

# Simulación de Tráfico

A01028997 Agustín Pumarejo

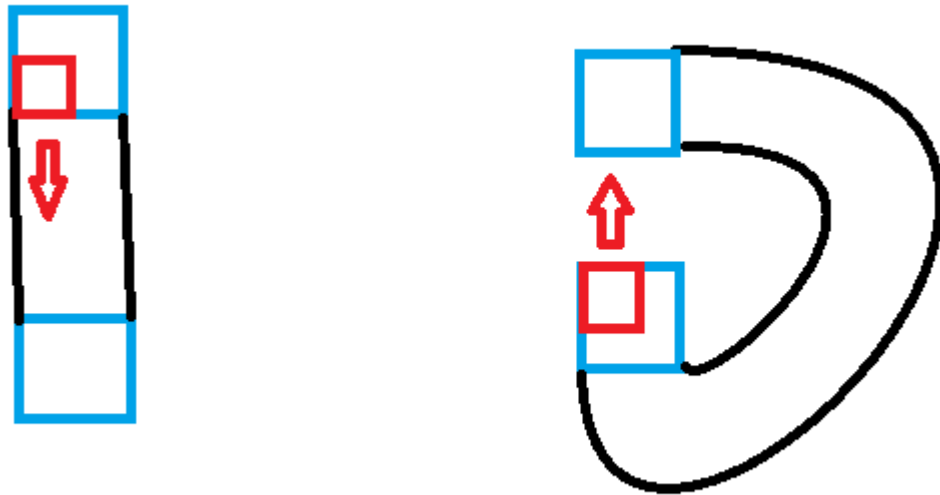
Para simular un conjunto de cuerdas, se decidió diseñar un modelo multiagente que contara con coches, carriles, intersecciones, semáforos, destinos y obstáculos. Estos agentes son suficientes para generar una simulación realista sin incluir complejidad adicional. Los coches son los agentes principales del modelo, todas las decisiones que tomamos fueron para asegurar el comportamiento adecuado de los coches. Para que la simulación fuera exitosa cada coche debía de llegar a su destino tarde o temprano (esto implicó eliminar la posibilidad de que los coches quedaran atrapados dando vueltas), los coches nunca debían chocar, nunca debían de ir en sentido contrario, desobedecer un semáforo y debían de acercarse a su destino por la ruta más rápida posible.

Las intersecciones son el esqueleto de la simulación, es el único lugar donde los coches tienen que tomar decisiones realmente importantes, por esta razón, cada intersección tenía un identificador y estaba relacionada con un nodo en un grafo que servía para representar la estructura de la vialidad. Utilizando el algoritmo de Floyd Warshal, logramos generar una matriz que se puede usar para obtener la ruta más corta desde un nodo hacia cualquier otro, esta matriz es una especie de Waze para los coches, cada vez que llegan a una intersección consultan hacia donde deberían de dar vuelta en ella.

Los destinos solo entran en juego cuando el coche está cruzando la intersección final, una vez que cruce la intersección el coche calcula un gradiente hacia el destino y utiliza este para acercarse lo más posible sin ir en sentido contrario. Las calles solo sirven para conectar intersecciones, mientras los coches que las transitan no hacen más que seguir su sentido (en la mayoría de los casos). Los semáforos solo detienen a los coches antes de cada intersección, en realidad todos los coches podrían transitar exitosamente aunque no hubiera semáforos (lo cual no se puede decir de conductores humanos). Los obstáculos son únicamente decorativos, los coches solo necesitan las calles como referencia para saber hacia qué casillas se pueden mover.

La ventaja más clara de nuestro modelo es el uso de un grafo para representar las calles e intersecciones, teniendo un grafo es mucho más fácil calcular flujo de tráfico y rutas. Sin un grafo no creo que sea posible calcular eficientemente una ruta de un punto a otro de la cuadrícula ya que se tendrían que recorrer todas las celdas recursivamente. Utilizar un gradiente no es tan efectivo ya que puede haber casos que un coche tiene que alejarse de la

ubicación de destino al principio de la ruta para poder llegar a este, respetando el sentido de las calles. Algunas áreas de oportunidad del modelo es que se utiliza el gradiente para navegar dentro de una intersección, esto funciona muy bien con la vialidad que tenemos pero no funcionaría si la calle que se encuentra entre dos intersecciones diera una vuelta muy grande como se muestra en el dibujo.



En el caso del dibujo el coche intentaría ir hacia arriba en la intersección a pesar de que para llegar a la próxima intersección en su ruta debe ir hacia abajo. Para eliminar esta desventaja sería útil que cada casilla de intersección tuviera la información de que calles están conectadas a ella, como si fuera la señalización del cruce. Así no importa la localización geográfica de la siguiente intersección, solo importa cómo están conectadas las intersecciones entre sí mediante calles. Cuando el coche entraría a una intersección, buscaría cuales casillas están conectadas a una calle que lleva a la siguiente intersección a la que desea llegar.

### Proceso de aprendizaje

Al principio del bloque sabía muy poco respecto a inteligencia artificial y machine learning. Todavía siento que sigo sabiendo muy poco al respecto pero creo que aprender a diseñar agentes me ha ayudado a entender las bases de la inteligencia, no solo artificial. La lección más importante que aprendí fue que la inteligencia es muy diferente a los algoritmos. Un algoritmo sigue una serie de pasos ordenados para resolver un problema. El algoritmo es muy eficiente y muy preciso pero es muy frágil, si las condiciones del problema cambian un poco o no se sigue uno solo de los pasos al pie de la letra, el algoritmo colapsa. La

inteligencia es más como una aproximación lenta pero segura, que es flexible y reacciona a su ambiente.