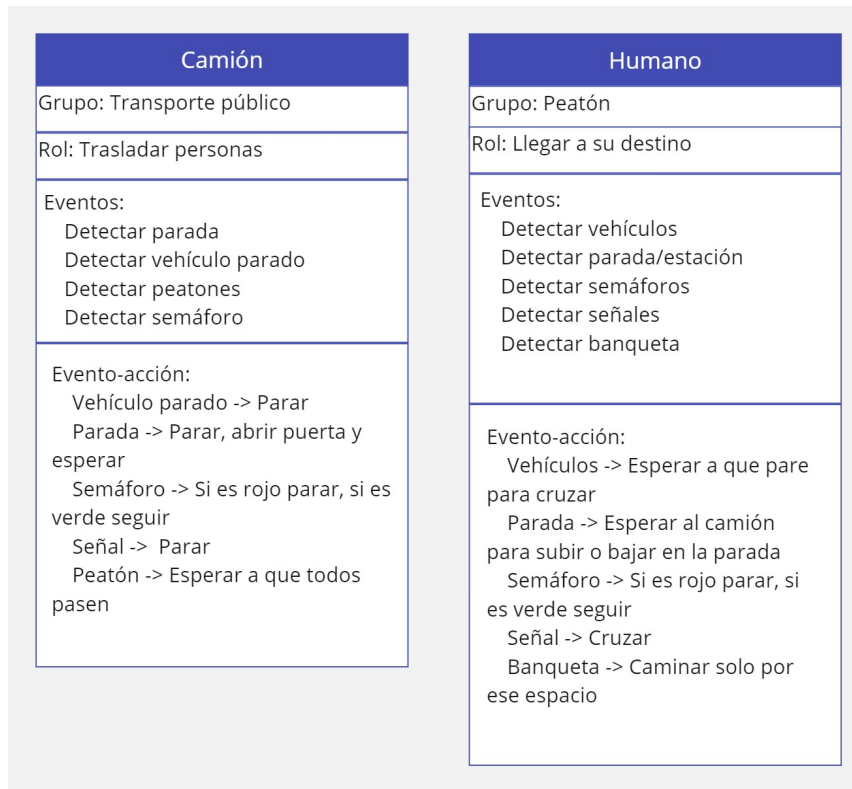


Figura 2. Diagrama de clase de camión y humano

Diagrama de organización SMA

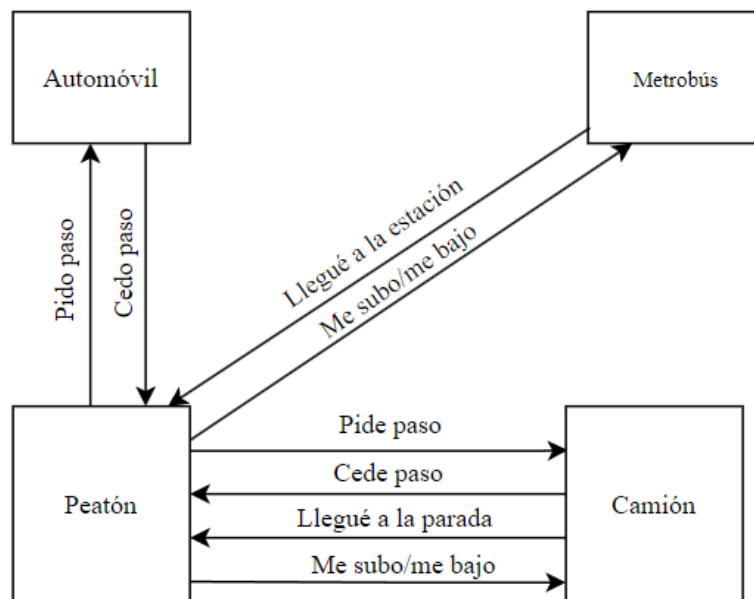


Figura 3. Diagrama de organización de nuestro SMA

Diagrama de interacción entre agentes

Agentes (Camión, Metrobús, Automóvil y Persona)

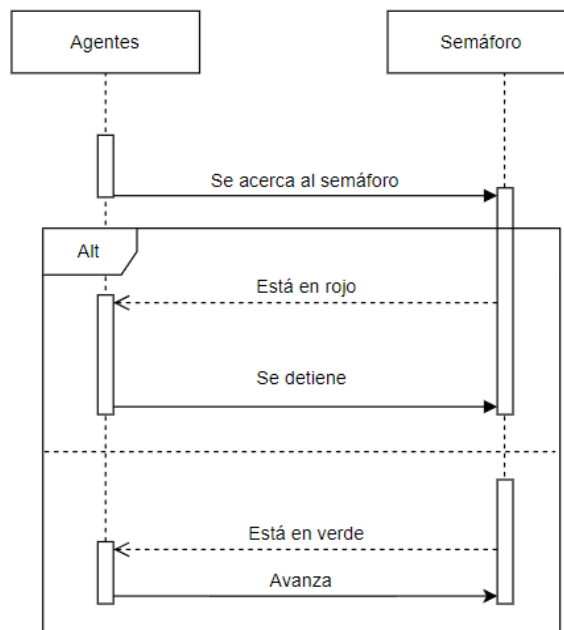


Figura 4. Diagrama de secuencia Agentes (Camión, Metrobús, Automóvil y Persona) - Semáforo

Transporte(Camión, Metrobús, Automóvil) - Semáforo

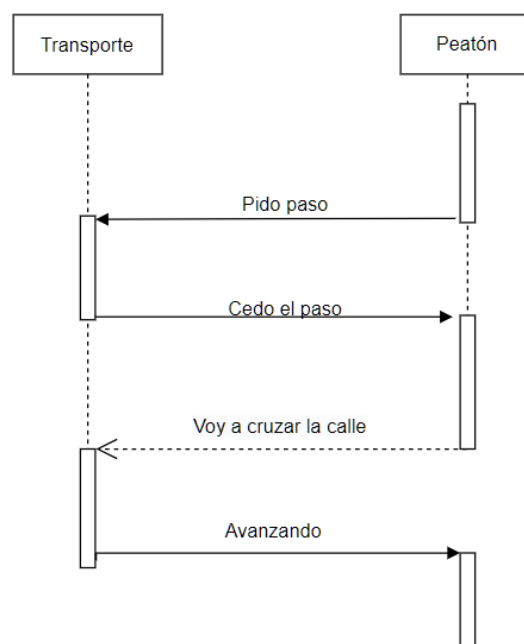


Figura 5. Diagrama de secuencia Transporte - Peatón

Transporte público(Camión y Metrobús) - Semáforo

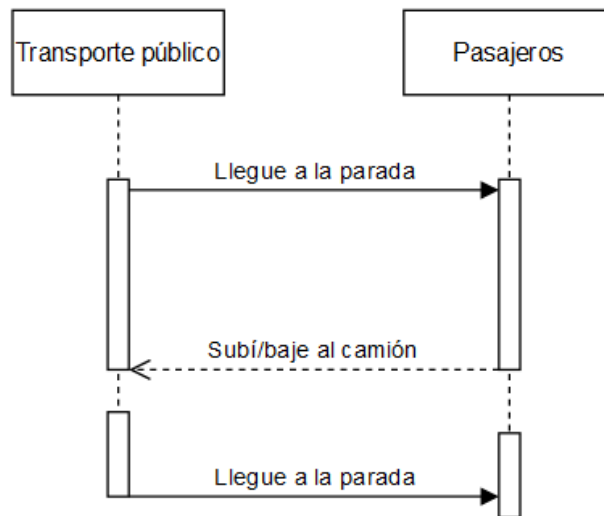


Figura 6. Diagrama de secuencia Transporte público - Pasajero

Transporte privado (Vehículo A y Vehículo B)

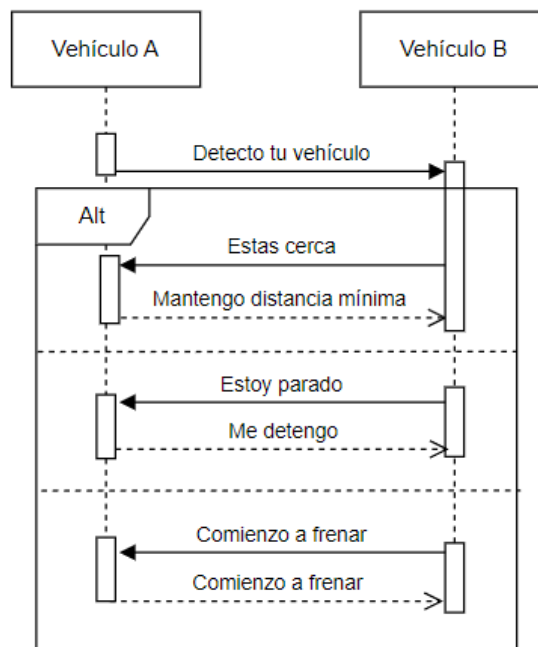


Figura 7. Diagrama de secuencia Vehículo - Vehículo

Código (60%) de la implementación de los agentes

[Multiagentes Python/MESA](#)

Código (60%) de la implementación de la parte gráfica de la solución

[Modelado ciudad Unity](#)

Plan de trabajo

Para esta tercera revisión fue necesario establecer las actividades pendientes y el tiempo en el que se realizarán, así como los responsables de llevarlas a cabo, el intervalo de esfuerzo estimado y la fecha en las que se realizarán y terminarán. Adicionalmente, mostraremos cuáles son las actividades que ya han sido finalizadas, realizando una comparación de diferencia del tiempo que nos tomó realizarlas y el tiempo que inicialmente estimamos.

Actividades pendientes						
Número de actividad	Actividad	Responsable	Esfuerzo estimado	Tiempo estimado	Fecha de inicio	Fecha de término
1	Actualizar el plan de trabajo	Ares Ortiz Botello	1 día	30 minutos	20/11/2023	21/11/2023
2	Definir las nuevas tareas por hacer	Todo el equipo	1 día	30 minutos	20/11/2023	21/11/2023
3	Hacer las correcciones necesarias de la descripción del medio ambiente	Ares Ortiz Botello	1 día	15 minutos	21/11/2023	22/11/2023
4	Definir el aprendizaje adquirido como equipo	Todo el equipo	1 día	15 minutos	22/11/2023	23/11/2023
5	Realizar reflexiones individuales	Todo el equipo	1 día	15 minutos	22/11/2023	23/11/2023

6	Subir documento de revisión 3 al Github	Andrés Iván Rodríguez Méndez	1 día	5 minutos	22/11/2023	23/11/2023
7	Subir el link de Github a canvas	Andrés Iván Rodríguez Méndez	1 día	5 minutos	22/11/2023	23/11/2023
8	Tener un avance del 90% en Unity	Rosa Itzel Figueroa Rosas	5 días	3 horas cada día	22/11/2023	27/11/2023
9	Tener un avance del 95% en Python y MESA	Arturo Montes González y Andrés Iván Rodríguez Méndez	4 días	2 horas cada día	22/11/2023	26/11/2023
10	Empezar con el servidor y API	Ares Ortiz Botello	2 días	1 hora y 30 minutos	23/11/2023	25/11/2023
11	Tener un avance del 80% en el servidor y API	Ares Ortiz Botello	4 días	3 horas cada día	23/11/2023	27/11/2023
12	Hacer pruebas unitarias	Todo el equipo	2 días	2 horas y 30 minutos cada día	25/11/2023	27/11/2023
13	Hacer pruebas de integración	Todo el equipo	2 días	2 horas y 30 minutos cada día	25/11/2023	27/11/2023
14	Agregar personas y metrobús en Python y Mesa	Arturo Montes González y Andrés Iván Rodríguez Méndez	1 día	3 horas	23/11/2023	24/11/2023
15	Conectar Unity con el servidor	Ares Ortiz Botello y Rosa Itzel Figueroa Rosas	1 día	1 hora	23/11/2023	24/11/2023
16	Conectar	Arturo	1 día	1 hora	24/11/2023	25/11/2023

	Python con el servidor	Montes González, Ares Ortiz Botello y Andrés Iván Rodríguez Méndez				
17	Realizar pruebas/peticiones del servidor a Unity	Ares Ortiz Botello y Rosa Itzel Figueroa Rosas	2 días	2 horas y 30 minutos cada día	24/11/2023	26/11/2023
18	Tener el reto completo en un 100%	Todo el equipo	7 días	4 horas cada día	21/11/2023	28/11/2023
19	Diseñar la presentación final	Todo el equipo	2 días	1 hora cada día	26/11/2023	28/11/2023
20	Elaborar documentos de entregas finales	Todo el equipo	2 días	2 horas cada día	28/11/2023	30/11/2023
21	Subir evidencias a elumen y canvas	Todo el equipo	1 día	30 minutos	29/11/2023	30/11/2023

Figura 10. Tabla de actividades pendientes

Actividades finalizadas					
Número de actividad	Actividad	Responsable	Tiempo estimado	Tiempo real	Diferencia entre tiempos
1	Reunión de inicio del proyecto para definir objetivos y roles	Todo el equipo	1 hora	1 hora	Ninguna
2	Crear el documento inicial	Ares Ortiz Botello	5 minutos	3 minutos	2 minutos
3	Crear repositorio de Github	Arturo Montes González	10 minutos	5 minutos	5 minutos
4	Identificar fortalezas y áreas de oportunidad de	Todo el equipo	20 minutos	10 minutos	10 minutos

	cada integrante				
5	Definir expectativas y metas para el proyecto	Todo el equipo	30 minutos	20 minutos	10 minutos
6	Definir compromisos personales	Todo el equipo	20 minutos	10 minutos	10 minutos
7	Identificar los agentes de nuestro proyecto y definir sus relaciones/interacciones	Todo el equipo	45 minutos	45 minutos	Ninguna
8	Crear los diagramas de clase y protocolos de interacción	Todo el equipo	1 hora 30 minutos	2 horas	30 minutos
9	Redactar aprendizajes obtenidos y reflexiones individuales	Todo el equipo	15 minutos	10 minutos	5 minutos
10	Empezar a modelar los agentes en Python con el uso de MESA	Arturo Montes Gonzáles y Andrés Iván Rodríguez Méndez	2 horas	2 horas	Ninguna
11	Empezar a modelar la ciudad en Unity	Rosa Itzel Figueroa Rosas	2 horas	2 horas	Ninguna
12	Subir documento "Revisión 1" a Github	Andrés Iván Rodríguez Méndez	5 minutos	3 minutos	2 minutos
13	Entregar link de Github en canvas	Andrés Iván Rodríguez Méndez	5 minutos	2 minutos	3 minutos
14	Reunión con el profesor para revisión y retroalimentación de los diagramas realizados en la primera entrega	Todo el equipo	2 horas	2 horas	Ninguna
15	Creación del documento para la Revisión 2	Ares Ortiz Botello	5 minutos	3 minutos	2 minutos
16	Agregar un índice	Rosa Itzel Figueroa Rosas	15 minutos	15 minutos	Ninguna
17	Corregir los diagramas necesarios	Rosa Itzel Figueroa Rosas y Ares Ortiz Botello	45 minutos	40 minutos	5 minutos

18	Hacer la descripción del medio ambiente	Todo el equipo	50 minutos	40 minutos	10 minutos
19	Hacer la descripción PEAS de cada agente	Todo el equipo	50 minutos	50 minutos	Ninguna
20	Diseñar un diagrama de organización SMA	Rosa Itzel Figueroa Rosas y Ares Ortiz Botello	30 minutos	40 minutos	10 minutos
21	Documentar aprendizajes adquiridos y reflexiones individuales	Todo el equipo	15 minutos	10 minutos	5 minutos
22	Subir documento al repositorio de Github	Andrés Iván Rodríguez Méndez	5 minutos	5 minutos	Ninguna
23	Hacer la entrega del link en canvas	Andrés Iván Rodríguez Méndez	5 minutos	3 minutos	2 minutos
24	Continuar y avanzar con la modelación de la ciudad en Unity	Rosa Itzel Figueroa Rosas	1 hora	1 hora	Ninguna
25	Continuar y avanzar con la programación del SMA en Python con el uso de MESA	Arturo Montes Gonzáles y Andrés Iván Rodríguez Méndez	1 hora	1 hora	Ninguna
26	Hacer la entrega del link en canvas	Andrés Iván Rodríguez Méndez	5 minutos	3 minutos	2 minutos
27	Crear documento para la Revisión 3	Rosa Itzel Figueroa Rosas	5 minutos	3 minutos	2 minutos

Figura 11. Tabla de actividades finalizadas

Aprendizaje adquirido y reflexiones individuales

Arturo: A lo largo del desarrollo del reto, se ha aprendido mucho acerca de convertir un modelo con coordenadas discretas a uno con coordenadas continuas. También se ha aprendido en la sincronización y envío de datos entre dos sistemas dependientes para tener una simulación en tiempo real.

Ares: Al haber realizado esta actividad pude darme cuenta de lo importante que es llevar un plan de trabajo actualizado para cada entrega, puesto a que así hay una mayor organización y noción de lo que se tiene que hacer y cómo. De igual forma, con los continuos cambios y correcciones que se han hecho en los diagramas de agentes, he podido entender con más profundidad lo que caracteriza a un SMA y al medio ambiente en el que se desarrolla. Finalmente, empecé a aprender sobre la creación de un servidor usando flask, para así lograr una comunicación entre todos nuestros programas.

Andrés: En general he aprendido mucho sobre la implementación de un sistema multiagentes, gracias al desarrollo de los diagramas y las correcciones que hemos realizado he podido pensar en posibles mejoras para la implementación del reto.

Rosa: Durante esta revisión logré profundizar mis conocimientos y poner en práctica el uso de distintas herramientas dentro del entorno de Unity. Por otro lado, también pude entender de una forma más clara el proceso de desarrollo de los diagramas necesarios para crear un Sistema Multiagente.