

Evidencia 2. Avances



A00572499 - Santiago Gutierrez

A00832425 - Daniel Rubies Isla

A01252831 - Luis Carlos Rico Almada

A00837426 - Pedro Gabriel Sánchez Valdez

Equipo 5



Planteamiento del problema y objetivos

Planteamiento :

- Desarrollar una simulación realista y funcional en Unity que modele las interacciones entre peatones y conductores en un entorno urbano= considerando una amplia variedad de comportamientos y situaciones inesperadas para analizar y mejorar la seguridad vial= la eficiencia del tránsito y la dinámica de movilidad urbana.

Objetivos :

- Diseñar comportamientos variados para peatones que reflejan situaciones del mundo real
- Implementar comportamientos realistas en los conductores
- Simular escenarios urbanos con variaciones en la infraestructura y el entorno
- Incorporar comportamientos proactivos y reactivos en peatones y conductores
- Crear un sistema de registro y análisis de datos dentro de Unity



Definición de ambiente

Descrito de manera formal= el ambiente con el que trabajamos es discreto por ser un Grid de AgentPy= es accesible ya que todos los agentes tienen acceso a la matriz de información tal como la posición de los demás= determinístico porque el ambiente no cambia de estados en base a circunstancias externas y es predecible en sí mismo= episódico porque correr la simulación no es influenciada por ejecuciones previas ni influencia en futuras= y estática por no ser cambiante.



Definición de agentes

Dentro de nuestro ambiente existirán 3 diferentes agentes

1. Peatón : Agente que caminará en las banquetas y cruces peatonales
2. Vehículo : Agente que maneja únicamente en la calle= evitando colisiones con otros agentes.
3. Semáforo : Agente que modifica el ambiente y los espacios en los que los peatones o vehículos puedan estar.

UML

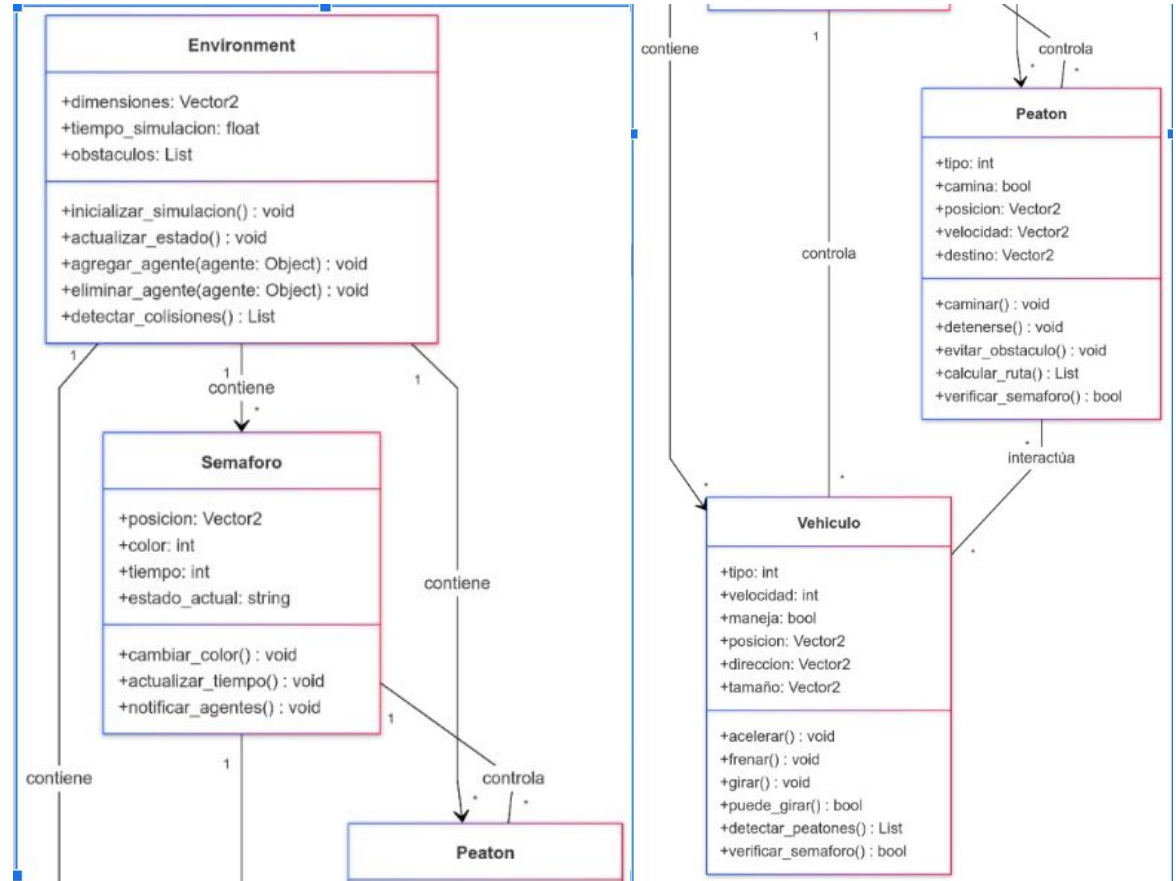


Diagrama de flujo

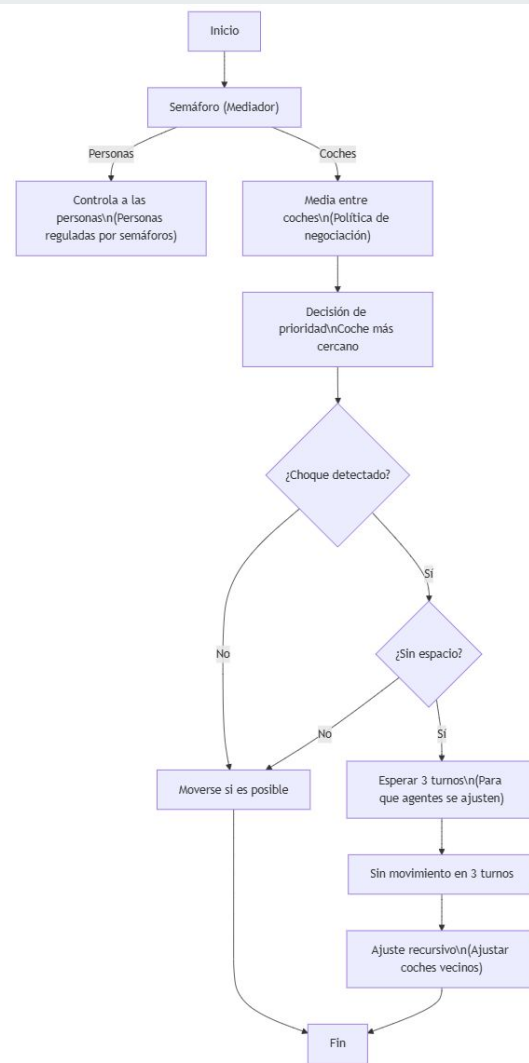
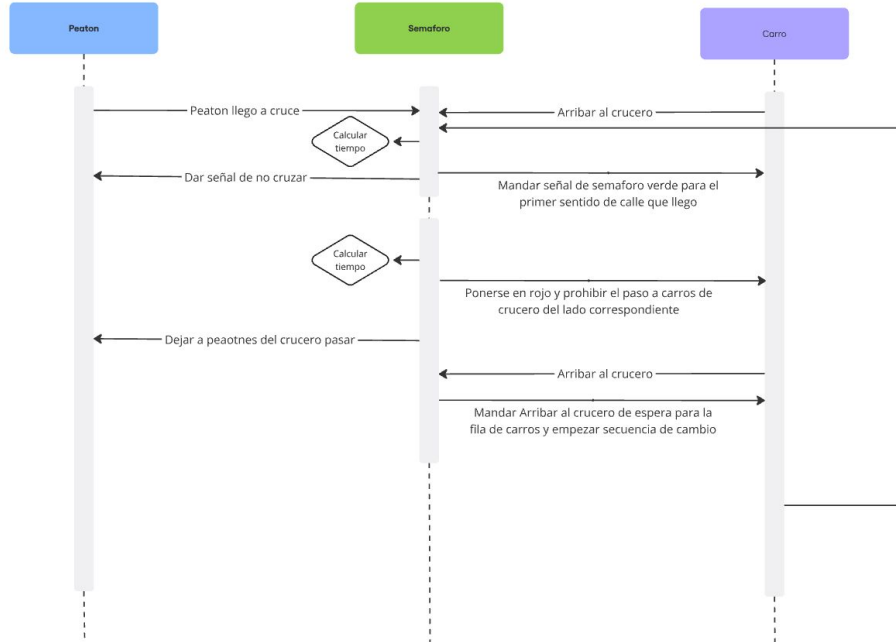


Diagrama de comunicación



$$Q(s, a) \leftarrow Q(s, a) + \alpha[r + \gamma \max_{a'} Q(s', a') - Q(s, a)]$$



Análisis de la solución

¿Por qué se seleccionó este modelo multi-agente? Q - learning permite que los agentes aprendan lo cual es más realista para una simulación

¿Qué variables se consideraron en la toma de decisiones? Estado de semáforo= posición/orientación= tipo de celda= recompensas=

¿Cómo interactúan estas variables con los resultados de la simulación? Epsilon controla la exploración= recompensas guían rutas= orientación penaliza= semáforos regulan cruces

¿Por qué se eligió este diseño gráfico? Colores por cada elemento para una clara visualización= hay semáforos= cruces= direcciones

¿Cuáles son las ventajas de la solución final? Aprendizaje adaptativo= Múltiples agentes = Reglas realistas y Visual claro

¿Cuáles son las desventajas? Largo entrenamiento = Baja escalabilidad

¿Qué modificaciones podrían mejorar o eliminar estas desventajas? Optimizar entrenamiento



Ambiente

e r g w

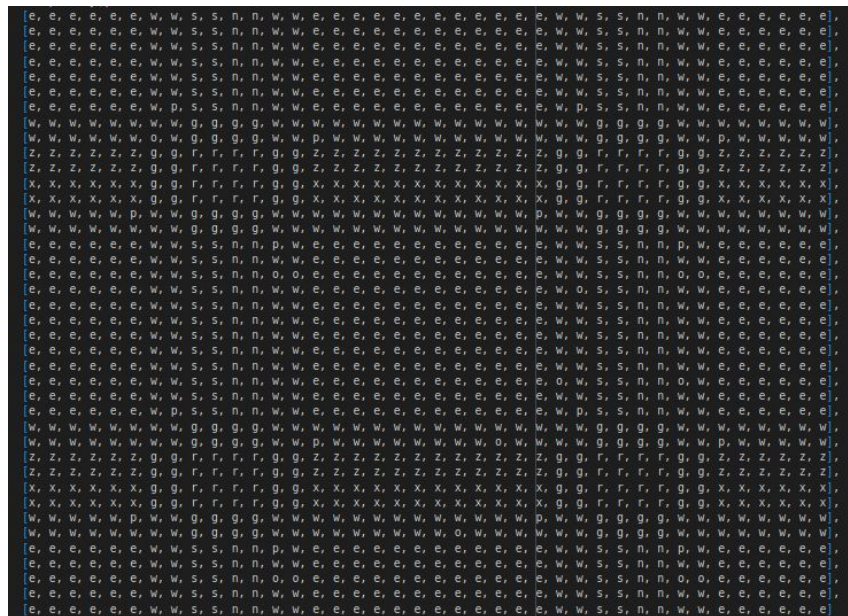
100, 10, 3, 2,

explorer goal car

-1, -2, -3

n,s,z,x,p,b,l

11,12,13,14,98, 100, 2



Rutas

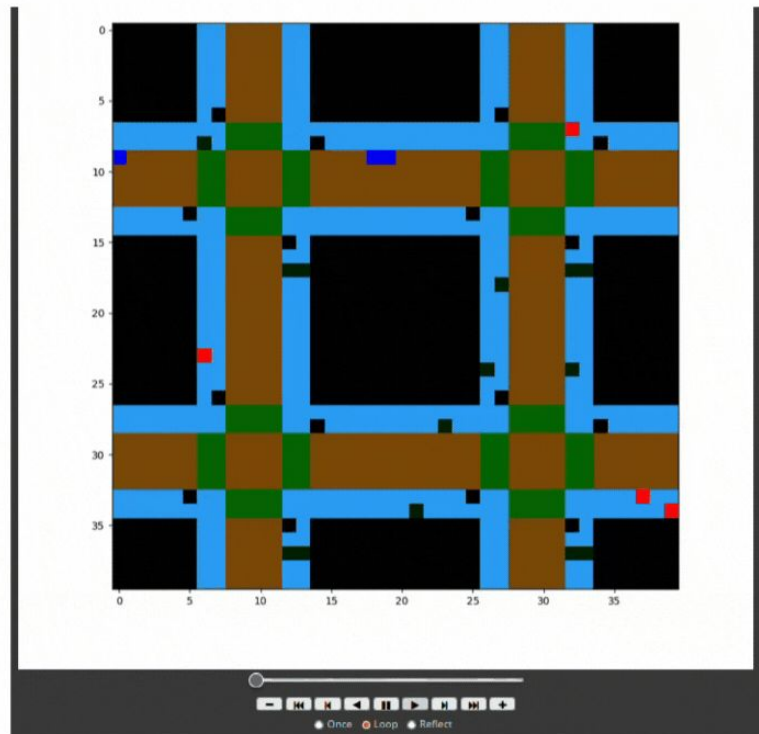
```
'epsilon': 0.1=
```

```
'alpha': 0.5=
```

```
'gamma': 0.99=
```

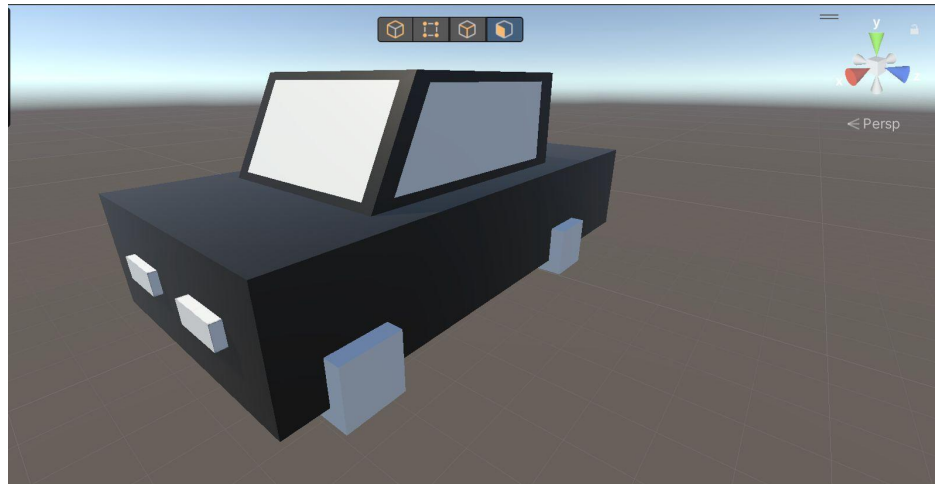
```
'episodes': 500=
```

```
'steps': 50=
```



Elementos gráficos en UNITY

Agente vehículo será representado por



Elementos gráficos en UNITY

Agente peatón



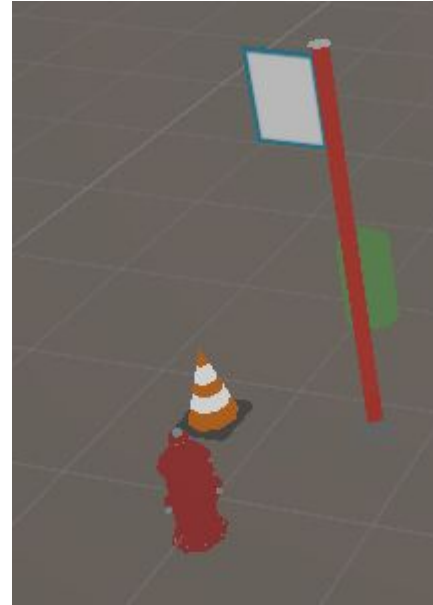
Agente Semáforo



Edificios



Obstaculos





Simulaciones planteadas y MÉTRICAS PROPUESTA

SIMULACIONES PLANTEADAS:

- Número de agentes elevado:
- Conflictos en el centro
- Coche en posición inicial en sentido contrario
- Agente que tiene que entrar a la calle

1. Múltiples agentes dentro de la simulación y su impacto en el numero de pasos
2. % De agentes que llegan a su destino vs num. de agentes
3. Tiempo promedio de espera de peatones en cruces regulados
4. Número de autos vs espera por agente por celdas ocupadas
5. %Peatones esquivando obstáculos vs numero de pasos
6. Cantidad de obstaculos vs colisiones o espera de coche por presencia de agentes