

Trabajo de Fin de Grado

Grado en Ingeniería Informática

Simulación y gestión del tráfico en una autopista en situaciones de congestión.

*Traffic simulation and management in a highway during traffic
congestion situations*

Alejandro Lorenzo Dávila

La Laguna, 2 de septiembre de 2019

D. **Rafael Arnay del Arco**, con N.I.F. 78.569.591-G Profesor Ayudante Doctor de Universidad adscrito al Departamento de Ingeniería Informática y de Sistemas de la Universidad de La Laguna, como tutor

D. **Iván Castilla Rodríguez**, con N.I.F. 78.565.451-G Profesor Ayudante Doctor de Universidad adscrito al Departamento de Ingeniería Informática y de Sistemas de la Universidad de La Laguna, como cotutor

C E R T I F I C A (N)

Que la presente memoria titulada:

“Simulación y gestión del tráfico en una autopista en situaciones de congestión.”

ha sido realizada bajo su dirección por D. **Alejandro Lorenzo Dávila**,
con N.I.F. 79.074.179-H.

Y para que así conste, en cumplimiento de la legislación vigente y a los efectos oportunos firman la presente en La Laguna a 2 de septiembre de 2019

Agradecimientos

A mis padres, a mi abuela, a mis hermanos, a mi pareja, a mis amigos, y a todos aquellos que me han acompañado y apoyado durante estos 4 años de carrera, que estuvieron ahí antes y durante, y que espero que me sigan acompañando el resto del camino.

A Rafael Arnay y a Iván Castilla, por apoyarme y aconsejarme como tutores durante el desarrollo de este TFG.

Al Cabildo, por su colaboración al permitirnos acceder a los datos que se han usado en este proyecto.

A todos mis compañeros de universidad, y al profesorado, por permitirme avanzar y crecer, no solo como informático, sino como persona.

Licencia



© Esta obra está bajo una licencia de Creative Commons Reconocimiento-CompartirIgual 4.0 Internacional.

Resumen

El objetivo de este trabajo ha sido la realización de una simulación de la autopista TF-5, situada en la isla de Tenerife, que sirva como base para desarrollos posteriores. Además, se ha procedido a analizar el comportamiento de la circulación de la misma, así como a buscar una solución que permita mejorar el tráfico de esta, minimizando el impacto social, económico y natural de la solución propuesta.

Como prueba del desarrollo de ésta simulación, la solución planteada en el trabajo es la inclusión de semáforos inteligentes en las entradas de la autopista, así como en los carriles de baja velocidad de la misma. Esto genera dos situaciones: el modelo actual, que es una simulación fidedigna del estado de la autopista hoy en día, sin semáforos; y una simulación que contiene los semáforos nombrados. Por tanto, se ha procedido a establecer, optimizar y comparar los resultados de estas simulaciones, a fin de determinar la eficacia de la solución propuesta.

Este trabajo incluye como apéndice la solución a errores propios de la librería “Road Traffic Library” perteneciente a la herramienta AnyLogic, puesto que la librería no posee documentación asociada a los mismos.

Palabras clave: Simulación, TF-5, AnyLogic, Semáforos, Semáforos Inteligentes, Control de Errores, Control Point Error, Road Traffic Library

Abstract

The objective of this project has been the realization of a simulation of the highway TF-5, located on the island of Tenerife, which serves as the basis for further developments. In addition, it has proceeded to analyse the behaviour of the circulation of the same, as well as to look for a solution that allows to improve its traffic, minimizing the social, economic and natural impact of the proposed solution.

As proof of the development of this simulation, the solution proposed at work is the inclusion of intelligent traffic lights at the entrances of the highway, as well as in the low-speed lanes of the highway. This generates two situations: the current model, which is a reliable simulation of the state of the highway today, without traffic lights; and a simulation containing the named traffic lights. Therefore, the results of these simulations have been established, optimized and compared, in order to determine the effectiveness of the proposed solution.

This work includes as an appendix the solution to errors of the “Road Traffic Library” belonging to the AnyLogic tool, because the library does not have documentation associated with them.

Keywords: Simulation, TF-5, AnyLogic, Traffic Lights, Intelligent Traffic Lights, Error Control, Control Point Error, Road Traffic Library

Índice general

Capítulo 1 Introducción	1
1.1 Motivación.....	1
1.2 Antecedentes.....	1
1.2.1 Descripción de la autopista.....	1
1.2.2 Problemática.....	2
1.2.3 Soluciones actuales.....	2
1.3 Origen de los datos.....	2
Capítulo 2 Desarrollo	4
2.1 Concepto básico.....	4
2.2 Herramientas.....	4
2.2.1 AnyLogic.....	4
2.2.2 Photoshop.....	4
2.2.3 Excel.....	4
2.2.4 Python.....	5
2.2.5 Git.....	5
2.3 Consideraciones previas.....	5
2.4 Primeros pasos.....	5
2.4.1 Planteamiento.....	5
2.5 Diseño de la autopista.....	6
2.5.1 Elementos disponibles / Herramientas.....	6
2.5.2 Planteamientos iniciales y Evolución.....	6
Primer Planteamiento.....	6
Segundo Planteamiento.....	7
Tercer Planteamiento.....	7
Cuarto Planteamiento - Final.....	8
2.6 Lógica.....	9
2.6.1 Flujo de agentes.....	9
2.6.2 Cálculo de las probabilidades por salidas.....	12
2.6.3 Selección de días.....	13
2.6.4 Selección de horas.....	14
2.6.5 Velocidad.....	15
2.6.6 Cálculo de las tasas por entradas.....	15
A partir de la MatrizOD.....	15

A partir de los puntos de medición.....	15
2.7 Simulación inicial	18
2.8 Semáforos	21
2.8.1 Motivación de la solución y justificación	21
2.8.2 Diseño simple.....	22
2.8.3 Optimización de parámetros.....	24
2.8.4 Semáforos Inteligentes	27
Capítulo 3 Resultados	30
3.1 Datos obtenidos.....	30
Simulación simple	30
Simulación con semáforos sin optimizar	30
Simulación con semáforos optimizados.....	30
Simulación con semáforos inteligentes.....	30
3.2 Comparativa.....	32
Simulación con semáforos sin optimizar	32
Simulación con semáforos optimizados.....	32
Simulación con semáforos inteligentes.....	32
Capítulo 4 Conclusiones y líneas futuras	33
Capítulo 5 Summary and Conclusions.....	36
Capítulo 6 Presupuesto	39
Capítulo 7 Problemas encontrados en AnyLogic	40
7.1 Introducción	40
7.2 Control Point Error	40
7.3 Segment with length = 0	42
7.4 Initialization of road network Error.....	43
7.5 StopLine: Not existing lanes specified	45

Índice de figuras

Figura 1: Herramientas Road Traffic Library	6
Figura 2: Entradas y Salidas del Tercer Planteamiento	8
Figura 3: Mapa de la autopista TF-5 con las E/S marcadas.....	8
Figura 4: Ejemplo elementos usados para la lógica.....	10
Figura 5: Elementos de programación en AnyLogic.....	10
Figura 6: Lógica de la Simulación Simple.....	11
Figura 7: MatrizOD.....	12
Figura 8: Curvas de concurrencia horaria.....	14
Figura 9: Curvas de la concurrencia horaria de los días más saturados.....	15
Figura 10: Ejemplo poco tráfico.....	20
Figura 11: Ejemplo mucho tráfico.....	20
Figura 12: Semáforo de carril.....	21
Figura 13: Semáforo en autopista A-5	21
Figura 14: Símbolo semáforo de AnyLogic.....	22
Figura 15: Semáforo de AnyLogic.....	22
Figura 16: Configuración de un semáforo sin optimizar.....	22
Figura 17: Semáforo con parámetros.....	24
Figura 18: Optimización de parámetros 1.....	25
Figura 19: Optimización de parámetros 2.....	25
Figura 20: Semáforo con tráfico retenido.....	27
Figura 21: Ejemplo conceptual de la lógica de los semáforos inteligentes.....	27
Figura 22: Comparativa valores promedio.....	31
Figura 23: Comparativa Desviación Estándar.....	31
Figura 24: Entrada bloqueada por el ceda al paso, modelo simple.....	33
Figura 25: Entrada guiada por un semáforo, modelo inteligente.....	33
Figura 26: Registro de velocidades en 2018 en la autopista TF-5.....	34
Figura 27: Entrance blocked by yielding, simple model.....	36
Figura 28: Entrance guided by a traffic light, intelligent model.....	36
Figura 29: Registro de velocidades en 2018 en la autopista TF-5.....	37
Figura 30: Control Point Error	40
Figura 31: Consola Control Point Error	40
Figura 32: Ejemplo curva que provoca un Control Point Error.....	41
Figura 33: Curva arreglada.....	41

Figura 34: Otro ejemplo de Control Point Error.	42
Figura 35: Segment with length = 0	42
Figura 36: Initialization of road network Error.....	43
Figura 37: Intersección muy corta.....	44
Figura 38: Intersección corregida.....	44
Figura 39: StopLine Error	45
Figura 40: Ejemplo de indicador StopLine Error.....	45

Índice de tablas

Tabla 1: Probabilidades viajes entre municipios.	12
Tabla 2: Aforo 2018 por día de semana	13
<i>Tabla 3: Relación puntos de medida con entradas y salidas.....</i>	17
<i>Tabla 4: Resultados Simulación Simple.....</i>	19
<i>Tabla 5: Resultados Simulación Semáforos tradicionales.....</i>	23
<i>Tabla 6: Resultados Simulación Semáforos optimizados.</i>	26
<i>Tabla 7: Resultados Simulación Semáforos inteligentes.</i>	29
<i>Tabla 8: Resultados promedios.</i>	30
<i>Tabla 9: Desviación Estándar.....</i>	31
<i>Tabla 10: Comparativa Simulación Simple con Semáforos.</i>	32
<i>Tabla 11: Presupuesto.....</i>	39

Capítulo 1 Introducción

1.1 Motivación

Este TFG busca analizar y encontrar una solución a los problemas de tráfico que se ocasionan en la autopista TF-5. Como usuario habitual de la autopista en cuestión, encontré atractiva la temática propuesta, pues el problema es conocido desde hace años, pero en 2018 tuvo una especial repercusión mediática y, por consiguiente, una toma de conciencia por mi parte de la existencia de dicho problema.

En este TFG se desarrollará una propuesta tecnológica que, buscando minimizar el impacto social, económico y natural; sirva como mejora para los atascos que se ocasionan. Esta propuesta será observada, analizada y mejorada mediante su implantación en una simulación de dicha autopista.

La estrategia implementada en el desarrollo de este trabajo, buscará que los resultados obtenidos puedan ser extrapolables a otras autopistas. Del mismo modo, se plantea una futura ampliación de este trabajo por parte de otro alumnado, si este decide tomarlo como base para su propuesta. Partiendo de esta idea, se optará por una línea de desarrollo que permita la continuidad del proyecto.

Este proyecto es el primer paso para desarrollar una aplicación que ayude a reducir el tráfico en la TFG de manera dinámica. Como prueba de conceptos, y forma de probar la simulación desarrollada, así como base para un desarrollo futuro, se ha planteado en el TFG, el uso de semáforos para intentar reducir los atascos ocasionados en la autopista. La decisión de elegir esta solución, se basa en el intento de minimizar el impacto económico, social y natural que otras medidas ocasionan (como la inclusión de más carriles, que suponen expropiaciones, obras, millones de euros, etc); También se basa en una propuesta realizada por un experto en movilidad, Miguel Pintor.

1.2 Antecedentes

1.2.1 Descripción de la autopista

La autopista TF-5, también llamada Autopista del Norte de Tenerife, es una de las dos autopistas presentes en esta isla, conformando las dos principales vías de circulación.

Cuenta con una longitud de 61.700 metros, conectando el municipio de Santa Cruz de Tenerife con El Tanque, pasando por La Laguna, Tacoronte, El Sauzal, La Matanza, La Victoria, Santa Úrsula, La Orotava, El Puerto de la Cruz y Los Realejos.

([28°27'31"N 16°27'02"O](#))

En 2014, se completó el tramo que une Icod de los Vinos con El Tanque. En estos últimos años se han realizado labores de mantenimiento, pero no modificaciones de la misma. Actualmente, existen varios proyectos que buscan mejorar esta, entre los que destacan:

- Agregar un tercer carril por sentido (creciente y decreciente), hasta el Puerto de la Cruz.
- Conectar la autopista con la TF-1, creando el Anillo Insular.
- Agregar carriles Bus-VAO.

Todos estos proyectos, y otros más, se encuentran planteados pero ninguno se está llevando a cabo actualmente, en gran medida por la necesidad de realizar expropiaciones en los alrededores de la misma.

1.2.2 Problemática

La problemática que se encuentran los usuarios de la autopista radica en su saturación entre las 07-09 horas y las 14-17 horas. (En otras franjas la autopista tiene una afluencia normal, cumpliendo su función).

Dicho colapso provoca usualmente retrasos y accidentes entre los usuarios de la vía.

No existe, como tal, un único y claro motivante de esta saturación (más concreto que el “exceso de coches”). Algunos medios culpan a la coincidencia de horarios universitarios con los colapsos, otros a los horarios laborales, al bajo uso de los transportes públicos, etc.

Tras reuniones con Miguel Pintor, máximo responsable de Movilidad del Cabildo de Santa Cruz de Tenerife, y con Manuel Ortega, anterior Consejero Delegado de la compañía de guaguas TITSA, creemos que uno de los grandes problemas es la gestión de las entradas y salidas de la autopista. Esto sentará el enfoque de la investigación y propuesta realizada.

1.2.3 Soluciones actuales

Actualmente se están barajando varias soluciones, las principales radican en:

- El fomento del transporte público, contando con el Tranvía como conexión entre Santa Cruz y La Laguna, y la red de Guaguas por parte de Titsa.
- Reducir el número de coches mediante aplicaciones para compartir coche.
- Horarios universitarios escalonados (Planteamiento sin desarrollar).
- Zonas de aparcamiento en puntos de poca concurrencia de tráfico, con conexiones a núcleos urbanos. Los trabajadores irían a esos puntos conduciendo y posteriormente utilizarían guaguas o tranvías para ir al trabajo.

Esto evita que los trabajadores que entran más tarde compitan por buscar aparcamiento, de manera que no saldrían antes de las casas.

(Planteamiento sin desarrollar)

1.3 Origen de los datos

Los datos utilizados en el TFG fueron cedidos por parte de Miguel Pintor, extrayéndose de dos fuentes principales:

- **Matriz OD terminales móviles:** contiene los movimientos de los terminales móviles de Orange, estos datos han sido procesados para poder tener datos estadísticos lo más reales a los usuarios de las demás compañías.

La aplicación web permite filtrar los movimientos por municipios, franjas horarias, distancia de viajes, actividad, etc.

Estos datos en un primer momento eran de carácter privado pero actualmente han sido liberados. [1]

Los datos hacen referencia solo a los movimientos del mes de Octubre de 2018.

- **Estaciones permanentes:** contiene las mediciones de 12 puntos a lo largo de la autopista referentes al Aforo por horas.

Estos datos están recogidos desde 2005, aunque los más antiguos no son tan completos, por esto se han tomado solo los datos de 2018.

Los datos son accedidos mediante una consola PowerBI.

Además, se facilitó un archivo *.shp* (shape), que contiene todas las carreteras de la isla; así como, se ha accedido a *Google Maps* para la observación espacial de la autopista

Capítulo 2 Desarrollo

2.1 Concepto básico

Para poder comparar los resultados de diferentes medidas realizadas en la autopista, es necesario algún sistema que nos muestre esos resultados. Una solución poco viable sería el ensayo y error en un entorno real, como esto no funcionaría, y las malas pruebas influirían en el uso de la autopista para pruebas futuras, tiene mucho sentido realizar una simulación para poder comparar el uso de estas medidas.

Para poder realizar la simulación deberemos crear un mapa de la misma, en el que poder particionar esta por zonas que contengan las entradas y salidas.

Una vez detectadas estas zonas deberemos construir la simulación, para posteriormente definir el flujo con el que los agentes o vehículos se mueven. Este flujo deberá ser previamente estudiado y analizado.

Tras realizar todos estos pasos, la simulación quedaría lista para su uso, de manera que será el momento de realizar diferentes pruebas que nos permitan no solo comprobar el buen estado de la simulación, sino medir la eficacia de las modificaciones de la autopista al compararlas con el estado actual de esta.

2.2 Herramientas

2.2.1 AnyLogic

A la hora de realizar la simulación, se ha optado por la herramienta AnyLogic. Esta está considerada actualmente la herramienta de simulación más completa del mercado. [2]

Este *software* se ha escogido principalmente por contar con la librería *Road Traffic Library*, que permite crear agentes y definir su lógica. Además, tiene compatibilidad con las otras librerías del programa.

El *software* no es *open source*, pero tiene una versión gratuita que permite simular hasta una hora.

2.2.2 Photoshop

Debido a la naturaleza del proyecto, es necesario el procesamiento de mapas, para esto se ha optado por la herramienta de Photoshop, puesto que se contaba con conocimientos previos suficientes para un buen desarrollo de la idea.

Photoshop se trata del *software* líder de mercado para la edición de imágenes.

2.2.3 Excel

Se ha optado por Excel para la visualización y creación de gráficas, así como para mejorar el entendimiento de los datos.

Excel es una hoja de cálculo desarrollada por Microsoft que permite desarrollar cálculos y programas con datos numéricos.

2.2.4 Python

A la hora del análisis y tratamiento de los datos, he escogido Python pues posee varias librerías como *Matplotlib*, *Numpy* y *Pandas*, que simplifican la modificación masiva de los datos, así como las operaciones sobre estos.

Python se trata de un lenguaje de programación interpretado multiparadigma, actualmente es el lenguaje votado como más popular por la TIOBE. [3]

2.2.5 Git

La naturaleza del proyecto fomenta el ensayo y error, de manera que es muy recomendable el uso de un sistema de control de versiones.

Para este TFG se ha optado por usar Git de manera local, como herramienta de control de versiones.

2.3 Consideraciones previas

A la hora de desarrollar el proyecto se han tomado varias reglas o acciones a tener en cuenta:

- Puesto que no tenemos los datos del punto exacto en el que los vehículos que van del municipio A al B abandonan la autopista, se considera que cada una de las salidas del municipio tiene la misma posibilidad de ser tomada por parte del vehículo. Del mismo modo, la cantidad de coches generados por el municipio se repartirá equitativamente entre sus entradas.
- Dado que el sistema a desarrollar se aplica de igual forma tanto en sentido creciente (Santa Cruz – El Tanque), como en decreciente (El Tanque – Santa Cruz), se ha optado por solo simular uno de los dos sentidos, el decreciente.
- Para realizar el estudio se han seleccionado los 9 días con mayor colapso de tráfico de 2019. De estos días sacaremos su aforo de los puntos de medición que nos han proporcionado. Aparte, se usarán los datos medios del mes de Octubre del 2018 a la hora escogida (el origen de estos datos será la MatrizOD proporcionada por el Cabildo).

2.4 Primeros pasos

2.4.1 Planteamiento

Como se ha hecho alusión anteriormente, el problema de la autopista se fundamenta en la gran cantidad de atascos que se producen, esto quiere decir que podemos medir el problema desde dos perspectivas.

La primera sería buscar que aumente el número de coches que conseguimos enviar desde su origen a su destino, la segunda es intentar minimizar el tiempo que estos coches están en la autopista.

Ambas magnitudes tienen una relación directa, pues cuanto menor sea el tiempo que un coche pasa en la autopista, más libre estará esta y más coches podrán llegar a su destino.

El planteamiento para conseguir optimizar estas magnitudes se basa en estudiar la repercusión que

tendría el uso de semáforos en la autopista, tanto en las entradas, como en el carril más lento. Para ello, en primer lugar, se debe realizar una simulación que, a partir de los datos mencionados previamente, nos sirva de comparativa para comprobar si las soluciones propuestas son viables o no.

La simulación debe tener un conjunto de segmentos que conecten las entradas y salidas principales de la autopista. Además, cada entrada debe tener un conjunto de salidas asociadas a esta y la probabilidad de tomar las mismas.

2.5 Diseño de la autopista

2.5.1 Elementos disponibles / Herramientas

AnyLogic permite dibujar carreteras usando 5 elementos principales: *Road*, *Intersection*, *Stop Line*, *Bus Stop* y *Parking Lot*. Por las necesidades de este TFG usaremos solo los 3 primeros:

- **Road:** Permite dibujar carreteras simples mediante la colocación de puntos de ancla en el mapa. Estos segmentos de carretera pueden ser rectos o curvos.
- **Intersection:** Permite unir *Roads*, así como definir el movimiento entre carriles de dos *roads* conectados.
- **Stop Line:** Indican a los agentes reglas de circulación, como limitaciones de velocidad, *Stops* propiamente y *Cedas al Paso*.
 - Cabe aclarar que en la versión española de AnyLogic, un *Yield*, Ceda al Paso en inglés, se encuentra traducido como *Rendimiento*.

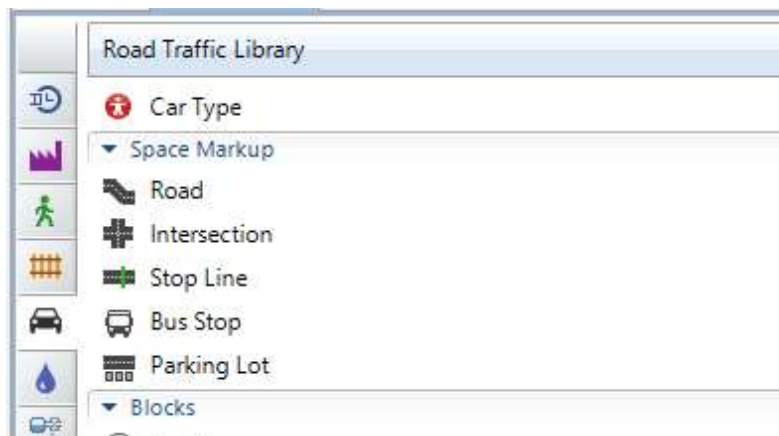


Figura 1: Herramientas Road Traffic Library

2.5.2 Planteamientos iniciales y Evolución

El diseño de la autopista ha sido realizado de 4 formas.

Primer Planteamiento

En un primer momento se optó por un diseño altamente realista que cubriera tanto los carriles crecientes como los decrecientes de la autopista. Sin embargo, este diseño solo cubriría la distancia entre Icod de los Vinos y Santa Cruz de Tenerife.

Para realizar el diseño, se contó con un archivo proporcionado por el Cabildo que contenía el mapa de carreteras de toda la isla. Este mapa se conecta con AnyLogic mediante un mapa GIS, que permite seleccionar un rectángulo que será el cargado en el modelo.

Sin embargo, este sistema se descartó debido a la imposibilidad de cargar este mapa de carreteras. El archivo en cuestión no era soportado por AnyLogic, puesto que contenía demasiada información.

Observado esto, se optó por realizar una simplificación del mapa. Tras probar diferentes softwares, encontramos que la mejor herramienta para realizar esto es *OpenJump*. Este programa permite, de manera gráfica, seleccionar qué carreteras borrar, y así “descargar” el fichero.

A pesar de haber reducido casi al mínimo el mapa de carreteras, AnyLogic tiene problemas para cargar archivos pesados, así que el uso de este fue descartado.

Segundo Planteamiento

Al no poder realizar de manera automática el mapa de carreteras, el segundo modelo de la autopista se basó en construir una simulación a partir de una imagen de *Google Maps*. Este modelo buscaría actuar de manera medianamente realista en conceptos de entradas y salidas, puesto que las limitaciones a la hora de realizar una captura de pantalla con el suficiente detalle, no permitían poder copiar la ubicación exacta de las entradas y salidas. Aun así, se colocaron con la mayor exactitud posible.

Este modelo planteaba dos grandes problemas:

El principal problema era la aparición de errores que imposibilitarían la ejecución del modelo. Estos errores carecían de información asociada al mismo, de manera que con el conocimiento actual de Java y AnyLogic, no era posible resolverlos.

El segundo problema encontrado se basaba en el desconocimiento inicial de las tasas horarias de creación de Agentes por entradas y salidas. Este problema tenía una solución sencilla que explicaré en su punto correspondiente.

La unión de los dos problemas desembocó en desechar el modelo creado.

Tercer Planteamiento

Para poder realizar una conexión directa entre los puntos de medida que se tienen, así como intentar solventar los problemas de Java que se obtenían al ejecutar la simulación, opté por simplificar las entradas y salidas, ligando estas directamente a los datos medidos.

Este modelo se desarrolló siguiendo los pasos del Segundo, con la diferencia de que solo habría una entrada y salida por sentido y segmento.

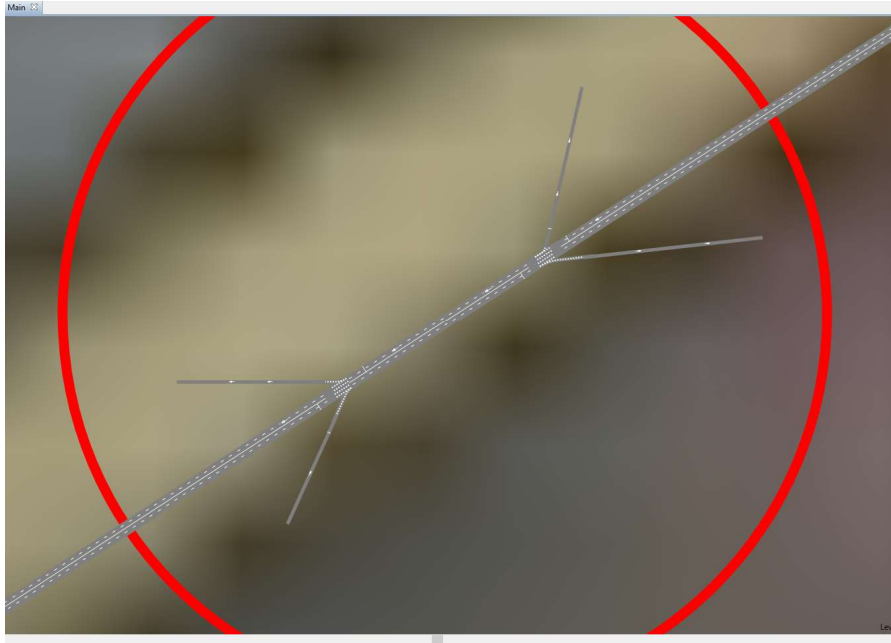


Figura 2: Entradas y Salidas del Tercer Planteamiento

Este modelo, a pesar de funcionar, fue descartado tras decidir en una reunión premiar al máximo el realismo en las entradas y salidas.

Cuarto Planteamiento - Final

El cuarto modelo buscaría ser lo más fiel posible al mundo real en lo referente a las entradas y salidas de la autopista.

Dado que esta tarea tiene una dimensionalidad superior a los modelos anteriores, se optó por solo realizar el sentido decreciente de la autopista, puesto que una vez optimizado el tráfico de un sentido, los pasos a realizar serían duplicables en el contrario.

Además, se amplió el área diseñado hasta el Tanque, barriendo toda la amplitud de la TF-5.

Para empezar a realizar el mapa de la manera más fiel posible, el primer paso fue realizar capturas de pantalla a la mayor distancia posible que permitiera simplificar el proceso de dibujo de las entradas y salidas al verse estas claramente en la captura. Esto tomó un lugar de 58 capturas que posteriormente hubo que recortar y unir de manera manual. El mapa final es el siguiente:

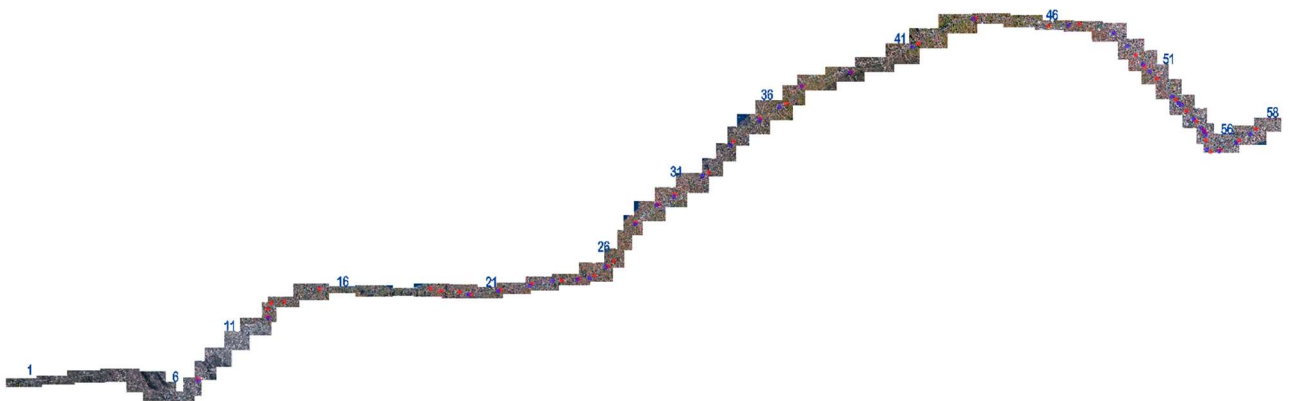


Figura 3: Mapa de la autopista TF-5 con las E/S marcadas.

El mapa ha sido escalado al 28% de calidad para poder ser incluido en el documento, el peso del archivo original es de

124MB. Se añadieron marcas referentes a las entradas y salidas para facilitar su posterior dibujo.

A lo largo del desarrollo, se apreció que el rendimiento asociado al uso de segmentos rectos es muy superior al uso de curvos, de manera que en un primer momento se diseñó la carretera usando solo segmentos rectos, pero siendo lo más fiel posible a la realidad.

Este modelo sufriría posteriormente una modificación para arreglar el *Control Point Error*, que provocaría cambiar la mayor parte de los segmentos rectos por curvos, perjudicando altamente el rendimiento, pero dando por finalizado el diseño de la autopista. Gracias a los conocimientos conseguidos por el desarrollo de los primeros 3 modelos, el desarrollo de este último fue mucho más fluido a pesar de su magnitud.

Al cambiar estos segmentos a curvos se mantuvo prácticamente todo el realismo que se tenía. Sin embargo, por limitaciones de AnyLogic, aquellas entradas que son especialmente cortas se deben ampliar para que el modelo pueda funcionar.

2.6 Lógica

2.6.1 Flujo de agentes

AnyLogic incluye varios elementos para controlar el flujo de los agentes (vehículos). Partiendo de esto, se procederá a nombrar los usados en el TFG:

- **CarSource:** Genera los vehículos. Debe estar conectado a un MoveTo para funcionar, así como indicar un camino del que generar los agentes, y una cantidad/tiempo (Ej.: 1000/hora).
- **CarDispose:** Elimina los agentes que llegan hasta él, se debe usar como nodo final del flujo de un agente.
- **MoveTo:** Define el objetivo al que debe desplazarse el vehículo, puede conectarse en serie con otros MoveTo.

AnyLogic calcula automáticamente el camino más corto entre el nodo anterior y el nodo destino especificado.

En caso de que un agente no pueda realizar una acción de la ruta calculada como, por ejemplo, si otro vehículo impide un cambio de carril, se puede especificar una alternativa a este destino no alcanzado. (Si no se especifica, el programa entero se detendrá; pues se considera un error).

- **Select Output, Select Output3 y Select Output5:** Permiten tomar decisiones a la hora de escoger uno o más caminos. Puede estar basado en probabilidad o en programación.
- **Connectors:** Son las líneas que indican gráficamente las conexiones entre los elementos.

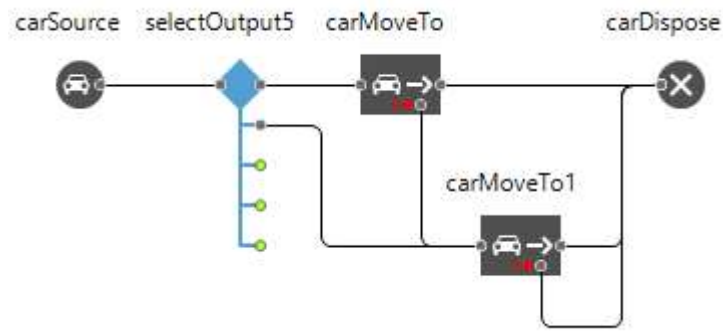


Figura 4: Ejemplo elementos usados para la lógica.

En la imagen se puede ver cómo los agentes son generados en *carSource*, así como deben tomar una decisión en *selectOutput5*, el cual tiene tres salidas aún disponibles. Luego, en caso de escoger ir a *carMoveTo*, pueden llegar al destino y ser destruidos; o en caso de no poder alcanzar el destino, son redirigidos a *carMoveTo1*, en este, si llegan al destino son destruidos al finalizar y, por contrario, si no son capaces de alcanzar el destino, son directamente destruidos para evitar errores.

- **Parameter:** Son un tipo de variables pensadas para no ser modificadas salvo en la pantalla de espera de la simulación. Su objetivo es definir características del modelo. Pueden tomar cualquier tipo de variable o clase de Java.
- **Variable:** Son cualquier tipo de variable o clase de Java. Los agentes pueden contenerlas y modificarlas. Están pensadas para poder ser modificadas durante la simulación si se desea.
- **Functions:** Son funciones de Java. Pueden recibir variables de entrada, así como devolver valores.
- **ExcelFile:** Se trata de un objeto que contiene un archivo Excel, y permite usar una serie de funciones para manejarlo, entre los que se incluyen cargar un documento diferente, leerlo y escribirlo.

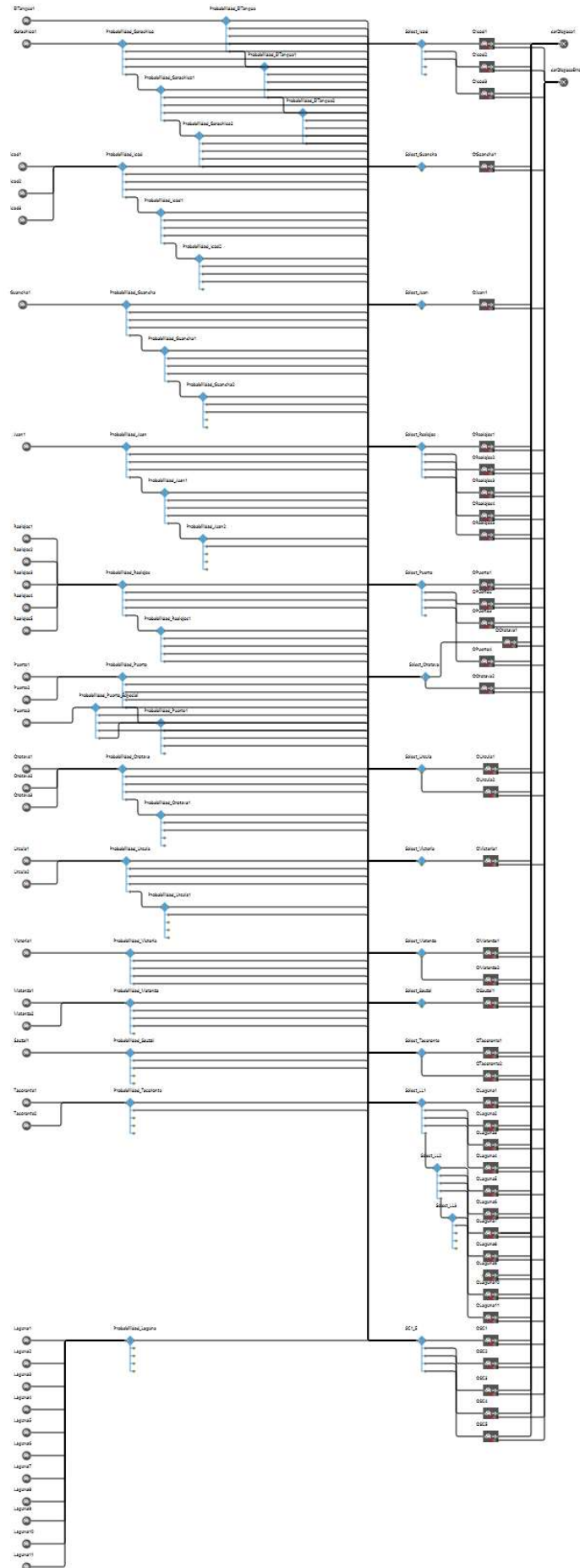


Figura 5: Elementos de programación en AnyLogic.

Figura 6: Lógica de la Simulación Simple.

En la siguiente imagen, se observa la combinación de parte de estos elementos para crear la lógica seguida por los vehículos.

A la izquierda, están las entradas (*CarSource*); en el centro, la probabilidad de tomar las salidas (*Select OutputN*); al lado, las salidas (*MoveTo*) y a la derecha, la eliminación de los agentes que alcanzan o no el destino (*CarDispose*).



2.6.2 Cálculo de las probabilidades por salidas

A la hora de escoger a qué salida debe ir un agente, este debe tomar una decisión probabilística.

Este valor, de 0 a 1, repartido entre las 5 salidas de los *Select Output5* usados, se calcula mediante la probabilidad de destino de los vehículos de la *MatrizOD* del Cabildo, que contiene los movimientos medios por horas del mes de Octubre.



Figura 7: MatrizOD

Movimientos Generados desde el municipio de El Tanque, a las 7 am, de más de 1000 metros en actividades frecuentes (casa, trabajo, etc.).

A partir de la recopilación de estos valores, obtenemos la probabilidad Origen-Destino de cada municipio.

Origen - v /	El Tanque	Garachico	Icod de los V	La Guancha	San Juan de	Los Realejos	Puerto de la	La Orotava	Santa Ursula	La Victoria d	La Matanza c	El Sauzal	Tacoronte	La Laguna	Santa Cruz d
El Tanque	0	0	0,51383875	0,0589651	0,02045728	0,13237064	0,09987966	0,05415162	0,03008424	0,00722022	0	0,00481348	0,00240674	0,04211793	0,03369434
Garachico	0	0	0,61700632	0,07097681	0,02740689	0,09627547	0,07449051	0,03302881	0,00843289	0	0,0035137	0,00281096	0,03654252	0,02951511	
Icod de los V	0	0	0	0,28280702	0,07625731	0,15695906	0,16818713	0,08912281	0,02526316	0,00023392	0,00233918	0,01637427	0,0128655	0,10081871	0,06877193
La Guancha	0	0	0	0	0,10499398	0,19133574	0,13658243	0,0944645	0,13327316	0,01083032	0,01624549	0,05445247	0,09747292	0,09927798	0,061071
San Juan de	0	0	0	0	0	0,27043391	0,33198789	0,09283552	0,08879919	0,00807265	0,01614531	0,02320888	0,03632694	0,09788093	0,03430878
Los Realejos	0	0	0	0	0	0	0,43944151	0,31064199	0,08517603	0,00093527	0,00474314	0,00688089	0,01255929	0,09011958	0,0495023
Puerto de la	0	0	0	0	0	0	0	0,54912135	0,17981182	0,0069185	0,01196901	0,01542826	0,01840321	0,13657119	0,08177667
La Orotava	0	0	0	0	0	0	0	0	0,32639885	0,017934	0,0143472	0,03969393	0,04244381	0,33608321	0,223099
Santa Ursula	0	0	0	0	0	0	0	0	0	0,15345413	0,16912042	0,10268026	0,10796527	0,29331823	0,17346168
La Victoria d	0	0	0	0	0	0	0	0	0	0	0,17263844	0,14006515	0,07491857	0,45439739	0,15798046
La Matanza c	0	0	0	0	0	0	0	0	0	0	0	0	0,34482759	0,14655172	0,29693487
El Sauzal	0	0	0	0	0	0	0	0	0	0	0	0	0	0,37591703	0,43308879
Tacoronte	0	0	0	0	0	0	0	0	0	0	0	0	0	0,76982592	0,23017408
La Laguna	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
Santa Cruz d	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Tabla 1: Probabilidades viajes entre municipios.

La probabilidad ha sido calculada en dos condiciones, la primera es la expresada anteriormente, y la segunda variando el rango horario entre las 7 y las 10 am.

2.6.3 Selección de días

Analizando los datos localizados en la consola *PowerBI* a la que se dio acceso, se observó el aforo por día de semana en el 2018.

	Total	> Media	> 75%	>90%	>95%	>99%
Lunes	1810948	27	19	14	9	2
Martes	1732160	27	8	1	1	0
Miércoles	1742038	33	13	0	0	0
Jueves	1739853	32	11	3	0	0
Viernes	1807023	44	27	11	5	1
Sábado	1644223	18	12	7	3	0
Domingo	1506814	1	1	0	0	0

Tabla 2: Aforo 2018 por día de semana

Esta tabla muestra como los lunes y los viernes del 2018 tienen una afluencia de tráfico significativamente mayor que el resto de días de la semana.

Además, observamos que existen 9 lunes que superan el percentil 95 de afluencia y 5 viernes.

Debido a que el dato que buscamos medir son las horas pico, creemos que los lunes son mejor indicadores que los viernes, pues se entiende que en estos últimos se juntan los transportes por vacaciones.

Los días que cumplen este percentil 95 son: 01/01, 29/01, 05/03, 26/03, 23/04, 18/06, 13/08, 29/10 y 03/12.

2.6.4 Selección de horas

La siguiente gráfica muestra el número total de coches por hora en el 2018.

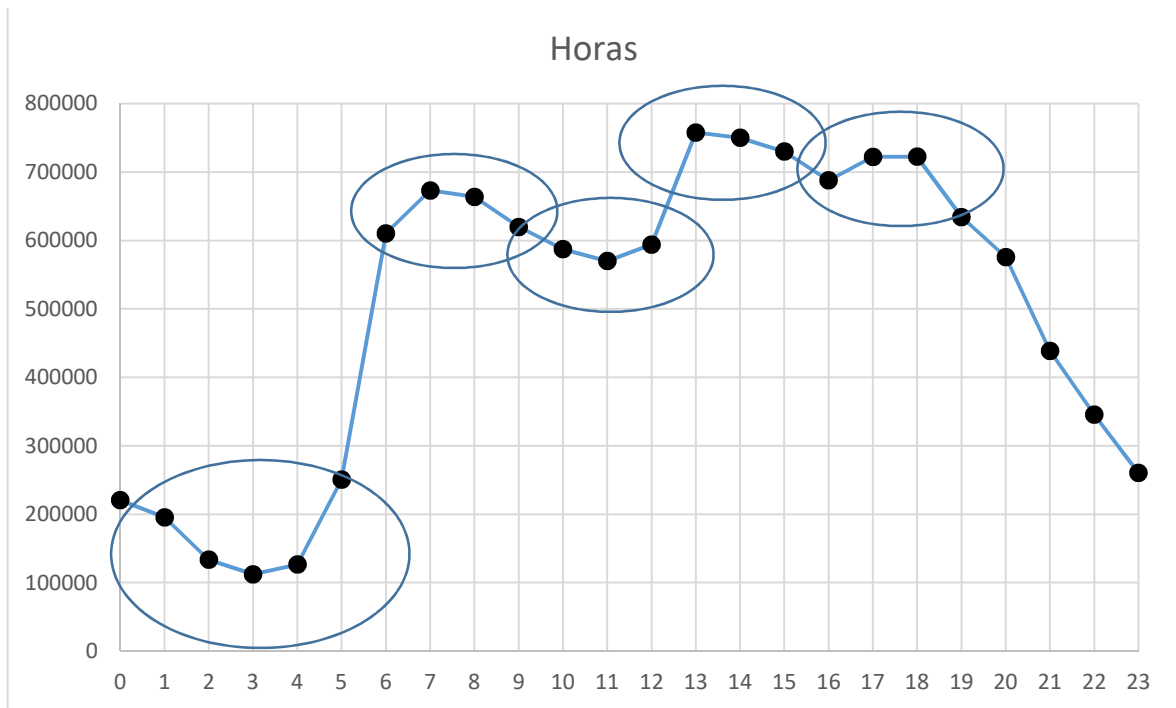


Figura 8: Curvas de concurrencia horaria.

Se observa que el primer pico ocurre entre las 5 y las 6 am, y dura hasta las 8-9 am. El siguiente pico empieza entre las 12 y las 13 horas, el cual dura hasta las 18 horas con una caída a las 16 horas.

Tomando los valores máximos y mínimos, encontramos las 7 horas como un pico que, perteneciendo a los máximos, resulta intermedio y adecuado para el TFG, puesto que además coincide con la entrada de los conductores al trabajo.

En cumplimiento de lo anterior, los días seleccionados tienen la siguiente disposición horaria:

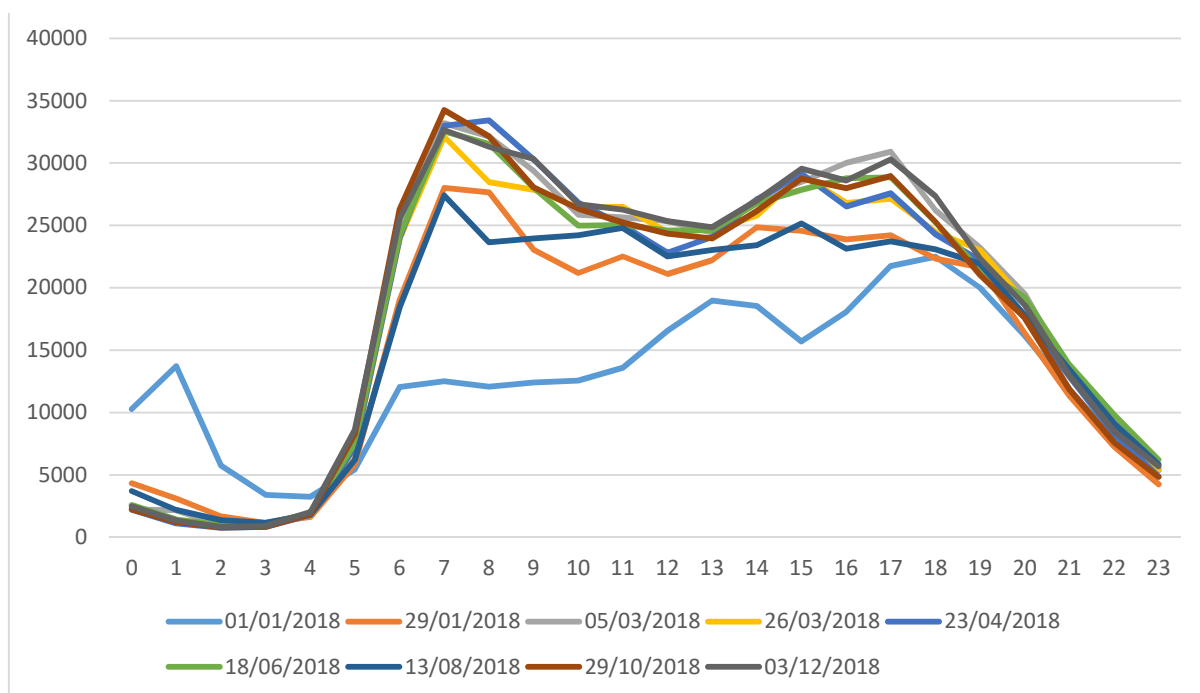


Figura 9: Curvas de la concurrencia horaria de los días más saturados.

Podemos observar que, salvo el 1 de Enero, todos se comportan de manera similar.

2.6.5 Velocidad

Para la velocidad media de los agentes, se ha calculado la velocidad media de los coches en 2018 en la autopista, siendo esta de 84,09 km/h, que es la velocidad que será tomada como preferente por los vehículos, pudiendo estos variarla a decisión propia.

2.6.6 Cálculo de las tasas por entradas

A la hora de establecer tasas por entradas, podemos distinguir entre dos tipos:

A partir de la MatrizOD

Esta primera forma, aprovecha los movimientos de los municipios observados en la MatrizOD para hallar, casi directamente, las tasas correspondientes a cada entrada.

A partir de los puntos de medición

Una vez seleccionados los días a simular, nos encontramos con 11 puntos de medición, 14 municipios y en estos un total de 40 entradas y 39 salidas con la siguiente disposición:

	Salidas Antes	Entradas Antes
El Amparo		Tanque Garachico 1
El Pedregal	Icod 1	Icod 1
BCO Patronato	Icod 2 Icod 3 Guancha 1 Juan 1 Realejos 1 Realejos 2 Realejos 3	Icod 2 Icod 3 Guancha 1 Juan 1 Realejos 1 Realejos 2 Realejos 3
La Orotava	Realejos 4 Realejos 5 Puerto 1 Puerto 2 Puerto 3	Realejos 4 Realejos 5 Puerto 1 Puerto 2
La Matanza	Orotava 1 Puerto 4 Orotava 2 Úrsula 1 Úrsula 2 Victoria 1 Matanza 1	Orotava 1 Orotava 2 Puerto 3 Orotava 3 Úrsula 1 Úrsula 2 Victoria 1 Matanza 1
Ermita	Matanza 2 Sauzal 1 Tacoronte 1	Matanza 2 Sauzal 1 Tacoronte 1
Bomberos	Tacoronte 2 LL 1	Tacoronte 2 LL 1 LL 2
Padre	LL 2 LL 3	LL 3
Campus	LL 4 LL 5 LL 6	LL 4 LL 5 LL 6
Hospital	LL 7 LL 8 LL 9 LL 10 LL 11 SC 1	LL 7 LL 8 LL 9 LL 10 LL 11 SC 1
Somosierra	SC 2 SC 3 SC 4 SC 5	SC 2

Tabla 3: Relación puntos de medida con entradas y salidas.

Con esta tabla que hemos generado, y sabiendo el aforo que toma cada punto de medida, podemos intentar hallar los valores individuales de cada municipio.

A los puntos de medida los llamaremos a partir de ahora Q .

Sabemos que el valor Q será igual al número de coches que pasen por el punto en una hora. Estos coches son originados en un municipio, y tienen como destino otro. Este movimiento de coches se genera en base a una probabilidad asociada a cada viaje (2.5.2). De manera que podemos expresar ese movimiento de coches como la probabilidad de viajar del punto A al B por el número total de coches que salen de A:

$$P_B^A N^A$$

Sin embargo hay municipios de llegada que son cortados por el punto Q , de manera que algunos coches que viajan de A a B son enviados al primer punto de B, y otros al segundo, quedando posteriores al punto Q , que son contados, mientras que los del primer punto no. Esto supone modificar la probabilidad, pues hemos presupuesto que cada salida tiene la misma probabilidad al no tener los datos exactos:

$$\frac{\text{Puntos B después de Q}}{\text{Total de puntos de B}} P_B^A N^A$$

Del mismo modo que existen puntos de destino (B) que son cortados por Q , es posible encontrar que el punto A sea cortado por el punto Q , pues envía coches antes y después del lugar de medida:

$$\frac{\text{Puntos A antes de Q}}{\text{Total de puntos de A}} \frac{\text{Puntos B después de Q}}{\text{Total de puntos de B}} P_B^A N^A$$

Con esta ecuación se obtiene la parte de Q que pertenece a los viajes de A a B. De manera que el valor total de Q serán todos los viajes que se produzcan entre salidas anteriores a este punto a salidas posteriores a este punto.

$$Q = \sum_{i,j} \frac{\text{Puntos } i \text{ antes de Q}}{\text{Total de puntos de } i} \frac{\text{Puntos } j \text{ después de Q}}{\text{Total de puntos de } j} P_j^i N^i$$

Esta ecuación se puede aplicar a todos los viajes posibles, pues si el viaje se origina después del punto Q , éste tendrá 0 puntos antes del punto, así que su aporte a Q será 0.

Teniendo esto en cuenta, se forman 11 ecuaciones, pero disponemos de 14 municipios o incógnitas. Por ello es necesario incluir 3 ecuaciones que permitan resolver el sistema.

Estas ecuaciones consisten en la relación que tienen dos municipios entre sí. Si observamos por ejemplo la primera ecuación, El Amparo, tomaría la siguiente forma una vez simplificada:

$$\text{Amparo} = P_{\text{Resto de Municipios}}^{\text{El Tanque}} N^{\text{El Tanque}} + P_{\text{Resto de Municipios}}^{\text{Garachico}} N^{\text{Garachico}}$$

Como se puede observar, las probabilidades y el valor de el Amparo son conocidos, así que quedan dos variables que son la cantidad de coches que se originan en el Tanque, y la cantidad originada en

Garachico. Sin embargo, gracias a la MatrizOD, podemos aproximar que la cantidad de coches generados en Garachico, son 1,512768131 veces los coches generados en el Tanque:

$$N^{Garachico} = 1,512768131 * N^{El\ Tanque}$$

Si unimos ambas ecuaciones:

$$N^{El\ Tanque} = \frac{Amparo}{P_{Resto\ de\ Municipios}^{El\ Tanque} + 1,512768131 * P_{Resto\ de\ Municipios}^{Garachico}}$$

Con esto somos capaces de averiguar los valores N de El Tanque, Garachico e Icod, si además añadimos la relación entre La Guancha y San Juan de la Rambla, y La Guancha y Los Realejos, el sistema de ecuaciones queda completo y podemos hallar las tasas por municipios.

2.7 Simulación inicial

Una vez definida la lógica y cargadas las tasas, se procederán a realizar una serie de simulaciones de una hora de duración (3599 segundos) para comprobar el estado del tráfico con los datos iniciales.

Cada simulación usa una semilla aleatoria.

A continuación, se muestra la tabla con las 10 ejecuciones de la simulación para los datos procedentes de la MatrizOD.

time_in	time_out	cars_created	cars_dispose	cars_error_dispose	source_name	time_model	tag
21:59:02	22:06:25	9795	4820	1	MatrizOD	3599	Simple 2
22:07:09	22:14:22	9851	4929	0	MatrizOD	3599	Simple 2
22:35:49	22:43:33	9715	4895	0	MatrizOD	3599	Simple 2
22:45:34	22:52:12	9744	4748	1	MatrizOD	3599	Simple 2
23:36:02	23:47:43	9719	4875	0	MatrizOD	3599	Simple 2
23:36:00	23:47:44	9746	4741	0	MatrizOD	3599	Simple 2
23:36:07	23:48:03	9713	4845	0	MatrizOD	3599	Simple 2
23:36:09	23:48:07	9694	4834	1	MatrizOD	3599	Simple 2
23:36:04	23:48:16	9755	4758	0	MatrizOD	3599	Simple 2
23:49:18	0:16:38	9658	4749	2	MatrizOD	3599	Simple 2
23:49:21	0:17:35	9807	4902	0	MatrizOD	3599	Simple 2
23:49:36	0:17:48	9704	4826	0	MatrizOD	3599	Simple 2
23:49:28	0:17:55	9755	4750	0	MatrizOD	3599	Simple 2
23:49:25	0:17:57	9803	4796	0	MatrizOD	3599	Simple 2
23:49:30	0:17:58	9663	4729	0	MatrizOD	3599	Simple 2
23:49:33	0:18:20	9713	4865	1	MatrizOD	3599	Simple 2
23:49:39	0:18:22	9760	4811	0	MatrizOD	3599	Simple 2
23:49:42	0:18:48	9751	4806	0	MatrizOD	3599	Simple 2
0:22:28	0:29:43	9789	4869	0	MatrizOD	3599	Simple 2
0:22:30	0:29:47	9796	4836	0	MatrizOD	3599	Simple 2

Tabla 4: Resultados Simulacion Simple

Esta tabla muestra el momento en que fue ejecutada, el tiempo que tarda la simulación a máxima velocidad, el número de coches creados, así como los que llegaron al destino y los que no pudieron llegar por pasarse la salida, el origen de los datos y el tiempo de transcurrido dentro del modelo.

Podemos observar además claras zonas con una afluencia regular de tráfico, como zonas saturadas, especialmente en las entradas y salidas (ambas al mismo tiempo).



Figura 10: Ejemplo poco tráfico.

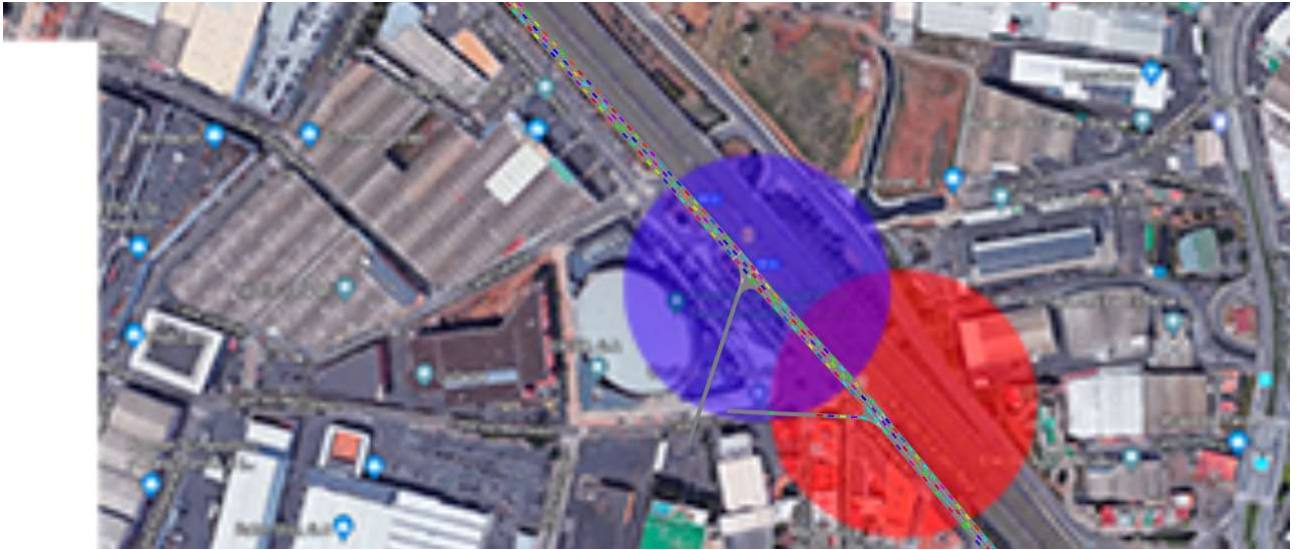


Figura 11: Ejemplo mucho tráfico.

2.8 Semáforos

2.8.1 Motivación de la solución y justificación

Como hemos podido observar en el primer capítulo, casi todas las soluciones planteadas por parte de organismos, suponen un impacto económico, social y natural muy importante, así como obras que pueden llevar desde unos meses a varios años.

Es por esto, que la solución que se propone en este trabajo gana fuerza, pues no requiere de una cantidad económica tan elevada como puede ser la incorporación de más carriles.

Del mismo modo no requiere de expropiaciones en las inmediaciones de la autopista, y el impacto natural es menor.

La idea de poner semáforos en autopistas no es algo nuevo, es habitual encontrar semáforos de carril. Estos permiten usar un carril central de una vía de doble sentido en el sentido más conveniente según las necesidades del tráfico actual.

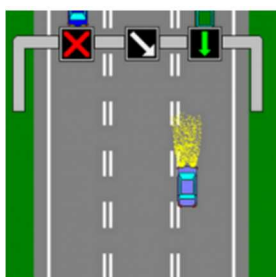


Figura 12: Semáforo de carril.

Del mismo modo, encontramos semáforos tradicionales en autopistas españolas como en el caso de la autopista A-5 (Madrid-Extremadura), que posee 3.



Figura 13: Semáforo en autopista A-5

Según un estudio realizado por el MIT [4], en carreteras de alta velocidad 55mps (~89km/h), el 90% de los conductores de la vía se detienen al encontrar un semáforo con una señal que avise de su cambio entre 79 y 90 metros del punto de parada.

El estudio además comenta que de los 5 sitios de estudio, 3 tenían una limitación de velocidad de 70mps (~123km/h), en la que la mayoría de los conductores respetaban los semáforos.

2.8.2 Diseño simple

AnyLogic incluye en la librería de tráfico un modelo estándar de semáforo. Este semáforo puede funcionar mediante 3 configuraciones:

- A través de las *StopLines* de una intersección. El semáforo afecta a todos los carriles de cada carretera a la vez, sin distinguir entre ellos. Bloquea una carretera para ceder el paso a la otra.
- A través de los carriles de las carreteras. El semáforo distingue entre cada carril. Pero todos deben entrar en la fase amarilla a la vez. Esto impedia que un carril pueda estar todo el rato en verde por ejemplo.
- A través de *StopLines* que escogamos. Las *StopLines* se pueden crear en carriles concretos, de manera que en esta configuración tenemos control total sobre los carriles y sus estados, pudiendo usar unos y otros no de una misma carretera.

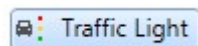


Figura 14: Símbolo semáforo de AnyLogic.



Figura 15: Semáforo de AnyLogic

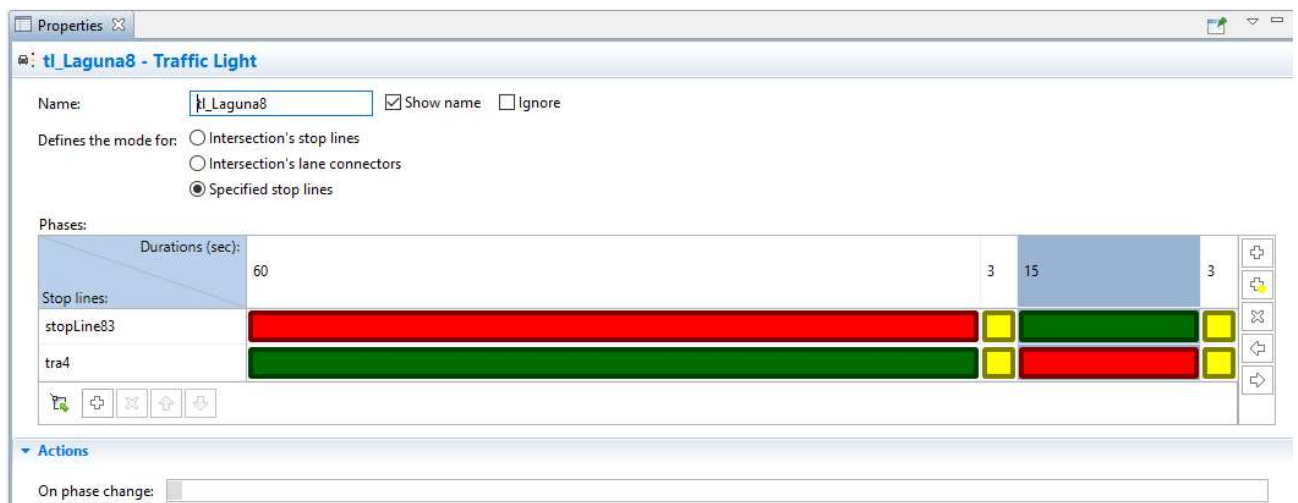


Figura 16: Configuración de un semáforo sin optimizar.

Los semáforos serán colocados en cada entrada de la autopista, a excepción de los puntos en los que la entrada suponga agregar un carril más a la misma.

Estos semáforos tendrán una configuración inicial de 60 segundos en verde para la autopista y 15 segundos para las incorporaciones, con cambios de fase de 3 segundos.

time_in	time_out	cars_created	cars_dispose	cars_error_dispose	source_name	time_model	tag
20:18:29	20:23:02	6489	4236	1	MatrizOD	3599	Semáforo
20:25:28	20:29:20	6608	4390	0	MatrizOD	3599	Semáforo
20:29:46	20:33:38	6624	4382	1	MatrizOD	3599	Semáforo
20:33:50	20:37:43	6560	4320	0	MatrizOD	3599	Semáforo
20:38:32	20:44:12	6627	4388	0	MatrizOD	3599	Semáforo
21:01:01	21:08:26	6573	4376	0	MatrizOD	3599	Semáforo
21:08:44	21:13:10	6638	4406	1	MatrizOD	3599	Semáforo
21:13:28	21:17:44	6572	4333	0	MatrizOD	3599	Semáforo
21:17:55	21:22:22	6574	4332	0	MatrizOD	3599	Semáforo
21:22:33	21:27:07	6621	4410	1	MatrizOD	3599	Semáforo
21:27:18	21:31:54	6619	4250	0	MatrizOD	3599	Semáforo
21:32:28	21:37:06	6627	4381	0	MatrizOD	3599	Semáforo
21:37:14	21:41:43	6614	4320	0	MatrizOD	3599	Semáforo
21:42:15	21:46:42	6559	4311	0	MatrizOD	3599	Semáforo
21:46:52	21:51:25	6539	4270	0	MatrizOD	3599	Semáforo
21:51:35	21:56:06	6496	4303	0	MatrizOD	3599	Semáforo
0:31:42	0:37:47	6611	4389	0	MatrizOD	3599	Semáforo
0:31:44	0:37:52	6540	4239	0	MatrizOD	3599	Semáforo
0:31:46	0:37:53	6569	4289	0	MatrizOD	3599	Semáforo
0:31:49	0:38:08	6543	4325	0	MatrizOD	3599	Semáforo

Tabla 5: Resultados Simulación Semáforos tradicionales.

Esta tabla muestra el momento en que fue ejecutada, el tiempo que tarda la simulación a máxima velocidad, el número de coches creados, así como los que llegaron al destino y los que no pudieron llegar por pasarse la salida, el origen de los datos y el tiempo de transcurrido dentro del modelo.

2.8.3 Optimización de parámetros

AnyLogic permite realizar una optimización de parámetros, de manera que se deben crear dos parámetros asociados a cada semáforo que contengan los segundos que estará en cada fase.



Figura 17: Semáforo con parámetros.

Puesto que se está usando la herramienta en su modo gratuito, la optimización de parámetros se realizará en grupos de 6.

Ya que esto resulta especialmente problemático, se ha procedido a seleccionar los semáforos que más afectan al rendimiento de la autopista, y serán estos los que se optimicen en primer lugar. Si en un futuro se contara con una versión completa del programa, la optimización sería mucho más fiable y rápida de realizar.

La versión completa de AnyLogic permite realizar esta operación de manera casi directa. Al no contar con esta, se tuvieron que efectuar algunos pasos previos aprovechando los conocimientos de Java que se tenían, para poder realizar la optimización. Esto supone que la optimización se realizará buscando maximizar el número de coches enviados y que alcanzan su destino puesto que, como se comentó en un principio, esta es una de las dos magnitudes que al mejorar suponen un progreso en el flujo de la autopista.

Para cada optimización se realizarán un total de 500 iteraciones, tomando un valor mínimo para cada fase 5 segundos y un máximo de 80, iterando en 5 segundos.

Optimization - Optimization Experiment

Name: Ignore

Top-level agent: ▾

Objective: minimize maximize

Number of iterations:

Automatic stop

Maximum available memory: ▾ Mb

Figura 18: Optimización de parámetros 1.

par_tl_60_9	discrete	5	80	5
par_tl_15_9	discrete	5	80	5
par_tl_60_10	discrete	5	80	5
par_tl_15_10	discrete	5	80	5
par_tl_60_11	discrete	5	80	5
par_tl_15_11	discrete	5	80	5

Figura 19: Optimización de parámetros 2.

Por supuesto, cada iteración tendrá un valor de semilla aleatorio para evitar caer en el *overfitting* propio de estas tareas.

time_in	time_out	cars_created	cars_dispose	cars_error_dispose	source_name	time_model	tag
11:28:54	12:06:56	10455	4753	0	MatrizOD	3599	Semáforos optimizados 1
11:28:52	12:07:12	10461	4625	0	MatrizOD	3599	Semáforos optimizados 1
11:28:57	12:07:22	10477	4747	0	MatrizOD	3599	Semáforos optimizados 1
11:29:00	12:07:32	10383	4696	0	MatrizOD	3599	Semáforos optimizados 1
11:28:39	12:08:39	10464	4703	2	MatrizOD	3599	Semáforos optimizados 1
11:28:44	12:08:55	10553	4820	0	MatrizOD	3599	Semáforos optimizados 1
11:28:42	12:09:04	10612	4722	0	MatrizOD	3599	Semáforos optimizados 1
11:29:13	12:13:19	10517	4838	0	MatrizOD	3599	Semáforos optimizados 1
11:28:47	12:14:14	10550	4812	0	MatrizOD	3599	Semáforos optimizados 1
11:28:49	12:14:23	10536	4706	0	MatrizOD	3599	Semáforos optimizados 1
11:29:08	12:16:28	10594	4801	1	MatrizOD	3599	Semáforos optimizados 1
12:07:32	12:32:22	10521	4770	0	MatrizOD	3599	Semáforos optimizados 1
12:07:09	12:32:47	10357	4687	0	MatrizOD	3599	Semáforos optimizados 1
12:07:40	12:33:03	10479	4693	0	MatrizOD	3599	Semáforos optimizados 1
12:07:21	12:33:12	10405	4703	0	MatrizOD	3599	Semáforos optimizados 1
12:09:04	12:33:17	10490	4775	0	MatrizOD	3599	Semáforos optimizados 1
12:08:48	12:33:42	10435	4735	0	MatrizOD	3599	Semáforos optimizados 1
12:09:13	12:34:04	10444	4742	0	MatrizOD	3599	Semáforos optimizados 1
12:11:07	12:34:22	10434	4663	0	MatrizOD	3599	Semáforos optimizados 1
12:51:19	12:59:29	10477	4827	1	MatrizOD	3599	Semáforos optimizados 1

Tabla 6: Resultados Simulación Semáforos optimizados.

Esta tabla muestra el momento en que fue ejecutada, el tiempo que tarda la simulación a máxima velocidad, el número de coches creados, así como los que llegaron al destino y los que no pudieron llegar por pasarse la salida, el origen de los datos y el tiempo de transcurrido dentro del modelo.

2.8.4 Semáforos Inteligentes

Un semáforo inteligente, es aquel que combina un semáforo tradicional con un conjunto de sensores que permiten modificar los estados del semáforo en base a la situación actual de la carretera en la que se encuentra.

Estos semáforos además podrían compartir información entre ellos, modificar conjuntamente sus estados, fomentar la circulación de ciertos vehículos como guaguas, etc.

En este trabajo, se han localizado las zonas que una vez optimizados los semáforos tradicionales, tienen tendencia a mostrar una alta saturación de las vías circundantes.

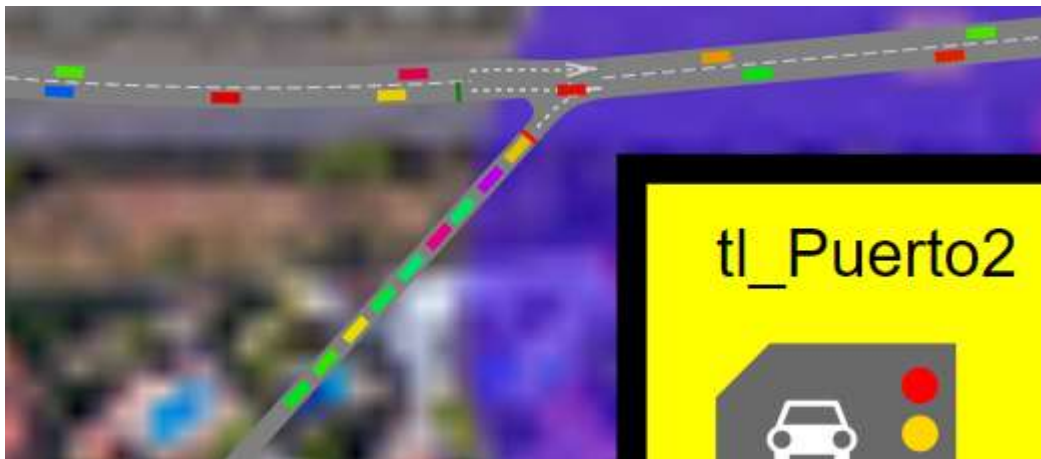


Figura 20: Semáforo con tráfico retenido.

Una vez localizados estos puntos se ha procedido a crear los “sensores”, estos consisten en dos *StopLines* por carril, que miden la cantidad de coches que se encuentran entre estos, y en base al estado, emiten la orden correspondiente al semáforo, que cambiará para fomentar la circulación por un carril u otro. La disposición es la siguiente:

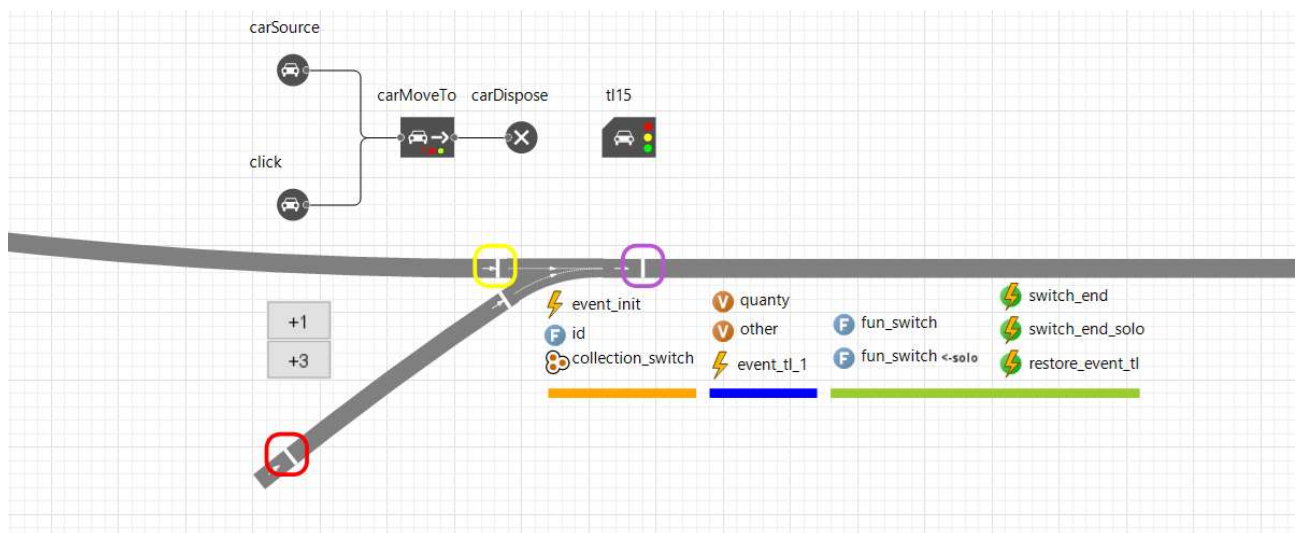


Figura 21: Ejemplo conceptual de la lógica de los semáforos inteligentes.

En la imagen se observa un carril principal, el horizontal, al que se le une un carril secundario con un ceda al paso. En la parte superior vemos la lógica, *carSource* envía vehículos por el carril principal de manera periódica, mientras que por el carril secundario se originaran agentes en base a los botones

de la izquierda.

Al lado de estos, se observa un semáforo, que el evento *event_init*, apagará según empiece el programa. Este semáforo pone “en verde” el carril secundario, y “en rojo” al principal de manera indefinida.

Dentro del grupo de color naranja, observamos una función *id* que devuelve el valor *hash* de un objeto, y un array *collection_switch*, que almacena estos *hash*.

El funcionamiento del semáforo es el siguiente:

- Cuando un coche se origina en el carril principal, al pasar por el *StopLine* amarillo, el contador *other* aumenta.
- Cuando un coche se origina en el carril secundario, al pasar por el *StopLine* rojo, el contador *quanty* aumenta.
- Cuando un coche pasa por el *StopLine* violeta, se comprueba si *other* es mayor que “1”, en este caso se reduce *other*, si su valor fuera “0”, se reduciría el contador *quanty*.

Esto permite tener control de los coches que se originan en el carril secundario y que aún no han terminado de pasar la intersección, puesto que aunque el sensor se podría haber puesto en el final del carril secundario, los agentes tienen tendencia a detenerse en la propia intersección para realizar el ceda al paso, de manera que en esa situación, estos no se contarían.

- El evento *event_tl_1*, tiene una condición que comprueba si el valor de *quanty*, los coches que esperan en el carril secundario, supera o iguala cierto valor. En el caso de que lo supere, se llama a la función *fun_switch*
- La función *fun_switch*, activa el semáforo que se indique. Posteriormente espera una cantidad de tiempo especificada, y lanza dos eventos, uno que desactive el semáforo que se activó *switch_end_solo*, y otro evento que permite que el evento *event_tl_1*, que comprueba la condición antes nombrada, pueda volver a comprobarse, pues estos se apagan al cumplir su condición.
- Como se puede observar, la función *fun_switch*, está sobrecargada, para contemplar dos situaciones:
 - La primera es la planteada, una intersección en la que no hay semáforos activos, y se fomenta con un evento el uso de una carretera frente a otra.
 - El segundo caso consiste en una intersección que tiene un semáforo por defecto, el cual la función apagará para dejar paso al semáforo que hemos nombrado anteriormente. Para finalmente, cuando haya transcurrido el tiempo especificado, se apague el semáforo inteligente, y se vuelva a encender el semáforo por defecto (evento *switch_end*).

Este sistema ha sido aplicado en diferentes puntos de la autopista, logrando los siguientes resultados:

time_in	time_out	cars_created	cars_dispose	cars_error	dispose	source_name	time_model	tag
18:45:49	19:29:04	10453	4686	0		MatrizOD	3599	Semáforos inteligentes 1
18:45:55	19:29:28	10506	4696	1		MatrizOD	3599	Semáforos inteligentes 1
18:45:41	19:29:30	10540	4742	0		MatrizOD	3599	Semáforos inteligentes 1
18:45:39	19:29:44	10549	4810	1		MatrizOD	3599	Semáforos inteligentes 1
18:45:52	19:30:09	10613	4827	0		MatrizOD	3599	Semáforos inteligentes 1
18:46:02	19:30:29	10596	4778	0		MatrizOD	3599	Semáforos inteligentes 1
18:46:07	19:30:30	10667	4830	0		MatrizOD	3599	Semáforos inteligentes 1
18:45:37	19:30:54	10543	4679	0		MatrizOD	3599	Semáforos inteligentes 1
18:45:44	19:31:05	10587	4915	0		MatrizOD	3599	Semáforos inteligentes 1
18:48:14	19:31:59	10642	4912	0		MatrizOD	3599	Semáforos inteligentes 1
18:46:11	19:32:30	10660	4803	2		MatrizOD	3599	Semáforos inteligentes 1
19:35:15	20:18:23	10627	4774	1		MatrizOD	3599	Semáforos inteligentes 1
19:35:08	20:18:48	10484	4825	0		MatrizOD	3599	Semáforos inteligentes 1
19:35:10	20:19:50	10505	4699	2		MatrizOD	3599	Semáforos inteligentes 1
19:35:34	20:22:02	10442	4750	0		MatrizOD	3599	Semáforos inteligentes 1
19:35:13	20:22:11	10549	4852	0		MatrizOD	3599	Semáforos inteligentes 1
19:35:22	20:22:30	10576	4815	0		MatrizOD	3599	Semáforos inteligentes 1
19:35:19	20:22:36	10600	4726	0		MatrizOD	3599	Semáforos inteligentes 1
19:35:29	20:22:38	10490	4763	0		MatrizOD	3599	Semáforos inteligentes 1

Tabla 7: Resultados Simulación Semáforos inteligentes.

Esta tabla muestra el momento en que fue ejecutada, el tiempo que tarda la simulación a máxima velocidad, el número de coches creados, así como los que llegaron al destino y los que no pudieron llegar por pasarse la salida, el origen de los datos y el tiempo de transcurrido dentro del modelo.

Capítulo 3 Resultados

3.1 Datos obtenidos

En el capítulo anterior se han realizado varios tipos de simulaciones. Cada tipo de simulación ha sido realizada más de 20 veces en su versión final.

En las simulaciones finales se han exportado una serie de valores a un archivo .xlsx. Estos datos miden las siguientes magnitudes:

- Tiempo de ejecución.
- Número de coches creados.
- Número de coches que han alcanzado su destino.
- Número de coches que han producido un error al no poder llegar a su destino.

El número de coches que producen error se omitirá a partir de ahora por tener un valor medio muy próximo al 0 en todas las simulaciones, así como el tiempo de ejecución, pues solo se tomarán en cuenta las simulaciones que hayan completado los 3599 segundos de simulación.

Simulación simple

Simulación realista de la autopista tal y como es a día de hoy.

Simulación con semáforos sin optimizar

Simulación simple con semáforos en las intersecciones. Los tiempos de los semáforos son iguales para todos, 60 segundos la autopista, 15 segundos la incorporación.

Simulación con semáforos optimizados

Simulación con semáforos, pero solo activos en aquellos lugares donde se produce una saturación mayor. Sus tiempos han sido optimizados con la versión gratuita de AnyLogic.

Simulación con semáforos inteligentes

Simulación con semáforos optimizados, en los que se han incluido eventos que modifican los valores de los semáforos activos, y que crean nuevos semáforos según el estado de varios puntos de la vía.

Valores promedio	Coches Creados	Coches Enviados
Simple	9746,55	4819,2
Sin optimizar	6577,896552	4333,37931
Optimizados	10482,2	4740,9
Inteligentes	10578,72222	4754,833333

Tabla 8: Resultados promedios.

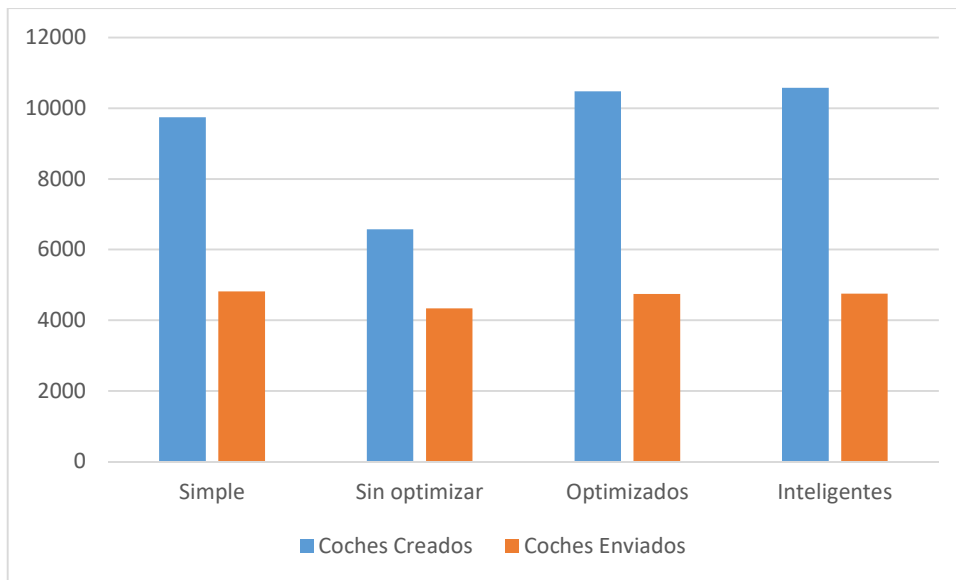


Figura 22: Comparativa valores promedio.

Desviación Estándar	Coches Creados	Coches Enviados
Simple	50,31215716	59,43824748
Sin optimizar	40,08503892	53,88175797
Optimizados	66,3718393	58,56161847
Inteligentes	65,5574618	68,23643108

Tabla 9: Desviación Estándar.

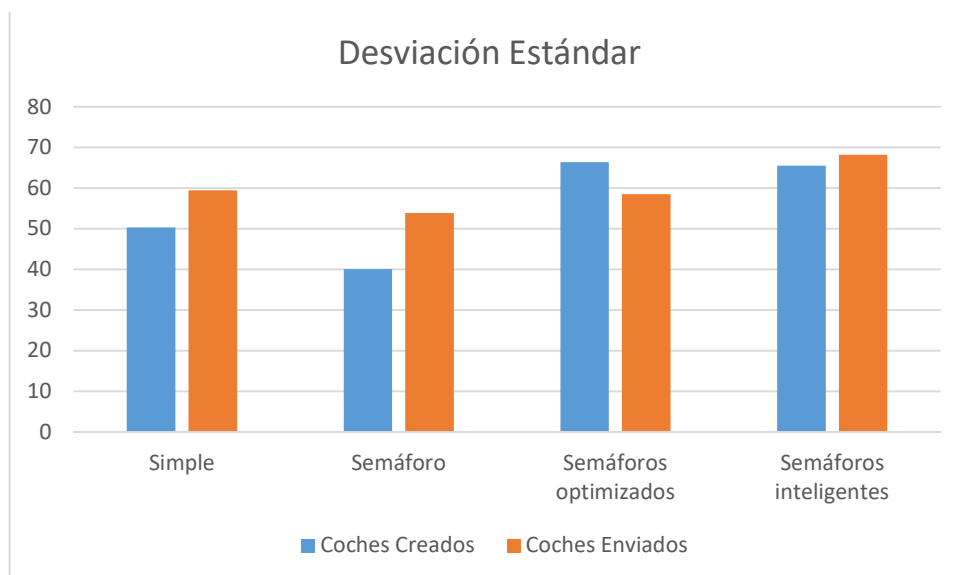


Figura 23: Comparativa Desviación Estándar.

3.2 Comparativa

A primera vista podemos observar que el número de coches que alcanzan su destino en la simulación simple, es superior a los tres casos restantes. Sin embargo el número de coches generados es inferior al modelo optimizado y al modelo inteligente.

Relación con Simple	Coches Creados	Coches Enviados
Sin optimizar	0,674894866	0,899190594
Optimizados	1,07547799	0,98375249
Inteligentes	1,085381209	0,986643703

Tabla 10: Comparativa Simulación Simple con Semáforos.

Simulación con semáforos sin optimizar

La simulación es capaz de crear solo el 67% de los coches que genera el modelo Simple.

Por contraparte, el número de coches que alcanzan su destino se mantiene próximo al 90%.

Estos datos nos hacen pensar que a pesar de conseguir enviar un número de coches similar, el sobrante que no consigue entrar en la autopista, permanece en las inmediaciones de la misma.

Simulación con semáforos optimizados

Este modelo es capaz de crear un 7,5% más de los coches que genera el modelo Simple, una cifra cercana al modelo actual, pero que denota un mayor número de coches circulando paralelamente en la autopista.

De manera similar, pero mejor al modelo sin optimizar, encontramos que el número de coches que alcanzan su destino es superior al 98%, una cifra que podríamos considerar prácticamente igual al estado actual.

El aumento del número de coches generados, desembocan en la idea de una menor afluencia en las inmediaciones de la autopista.

Simulación con semáforos inteligentes

Con resultados ligeramente superiores al modelo optimizado, esta simulación logra generar un 8,5% más de coches, y enviar a su destino casi un 99% de los coches actuales.

Esta simulación denota que la autopista acoge de manera coordinada a un mayor número de vehículos, pero que la sensación para los usuarios resulta igual.

Capítulo 4 Conclusiones y líneas futuras

A partir de los datos recopilados de las diferentes simulaciones, podemos observar una mejora notable en las inmediaciones de la autopista.

Del mismo modo se observa como el número de vehículos que alcanzan su destino se mantiene estable a lo largo de las diferentes simulaciones, variando muy poco, 0.0134%, en el caso de los semáforos inteligentes, respecto a la situación actual de la autopista.

Podemos decir a partir de estos datos, que el uso de semáforos parece no modificar en gran medida a la cantidad de agentes que alcanzan su objetivo en una hora, conservándose esta estable. Por otra parte, si notamos un aumento considerable del número de vehículos que son capaces de alcanzar la autopista. Esto hace pensar que **el uso de semáforos mejora las inmediaciones de la autopista**, dicho con otras palabras, aumenta la densidad de la autopista sin afectar a su rendimiento, liberando significativamente las entradas y salidas de coches en espera.

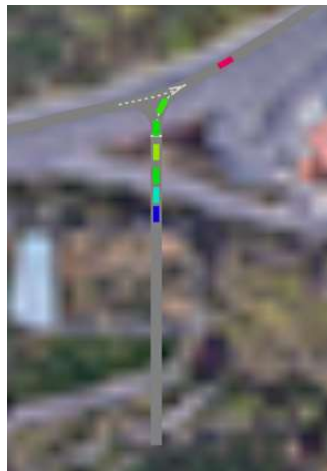


Figura 24: Entrada bloqueada por el ceda al paso, modelo simple.



Figura 25: Entrada guiada por un semáforo, modelo inteligente.

Analizando el flujo de las simulaciones, es posible afirmar que existe una **mayor coordinación entre los vehículos mediante el uso de semáforos**, y que aunque no son una medida que reduzca de manera drástica los problemas de tráfico de la autopista, sí se acercan a una posible mejora de los mismos.

Debo añadir que el uso de los semáforos inteligentes elimina el problema de los excesos de velocidad que se producen en las vías, pues entre los datos a los que se tiene acceso desde la consola de *PowerBI*, podemos observar como existe un alto número de usuarios que en horas no pico, exceden

el límite de velocidad, llegando incluso a superar los 250 km/h.

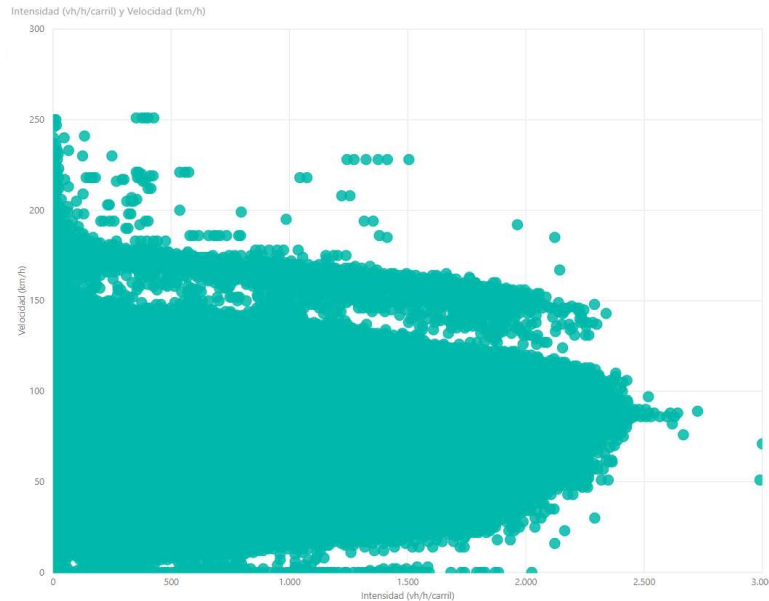


Figura 26: Registro de velocidades en 2018 en la autopista TF-5.

Si evaluamos objetivamente los resultados obtenidos con el planteamiento inicial, aumentar el número de vehículos que llegan a su destino, la respuesta que se obtiene de los datos es un **no, bajo estos criterios**.

Sin embargo esta pregunta puede plantearse al revés, ¿empeora significativamente el uso de semáforos inteligentes el número vehículos que alcanzan su destino?, la respuesta es **no, no empeora significativamente su uso**.

Cabe recordar que el modelo de semáforos inteligentes toma su base de la optimización de los semáforos previamente instalados. Esta modificación no ha sido realizada de la manera más óptima, pues se debió hacer por grupos.

Es presumible que el uso de **una versión completa de AnyLogic mejoraría los resultados obtenidos por la optimización**, influyendo, y muy posiblemente **equiparando o superando los resultados obtenidos** por los semáforos, al estado actual de la autopista, pudiendo concluir con una mejoría del tráfico de la autopista.

AnyLogic es un software que actualmente grandes empresas del mundo usan para sus tareas: IBM, Nasa, Ford, etc. Sin embargo, su aprendizaje tiene una curva de dificultad muy pronunciada, y carece de documentación pública sobre los errores de varias de sus librerías.

Este trabajo ha buscado realizar una simulación realista basándose en esta tecnología. Dicha tarea se ha logrado con éxito, y además se ha podido desarrollar en el proceso documentación asociada a varios de los errores que el programa no es capaz de solventar de manera sencilla.

Además, se han planteado 3 modelos para intentar resolver los atascos que se producen en la autopista TF-5. El análisis de los modelos ha permitido descartar uno de ellos, los semáforos tradicionales, pero se han planteado las condiciones propicias para no cerrar la puerta a un futuro desarrollo que mejore los otros dos, semáforos optimizados y semáforos inteligentes.

Este proyecto en gran medida ha estado marcado por las limitaciones que la versión gratuita de AnyLogic marcaba. Esto, unido a lo anterior, me hace pensar que el uso de una versión completa del programa en el futuro, podría arrojar unos resultados que cambien la perspectiva actual sobre el uso

de semáforos en las autopistas.

Actualmente muy pocos lugares del mundo implementan esta propuesta en carreteras de alta velocidad, sin embargo, la observación de las simulaciones dejan ver que la circulación se realiza de manera más coordinada que sin ellos.

Este trabajo ha conseguido realizar todas las metas que se han ido planteando, abriendo la posibilidad de que en un futuro se tome como base lo desarrollado en este proyecto y permita agilizar el desarrollo de otros. Si esto fuera así, se podría:

- Buscar un mejor sistema de optimización, en el caso de que no se contara con la versión profesional del programa.
- Mejorar la inteligencia dada a los semáforos
- Obtener datos exactos de entradas y salidas por municipios.
- Implementar novedades de futuras versiones de AnyLogic, como por ejemplo *Redes Neuronales*, las cuales están en fase beta.
- Añadir otros tipos de vehículos como guaguas o vehículos pesados.

Capítulo 5 Summary and Conclusions

From the data collected from the different simulations, we can see a remarkable improvement in the vicinity of the highway.

In the same way it is observed how the number of vehicles that reach their destination remains stable throughout the different simulations, varying very little, 0.0134%, in the case of intelligent traffic lights, with respect to the current situation of the highway.

We can say from these data, that the use of traffic lights seems not to greatly modify the number of agents that reach their goal in one hour, keeping it stable. On the other hand, if we notice a considerable increase in the number of vehicles that are able to reach the highway. This suggests that **the use of traffic lights improves the immediate vicinity of the highway**, in other words, increases the density of the highway without affecting its performance, significantly releasing the entrances and exits of cars waiting.



Figura 27: Entrance blocked by yielding, simple model.



Figura 28: Entrance guided by a traffic light, intelligent model.

Analyzing the flow of the simulations, **it is possible to affirm that there is greater coordination between the vehicles through the use of traffic lights**, and that although they are not a measure that drastically reduces the traffic problems of the highway, they do approach a possible improvement of them.

I must add that the use of smart traffic lights eliminates the problem of speeding that occurs on the roads, because among the data that is accessed from the PowerBI console, we can see how there is a high number of users who in non-peak hours, they exceed the speed limit, even exceeding 250 km/h.

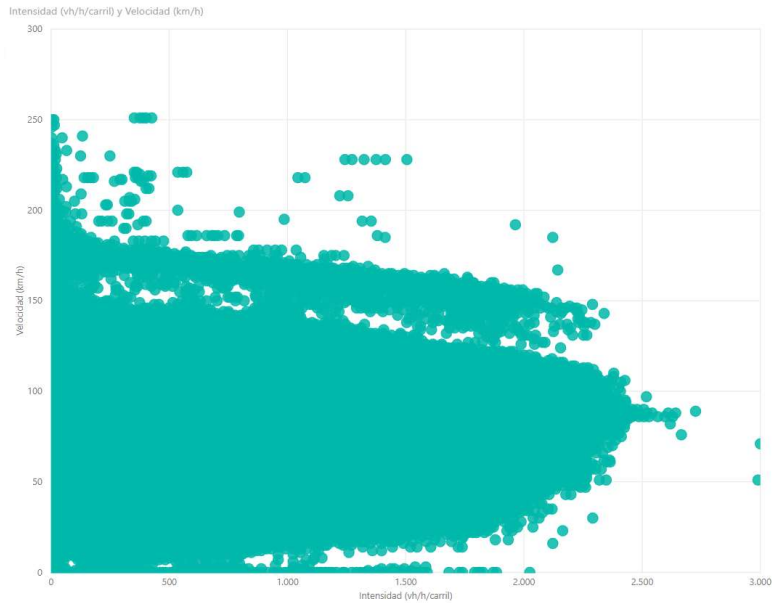


Figura 29: Registro de velocidades en 2018 en la autopista TF-5.

If we objectively evaluate the results obtained with the initial approach, increase the number of vehicles arriving at their destination, the answer obtained from the data is a **no, under these criteria**.

However, this question can be asked the other way around, does the number of vehicles that reach their destination significantly worsen the use of intelligent traffic lights? The answer is **no, it does not significantly worsen their use**.

It should be remembered that the intelligent traffic light model takes its base from the optimization of previously installed traffic lights. This modification has not been carried out in the most optimal way, as it should have been done in groups.

It is presumable that **the use of a full version of AnyLogic would improve the results obtained by the optimization**, influencing, and **very possibly equating or surpassing the results obtained by the traffic lights**, to the current state of the highway, being able to conclude with an improvement of the traffic of the freeway.

AnyLogic is a software that currently large companies in the world use for their tasks: IBM, Nasa, Ford, etc. However, their learning has a very pronounced difficulty curve, and it lacks public documentation about the errors of several of its libraries.

This work has sought to perform a realistic simulation based on this technology. This task has been successfully accomplished, and it has also been possible to develop in the process documentation associated with several of the errors that the program is not able to solve in a simple way.

In addition, 3 models have been proposed to try to resolve the traffic jams that occur on the TF-5 motorway. The analysis of the models has allowed us to rule out one of them, the traditional traffic lights, but the favorable conditions have been raised so as not to close the door to a future development that improves the other two, optimized traffic lights and intelligent traffic lights.

This project has largely been marked by the limitations that the free version of AnyLogic marked. This, together with the above, makes me think that the use of a full version of the program in the future could yield results that change the current perspective on the use of traffic lights on highways.

Currently, very few places in the world implement this proposal on high-speed roads, however, the observation of the simulations shows that the circulation is carried out in a more coordinated way than without them.

This work has managed to realize all the goals that have been raised, opening the possibility that in the future the basis of what was developed in this project will be taken as a basis and allow the development of others to be accelerated. If this were so, you could:

- Search for a better optimization system, in the event that the professional version of the program is not available.
- Improve intelligence given to traffic lights
- Obtain exact data of entrances and exits by municipalities.
- Implement news of future versions of AnyLogic, such as Neural Networks, which are in beta.
- Add other types of vehicles such as buses or heavy vehicles.

Capítulo 6 Presupuesto

Presupuesto referente a la elaboración de este proyecto:

Tarea	Horas	Precio por hora	Precio
Análisis estado del arte	20 horas	20€	400€
Análisis datos iniciales	60 horas	20€	1.200€
Aprendizaje AnyLogic	80 horas	20€	1.600€
Creación de modelos de simulación	200 horas	20€	4.000€
Control de errores AnyLogic	100 horas	20€	2.000€
Licencia AnyLogic	1	5.591€	5.591€
Total	460 horas		14.791€

Tabla 11: Presupuesto

Capítulo 7 Problemas encontrados en AnyLogic

7.1 Introducción

AnyLogic es una herramienta muy potente en el ámbito de la simulación. Sin embargo, posee muy poca o nula documentación asociada a los errores de algunas de sus librerías como *Road Traffic Library*.

A lo largo del desarrollo del trabajo, se han dado muchos errores que, al realizar una búsqueda en Internet, no tienen resultados. En algunas ocasiones estos errores aparecen en *StackOverflow*, pero apenas tienen interacciones y respuestas.

Estos errores son tan complicados de resolver porque normalmente no devuelven *feedback* al usuario que indique el motivo del error o su solución. Por supuesto, solo se está haciendo referencia a este tipo de errores, y no a aquellos propios de errores de compilación que sí avisan de la línea de código de la que parte el error, sino aquellos que ocurren al inicio o en medio de una simulación.

Es por ello que se han recopilado las acciones tomadas con varios errores que, a través de horas y en algunos casos semanas de depuración, se han conseguido solucionar.

7.2 Control Point Error

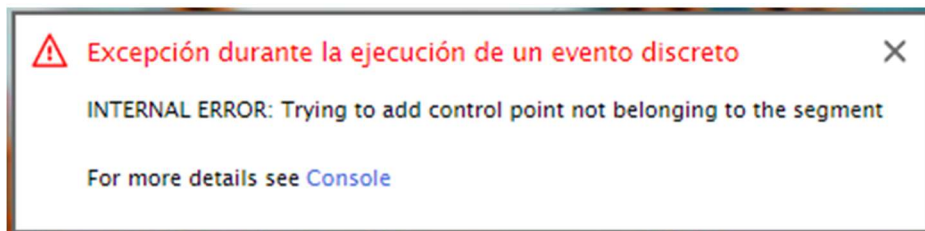


Figura 30: Control Point Error

```
Excepción durante la ejecución de un evento discreto:  
INTERNAL ERROR: Trying to add control point not belonging to the segment  
java.lang.RuntimeException: INTERNAL ERROR: Trying to add control point not belonging to the segment  
at com.anylogic.libraries.road.x.a(Unknown Source)  
at com.anylogic.libraries.road.RoadNetworkDescriptor.a(Unknown Source)  
at com.anylogic.libraries.road.Car_xjal.a(Unknown Source)  
at com.anylogic.libraries.road.Car_xjal.a(Unknown Source)  
at com.anylogic.libraries.road.Car_xjal.c(Unknown Source)  
at com.anylogic.libraries.road.Car_xjal.z(Unknown Source)  
at com.anylogic.libraries.road.Car_xjal.s(Unknown Source)  
at com.anylogic.libraries.road.RoadNetworkDescriptor.executeActionOf(Unknown Source)  
at com.anylogic.engine.EventTimeout.execute(Unknown Source)  
at com.anylogic.engine.Engine.h(Unknown Source)  
at com.anylogic.engine.Engine.nf(Unknown Source)  
at com.anylogic.engine.Engine.g(Unknown Source)  
at com.anylogic.engine.Engine$.run(Unknown Source)
```

Figura 31: Consola Control Point Error

Este se trata con diferencia del error que conlleva un depurado más complejo.

El error en cuestión aparece de manera “aleatoria” durante las ejecuciones. Si fijamos la semilla, este siempre aparecerá en el mismo punto, pero si la variamos observamos que puede aparecer en diferentes momentos.

Como se observa, no proporciona información sobre el origen del mismo, y para encontrar la solución se tuvo que partir la simulación en pedazos hasta encontrar el segmento mínimo que producía el error. Esto se realizó siguiendo el consejo de Benjamin Schumann, un experto en AnyLogic que imparte clases del curso oficial de AnyLogic, en Reino Unido.

Cabe aclarar que este error apareció durante varias semanas hasta conseguir contactar con Benjamin, el cual explicó cómo él buscaría el fallo, lo cual no significa que supiese resolverlo.

También se contactó con AnyLogic a fin de resolverlo, pero no supieron encontrar el error.

Aclarado esto, el error se produce cuando un agente pasa por un punto con un ángulo que no le permite realizar el giro de manera correcta, provocando un error al intentar tomar “la curva”, y su posterior interrupción de la simulación.

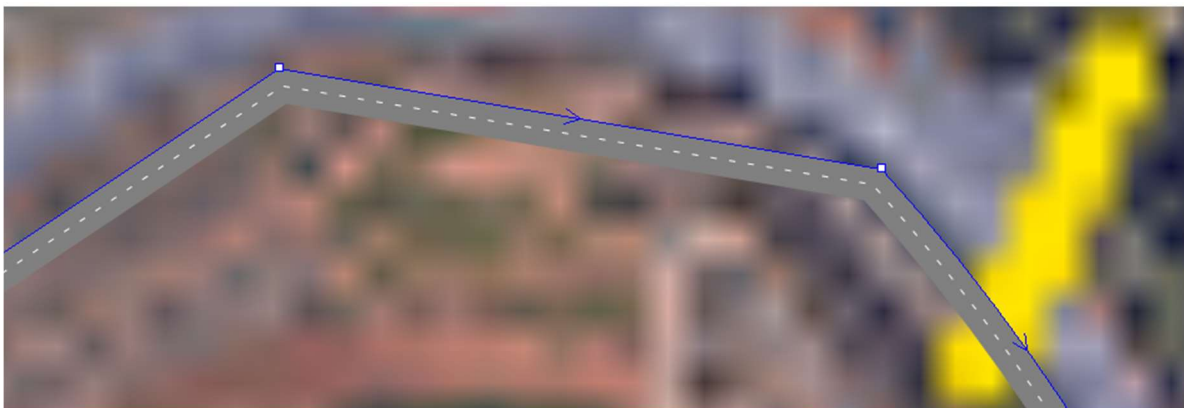


Figura 32: Ejemplo curva que provoca un Control Point Error.

Este es un ejemplo de un ángulo muy pronunciado, por el que los coches pueden pasar con normalidad, pero que ocasionalmente genera el error.

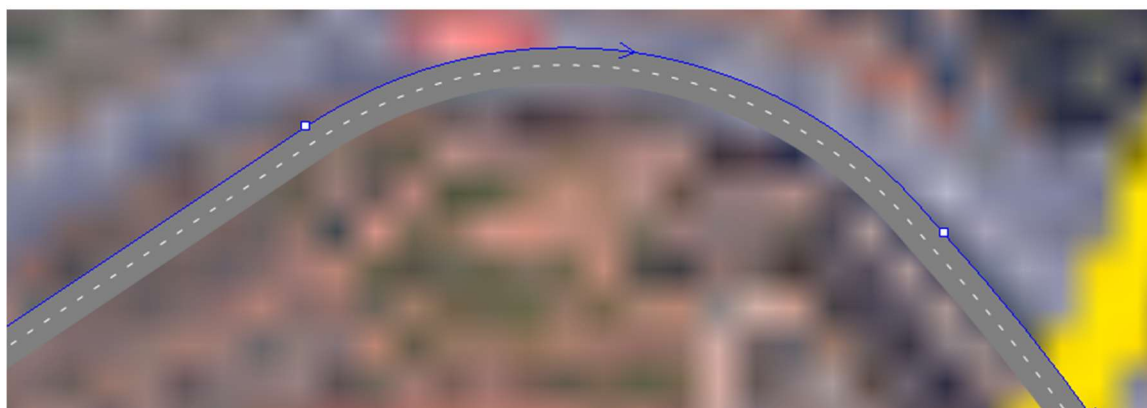


Figura 33: Curva arreglada.

Este es el mismo punto, pero con el segmento en forma de curva, que a pesar de afectar negativamente al rendimiento de la simulación, evita que se produzca el error.

Cabe aclarar que este error puede producirse en cualquier ángulo que sea diferente a 180°, pero con mayor probabilidad cuanto más pronunciado sea.

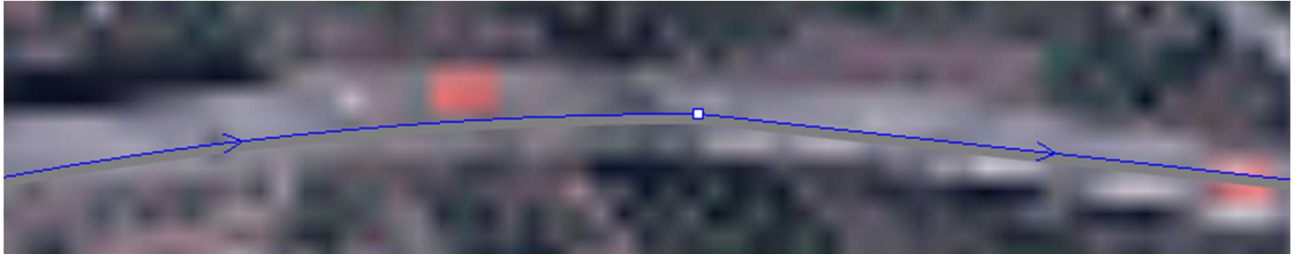


Figura 34: Otro ejemplo de Control Point Error.

Este segmento produce igualmente el error, aun teniendo un ángulo mucho más cercano a los 180°.

Con objeto de depurar el error, lo más sencillo es eliminar todos los ángulos de la simulación, sin embargo si fuera necesario conservar alguno, con la semilla aleatoria, habría que desactivar todos los *CarSource*, a excepción de uno que barra dos puntos relativamente próximos.

Mediante una simulación de tipo MonteCarlo (realiza múltiples iteraciones), podemos averiguar si el segmento produce un error. En ese caso, deberemos imprimir por pantalla el número de la semilla para poder replicarlo y, posteriormente, realizar simulaciones simples con esta semilla. Con esto sabremos que, en el segmento escogido para la semilla en cuestión, el ángulo lanza la excepción y ahí se encuentra un ángulo que produce el error.

En caso de que en la simulación MonteCarlo no se produzca el error, se podrá presuponer que ese segmento funcionará bien en el futuro.

7.3 Segment with length = 0

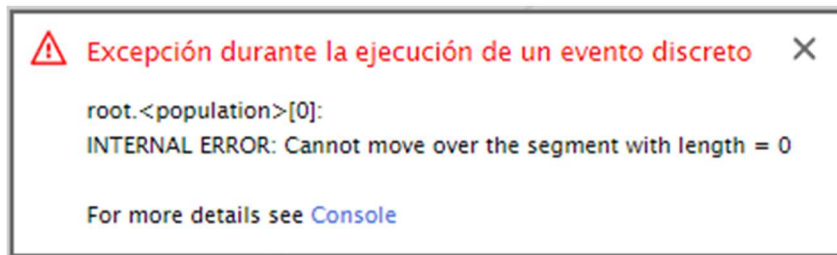


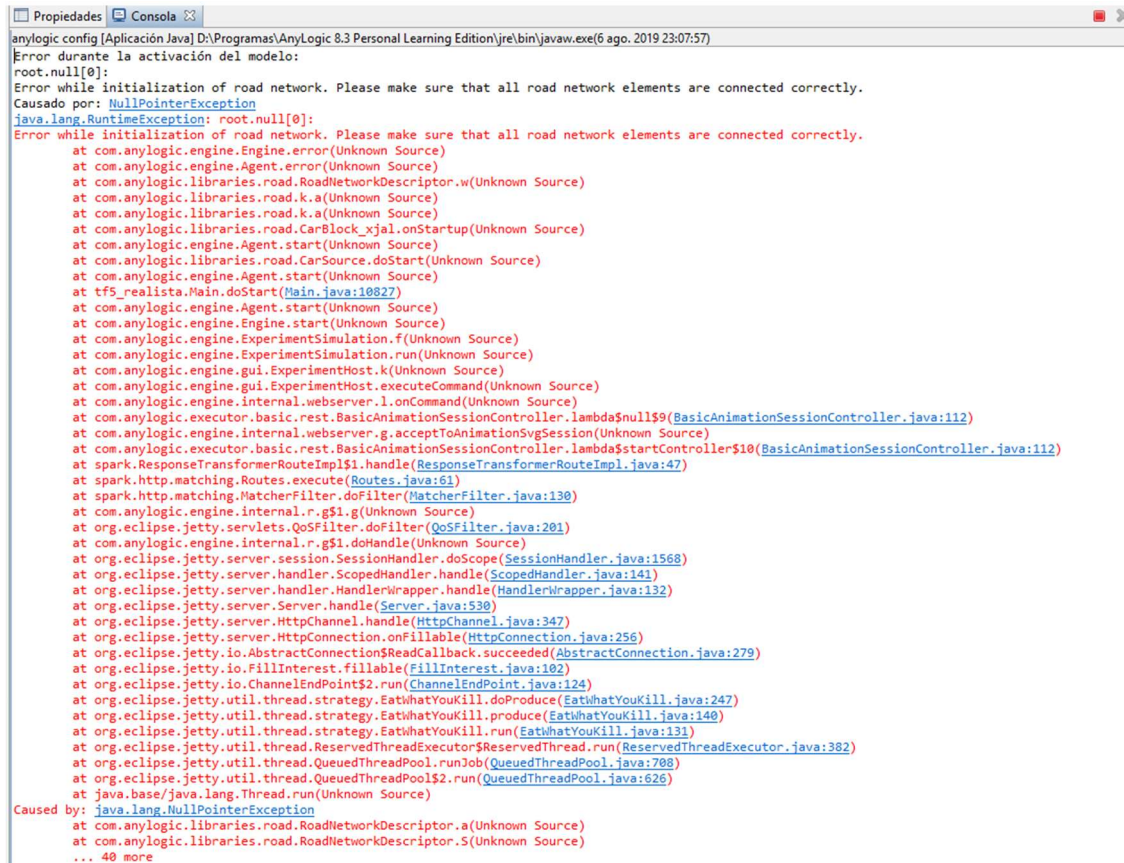
Figura 35: Segment with length = 0

Este error se produce cuando en una intersección, sea porque el espacio es muy pequeño o porque hemos movido el *StopLine*, esta se oculta dentro de la propia intersección.

Normalmente, el primer agente que entre a la intersección provocará el error así que, observando la simulación, se puede encontrar de manera sencilla el punto a arreglar.

La manera más sencilla de solucionarlo consiste en borrar la intersección y crear una nueva donde esta, en el caso de que hayamos movido la *StopLine* o si la intersección es muy pequeña, se alargue por uno de sus extremos.

7.4 Initialization of road network Error



```
anylogic config [Aplicación Java] D:\Programas\AnyLogic 8.3 Personal Learning Edition\jre\bin\javaw.exe(6 ago. 2019 23:07:57)
Error durante la activación del modelo:
root.null[0]:
Error while initialization of road network. Please make sure that all road network elements are connected correctly.
Causado por: NullPointerException
java.lang.RuntimeException: root.null[0]:
Error while initialization of road network. Please make sure that all road network elements are connected correctly.
at com.anylogic.engine.Engine.error(Unknown Source)
at com.anylogic.engine.Agent.error(Unknown Source)
at com.anylogic.libraries.road.RoadNetworkDescriptor.w(Unknown Source)
at com.anylogic.libraries.road.k.a(Unknown Source)
at com.anylogic.libraries.road.k.a(Unknown Source)
at com.anylogic.libraries.road.CarBlock_xjal.onStartUp(Unknown Source)
at com.anylogic.engine.Agent.start(Unknown Source)
at com.anylogic.libraries.road.CarSource.doStart(Unknown Source)
at com.anylogic.engine.Agent.start(Unknown Source)
at tf5_realista.Main.doStart(Main.java:10827)
at com.anylogic.engine.Agent.start(Unknown Source)
at com.anylogic.engine.Engine.start(Unknown Source)
at com.anylogic.engine.ExperimentSimulation.f(Unknown Source)
at com.anylogic.engine.ExperimentSimulation.run(Unknown Source)
at com.anylogic.engine.gui.ExperimentHost.k(Unknown Source)
at com.anylogic.engine.gui.ExperimentHost.executeCommand(Unknown Source)
at com.anylogic.engine.internal.webserver.l.onCommand(Unknown Source)
at com.anylogic.executor.basic.rest.BasicAnimationSessionController.lambda$null$19(BasicAnimationSessionController.java:112)
at com.anylogic.executor.basic.rest.BasicAnimationSvgSession(Unknown Source)
at com.anylogic.executor.basic.rest.BasicAnimationSessionController.lambda$startController$10(BasicAnimationSessionController.java:112)
at spark.ResponseTransformerRouteImpl$1.handle(ResponseTransformerRouteImpl.java:47)
at spark.http.matching.Routes.execute(Routes.java:61)
at spark.http.matching.MatcherFilter.doFilter(MatcherFilter.java:130)
at com.anylogic.engine.internal.r.g$1.g(Unknown Source)
at org.eclipse.jetty.servlets.QoSFilter.doFilter(QoSFilter.java:201)
at com.anylogic.engine.internal.r.g$1.doHandle(Unknown Source)
at org.eclipse.jetty.server.session.SessionHandler.doScope(SessionHandler.java:1568)
at org.eclipse.jetty.server.handler.ScopedHandler.handle(ScopedHandler.java:141)
at org.eclipse.jetty.server.handler.HandlerWrapper.handle(HandlerWrapper.java:132)
at org.eclipse.jetty.server.Server.handle(Server.java:530)
at org.eclipse.jetty.server.HttpChannel.handle(HttpChannel.java:347)
at org.eclipse.jetty.server.HttpConnection.onFillable(HttpConnection.java:256)
at org.eclipse.jetty.io.AbstractConnection$ReadCallback.succeeded(AbstractConnection.java:279)
at org.eclipse.jetty.io.FillInterest.fillable(FillInterest.java:102)
at org.eclipse.jetty.io.ChannelEndPoint$2.run(ChannelEndPoint.java:124)
at org.eclipse.jetty.util.thread.strategy.EatWhatYouKill.doProduce(EatWhatYouKill.java:247)
at org.eclipse.jetty.util.thread.strategy.EatWhatYouKill.produce(EatWhatYouKill.java:140)
at org.eclipse.jetty.util.thread.strategy.EatWhatYouKill.run(EatWhatYouKill.java:131)
at org.eclipse.jetty.util.thread.ReservedThreadExecutor$ReservedThread.run(ReservedThreadExecutor.java:382)
at org.eclipse.jetty.util.thread.QueuedThreadPool.runJob(QueuedThreadPool.java:708)
at org.eclipse.jetty.util.thread.QueuedThreadPool$2.run(QueuedThreadPool.java:626)
at java.base/java.lang.Thread.run(Unknown Source)
Caused by: java.lang.NullPointerException
at com.anylogic.libraries.road.RoadNetworkDescriptor.a(Unknown Source)
at com.anylogic.libraries.road.RoadNetworkDescriptor.S(Unknown Source)
... 40 more
```

Figura 36: Initialization of road network Error

Este error se produce ocasionalmente cuando el tamaño de una intersección no es suficientemente grande. Puede ocurrir simplemente al crear la intersección.

La manera más sencilla de arreglarlo es comentar todas las intersecciones e ir las descomentando por grupos hasta encontrar la que causa el error. Luego de localizarla, simplemente hay que alargarla por uno de sus extremos.

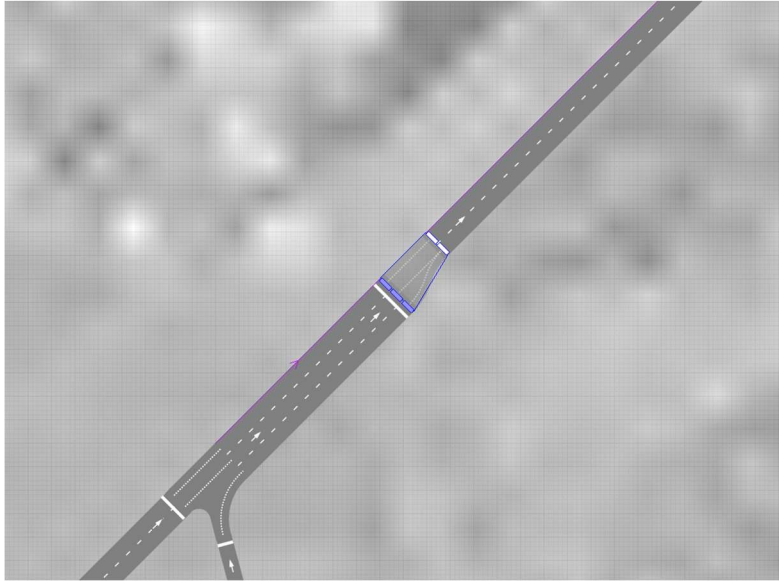


Figura 37: Intersección muy corta.

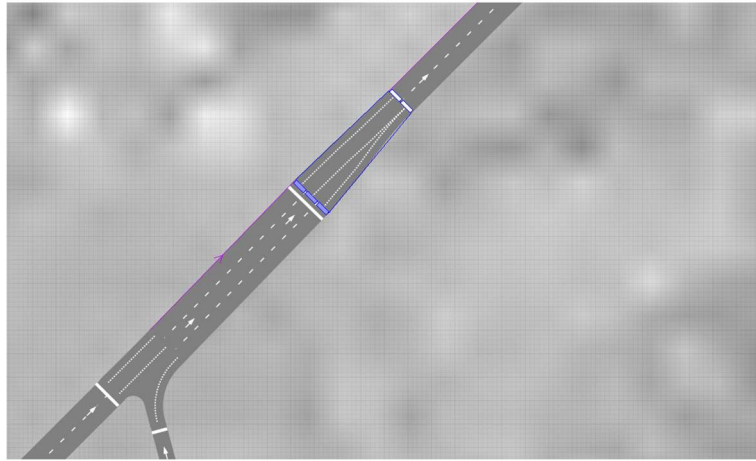


Figura 38: Intersección corregida.

7.5 StopLine: Not existing lanes specified

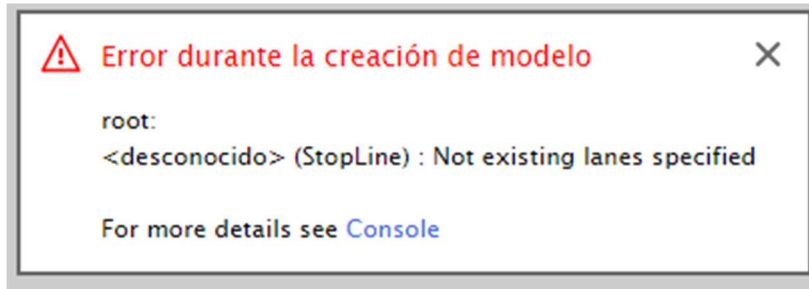


Figura 39: StopLine Error

Este error ocurre cuando se crea de manera automática una intersección, otorgándosele automáticamente el nombre a la *StopLine*, el cual ya está siendo usado por otra intersección.

Suele ocurrir mayoritariamente cuando se crean dos *Road Networks* no conectadas entre sí, pues pueden tomar los mismos nombres. Si estas se unen, puede ocurrir que los nombres se superpongan.

El error es de los más sencillos de solucionar, pues si observamos atentamente la lista de elementos del proyecto, observaremos que algunos aparecerán con una pequeña *x* roja en una esquina, indicando que se trata de un elemento con el nombre duplicado.

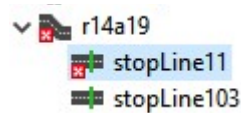


Figura 40: Ejemplo de indicador StopLine Error

Para resolverlo basta con cambiar el nombre de uno de los dos elementos que comparten nombre

Bibliografía

[1] MatrizOD

<http://154.48.153.16:8080/odOCT2018/>

[2] Comparativa software simulación

<https://www.anylogic.com/resources/white-papers/simulation-software-comparison/>

[3] Lenguajes de programación más populares

<https://www.techrepublic.com/article/programming-languages-which-was-most-popular-each-year/>

[4] Estudio realizado por MIT sobre semáforos en carreteras de alta velocidad

https://stuff.mit.edu/afs/athena/course/11/11.951/OldFiles/oldstuff/albacete/Other_Documents/Europe%20Transport%20Conference/traffic_engineering_an/review_of_traffic_1545.pdf