

Sistema Multi-Agente: Intersección con Semáforos Inteligentes

A00827038 - Fernando Doddoli Lankenau

Modelo

Situación Problema

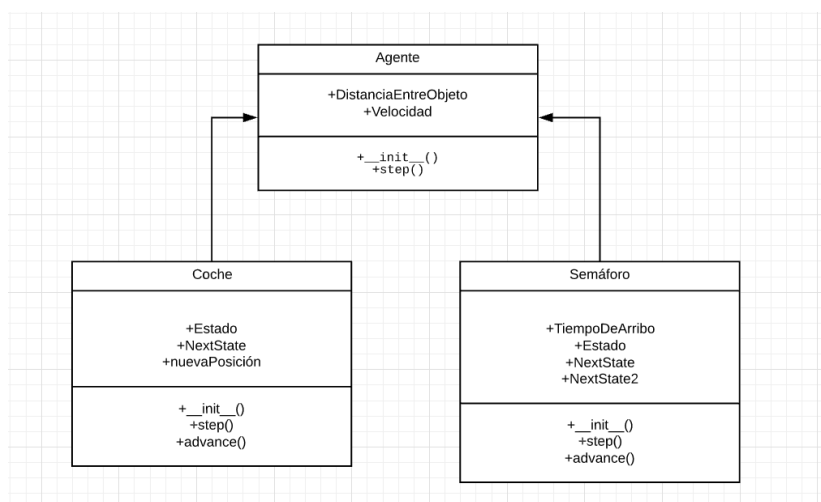
En este modelo, diseñaremos un sistema multiagente necesario para simular una intersección controlada por señales de semáforos inteligentes. El objetivo de esto es concretizar la idea y funcionamiento del sistema para después codificarlo fácilmente.

El problema nos dice que las siguientes tres condiciones son necesarias para modelar el sistema. Mientras no haya un vehículo cercano, el semáforo estará en luz amarilla. Cuando un vehículo se acerque a la intersección, enviará un mensaje con el tiempo estimado de arribo. El semáforo dará luz verde al semáforo más cercano y establecerá un programa de luces a partir de ese punto para el resto de los vehículos.

En base a lo anterior, el sistema tendrá dos diferentes tipos de agentes: coches y semáforos. Estos agentes serán agentes reactivos simples y la interacción entre ellos será cooperativa. Los agentes coche serán benevolentes entre ellos porque su objetivo es llegar de un punto A a un punto B sin chocar y en el menor tiempo posible. En efecto, para lograr esto es necesario que los coches respeten a los demás y mantengan una distancia entre ellos. Por el otro lado, los agentes semáforos también serán benevolentes entre ellos porque su objetivo común es reducir el tráfico en el sistema. Consecuentemente, tienen que trabajar juntos para lograr esto.

Los protocolos de comunicación es la información que se necesita pasar entre los diferentes tipos de agentes para modelar el sistema correctamente. En efecto, para modelar esta situación exitosamente, los semáforos inteligentes pasarán su estado (tipo de luz actual), y el tiempo de arribo entre ellos. Por el otro lado, los coches se pasarán la distanciaEntreObjeto y su estado entre ellos.

A continuación, se muestra gráficamente el funcionamiento de los diferentes tipos de agentes involucrados a través del uso de diagramas UML de clase, y flujo.



Diagramas de Clase para los Agentes Involucrados

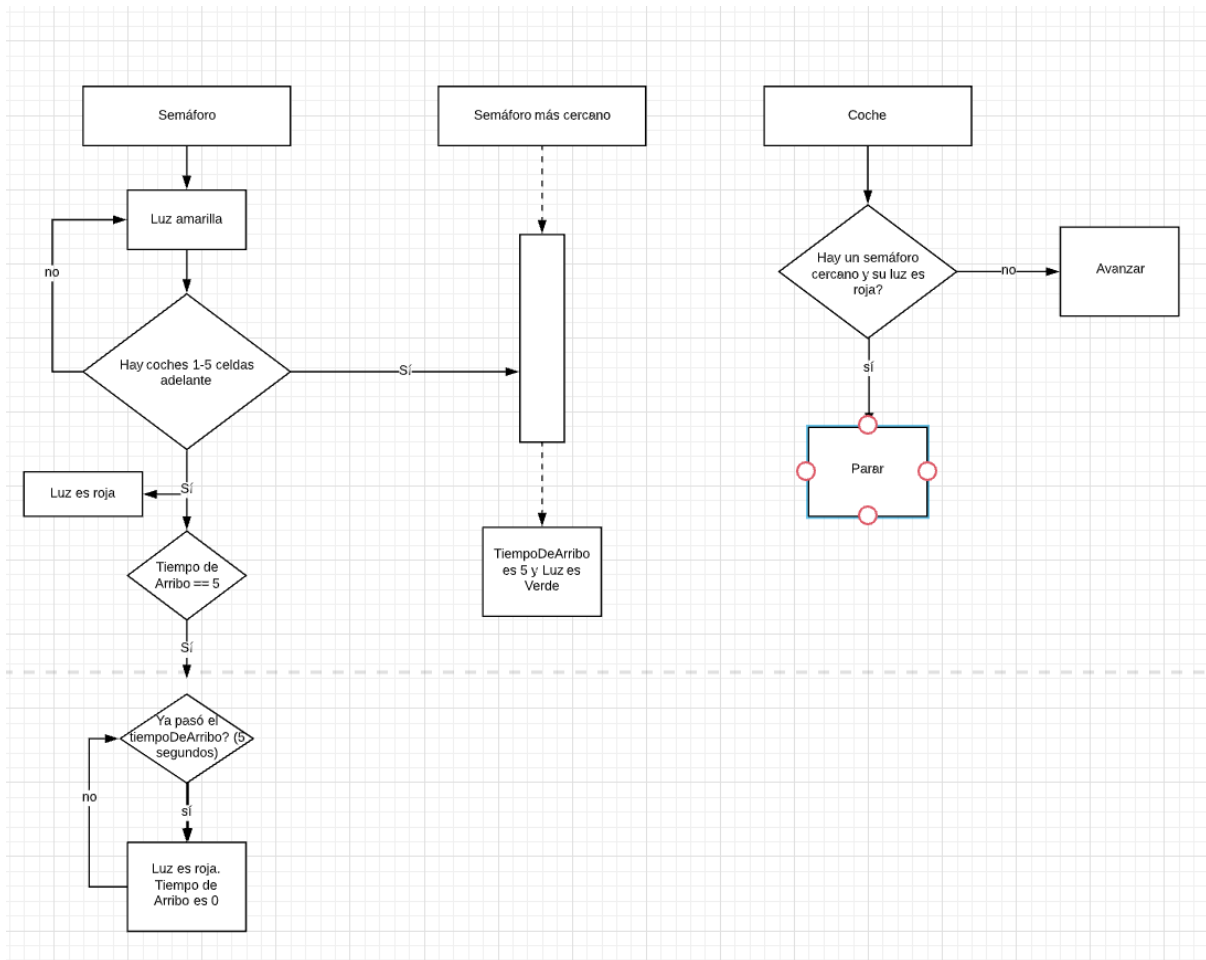


Diagrama de Flujo

Funcionamiento del Programa

En base a lo anterior y al diagrama de flujo específicamente, el funcionamiento de los agentes involucrados en el programa será el siguiente:

Semáforos

El semáforo tiene una variable de estado que puede tener los siguientes valores: 1, 2, y 3, y que representan las luces verdes, amarillas y rojas, respectivamente.

El semáforo también tiene una variable de tiempoEstimadoArribo, que mide el tiempo estimado para que un carro llegue del grid 5 al semáforo. La distancia siempre es igual porque el semáforo calcula el TiempoDeArribo cuando nota que hay un carro 5 celdas adelante. Del mismo modo, como todos los carros tienen la misma velocidad, este valor es constante y se puede calcular utilizando la siguiente fórmula $\text{TiempoDeArribo} = \text{Distancia} / \text{Velocidad} = 5/1 = 5$.

Función de Init:

self.estado = 2

self.tiempoDeArribo = 0

self.nextState = None

self.nextState2 = None

self.nextTiempoDeArribo = None

self.nextTiempoDeArribo2 = None

//nextState representa el futuro estado de este semáforo.

//nextState2 representa el futuro estado del semáforo más cercano.

Función de step:

- self.nextState = self.estado
- self.nextTiempoDeArribo = self.TiempoDeArribo
- Si hay carros 1-5 grids adelante
 - self.nextTiempoDeArribo2 = 5
 - self.nextState2 = 1.
 - self.nextState = 3
- Si self.nextTiempoDeArribo == 5
 - Si ya pasaron 5 segundos
 - self.nextState = 3
 - self.nextTiempoDeArribo = 0

Función advance:

- self.estado = self.nextState
- self.TiempoDeArribo = self.nextTiempoDeArribo
- otroSem.estado = self.nextState2
- otroSem.TiempoDeArribo = self.nextTiempoDeArribo2

Coche

El coche tiene una variable de estado que puede tener los siguientes valores: 1 y 2, y que representan avanzar o parar, respectivamente. Dependiendo si hay un semáforo enfrente y el tipo de luz es roja.

Además, el coche hereda del modelo las variables de distanciaEntreObjeto y velocidad. La distanciaEntreObjeto representa la distancia que un coche tiene que mantener con un coche o semáforo enfrente. Finalmente, la velocidad representa la velocidad del coche, es decir, en cuantas celdas se mueve por segundo. Utilizaremos un valor constante de 1 celda/segundo.

Función Init:

```
self.estado = 1
```

```
self.nextState = None
```

```
self.nextPosition = None
```

Función de step:

```
self.nextState = self.estado
```

```
nuevaPosición = una celda adelante
```

- Si hay un semáforo distanciaEntreObjeto adelante y su luz es roja
 - self.nextState = 2
- Else
 - self.nextState = 1
 - self.nextPosition = nuevaPosición

Función advance:

```
self.estado = selfNextState
```

```
self.pos = nuevaPosición
```