

Tecnológico de Monterrey

Campus Querétaro

Modelación de sistemas multiagentes con gráficas computacionales

### **Reporte sobre la Solución**

Profesores

Denisse L. Maldonado Flores

Alejandro Fernández Vilchis

Pedro Oscar Pérez Murueta

Presenta

Cristian Rogelio Espinosa Díaz A01702752

Oscar Eduardo Nieto Espitia A01705090

Jaime López Hernández A00571842

Fecha:

02 de Diciembre de 2022

# Índice

<b>1. Introducción del problema</b>	<b>3</b>
<b>2. Propuesta de solución</b>	<b>3</b>
<b>3. Comportamiento del entorno</b>	<b>4</b>
3.1 Comportamiento del Modelo	4
3.2 Comportamiento del Agente	4
<b>4. Resultados obtenidos</b>	<b>5</b>
<b>5. Conclusión</b>	<b>6</b>

# **Reporte solución Reto Multiagentes**

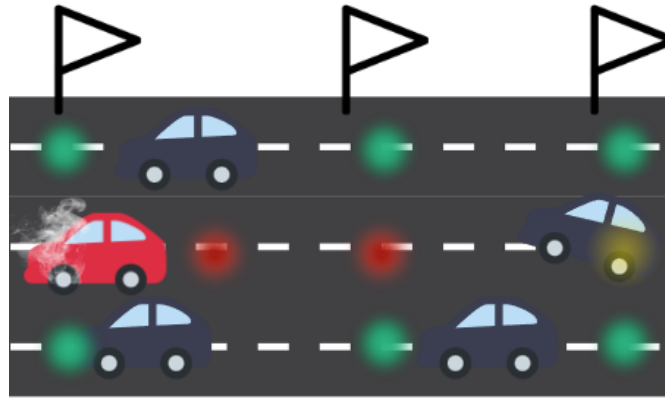
## **1. Introducción del problema**

La congestión (embotellamientos) es una fuente importante de descontento y frustración para los habitantes de áreas urbanas y suburbanas. Debido a los embotellamientos, los países de todo el mundo pierden miles de millones de dólares cada año. Según el informe de Movilidad Urbana 2010 del Instituto de Transporte de Texas indica que alrededor de 439 áreas urbanas en Estados Unidos están experimentando problemas debido a la congestión y está empeorando en regiones de todos los tamaños.

Se nos plantea que el primer escenario en donde hay una carretera de 1 km de largo y la velocidad máxima en los tres carriles es de 60 km/h, el problema plantea que a la mitad de la simulación un vehículo se descompone y se detiene a la mitad de la carretera en el carril central. También el primer escenario plantea que los vehículos solo deben de cambiarse de carril una vez que los vehículos solo intentarán cambiar de carril una vez que detecten el vehículo descompuesto.

## **2. Propuesta de solución**

Con el fin de agilizar la movilidad vial del primer escenario, nosotros propusimos luces en la carretera como medio para avisar al conductor que en el carril actual existe congestión o no. Esta solución se compone por dos partes del entorno que trabajan en conjunto, por una parte está el checkpoint que es el encargado de recolectar y almacenar la cantidad de vehículos dentro de X distancia y almacena la velocidad promedio dentro de ese segmento, por otro lado, tenemos las luces, estas son controladas por el checkpoint, el color de la luz va a ser determinado por la cantidad de vehículos en dicho segmento y la velocidad promedio, la idea es que el vehículo “observe” las luces colocadas en su carril y determine si el o los carriles adyacentes es una mejor opción para llegar a su destino. Con esta propuesta nos podemos asegurar que los vehículos siempre elegirán el carril más rápido y con menor cantidad de vehículos, con el fin de evitar el tráfico sin necesidad de “observar” el vehículo descompuesto



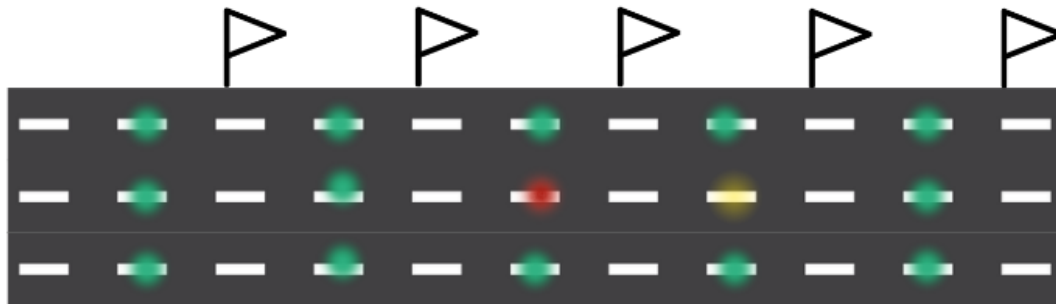
*Figura 1. Ilustración propuesta de solución*

### 3. Comportamiento del entorno

#### 3.1 Comportamiento del Modelo

El modelo spawna vehículos en cada carril dependiendo de la probabilidad de spawneo, con esto nos aseguramos que cada simulación sea un poco diferente a la anterior con el fin de analizar los dos escenarios. Además de esto, también se encarga de llevar el control de los checkpoints.

Para la generación de los checkpoints, después de hacer varias pruebas, determinamos que lo mejor es colocar un checkpoint cada 200 metros, a nivel de código y por conveniencia lo diseñamos como una matriz de 3 x 5 la cual guarda cantidad de vehículos y velocidad promedio del segmento, lo cual quedaría algo así:



*Figura 2. Ilustración checkpoints*

#### 3.2 Comportamiento del Agente

Pensar en las percepciones y en los actuadores del vehículo fue una de las partes más difíciles de este reto, básicamente nos llevó a pensar en la forma más primitiva del comportamiento de un vehículo.

El comportamiento es el siguiente: Primero verifica la distancia que tiene con el vehículo de enfrente, usamos una variable para determinar su campo de visión y si un vehículo no se encuentra dentro de su campo de visión simplemente va acelerando hasta alcanzar la velocidad máxima, En el caso de que tenga un vehículo enfrente comprueba su velocidad y si ese vehículo tiene menor velocidad desacelera, y si tiene mayor velocidad acelera.

Nuestra propuesta de solución determina cuando debe de cambiar de carril el vehículo, la secuencia de pasos que sigue el agente para cambiar de carril es el siguiente:

- Primero determina si el carril o los carriles adyacentes son una mejor opción que en el que me encuentro ahora, esto se define por varios parámetros, mi velocidad versus la velocidad promedio del carril adyacente, en el caso de que sea la misma, también compara la cantidad de vehículos en los carriles adyacentes.
- Una vez que determina si un carril es mejor opción, el vehículo va a tomar la decisión si el camino está despejado para cambiar de carril, si no encuentra vehículo o si el vehículo está suficientemente lejos para cambiar de carril, este ejecuta una animación para cambiar de carril, hasta finalmente incorporarse al carril.

#### 4. Resultados obtenidos

Tras correr la simulación 10 veces por escenario (Muestra no representativa), obtuvimos una mejora en la velocidad promedio de los vehículos, como podemos observar en la figura 3, hay un incremento en la variedad de velocidades, así como una reducción de la cantidad de vehículos que se quedan totalmente estacionarios. El promedio de las velocidades de los vehículos fue de 12.6 m/s para nuestra propuesta de solución y de 11.51 m/s para el caso base.

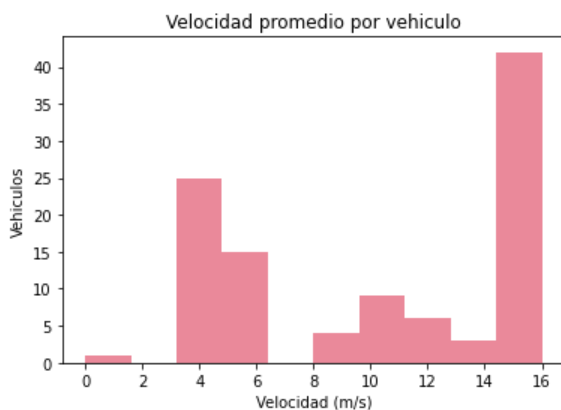


Figura 3. Con propuesta de solución

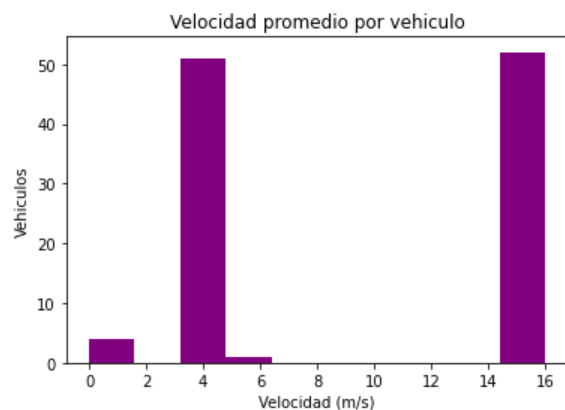


Figura 4. Sin propuesta de solución

En cuanto a la cantidad de vehículos que llegaron a su destino, obtuvimos un promedio de 350 vehículos con nuestra propuesta de solución y 323 sin la propuesta de solución. La simulación se ejecutó en un lapso de 5 minutos.

Link de los resultados obtenidos:

[https://docs.google.com/spreadsheets/d/1s6pqdyU7eHMjtKXgikagML4iiKK9\\_71L030zP1qqdd4/edit?usp=sharing](https://docs.google.com/spreadsheets/d/1s6pqdyU7eHMjtKXgikagML4iiKK9_71L030zP1qqdd4/edit?usp=sharing)

## **5. Conclusión**

Este reto comenzó con unas reglas relativamente sencillas que había que seguir, y a partir de ahí empezamos el análisis y el diseño de los agentes y el entorno. Uno de los retos a los que nos enfrentamos tenía que ver con el envío de la información necesaria para evaluar adecuadamente qué carriles eran los más rápidos para los agentes. Asimismo, al trasladar el modelo a Unity hubo algunos problemas con las frecuencias de los agentes y esto hizo que la posición de los vehículos, de vez en cuando, no estuviera sincronizada con la información que se proporcionaba.

Por otra parte, al visualizar la comparativa de la propuesta con el caso base, se puede observar que las mejoras identificadas, tales como la cantidad de vehículos que llegaron a su destino dentro del plazo establecido y las mayores velocidades, tienen un ligero aumento en estos parámetros. Por lo consiguiente, se podría trabajar en el modelo actual para ajustarlo y mejorarlo, o podría adoptarse un enfoque diferente con el propósito de crear menos embotellamientos y de esta manera agilizar el flujo del tráfico en general.