

Índice

1	<i>Introducción y Objetivos del Proyecto</i>	1
1.1	Introducción	1
1.2	Simuladores microscópicos de tráfico	1
1.2.1	Etapas en el proceso de simulación	2
1.2.2	Componentes de los simuladores microscópicos de tráfico	2
1.3	Objetivos del Proyecto	4
2	<i>Modelo de cambio de carril</i>	7
2.1	Introducción	7
2.2	Notación	7
2.3	Revisión histórica de modelos de cambios de carril	9
2.3.1	Modelo de cambio de carril de Gipps	9
2.3.2	Modelo de cambio de carril de Yang y Koutsopoulos	15
2.3.3	Selección de carril deseado	19
2.3.4	Espacio disponible para realizar el cambio de carril (Modelos de aceptación de Gap)	20
2.3.5	Modelo de fusión obligatoria (Merging Model)	23
2.3.6	Proceso de cambio de aceleración durante cambio obligado de carril	25
2.4	Diseño de un modelo de cambio de carril	26
2.4.1	Procedimiento general de cambio de carril	26
2.4.2	Cambio de carril obligatorio	28
2.4.2.1	Comprobación de distancia	29
2.4.2.2	Búsqueda del carril de destino obligatorio	30
2.4.2.3	Comprobación de la distancia de seguridad de la maniobra (gap)	33
2.4.2.3.1	Búsqueda de influencias	33
2.4.2.3.2	Aceptación de la distancia de seguridad en la maniobra	33
2.4.2.3.3	Probabilidad de cortesía (Modelo de Merging)	36
2.4.3	Cambio de carril por mejora	39
2.4.3.1	Condiciones previas	39
2.4.3.2	Búsqueda de carril destino por mejora	40
2.4.3.3	Comprobación de la distancia de seguridad de la maniobra (gap)	41
2.4.4	Inicio de cambio de carril	42
2.4.4.1	Creación de un vehículo ficticio	42
2.4.4.2	Actualización de influencias	43
2.4.5	Finalización de un cambio de carril	43
2.4.5.1	Vehículo real	43
2.4.5.2	Vehículo ficticio	44
3	<i>Función de generación de vehículos</i>	45
3.1	Introducción	45
3.2	Función de generación trapezoidal	45
3.3	Distribuciones estadísticas	47

3.3.1 Función de distribución normal	47
3.3.1.1 Método Box Müller	48
3.3.1.2 Método Marsaglia Bray	48
3.3.2 Función de distribución exponencial	49
3.3.2.1 Método inverso	50
3.3.3 Función de distribución triangular	50
3.3.3.1 Método de transformación	51
3.3.4 Función de distribución de Poisson	51
3.3.4.1 Método de generación	52
3.3.5 Test de bondad de la chi cuadrado de Pearson	53
3.3.5.1 Contraste de la distribución normal	53
3.3.5.2 Contraste de la distribución exponencial	54
3.3.5.3 Contraste de la distribución triangular	54
3.3.5.4 Contraste de la distribución de Poisson	55
3.4 Procedimiento de generación de vehículos	55
3.4.1 Cálculo del área de la sección del trapecioide	56
3.4.2 Aplicación de las funciones de distribución	61
3.4.3 Posición de generación de los vehículos	62
3.4.4 Comprobación de distancia	62
4 Estadísticas	64
4.1 Introducción	64
4.2 Estadísticas sobre cambios de carril	64
4.3 Estadísticas sobre la generación de vehículos	65
4.4 Estadísticas relacionadas con los semáforos	65
4.5 Estadísticas genéricas	65
5 Modelado orientado a objetos	67
5.1 Introducción a UML	67
5.1.1 Breve historia de UML	67
5.1.2 ¿Qué es UML? ¿Por qué usar UML?	69
5.2 Especificación funcional del simulador mediante UML	71
5.2.1 Diagramas de casos de uso	71
5.2.1.1 Diagrama de casos de uso de nivel 0	73
5.2.1.2 Diagrama de casos de uso de nivel 1	74
5.2.1.3 Diagrama de casos de uso de nivel 2	75
5.2.1.4 Diagrama de casos de uso de nivel 3	77
5.2.2 Diagrama de paquetes	78
5.2.2.1 Descripción de los paquetes existentes en el simulador	79
5.2.2.2 Diagrama de paquetes del simulador	79
5.2.3 Diagramas de clases	80
5.2.3.1 Paquete de datos "Red"	84
5.2.3.1.1 Descripción de las clases del paquete "Red"	84
5.2.3.1.1.1 Clase "Nodo"	84
5.2.3.1.1.2 Clase "Micro_tramo"	87
5.2.3.1.1.3 Clase "Giro"	92

5.2.3.1.1.4 Clase "Movimiento" _____	93
5.2.3.1.1.5 Clase "Codigo_movimiento" _____	95
5.2.3.1.1.6 Clase "Indice_movimiento" _____	95
5.2.3.1.1.7 Clase "In_movimiento" _____	96
5.2.3.1.1.8 Clase "Meso_carril" _____	97
5.2.3.1.1.9 Clase "Codigo_carril" _____	97
5.2.3.1.1.10 Clase "Aparcamiento" _____	98
5.2.3.1.1.11 Clase "Aparcamiento_masivo" _____	98
5.2.3.1.1.12 Clase "Codigo_aparcamiento" _____	100
5.2.3.1.2 Diagrama de clases del paquete "Red" _____	100
5.2.3.2 Paquete de datos "Grupo Semafórico" _____	101
5.2.3.2.1 Descripción de las clases del paquete "Grupo Semafórico" _____	103
5.2.3.2.1.1 Clase "Estructura" _____	103
5.2.3.2.1.2 Clase "Codigo_estructura" _____	104
5.2.3.2.1.3 Clase "Codigo_senal" _____	104
5.2.3.2.1.4 Clase "Fase" _____	105
5.2.3.2.1.5 Clase "Codigo_fase" _____	106
5.2.3.2.1.6 Clase "Intermedio" _____	106
5.2.3.2.1.7 Clase "Codigo_intermedio" _____	106
5.2.3.2.1.8 Clase "Senal_semaforica" _____	107
5.2.3.2.2 Diagrama de clases del paquete "Grupo Semafórico" _____	108
5.2.3.3 Paquete de datos "Simulador Microscópico" _____	108
5.2.3.3.1 Descripción de las clases del paquete "Simulador Microscópico" _____	109
5.2.3.3.1.1 Clase "Vehiculo" _____	109
5.2.3.3.1.2 Clase "Tipo_vehiculo" _____	115
5.2.3.3.1.3 Clase "Tipo_conductor" _____	115
5.2.3.3.1.4 Clase "Configuracion_Microscopico" _____	117
5.2.3.3.1.5 Clase "Obstaculo" _____	118
5.2.3.3.1.6 Clase "Parada" _____	119
5.2.3.3.2 Diagrama de clases del paquete "Simulador Microscópico" _____	119
5.2.3.4 Paquete de datos "Generación Vehículos" _____	120
5.2.3.4.1 Descripción de las clases del paquete "Generación Vehículos" _____	120
5.2.3.4.1.1 Clase "Aleatorio" _____	120
5.2.3.4.1.2 Clase "Exponencial" _____	121
5.2.3.4.1.3 Clase "Normal" _____	122
5.2.3.4.1.4 Clase "Poisson" _____	122
5.2.3.4.1.5 Clase "Triangular" _____	123
5.2.3.4.1.6 Clase "Angulo" _____	123
5.2.3.4.1.7 Clase "Trapezio" _____	124
5.2.3.4.1.8 Clase "Generacion_pendiente" _____	127
5.2.3.4.2 Diagrama de clases del paquete "Generación Vehículos" _____	127
5.2.3.5 Paquete de datos "Estadísticas" _____	128
5.2.3.5.1 Clase "Elemento_estadistico" _____	128
5.2.3.5.2 Diagrama de clases del paquete "Estadísticas" _____	129
5.2.3.6 Paquete de datos "Listas" _____	129
5.2.3.6.1 Clase "Lista_inf" _____	129
5.2.3.6.2 Diagrama de clases del paquete "Listas" _____	131

6	<i>Diseño del Simulador Microscópico de Tráfico</i>	133
6.1	Introducción	133
6.2	Diagramas de flujo del simulador microscópico de tráfico	133
6.2.1	Simulación de tramos	137
6.2.2	Simulación de nodos	143
6.2.3	Actualización de la simulación	144
6.2.3.1	Actualización de un tramo	145
6.2.3.2	Generar vehículos	145
6.2.3.3	Comprobación de salidas de tramos	153
6.2.3.4	Cambios de carril en el tramo	153
6.2.3.4.1	Cambio de carril obligatorio	154
6.2.3.4.2	Cambio de carril por mejora	168
6.2.3.4.3	Finalización de un cambio de carril	171
6.2.3.5	Salida de aparcamientos	171
6.2.3.6	Estadísticas de semáforos	172
6.2.3.7	Actualización de nodos	174
6.2.4	Escritura de estadísticas finales	174
6.3	Interfaz de usuario	175
6.3.1	Interfaz gráfico	175
6.3.2	Ficheros de entrada	176
6.3.3	Ficheros de salida	181
7	<i>Pruebas del Simulador Microscópico de Tráfico</i>	188
7.1	Introducción	188
7.2	Pruebas de validación	188
7.2.1	Modelo de cambio de carril	189
7.2.2	Función de generación de vehículos	194
7.2.3	Estadísticas	197
7.3	Pruebas del Sistema	202
7.3.1	Prueba básica	202
7.3.2	Prueba extendida	218
8	<i>Conclusiones y posibles extensiones</i>	238
8.1	Conclusiones	238
8.2	Posibles extensiones	239
	<i>Apéndice A. Notación de los diagramas de flujo</i>	240
	<i>Apéndice B. Bibliografía</i>	242
	<i>Apéndice C. Índice de figuras</i>	244
	<i>Apéndice D. Índice de tablas</i>	248

1 Introducción y Objetivos del Proyecto

1.1 Introducción

El presente proyecto tiene por objeto la realización de una simulación microscópica de tráfico en un entorno urbano con el máximo nivel de detalle posible, es decir, que el comportamiento de los vehículos que circulen por el viario sea lo más parecido posible al que tendrían en la realidad.

Este tipo de simulación va adquiriendo un creciente interés debido al incremento de movilidad que sufren las ciudades hoy en día. La realización de simulaciones, y sobre todo con el análisis de sus resultados, permite extraer importantes conclusiones que pueden llevar a un mayor conocimiento de la situación general del tráfico en una ciudad, permitiendo ofrecer posibles soluciones a los problemas encontrados.

Además este tipo de estudio permite la realización de previsiones sobre la situación del tráfico bajo determinadas condiciones, o bien cómo podrían afectar la realización de ciertas modificaciones en la organización general del tráfico. Por ejemplo, se puede estudiar la conveniencia de variar los tiempos de permanencia en rojo de algunos semáforos con el fin de que las colas que se produzcan en los mismos queden minimizadas.

Antes de entrar a detallar los objetivos que se persiguen con la realización de este proyecto, es necesario realizar una introducción a los simuladores microscópicos de tráfico.

1.2 Simuladores microscópicos de tráfico

Los modelos de simulación microscópica de tráfico, como los desarrollados en el presente proyecto, tratan de describir con gran nivel de detalle el comportamiento de los vehículos que circulan a través del viario. Para ello, será necesario describir exhaustivamente tanto el entorno de simulación (escenario que se desea simular) como el modo de comportarse los vehículos en las distintas situaciones.

Los modelos empleados en este tipo de simulaciones pueden tener distinta naturaleza, en función de las necesidades de aquello que se desea simular:

- Modelos de tiempo discreto: se aplican para la caracterización del comportamiento de los conductores. El tiempo total de simulación se descompone en pequeños intervalos de tiempo, tras cada uno de los cuales, los conductores deciden qué acción efectuar en función de la situación con la que se encuentren.
- Modelos de eventos: son empleados para determinar el efecto que ejercen los semáforos, obstáculos y otros sucesos puntuales sobre los conductores que transitan por el viario.

- Modelos estocásticos: se aplican a todas aquellas situaciones en las que existe cierta aleatoriedad en lo que puede ocurrir. Son utilizados, por ejemplo, en la generación de nuevos vehículos y en todo aquello que tenga cierta probabilidad de producirse.

1.2.1 Etapas en el proceso de simulación

Todo proceso de simulación está sujeto al seguimiento de una serie de etapas:

- Especificación del problema y del objetivo del modelo.
- Especificación de aquello que el sistema debe estudiar.
- Desarrollo del modelo.
- Calibración del modelo.
- Validación del modelo.

En el apartado 1.3 se verá cómo han sido aplicadas las etapas anteriores para el desarrollo del simulador microscópico de tráfico del que es objeto el presente proyecto.

1.2.2 Componentes de los simuladores microscópicos de tráfico

Existen dos tipos diferentes de elementos presentes en estas simulaciones, en función de la naturaleza de los mismos. Por un lado se encuentran los componentes estáticos que son aquellos que determinan con precisión el escenario que se desea simular, permanecen inalterables a lo largo de la simulación. Por otro lado, se encuentran los componentes dinámicos que determinan las interacciones entre los diferentes elementos. En los siguientes apartados se estudian con detalle ambos tipos de componentes.

Componentes estáticos

Un viario está compuesto principalmente por las calles por donde circulan los vehículos y las intersecciones o cruces de las mismas. En el modelo que se va a utilizar, este viario se va a representar mediante un grafo dirigido $G(V,A)$, en el que las calles o tramos serán las aristas (A) y donde los cruces serán los nodos (V) del grafo. En la figura 1.1 se muestra un viario, en el que se han señalado los nodos y los tramos, así como el grafo asociado al escenario marcado en rojo, suponiendo que todos los tramos son bidireccionales.

Los tramos se descomponen en diferentes carriles, que pueden ser de muy distinto tipo, como por ejemplo, de ensanchamiento, de aceleración, etc. Tanto los tramos como los carriles están caracterizados mediante una serie de atributos que permiten diferenciarlos de los demás. Estos atributos, así como los correspondientes al resto de componentes estáticos y dinámicos de la simulación, serán estudiados en detalle en el apartado 5 del proyecto.

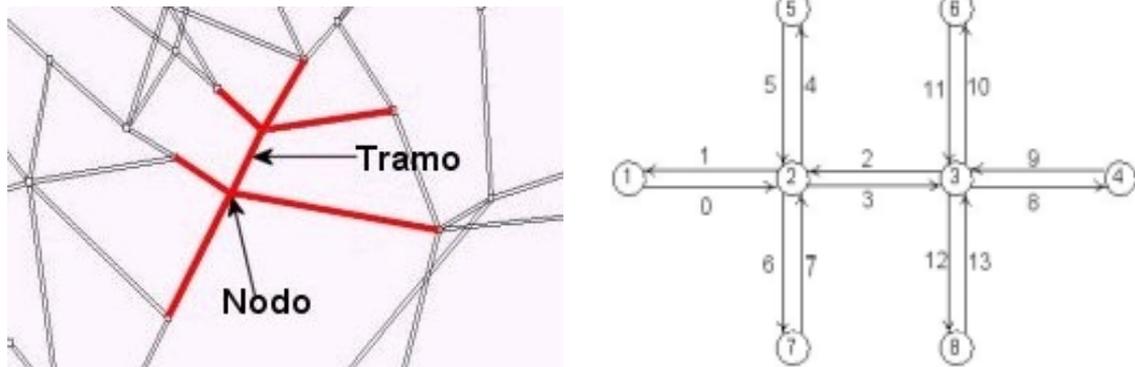


Figura 1.1. Representación de nodos y tramos del viario en un grafo

En el interior de las intersecciones se definen giros y movimientos. Los giros representan las posibilidades de desplazamiento de un vehículo que procede de un tramo con destino a otro tramo. Los movimientos representan una desagregación de los giros donde se incrementa el grado de detalle, representando las trayectorias que sigue el vehículo a lo largo del giro. En la figura 1.2 se representa un ejemplo de los distintos movimientos que se pueden producir en el interior de un nodo para llevar a cabo un giro.

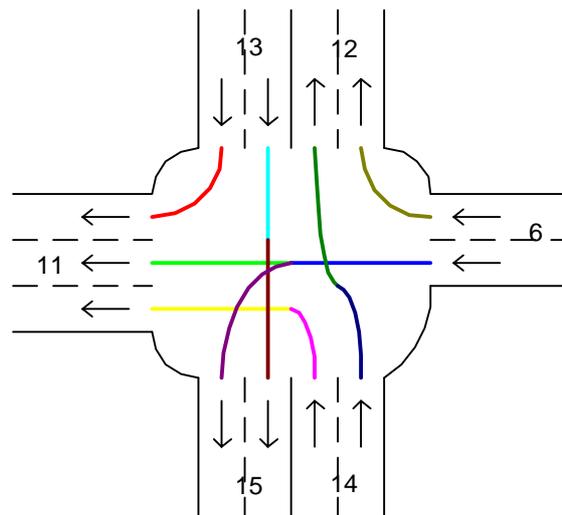


Figura 1.2. Movimientos en el interior de una intersección

Otros componentes estáticos de importancia fundamental son las señales de tráfico que influyen en el comportamiento de los vehículos sobre el viario, haciendo que se atengan a determinadas normas.

El tipo de conductor que maneja cada uno de los vehículos que circulan por el viario es otro de los componentes estáticos. Existe una interminable cantidad de características que diferencian a unos conductores de otros, por lo que se ha optado por agruparlas en cuatro tipos distintos de conductores, a los que se han denominado: *novatos*, *experimentados*, *agresivos* y *defensivos*.

Además, existen otros componentes estáticos, como son los aparcamientos y las rutas que deben seguir determinados vehículos. Todos ellos, se caracterizan de manera unívoca mediante una serie de atributos.

Componentes dinámicos

Los componentes dinámicos se caracterizan porque varían en el tiempo y porque determinan las interrelaciones existentes entre los distintos componentes del simulador microscópico de tráfico. El nivel de detalle con el que se definan estos componentes va a determinar la precisión del modelo de simulación.

Los componentes dinámicos de mayor importancia van a ser los modelos que reflejen el comportamiento de los vehículos por el viario. Entre ellos se encuentran:

- Modelos de aceleración: determinan la aceleración que se aplica a todos los vehículos en cada instante de simulación. Esta aceleración se determina en función de la influencia a la que se vea sometido cada vehículo, además de otras consideraciones.
- Modelos de cambio de carril: se encargan de determinar cuándo y cómo se van a llevar a cabo las maniobras de cambio de carril. Habrá dos tipos fundamentales de cambio de carril, obligatorio y por mejora, y serán descritos en detalle en el apartado 2, por ser uno de los objetivos principales de este proyecto.
- Modelos de selección de ruta y de aparcamiento: definen el recorrido que deben efectuar los vehículos en el escenario de simulación, así como los lugares en los que se deben realizar los aparcamientos.
- Modelos de generación de vehículos: se encargan de determinar cómo se incorporan nuevos vehículos a la simulación. Estos modelos serán objeto de estudio en el apartado 3.
- Modelos de incidencias: se emplean para determinar cuál es el comportamiento de los vehículos ante determinados eventos que se producen en momentos puntuales de la simulación.
- Otros componentes dinámicos a considerar son los semáforos, ya que afectan considerablemente al comportamiento que pueda tener un vehículo en un determinado instante de tiempo.

1.3 Objetivos del Proyecto

El objetivo del proyecto es el desarrollo de un simulador microscópico de tráfico en el que se describa el comportamiento de los vehículos en un entorno urbano. En particular, se van a implementar nuevos modelos de cambios de carril y de generación de vehículos, así como un módulo de estadísticas.

Sin embargo, el proceso necesario para la consecución de dicho objetivo es laborioso. En este proyecto se han seguido los siguientes pasos: especificación de los modelos de simulación microscópica que se van a emplear, diseño de dichos modelos empleando una terminología orientada a objetos (Unified Modeling Language, UML), realización de la simulación propiamente dicha e interpretación de los resultados obtenidos en dicha simulación. A continuación se definen brevemente estos puntos que serán abordados con detenimiento a lo largo del proyecto.

Especificación de modelos

El primer paso que hay que dar para la realización de la simulación es determinar aquellos modelos que se van a emplear. Esta especificación es de vital importancia, ya que del grado de detalle con el que dichos modelos se definan, dependerá que los resultados que se obtengan sean más o menos aproximados a las situaciones que se dan en la realidad. Para ello, se comienza por realizar un estudio teórico pormenorizado de los modelos que se van a utilizar.

En particular, en el proyecto que nos ocupa, se enmarca en los siguientes modelos:

- **Modelo de Cambio de Carril:** Este modelo pretende definir la forma en la que los vehículos llevan a cabo los cambios de un carril a otro. Se comienza por realizar un estudio de los modelos de cambio de carril ya existentes, desarrollando posteriormente un nuevo modelo. Este nuevo modelo incluye una duración temporal determinada para la realización de la maniobra, posibilidad de cambios de carril por cortesía y otros muchos detalles que se describen en el apartado 2.3.
- **Modelo de Generación de Vehículos:** establece cómo y cuándo se incorporan nuevos vehículos al escenario de simulación. En particular, se emplean funciones de generación de vehículos trapezoidales combinadas con funciones estadísticas, tal y como se desarrolla en el apartado 3.
- **Estadísticas:** van a proporcionar un formato ordenado y adecuado para la representación de los resultados obtenidos en la simulación. En el apartado 4 se detallan todas aquellas estadísticas desarrolladas en el proyecto.

Diseño de los modelos

Para la implementación de los modelos comentados con anterioridad en el simulador, es necesario diseñar una estructura de datos, además de las funciones que realicen lo propuesto en el modelo. Ambas cosas han de ser convenientemente diseñadas y documentadas.

La implementación de los modelos empleados en el simulador ha sido realizada mediante lenguaje C++, se ha creído conveniente desarrollar dichos modelos utilizando una metodología orientada a objetos denominada UML (Unified Modeling Language). Dicho lenguaje se basa principalmente en **diagramas de casos de uso**, **diagramas de clases** y **diagramas de paquetes** los cuales son convenientemente explicados en el apartado 5.

Por último, y con el objetivo de describir detalladamente la implementación mediante C++ de los modelos de cambio de carril, estadísticas y generación de vehículos se ha estimado oportuno incluir los diagramas de flujo asociados al simulador en el apartado 6.

Pruebas al modelo de simulación

Este punto consiste en la verificación del correcto funcionamiento del modelo de simulación así como de los nuevos modelos desarrollados en este trabajo. Para ello, se debe tomar en consideración la información proporcionada por la aplicación TRAMOS (Traffic and Transport Analysis Modeling and Optimization System) tal como son la descripción del viario, cajas semafóricas y funciones de generación. Además recibe otros datos a través de una interfaz de usuario diseñada y desarrollada en este proyecto. Así mismo, los resultados de la simulación son mostrados al usuario a través una interfaz.

El diseño y desarrollo de las interfaces necesarias para la comunicación del simulador con TRAMOS y con el usuario, también se encuentran dentro del alcance del proyecto.

Las pruebas se han dividido en dos grupos, el primer grupo tiene como objetivo comprobar y validar los modelos desarrollados, así como verificar la integración entre los modelos. El segundo conjunto de pruebas está destinado a mostrar los resultados de la aplicación del simulador en escenario reales, para ello se realizan diversas pruebas basadas en diferentes escenarios reales de la ciudad de Sevilla.

Interpretación de resultados

El último paso, aunque no el menos importante, consiste en el análisis y correcta interpretación de los resultados obtenidos en la simulación. Estos resultados son lo que se buscaba desde el principio, por lo que hay que prestar especial interés a la hora de interpretarlos. Para ello, los resultados ofrecidos por el simulador son empleados por el usuario para estudiar el comportamiento de los modelos y extraer las conclusiones que estime oportunas.

Para finalizar, a partir de los resultados se obtienen una serie de representaciones gráficas que facilitan la interpretación de los resultados, además de proporcionar ciertas comparativas que son de gran utilidad.

2 Modelo de cambio de carril

2.1 Introducción

Las maniobras de cambios de carril que llevan a cabo los vehículos en un tramo del viario resultan esenciales a la hora de la realización de una simulación microscópica de tráfico, ya que son muchas y de muy variada naturaleza las maniobras de este tipo que se producen en la vida real. Por lo tanto, el modelo debe tratar de reflejar todos los aspectos que resultan decisivos en la realización de los cambios de carril por parte de los vehículos que intervienen en la simulación.

Los cambios de carril se producen a causa de algún o algunos de los factores enumerados a continuación:

- Factores dependientes del propio vehículo: velocidad deseada, carril de preferencia para llegar al destino, realización de un aparcamiento...
- Interacción del vehículo con el resto de elementos que componen el escenario de simulación: vehículos adyacentes, señales de tráfico, semáforos, obstáculos...

Para llevar a cabo un cambio de carril, se suelen seguir las siguientes etapas:

- Toma de la decisión de realizar el cambio de carril, al concurrir algunos de los factores anteriores.
- Selección del carril hacia el que desea desplazarse el vehículo.
- Comprobación de que la maniobra puede realizarse con seguridad mediante la aceptación de la existencia de suficiente espacio disponible para la realización de la maniobra.
- Realización de la maniobra propiamente dicha.

Como se puede observar, la realización de una maniobra de cambio de carril requiere de todo un proceso bastante complejo, en el cual se deben tener en cuenta una gran cantidad de detalles que influirán decisivamente tanto a la hora de decidir si se va a llevar cabo la maniobra, como a la forma en que dicha maniobra se realizará.

2.2 Notación

A continuación se detalla la notación empleada para describir el procedimiento de cambio de carril, tanto en los modelos ya existentes como en el nuevo modelo desarrollado en este proyecto. El siguiente gráfico de la figura 2.1 muestra un ejemplo donde un vehículo (n) desea realizar una maniobra de cambio de carril (movimiento al carril adyacente). Por el carril adyacente se desplazan dos vehículos separados cierta distancia (vehículo n-1 y vehículo n+1).

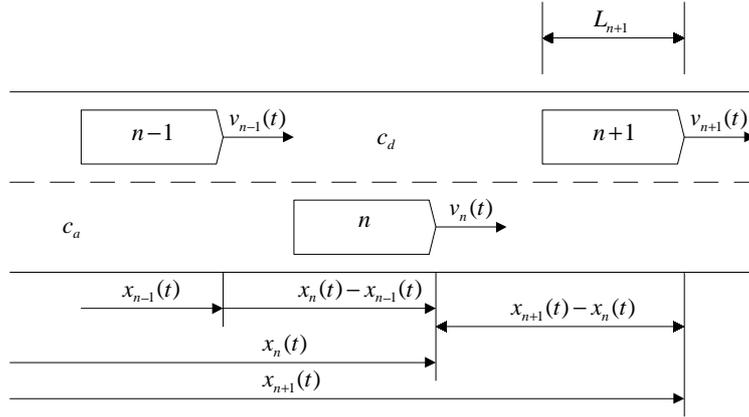


Figura 2.1. Notación cambio carril

La notación empleada es:

- c_a : Carril por donde se desplaza el vehículo n (carril actual).
- c_d : Carril destino o donde desea situarse el vehículo.
- $a_{n,max}^-$: Aceleración de frenada máxima que puede aplicar el vehículo n.
- $a_{n-1,max}^-$: Aceleración de frenada máxima que puede aplicar el vehículo n-1.
- $a_{n+1,max}^-$: Aceleración de frenada máxima que puede aplicar el vehículo n+1.
- $a_n(t)$: Aceleración del vehículo n en el instante t.
- $a_{n-1}(t)$: Aceleración del vehículo n-1 en el instante t..
- $a_{n+1}(t)$: Aceleración del vehículo n+1 en el instante t..
- $v_n(t)$: Velocidad del vehículo n en el instante t.
- $v_{n+1}(t)$: Velocidad del vehículo n+1 en el instante t.
- $v_{n-1}(t)$: Velocidad del vehículo n-1 en el instante t.
- $x_n(t)$: Posición del vehículo n en el instante t.
- $x_{n+1}(t)$: Posición del vehículo n+1 en el instante t.
- $x_{n-1}(t)$: Posición del vehículo n-1 en el instante t.
- n_c : Número de carriles en el tramo.
- $D_n^{seg}(t)$: Distancia de seguridad del vehículo n, en el instante t.
- $D_{n-1}^{seg}(t)$: Distancia de seguridad del vehículo n-1, en el instante t.
- $D_n^{inf}(t)$: Distancia de influencia del vehículo n, en el instante t.
- $D_{n-1}^{inf}(t)$: Distancia de influencia del vehículo n-1, en el instante t.
- L_n : Longitud del vehículo n.
- L_{n+1} : Longitud del vehículo n+1.

2.3 Revisión histórica de modelos de cambios de carril

En este apartado se va a realizar un estudio de algunos de los modelos de cambios de carril que se han venido implementando en la mayoría de los simuladores de tráfico. Dichos modelos han servido como base al nuevo modelo desarrollado y mucha de la terminología empleada en ellos también es la que se utilizará en lo sucesivo para explicar el nuevo modelo.

2.3.1 Modelo de cambio de carril de Gipps

El primer modelo de cambio de carril fue desarrollado por Gipps, que presentó un modelo de cambio de carril para ser empleado en los simuladores de tráfico microscópicos. El modelo fue diseñado para cubrir diferentes situaciones en el proceso de conducción por el viario, donde las señales de tráfico, obstáculos y vehículos pesados afectaban a la decisión del conductor de realizar el cambio de carril. El desarrollo del modelo está basado en un conjunto de condiciones que el conductor impone para realizar la maniobra.

Las condiciones de cada conductor pueden ser dispares ya que cada uno posee su propio objetivo para desplazarse desde X a Y con seguridad y confort en un tiempo determinado. La generalidad del objetivo planteado conduce a dividirlo en un número de objetivos específicos y limitados que son más fáciles de emplear.

Los objetivos no tienen por que ser consistentes entre sí. Un conductor desplazándose por el viario puede encontrar varios inconvenientes. El deseo de un conductor de mantener la velocidad puede chocar con la necesidad de situarse en el carril adecuado para realizar una maniobra de giro. Pueden surgir otros conflictos cuando existen carriles reservados.

El modelo fue diseñado para ser empleado conjuntamente con el modelo de seguimiento de vehículos de Gipps donde existe una aceleración de frenada que permite calcular la velocidad de seguridad con respecto al vehículo precedente.

$$v_n(t+T) \leq a_{n,max}^- T + \left(a_{n,max}^{(-)2} T^2 - a_{n,max}^- \left(2[x_{n+1}(t) - L_{n+1} - x_n(t)] - v_n(t)T - \frac{v_{n+1}^2(t)}{a^-} \right) \right)$$

donde, a^- corresponde con la estimación de la deceleración del vehículo n+1, realizada por el vehículo n.

La velocidad de seguridad está acotada superiormente por la velocidad deseada del conductor, para prevenir obstrucciones lejos de la zona de decisión del conductor.

Debido a la gran cantidad de conflictos que pueden existir entre los objetivos, se describen los factores que pueden influir sobre los conductores en el instante de realizar un cambio de carril.

1. Seguridad en el cambio de carril.

Por términos generales los vehículos no cambiarán de carril si existe algún riesgo de colisión. Antes de realizar la maniobra, el conductor verifica que el espacio disponible en el carril destino (gap) es el necesario. Incluso los conductores comprueban la velocidad de los vehículos situados por detrás que ocupan el carril destino. Sin embargo estas medidas de seguridad pueden variar dependiendo de muchos factores como el tipo de conductor, situación y proximidad de obstáculos en el carril.

2. Existencia de obstáculos permanentes.

Los conductores que se encuentran familiarizados con el viario intentan evitar verse atrapados por obstrucciones, como zonas de aparcamiento, seleccionando un carril que permita una circulación libre y con la menor obstrucción. El proceso de aprendizaje conduce a seleccionar carriles ignorando las ventajas que le ofrece el carril obstruido. Este comportamiento es modificado por la distancia que separa al conductor del obstáculo.

3. Existencia de carril reservado.

En algunos tramos del viario existen carriles destinados a la circulación de transporte público o bien carriles destinados a vehículos de alta ocupación. Los vehículos no destinados al uso de este tipo de carril tienden a emplearlo ya que ofrecen velocidades superiores. En general los vehículos destinados a emplear estos carriles tienden a desplazarse a los carriles adyacentes para evitar obstáculos. Los vehículos no destinados a emplear dichos carriles se ven obligados a emplearlo en circunstancias especiales como giros a la derecha.

4. Realización de giros.

Otro factor que afecta a la decisión del conductor de realizar un cambio de carril es el deseo de realizar un giro a la finalización del tramo. Mientras que la distancia entre el vehículo y el comienzo del giro sea grande el conductor no realiza el cambio de carril para situarse en el carril asociado al giro, pero su deseo de cambiar de carril aumentará en la medida que se aproxime al inicio del giro.

5. La presencia de vehículos pesados.

Los conductores intentan evitar ser atrapados detrás de vehículos pesados ya que su aceleración es muy baja. El efecto es mucho más acusado cuando el vehículo se mueve a velocidad lenta. El factor puede ser agravado por la existencia de un gran espacio delante del vehículo que permite al conductor aumentar la aceleración y aproximarse a su velocidad deseada.

6. Velocidad.

El último factor que afecta a la decisión de los conductores para realizar el cambio de carril es el tráfico en el carril actual o bien el tráfico en los carriles adyacentes. Los conductores observan si en los carriles adyacentes se les permite

umentar su velocidad. El factor se ve influenciado no sólo por los carriles adyacentes sino incluso por los carriles más alejados del vehículo.

Gipps propuso que el comportamiento de los conductores en el proceso de cambio de carril seguía un modelo donde los conductores buscan la mejor situación (carril) para realizar un giro. Mientras el giro se encuentre alejado no existe ningún tipo de efecto o influencia y el conductor se centra en mantener la velocidad deseada. A partir de cierta distancia el conductor comienza a ignorar las oportunidades de mejorar la velocidad y comienzan a estudiar los posibles cambios de carril para poder realizar el giro.

En la figura 2.2 se muestra el proceso de decisión donde se intentan cubrir todas las situaciones que el conductor puede encontrarse en el viario.

Las preguntas que contiene el diagrama de dicha figura pueden clasificarse en dos tipos: objetivas y subjetivas. Las preguntas objetivas tales como ¿Se encuentra el vehículo en un carril reservado a transporte público? Son fáciles de contestar y afectan a diferencias locales en el modelo. Por otra parte las preguntas subjetivas pueden sugerir la redefinición del modelo cuando se aplique al entorno.

Los principales puntos del modelo son descritos a continuación:

Selección del carril

El modelo está basado en dividir el tramo en N carriles disponibles para el tráfico. Estos carriles son etiquetados con número creciente desde la acera hasta la mediana del tramo. Para cada conductor el modelo define el carril preferido, c_p y el carril destino, c_d , en función del carril donde se encuentra actualmente c_a .

El carril preferido es considerado como el carril adyacente al actual y más cercano al carril destino del conductor, es decir:

$$c_p = \begin{cases} c_p + 1 & \text{si el conductor desea realizar un giro hacia la izquierda} \\ c_p - 1 & \text{si el conductor desea realizar un giro hacia la derecha.} \end{cases}$$

El carril destino es aquel al cual el conductor desea desplazarse. Inicialmente el carril preferido y destino coinciden, pero el modelo contempla la posibilidad de cambiar a un carril que mejore las condiciones.

Admisibilidad del proceso de cambio de carril

La primera pregunta que el conductor considera es si es posible moverse al carril destino. El proceso actúa como un primer filtro donde se eliminan las situaciones donde el cambio de carril no puede realizarse. La respuesta afirmativa a la pregunta sobre la posibilidad de cambio es obtenida cuando un conjunto de condiciones son satisfechas.

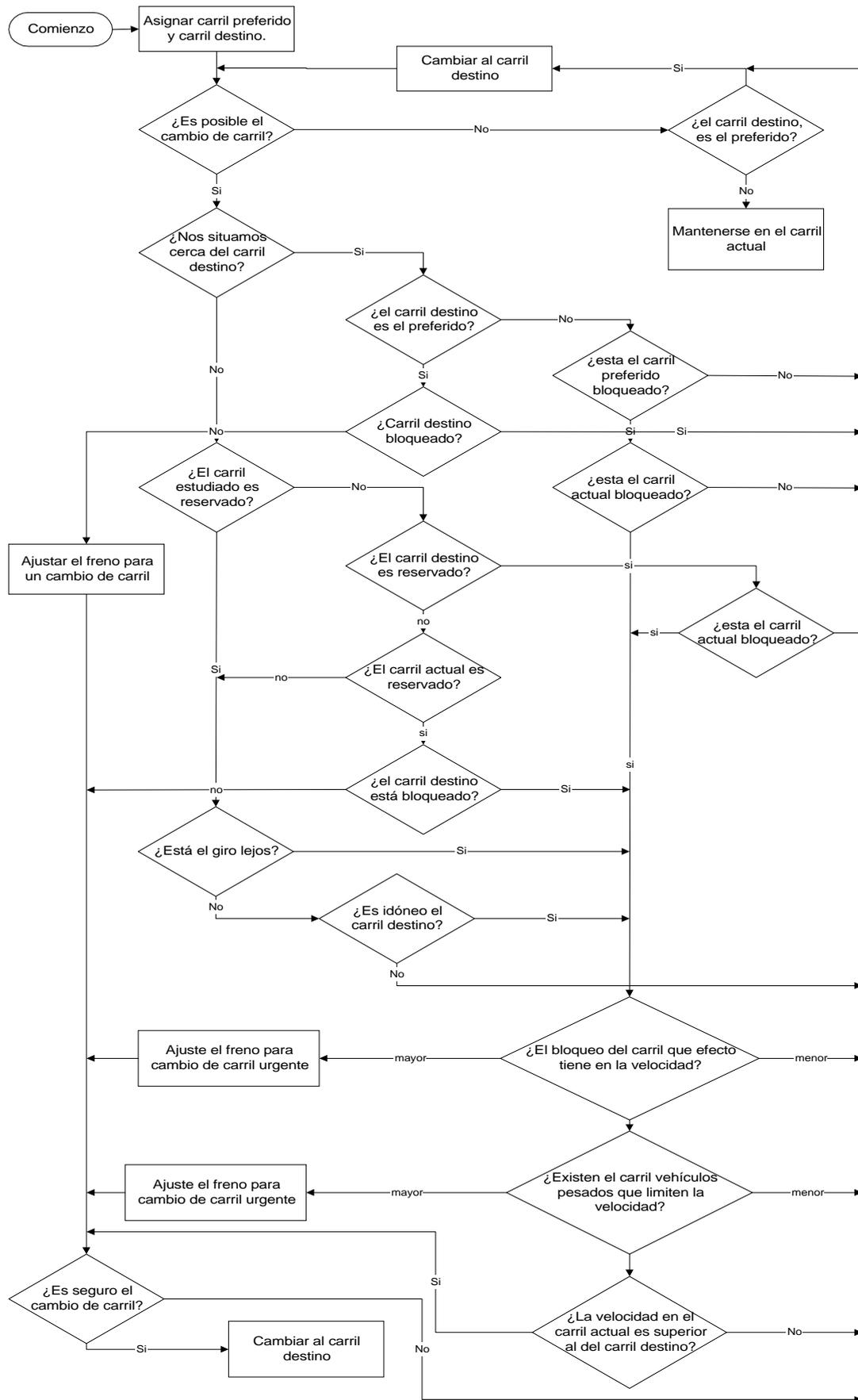


Figura 2.2. Diagrama de cambio de carril del modelo de Gipps

El carril destino debe ser un carril disponible al tráfico, es decir:

$$1 \leq c_d \leq N$$

La sección del tramo donde se realiza el cambio de carril, en el carril destino debe estar libre de obstrucciones y libre de otros vehículos.

El proceso de implementación, obtiene una aceleración del vehículo que le permita cambiar de carril con seguridad. El cambio de carril no será realizado cuando la deceleración que debe aplicar el conductor sea inaceptable e insegura.

$$d = \frac{v_n(t+T) - v_n(t)}{T}$$

Comportamiento del conductor en la aproximación del giro

Cuando el cambio de carril es posible, el modelo procede a preguntar si el conductor se encuentra cerca del giro que desea realizar.

Cuando la respuesta es afirmativa, el conductor cambiará al carril preferido siempre que pueda realizarlo con seguridad y el carril no esté bloqueado antes del giro. La posibilidad de cambiar a un carril distinto al actual y al preferido sólo es admisible cuando ambos carriles se encuentren bloqueados. El término de aproximación es muy subjetivo por ello se considera que este modelo entra en ejecución cuando el vehículo se encuentra a unos 10 segundos del giro.

Cambio de carril urgente

Los intentos de cambios de carril aumentan cuando el vehículo se aproxima al giro y no se encuentra en el carril correcto. El sentido de la urgencia está reflejado en la aceptación de espaciados más pequeños junto a una deceleración más elevada.

La deceleración que el conductor aplica en este estado es:

$$a_n(t) = \left[2 - \frac{(X_{giro} - x_n(t))}{10} V_n^{des} \right] a_{n,max}^-$$

donde, X_{giro} es la posición del giro.

Carriles reservados

Los tramos pueden contener diferentes tipos de carriles, estos pueden ser de utilización conjunta o restringida. En la terminología este tipo de carriles se denominan carriles reservados y sólo pueden ser empleados por vehículos que cumplen ciertas condiciones. Los vehículos especiales que pueden circular por los carriles reservados no tienen restringida la circulación por otros carriles.

Invasión de vehículos en carriles reservados

Aunque normalmente los carriles reservados sólo pueden ser empleados por vehículos especiales, existe la posibilidad de utilización en caso de obstrucciones en otros carriles.

Salida de vehículos de los carriles reservados

La posibilidad de invasión de vehículos en los carriles reservados conduce a definir una situación de salida de estos carriles. El modelo debe contener la lógica para asegurar que los vehículos no permitidos abandonan el carril. Por ello, el conductor debe comprobar continuamente si es posible realizar el cambio de carril para abandonar el no permitido.

Comportamiento del conductor a media distancia

Si el conductor está muy separado del giro, la dirección en la cual desea girar no influye sobre su comportamiento. Pero cuando el giro se aproxima, comienza a tener influencia y a modificar el comportamiento del conductor y por tanto no todos los carriles son aceptables.

El modelo debe definir la distancia media, podemos considerar que un carril es aceptable si,

$$(c_p - c_d)(c_a - 1)(c_a - N) = 0$$

Ventajas del carril actual y carril destino

Si ninguno de los factores considerados anteriormente han forzado a los conductores a plantearse un cambio de carril, éste es libre de plantearse las ventajas relativas del carril actual y carril destino. Al principio el conductor considera las obstrucciones del carril actual y carril destino y después selecciona el carril en el cual la obstrucción tiene un menor efecto.

Presencia de vehículos pesados

Los conductores observan la presencia de vehículos pesados, seleccionando aquellos carriles que le proporcionan una mayor velocidad. El método busca aquellos carriles donde no existan vehículos pesados evitando la disminución de velocidad.

El efecto del vehículo precedente

Los vehículos realizan cambio de carril por varias opciones, entre ellas está la posibilidad de obtener beneficios en la velocidad. La ventaja obtenida es subjetiva y depende del carril actual, carril destino y tipo de vehículo, pudiendo ser negativo en ciertas circunstancias.

Gipps propuso como ventaja suficiente como para cambiar de carril, desplazándose a carriles más interiores (cerca de la mediana), cuando el aumento de la velocidad es superior a 3.6 km/h. Los vehículos se desplazarán a carriles más exteriores (cerca de la acera) cuando la velocidad desciende al menos 0.1 m/s.

Seguridad

Cuando el conductor ha decidido cambiar de carril, la cuestión final es si puede hacerlo con seguridad. Esta pregunta se destina al final porque el nivel de seguridad exigido por el conductor puede variar según las condiciones y la urgencia. Por ejemplo, un conductor que no se encuentra en el carril destino y se acerca al giro a realizar, aceptará un espacio entre vehículos inferior que cuando se encuentra muy alejado del objetivo.

Cambio al carril destino

Si el vehículo ha rechazado cambiar a su carril seleccionado, considera la posibilidad de cambiar al carril en sentido contrario al carril destino modificando en tal caso el carril destino propiamente.

2.3.2 Modelo de cambio de carril de Yang y Koutsopoulos

El modelo de cambio de carril desarrollado por Yang y Koutsopoulos e implementado en el simulador MITSIM, está basado en el modelo de cambio de carril de Gipps y estructurado en tres pasos:

- 1.- Comprobar si el cambio es necesario y definir el tipo de cambio a realizar.
- 2.- Seleccionar el carril deseado.
- 3.- Realizar el cambio de carril si el espacio disponible (gap) es aceptable.

MITSIM clasifica los cambios de carril en obligatorio y por mejora. Los cambios de carril obligatorio (Mandatory Lane Change, MLC) se producen cuando el conductor necesita cambiar de carril por:

- Necesidad de cambio hacia otro carril para mantener la ruta del conductor.
- Cambio al carril apropiado antes de realizar un giro en una intersección.
- Cambios de carril para la incorporación desde carriles de aceleración.
- Cambios de carril para la incorporación a carriles de deceleración.
- Cambios provocados por bloqueo de carril.
- Evitar la entrada a carriles restringidos.
- Responder a mensajes de señales.

La necesidad de realizar un cambio de carril obligatorio aumenta en la medida que se aproxima al inicio de la maniobra, esto supone que la probabilidad de realizar un

cambio de carril aumenta cuanto más próximo se encuentre del comienzo de la maniobra. Una función que representa esta condición fue proporcionada por Yang y Koutsopoulos,

$$p_n = \begin{cases} 1 & x_n \leq x_0 \\ \exp\left[\frac{-(x_n - x_0)^2}{\sigma_n^2}\right] & x_n > x_0 \end{cases}$$

donde, p_n es la probabilidad que el vehículo n comience una maniobra cambio de carril obligatoria; x_n es la distancia mínima desde el final de tramo o fin del carril a partir de la cual el vehículo debe realizar obligatoriamente el cambio de carril; x_0 es la distancia crítica asociada a la posición del mensaje de la señal (como final de la salida...); y σ_n^2 es una variable definida a continuación:

$$\sigma_n^2 = \alpha_0(1 + \alpha_1 m_n + \alpha_2 K)$$

donde m_n es el número de carriles que el vehículo necesita cruzar para situarse en el carril correcto; K es la densidad de tráfico en el tramo y donde α_0, α_1 y α_2 son parámetros.

Kazi propuso una formulación alternativa para determinar la probabilidad de realizar un cambio de carril obligatorio, la cual incluye y depende de un término aleatorio asociado a cada conductor. La expresión es:

$$p_t(MLC | u_n) = \frac{1}{1 + e^{-X_n(t)\beta - \alpha u_n}}$$

donde,

X_n , vector de variables que afectan a las condiciones de un cambio de carril obligatorio.

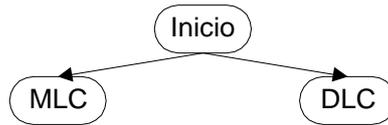
β , vector de parámetros.

u_n , término de aleatoriedad asociado a cada conductor, se asume que se distribuye según una normal.

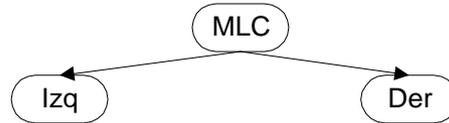
α , parámetro de u_n .

Las siguientes figuras muestran la estructura del modelo. Los procesos y decisiones seguidas en el cambio de carril son representados por rectángulos y elipses respectivamente.

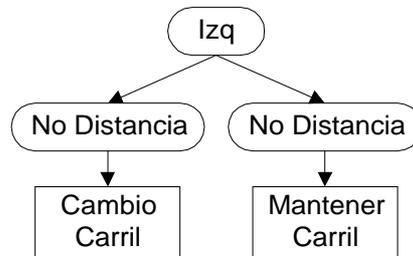
El modelo comienza seleccionando el tipo de cambio de carril que intentará realizar, este proceso de selección se realiza aplicando a la expresión anterior.



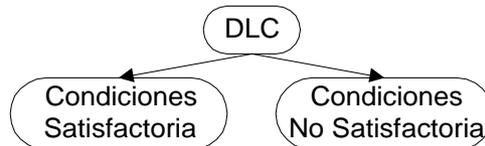
Seleccionado el cambio de carril obligatorio, se procede a seleccionar el carril destino, que puede ser el izquierdo o el derecho. La selección de un carril depende del carril destino del conductor.



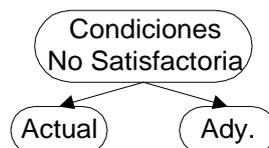
La siguiente tarea es comprobar si existe espacio suficiente para realizar la maniobra de cambio de carril, en caso afirmativo se realiza la maniobra desplazándose el vehículo al nuevo carril. Si la distancia existente no es la suficiente para realizar el cambio de carril el vehículo desiste del proceso, para volverlo a intentar posteriormente.



El proceso de cambio de carril por mejora, se resume en decidir si las condiciones del tráfico son satisfactorias o no satisfactorias.



Las condiciones son satisfactorias cuando los requerimientos exigidos por el conductor son correctos, cada conductor puede desear diferentes requisitos tales como la velocidad deseada, presencia de vehículos pesados por delante y detrás del vehículo, etc. Si las condiciones no son satisfactorias, el conductor compara las condiciones del carril actual con los carriles adyacentes, los factores a tener en cuenta pueden ser el estado del tráfico y pavimento en los carriles adyacentes, velocidad deseada, densidad de tráfico en los diferente carriles, velocidades relativas con respecto a vehículos seguidores en otros carriles, la presencia de vehículos pesados en diferentes carriles, etc.



Cuando las condiciones de los carriles adyacentes son mejores a las actuales se procede a seleccionar el nuevo carril para posteriormente comprobar si la distancia es

segura para realizar el cambio de carril. El proceso de comprobación de la distancia depende de varios factores tales como velocidad actual del vehículo, velocidad actual del futuro vehículo precedente y sucesor situados en el carril destino, distancia restante a la finalización del tramo, retraso, etc.

En los modelos de cambios de carril por mejora (Discretionary lane change, DLC) las decisiones de cambiar de carril están basadas en las condiciones del tráfico en el carril actual y carril destino.

El modelo de cambio de carril por mejora considera que los conductores realizan cambios de carril para mantener o mejorar las condiciones de conducción. Un vehículo realiza un cambio de carril para obtener:

- a) Incremento de la velocidad.
- b) Para adelantar un vehículo lento o pesado.
- c) Evitar carriles conectados a salidas de la calzada.

Donde la decisión de cambiar de carril es determinada por las características del conductor y los estímulos externos.

Entre las características del conductor que pueden afectar al cambio de carril se encuentran:

- Agresividad del conductor
- Velocidad deseada en flujo libre.
- Máximo y normal aceleración y deceleración.
- Impaciencia del conductor.

Los estímulos externos pueden proceder del mismo carril o bien de carriles adyacentes. Entre los estímulos externos consideramos dos grupos:

1.- Estímulos procedentes del mismo carril.

- Distancia relativa entre el vehículo y el vehículo predecesor.
- Velocidad relativa entre el vehículo y el predecesor.
- El vehículo predecesor realice un cambio de carril.
- El vehículo predecesor realiza un cambio de sentido a izquierda o derecha.
- Existencia de vehículos pesados.
- Vehículos estacionados.
- Agresividad del conductor seguidor.

2.- Estímulos procedentes de carriles adyacentes.

- Distancia relativa entre el vehículo y el vehículo candidato a ser su predecesor en el nuevo carril.
- Velocidad relativa entre el vehículo y el vehículo candidato a ser su predecesor en el nuevo carril.
- El vehículo candidato a ser el nuevo predecesor está realizando un cambio de sentido.
- El vehículo candidato a ser el nuevo predecesor es un vehículo pesado o está estacionado.

La decisión de cambio de carril está basada en condiciones del tráfico de los carriles adyacentes y del propio carril donde el vehículo se encuentra. Por ejemplo, si un vehículo tiene una velocidad inferior a la deseada, comprueba la posibilidad de aumentar su velocidad en los carriles adyacentes. Existen varios parámetros, como factor de impaciencia, factor de indiferencia de velocidad, que determinan si la velocidad actual es lo suficientemente baja y la velocidad de los vehículos adyacentes es elevada para considerar un cambio de carril.

Kazi modela esta situación a través de un modelo Logit binario,

$$p_t(CNS \mid u_n) = \frac{1}{1 + e^{-X_n(t)\beta - \alpha u_n}}$$

donde,

X_n , vector de variables que expresan las condiciones de conducción del carril.

β , vector de parámetros.

u_n , término de aleatoriedad asociado a cada conductor, se asume que se distribuye según una normal.

α , parámetro de u_n .

Si las condiciones no son satisfactorias se compara la función de utilidad del carril con los carriles adyacentes (carriles candidatos), seleccionando aquel carril que ofrece una mayor mejora de las condiciones.

2.3.3 Selección de carril deseado

En el proceso de cambio de carril el conductor debe determinar el conjunto de carriles admisibles. Un carril es admisible cuando cumple una serie de criterios tales como, regulación del carril al que se desea ir, privilegios del carril, conexión del carril con otros tramos, estado de las señales asociadas al carril, condiciones de tráfico, velocidad deseada por el conductor y señales que rigen en el carril, por ejemplo velocidades máximas y mínimas. En caso de selección de carril para modelos de cambio discrecional, un carril es admisible cuando mejora o iguala las condiciones del carril actual.

2.3.4 Espacio disponible para realizar el cambio de carril (Modelos de aceptación de Gap)

Una vez seleccionado el carril, se debe examinar la distancia existente para poder realizar la maniobra. El proceso de comprobación se basa en examinar la distancia existente entre el vehículo que desea realizar el cambio de carril y los futuros vehículos sucesor y predecesor “n-1” y ”n+1” respectivamente. La siguiente gráfica muestra los parámetros implicados en proceso de cambio de carril que desea realizar el conductor del vehículo n.

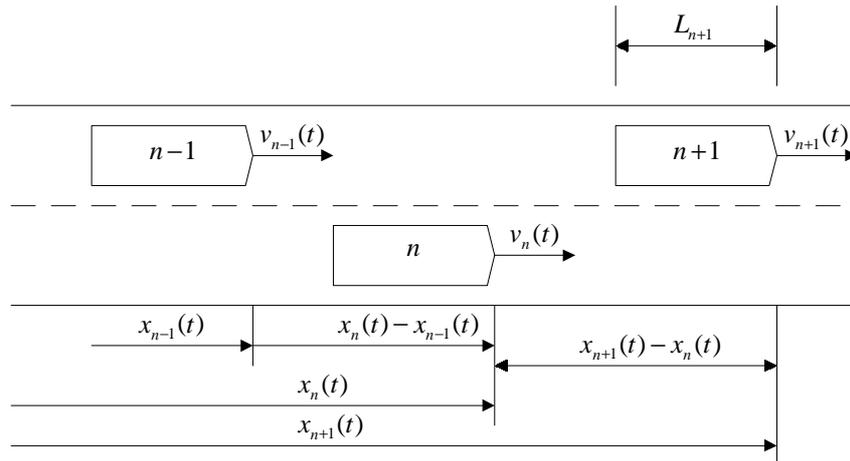


Figura 2.3. Representación del GAP

Los parámetros que influyen en la decisión de cambio de carril son la distancia de separación entre el vehículo n y n-1 (llamada gap trasero) y expresada como $g_n^t(t) = x_n(t) - L_n - x_{n-1}(t)$, la distancia de separación entre el vehículo n+1 y el vehículo n (gap delantero) y expresado por, $g_n^d(t) = x_{n+1}(t) - x_n(t) - L_{n+1}$. También es importante la velocidad de los vehículos, así el conductor para realizar la maniobra de cambio de carril no sólo tiene que comprobar el espacio disponible sino la velocidad de los vehículos del carril destino. Otros parámetros de menor consideración son la distancia y velocidad del vehículo con respecto a otros vehículos que circulan por el mismo carril.

El proceso de cambio de carril comienza tomando la decisión de cambiar de carril, posteriormente si la decisión es positiva se procede a examinar el espacio disponible para asegurar una maniobra segura, para ello se comprueba la distancia con el vehículo delantero y posterior respectivamente.

La expresión que determina si un cambio de carril es seguro, es decir si la distancia del vehículo al vehículo predecesor y al sucesor del nuevo carril, queda determinada por la expresión:

$$g_n^i(t) = g_{n,\text{inf}}^i + \epsilon_n^i \quad i = \{d, t\}$$

donde $g_{n,\text{inf}}^i(t)$ es una cota inferior de la distancia que el conductor n, en el instante t, considera como aceptable para realizar el cambio de carril; $g_n^i(t)$ es el gap medio; y

ε_n^i es un término aleatorio que varía según el conductor y está distribuido según una normal.

Los primeros modelos de aceptación de gap fueron desarrollados en la década de los 60 y están basados en la presunción de existencia de un gap crítico, es decir una distancia mínima de seguridad. Los primeros trabajos de Herman y Weiss establecieron que el gap crítico seguía una distribución exponencial. Otros autores como Drew establecieron que seguía una distribución logarítmica y Miller que se distribuía según una normal. Ninguno de los autores anteriores recogieron el efecto sobre el conductor de los intentos de cambio de carril fallido.

Los primeros trabajos donde se incorporó la variable de intentos de cambio de carril fallidos se deben a Mahmassani y Sheffi que asumieron que el gap crítico seguía una distribución normal. La función de gap desarrollada dependía de un conjunto de variables explicativas que permitía hacer depender el gap crítico de los intentos fallidos de cambio de carril.

En el modelo general de cambio de carril obligatorio (MLC), el gap aceptable para realizar la maniobra de cambio de carril disminuye a medida que nos acercamos al final de carril o fin de tramo. El modelo de aceptación de gap, tiene la siguiente forma:

$$g_n^i(t) = \varepsilon_n^i + \begin{cases} g_{n,\text{sup}}^i & x_n \geq x_{\text{max}} \\ g_{n,\text{inf}}^i + (g_{n,\text{sup}}^i - g_{n,\text{inf}}^i) \frac{(x_n - x_{\text{min}})}{x_{\text{max}} - x_{\text{min}}} & x_{\text{min}} < x_n < x_{\text{max}} \\ g_{n,\text{inf}}^i & x_n \leq x_{\text{min}} \end{cases}$$

donde i representa si el parámetro es para el gap delantero o gap trasero; g_n^i es el gap mínimo que el conductor n acepta para realizar un cambio de carril; $g_{n,\text{inf}}^i$ y $g_{n,\text{sup}}^i$ son respectivamente cotas inferiores y cotas superiores, x_n posición actual del vehículo, x_{min} y x_{max} son distancias que definen el rango de variación del gap crítico desde $g_{n,\text{inf}}^i$ hasta, $g_{n,\text{sup}}^i$, y ε_n^i es un término de error.

Yang propone un modelo, implementado en MITSIM que a diferencia del anterior tiene en consideración las velocidades relativas de los futuros vehículos, precedente y seguidor.

Para cambio de carril por mejora propone:

$$g_n^d = \max(g_{n,\text{inf}}^d, g_{n,\text{inf}}^d + \beta_1 v_n + \beta_2 (v_n - v_{n+1}) + \varepsilon_n^d)$$

$$g_n^t = \max(g_{n,\text{inf}}^t, g_{n,\text{inf}}^t + \beta_1 v_{n-1} + \beta_2 (v_{n-1} - v_n) + \varepsilon_n^t)$$

Para el cambio de carril obligatorio la expresión es:

$$g_n^d = \max(g_{n,\text{inf}}^d, g_{n,\text{inf}}^d + [\beta_1 v_n + \beta_2 (v_n - v_{n+1})](1 - e^{-\lambda_n^2}) + \varepsilon_n^d)$$

$$g_n^t = \max(g_{n,\text{inf}}^t, g_{n,\text{inf}}^t + [\beta_1 v_{n-1} + \beta_2 (v_{n-1} - v_n)](1 - e^{-\lambda_n^2}) + \varepsilon_n^d)$$

donde, β_1 y β_2 son parámetros del modelo.

El parámetro β_2 es una medida del grado de importancia que tiene la velocidad relativa e influye sobre la pendiente de la recta que representa el gap, es decir a valores mayores de, β_2 el gap para velocidades relativas negativas es superior. El parámetro β_1 mide la relación entre la velocidad del vehículo en el cambio de carril.

Los últimos estudios se basan en obtener los valores inferiores, $g_{n,\text{inf}}^i$, y superiores, $g_{n,\text{sup}}^i$, de la distancia de separación entre los vehículos. Entre los estudios se encuentran los realizados por Yang y Koutsopoulos, los cuales aplican un modelo logit para la obtención de las distancias:

$$g_{n,\text{inf}}^d = e^z$$

$$z = 2.72 - 0.555u_n + \varepsilon_n^d(t)$$

donde, $g_{n,\text{inf}}^d$, es el gap mínimo, u_n es un término aleatorio, constante para ciertos conductores y distribuido según una normal y $\varepsilon_n^{\text{lead}}(t) \approx N(0, 1.6^2)$ término aleatorio.

Para el $g_{n,\text{inf}}^t$ se tiene,

$$g_{n,\text{min}}^t = e^z$$

$$Z = -9.32 + 0.117 \times \min(v_{n+1}(t) - v_n(t), 10) + 0.1174 \times \max(v_{n+1}(t) - v_n(t), -10) + 1.57\delta_n^{\text{gap}}(t) + 1.88 \times \ln(L_n^{\text{rem}}(t)) + 1.90u_n + \varepsilon_n^t(t)$$

donde,

$$\delta_n^{\text{gap}}(t) = \begin{cases} 1 & \text{si retraso}(t) = 0 \\ 0 & \end{cases}$$

retraso(t), tiempo transcurrido desde que comenzó el cambio de carril.

L_n^{rem} , distancia restante hasta el punto de finalización de cambio de carril.

$$\varepsilon_n^{\text{lead}}(t) \approx N(0, 1.3^2)$$

Los valores obtenidos son utilizados para obtener la probabilidad de que se realice un cambio de carril condicionado a que la distancia sea aceptable:

$$P_n(\text{Cambio de carril} \mid \text{distancia}) = \frac{1}{1 + e^{1.90 - 0.52 \text{delay}_n(t)}}$$

El modelo desarrollado por Yang y Koutsopoulos, no puede ser aplicado al caso de modelos de fusión de carriles y modelos de cortesía donde el conductor cede el paso al vehículo que desea incorporarse al carril.

Kazi y Ahmed amplían el modelo anterior, expresando el gap crítico como:

$$g_n^i(t) = e^{X_n^i(t)\beta^i + \alpha^i v_n + \varepsilon_n^i(t)} \quad i=\{d,t\}$$

donde α^i es un parámetro de v_n (valores aleatorios asociados a cada conductor).

La forma exponencial dada al gap garantiza que el gap crítico no será negativo. Otra característica de la formulación, es la inclusión de conductores con rasgos conservadores, el cual que no está dispuesto a aceptar algunos tamaños de gap. Esta especificación es recogida en el producto de α^i y v_n .

La probabilidad de que un conductor acepte el espacio para poder realizar un cambio de carril viene expresado como:

$$P_t(\text{AceptaciónGAP}|v_n) = P_t(\text{AceptacionGAP}^d|v_n)P_t(\text{AceptacionGAP}^l|v_n) = P(g_n^d(t) > g_n^{\text{crítico},d}(t)|v_n)P(g_n^t(t) > g_n^{\text{crítico},t}(t)|v_n)$$

2.3.5 Modelo de fusión obligatoria (Merging Model)

Los modelos de fusión obligatoria son utilizados cuando el conductor no ha encontrado en cada uno de los intentos realizados el espacio necesario para realizar la maniobra de cambio de carril obligatorio. Este efecto suele producirse en los viarios donde la densidad de tráfico es elevada con velocidades de desplazamiento muy pequeñas, y por tanto las condiciones no son las más adecuadas para un cambio de carril.

Los modelos de fusión obligatoria pretenden facilitar las maniobras de cambio de carril obligatorio mediante la creación del espacio suficiente entre los vehículos de tal forma que permita la incorporación del vehículo al carril.

El modelo de fusión obligatoria suele aplicarse cuando al vehículo, después de varias comprobaciones de espacio, le ha sido imposible cambiarse de carril y la distancia al punto donde el cambio de carril debe estar totalmente finalizado es muy pequeña. El procedimiento se encarga de actualizar la aceleración y velocidad de los vehículos del carril adyacente (futuro vehículo precedente y futuro vehículo predecesor) para que la distancia entre ellos aumente la separación. Algunos modelos comprueban el espacio y actúan sobre la aceleración del futuro vehículo seguidor, otros provocan la actualización de aceleración de los vehículos situados en el carril adyacente y el mismo carril.

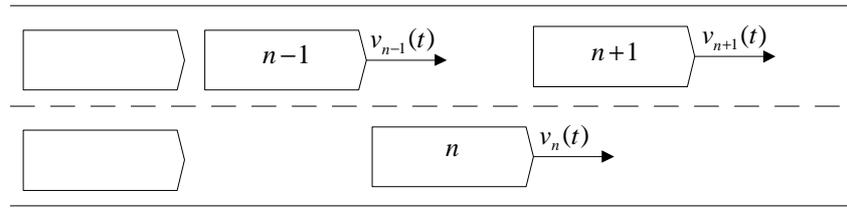


Figura 2.4. Modelo de fusión de Kazi

Kazi propone un modelo de fusión obligatorio. Sea, $S_n(t)$ el estado del vehículo n en el instante t . Usando un modelo logit, se define la probabilidad de realizar un cambio de carril con el modelo de fusión obligatorio condicionado a un conjunto de variables aleatorias como:

$$P_n(\text{Cambio} / v_n) = \frac{1}{1 + e^z}$$

$$z = -X_n(t)\beta - \alpha u_n$$

donde $X_n(t)$ representa un vector de variables que definen las características del modelo de fusión, α y β son parámetros de las variables asociadas al vector del modelo.

Las variables que definen las características del modelo incluyen:

Velocidad relativa del conductor leader, sólo cuando la velocidad del leader es baja. Cuando la velocidad del futuro leader es baja, el conductor tiende a descender la velocidad ajustándola a la velocidad del leader, para centrarse en la velocidad del futuro seguidor.

Velocidad relativa con el conductor seguidor, sólo cuando la velocidad del vehículo seguidor es elevada, el conductor tiende a acelerar el vehículo e intentar el cambio de carril siempre que la velocidad del vehículo leader no sea inferior al vehículo seguidor.

A medida que el conductor se aproxima a un punto donde el cambio de carril deseado debe ser completo aumenta la ansiedad y la necesidad de cambiar de carril lo que se traduce en un aumento de la probabilidad de realizar el cambio de carril.

El retraso, tiempo que el conductor debe esperar hasta aplicar las condiciones de cambio de carril, afecta a la agresividad o necesidad de cambio de carril.

Gap total, influye en la probabilidad de realizar el cambio de carril.

En cada instante se asume que el conductor:

- Evalúa las condiciones del tráfico en el carril destino para decidir si se intenta un cambio de carril forzado.
- El conductor intenta comunicarse con el conductor seguidor para saber si este le cederá el paso o por el contrario aumentará su velocidad impidiendo

el cambio de carril. Si el conductor seguidor cede el paso, el cambio de carril será finalizado con éxito y el proceso se terminará.

La estructura del proceso de cambio de carril incluyendo el modelo de cambio obligatorio es la mostrada en la siguiente figura 2.5:

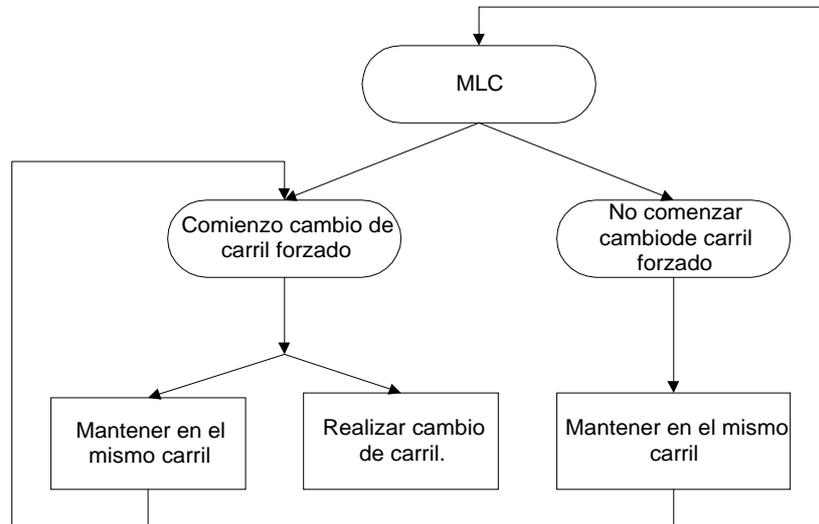


Figura 2.5. Estructura cambio carril obligatorio

2.3.6 Proceso de cambio de aceleración durante cambio obligado de carril

Frecuentemente los conductores tienen que ajustar su aceleración para la realización de maniobras de cambio de carril. Se consideran los siguientes casos:

1.- Cambio de carril sin cambio de aceleración.

Es el caso más simple e ideal donde el cambio de carril puede ser realizado sin cambios en la aceleración del vehículo y de otros vehículos.

2.- El vehículo necesita acelerar.

El lag no cumple los mínimos requerimientos y hay suficiente distancia entre el nuevo vehículo predecesor y el vehículo sucesor en el carril destino de la maniobra.

El lag y el gap no cumplen los requerimientos mínimos pero hay espacio suficiente entre el nuevo vehículo predecesor y el vehículo sucesor en el carril destino de la maniobra.

Incrementando la aceleración del vehículo puede ayudar a encontrar los requerimientos suficientes para completar la maniobra de cambio de carril. El movimiento de aceleración del vehículo es restringido por el espacio existente entre él y el vehículo predecesor, la distancia restante para la finalización del tramo y el tiempo necesario para completar la aceleración requerida en la maniobra.

3.- El vehículo necesita frenar.

El proceso de frenada para la maniobra de cambio de carril puede ser debido a 2 situaciones:

El lag no cumple los mínimos requerimientos y hay suficiente distancia entre el nuevo vehículo sucesor y el vehículo sucesor en el carril destino de la maniobra.

El lag y el gap no cumplen los requerimientos mínimos pero hay espacio suficiente entre el nuevo vehículo predecesor y el vehículo sucesor en el carril destino de la maniobra.

4.- Los vehículos en el carril destino cambian su velocidad.

En algunos casos el cambio de carril es posible si el futuro vehículo sucesor frena para que exista espacio suficiente (cortesía del conductor). Los modelos introducen la probabilidad de cortesía que indica en qué porcentaje un vehículo frena para permitir que otro vehículo cambie de carril.

5.- Los vehículos realizan una parada total.

La imposibilidad del vehículo de realizar un cambio de carril provoca que éste se detenga antes de finalizar su desplazamiento por el tramo para complementar la maniobra.

2.4 Diseño de un modelo de cambio de carril

El modelo de cambio de carril aquí desarrollado, al igual que los comentados anteriormente, se basa en la decisión que toma en cada instante un vehículo de seguir por el carril por el que se encuentra circulando actualmente, o si por el contrario, decide realizar una maniobra de cambio de carril que lo lleve a circular por un carril distinto del mismo tramo en el que se encuentra.

La decisión de llevar a cabo un cambio de carril puede deberse a diferentes motivos, entre los que cabe destacar el querer mantener o alcanzar una velocidad determinada, el situarse en el carril adecuado para seguir una ruta predefinida o realizar un giro determinado al llegar al final del tramo, el evitar un obstáculo que se encuentra en su carril o el situarse en el carril adecuado para realizar una maniobra de aparcamiento. Todo esto lleva a la definición de dos tipos diferentes de cambios de carril, obligatorios y por mejora, que serán estudiados con detenimiento en siguientes apartados.

2.4.1 Procedimiento general de cambio de carril

En este apartado se pretende dar una visión general del procedimiento que se lleva a cabo para realizar una maniobra de cambio de carril, así como de las condiciones que deben verificarse para que se acepte dicha realización. Estas condiciones varían ostensiblemente según el tipo de cambio que el vehículo quiera llevar a cabo.

Se distinguen dos tipos distintos de cambios de carril, en función de que el mismo tenga carácter obligatorio, o bien, se produzca para mejorar las condiciones de circulación del vehículo. La dependencia de realizar un tipo u otro de cambio de carril, está fuertemente influenciada por la distancia a la que se encuentre el vehículo del final del tramo.

Para llevar a efecto cualquiera de estos cambios, es necesario comprobar la seguridad del proceso, es decir, que se cumple una distancia mínima, denominada *gap*, con los vehículos que se encuentra el vehículo que desea cambiar en el nuevo carril, así como otra serie de condiciones.

La idea fundamental del algoritmo para el cambio de carril desarrollado se basa en la duración temporal de dicha maniobra. Es decir, los cambios de carril no son instantáneos, sino que tienen una duración predeterminada. Para ello, en el momento de inicio del cambio se crea un vehículo ficticio que se mantiene en el carril original mientras dure la maniobra, siendo eliminado una vez que haya finalizado la misma.

Todo este procedimiento general de cambio de carril se describe a grandes rasgos en la figura 2.6, en la cual se ha empleado la notación para diagramas de flujo expuesta en el Apéndice A.

En este diagrama de flujo se aprecia que en primer lugar se comprueba si el vehículo que está siendo objeto de estudio está realizando ya una maniobra de cambio de carril, para que en el caso de ser así, se lleve a cabo la finalización del proceso si se hubiera cumplido el tiempo destinado para ello.

Cuando el vehículo no está cambiando de carril, se comprueba si el vehículo debe realizar un cambio de carril obligatorio. Si esto es así y se dan las condiciones oportunas para ello, se procede a iniciar la maniobra correspondiente a dicho cambio de carril.

En el caso de que el cambio de carril obligatorio no se lleve a efecto, se comprueba si es conveniente que el vehículo realice un cambio de carril por mejora. De ser así, se verifica que se den las condiciones apropiadas para realizarlo, y si es así, da comienzo el cambio de carril por mejora.

Todos estos procedimientos son estudiados con muchos más detalles en los siguientes apartados.

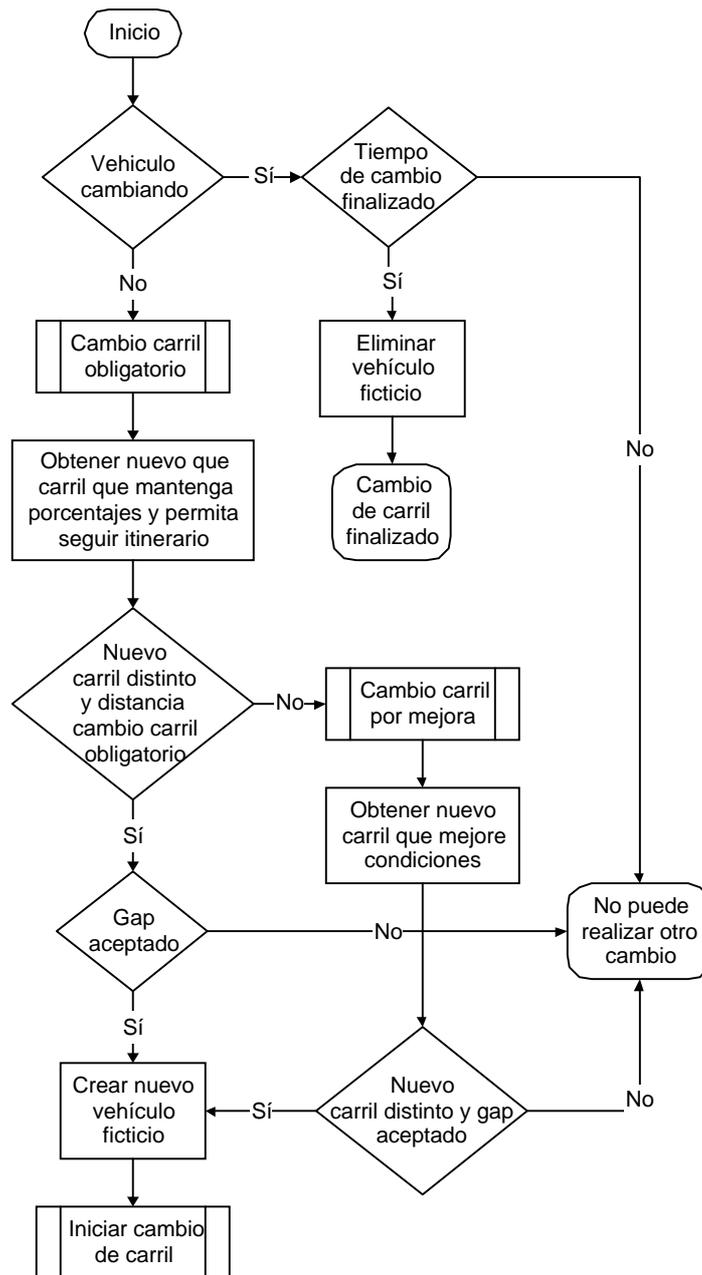


Figura 2.6. Procedimiento General de Cambio de Carril

2.4.2 Cambio de carril obligatorio

Este tipo de cambio de carril es el que se ven forzados a realizar los vehículos para situarse en el carril que les permite seguir el itinerario que tienen fijado, es decir, para situarse en aquellos carriles que les permiten realizar giros adecuados o seguir de frente al llegar a una intersección según les convenga, también sirven para situar al vehículo en el carril apropiado para realizar un aparcamiento. Es por ello, que estos cambios son más comunes cuanto más próximos al fin del tramo se encuentren los vehículos, ya que será entonces cuando deban decidir cuál es el carril apropiado.

2.4.2.1 Comprobación de distancia

Para que el vehículo que lo desee (vehículo n en la figura 2.1) lleve a cabo este tipo de cambio de carril, inicialmente hay que verificar que se encuentre a cierta distancia del final del tramo por el cual se encuentra circulando en ese instante.

Esta condición se cumplirá cuando el vehículo n se encuentre a una distancia del final del tramo menor que la distancia de cambio de carril obligatorio $D_n^o(t)$ definida mediante la expresión:

$$D_n^o(t) = D_n^{seg}(t) + D_n^{inf}(t) \times \left[1 + \frac{|c_d - c_a|}{n} \right]^\lambda$$

donde $D_n^{seg}(t) = \frac{v_n^2(t)}{-2a_{n,max}^-}$ es la distancia de seguridad del vehículo, $D_n^{inf}(t)$ es la distancia de influencia del vehículo en ese instante, c_d y c_a son los carriles de destino y actual respectivamente, n es el número de carriles del tramo y λ es un parámetro que depende del tipo de conductor, pero que por simplicidad se va a considerar igual a la unidad.

El último factor de la anterior expresión se emplea para considerar el efecto que puede tener el que haya que realizar más de un cambio de carril para alcanzar el carril de destino deseado. De forma que si hay que realizar más de un cambio, se adelante la distancia a la que se permite realizar este tipo de cambios.

En la gráfica de la figura 2.7 se observa el valor que toma la distancia mínima hasta el final del tramo a la que se debe encontrar un vehículo para realizar un cambio de carril obligatorio en función de la velocidad que lleva el vehículo en ese instante. Se muestran los resultados para distintos valores del parámetro λ , aunque ya se ha comentado que sólo será considerado el caso en el que toma el valor unidad.

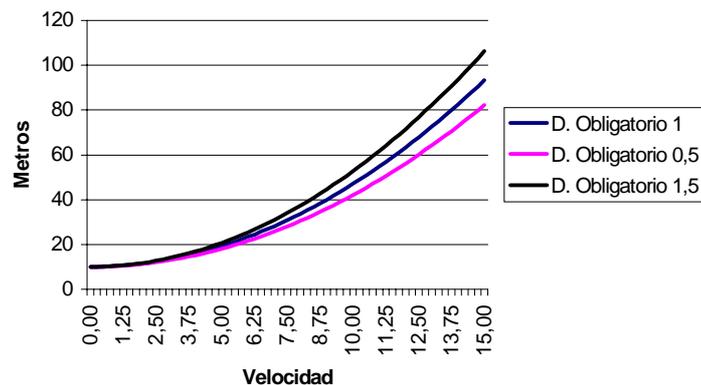


Figura 2.7. Distancia mínima para cambio de carril obligatorio

Por lo tanto, ya se conoce la distancia a partir de la cual un vehículo sólo realiza cambios de carril obligatorios. En el resto del tramo, existe una cierta probabilidad de

que el vehículo pueda realizar un cambio de carril obligatorio y otra de que lo haga por mejora.

La función que especifica la probabilidad de realizar un cambio de carril obligatorio antes de llegar a la posición del tramo en la que éste ha de hacerse forzosamente, viene dada por la siguiente expresión:

$$f(x) = \left(\frac{1 - e^{-x}}{1 - e^{-d}} \right)^\gamma$$

donde x es la posición del vehículo en el tramo, $d = longitud - D_n^o(t)$, *longitud* es la longitud del tramo y γ es un parámetro que depende del tipo de conductor que lleva el vehículo.

La forma de esta función de probabilidad hace que la probabilidad de seleccionar un cambio de carril obligatorio antes que uno por mejora aumente conforme el vehículo se acerca a la distancia mínima de cambio de carril obligatorio. Esto se aprecia en la figura 2.8, en la que se ha realizado un experimento con dos valores distintos del parámetro γ , apreciando que cuanto mayor es el valor de este parámetro, los cambios de carril obligatorios se producen con mayor probabilidad más cerca del final del tramo.

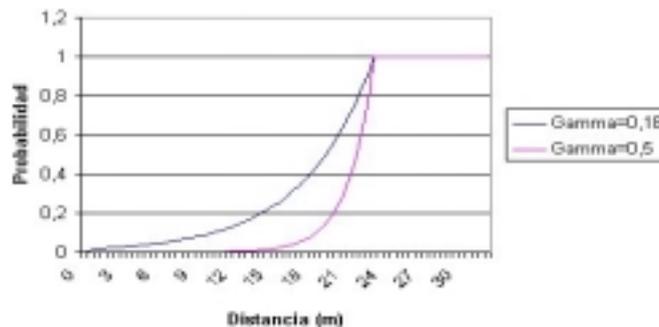


Figura 2.8. Probabilidad de seleccionar un cambio de carril obligatorio

Todo lo comentado en este apartado acerca de la distancia a partir de la cual se puede realizar un cambio de carril obligatorio, está referido a aquellos cambios obligatorios que buscan el estar situados en el carril correcto al finalizar el tramo. Es decir, quedan excluidos los cambios de carril obligatorios para aparcamientos.

Los cambios de carril para situarse en el carril apropiado para realizar una maniobra de aparcamiento, pueden llevarse a cabo siempre que el vehículo se encuentre a una distancia fija del lugar en el que desea aparcar.

2.4.2.2 Búsqueda del carril de destino obligatorio

El siguiente paso que se debe dar a la hora de realizar un cambio de carril obligatorio consiste en buscar el carril destino hacia el que el vehículo quiere desplazarse.

Antes de ver cómo se realiza esta búsqueda, conviene aclarar el concepto de *superávit* de vehículos que circulan por un carril. Este concepto está relacionado con el porcentaje de vehículos que deben circular por cada carril del total de vehículos que circulan por el tramo. En el caso de que en un carril exista un porcentaje de vehículos superior al que debiera, se dice que hay *superávit* de vehículos en el carril, si por el contrario, existe un tanto por ciento inferior, se dice que el carril presenta un déficit de vehículos.

El porcentaje de vehículos adecuado para cada carril se reparte según el giro que se puede alcanzar desde cada uno de los carriles del tramo, y por el tanto por ciento de vehículos que desean realizar cada uno de los giros.

El proceso de búsqueda del carril de destino en un cambio obligatorio puede descomponerse en dos pasos claramente diferenciados:

a) Comprobación de superávit de vehículos.

El primer paso consiste en comprobar si el carril en el que se encuentra actualmente el vehículo presenta un superávit de vehículos, es decir, si hay más de los que debería haber para mantener los porcentajes deseados. En caso de ser así, se busca un carril de ese tramo que tenga déficit, y si lo hay, ése será el carril de destino deseado por el vehículo. Si el carril actual y el deseado estuviesen separados por algún otro carril intermedio, el carril deseado será justamente el adyacente al actual en la dirección donde está el deseado, de esta forma se evita que se produzcan cambios de dos carriles en una única maniobra.

b) Comprobación de itinerario deseado.

El siguiente paso consiste en encontrar un carril que permita al vehículo dirigirse al tramo siguiente deseado. En el caso de que no haya un nodo de destino (es decir, se trate de un tramo final), o bien, el carril deseado obtenido en el primer paso permita al vehículo seguir su itinerario, el carril de destino obligatorio será el obtenido en el paso anterior. En caso contrario, se buscará el carril del tramo actual más próximo a aquel por el que circula el vehículo y que le permita seguir su camino. Si el nuevo carril deseado no es adyacente al carril actual, se procederá de forma análoga a la comentada en el apartado a).

Todo este proceso que se debe seguir para buscar cuál es el carril hacia el que se debe desplazar un vehículo a la hora de realizar un cambio de carril obligatorio, se observa en el diagrama de flujo de la figura 2.9, en el que se aprecian con claridad los dos pasos anteriormente comentados.

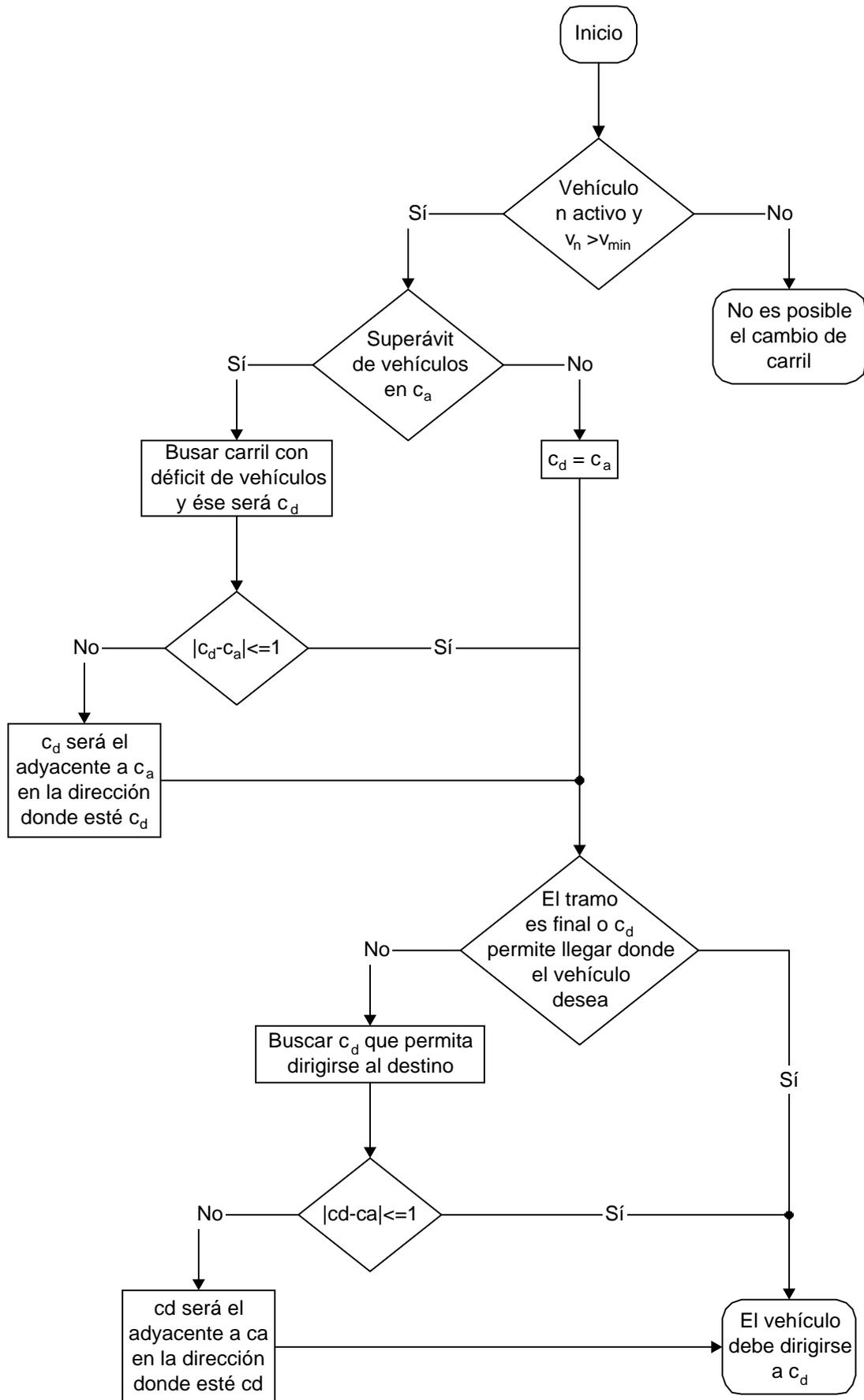


Figura 2.9. Búsqueda de carril para cambio obligatorio

2.4.2.3 Comprobación de la distancia de seguridad de la maniobra (gap)

Cuando se decide acometer la maniobra de cambio de carril, los conductores comprueban si se puede realizar dicho cambio con la suficiente seguridad de que no va a producirse una colisión entre vehículos. Para ello, hay que verificar que existe espacio suficiente en el carril de destino para realizar esta maniobra, a este proceso se le llama comprobación del gap.

Esta comprobación se puede dividir en dos procesos consecutivos, en el primero se trata de encontrar cuáles son las influencias a las que se verá sometido el vehículo en el nuevo carril, así como la influencia que él mismo pasará a ejercer en dicho carril, a continuación se procede a verificar que existe distancia suficiente para realizar el cambio de carril. Veamos dichos procesos en dos apartados por separado, en los que seguiremos la notación indicada en el apartado 2.2.

2.4.2.3.1 Búsqueda de influencias

Para comenzar hay que buscar el futuro vehículo sucesor (vehículo $n-1$ en la figura 2.1) en el nuevo carril (c_d) del vehículo que desea realizar la maniobra de cambio obligatorio (vehículo n). Dicho futuro vehículo sucesor no tiene por qué encontrarse en el mismo tramo, ya que es posible que se encuentre en el movimiento del nodo anterior que conecta con el carril c_d . En el caso de que se determine la existencia del vehículo $n-1$, habrá de tenerse en cuenta la distancia a la que se van a encontrar, para poder verificar posteriormente si se cumple el gap trasero mínimo deseado.

El siguiente paso consiste en determinar el futuro vehículo precedente (vehículo $n+1$) que influirá al vehículo que desea cambiar de carril una vez haya finalizado la maniobra. En el caso de que aquello que se encuentre en el nuevo carril inmediatamente por delante de él sea otro vehículo (y no un obstáculo o un semáforo), ése será el vehículo del que haya que estudiar la distancia a la que se encuentra para determinar si se verifica el gap delantero mínimo necesario. Al igual que antes, el futuro vehículo precedente puede estar situado en un movimiento del nodo siguiente.

Por lo tanto, al finalizar este proceso de búsqueda de influencias, se conoce si el vehículo n va a encontrar vehículos por delante o por detrás en el nuevo carril destino c_d , cuáles son esos vehículos y en qué posición se encuentran.

2.4.2.3.2 Aceptación de la distancia de seguridad en la maniobra

La distancia de seguridad de la maniobra, o gap, se define como la distancia existente entre la parte delantera de un vehículo y la trasera de su predecesor en ese mismo carril. En este apartado se hace referencia a gap delantero como la distancia entre el vehículo que desea cambiar de carril (vehículo n) y el que le precede (vehículo $n+1$), mientras que el gap trasero es el referente a la distancia respecto al vehículo que le sucede en el carril (vehículo $n-1$).

Este proceso depende de si en el nuevo carril existe un vehículo por delante del que desea cambiar de carril, por detrás, por ambos lados o por ninguno. En el caso de no existir ningún vehículo ni por delante ni por detrás, el gap será evidentemente aceptado, ya que no tiene ninguna limitación en cuanto a espacio para realizar el cambio de carril.

Cuando mediante el proceso descrito en el apartado anterior se haya determinado que existe influencia de un vehículo delantero, trasero o ambos, se debe proceder a la aceptación de las distancias correspondientes.

Una primera condición que deben cumplir estas distancias para que sean aceptadas es que sean superiores a la distancia de seguridad de los vehículos, es decir, que los vehículos tengan espacio suficiente para frenar antes de colisionar con su vehículo predecesor en el caso de que éste quede parado repentinamente. Estas condiciones pueden expresarse de manera matemática mediante las siguientes expresiones, que se refieren a las distancias delantera y trasera respectivamente:

$$x_{n+1}(t) - x_n(t) - L_{n+1} \geq D_n^{seg}(t)$$

$$x_n(t) - x_{n-1}(t) - L_n \geq D_{n-1}^{seg}(t)$$

Por otro lado, en el caso de que la distancia entre los vehículos sea superior a la distancia de influencia, la distancia de seguridad de la maniobra es aceptada sin más comprobaciones, ya que la distancia del vehículo con su predecesor es lo suficientemente grande como para no verse influido por él. Esto implica que se cumpla:

$$x_{n+1}(t) - x_n(t) - L_{n+1} \geq D_n^{inf}(t)$$

$$x_n(t) - x_{n-1}(t) - L_n \geq D_{n-1}^{inf}(t)$$

Por lo tanto, cuando la distancia entre los vehículos se encuentra entre los dos límites anteriores es cuando hay que estudiar si dichas distancias son aceptadas o no. Además, en los cambios de carril obligatorios, al aproximarse al punto en el que se debe iniciar la maniobra obligatoria, y por lo tanto, donde el cambio de carril debe estar finalizado, se van aceptando distancias cada vez más pequeñas conforme el vehículo se acerca a dicho punto. Todo esto hace que la función de probabilidad de aceptar un gap en un cambio obligatorio tenga la forma que se observa en las gráficas que se muestran en la figura 2.10:

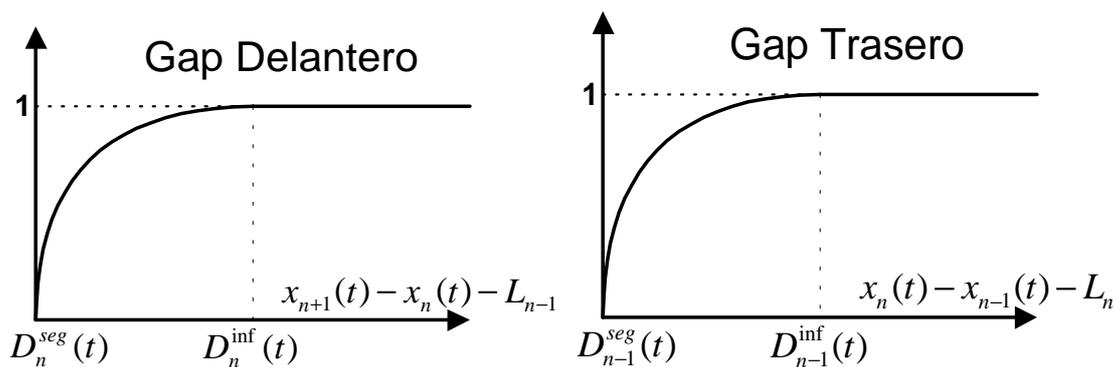


Figura 2.10. Probabilidad de aceptar gap en cambio de carril obligatorio

Las funciones de probabilidad que representan la anterior situación tienen la siguiente expresión matemática para los casos de distancias delantera y trasera respectivamente:

$$f^d(x) = \left(\frac{1 - e^{-z_1}}{1 - e^{-z_2}} \right)^y ; z_1 = D_n^{\text{inf}}(t) - (x_{n+1}(t) - x_n(t) - L_{n+1}) \text{ y } z_2 = D_n^{\text{inf}}(t) - D_n^{\text{seg}}(t)$$

$$f^t(x) = \left(\frac{1 - e^{-z_1}}{1 - e^{-z_2}} \right)^y ; z_1 = D_{n-1}^{\text{inf}}(t) - (x_n(t) - x_{n-1}(t) - L_n) \text{ y } z_2 = D_{n-1}^{\text{inf}}(t) - D_{n-1}^{\text{seg}}(t)$$

El parámetro y que aparece en las anteriores expresiones es el que permite que la función de probabilidad aumente su valor conforme el vehículo se aproxima a la posición en la que debe realizar el cambio de carril. Este parámetro viene dado por las siguientes expresiones para ambos gaps:

$$y = \left(\frac{D_n^{\text{inf}}(t) - (x_{n+1}(t) - x_n(t) - L_{n+1})}{D_n^{\text{inf}}(t) - D_n^{\text{seg}}(t)} \right)^\gamma ; y = \left(\frac{D_{n-1}^{\text{inf}}(t) - (x_n(t) - x_{n-1}(t) - L_n)}{D_{n-1}^{\text{inf}}(t) - D_{n-1}^{\text{seg}}(t)} \right)^\gamma$$

El parámetro γ es el mismo que se utilizó en apartados anteriores y cuyo valor depende del tipo de conductor.

En la gráfica 2.11 se muestra un ejemplo de una función de probabilidad de aceptación de distancia delantera para distintos espacios y distintos valores del parámetro y :

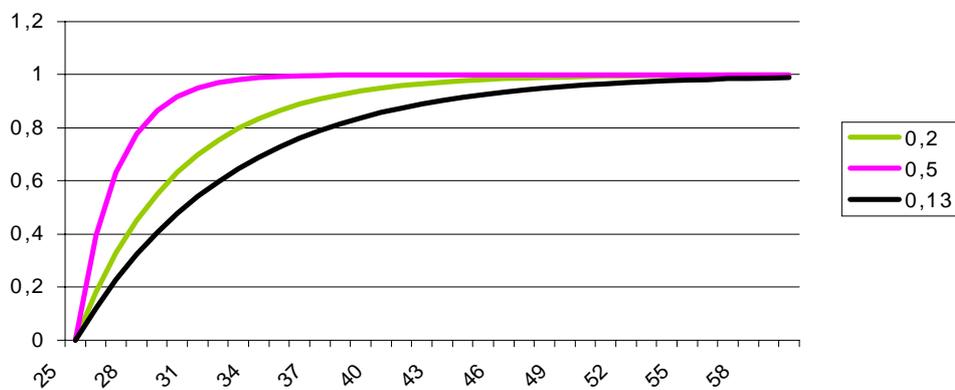


Figura 2.11. Función de aceptación de gap delantero

En la figura 2.12 se muestra el diagrama de flujo del proceso de aceptación de gap en función del tipo de influencia de vehículos que se vaya a encontrar el vehículo n en el carril hacia el que desea desplazarse. En él se incluye la aceptación del gap de cortesía que será objeto de análisis en el siguiente apartado:

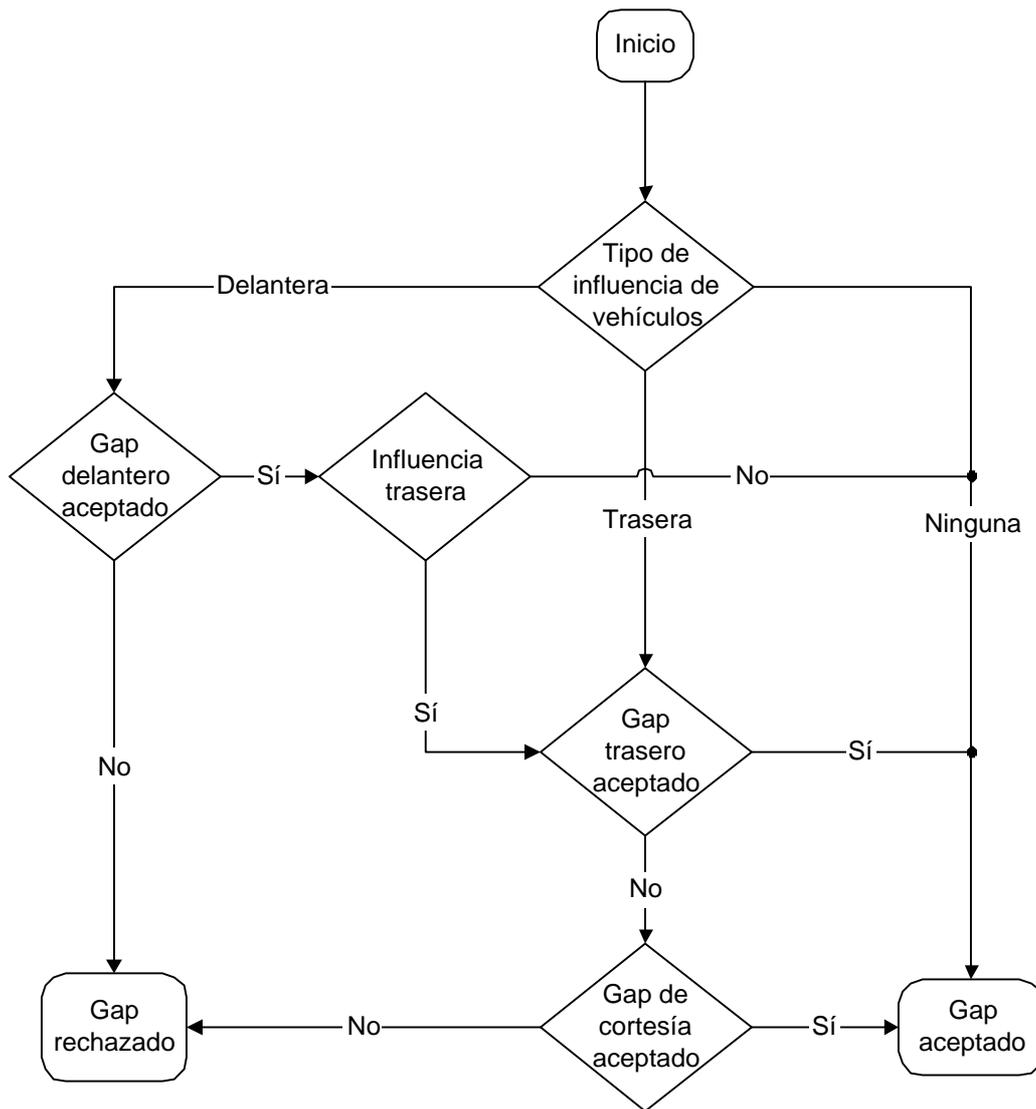


Figura 2.12. Aceptación de distancia de seguridad en cambio de carril

2.4.2.3.3 Probabilidad de cortesía (Modelo de Merging)

La probabilidad de cortesía consiste en una segunda oportunidad para aquellos vehículos cuyo cambio de carril ha sido rechazado por no satisfacer el gap trasero las condiciones impuestas. El cambio de carril por cortesía se produce cuando el futuro vehículo sucesor disminuye la velocidad permitiendo la realización de la maniobra.

En ciertas ocasiones, el vehículo sucesor no tiene que llegar a disminuir su velocidad para permitir este cambio de carril, sino que simplemente se produce una disminución en la aceleración que podría aplicar en el caso de no permitir la maniobra de cambio de carril.

Lógicamente, no se puede permitir este cambio en cualquier situación, sino que es necesario definir un límite menos restrictivo para el gap trasero. El nuevo límite de la distancia de seguridad mínima para permitir el cambio de carril viene determinado en función de la velocidad relativa del vehículo que va a cambiar de carril respecto al que le cede el paso.

Si la velocidad correspondiente al vehículo que va a realizar el cambio es superior a la del posible vehículo cortés ($v_n \geq v_{n-1}$), entonces el gap es inmediatamente aceptado, ya que en el caso de que el vehículo que va por delante frene bruscamente, al vehículo de detrás también le da tiempo de quedar completamente detenido, evitando la colisión con el vehículo precedente.

En la figura 2.13 se muestra una representación gráfica de esta situación, en la que se aprecia que a pesar de existir una distancia pequeña entre los vehículos n y n-1, la maniobra se puede realizar con seguridad, ya que al ser superior la velocidad del vehículo n, en el instante siguiente la distancia entre ambos aumenta. Además, el espacio que necesita el vehículo n-1 para detenerse es inferior que el que necesita el vehículo n, ya que su velocidad también lo es.

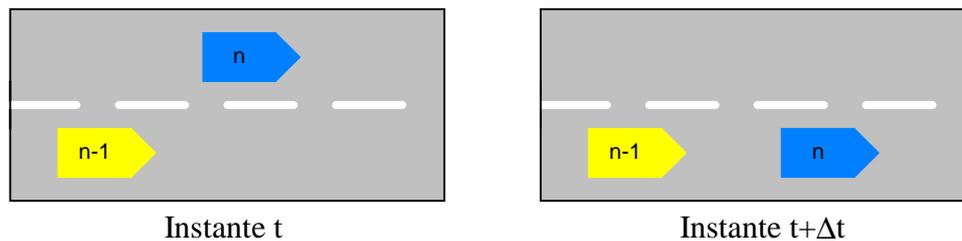


Figura 2.13. Cambio de carril por cortesía cuando $v_n \geq v_{n-1}$

En el caso de que el vehículo que va a dejarle paso al otro circule a mayor velocidad ($v_n < v_{n-1}$), el gap de cortesía se cumple siempre que el vehículo trasero tenga tiempo de frenar hasta la velocidad del que le precede antes de colisionar con él. Una vez que ha alcanzado dicha velocidad, se aplicaría el caso anterior.

La condición que debe cumplir la distancia entre los vehículos se expresa mediante la siguiente expresión:

$$gap > dist_frenado - dist_recorrida$$

donde $dist_frenado$ es la distancia que recorre el vehículo n-1 hasta que su velocidad se iguala a la del vehículo n, mientras que $dist_recorrida$ es la distancia que recorre este último vehículo durante ese mismo tiempo.

Llamando d_{max} a la máxima frenada (en valor absoluto) permitida para el vehículo n-1 en esta maniobra, se puede calcular una expresión matemática para la distancia necesaria para un cambio de carril por cortesía de la siguiente manera:

- El tiempo que necesitaría el vehículo trasero para frenar hasta la velocidad del delantero suponiendo que aplicara la máxima frenada posible, viene dado por:

$$t_f = \frac{v_{n-1}(t) - v_n(t)}{d_{max}}$$

- Durante ese tiempo y si aplicara esa frenada, el vehículo trasero recorrería una distancia:

$$dist_frenado = \frac{v_{n-1}^2(t) - v_n^2(t)}{2 \cdot d_{\max}}$$

- La distancia recorrida por el vehículo n que desea cambiar de carril durante ese tiempo suponiendo que mantuviera la velocidad que llevaba por su carril de origen, sería:

$$dist_recorrida = v_n(t) \cdot \frac{v_{n-1}(t) - v_n(t)}{d_{\max}}$$

- Por lo tanto, la distancia mínima necesaria tiene la siguiente expresión:

$$x_n(t) - x_{n-1}(t) - L_n > \frac{(v_n(t) - v_{n-1}(t))^2}{2 \cdot d_{\max}}$$

En la figura 2.14 se muestran las distancias involucradas en este cálculo, para lo que se han representado las posiciones de los vehículos n y n-1 en los instantes de simulación t y t + t_f.

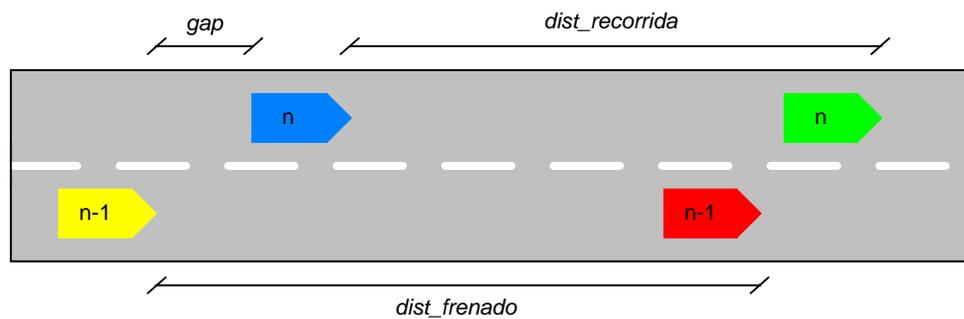


Figura 2.14. Comprobación de distancia en cambio de carril por cortesía

El hecho de utilizar una deceleración distinta de la máxima que puede efectuar un vehículo, tiene el objetivo de evitar que un vehículo cortés frene muy bruscamente para dejar paso al que desea cambiar de carril, ya que esta situación no tiene sentido en la realidad.

En el caso de cumplimiento del gap de cortesía, se procede a verificar si se cumple la probabilidad de cortesía, ya que sólo en algunas de las ocasiones en las que la distancia entre vehículos es suficiente, se lleva a cabo finalmente este cambio de carril por cortesía.

Para verificar esta probabilidad, se realiza el producto de dos factores:

- Por un lado, el número de veces consecutivas que el vehículo que desea cambiar de carril ha intentado realizar ese mismo cambio de carril y sin embargo ha sido rechazado por no cumplir el gap (ya sea el normal o el de cortesía).

- El otro factor es un factor de cortesía dependiente del tipo de conductor del vehículo que debe permitir al otro pasar a su propio carril. Este factor es mayor, por ejemplo, para conductores novatos que para conductores con actitud agresiva.

El producto de estos dos factores determina la probabilidad de que se efectúe el cambio de carril por cortesía. De esta forma, se permiten cambios de carril que en un principio no se podían producir, teniendo mayores probabilidades de realizarlos aquellos vehículos que ya han intentado llevarlo a cabo otras veces y que se encuentran en el nuevo carril a determinado tipo de conductores.

La siguiente gráfica de la figura 2.15 representa la probabilidad de realización de un cambio por cortesía una vez aceptada la distancia correspondiente. Dicha probabilidad depende del número de intentos que ha realizado el vehículo n y del tipo de conductor del vehículo n-1.

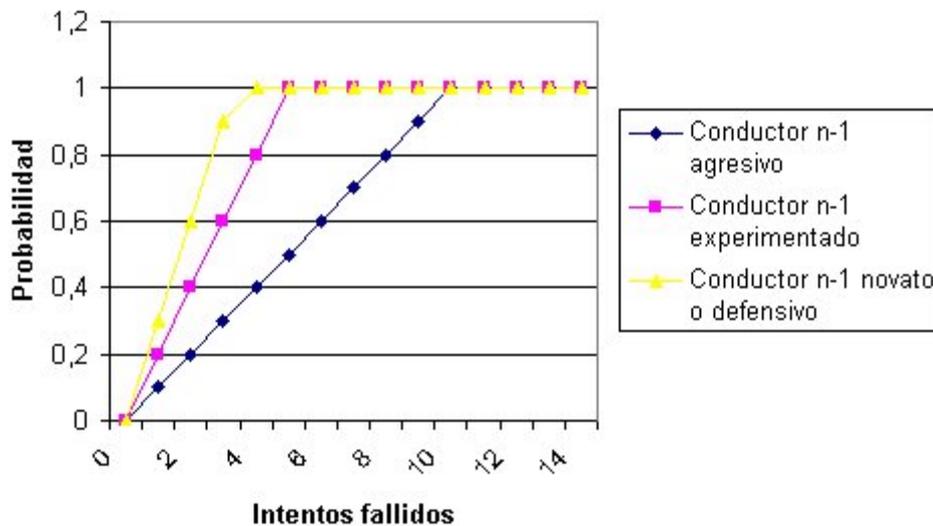


Figura 2.15. Probabilidad de aceptar cambio de cortesía

2.4.3 Cambio de carril por mejora

Este tipo de cambio de carril se produce cuando los vehículos desean mejorar su situación en el viario, es decir, porque observan que circulando por otro carril distinto al actual, podrían aumentar su velocidad. La primera condición para llevar a cabo un cambio de este tipo es que no se cumplan las condiciones vistas en el apartado 2.4.2.1 en las que se determinaba la distancia de cambio de carril obligatorio.

2.4.3.1 Condiciones previas

Para que se produzca un cambio de carril por mejora se deben satisfacer necesariamente una serie de condiciones tales como:

- El vehículo que se encuentra inmediatamente por delante en su mismo carril no debe estar llevando a cabo también otra maniobra de cambio de carril, ya que al existir un vehículo ficticio y otro real, no se tiene una

percepción demasiado exacta de cómo puede mejorar la velocidad si el vehículo se desplaza a otro carril.

- El vehículo debe encontrarse activo en el tramo y no estar realizando ya otro cambio de carril.
- El vehículo no debe tener ningún aparcamiento asignado, o en el caso de tenerlo, entonces debe estar influido por un obstáculo, ya que en ese caso debe cambiar de carril para evitarlo.
- En el caso de no estar influido por un obstáculo, el vehículo debe llevar una velocidad que sea superior a una velocidad mínima predefinida para poder realizar un cambio de carril.

2.4.3.2 Búsqueda de carril destino por mejora

Este proceso consiste en encontrar el carril perteneciente al mismo tramo por el que el vehículo puede circular a una mayor velocidad, o lo que es lo mismo, aquel carril en el que la aceleración del vehículo en el siguiente instante de simulación tenga un mayor valor.

Se comienza por recorrer todos los carriles del tramo, comprobando en cada uno de ellos si se mejoran las condiciones. Aquel carril en el que se mejoren las condiciones será el carril de destino.

La comprobación de si se produciría una mejora en las condiciones al pasar el vehículo de un carril a otro, consiste en calcular cuál sería la aceleración futura del vehículo si circulara por otro carril, viéndose por lo tanto en otras condiciones de influencia diferentes a las que tiene en su carril actual. Cuando la aceleración en el nuevo carril podría mejorar en un tanto por ciento determinado respecto la que tendría ese vehículo de continuar en el mismo carril, se considera que se producen las mejoras de condiciones.

Este tanto por ciento en el que debe aumentar la aceleración futura del vehículo para considerar que se dan las condiciones de mejora, depende del tipo de conductor que lleve el vehículo, considerando, por ejemplo, que un conductor novato necesita mayor margen de mejora que uno experimentado para llevar a cabo un cambio de carril por mejora.

Una vez finalizado el proceso anterior, el carril de destino definitivo será el que se encuentre inmediatamente a la derecha o a la izquierda del carril actual, en función de que el obtenido anteriormente se encuentre hacia un lado o hacia otro. De esta forma, se evita que se produzca un cambio de dos o más carriles simultáneamente, ya que es algo indeseado en nuestro modelo.

2.4.3.3 Comprobación de la distancia de seguridad de la maniobra (gap)

El proceso de comprobación de esta distancia para un cambio de carril por mejora es totalmente análogo al visto en el apartado 2.4.2.3 para un cambio obligatorio, tanto en la búsqueda de influencia, en la aceptación del gap y en la probabilidad de cortesía (o modelo de Merging).

La diferencia respecto al cambio obligatorio consiste en que los conductores buscan espacios mayores para realizar este tipo de cambios, es decir, para realizar un cambio por mejora, la distancia entre los vehículos debe tener un valor más cercano a la distancia de influencia que a la de seguridad. De esta forma, los conductores aceptarán distancias superiores con mayor probabilidad, por lo que arriesgan menos a la hora de realizar un cambio de carril por mejora que uno obligatorio. Esta tendencia se aprecia en las siguientes figuras:

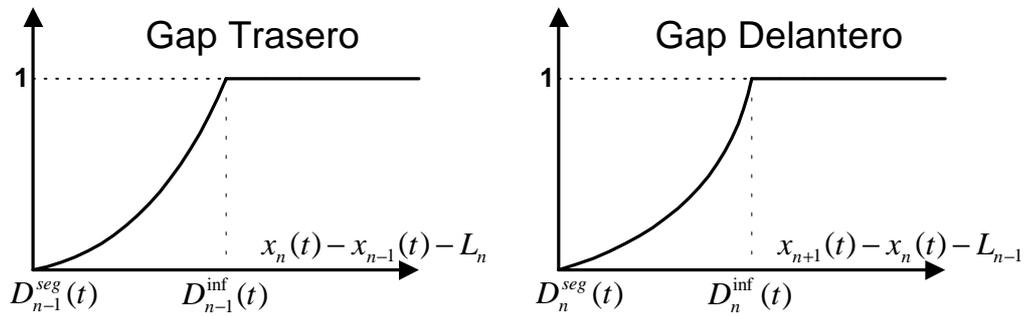


Figura 2.16. Probabilidad de aceptación de gap en cambios por mejora

Las funciones matemáticas que determinan este comportamiento para las distancias delanteras y traseras respectivamente son:

$$f^d(x) = \left(\frac{1 - e^{-z_1}}{1 - e^{-z_2}} \right)^\gamma;$$

$$z_1 = (x_{n+1}(t) - x_n(t) - L_{n+1}) - D_n^{seg}(t), \quad z_2 = D_n^{inf}(t) - (x_{n+1}(t) - x_n(t) - L_{n+1})$$

$$y = \left(\frac{D_n^{inf}(t) - (x_{n+1}(t) - x_n(t) - L_{n+1})}{D_n^{inf}(t) - D_n^{seg}(t)} \right)^\gamma$$

$$f^t(x) = \left(\frac{1 - e^{-z_1}}{1 - e^{-z_2}} \right)^\gamma;$$

$$z_1 = (x_n(t) - x_{n-1}(t) - L_n) - D_{n-1}^{seg}(t), \quad z_2 = D_{n-1}^{inf}(t) - (x_n(t) - x_{n-1}(t) - L_n)$$

$$y = \left(\frac{D_{n-1}^{inf}(t) - (x_n(t) - x_{n-1}(t) - L_n)}{D_{n-1}^{inf}(t) - D_{n-1}^{seg}(t)} \right)^\gamma$$

donde el factor γ es el mismo que se vio con anterioridad.

En la figura 2.17 se muestra esta función para dos tipos distintos de conductores con distancia de seguridad de 25 metros y distancia de influencia de 50 metros, en ella se observa que los conductores más prudentes (mayor γ) aceptan espacios mayores.

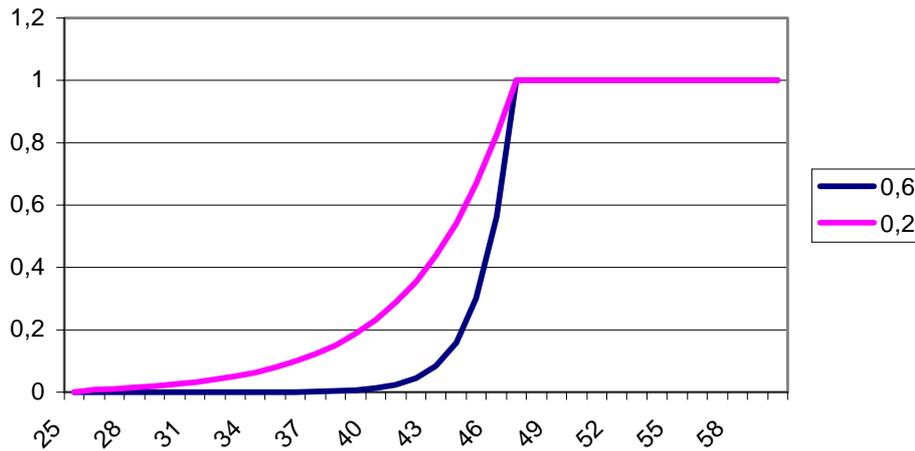


Figura 2.17. Aceptación de gap delantero en cambio por mejora

2.4.4 Inicio de cambio de carril

Este proceso de inicio de cambio de carril se lleva a cabo una vez que se ha comprobado que el gap ha sido aceptado. Este inicio es independiente de que el cambio sea por mejora u obligatorio, por lo que todo lo que se diga en este apartado será aplicable a ambos tipos de cambio.

Los pasos a seguir en este proceso son la creación de un vehículo ficticio en el carril antiguo por el que circulaba el vehículo y la actualización de las influencias de todos los vehículos que se ven afectados por este cambio de carril.

2.4.4.1 Creación de un vehículo ficticio

Estos cambios de carril no se producen de manera instantánea, sino que tienen una duración predefinida que depende del tipo de conductor que se encuentre conduciendo el vehículo.

Para conseguir este efecto de duración temporal del cambio, la idea fundamental es la creación de un vehículo ficticio que va a permanecer en el carril original durante el tiempo que ha sido predefinido, mientras que el vehículo real va a pasar inmediatamente al nuevo carril deseado. De esta forma, el vehículo que está realizando el cambio de carril se desdobra en dos vehículos durante el tiempo que se tarda en finalizar la maniobra de cambio de carril.

Este efecto se aprecia en el gráfico de la figura 2.18, en el que el vehículo que circulaba por el carril 1 en el instante t , pasa al carril 0 en el instante $t+\Delta t$, apareciendo un vehículo ficticio en el carril 1 en la misma posición en la que se encuentra el vehículo real.

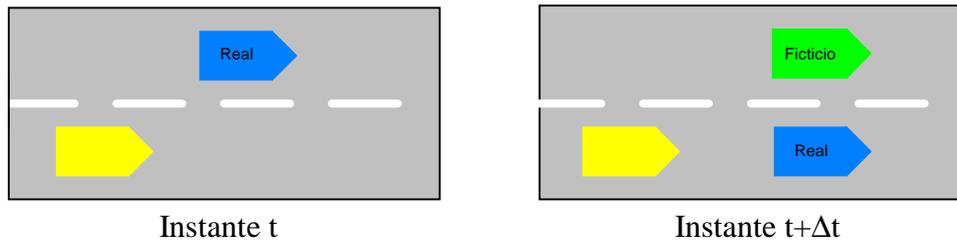


Figura 2.18. Creación de un vehículo ficticio al inicio de un cambio de carril

Estos dos vehículos (que en realidad son el mismo) van a circular prácticamente en paralelo por dos carriles adyacentes durante el tiempo que dura el proceso de cambio de carril. Aunque es posible que alguno vaya algo más rápido que el otro debido a las distintas influencias a las que se ven sometidos.

2.4.4.2 Actualización de influencias

Este proceso de inicio de cambio de carril continúa con la actualización de las influencias a las que se ven sometidos tanto los vehículos (real y ficticio) que están llevando a cabo el cambio de carril, como aquellos que "influyen a" o "se ven influidos por" ellos.

Esto se debe a que los vehículos que circulan por el carril origen, ahora se verán afectados por la presencia del nuevo vehículo ficticio que se ha introducido en dicho carril, lo mismo ocurre con los vehículos que circulan por el nuevo carril, que ahora se verán afectados por el vehículo real que ha cambiado de carril.

2.4.5 Finalización de un cambio de carril

Este proceso de finalización de un cambio de carril es exactamente el mismo para cualquier cambio que se produzca, independientemente del tipo de cambio del que se trate.

Se inicia cuando nos encontramos con un vehículo que está cambiando de carril y cuyo tiempo predefinido para realizar este cambio ya se ha cumplido. Cabe recordar que este tiempo que tiene un vehículo para llevar a cabo la maniobra de cambio de carril depende el tipo de conductor.

En función de que el coche que estemos analizando sea el vehículo real que se encuentra circulando por el nuevo carril, o el ficticio que se mantiene por el carril antiguo, este proceso es diferente. Veamos a continuación cómo hay que proceder en función del vehículo del que se trate.

2.4.5.1 Vehículo real

En el caso de tratarse del vehículo real que circula por el nuevo carril, la única acción que hay que realizar es el cambio en el estado de cambio del vehículo, ya que hay que indicar que el vehículo pasa a estar sin cambiar de carril.

De esta forma, el vehículo puede proceder a iniciar un nuevo cambio de carril en el momento en que lo vea necesario.

2.4.5.2 Vehículo ficticio

Este vehículo ficticio ya no tiene sentido una vez que ha concluido el tiempo asignado al cambio de carril, por lo que debe ser eliminado, no sin antes realizar algunas acciones previas.

Se debe comenzar por buscar el vehículo que estaba siendo influido por el vehículo ficticio, ya que cuando éste desaparezca del viario, va a cambiar de influencia de su perseguidor, pasando a estar influido por aquello que influía hasta ahora al propio vehículo ficticio.

3 Función de generación de vehículos

3.1 Introducción

La función de generación determina la manera en la que los vehículos se incorporan al viario. Los vehículos se incorporan a la red según una función trapezoidal, es decir, la evolución temporal del número de vehículos introducidos en cada tramo tendrá forma de trapecoide.

El trapecoide representa el número de vehículos que se incorporan durante un intervalo de tiempo, de esta forma, en cada instante habría que calcular el área de una sección del trapecoide correspondiente y ése sería el número de vehículos a generar. Sin embargo, se introduce una pequeña variación, consistente en que ese área es el valor medio de cierta distribución estadística que obtiene el número de nuevos vehículos. De esta forma, se introduce un cierto grado de aleatoriedad en la función de generación, lo que la hace más similar a la realidad.

Por lo tanto, todos los tramos en los que se desee que aparezcan nuevos vehículos, deben tener asociada una determinada función de generación de vehículos que quedan definidas por una distribución estadística, así como por algún o algunos trapecoides.

3.2 Función de generación trapezoidal

Este apartado tiene como objetivo la descripción de la función de generación trapezoidal, así como de los procedimientos destinados al cálculo de las áreas que componen el trapecoide.

Considérese un pulso donde los vehículos son generados siguiendo una función trapezoidal de la cual conocemos los puntos que la describe (P_1, P_2, P_3 y P_4). El área de la función de generación indica el número de vehículos que se generan en el pulso.

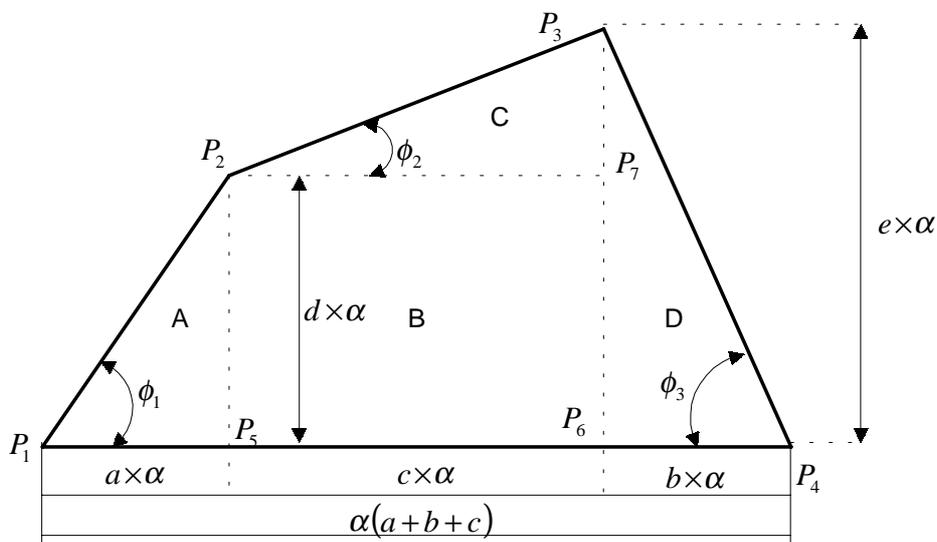


Figura 3.1. Función de generación trapezoidal

Toda figura trapezoidal está caracterizada por 4 zonas que llamamos A, B, C y D que representan subdivisiones del área total del trapezoide. Cada zona está delimitada por una serie de puntos que nos permiten conocer la altura y la base, y por lo tanto el área, de cada una de las zonas.

El área de la figura viene determinada por la expresión:

$$Area = A + B + C + D$$

donde A, B, C y D son las áreas respectivas de los triángulos y rectángulos formados de la intersección entre las líneas de proyección de los puntos de la figura sobre las aristas del polígono.

Donde el área de la figura representa el número de vehículos que se desean generar y α es un parámetro utilizado para parametrizar la función de generación, el cual permite determinar el número de vehículos generados en el pulso.

Aplicando las fórmulas para el cálculo de las áreas de cada zona diferenciada, la expresión quedaría:

Para la zona A, representada mediante un triángulo, el área queda definido por:

$$A = \frac{(a \times \alpha) \times (\alpha \times d)}{2};$$

Para la zona B que delimita un rectángulo, la expresión que representa el área es:

$$B = (c \times \alpha) \times (\alpha \times d);$$

El área del triángulo delimitado por la zona C es:

$$C = \frac{(c \times \alpha) \times (\alpha \times (e - d))}{2}$$

Por último el área del triángulo caracterizado por la zona D es:

$$D = \frac{(b \times \alpha) \times (\alpha \times e)}{2}$$

El área total del trapezoide viene determinado por la expresión:

$$Area = \alpha^2 \left(\frac{a \times d}{2} + c \times d + \frac{c \times (e - d)}{2} + \frac{b - e}{2} \right)$$

En el modelo aquí desarrollado, se va a emplear una función de generación simple, es decir, se va a utilizar una única función trapezoidal que podrá repetirse en el tiempo tantas veces como fuera necesario.

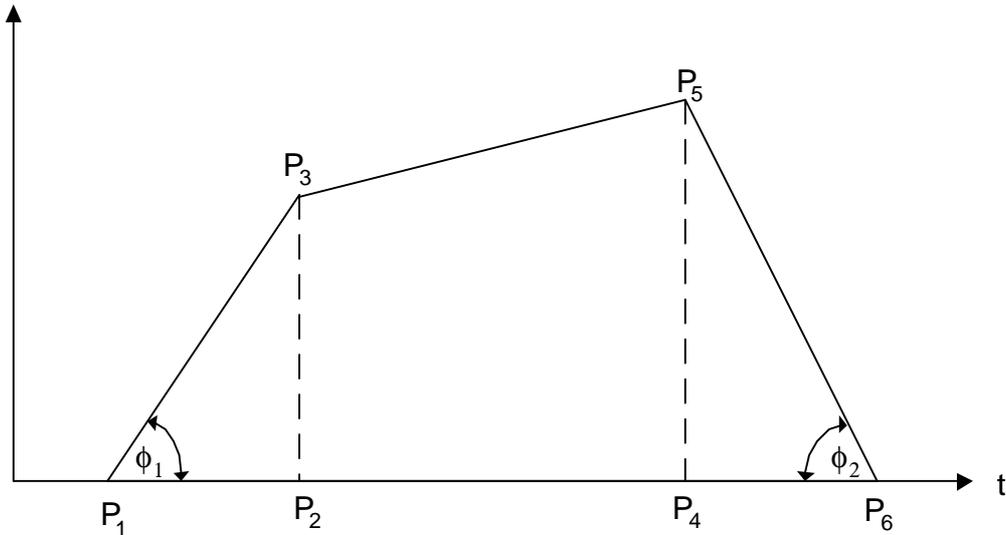


Figura 3.2. Función trapezoidal simple

Como ya se ha comentado en la introducción, esta función trapezoidal no nos dará exactamente la cantidad de vehículos que se deben generar en cada instante, sino la media de una distribución estadística que dará ese número de vehículos.

3.3 Distribuciones estadísticas

Las funciones trapezoidales están caracterizadas por diferentes funciones de distribución estadísticas, que determinan el número de vehículos que deben ser generados en cada instante. Los siguientes apartados describen los procedimientos de generación de los cuatro tipos de distribuciones pseudoaleatorias que se emplean.

3.3.1 Función de distribución normal

Esta función de distribución suele darse con bastante frecuencia en los procesos aleatorios, ya que en virtud del Teorema Central del Límite una serie de procesos aleatorios independientes cuyos efectos son acumulativos, dan como resultado una distribución de este tipo.

La función de densidad de probabilidad de esta distribución se puede expresar de la siguiente manera:

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}; \quad -\infty < x < \infty$$

La esperanza matemática y la varianza de la distribución son respectivamente:

$$E(x) = \mu \quad \text{Var}(x) = \sigma^2$$

En el caso de una normal tipificada (media 0 y varianza 1), esta función de distribución tiene la forma mostrada en la figura 3.3.

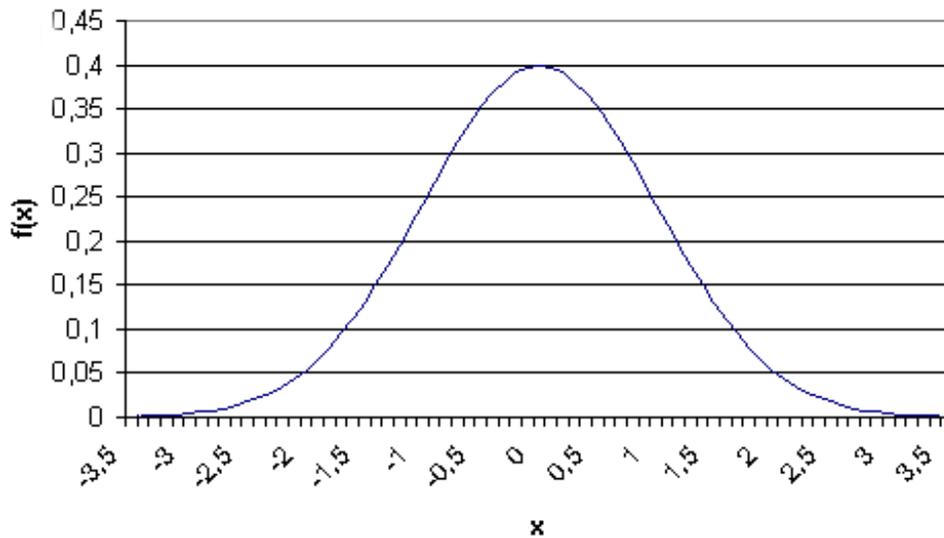


Figura 3.3. Distribución normal

3.3.1.1 Método Box Müller

Éste es un método bastante exacto que trata de generar números aleatorios que siguen una determinada función de distribución normal. Para ello, genera en primer lugar un número según la normal de media 0 y varianza 1, para posteriormente destipificarla a la normal que deseemos de media μ y desviación típica σ .

El algoritmo propuesto por Box y Müller se basa en los siguientes pasos:

- 1.- Se generan dos números aleatorios según una distribución uniforme entre 0 y 1 a los que se llama u_1 y u_2 .
- 2.- El número aleatorio según la $N(0,1)$ se obtiene de la aplicación de:

$$x = \sqrt{-2 \ln(u_1)} \cos(2\pi u_2)$$

- 3.- Para destipificarlo, y por lo tanto obtener el número perteneciente a la distribución normal que se desee, se realiza:

$$y = \mu + \sigma x$$

3.3.1.2 Método Marsaglia Bray

Este método desarrollado por Marsaglia y Bray tiene menor exactitud, aunque cuenta con la ventaja de ser más eficiente, debido a la menor carga computacional que requiere. El algoritmo consiste en los siguientes puntos:

- 1.- Se generan dos números aleatorios según una distribución uniforme entre 0 y 1 a los que se llaman u_1 y u_2 .
- 2.- Se calculan los números: $v_1 = 2u_1 - 1$ y $v_2 = 2u_2 - 1$.
- 3.- Se comprueba si se cumple la condición $s = v_1^2 + v_2^2 < 1$, en caso de no ser así se vuelve al primer punto para calcular dos nuevos números aleatorios.

4.- El número que sigue la normal tipificada es $x = cv_2$, donde el parámetro c viene dado por $c = \sqrt{-2\ln(s)/s}$.

5.- Para destipificarlo, se procede igual que antes, haciendo $y = \mu + \sigma x$.

3.3.2 Función de distribución exponencial

Las variables que siguen una distribución exponencial son también bastante utilizadas, ya que pueden servir, por ejemplo, para modelar el intervalo de tiempo aleatorio que transcurre entre la llegada de dos elementos a un sistema, lo cual se asemeja bastante a lo que se pretende en esta sección.

Su función densidad de probabilidad se expresa mediante:

$$f(x) = \begin{cases} \frac{1}{\theta} e^{-x/\theta} & \Rightarrow x \geq 0 \\ 0 & \Rightarrow x < 0 \end{cases}$$

La media y la varianza son en este caso:

$$E(x) = \theta \quad \text{Var}(x) = \theta^2$$

En la gráfica de la figura 3.4 se muestra la forma de esta función de distribución para un valor del parámetro $\theta = 1$, es decir, la media de la distribución tiene valor 1, lo que implica que el área bajo la curva que queda a la izquierda del 1, es igual que la que queda a su derecha.

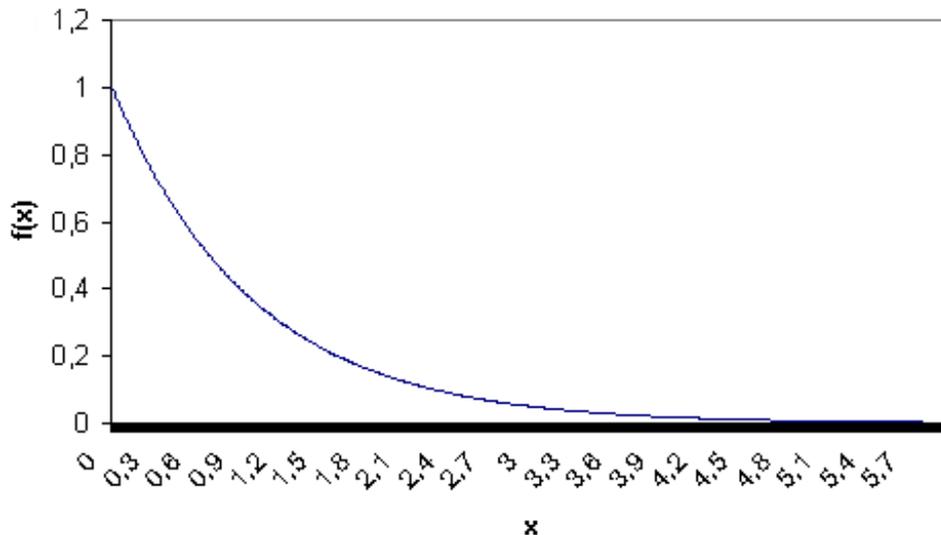


Figura 3.4. Distribución exponencial

3.3.2.1 Método inverso

Este método trata de generar números pseudoaleatorios que sigan una función de distribución exponencial de parámetro θ . Como su propio nombre indica, se trata de realizar la operación inversa a la del cálculo de la función de distribución. El algoritmo se puede expresar en los siguientes pasos:

- 1.- En primer lugar se calcula un número aleatorio u que responda a una distribución uniforme entre 0 y 1.
- 2.- El número aleatorio que sigue la distribución exponencial de parámetro θ vendrá dado por la expresión: $x = -\theta \ln(u)$.

3.3.3 Función de distribución triangular

Esta función de distribución no se corresponde con ningún proceso físico real, pero no por ello deja de ser interesante, ya que es muy utilizada para aproximaciones de modelos de comunicaciones entre otros.

Viene definida por tres parámetros a , b y c , que indican respectivamente los valores de x para los que la distribución comienza a ser distinta de cero, alcanza su máximo (moda de la distribución) y vuelve a ser cero. Por lo tanto, la función de distribución puede expresarse en cada uno de los intervalos por las siguientes expresiones:

$$f(x) = \begin{cases} 0 & \Rightarrow x < a \\ \frac{2(x-a)}{(b-a)(c-a)} & \Rightarrow a \leq x \leq b \\ \frac{2(c-x)}{(c-b)(c-a)} & \Rightarrow b \leq x \leq c \\ 0 & \Rightarrow x > c \end{cases}$$

La esperanza y la varianza de esta distribución vienen dadas por las siguientes expresiones:

$$E(x) = \frac{(a+b+c)}{3} \qquad \text{Var}(x) = \frac{(a^2 + b^2 + c^2 - ab - ac - bc)}{18}$$

En la figura 3.5 se muestra la forma de esta función de distribución para unos parámetros genéricos a , b y c . Como se aprecia en la figura, la función sólo toma valores distintos de cero para valores de x comprendidos entre a y c , alcanzando su valor máximo en el punto b (*moda* de la distribución).

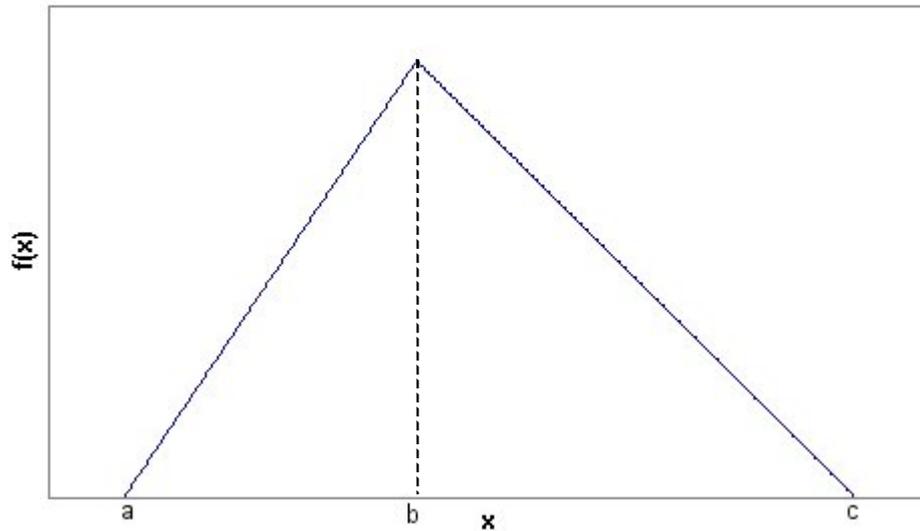


Figura 3.5. Distribución triangular

3.3.3.1 Método de transformación

Para la obtención de números aleatorios que sigan una distribución triangular se utiliza un método de transformación de variables que se basa en obtener los números pseudoaleatorios a partir de la función de distribución. El algoritmo que resulta de aplicar este método se expone a continuación:

- 1.- Calcular $d = \frac{c-a}{b-a}$ y u que será un número aleatorio correspondiente a una distribución uniforme entre 0 y 1.
- 2.- Si $u \leq d \Rightarrow num = \sqrt{d \cdot u}$
- 3.- Si $u > d \Rightarrow num = 1 - \sqrt{(1-d) \cdot (1-u)}$
- 4.- El número deseado es el que resulta de realizar la siguiente operación:
 $x = a + (b-a) \cdot num$

3.3.4 Función de distribución de Poisson

A diferencia de las distribuciones anteriores, la de Poisson se trata de una función discreta, es decir, sólo tienen sentido valores enteros de la variable independiente. Su función de densidad de probabilidad tiene la siguiente expresión:

$$f(x) = \frac{\lambda^x e^{-\lambda}}{x!} \quad x = 0, 1, 2, \dots$$

La esperanza y la varianza de la distribución tienen el siguiente valor:

$$E(x) = \lambda \quad Var(x) = \lambda$$

Para ver la forma de esta distribución, se representa la misma en la siguiente figura para una distribución de Poisson de media 5, en la que en el eje vertical se representa la probabilidad de que se generen los valores indicados en el eje horizontal.

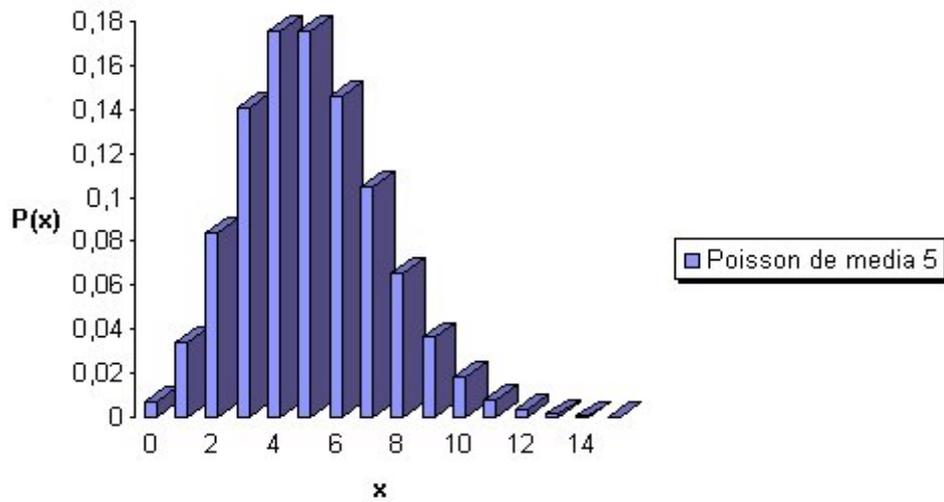


Figura 3.6. Distribución de Poisson de media 5

3.3.4.1 Método de generación

La forma de generar números pseudoaleatorios según una distribución de Poisson es mediante un método iterativo. Para explicar este método lo más conveniente es observar su diagrama de flujo.

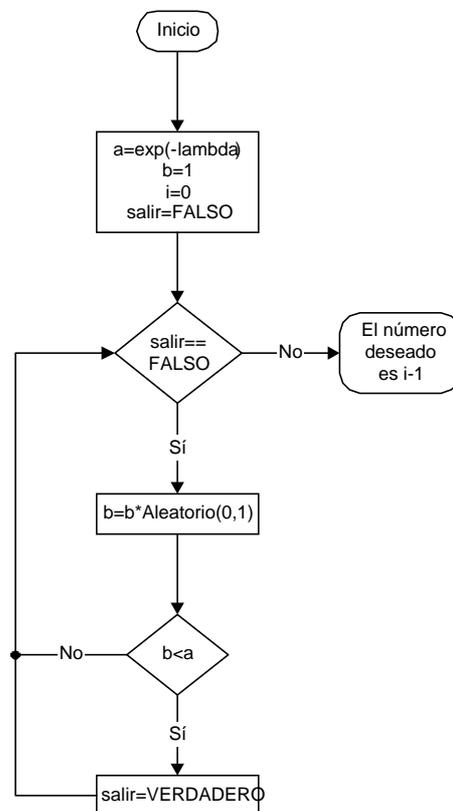


Figura 3.7. Generación de números según distribución de Poisson

3.3.5 Test de bondad de la chi cuadrado de Pearson

Mediante la realización de este test se pretende comprobar cómo de buenos son los números pseudoaleatorios generados mediante los métodos anteriores, es decir, cuál es el grado de confianza con el que los números anteriores se ajustan a la función de distribución deseada.

Para la realización de este contraste se comienza por obtener una muestra de números aleatorios de tamaño n generados mediante los métodos anteriores. A continuación se realiza una partición en k intervalos del espacio en el que toma sus valores la función de distribución, y se calcula la probabilidad ($p_i, i=1, \dots, k$) de que dichos valores caigan en cada uno de los intervalos. Por otro lado, se cuentan el número de elementos ($n_i, i=1, \dots, k$) de la muestra que caen en cada uno de los intervalos anteriores.

Con todos estos datos ya se está en condiciones de calcular el estadístico de Pearson, que responde a la siguiente expresión:

$$\Lambda = \sum_{i=1}^k \frac{(n_i - np_i)^2}{np_i}$$

este estadístico tiene una distribución asintótica de χ_{k-1}^2 bastante aproximada siempre que $np_i \geq 10$ para todo $i=1, \dots, k$.

De esta forma, se puede establecer un contraste de nivel de significación α para la hipótesis de que la muestra procede de la población. Para ello, se determina un umbral crítico π_0 mediante la condición

$$P(\chi_{k-1}^2 \geq \pi_0) = \alpha$$

si $\Lambda < \pi_0$ la hipótesis será aceptada, y en caso contrario será rechazada.

En los siguientes apartados se realiza este test de bondad para cada una de las funciones de distribución estudiadas con anterioridad.

3.3.5.1 Contraste de la distribución normal

Para la realización de este contraste, se generó una muestra de 500 números aleatorios de una población normal de media 5 y varianza 1 mediante el método Box-Müller. En la siguiente tabla se muestran los datos necesarios de cada uno de los 9 intervalos utilizados para la realización del contraste.

Intervalo	n_i	p_i	np_i	$(n_i - np_i)^2 / np_i$
<4.05	83	0.1711	85.55	0.076
4.05 - 4.45	66	0.1201	60.05	0.5896
4.45 - 4.75	64	0.1101	55.05	1.4551
4.75 - 4.95	36	0.0788	39.4	0.2934
4.95 - 5.05	16	0.0398	19.9	0.7643
5.05 - 5.25	30	0.0788	39.4	2.2426
5.25 - 5.55	56	0.1101	55.05	0.0164
5.55 - 5.95	52	0.1201	60.05	1.0791
>5.95	97	0.1711	85.55	1.5325

Tabla 3.1. Contraste de la distribución normal

Sumando los valores de la última columna, se obtiene $\Lambda=8.049$. Tomando un nivel de significación del 5% ($\alpha=0.05$), el umbral crítico viene dado por $P(\chi_8^2 \geq \pi_0) = 0.05$, que consultando en las tablas resulta $\pi_0 = 15.51$. Por consiguiente, se acepta la hipótesis de que la muestra procede de una distribución normal.

3.3.5.2 Contraste de la distribución exponencial

Para este caso se generaron 500 números procedentes de una población exponencial de media 5. Los resultados obtenidos para cada intervalo se resumen en la siguiente tabla:

Intervalo	n_i	p_i	np_i	$(n_i - np_i)^2 / np_i$
0 - 1	96	0.1813	90.63	0.3182
1 - 2	81	0.1484	74.21	0.6213
2 - 3	66	0.1215	60.75	0.4537
3 - 4	47	0.0995	49.74	0.1509
4 - 5	41	0.0814	40.72	0.0019
5 - 6	28	0.0667	33.34	0.8553
6 - 7	30	0.0546	27.3	0.267
7 - 8	14	0.0447	22.35	3.1196
8 - 9	17	0.0366	18.3	0.0923
>9	80	0.1653	82.65	0.085

Tabla 3.2. Contraste de la distribución exponencial

El valor del estadístico de Pearson para esta muestra es de $\Lambda=5.9652$. Tomando también un nivel de significación del 5%, se obtiene un umbral crítico de valor $\pi_0 = 16.92$ procedente de hacer $P(\chi_9^2 \geq \pi_0) = 0.05$, que es superior a Λ , por lo que la hipótesis es aceptada.

3.3.5.3 Contraste de la distribución triangular

En la realización de este contraste se empleó una población triangular de parámetros $a=4.5$, $b=5$ y $c=5.5$. Los resultados obtenidos fueron los siguientes:

Intervalo	n_i	p_i	np_i	$(n_i - np_i)^2 / np_i$
4.5 - 4.65	20	0.045	22.5	0.2778
4.65 - 4.78	52	0.1118	55.9	0.2721
4.78 - 4.88	67	0.132	66	0.0152
4.88 - 4.97	82	0.153	76.5	0.3954
4.97 - 5.03	44	0.1164	58.2	3.4646
5.03 - 5.12	83	0.153	76.5	0.5523
5.12 - 5.22	71	0.132	66	0.3788
5.22 - 5.35	55	0.1118	55.9	0.0145
5.35 - 5.5	26	0.045	22.5	0.5444

Tabla 3.3. Contraste de la distribución triangular

La suma de los valores pertenecientes a la última columna proporciona el valor del estadístico de Pearson que es $\Lambda=5.9151$. En este caso, dado que hay 9 intervalos, y con el mismo nivel de significación del 5%, el valor del umbral crítico es $\pi_0 = 15.51$. Como $\Lambda < \pi_0$, la hipótesis de que la muestra proviene de una distribución triangular es aceptada.

3.3.5.4 Contraste de la distribución de Poisson

La tabla que se obtiene para las 500 muestras obtenidas de una población de Poisson de parámetro $\lambda=5$ es la que se muestra en la tabla 3.4, en la que los intervalos quedan reducidos a valores concretos, ya que es una distribución discreta.

Intervalo	n_i	p_i	np_i	$(n_i - np_i)^2 / np_i$
0,1	14	0.0404	20.2	1.903
2	28	0.0842	42.1	4.7223
3	72	0.1404	70.2	0.0462
4	106	0.1755	87.75	3.7956
5	77	0.1755	87.75	1.317
6	79	0.1462	73.1	0.4762
7,8	86	0.1697	84.85	0.0156
>9	38	0.068	34	0.4706

Tabla 3.4. Contraste de la distribución de Poisson

Para esta tabla, el estadístico de Pearson correspondiente toma el valor $\Lambda=12.7465$. En este caso, el umbral crítico viene dado por la condición $P(\chi_7^2 \geq \pi_0) = 0.05$, obteniéndose $\pi_0 = 14.07$, por lo que se acepta la hipótesis de que la muestra proviene de una distribución de Poisson.

3.4 Procedimiento de generación de vehículos

Mediante este procedimiento se generan todos los vehículos que circulan por el viario. Estos vehículos se generan en aquellos tramos que tengan una determinada función de generación, definida por un trapecoide y una distribución estadística.

Este procedimiento de generación puede descomponerse en una serie de procesos que son estudiados por separado en los siguientes apartados.

3.4.1 Cálculo del área de la sección del trapecoide

El primer paso consiste en discretizar el tiempo en pequeños intervalos, con lo que el trapecoide queda descompuesto en pequeñas secciones cuya base es igual al intervalo de tiempo. De tal forma que en función del instante de tiempo en el que se encuentre la simulación, variará la manera de calcular el área de la sección de trapecoide correspondiente.

La función de generación tiene una forma trapezoidal tal y como la que se muestra en la siguiente figura, en el caso de que los puntos P_3 y P_5 se encuentren a la misma altura, se tiene una forma de trapecio, que no deja de ser más que un caso particular de la figura 3.8. En la misma figura se señalan algunos de los parámetros característicos de la función de generación, como son $d1$, $d2$, $h1$ y $h2$. Además, se considera que la longitud total del trapecoide, incluyendo $d1$ y $d2$, es d .

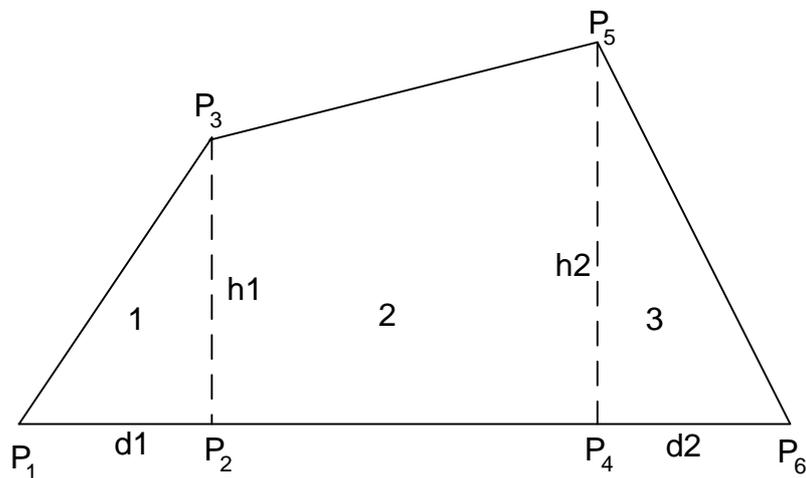


Figura 3.8. Forma trapezoidal de la función de generación

En la anterior figura se observan tres zonas claramente delimitadas, a continuación se expone el modo de calcular el área de la sección de trapecoide y la varianza correspondientes, en función de la zona en que se encuentre el instante de simulación:

- Zona 1. Esta zona se corresponde con una recta creciente, lo que provoca que el número de vehículos que se deben generar va aumentando conforme transcurre el tiempo. Se distinguen 2 casos:
 - Caso A. El intervalo de tiempo Δt empleado para conocer el número de vehículos no se encuentra completamente en el interior del trapecoide por tanto el número de vehículos que deben generarse coincide con el área del triángulo delimitado por la parte de tiempo en el interior del trapecoide.

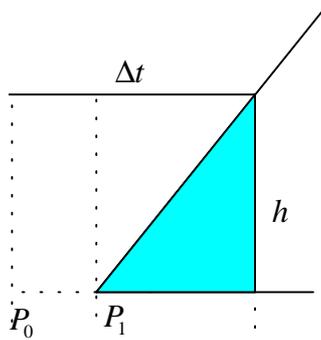


Figura 3.9. Cálculo de área: zona 1, caso A

El número de vehículos, n_v , generados en el intervalo de tiempo Δt viene dado por:

$$n_v = \frac{(\Delta t - (P_1 - P_0)) \times h}{2}$$

donde, P_0 es el instante de tiempo donde comienza la generación, P_1 es el instante de comienzo de la función trapezoidal, h es la altura del triángulo delimitado por el trapezoide y final del intervalo de tiempo y n_v es el número de vehículos o el área.

En este caso, la varianza tiene la misma expresión que el área, por lo tanto $\text{var} = n_v$.

- Caso B. El intervalo de tiempo Δt empleado para conocer el número de vehículos se encuentra completamente en el interior del trapezoide, por tanto, el número de vehículos que deben generarse coincide con el área del triángulo (1) y el rectángulo (2) delimitados en el interior del trapezoide.

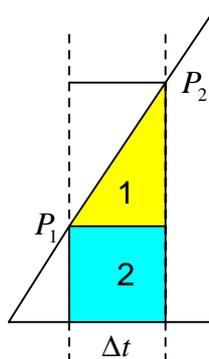


Figura 3.10. Cálculo de área: zona 1, caso B

El número de vehículos, n_v , generados en este caso en el intervalo de tiempo Δt es:

$$n_v = Area_1 + Area_2 = \frac{\Delta t \times (P_{2,y} - P_{1,y})}{2} + \Delta t \times P_{1,y} = \frac{\Delta t \times (P_{1,y} + P_{2,y})}{2}$$

donde, P_1 es el punto de corte entre el instante de comienzo de la generación con el trapezoide, P_2 es el punto de corte del final del intervalo de tiempo con el trapezoide y n_v es el número de vehículos o el área.

La expresión de la varianza viene dada por:

$$var = \frac{\Delta t \times (P_{2,y} - P_{1,y})}{2}$$

- Intervalo comprendido entre zonas 1 y 2. En este caso, el intervalo de tiempo abarca partes de las zonas 1 y 2, como se puede apreciar en la figura siguiente. Por lo que el área que se debe calcular es la suma de la de los rectángulos (1) y (3) más la de los triángulos (2) y (4).

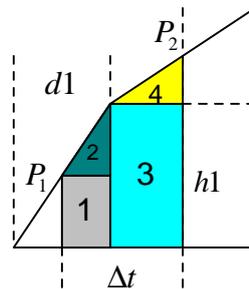


Figura 3.11. Cálculo de área: zonas 1 y 2

Por lo tanto, el área viene determinada por la expresión:

$$n_v = Area_1 + Area_2 + Area_3 + Area_4 = \frac{(d1 - P_{1,x}) \times (P_{1,y} + h1)}{2} + \frac{(P_{2,x} - d1) \times (h1 + P_{2,y})}{2}$$

donde P_1 y P_2 son los puntos de corte de los instantes de inicio y final (respectivamente) del intervalo de tiempo actual con el trapezoide, mientras que $d1$ y $h1$ son dos de los parámetros característicos del trapezoide.

El cálculo del valor de la varianza en este caso se realiza considerando solamente la parte correspondiente a la zona 2:

$$var = \left| \frac{(P_{2,x} - d1) \times (P_{2,y} - h1)}{2} \right|$$

En el gráfico se ha supuesto que en la zona 2 también aumenta el número de vehículos a generar conforme transcurre el tiempo, pero esto no tiene por qué ser así necesariamente, ya que dicho número puede mantenerse constante (función de generación en forma de trapecio), o incluso ir disminuyendo. Sin embargo, las expresiones del área y la varianza permanecen invariables en cualquiera de los tres casos.

- Zona 2. En esta zona, como ya se ha comentado en el caso anterior, el número de vehículos a generar puede aumentar, disminuir o mantenerse constante a lo largo del tiempo. Las expresiones siguientes son válidas para cualquiera de los casos.

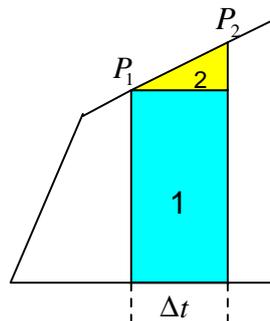


Figura 3.12. Cálculo de área: zona 2

El número de vehículos a generar tiene la siguiente expresión:

$$n_v = Area_1 + Area_2 = \frac{\Delta t \times (P_{2,y} + P_{1,y})}{2}$$

donde el significado de P_1 y P_2 es el mismo que el ya explicado en casos anteriores.

La varianza responde al siguiente valor:

$$var = \left| \frac{\Delta t \times (P_{2,y} - P_{1,y})}{2} \right|$$

- Intervalo comprendido entre zonas 2 y 3. Este caso se representa en la figura 3.13, donde se aprecia que el área viene dada por la suma de los rectángulos (1) y (3) más los triángulos (2) y (4).

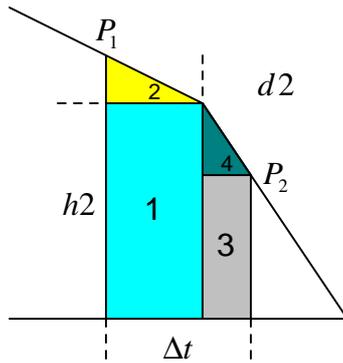


Figura 3.13. Cálculo de área: zonas 2 y 3

El número de vehículos a generar viene dado por:

$$n_v = Area_1 + Area_2 + Area_3 + Area_4 = \frac{(d - d2 - P_{1,x}) \times (P_{1,y} + h2)}{2} + \frac{(P_{2,x} - (d - d2)) \times (h2 + P_{2,y})}{2}$$

Para el cálculo de la varianza se emplea la siguiente ecuación:

$$var = \left| \frac{(P_{2,x} - d - d2) \times (h2 - P_{2,y})}{2} \right|$$

Al igual que en los casos anteriores, estas expresiones son independientes de que en la zona 2 del trapezoido el número de vehículos a generar sea creciente, decreciente o constante en el tiempo.

- Zona 3. Esta zona se caracteriza por generar un número decreciente de vehículos en el transcurso del tiempo. Al igual que ocurría en la zona 1, se pueden dar dos casos diferentes según el intervalo de tiempo esté contenido totalmente en el trapezoido o no.
 - Caso 1. En este primer caso, todo el intervalo de tiempo se encuentra contenido en el trapezoido de la función de generación. Esto se aprecia en el siguiente gráfico.

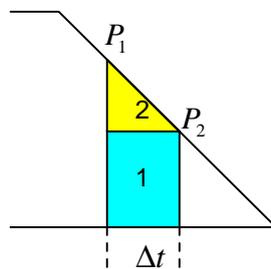


Figura 3.14. Cálculo de áreas: zona 3, caso1

El área viene dada por la adición de las correspondientes al rectángulo (1) y al triángulo (2), respondiendo a la ecuación:

$$n_v = Area_1 + Area_2 = \frac{\Delta t \times (P_{2,y} + P_{1,y})}{2}$$

La varianza viene dada por:

$$var = \left| \frac{\Delta t \times (P_{1,y} - P_{2,y})}{2} \right|$$

- o Caso 2. En este caso el intervalo de tiempo a considerar sobrepasa el instante de tiempo en el que finaliza la función de generación, como se muestra en la siguiente figura.

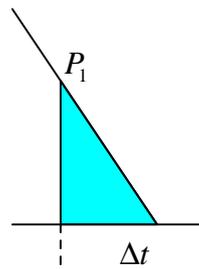


Figura 3.15. Cálculo de áreas: zona 3, caso 2

El área del triángulo que hay que considerar (número de vehículos a generar) es la siguiente:

$$n_v = \frac{(d - P_{1,x}) \times P_{1,y}}{2}$$

La varianza en este caso coincide con el número de vehículos a generar, calculado mediante la expresión anterior.

En el caso de que la función trapezoidal no comience en el instante 0 de simulación, sino que existiera un cierto desfase, todas las expresiones anteriores son válidas sin más que añadir este valor de desfase cuando se consideren los parámetros d_1 y d_2 .

3.4.2 Aplicación de las funciones de distribución

El área calculada en el punto anterior no da directamente el número de vehículos a generar en cada instante, sino que a continuación hay que aplicar el método de generación de números pseudoaleatorios asociado a la función de distribución del tramo correspondiente.

El área de la sección del trapecoide correspondiente servirá de media para el cálculo del número pseudoaleatorio correspondiente. En el caso de la distribución

normal también es necesario conocer la varianza, calculada mediante los métodos expuestos en el apartado anterior.

Este número generado da la cantidad de vehículos que se deben crear en ese instante. Sin embargo, ya que el número generado no tiene por qué ser entero (excepto en el caso de la distribución de Poisson) la parte fraccionaria restante se almacena para el siguiente instante de simulación. De esta manera se consiguen resultados que se ajustan con mayor precisión a la forma de la función trapezoidal.

3.4.3 Posición de generación de los vehículos

Para calcular la posición del tramo en la que deben aparecer los vehículos que se han de generar, existen dos alternativas, entre las que se podrá elegir en función de aquello que convenga más al escenario que se simula.

- Por un lado, se puede hacer que cada uno de los vehículos a generar aparezca en una posición aleatoria del tramo, es decir, los vehículos aparecerán a lo largo del tramo correspondiente de una forma relativamente uniforme.
- La segunda alternativa consiste en generar vehículos solamente en determinadas posiciones del tramo. Estas posiciones pueden corresponderse con los lugares en los que desembocan otras calles en el tramo en estudio, con salidas de aparcamientos masivos, etc. Cada una de estas posiciones tendrá una determinada probabilidad de que los vehículos se generen allí. Esta opción resulta mucho más cercana a la situación que se da en la realidad.

En cualquiera de las dos opciones se puede escoger entre generar el vehículo en un carril determinado, o bien, que sea en cualquiera de los carriles del tramo con una determinada probabilidad.

3.4.4 Comprobación de distancia

La introducción de un nuevo vehículo en el tramo puede provocar que otros vehículos que circulan por el mismo se vean afectados, por lo tanto, antes de generar el nuevo vehículo hay que comprobar que no se pone en dificultades al resto de vehículos del tramo.

Denotando como x_g a la posición donde se quiere generar el vehículo y con el subíndice e a los vehículos existentes, es necesario que todos los vehículos ya existentes que circulan en ese tramo por el carril donde se desea generar el nuevo vehículo, cumplan una determinada distancia de seguridad que depende de que el vehículo existente circule por delante o por detrás de donde se desea realizar la generación.

- Si $x_e(t) > x_g$ la condición que se debe cumplir para poder generar el vehículo en esa posición es que entre el existente y el que se va a generar exista al menos espacio para otro vehículo, es decir:

$$x_g \leq x_e(t) - 2 \cdot L_e$$

- Si $x_e(t) \leq x_g$ la condición consiste en que la distancia que separa ambos vehículos sea igual al menos a la distancia de seguridad del vehículo existente (donde L_g es la longitud del vehículo que se desea generar):

$$x_e(t) + D_e^{seg}(t) \leq x_g - L_g$$

En la figura 3.16 se muestra la notación empleada en estas condiciones, en la que el vehículo amarillo representa al vehículo que se desea generar y el azul a un vehículo ya existente en ese mismo carril.

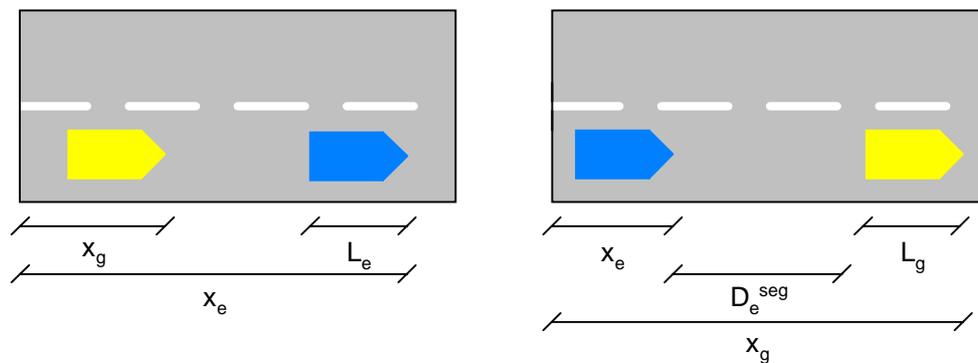


Figura 3.16. Notación empleada en la generación de vehículos

En el caso de que se cumplan estas condiciones para todos los vehículos del tramo, se podrá generar el vehículo en la posición deseada. De no ser así, la generación deberá ser pospuesta para el siguiente instante de simulación, cuando se volverán a hacer las comprobaciones correspondientes y se procederá a la generación del vehículo en el caso de que se cumplan las condiciones, y así sucesivamente.

4 Estadísticas

4.1 Introducción

El interés fundamental de la realización de una simulación microscópica de tráfico radica en poder analizar los resultados que se obtienen de la simulación para sacar las conclusiones convenientes. Para la realización de este análisis resulta conveniente la existencia de algún módulo dentro del simulador que aporte todos los datos y estadísticas necesarias para facilitar la labor.

Las estadísticas pueden ser de naturaleza muy distinta, por ejemplo, pueden abordar datos relacionados con los cambios de carril que se producen, generación de vehículos y otros que se verán en sucesivos apartados.

Las estadísticas pueden ser generadas en formatos muy diferentes, como ficheros de texto, gráficos o cualquier otro tipo de formato que haga claramente inteligible su información. En el presente proyecto se ha optado por generarlas en forma de ficheros de texto, para ser posteriormente procesados y sacar de ellos los gráficos que pudieran resultar de interés.

4.2 Estadísticas sobre cambios de carril

En relación con los cambios de carril que efectúan los vehículos en el transcurso de la simulación se pueden extraer las estadísticas que se enumeran a continuación.

- Un primer dato de interés es el saber cuántos cambios de carril en total se han producido durante la simulación. Además, interesa realizar una comparativa sobre el número de cambios de cada uno de los tipos estudiados con anterioridad. Estas estadísticas se extraen para cada uno de los tramos y para la totalidad de ellos.
- Para todos los cambios de carril que se producen en cada tramo se almacena la posición en la que han tenido lugar. Esto permitirá realizar un estudio acerca de la posición en la que suelen producirse los distintos tipos de cambio de carril, para poder comprobar que los cambios obligatorios suelen ocurrir más cerca del final del tramo que los de mejora.
- También se extraen las estadísticas sobre el gap, tanto delantero como trasero, de todos los cambios acaecidos. La conclusión a extraer debe ser que a la hora de realizar un cambio obligatorio se deben aceptar gaps más pequeños que en los de mejora. Además, en los cambios por cortesía el gap trasero aceptado debe resultar más pequeño que en los otros dos tipos de cambios de carril.
- Otra estadística de interés es el número de cambios de carril que realiza cada uno de los vehículos durante el tiempo que dura la simulación. De estos datos se calcula el número medio de cambios de carril que realiza un vehículo en el escenario a simular.

4.3 Estadísticas sobre la generación de vehículos

La generación de vehículos en los diferentes tramos es otro aspecto importante sobre el que es deseable conocer ciertas estadísticas:

- En primer lugar, el número de vehículos generados en cada instante de tiempo en cada uno de los tramos va a servir para verificar si realmente se cumplen las funciones de generación que se han expuesto con anterioridad.
- Otro dato a considerar es la posición en la que se generan los vehículos. Esto va a permitir verificar las condiciones impuestas para la posición de generación de vehículos, ya sea equiprobable a lo largo de todo el tramo o sólo en determinadas posiciones.
- La posición media en la que se generan los vehículos en cada tramo va a facilitar el conocimiento de, por ejemplo, cuál sería la zona del tramo en la que podría ser más recomendable habilitar un nuevo punto de acceso al tramo para los vehículos.

4.4 Estadísticas relacionadas con los semáforos

En relación con los vehículos que se encuentran detenidos en un semáforo, es necesario conocer:

- Para todos los vehículos que se detienen en un semáforo resulta interesante saber los instantes de tiempo en los que el vehículo se detiene y vuelve a arrancar, de esta forma se sabrá el tiempo que cada vehículo está parado en los semáforos de cada uno de los tramos. A este tiempo se le denomina *tiempo de espera en semáforo*.
- Conociendo los valores medios de los tiempos anteriores para cada tramo, y comparándolos con otros datos como el tiempo medio que un vehículo está en el tramo, se podrá determinar si resulta conveniente disminuir el tiempo que el semáforo permanece en rojo.
- El número de vehículos que se encuentran detenidos en cada instante en los semáforos también puede ser de interés para comparar los distintos tramos, y sacar conclusiones acerca de los mismos.

4.5 Estadísticas genéricas

A continuación se exponen otra serie de estadísticas de especial interés:

- Número de vehículos que circulan en cada instante de tiempo por los carriles de todos los tramos, así como el número total de vehículos en cada tramo. Los valores medios de estas cantidades permitirán conocer cuáles son aquellos tramos que tienen mayor saturación de vehículos, así como los carriles más transitados.

- Velocidad media de los vehículos en cada tramo que compone el viario, para lo que es necesario conocer el tiempo que circula por cada tramo y el espacio del mismo que recorre. A continuación, se puede realizar una comparativa entre las velocidades medias desarrolladas en cada tramo.
- Tiempo que permanece un vehículo en cada uno de los tramos y tiempo medio que tarda un vehículo en circular por un determinado tramo. Estos datos darán a conocer cuáles son los tramos que se recorren en mayor o menor tiempo.
- Tiempo en el sistema de los vehículos, que incluye tanto el tiempo empleado en recorrer los tramos como el tiempo empleado en los movimientos, es decir, en el interior de las intersecciones entre tramos. Tanto en esta estadística como en la comentada en el punto anterior se incluye el tiempo que un vehículo está parado en un semáforo, pero no así el que se encuentre aparcado.
- Para finalizar, se encuentran las estadísticas relativas a los aparcamientos realizados. Sobre ellos, resulta interesante conocer el instante, el tramo y la posición en que se realizan cada uno de ellos. Calculando el número total de aparcamientos realizados en cada tramo y la posición media en la que se producen en el interior de los mismos, se podrán extraer datos acerca de los tramos más solicitados para aparcar y las zonas de ellos en las que resulta más apetecible realizar el aparcamiento.

5 Modelado orientado a objetos

5.1 Introducción a UML

En este apartado se va a hacer una revisión de cómo surge, qué es y por qué se utiliza el Lenguaje de Modelado Unificado (UML). Este lenguaje es una especificación de notación orientada a objetos y se emplea para modelar el sistema y representar las distintas relaciones entre los componentes del mismo.

5.1.1 Breve historia de UML

Los lenguajes de modelados orientados a objetos aparecieron a mediados de los 70 y finales de los 80, influidos por técnicas como los modelos Entidad / Relación y el *Specification & Description Language* (SDL, circa 1976, CCITT), como diversas metodologías que se aproximaban al análisis y diseño orientados a objetos.

El número de lenguajes de modelado sufrió un espectacular incremento entre los años 1989 y 1994, llegando a identificarse hasta 50 lenguajes diferentes con los problemas que esto conllevaba para los usuarios. A mediados de los 90 aparecieron nuevas interacciones entre los diversos métodos, destacando el Booch'93, la evolución del OMT (*Object Modeling Techniques*) y Fusion. Los métodos empezaron a incluir técnicas unos de los otros, predominando un pequeño grupo de ellos, incluyendo el OOSE (*Object Oriented Software Engineering*), OMT-2 y el Booch'93. Cada uno de ellos era un método completo con sus puntos fuertes y flaquezas.

Simplificando, OOSE era un lenguaje que se orientaba hacia los casos de usos, dando un excelente soporte para los análisis de requerimientos en la ingeniería. OMT-2 era especialmente adecuado para el análisis de sistemas de información y bases de datos. Finalmente el Booch'93 destacaba por su utilidad en las fases de diseño y construcción de proyectos, llegando a ser muy popular para el desarrollo de proyectos de ingeniería.

Con este panorama, en Octubre del 94, comenzó el desarrollo del UML cuando Grady Booch y Jim Rumbaugh, del grupo *Rational Software Corporation*, comenzaron un trabajo conjunto para unificar los métodos Booch y OMT. Se daba el caso de que los métodos Booch y OMT estaban conocidos en todo el mundo como los líderes entre los métodos de modelado orientados a objetos, por lo que Booch y Rumbaugh unieron sus esfuerzos para unificar sus trabajos. Un primer boceto (versión 0.8) del *Unified Method*, como fue llamado entonces, se dio a conocer en Octubre del 95. A finales de 1995, Ivar Jacobson y su compañía se unieron a *Rational* y a su esfuerzo por unificar el lenguaje, aportando lo ya desarrollado en el OOSE.

Los autores de los métodos Booch, OMT y OOSE (Grady Booch, Jim Rumbaugh e Ivar Jacobson), estaban por lo tanto decididos a crear un lenguaje unificado de modelado por tres razones:

- En primer lugar, los métodos se estaban desarrollando unos independientemente de los otros. Era de sentido común continuar la

evolución conjuntamente mejor que por separado, eliminando la posibilidad de crear diferencias innecesarias que posteriormente podrían confundir a los usuarios.

- Por otro lado, unificar la nomenclatura y la semántica traería estabilidad al mercado de los lenguajes orientados a objetos, permitiendo que los proyectos se centraran en un solo lenguaje, dejando a los desarrolladores de herramientas crear mejores aplicaciones.
- Finalmente, ellos esperaban que su colaboración pudiera conllevar importantes mejoras en los tres métodos, de forma que se pudieran abarcar nuevos problemas que ninguno de los tres métodos originales resolvía adecuadamente.

Cuando se comenzó la unificación, se establecieron cuatro objetivos primordiales en los que centrar sus esfuerzos:

- Permitir el modelado de sistemas, y no sólo de software, usando conceptos de la orientación hacia objetos.
- Modelar el sistema, desde el concepto hasta los artefactos ejecutables, utilizando técnicas orientadas a objetos.
- Cubrir las cuestiones relacionadas con el tamaño inherente a los sistemas complejos y críticos.
- Crear un lenguaje de modelado utilizable tanto por las personas como por las máquinas.

En Junio de 1996 se publicaron las versiones 0.9 y 0.91. Durante ese año los autores invitaron, y recibieron respuesta, de toda la comunidad internacional relacionada con el tema. Además, varias corporaciones vieron el UML como un punto estratégico en el desarrollo de sus actividades, algunas de las empresas que contribuyeron a la aparición de la versión 1.0 en Enero de 1997 fueron IBM, Hewlett - Packard, Dell, Texas Instruments... Esta versión fue ofrecida a la OMG (*Object Management Group*) para su estandarización, en respuesta a su solicitud de propuestas para un lenguaje estándar de modelado.

Entre Enero y Julio de 1997, el grupo inicial de colaboradores se amplió para incluir prácticamente a todas las organizaciones que habían aportado algo al proyecto, creándose un grupo de trabajo liderado por Cris Kobryn para trabajar en la semántica. Como resultado se le entregó a la OMG la versión 1.1 para su estandarización en Julio de 1997. Esta versión se aceptó el 14 de Noviembre de 1997 por la OMG.

El control del mantenimiento de UML fue asumido por la *OMG Revision Task Force*, dirigida por Cris Kobryn. La RTF publicó una revisión editorial, UML 1.2, en Junio de 1998. En otoño de 1998 la RTF publicó UML 1.3, que es la versión utilizada en este proyecto.

5.1.2 ¿Qué es UML? ¿Por qué usar UML?

UML es un lenguaje estándar para escribir planos de software. Se puede utilizar para especificar, construir y documentar los artefactos de un sistema que involucra una gran cantidad de software.

UML es apropiado para modelar desde sistemas de información de empresas, hasta aplicaciones distribuidas basadas en la web, e incluso para sistemas empotrados de tiempo real muy exigentes. Es un lenguaje expresivo, que cubre todas las vistas necesarias para desarrollar y luego desplegar tales sistemas.

UML es sólo un lenguaje, por lo tanto es tan sólo una parte de un método de desarrollo de software. Es independiente del proceso, aunque para utilizarlo óptimamente se debería usar en un proceso que fuese dirigido por los casos de uso, centrado en la arquitectura, iterativo e incremental.

En general, UML es un lenguaje para visualizar, especificar, construir y documentar los artefactos de un sistema con gran cantidad de software.

Como lenguaje, UML proporciona un vocabulario y unas reglas para combinar las palabras de dicho vocabulario con el objetivo de posibilitar la comunicación. Un lenguaje de modelado es un lenguaje cuyo vocabulario y reglas se centran en la representación conceptual y física del sistema. Un lenguaje de modelado como UML es, por tanto, un lenguaje estándar para los planos de software.

El modelado proporciona una comprensión de un sistema. Nunca es suficiente un único modelo. Más bien, para comprender cualquier cosa, a menudo se necesitan múltiples modelos conectados entre sí, excepto en los casos más triviales. Para sistemas con gran cantidad de software, se requiere un lenguaje que cubra las diferentes vistas de la arquitectura de un sistema mientras evoluciona a través del ciclo de vida del desarrollo del software.

El vocabulario y las reglas de un lenguaje como UML indican cómo crear y leer modelos bien formados, pero no dicen qué modelos se deben crear y cuándo se deberían crear. Ésta es la tarea del proceso del desarrollo del software. Un proceso bien definido guiará a su usuario a decidir qué artefactos producir, qué actividades y personal se emplean para crearlos y gestionarlos, y cómo usar estos artefactos para medir y controlar el proyecto de forma global.

Para muchos programadores, la distancia entre pensar en una implementación y transformarla en código es casi nula. Lo piensas, lo codificas. De hecho, algunas cosas se modelan mejor directamente en código. En estos casos, el programador todavía está haciendo mentalmente algo de modelado, si bien lo hace de forma completamente mental. No obstante, esta manera de proceder plantea algunos problemas:

- Primero, la comunicación de estos modelos conceptuales se torna complicada y está sujeta a errores, salvo que las dos partes hablen el mismo lenguaje. Normalmente, los proyectos y las organizaciones desarrollan su propio lenguaje, y es difícil de comprender lo que está pasando para alguien ajeno al grupo.

- Segundo, hay algunas cuestiones en un sistema de software que no se pueden entender a menos que se construyan sobre modelos que trasciendan el lenguaje de programación textual. Por ejemplo, el significado de una jerarquía de clases puede inferirse, pero no capturarse completamente inspeccionando el código de todas las clases en la jerarquía.
- Tercero, si el desarrollador que escribió el código no dejó documentación escrita de los modelos que hacía en su cabeza, esa información se perderá para siempre, o como mucho, será sólo parcialmente reproducible a partir de la implementación una vez que el desarrollador se haya marchado.

Al escribir modelos en UML, se afronta el tercer problema: un modelo explícito facilita la comunicación.

Algunas cosas se modelan mejor textualmente; otras se modelan mejor de forma gráfica. En realidad, en todos los sistemas interesantes hay estructuras que trascienden lo que puede ser representado mediante un lenguaje de comunicación. UML es uno de estos lenguajes gráficos. Así afronta el segundo problema mencionado anteriormente.

UML es algo más que un simple montón de símbolos gráficos. Más bien, detrás de cada símbolo en la notación UML hay una semántica bien definida. De manera que un desarrollador puede escribir un modelo en UML y otro desarrollador, o incluso otra herramienta, puede interpretar ese modelo sin ambigüedad. Así afronta el primer problema mencionado con anterioridad.

En este contexto, especificar significa construir modelos precisos, no ambiguos y completos. En particular, UML cubre la especificación de todas las decisiones de análisis, diseño e implementación que se deben realizar al desarrollar y desplegar un sistema con gran cantidad de software.

UML no es un lenguaje de programación visual, pero sus modelos pueden conectarse a gran variedad de lenguajes de programación. Esto significa que es posible establecer correspondencias desde un modelo UML a un lenguaje de programación como C++, Java o Visual Basic, o incluso tablas en una base de datos relacional o almacenamiento persistente en una base de datos orientada a objetos. Las cosas que se expresan mejor gráficamente también se representan gráficamente en UML, mientras que las cosas que se representan mejor textualmente se plasman con el lenguaje de programación.

Esta correspondencia permite la ingeniería directa: generación de código a partir de un modelo UML en un lenguaje de programación. Lo contrario también es posible: se puede reconstruir un modelo en UML a partir de una implementación. Pero este proceso no es automático, a menos que se codifique esa información en la implementación, la información se pierde cuando se pasa de los modelos al código. La ingeniería inversa requiere, por lo tanto, herramientas que la soporten e intervención humana. La combinación de estas dos vías de generación de código y de ingeniería inversa produce una ingeniería "de ida y vuelta", entendiéndose por eso la posibilidad de trabajar en una vista gráfica o textual, mientras que las herramientas mantienen la consistencia de las dos vistas.

Además de esta correspondencia directa, UML es lo suficientemente expresivo y no ambiguo como para permitir la ejecución directa de modelos, la simulación de sistemas y la instrumentación de sistemas en ejecución.

Hoy en día, una organización de software produce toda clase de artefactos además de código ejecutable. Estos artefactos incluyen (sin limitarse a ello):

- Requisitos.
- Arquitectura.
- Diseño.
- Código fuente.
- Planificación de proyectos.
- Pruebas.
- Prototipos.
- Versiones.

Dependiendo de la cultura de desarrollo, algunos de estos artefactos se tratan más o menos formalmente que otros. Tales artefactos no son sólo los entregables de un proyecto, también son críticos en el control, la medición y comunicación que requiere un sistema durante su desarrollo y después de su despliegue.

UML cubre la documentación de la arquitectura de un sistema y todos sus detalles. UML también proporciona un lenguaje para expresar requisitos y pruebas. Finalmente, UML proporciona un lenguaje para modelar las actividades de planificación de proyectos y gestión de versiones.

5.2 Especificación funcional del simulador mediante UML

El simulador microscópico de tráfico del que es objeto el presente proyecto ha sido desarrollado en un lenguaje de programación orientado a objetos como es C++, por lo tanto, resulta adecuado el empleo de UML para explicar su diseño.

A continuación se presentan algunos de los diagramas que forman parte del lenguaje de modelado unificado (UML) aplicados al simulador, realizando una breve introducción a los mismos para explicar su notación y sus objetivos.

5.2.1 Diagramas de casos de uso

Los casos de uso permiten describir el comportamiento de un sistema desde el punto de vista del usuario basándose en un conjunto de acciones y reacciones. Es por lo tanto una técnica que permite capturar los requisitos funcionales del sistema. De esta forma queda delimitado el alcance del sistema y cuál es su relación con el entorno.

En estos diagramas, el sistema queda reducido a una "caja negra", ya que no interesa cómo lleva a cabo sus funciones, sino simplemente qué acciones visibles desde el exterior son las que realiza.

Los casos de uso están basados en lenguaje natural, lo que los hace accesibles a cualquier usuario. Además, aquellos casos de uso que resulten muy complejos pueden descomponerse en nuevos casos de uso de un nivel inferior, hasta llegar a un nivel tal que resulten fáciles de analizar.

Los casos de uso guían todo el proceso de desarrollo del sistema, lo que quiere decir que en momentos determinados de dicho proceso, el sistema debe ser validado comprobando que se ajusta al diagrama de casos de uso.

Los diagramas de casos de uso están formados por 3 elementos fundamentales:

- **Actores.** Los actores son los participantes de los casos de uso, se corresponden con los usuarios que interactúan con el sistema. Estos actores pueden ser seres humanos, dispositivos externos que interactúen con el sistema, o incluso temporizadores que envíen eventos al sistema. Un actor se caracteriza por la forma de interactuar con el sistema, por lo que un mismo usuario puede ejercer de varios actores, y un actor puede representar a varios usuarios.
- **Casos de uso.** Son los escenarios de interacción de los actores. Representan el comportamiento del sistema en relación con los usuarios. De esta forma, un caso de uso define la secuencia de interacciones entre uno o más usuarios y el sistema.
- **Relaciones.** Representan el flujo de información intercambiada entre los actores y los casos de uso, o entre diferentes casos de uso. Normalmente, se emplean para que un caso de uso obtenga la información necesaria para llevar a cabo alguna acción, o para que el proceso proporcione algún resultado. Estas relaciones pueden ser unidireccionales o bidireccionales.

Los diagramas de casos de uso se clasifican en diferentes niveles, en función del grado de detalle con el que se represente el funcionamiento del sistema. De esta forma, los diagramas de *nivel 0* representan el sistema completo con un nivel de detalle muy bajo, mientras que al aumentar el nivel, va incrementándose el grado de detalle.

Notación

A continuación se muestra la notación empleada para la representación de los distintos elementos que forman de los diagramas de casos de uso.

Los actores son representados mediante figuras de *hombres de palo*, con su nombre correspondiente debajo de la figura.



Figura 5.1. Notación de actores

Los casos de uso, o procesos, se representan mediante una elipse, con su nombre correspondiente debajo o dentro de la misma.



Figura 5.2. Notación de casos de uso

Por último, las relaciones se representan mediante flechas que unen los casos de uso, o el caso de uso y el actor, entre los que existe un flujo de información.



Figura 5.3. Notación de relaciones

5.2.1.1 Diagrama de casos de uso de nivel 0

En este diagrama se muestra el simulador con el menor grado de detalle, dando una visión global del funcionamiento del mismo.

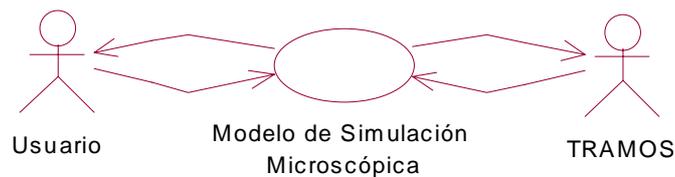


Figura 5.4. Casos de uso de nivel 0

El *usuario* es la persona que se encuentra realizando la simulación. Su relación con el *Modelo de Simulación Microscópica* es bidireccional, ya que le proporciona los parámetros necesarios para la simulación, como son los correspondientes a los tipos de conductores y los tipos de vehículos, mientras que recibe del mismo los resultados generados en la simulación.

El otro actor que interviene en este diagrama es la herramienta *TRAMOS* (Traffic and Transport Analysis Modeling and Optimization System). Esta herramienta está dedicada al modelado y la optimización del tráfico en las ciudades, y también tiene una relación bidireccional con el *Modelo de Simulación Microscópica*.

TRAMOS proporciona al modelo de simulación toda la información que hace referencia al viario que se desea simular, desde los tramos y nodos que lo conforman, hasta las paradas o aparcamientos que deben realizar los vehículos, pasando por los movimientos, giros y cajas semafóricas. Además, obtiene del simulador, las estadísticas que han sido generadas, para darles un formato de representación adecuado.

5.2.1.2 Diagrama de casos de uso de nivel 1

En este nivel, se describe con un grado mayor de detalle el funcionamiento del *Modelo de Simulación Microscópica*.

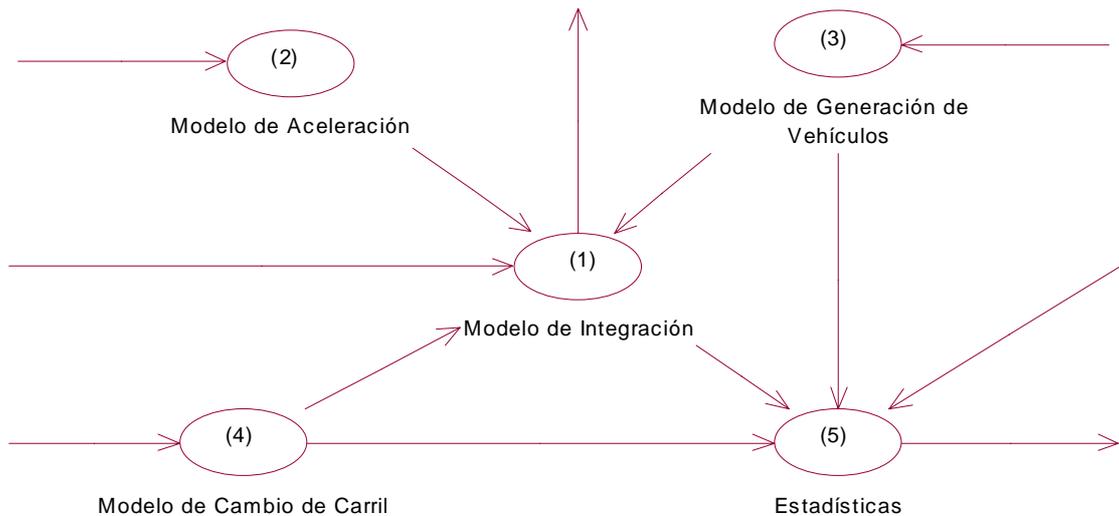


Figura 5.5. Casos de uso de nivel 1

En este diagrama se aprecia que el simulador está formado por 5 procesos que intercambian información entre ellos. A continuación se explican brevemente cada uno de ellos, entrando a detallar los modelos de cambio de carril, de generación de vehículos y estadísticas.

El *Modelo de Integración* (1) es el encargado de reunir los distintos modelos existentes en el simulador y agruparlos de forma que se ejecuten de forma coordinada. Por ello, es necesario que reciba los resultados de la ejecución de los otros modelos. Además, recibe de *TRAMOS* la información relacionada con el viario y le aporta algunos de los resultados de la simulación. Estos resultados también son empleados por el módulo de *Estadísticas* para su procesamiento.

El *Modelo de Aceleración* (2) es el encargado de obtener la aceleración que deben desarrollar los vehículos en cada instante de simulación, en función de las influencias que se encuentra, ya sean del viario o de otros vehículos. Es por ello, que recibe información de *TRAMOS*, mientras que el resultado que genera es empleado por el *Modelo de Integración*.

El *Modelo de Generación de Vehículos* (3) tiene la misión de generar los nuevos vehículos que se introducen en el escenario. Para ello, necesita conocer el viario, las funciones de generación de vehículos asociadas a cada tramo y los vehículos que ya se

encuentran circulando. Los vehículos generados son empleados por el *Modelo de Integración*, así como por el módulo de *Estadísticas*.

El *Modelo de Cambio de Carril* (4) es el encargado de comprobar si es necesario, y realizar en su caso, las maniobras de cambio de carril. Para su realización, necesita conocer los datos relacionados con los distintos tipos de conductores y de vehículos, así como los datos correspondientes al viario. Las maniobras que se llevan a cabo deben ser comunicadas al *Modelo de Integración* y al módulo de *Estadísticas*.

El módulo de *Estadísticas* (5) se encarga de procesar los resultados generados por los distintos modelos del simulador para generar las estadísticas correspondientes a la simulación. Dichas estadísticas son proporcionadas al *Usuario* de la figura 5.4 en un formato adecuado. Este módulo también necesita conocer los datos del viario que le proporciona *TRAMOS*.

5.2.1.3 Diagrama de casos de uso de nivel 2

El siguiente caso de uso muestra con mayor nivel de detalle los modelos desarrollados en este proyecto, como son los correspondientes al cambio de carril, la generación de vehículos y las estadísticas.

Modelo de Cambio de Carril

El *Modelo de Cambio de Carril* (4) diseñado en el simulador se puede desagregar de la forma indicada en la figura 5.6.

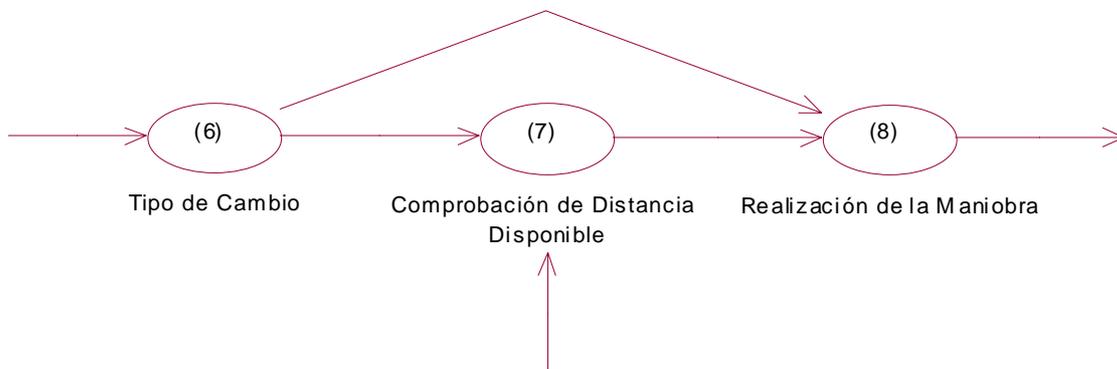


Figura 5.6. Casos de uso de nivel 2. Modelo de Cambio de Carril

Para la realización de un cambio de carril, lo primero es conocer qué tipo de cambio de carril se desea llevar a cabo. Este procedimiento se realiza en el proceso *Tipo de Cambio* (6), donde es necesario conocer la información referente al viario, tal y como la posición del vehículo en el tramo o el itinerario que debe seguir. El tipo de cambio de carril que se decida realizar, es empleado tanto por el proceso de *Comprobación de Distancia Disponible*, como por el de *Realización de la Maniobra*.

El caso de uso de *Comprobación de Distancia Disponible* (7) es el proceso encargado de comprobar si existe distancia suficiente para realizar con seguridad la maniobra de cambio de carril. Esta distancia depende del tipo de cambio que se quiera realizar y de la posición del resto de vehículos en el tramo. El resultado de este

procedimiento debe ser comunicado al proceso *Realización de la Maniobra* para poder llevarlo a cabo.

En el proceso de *Realización de la Maniobra* (8), se ejecutan las acciones necesarias para llevar a un vehículo de un carril a otro del mismo tramo. El resultado de este proceso es notificado al *Modelo de Integración* de la figura 5.5.

Modelo de Generación de Vehículos

El diagrama de casos de uso correspondiente al *Modelo de Generación de Vehículos* se muestra en la figura 5.7.

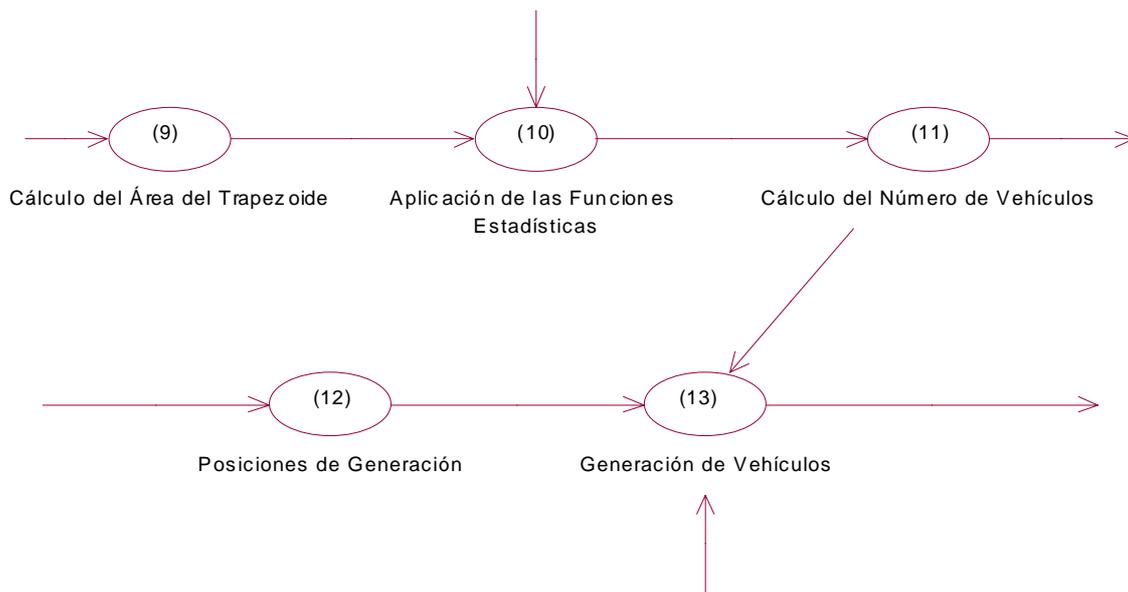


Figura 5.7. Casos de uso de nivel 2. Modelo de Generación de Vehículos

La generación de vehículos en un tramo comienza mediante el *Cálculo del Área del Trapezoide* (9) correspondiente al intervalo de simulación en el que se deba realizar la generación y teniendo en cuenta la forma de la función trapezoidal asociada al tramo en cuestión.

El valor obtenido del área del trapezoide es modificado mediante la *Aplicación de las Funciones Estadísticas* (10) que estén asociadas a cada uno de los tramos, de esta forma se realiza el *Cálculo del Número de Vehículos* (11), que indica el número de vehículos que han de generarse en cada tramo en cada instante de simulación.

En el caso de uso *Posiciones de Generación* (12) se determina dónde deben ser introducidos los nuevos vehículos dentro del tramo. Para ello, puede emplear unas posiciones predeterminadas con ciertas probabilidades de ocurrencia, o bien, unas posiciones totalmente aleatorias.

Una vez conocidos el número de vehículos que hay que generar y las posiciones donde se deben generar, se lleva a cabo el proceso de *Generación de Vehículos* (13), donde también se tienen en cuenta los valores de los parámetros predeterminados para

los diferentes vehículos, y cuyo resultado (vehículos generados) es empleado en el *Modelo de Integración* (1) de la figura 5.5.

Estadísticas

El módulo de *Estadísticas* (5) se descompone en dos casos de uso interrelacionados que se muestran en la figura 5.8.



Figura 5.8. Casos de uso de nivel 2. Estadísticas

El *Procesamiento de Estadísticas* (14) recoge los datos relacionados con los distintos aspectos de la simulación sobre los que se desea realizar las correspondientes estadísticas (enumerados en el apartado 4) y genera las estructuras de datos adecuadas que almacenan dichas estadísticas.

En el caso de uso *Escritura de Estadísticas* (15) se lleva a cabo la escritura de las estadísticas generadas en el proceso anterior en sus correspondientes ficheros de texto. Estos ficheros de texto son utilizados por *TRAMOS* para generar gráficas que sean útiles para el *Usuario*.

5.2.1.4 Diagrama de casos de uso de nivel 3

El nivel de detalle alcanzado en los diagramas de casos de uso del apartado anterior es suficiente para comprender el funcionamiento general del Simulador Microscópico de Tráfico. Sin embargo, resulta conveniente ahondar un poco más en el proceso de *Comprobación de Distancia Disponible* (7) para la realización de un cambio de carril.

Comprobación de Distancia Disponible

Este proceso, también conocido como aceptación de *gap*, se encarga de comprobar si existe distancia suficiente en el nuevo carril antes de iniciar la maniobra de cambio de carril.

La desagregación de este proceso en otros con mayor nivel de detalle se observa en el diagrama de la figura 5.9.

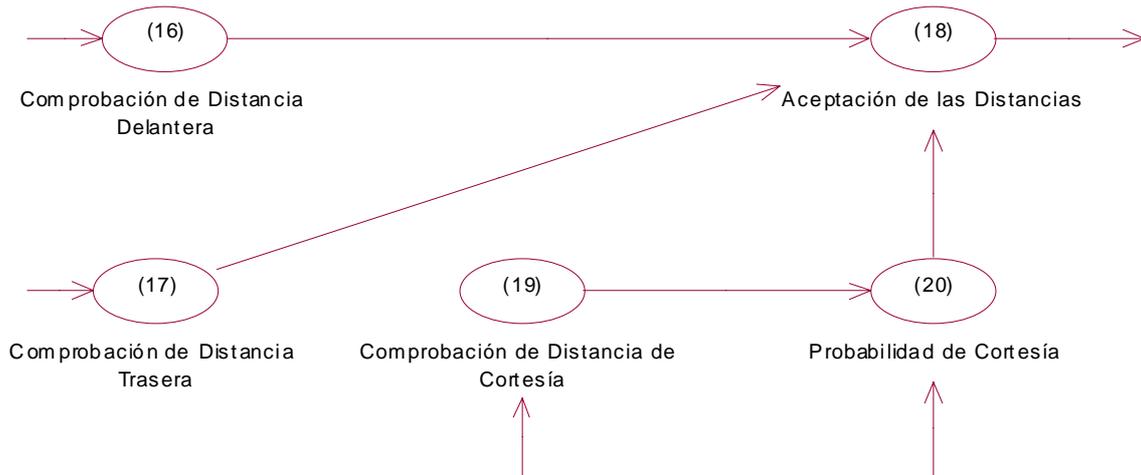


Figura 5.9. Casos de uso de nivel 3. Comprobación de Distancia Disponible

En los casos de uso *Comprobación de Distancia Delantera* (16) y *Comprobación de Distancia Trasera* (17), se verifica si el espacio, anterior y posterior respectivamente, que se encontraría el vehículo en el nuevo carril es suficiente para realizar el cambio de carril con seguridad. El resultado es utilizado por el proceso *Aceptación de las Distancias* (18).

En el caso de uso *Comprobación de Distancia de Cortesía* (19) se verifica si existe un espacio trasero suficiente en el nuevo carril para realizar el cambio de carril, en el caso de que el posible vehículo sucesor en el nuevo carril esté dispuesto a permitir la maniobra.

Con el resultado del proceso anterior y teniendo en cuenta los parámetros del tipo de conductor que conduce el futuro vehículo sucesor, el caso de uso *Probabilidad de Cortesía* (20) se encarga de determinar si dicho vehículo sucesor estaría dispuesto a permitir el cambio de carril.

Finalmente, el proceso *Aceptación de las Distancias* (18) determina si existe espacio suficiente para la realización de la maniobra. Su resultado es utilizado por el caso de uso *Realización de la Maniobra* (8) de la figura 5.6.

5.2.2 Diagrama de paquetes

El objetivo de este tipo de diagramas es obtener una visión mucho más clara del sistema de información orientado a objetos, organizándolo en diferentes subsistemas, agrupando los elementos del análisis, diseño o construcción y detallando las relaciones de dependencia entre ellos. El mecanismo de agrupación utilizado se denomina *paquete*.

Estrictamente hablando, los paquetes y sus dependencias son elementos de los diagramas de casos de uso, de clases y de componentes, por lo que se podría decir que el diagrama de paquetes es una extensión de éstos.

En estos diagramas se pueden diferenciar dos tipos de elementos:

- Paquetes: Un paquete es la agrupación de elementos, bien sea casos de uso, clases o componentes. Los paquetes pueden contener a su vez otros paquetes anidados que en última instancia contendrán alguno de los elementos anteriores.
- Dependencia entre paquetes: Existe una dependencia cuando un elemento de un paquete requiere de otro que pertenece a un paquete distinto. Es importante resaltar que las dependencias no son transitivas.

Los paquetes se van a representar mediante un símbolo en forma de carpeta, en el cual se coloca el nombre del paquete correspondiente. Por su parte, las dependencias entre paquetes se representan con una flecha discontinua con inicio en el paquete que depende del otro.

5.2.2.1 Descripción de los paquetes existentes en el simulador

Las distintas clases que componen el simulador microscópico de tráfico se han agrupado en seis paquetes de datos relacionados entre sí. Estos paquetes son:

- Red: Este paquete contiene las clases utilizadas para describir el viario. Por lo tanto, contendrá toda la información referente a los tramos y nodos existentes en el escenario de simulación, es decir, todo lo que hace referencia a los componentes estáticos del simulador.
- Grupo Semafórico: Se emplea para describir el funcionamiento de los semáforos que regulan el tráfico en las intersecciones del viario.
- Simulador Microscópico: Los datos contenidos en este paquete contienen toda la información que resulta precisa y necesaria para la representación de los componentes dinámicos que forman parte del simulador.
- Generación Vehículos: Este paquete de datos recoge la información que es utilizada en el procedimiento de generación de vehículos.
- Estadística: Recoge las estructuras de datos empleadas para la generación de distintas estadísticas del proceso de simulación.
- Listas: En este paquete se recoge la clase que se emplea para la creación y manejo de listas de cualquiera de los componentes pertenecientes a los paquetes anteriores.

5.2.2.2 Diagrama de paquetes del simulador

El diagrama de paquetes del simulador microscópico de tráfico es el que se presenta a continuación:

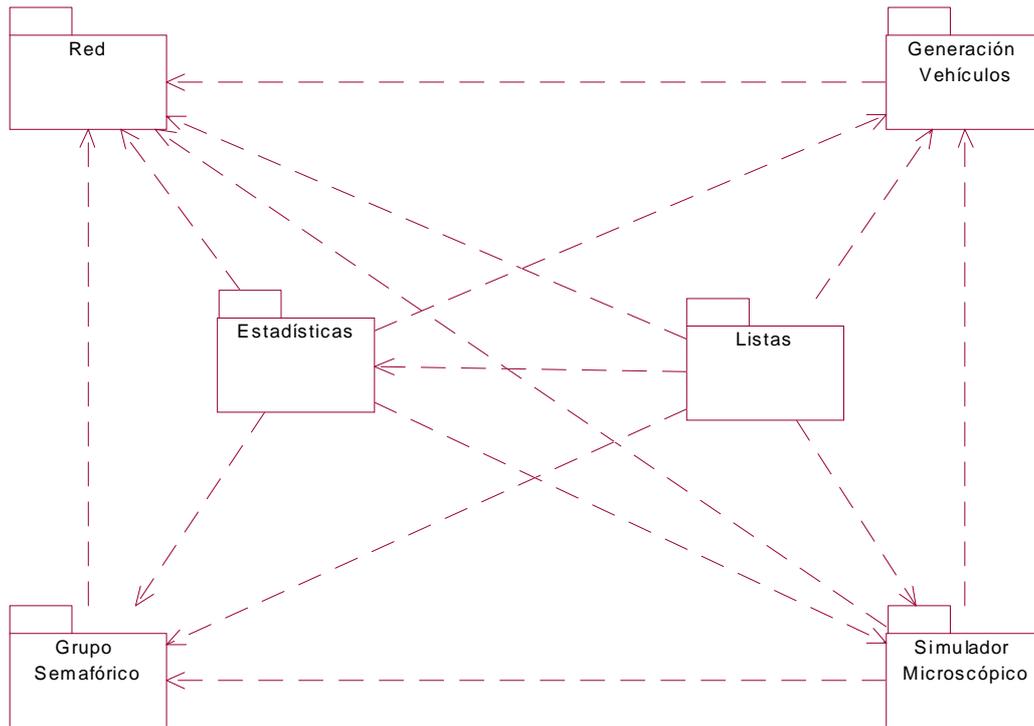


Figura 5.10. Diagrama de paquetes del simulador

5.2.3 Diagramas de clases

El objetivo principal de este modelo es la representación de los aspectos estáticos del sistema, utilizando diversos mecanismos de abstracción (clasificación, generalización, agregación).

El diagrama de clases recoge las clases existentes y sus correspondientes asociaciones. En este diagrama se representa la estructura y el comportamiento de cada uno de los objetos del sistema y sus relaciones con los demás objetos, pero no muestra ninguna información temporal. Con el fin de facilitar la comprensión del diagrama, se pueden incluir paquetes como elementos del mismo, donde cada uno de ellos agrupa un conjunto de clases que tienen algún tipo de relación. Este diagrama no refleja los comportamientos temporales de las clases, aunque para mostrarlos se puede utilizar un diagrama de transición de estados.

Los elementos básicos del diagrama son:

- Clases. Una clase describe un conjunto de objetos con propiedades (atributos) similares y un comportamiento común. Los objetos son instancias de las clases.

No existe un procedimiento inmediato que permita localizar las clases del diagrama de clases. Éstas suelen corresponderse con sustantivos que hacen referencia al ámbito del sistema de información y que se encuentran en los documentos de las especificaciones de requisitos y los casos de uso.

Dentro de la estructura de una clase se definen los atributos, que representan los datos asociados a los objetos instanciados por esa clase, y las operaciones o métodos, que representan las funciones o procesos propios de los objetos de una clase, caracterizando a dichos objetos.

El diagrama de clases permite representar clases abstractas. Una *clase abstracta* es una clase que no puede existir en la realidad, pero que es útil conceptualmente para el diseño del modelo orientado a objetos. Las clases abstractas no son instanciables directamente sino en sus descendientes. Una clase abstracta suele ser situada en la jerarquía de clases en una posición que le permita ser un depósito de métodos y atributos para ser compartidos o heredados por las subclases de nivel inferior.

Las clases y en general todos los elementos de los diagramas, pueden estar clasificados de acuerdo a varios criterios, como por ejemplo su objetivo dentro de un programa. Esta clasificación adicional se expresa mediante un *estereotipo*. Algunos de los autores de métodos orientados a objetos, establecen una clasificación de todos los objetos que pueden aparecer en un modelo. Los tipos son:

- Objetos entidad.
- Objetos límite o interfaz.
- Objetos de control.

En función de la herramienta empleada, también se puede añadir información adicional a las clases para mostrar otras propiedades de las mismas, como son las reglas de negocio, responsabilidades, manejo de eventos, excepciones...

Dentro de las clases es importante destacar un caso especial, las *plantillas* o *templates*. Éstas son clases que están parametrizadas, es decir, que necesitan de un parámetro a la hora de formarse.

- Relaciones. Los tipos más importantes de relaciones estáticas entre clases son los siguientes:
 - Asociación. Las relaciones de asociación representan un conjunto de enlaces entre objetos o instancias de clases. Es el tipo de relación más general, y denota básicamente una dependencia semántica. Por ejemplo, un Carril pertenece a un Tramo. Cada asociación puede presentar elementos adicionales que doten de mayor detalle al tipo de relación:
 - Rol, o nombre de la asociación, que describe la semántica de la relación en el sentido indicado. Por ejemplo, la asociación entre Carril y Tramo recibe el nombre de *pertenece a*, como rol en ese sentido.
 - Multiplicidad, que describe la cardinalidad de la relación, es decir, especifica cuántas instancias de una clase están asociadas a una

instancia de la otra clase. Los tipos de multiplicidad son: uno a uno, uno a muchos, muchos a muchos...

- Herencia. Las jerarquías de generalización / especialización se conocen como herencia. La herencia es el mecanismo que permite a una clase de objetos incorporar atributos y métodos de otra clase, añadiéndolos a los que ya posee. Con la herencia se refleja una relación *es-un* entre clases. La clase de la cual se hereda se denomina *superclase*, y la que hereda *subclase*.
- Agregación. La agregación es un tipo de relación jerárquica entre un objeto que representa la totalidad de ese objeto y las partes que lo componen. Permite el agrupamiento físico de estructuras relacionadas lógicamente. Los objetos *son-parte-de* otro objeto completo.
- Composición. La composición es una forma de agregación donde la relación de propiedad es más fuerte, e incluso coinciden los tiempos de vida del objeto completo y las partes que lo componen.
- Dependencia. Una relación de dependencia se utiliza entre dos clases o entre una clase y una interfaz, e indica que una clase requiere de otra para proporcionar alguno de sus servicios.

A continuación, se va a dar una explicación de la notación seguida para la representación de estos diagramas.

Una clase se representa como una caja, separada en tres zonas por líneas horizontales.

En la zona superior se muestra el nombre de la clase y propiedades generales como el estereotipo. El nombre de la clase aparece centrado y si la clase es abstracta se representa en cursiva. El estereotipo, si se muestra, se sitúa sobre el nombre y entre el símbolo: << ... >>. Este nombre es único, y sirve para diferenciarla del resto de las clases.

La zona central contiene una lista de atributos, uno en cada línea. La notación utilizada para representarlos incluye, dependiendo del detalle, el nombre del atributo, su tipo y su valor por defecto, con el formato:

Visibilidad nombre: tipo = valor-inicial

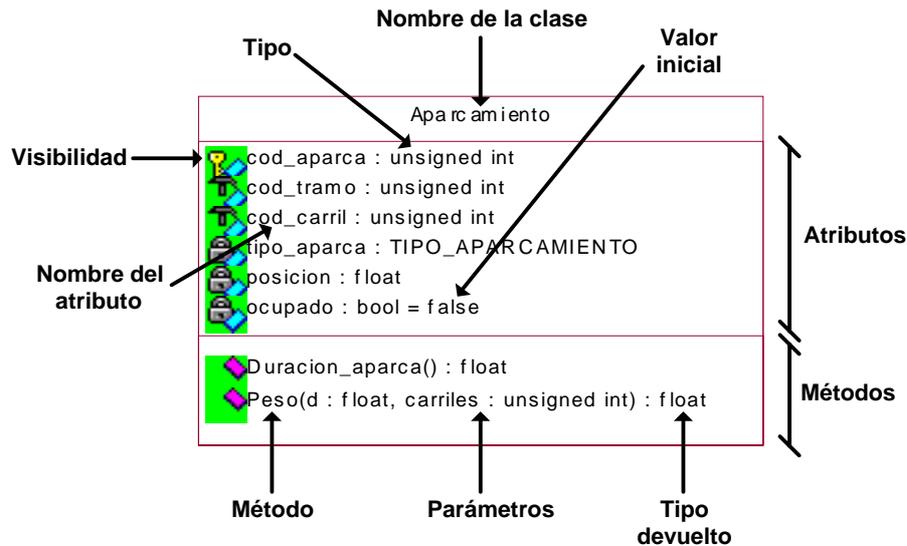


Figura 5.11. Ejemplo de clase

La visibilidad de los atributos puede ser pública (🔓) o privada (🔒). Dentro de los atributos privados, pueden darse otros dos tipos, como son las claves primarias (🔑) que indican que el atributo identifica unívocamente a la clase, y las claves ajenas (🔗) que indican que ese atributo, junto a otros identifican unívocamente a otra clase.

En la zona inferior se incluye una lista con las operaciones o métodos que proporciona la clase. Cada operación aparece en una línea con formato:

Visibilidad nombre (lista-de-parámetros): tipo-devuelto

La visibilidad de los métodos puede ser pública (🔓) o privada (🔒), aunque pueden existir otros tipos de visibilidad en función del lenguaje de programación.

En la definición de todas las clases que conforman el simulador del presente proyecto, se obviarán aquellos métodos que sirven para acceder o modificar los atributos de dicha clase. Asimismo, tampoco se expondrá la sobrecarga de los operadores de igualdad y comparación. Esto se debe a que están presentes en todas las clases y no aportan ningún dato clarificador acerca de las mismas.

Una relación de asociación se representa como una línea continua entre las clases asociadas. En una relación de asociación, ambos extremos de la línea pueden conectar con la misma clase, indicando que una instancia de una clase, está asociada a otras instancias de la misma clase, lo que se conoce como asociación reflexiva.

La relación puede tener un nombre y un estereotipo, que se colocan junto a la línea. El nombre suele corresponderse con expresiones verbales presentes en las especificaciones, y define la semántica de la asociación. Los estereotipos permiten clasificar las relaciones en familias y se escribirán entre los símbolos: << ... >>.

Las diferentes propiedades de la relación se pueden representar con la siguiente notación:

Multiplicidad: La multiplicidad puede ser un número concreto, un rango o una colección de números. La letra ‘n’ y el símbolo ‘*’ representan cualquier número.

Navegabilidad: La navegación desde una clase a la otra se representa poniendo una flecha sin relleno en el extremo de la línea, indicando el sentido de la navegación. En caso de no especificar nada el sentido es bidireccional.

Rol o nombre de la asociación: Este nombre se coloca junto al extremo de la línea que está unida a una clase, para expresar cómo esa clase hace uso de la otra clase con la que mantiene la asociación.



Figura 5.12. Ejemplo de asociación

Este ejemplo de asociación quiere decir que un tramo contiene de uno a *n* carriles y un carril pertenece a un tramo.

Además, existen notaciones específicas para los otros tipos de relación vistos con anterioridad, como son:

Agregación: Se representa con un rombo hueco en la clase cuya instancia es una agregación de las instancias de la otra.

Composición: Se representa con un rombo lleno en la clase cuya instancia contiene las instancias de la otra clase.

Dependencia: Una línea discontinua con una flecha apuntando a la clase cliente. La relación puede tener un estereotipo que se coloca junto a la línea, y entre los símbolos: << ... >>.

Herencia: Esta relación se representa como una línea continua con una flecha hueca en el extremo que apunta a la superclase.

5.2.3.1 Paquete de datos "Red"

Este paquete de datos es utilizado para representar todos los componentes principales que forman el viario de una ciudad. Dicho viario será representado por nodos y tramos. En concreto existirá un tramo para cada sentido de circulación que tengan las calles del viario, mientras que los nodos representan los cruces entre las distintas calles.

5.2.3.1.1 Descripción de las clases del paquete "Red"

5.2.3.1.1.1 Clase "Nodo"

Como ya se ha comentado, esta clase se utiliza para definir las intersecciones en el viario. A continuación se exponen los atributos y métodos que la componen.



Figura 5.13. Clase "Nodo"

Atributos de la clase "Nodo":

- *codigo_nodo*: Código que diferencia un nodo del resto de nodos de la red.
- *indice_senal*: Código de la señal que regula el tramo.
- *lista_movimientos*: Lista de los movimientos que tienen lugar en el nodo.
- *lista_giro*: Lista de los giros definidos en el nodo.
- *lista_vehiculos*: Lista de los vehículos que se encuentran circulando por el nodo.
- *lista_indices_movimientos*: Lista de los movimientos del nodo en la que sólo se indican los tramos y carriles origen y destino, además de un código.
- *lista_incompatibilidad*: Lista en la que se definen las incompatibilidades existentes entre los distintos movimientos del tramo.

Métodos de la clase "Nodo":

- *void Nodo (void)* : Crea las listas necesarias en un nodo.
- *void Destruir_lista (void)* : Destruye las listas anteriores.
- *void Modificar_elemento_indice (Movimiento m, unsigned int i)* : Actualiza el elemento *i* de la lista de movimientos al valor del movimiento *m*.

- *void Modificar_elemento_indice (Giro g, unsigned int i)* : Actualiza el elemento *i* de la lista de giros al valor del giro *g*.
- *void Add_elemento (Giro g)* : Añade el giro *g* a la lista de giros.
- *void Add_elemento (Movimiento m)* : Añade el movimiento *m* a la lista de movimientos.
- *void Add_elemento (Vehiculo v)* : Añade el vehículo *v* a la lista de vehículos.
- *void Add_elemento (In_movimiento m)* : Añade la incompatibilidad *m* a la lista de incompatibilidades.
- *void Add_elemento (Indice_movimiento m)* : Añade el índice de movimiento *m* a la lista de índices de movimientos.
- *unsigned int Buscar_influencia (Vehiculo veh, TIPO_INFLUENCIA *tipo_influencia, unsigned int tipo)* : Devuelve el tipo de influencia por la que se ve afectado el vehículo *veh* en el nodo.
- *unsigned int Buscar_influencia_vehiculo (Vehiculo veh, float* distancia_veh, unsigned int tipo)* : Devuelve el código del vehículo por el que se ve influido *veh* en el caso de que existiera.
- *void Simular (float At, float tiempo_simulacion)* : Realiza la simulación del nodo en el instante actual de simulación.
- *Vehiculo Buscar_vehiculo_influido (Vehiculo veh)* : Devuelve el vehículo que es influido por *veh*, o él mismo si no lo hubiera.
- *void Salida_movimiento_movimiento (unsigned int i, Vehiculo *vehiculo, unsigned int i_mov)* : Realiza la actualización de determinados parámetros cuando *vehiculo* sale de un movimiento para entrar en otro movimiento del mismo nodo.
- *bool Salida_movimiento (Vehiculo coche, unsigned int i, float tiempo_simulacion, bool *parado)* : Devuelve un valor lógico verdadero cuando *coche* sale de un movimiento para entrar en un tramo.
- *void Salida_movimiento_tramo (unsigned int i, Vehiculo *coche, float longitud, float tiempo_simulacion, unsigned int i_mov)* : Realiza las actualizaciones necesarias cuando *coche* pasa de un movimiento a un tramo.
- *void Borrar_nodo (void)* : Borra las listas correspondientes al nodo.
- *bool Existe_bloqueo (Movimiento mov, Vehiculo* veh, float* posicion)* : Devuelve un valor verdadero cuando existe un vehículo en una posición que bloquea el movimiento de *veh* debido a una incompatibilidad.
- *bool Obtener_vehiculo (Movimiento mov, float posicion_inc, Vehiculo* veh)* : Determina si un vehículo está bloqueando el paso por el movimiento.
- *bool Salida_nodo (unsigned int indice, float tiempo_simulacion)* : Determina si algún vehículo sale del nodo.

- *void Comprobar_salida_nodos (float tiempo_simulacion)* : Comprueba si algún vehículo sale del nodo en el instante actual de simulación.
- *void Actualiza_vehiculos_nodo (float tiempo_simulacion)* : Actualiza los datos de posición, velocidad y aceleración de los vehículos del nodo.

5.2.3.1.1.2 Clase "Micro_tramo"

Esta clase define todos los atributos y métodos relacionados con los tramos que conforman el escenario a simular.

```

Micro_tramo
+cod_tramo : unsigned int
+nodo_destino : unsigned int
+longitud : float
+indice_semaforo : unsigned int
+distancia_cambio_tramo : float
+num_vehiculos : unsigned int = 0
+num_carriles : unsigned int
+v_max : float
+velocidad_fin : float = 7
+regulado : bool
+funcion_generacion : int
+siguiente_generacion : int
+posicion_generacion : float = 0
+carril_generacion : unsigned int
+Lista_carriles : Lista_inf<Meso_carril>
+Lista_vehiculos : Lista_inf<Vehiculo>
+Lista_aparcamientos : Lista_inf<Aparcamiento>

+Micro_tramo()
+Salida_tramo(coche : Vehiculo, i : unsigned int, tiempo_simulacion : float) : bool
+Destruir_tramo()
+Borrar_tramo()
+Generar_carriles(num_carriles : unsigned int, senal : unsigned char *, porcentaje : unsigned char *)
+Salida_por_cambio_carril(coche : Vehiculo, carril : unsigned int, cod : unsigned int)
+Actualiza_ultimo_vehiculo_carril(carril : unsigned int, vehiculo : unsigned int)
+Add_vehiculos_carril(cod_carril : unsigned int, veh : unsigned int)
+Sub_vehiculos_carril(cod_carril : unsigned int, veh : unsigned int)
+Add_vehiculos(carril : unsigned int, numero : unsigned int)
+Sub_vehiculos(carril : unsigned int, numero : unsigned int)
+Actualiza_superavit_vehiculos_carril(cod_carril : unsigned int, veh : float)
+Distancia_cambio_carril(coche : Vehiculo*, carril_destino : unsigned int) : bool
+Buscar_vehiculo_influido(veh : Vehiculo) : Vehiculo
+Similar(At : float, tiempo_simulacion : float)
+Buscar_influencia(veh : Vehiculo, tiempo_simulacion : float, tipo_influencia : TIPO_INFLUENCIA*, tipo : unsigned int) : unsigned int
+Buscar_influencia_vehiculo(veh : Vehiculo, distancia_veh : float*, tipo : unsigned int) : unsigned int
+Actualiza_influencia_influido(codigo1 : unsigned int, codigo2 : unsigned int, num_carril : unsigned int, tipo : TIPO_INFLUENCIA)
+Cambio_carril(T : float, tiempo_simulacion : float)
+Cambio_carril(cod_carril : unsigned int, veh : Vehiculo*, tiempo_simulacion : float) : bool
+Aceptacion_gap(tipo : unsigned int, veh : Vehiculo, tiempo_simulacion : float) : unsigned int
+Aceptacion_gap(veh : Vehiculo, num_carril : unsigned int, tiempo_simulacion : float, distancia_trasera : float, distancia_delantera : float) : bool
+Aceptacion_gap(veh_n1 : Vehiculo, veh_n2 : Vehiculo, veh_n0 : Vehiculo, tipo : unsigned int, tiempo_simulacion : float) : unsigned int
+Aceptacion_gap(veh_n1 : Vehiculo, veh_n2 : Vehiculo, delantero : bool, tipo : unsigned int, tiempo_simulacion : float) : unsigned int
+Reparto_carril(cod_carril : unsigned int, num_veh : unsigned int)
+Mejora_condiciones(carril : unsigned int, veh : Vehiculo, At : float, ac_futura : float*, tiempo_simulacion : float) : bool
+Carril_destino_obligatorio(carril_origen : unsigned int) : unsigned int
+Carril_destino_obligatorio(carril_superavit : unsigned int, tramo_origen : unsigned int, tramo_destino : unsigned int...) : unsigned int
+Carril_destino_mejora(veh : Vehiculo, At : float, aceleracion : float*, tiempo_simulacion : float) : unsigned int
+Cambio_carril_mejora(At : float, tiempo_simulacion : float, coche : Vehiculo*)
+Aparcar(At : float, tiempo_simulacion : float)
+Aparcar_vehiculo(v : Vehiculo, tiempo_simulacion : float, At : float, pos_ap : float, carril_ap : unsigned int, tramo : unsigned int tramo)
+Eliminar_aparcando(v : Vehiculo*, o : Obstaculo, tiempo_simulacion : float)
+Buscar_aparcamiento(v : Vehiculo, pos_ap : float, carril_ap : unsigned int, tipo : TIPO_APARCAMIENTO*) : int
+Buscar_aparcamiento_util(v : Vehiculo, pos_ap : float, tipo : TIPO_APARCAMIENTO*) : int
+Buscar_aparcamiento_mer(v : Vehiculo, pos_ap : float, carril_ap : unsigned int) : int
+Cambio_carril_obligatorio(T : float, tiempo_simulacion : float, coche : Vehiculo*) : bool
+Salir_aparca(tiempo_simulacion : float, At : float)
+Introducir_nuevo_vehiculo(indice_parada : unsigned int, carril : unsigned int, posicion : float, velocidad : float, aceleracion : float, mov_destino : Codigo_movimiento...) : Vehiculo
+Generar_aparcas_mer(v : Vehiculo)
+Generacion_vehiculo(ruta : unsigned int, tiempo_simulacion : float)
+Estadisticas_cambio_carril(tiempo_simulacion : float, coche : Vehiculo, origen : unsigned int, tipo : unsigned int)
+Generar_cabecera_estadisticas_cambio_carril()
+Asigna_tramo_destino(destino : unsigned int) : unsigned int
+Generar_vehiculo(tiempo_simulacion : float)
+Salida_tramo(indice : unsigned int, tiempo_simulacion : float) : bool
+Comprobar_salida_tramos(tiempo_simulacion : float)
+Salida_tramo_aparcamiento(coche : Vehiculo, indice_ap : unsigned int, i : unsigned int, posicion_ap : float, tiempo_simulacion : float)
+Actualizacion(At : float, tiempo_simulacion : float, indice : unsigned int)
+Inicio_cambio_carril(veh : Vehiculo*, tiempo_simulacion : float)
+Aceptacion_gap_cortes(veh_n1 : Vehiculo, veh_n2 : Vehiculo, tiempo_simulacion : float) : bool
+Cambio_carril_salir_aparca(cod_carril : unsigned int, veh : Vehiculo*, tiempo_simulacion : float)
+Estadisticas_veh_car()
+Estadisticas_veh_sem(tiempo_simulacion : float)
    
```

Figura 5.14. Clase "Micro_tramo"

Atributos de la clase "Micro_tramo":

- *cod_tramo* : Código que identifica a cada tramo, además será el índice del tramo correspondiente en la lista de tramos.
- *nodo_destino* : Código del nodo en el que desemboca el tramo.
- *longitud* : Longitud en metros del tramo.
- *indice_semaforo* : Índice del semáforo que regula el tramo dentro de la lista de señales.
- *distancia_cambio_tramo* : Distancia al final del tramo a partir de la cual se considera que un vehículo cambia de tramo.
- *num_vehiculos* : Número de vehículos que circulan por el tramo.
- *num_carriles* : Número de carriles que conforman el tramo.
- *v_max* : Velocidad máxima a la que se permite circular por el tramo.
- *velocidad_fin* : Velocidad máxima permitida a la que se puede llegar al final del tramo.
- *regulado* : Indica si el tramo se encuentra regulado por un semáforo o no.
- *funcion_generacion* : Código de la función estadística empleada para la generación de vehículos en el tramo.
- *siguiente_generacion* : Número de vehículos que hay que generar en cada instante. La parte no entera se acumula para el siguiente período de tiempo.
- *posicion_generacion* : Posición del tramo en la que se debe generar el siguiente vehículo.
- *carril_generacion* : Carril en el que se desea introducir un nuevo vehículo.
- *Lista_carriles* : Lista que contiene los carriles del tramo.
- *Lista_vehiculos* : Lista que contiene los vehículos que circulan por el tramo.
- *Lista_aparcamientos* : Lista en la que figuran los aparcamientos existentes en el tramo.

Métodos de la clase "Micro_tramo":

- *void Micro_tramo (void)* : Crea las listas necesarias para un tramo.
- *bool Salida_tramo (Vehiculo coche, unsigned int i, float tiempo_simulacion)* : Comprueba si *coche* sale del tramo en el instante actual de simulación.
- *void Destruir_tramo (void)* : Destruye las listas, con sus elementos, del tramo.

- *void Borrar_tramo (void)* : Borra las listas correspondientes al tramo.
- *void Generar_carriles (unsigned int num_carriles, unsigned char* senal, unsigned char* porcentaje)* : Crea los carriles que debe haber en el tramo y los inicializa.
- *void Salida_por_cambio_carril (Vehiculo coche, unsigned int carril, unsigned int cod)* : Actualiza el último vehículo que circula por el carril en el caso de que *coche* haya cambiado de carril y fuera antes el último de su carril.
- *void Actualiza_ultimo_vehiculo_carril (unsigned int carril, unsigned int vehiculo)* : Procedimiento que marca a *vehiculo* como el último de *carril*.
- *void Add_vehiculos_carril (unsigned int cod_carril, unsigned int veh)* : Añade al superávit de vehículos del carril especificado el número que se indica.
- *void Sub_vehiculos_carril (unsigned int cod_carril, unsigned int veh)* : Resta del superávit de vehículos del carril especificado el número que se indica.
- *void Add_vehiculos (unsigned int carril, unsigned int numero)* : Añade al carril especificado el número de vehículos que se indican.
- *void Sub_vehiculos (unsigned int carril, unsigned int numero)* : Resta del carril especificado el número de vehículos que se indican.
- *void Actualiza_superavit_vehiculos_carril (unsigned int cod_carril, float veh)* : Actualiza el superávit de vehículos del carril *cod_carril* al valor *veh*.
- *bool Distancia_cambio_carril (Vehiculo* coche, unsigned int carril_destino)* : Comprueba si *coche* se encuentra a la distancia mínima del final del tramo para realizar un cambio de carril obligatorio. Debe tener en cuenta el carril hacia el que se quiere desplazar.
- *Vehiculo Buscar_vehiculo_influido (Vehiculo veh)* : Devuelve el vehículo que se ve influido por *veh*, o el propio *veh* si no lo hubiera.
- *void Simular (float At, float tiempo_simulacion)* : Función que realiza la simulación del tramo en el instante actual.
- *unsigned int Buscar_influencia (Vehiculo veh, float tiempo_simulacion, TIPO_INFLUENCIA *tipo_influencia, unsigned int tipo)* : Devuelve el código del elemento que influye a *veh* y de qué tipo es.
- *unsigned int Buscar_influencia_vehiculo (Vehiculo veh, float* distancia_veh, unsigned int tipo)* : Devuelve el código del vehículo que influye a *veh*, o el del propio *veh* si no lo hubiera.
- *unsigned int Buscar_influencia_obstaculo (Vehiculo veh, float* distancia_obs, float tiempo_simulacion, unsigned int tipo)* : Devuelve el código del obstáculo que influye a *veh*, o 0 si no lo hubiera.
- *void Actualiza_influencia_influido (unsigned int codigo1, unsigned int codigo2, unsigned int num_carril, TIPO_INFLUENCIA tipo)* : Actualiza la

influencia del vehículo que se veía influido por el vehículo *codigo2* a *codigo1*, ya que *codigo2* va a cambiar de carril y *codigo1* era su influencia.

- *void Cambio_carril (float T, float tiempo_simulacion)* : Función general que se encarga del procedimiento de cambio de carril de los vehículos.
- *bool Cambio_carril (unsigned int cod_carril, Vehiculo* veh, float tiempo_simulacion)* : Procedimiento que realiza los cambios de carril necesarios para que *veh* se coloque en el carril adecuado para aparcar.
- *unsigned int Aceptacion_gap (unsigned int tipo, Vehiculo veh, float tiempo_simulacion)* : Devuelve 0 si no se acepta el gap, 1 si se acepta el gap normal y 2 si se acepta el gap de cortesía para que *veh* realice un cambio de carril.
- *bool Aceptacion_gap (Vehiculo veh, unsigned int num_carril, float tiempo_simulacion, float distancia_trasera, float distancia_delantera)* : Comprueba si se cumple el gap necesario para que *veh* realice un cambio de carril para aparcar.
- *unsigned int Aceptacion_gap (Vehiculo veh_n1, Vehiculo veh_n2, Vehiculo veh_n0, unsigned int tipo, float tiempo_simulacion)* : Comprueba que se cumplan tanto el gap delantero como el trasero.
- *unsigned int Aceptacion_gap (Vehiculo veh_n1, Vehiculo veh_n2, bool delantero, unsigned int tipo, float tiempo_simulacion)* : Verifica el gap delantero o el trasero en función del parámetro *delantero*.
- *void Reparto_carril (unsigned int cod_carril, unsigned int num_veh)* : Obtiene el número de vehículos que deben existir en cada carril, indicando el déficit o superávit que exista.
- *bool Mejora_condiciones (unsigned int carril, Vehiculo veh, float At, float* ac_futura, float tiempo_simulacion)* : Determina si *veh* mejoraría sus condiciones de circulación si cambiara de carril.
- *unsigned int Carril_destino_obligatorio (unsigned int carril_origen)* : Determina el carril de destino de un cambio obligatorio para mantener el porcentaje de vehículos en cada carril.
- *unsigned int Carril_destino_obligatorio (unsigned int carril_superavit, unsigned int carril_origen, unsigned int tramo_origen, unsigned int tramo_destino, unsigned int* carril_final)* : Determina el carril de destino en un cambio obligatorio para que un vehículo pueda llegar a su destino.
- *unsigned int Carril_destino_mejora (Vehiculo veh, float At, float* aceleracion, float tiempo_simulacion)* : Devuelve el número del carril hacia el que se tiene que desplazar *veh* para realizar un cambio por mejora.
- *void Cambio_carril_mejora (float At, float tiempo_simulacion, Vehiculo* coche)* : Comprueba si *coche* debe realizar un cambio de carril por mejora, y lo realiza si así fuera.
- *void Aparcar (float At, float tiempo_simulacion)* : Comprueba si hay que realizar algún aparcamiento en el instante de simulación actual.

- *void Aparcar_vehiculo (Vehiculo v, float tiempo_simulacion, float At, float pos_ap, unsigned int carril_ap, unsigned int tramo) : Realiza la maniobra de aparcamiento del vehículo v.*
- *void Eliminar_aparcando (Vehiculo* v, Obstaculo o, float tiempo_simulacion) : Elimina el vehículo v de su carril y de la lista de obstáculos, al haber realizado ya el aparcamiento.*
- *int Buscar_aparcamiento (Vehiculo v, float pos_ap, unsigned int carril_ap, TIPO_APARCAMIENTO *tipo) : Busca un lugar donde v pueda aparcarse.*
- *int Buscar_aparcamiento_util (Vehiculo v, float pos_ap, TIPO_APARCAMIENTO* tipo) : Busca aparcamiento para un utilitario.*
- *int Buscar_aparcamiento_mer (Vehiculo v, float pos_ap, unsigned int carril_ap) : Busca aparcamiento para un vehículo de mercancías.*
- *bool Cambio_carril_obligatorio (float T, float tiempo_simulacion, Vehiculo* coche) : Comprueba, y lo realiza en su caso, si coche debe hacer un cambio de carril obligatorio.*
- *void Salir_aparca (float tiempo_simulacion, float At) : Realiza las salidas de los aparcamientos de los vehículos en tiempo_simulacion.*
- *Vehiculo Introducir_nuevo_vehiculo (unsigned int indice_parada, unsigned int carril, float posicion, float velocidad, float aceleracion,Codigo_movimiento mov_destino, bool Obstaculo, float Longitud_entidad, float longitud_vehiculo, float deceleracion_max, float aceleracion_max, float velocidad_max, float distancia_influencia, unsigned int tipo_conductor, TIPO_VEHICULO tipo_vehiculo, float distancia_seguridad, bool activo, bool aparca_asignado, unsigned int tramo_aparca, float t_cero_aparcado, float tiempo_simulacion, bool destruido) : Crea un nuevo vehículo en el escenario con los parámetros especificados.*
- *void Generar_aparcas_mer (Vehiculo v) : Genera los aparcamientos para el vehículo de mercancías v.*
- *void Generacion_vehiculo (unsigned int ruta, float tiempo_simulacion) : Crea los parámetros necesarios para generar un nuevo vehículo, y procede a su generación.*
- *void Estadisticas_cambio_carril (float tiempo_simulacion, Vehiculo coche, unsigned int origen, unsigned int tipo) : Genera las estadísticas correspondientes a los cambios de carril que se producen.*
- *void Generar_cabecera_estadisticas_cambio_carril (void) : Escribe la cabecera del fichero de texto de las estadísticas de cambio de carril.*
- *unsigned int Asigna_tramo_destino (unsigned int destino) : Asigna el tramo hacia el que debe dirigirse un vehículo cuando es generado.*
- *void Generar_vehiculo (float tiempo_simulacion) : Función destinada a realizar la generación de vehículos en tiempo_simulacion.*

- *bool Salida_tramo (unsigned int indice, float tiempo_simulacion) :* Comprueba si algún vehículo sale del tramo en ese instante.
- *void Comprobar_salida_tramos (float tiempo_simulacion) :* Es la función encargada de llamar a la anterior.
- *void Salida_tramo_aparcamiento (Vehiculo coche, unsigned int indice_ap, unsigned int i, float posicion_ap, float tiempo_simulacion) :* Realiza las actualizaciones oportunas debido a que *coche* abandona el tramo para aparcar.
- *void Actualizacion (float At, float tiempo_simulacion, unsigned int indice) :* Realiza la actualización de los parámetros de todos los vehículos al final de cada instante de simulación.
- *void Inicio_cambio_carril (Vehiculo* veh, float tiempo_simulacion) :* Realiza las operaciones necesarias para el comienzo de una maniobra de cambio de carril.
- *bool Aceptacion_gap_cortes (Vehiculo veh_n1, Vehiculo veh_n2, float tiempo_simulacion) :* Comprueba si se cumple el denominado gap de cortesía.
- *void Cambio_carril_salir_aparca (unsigned int cod_carril, Vehiculo* veh, float tiempo_simulacion) :* Maniobra de incorporación a un carril de *veh* tras salir de un aparcamiento.
- *void Estadisticas_veh_car (void) :* Calcula el número de vehículos que circulan por cada carril del tramo.
- *void Estadisticas_veh_sem (float tiempo_simulacion) :* Calcula el número de vehículos detenidos en el semáforo que regula el tramo.

5.2.3.1.1.3 Clase "Giro"

Los giros permiten la circulación entre dos tramos a través de un nodo. Esta clase será la encargada de caracterizar dichos giros.

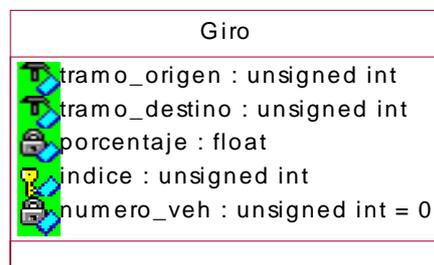


Figura 5.15. Clase "Giro"

Atributos de la clase "Giro":

- *tramo_origen* : Código del tramo del que proceden los vehículos que realizan el giro.

- *tramo_destino* : Código del tramo al que se dirigen los vehículos que realizan el giro.
- *porcentaje* : Porcentaje del número total de vehículos que circulan por el tramo origen que eligen el giro para seguir su itinerario.
- *indice* : Índice del giro en la lista de giros del nodo.
- *numero_veh* : Número de vehículos que circulan por el giro.

5.2.3.1.1.4 Clase "Movimiento"

Los movimientos son desagregaciones de los giros. Permiten simular con mucho mayor nivel de detalle cuál es el movimiento de los vehículos en el interior de las intersecciones.

En la siguiente figura se puede observar el interior de un nodo en el que se han señalado algunos de los giros que se pueden realizar. Además se han desagregado dichos giros en movimientos. Se aprecian los cuatro tipos distintos de movimientos que existen.

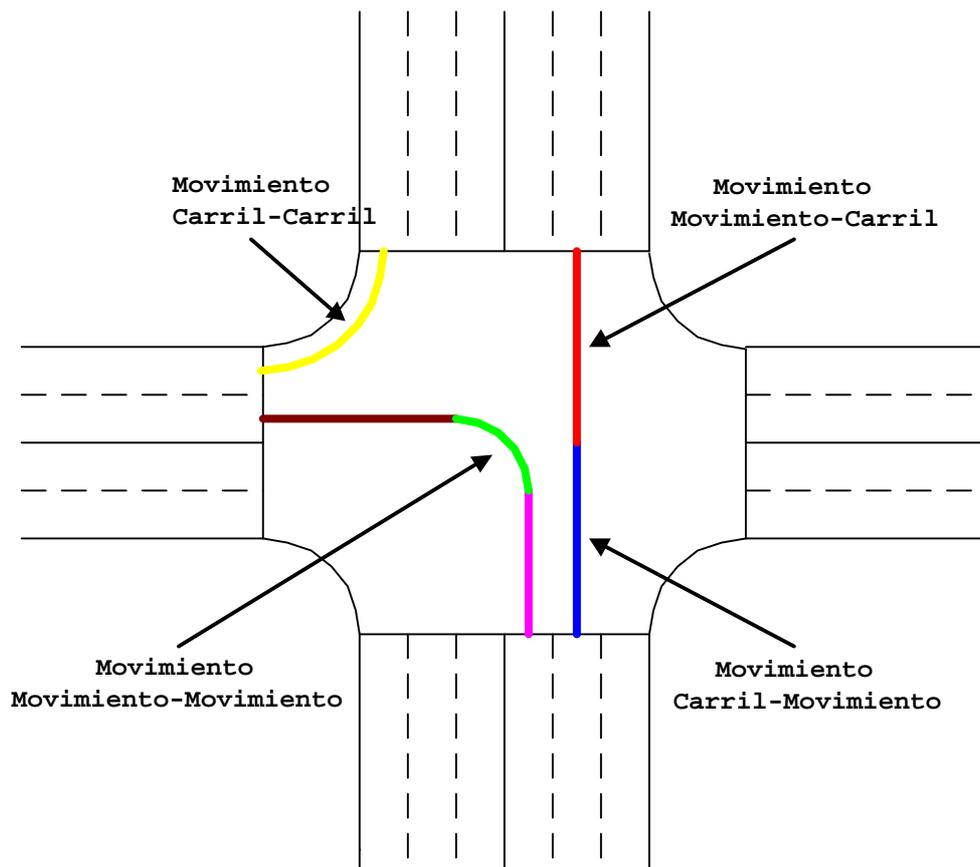


Figura 5.16. Representación de movimientos



Figura 5.17. Clase "Movimiento"

Atributos de la clase "Movimiento":

- *codigo* : Diferencia un movimiento del resto de movimientos.
- *tipo* : Indica el tipo de movimiento de que se trata en función de que el origen y el final del movimiento sean el carril de un tramo, o bien, otro movimiento. De esta forma, se tienen los siguientes tipos: CARRIL-CARRIL, CARRIL-MOVIMIENTO, MOVIMIENTO-MOVIMIENTO o MOVIMIENTO-CARRIL.
- *longitud* : Longitud en metros del movimiento.
- *origen_tramo* : Código del tramo origen, en el caso de que el origen sea un carril de un tramo.
- *destino_tramo* : Código del tramo destino, en el caso de que el origen sea un carril de un tramo.
- *origen* : Código del origen, ya sea el de un carril o el de otro movimiento del mismo nodo.
- *destino* : Código del destino, ya sea el de un carril o el de otro movimiento del mismo nodo.
- *senal* : Señal que regula el desplazamiento de los vehículos a través del movimiento.
- *ultimo_vehiculo* : Código del último vehículo que circula a través del movimiento.
- *indice_in* : Índice del código de incompatibilidad con otro movimiento.
- *regulado* : Indica si el movimiento se encuentra regulado por alguna señal.

5.2.3.1.1.5 Clase "Codigo_movimiento"

Esta clase permite identificar unívocamente los distintos movimientos existentes en el escenario de simulación.

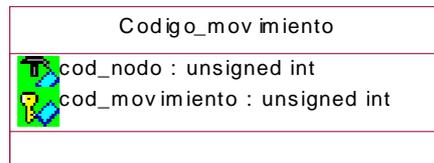


Figura 5.18. Clase "Codigo_movimiento"

Atributos de la clase "Codigo_movimiento":

- *cod_nodo* : Código del nodo en el que se encuentra situado el movimiento.
- *cod_movimiento* : Código del movimiento para diferenciarlo del resto.

5.2.3.1.1.6 Clase "Indice_movimiento"

Esta clase permite caracterizar los movimientos del interior de un nodo mediante su posición en la lista de índices de movimientos.

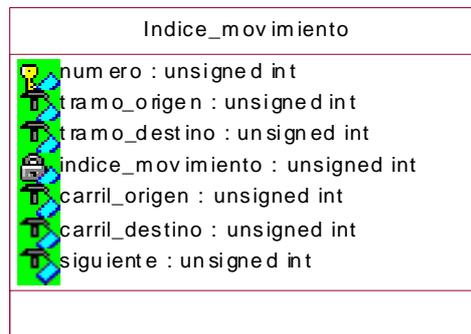


Figura 5.19. Clase "Índice_movimiento"

Atributos de la clase "Indice_movimiento":

- *numero* : Número representativo del movimiento.
- *tramo_origen* : Tramo de origen del movimiento.
- *tramo_destino* : Tramo destino del movimiento.
- *indice_movimiento* : Índice del movimiento en la lista de movimientos del nodo.
- *carril_origen* : Carril origen del movimiento.
- *carril_destino* : Carril destino del movimiento.
- *siguiente* : Índice del siguiente movimiento.

5.2.3.1.1.7 Clase "In_movimiento"

Con esta clase se pretenden representar las posibles incompatibilidades existentes entre los movimientos de un mismo nodo. Estas incompatibilidades se refieren a la imposibilidad de que dos vehículos diferentes se encuentren realizando en el mismo instante de tiempo dos movimientos que sean considerados incompatibles entre sí. A continuación se muestran los atributos de esta clase:

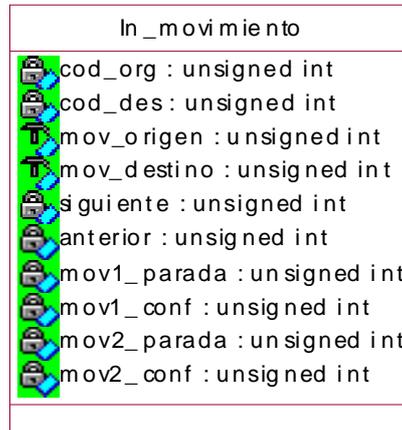


Figura 5.20. Clase "In_movimiento"

Atributos de la clase "In_movimiento":

- *cod_org* : Índice del movimiento que actúa como origen de la incompatibilidad en la lista de movimientos del nodo.
- *cod_des* : Índice del movimiento que actúa como destino de la incompatibilidad en la lista de movimientos del nodo.
- *mov_origen* : Código del movimiento origen de la incompatibilidad.
- *mov_destino* : Código del movimiento destino de la incompatibilidad.
- *siguiente* : Índice de la siguiente incompatibilidad del movimiento origen.
- *anterior* : Índice de la anterior incompatibilidad del movimiento origen.
- *mov1_parada* : Número de metros a partir del comienzo del movimiento donde debe parar el vehículo que va por el movimiento origen, en caso de que haya incompatibilidad.
- *mov1_conf* : Posición del movimiento origen a partir de donde comienza el conflicto.
- *mov2_parada* : Número de metros a partir del comienzo del movimiento donde debe parar el vehículo que va por el movimiento destino, en caso de que haya incompatibilidad.
- *mov2_conf* : Posición del movimiento destino a partir de donde comienza el conflicto.

5.2.3.1.1.8 Clase "Meso_carril"

Los tramos que conforman el viario se encuentran divididos en distintos carriles, por los que circulan los vehículos. Veamos cómo se caracterizan dichos carriles:

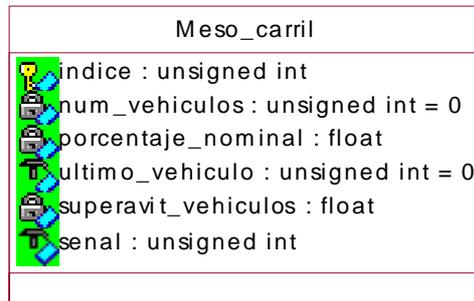


Figura 5.21. Clase Meso_carril

Atributos de la clase "Meso_carril":

- *indice* : Identifica al carril en la lista de carriles del tramo correspondiente.
- *num_vehiculos* : Número de vehículos que circulan por el carril.
- *porcentaje_nominal* : Porcentaje de vehículos del total que circulan por el tramo, que deben hacerlo por este carril.
- *ultimo_vehiculo* : Código del último vehículo que circula por el carril.
- *superavit_vehiculos* : Indica el exceso de vehículos que hay en el carril respecto a los que debería haber para cumplir los porcentajes deseados.
- *senal* : Código de la señal que regula la circulación en el carril.

5.2.3.1.1.9 Clase "Codigo_carril"

Esta clase se emplea para identificar un carril específico del resto de carriles.

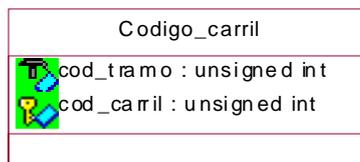


Figura 5.22. Clase "Codigo_carril"

Atributos de la clase "Codigo_carril":

- *cod_tramo* : Identifica el tramo en el que se encuentra el carril.
- *cod_carril* : Identifica al carril del resto de carriles del mismo tramo.

5.2.3.1.1.10 Clase "Aparcamiento"

Esta clase contiene los objetos que representan los lugares del viario en los que es posible realizar un aparcamiento.

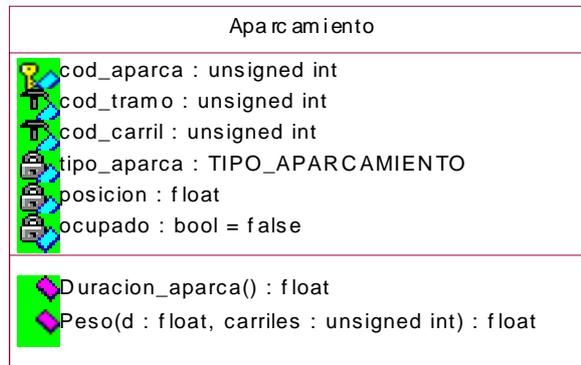


Figura 5.23. Clase "Aparcamiento"

Atributos de la clase "Aparcamiento":

- *cod_aparca* : Código del aparcamiento.
- *cod_tramo* : Tramo en el que se encuentra el aparcamiento.
- *cod_carril* : Carril más próximo al aparcamiento.
- *tipo_aparca* : Indica el tipo de aparcamiento de que se trata, pudiendo ser ACERA, BLOQUEO_CARRIL, CARGA_DESCARGA, FILA, BATERIA o APARCAMIENTO_MASIVO.
- *posicion* : Posición en el tramo en la que se encuentra el aparcamiento.
- *ocupado* : Indica si el aparcamiento se encuentra libre u ocupado.

Métodos de la clase "Aparcamiento"

- *float Duracion_aparca (void)* : Devuelve la duración del aparcamiento en función del tipo del que se trate.
- *float Peso (float d, unsigned int carriles)* : Devuelve el peso de un aparcamiento en relación con un vehículo, es decir, cuánto de atractivo le resulta el aparcamiento al vehículo. Para ello se tiene en cuenta la distancia al aparcamiento, el tipo de aparcamiento y el número de carriles del tramo.

5.2.3.1.1.11 Clase "Aparcamiento_masivo"

Esta clase se emplea para caracterizar aquellos lugares de aparcamiento con capacidad para más de un vehículo.

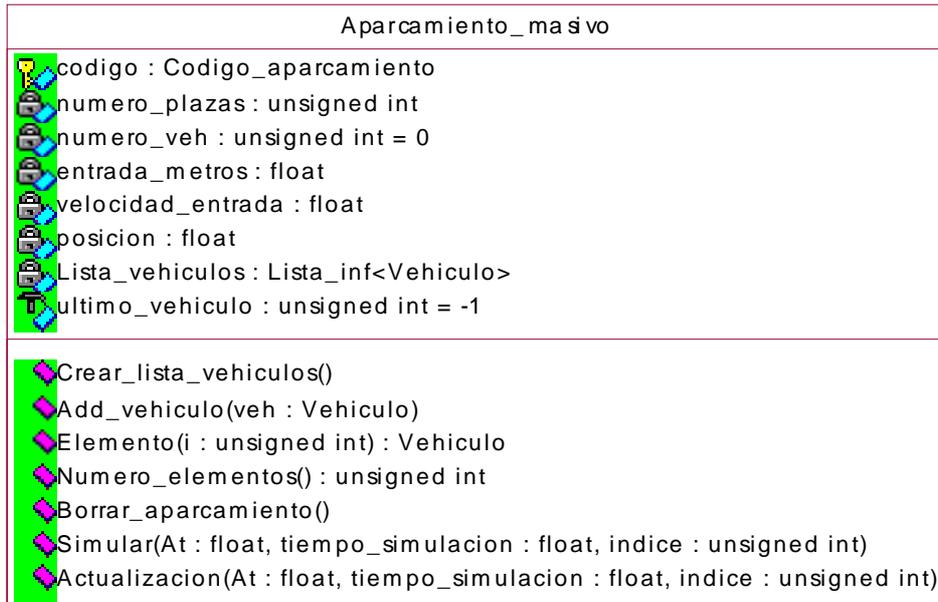


Figura 5.24. Clase "Aparcamiento_masivo"

Atributos de la clase "Aparcamiento_masivo":

- *codigo* : Sirve para diferenciar a un aparcamiento masivo del resto.
- *numero_plazas* : Número de plazas de aparcamiento de las que dispone.
- *numero_veh* : Indica el número de vehículos que hay aparcados en el aparcamiento masivo.
- *entrada_metros* : Longitud de la rampa de entrada en metros.
- *velocidad_entrada* : Velocidad máxima a la que se puede entrar en el aparcamiento masivo.
- *posicion* : Posición en la que se encuentra en el tramo.
- *Lista_vehiculos* : Contiene todos los vehículos que se encuentran en el aparcamiento.
- *ultimo_vehiculo* : Código del último vehículo que entró en el aparcamiento.

Métodos de la clase "Aparcamiento masivo":

- *void Crear_lista_vehiculos ()* : Crea la lista de vehículos del aparcamiento.
- *void Add_vehiculo (Vehiculo veh)* : Añade *veh* a la lista de vehículos del aparcamiento.
- *Vehiculo Elemento (unsigned int i)* : Devuelve el elemento *i* de la lista de vehículos.

- *unsigned int Numero_elementos (void)* : Devuelve el número de elementos de la lista de vehículos.
- *void Borrar_aparcamiento (void)* : Borra la lista de vehículos del aparcamiento.
- *void Simular (float At, float tiempo_simulacion, unsigned int indice)* : Se simulan todos los vehículos del aparcamiento masivo, obteniendo sus velocidades, posiciones y aceleraciones de ese instante.
- *void Actualizacion (float At, float tiempo_simulacion, unsigned int indice)* : Realiza la actualización de los valores de los vehículos del aparcamiento correspondientes a ese instante de simulación.

5.2.3.1.1.12 Clase "Codigo_aparcamiento"

Esta clase sirve para identificar a cada uno de los aparcamientos masivos.

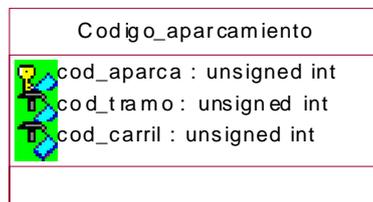


Figura 5.25. Clase "Codigo_aparcamiento"

Atributos de la clase "Codigo_aparcamiento":

- *cod_aparca* : Código único del aparcamiento masivo.
- *cod_tramo* : Tramo en el que se encuentra el aparcamiento.
- *cod_carril* : Carril por el que se accede al aparcamiento masivo.

5.2.3.1.2 Diagrama de clases del paquete "Red"

En el siguiente diagrama se muestran las relaciones entre las distintas clases que conforman este paquete.

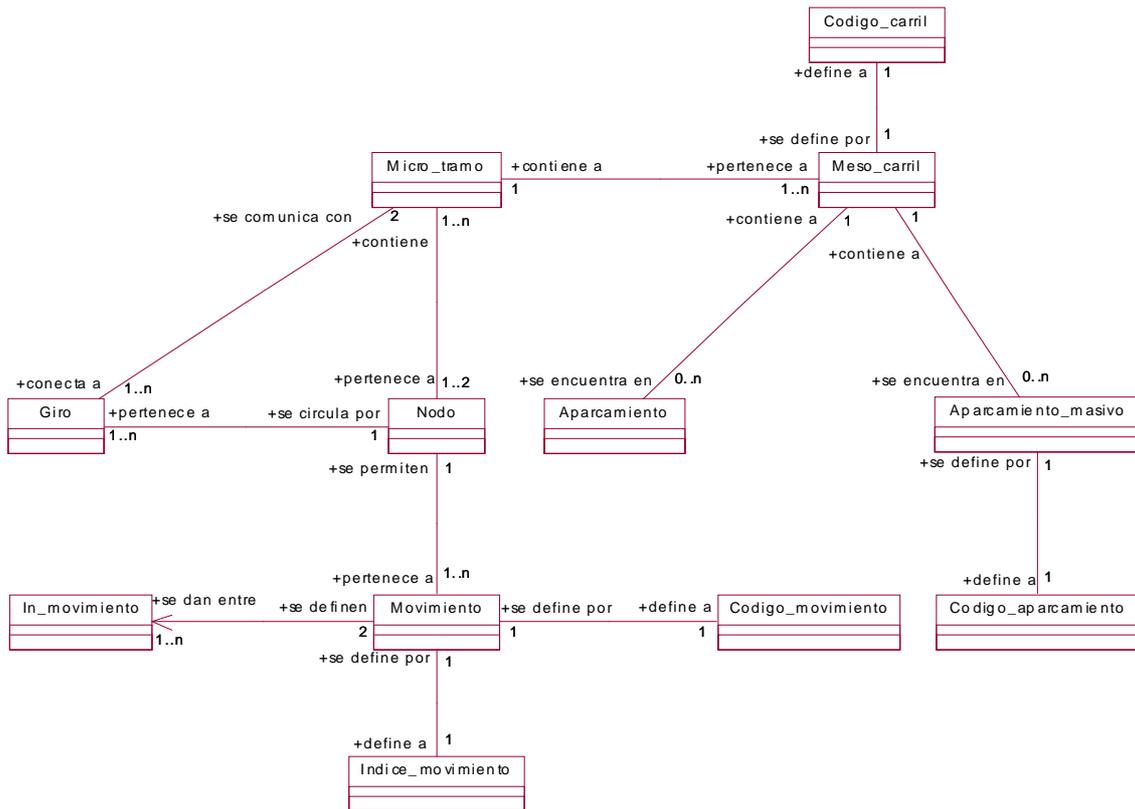


Figura 5.26. Diagrama de clases "Red"

5.2.3.2 Paquete de datos "Grupo Semafórico"

Este paquete de datos contiene la información concerniente al funcionamiento de los dispositivos reguladores de tráfico (*semáforos*). A continuación se introducen conceptos que permitirán una mejor comprensión de estos datos, para continuar con una explicación exhaustiva de cada una de las clases pertenecientes a este paquete.

Los semáforos son dispositivos físicos que restringen o conceden derecho de paso a un conjunto de vehículos y funcionan de acuerdo a una señal que regula el comportamiento (*estado*) del mismo. Los semáforos que presentan el mismo comportamiento se agrupan en *grupos semafóricos*.

Una *señal* describe el comportamiento de uno o varios semáforos a lo largo del tiempo. El comportamiento de la señal es cíclico, es decir, se repite en cada cierto intervalo de tiempo (*ciclo*). Las señales se agrupan para formar las cajas de señales o *estructuras* que contienen todas las señales definidas que regulan el comportamiento de los dispositivos en el nodo. Un nodo puede contener varias estructuras dependiendo del efecto que se desea producir en la regulación del tráfico.

Todas las señales de una estructura se dividen en etapas o *fases* que muestran el estado de la señal en un intervalo de tiempo. Asimismo, cada una de las fases suele subdividirse en pequeños intervalos de tiempo que se denominan *tiempos intermedios*. Los tiempos intermedios son tiempos fijos que garantizan la no ocurrencia simultánea

de cambios de rojo a verde para grupos diferentes. Su misión es principalmente la de establecer breves intervalos de tiempo de seguridad, con el objetivo de evitar conflictos entre los primeros vehículos que atraviesan la línea de detención asociada a la señal para un grupo que acaba de recibir derecho de paso y los últimos vehículos correspondientes a los que se les ha suprimido este derecho. Con frecuencia, los tiempos intermedios contienen el tiempo en ámbar asignado a los diferentes grupos. Cada etapa queda entonces definida por un conjunto de tiempos intermedios, de longitud fija y en igual número para todos los grupos, y un tiempo de longitud variable asignado a la fase.

El estado de la señal viene determinado por el color que puede tomar la fase o bien el tiempo de intermedio. Los posibles estados de la señal son:

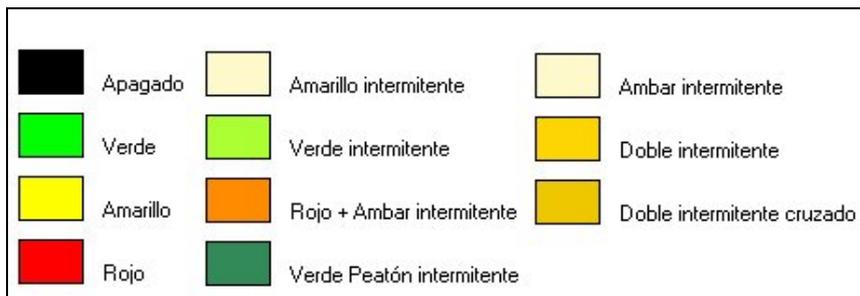


Figura 5.27. Posibles estados de la señal semafórica

A modo de ejemplo, a continuación se muestra una estructura formada por ocho señales. Cada una de dichas señales se encuentra dividida en tres fases, las dos primeras de ellas disponen de cuatro tiempos intermedios, mientras que la última fase tiene cinco tiempos intermedios.

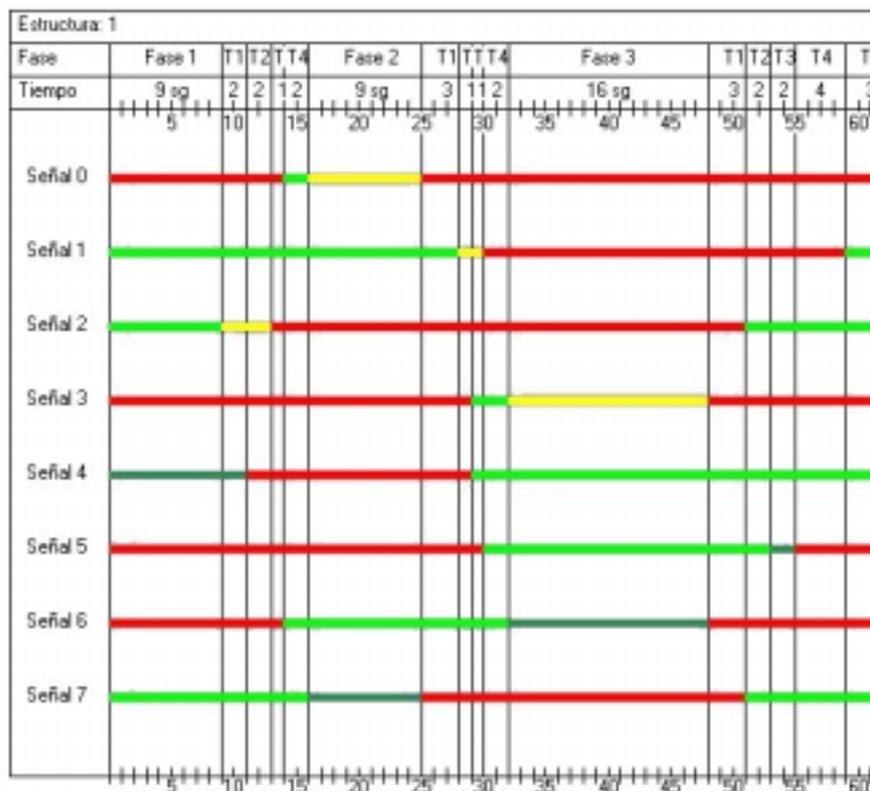


Figura 5.28. Ejemplo de Caja Semafórica

5.2.3.2.1 Descripción de las clases del paquete "Grupo Semafórico"

5.2.3.2.1.1 Clase "Estructura"

Esta clase agrupa el conjunto de grupos semafóricos que pueden imponerse en un nodo, el cual puede contener varias estructuras distintas que pueden aplicarse en cualquier instante. Incluso dos estructuras asignadas a un mismo nodo pueden tener ciclos diferentes, un número de fases distintas y estados desiguales.

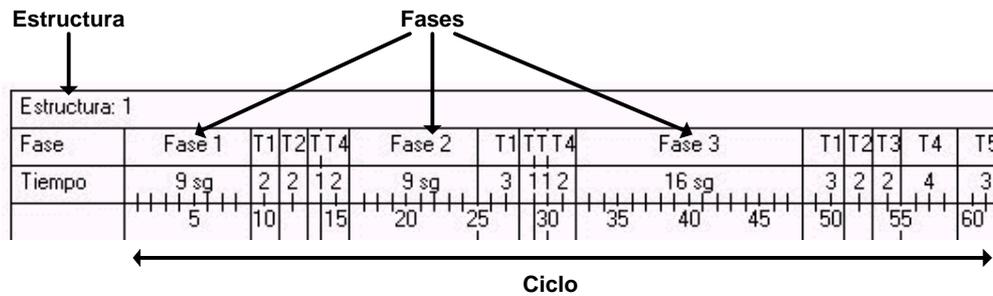


Figura 5.29. Estructura, fases y ciclo

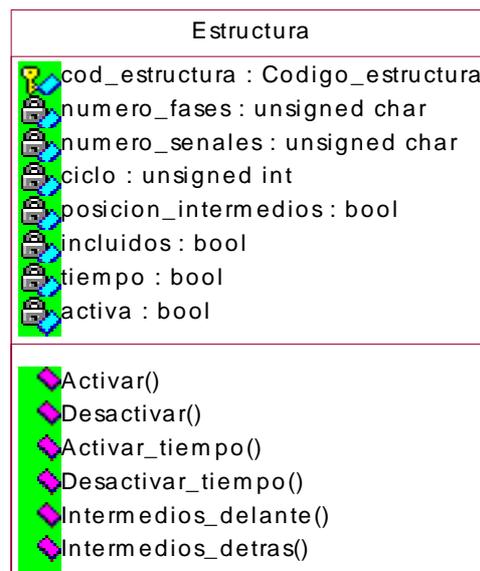


Figura 5.30. Clase "Estructura"

Atributos de la clase "Estructura":

- *cod_estructura* : Identifica unívocamente a la estructura.
- *numero_fases* : Número de etapas o fases de las que consta la estructura.
- *numero_senales* : Número de señales que se han definido en la estructura.
- *ciclo* : Tiempo en segundos que dura el ciclo de la estructura.
- *posicion_intermedios* : Indica si los tiempos intermedios se producen antes o después de cada fase.

- *incluidos* : Indica si los tiempos intermedios están incluidos en las señales.
- *tiempo* : Indica si los tiempos se expresan en porcentajes o en segundos.
- *activa* : Indica si la estructura está actualmente regulando el tráfico.

Métodos de la clase "Estructura":

- *void Activar (void)* : Activa la estructura.
- *void Desactivar (void)* : Desactiva la estructura.
- *void Activar_tiempo (void)* : Hace que los tiempos se expresen en segundos.
- *void Desactivar_tiempo (void)* : Hace que los tiempos se expresen en porcentajes.
- *void Intermedios_delante (void)* : Hace que los tiempos intermedios se produzcan antes que las fases.
- *void Intermedios_detras (void)* : Hace que los tiempos intermedios se produzcan después de las fases.

5.2.3.2.1.2 Clase "Codigo_estructura"

Esta clase se emplea para identificar unívocamente a una de las estructuras analizadas anteriormente.

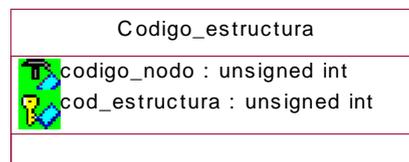


Figura 5.31. Clase "Codigo_estructura"

Atributos de la clase "Codigo_estructura":

- *codigo_nodo* : Nodo en el que se encuentra la estructura.
- *cod_estructura* : Código de la estructura en ese nodo.

5.2.3.2.1.3 Clase "Codigo_senal"

Representa los estados que tomarán uno o varios semáforos durante todo el ciclo de la estructura. Estos códigos tomarán un valor único que permita identificar las distintas señales.

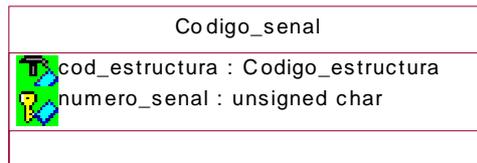


Figura 5.32. Clase "Codigo_senal"

Atributos de la clase "Codigo_senal":

- *cod_estructura* : Código de la estructura a la que pertenece la señal.
- *numero_senal* : Número propio de la señal en el interior de la estructura que la contiene.

5.2.3.2.1.4 Clase "Fase"

Las fases o etapas muestran el estado que tienen cada una de las señales durante un intervalo de tiempo concreto.

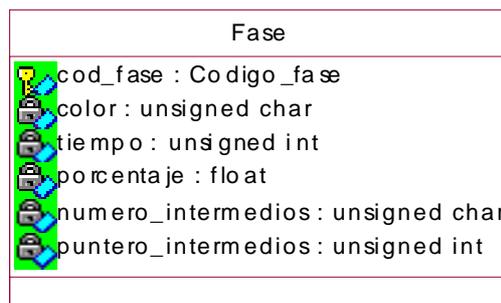


Figura 5.33. Clase "Fase"

Atributos de la clase "Fase":

- *cod_fase* : Identifica unívocamente cada fase.
- *color* : Estado que tiene la señal en la fase.
- *tiempo* : Duración de la fase en segundos.
- *porcentaje* : Tanto por ciento de duración de la fase respecto al ciclo total.
- *numero_intermedios* : Número de tiempos intermedios de los que dispone la fase.
- *puntero_intermedios* : Índice de la lista de intermedios en el que comienzan los intermedios correspondientes a la fase.

5.2.3.2.1.5 Clase "Codigo_fase"

Este código se utiliza para diferenciar a una fase concreta del resto de fases.

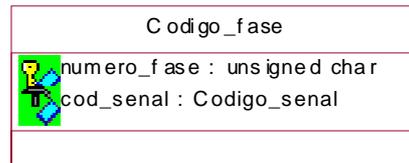


Figura 5.34. Clase "Codigo_fase"

Atributos de la clase "Codigo_fase":

- *numero_fase* : Número que identifica a la fase dentro de la señal correspondiente.
- *cod_senal* : Código de la señal a la que pertenece la fase.

5.2.3.2.1.6 Clase "Intermedio"

Los tiempos intermedios son tiempos fijos que garantizan la no ocurrencia simultánea de cambios de rojo a verde para grupos diferentes.

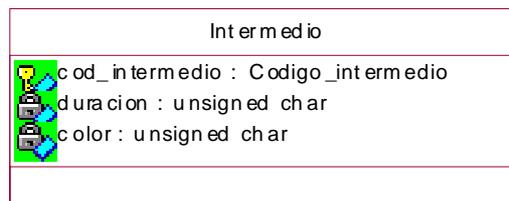


Figura 5.35. Clase "Intermedio"

Atributos de la clase "Intermedio":

- *cod_intermedio* : Código que identifica al intermedio.
- *duracion* : Tiempo en segundos que dura el intermedio.
- *color* : Estado asignado al tiempo intermedio.

5.2.3.2.1.7 Clase "Codigo_intermedio"

Este código permite identificar unívocamente a los tiempos intermedios.

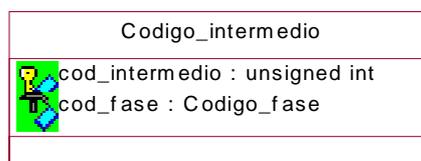


Figura 5.36. Clase "Codigo_intermedio"

Atributos de la clase "Codigo_intermedio":

- *cod_intermedio* : Identifica los distintos tiempos intermedios pertenecientes a una misma fase.
- *cod_fase* : Identifica la fase a la que pertenece el intermedio.

5.2.3.2.1.8 Clase "Senal_semaforica"

Un semáforo es un dispositivo físico que regula el desplazamiento de vehículos en el interior de una intersección.



Figura 5.37. Clase "Senal_semaforica"

Atributos de la clase "Senal_semaforica":

- *e* : Estructura a la que pertenece la señal semafórica.
- *desfase* : Tiempo de desfase entre los distintos semáforos.
- *lista_fases* : Lista de las fases que pertenecen a la señal semafórica.
- *lista_intermedios* : Lista de los tiempos intermedios que pertenecen a la señal semafórica.

Métodos de la clase "Senal_semaforica":

- *void Inicializacion (void)* : Crea las listas correspondientes a la señal semafórica.
- *void Destruir_lista (void)* : Destruye las listas correspondientes a la señal semafórica.
- *bool Procesar_cadena_importacion (char* cadena, int* descripcion, unsigned int numero, Lista_inf<Intermedio>* lista_int, Lista_inf<Fase>* lista_fas, unsigned int* numero_estructuras, unsigned int cod_nodo, FILE* F1)* : Procesa una cadena importada de un fichero en la que se encuentra información sobre la señal semafórica.
- *bool Importar_senal (char* nombre, unsigned int* numero_elementos, unsigned int elemento, unsigned int cod_nodo)* : En esta función se van cogiendo todas las líneas de un fichero para ir las pasando al método anterior para que las procese.

- *unsigned char Estado_senal (unsigned int senal, float instante_simulacion)*
Devuelve el estado de la señal semafórica teniendo en cuenta el instante de simulación y la señal de la que se desea ver el estado que tiene.

5.2.3.2 Diagrama de clases del paquete "Grupo Semafórico"

El correspondiente diagrama se muestra en la siguiente figura:

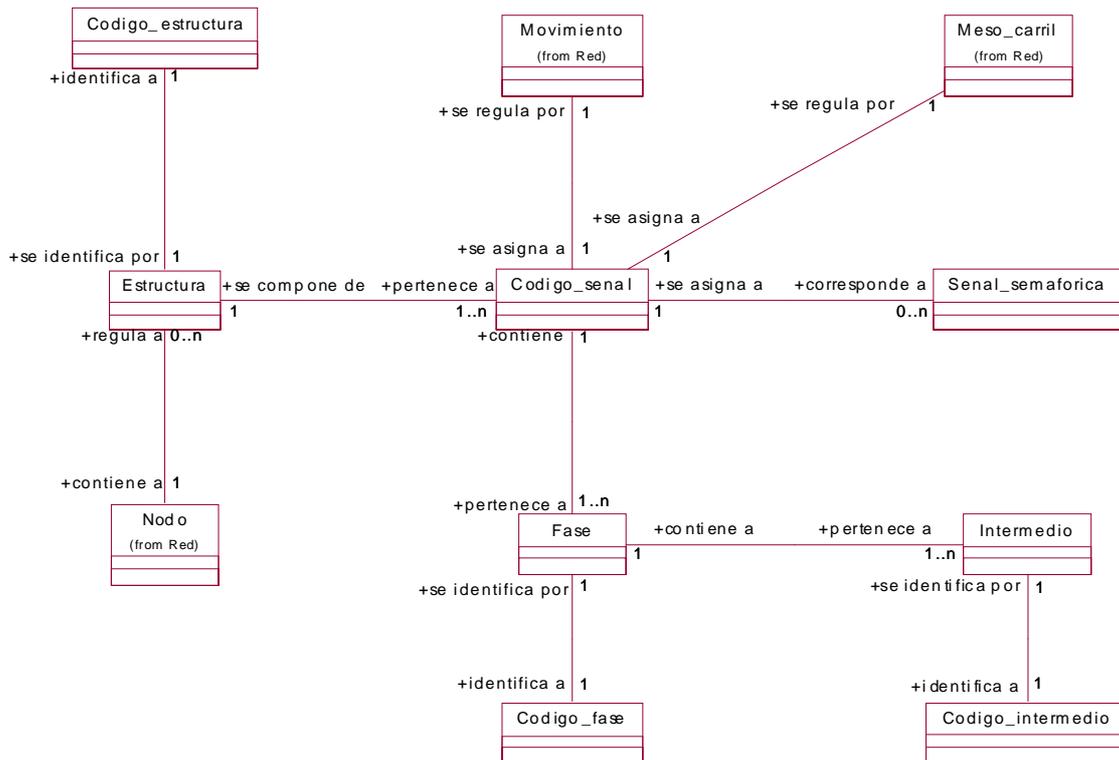


Figura 5.38. Diagrama de clases "Grupo Semafórico"

5.2.3.3 Paquete de datos "Simulador Microscópico"

El conjunto de clases que conforman este paquete contiene toda la información necesaria y precisa utilizada por los componentes dinámicos que forman parte del simulador. Las entidades principales de los simuladores microscópicos son los vehículos, de los cuales se van a estudiar sus comportamientos y las leyes que regulan sus desplazamientos por el viario.

Dependiendo del nivel de detalle del simulador microscópico, los vehículos pueden ser clasificados según diferentes criterios, permitiendo caracterizar el comportamiento dinámico de los vehículos por grupos. El comportamiento de los vehículos en el viario no está sólo en función del tipo de vehículo, sino que puede depender de diferentes factores como tipo de calzada, condiciones atmosféricas y sobre todo del tipo de conductor.

Ya se ha visto cómo el viario se representa con un nivel de detalle tal que los tramos se subdividen en carriles y los nodos están compuestos por movimientos, por lo tanto, los vehículos se desplazarán sobre los carriles que pertenecen a los tramos y por los movimientos de los nodos.

El comportamiento de los vehículos no sólo depende de las características de los conductores y vehículos, sino que depende de las influencias que ejercen sobre él otros componentes del viario como pueden ser semáforos, señales, aparcamientos, obstáculos e incluso otros vehículos.

5.2.3.3.1 Descripción de las clases del paquete "Simulador Microscópico"

5.2.3.3.1.1 Clase "Vehículo"

Los vehículos son las entidades fundamentales del simulador microscópico, ya que el fin del mismo es simular el desplazamiento de los vehículos por el viario. Como ya se ha comentado con anterioridad, el comportamiento de dichos vehículos vendrá determinado por una serie de factores intrínsecos al propio conductor y también por una serie de influencias externas al vehículo en sí.

Atributos de la clase "Vehículo":

- *influencia* : Código del elemento que influye al vehículo.
- *tipo_influencia* : Tipo de elemento que es el que influye al vehículo. Puede tomar los valores: NO_INFLUENCIA, VEHICULO, SENAL, SEMAFORO, OBSTACULO, APARCAMIENTO o APARCAMIENTO_M.
- *codigo* : Código del propio vehículo.
- *aparca_asignado* : Indica si el vehículo tiene asignado algún aparcamiento.
- *cambiando_entidad* : Indica si el vehículo se encuentra entrando o saliendo de un nodo o de un tramo.
- *tipo_vehiculo* : Tipo de vehículo del que se trata, pudiendo ser UTILITARIO o MERCANCIAS.
- *estado* : Estado de aceleración en el que se encuentra el vehículo. Toma uno de los siguientes valores: ACELERANDO, FRENANDO, APARCA, FLUJO_LIBRE, FRENANDO_SENAL, FRENANDO_AP o FRENANDO_OBSTACULO.
- *longitud* : Longitud en metros del vehículo.
- *tipo_conductor* : Código del tipo de conductor que conduce el vehículo. Se puede corresponder con un conductor novato, experimentado, agresivo o defensivo.
- *cod_tramo* : Tramo por el que circula el vehículo.
- *cod_carril* : Carril por el que transita el vehículo.

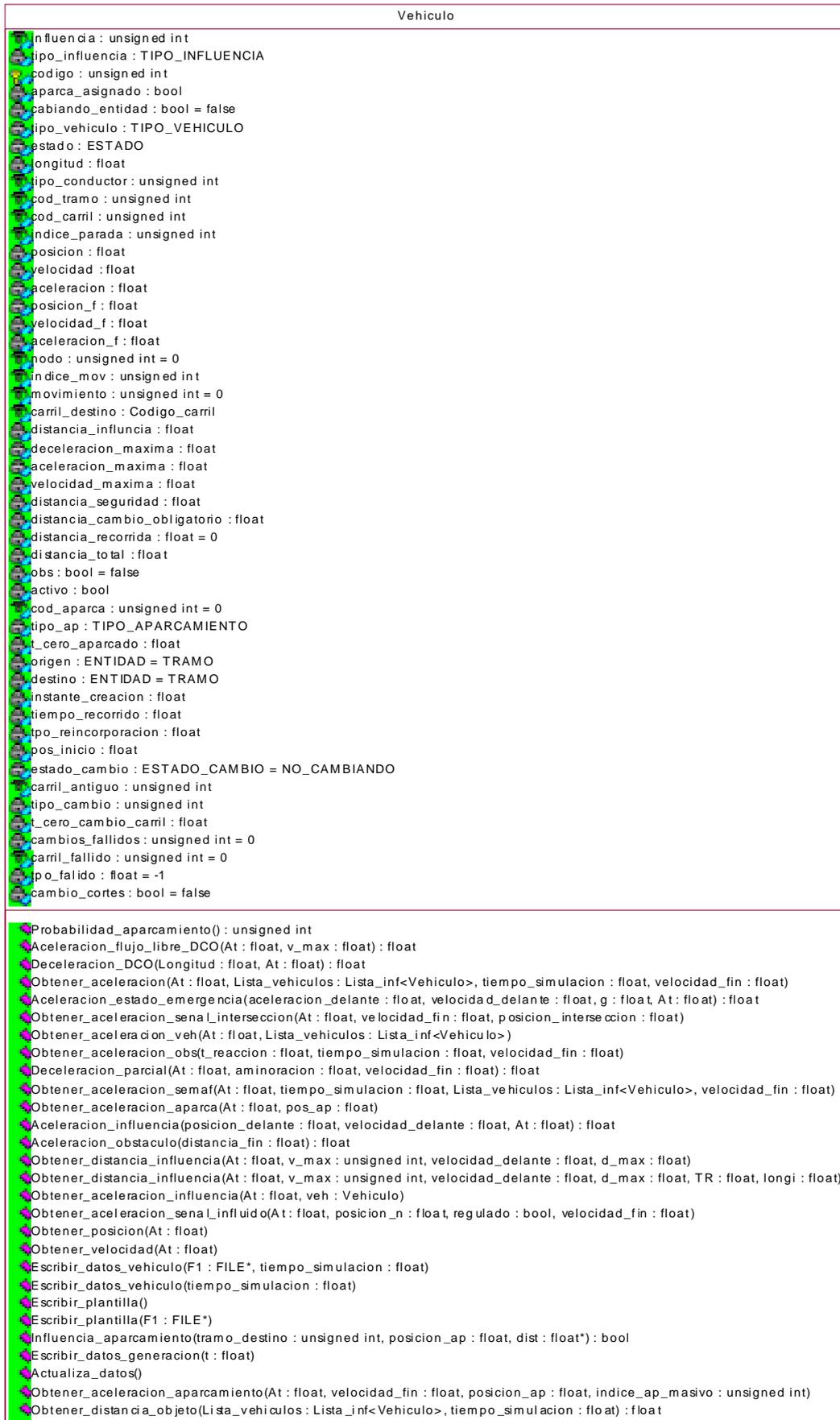


Figura 5.39. Clase "Vehiculo"

- *indice_parada* : Índice de la siguiente parada que debe efectuar el vehículo.
- *posicion* : Posición en metros del vehículo dentro del tramo.
- *velocidad* : Velocidad del vehículo en metros por segundo.
- *aceleracion* : Aceleración del vehículo en metros por segundo al cuadrado.
- *posicion_f* : Posición del vehículo en el siguiente instante de simulación.
- *velocidad_f* : Velocidad del vehículo en el siguiente instante de simulación.
- *aceleracion_f* : Aceleración del vehículo en el siguiente instante de simulación.
- *nodo* : Código del nodo por el que circula el vehículo.
- *indice_mov* : Índice del movimiento por el que circula el vehículo.
- *movimiento* : Código del movimiento por el que circula el vehículo.
- *carril_destino* : Código del carril hacia el que se dirige el vehículo.
- *distancia_influencia* : Distancia de influencia del vehículo, que depende de la velocidad que lleve en cada instante.
- *deceleracion_maxima* : Máxima deceleración en metros por segundo al cuadrado que puede efectuar un vehículo.
- *aceleracion_maxima* : Máxima aceleración que puede desarrollar el vehículo.
- *velocidad_maxima* : Máxima velocidad que puede alcanzar el vehículo.
- *distancia_seguridad* : Distancia de seguridad del vehículo, en la que puede llegar a detenerse completamente, depende de la velocidad que lleve y del tiempo de reacción del conductor.
- *distancia_cambio_obligatorio* : Distancia en metros hasta el final del tramo, a partir de la cual se puede realizar un cambio de carril obligatorio.
- *distancia_recorrida* : Distancia recorrida por el vehículo en metros.
- *distancia_total* : Distancia total que debe recorrer el vehículo.
- *obs* : Indica si el vehículo es un obstáculo en su carril.
- *activo* : Indica si el vehículo se encuentra activo sobre el viario.
- *destruido* : Indica si el vehículo ha sido destruido definitivamente.
- *cod_aparca* : Código del aparcamiento en el que va entrar el vehículo.
- *tipo_ap* : Tipo de aparcamiento que va a efectuar el vehículo. Los distintos tipos de aparcamiento posibles son los ya comentados en la clase "Aparcamiento".
- *t_cero_aparcado* : Tiempo en el que el vehículo entra en un aparcamiento.

- *origen* : Tipo de entidad por la que circula el vehículo. Puede tomar los valores TRAMO, NODO o NO_ENTIDAD.
- *destino* : Tipo de entidad destino del vehículo.
- *instante_creacion* : Instante de simulación en el que se genera el vehículo.
- *tiempo_recorrido* : Instante de simulación en el que el vehículo comenzó a circular por el tramo en el que se encuentra.
- *tpo_reincorporacion* : Instante de tiempo en el que el vehículo sale de un aparcamiento.
- *pos_inicio* : Posición del tramo en la que el vehículo comienza a circular por el mismo.
- *estado_cambio* : Señala el estado en el que se encuentra el vehículo respecto al cambio de carril. Puede tomar los valores: NO_CAMBIANDO si no está cambiando de carril, NUEVO si es un vehículo que está cambiando de carril o ANTIGUO si es un vehículo ficticio de la maniobra de cambio de carril.
- *carril_antiguo* : Número del carril del mismo tramo actual, por el que circulaba anteriormente el vehículo.
- *tipo_cambio* : Indica si el cambio de carril que realiza el vehículo es obligatorio o por mejora.
- *t_cero_cambio_carril* : Instante de simulación en el que el vehículo ha comenzado una maniobra de cambio de carril.
- *cambios_fallidos* : Número de veces que el vehículo ha intentado realizar el mismo cambio de carril sin éxito.
- *carril_fallido* : Número del último carril hacia el que el vehículo ha intentado realizar un cambio de carril sin conseguirlo.
- *tpo_fallido* : Instante de simulación en el que el vehículo intentó realizar su último cambio de carril sin éxito.
- *cambio_cortes* : Indica si el cambio de carril que realiza el vehículo se produce por cortesía de otro vehículo.

Métodos de la clase "Vehiculo":

- *unsigned int Probabilidad_aparcamiento (void)* : Devuelve la probabilidad de que un vehículo lleve a cabo un aparcamiento en función de la distancia que ha recorrido respecto a la que debe recorrer.
- *float Aceleracion_flujo_libre_DCO (float At, float v_max)* : Devuelve el valor de la aceleración que debe aplicar un vehículo cuando no se ve influido por ningún otro elemento.

- *float Deceleracion_DCO (float Longitud, float At)* : Devuelve el valor de la deceleración que debe ejercer un vehículo cuando debe frenar, pero tiene espacio suficiente para ello.
- *void Obtener_aceleracion (float At, Lista_inf<Vehiculo> Lista_vehiculos, float tiempo_simulacion, float velocidad_fin)* : Este procedimiento obtiene el valor de la aceleración del vehículo en cualquier situación en la que se encuentre.
- *float Aceleracion_estado_emergencia (float aceleracion_delante, float velocidad_delante, float g, float At)* : Devuelve el valor de la aceleración que debe aplicar un vehículo cuando el espacio que tiene para detenerse es incluso inferior a la distancia de seguridad.
- *void Obtener_aceleracion_senal_interseccion (float At, float velocidad_fin, float posicion_interseccion)* : Obtiene la aceleración de un vehículo cuando se ve influido por la señal que regula una intersección.
- *void Obtener_aceleracion_veh (float At, Lista_inf<Vehiculo> Lista_vehiculos)* : Busca el vehículo que le influye y obtiene la aceleración que debe aplicar un vehículo cuando se ve influido por otro vehículo que circula por su mismo carril.
- *void Obtener_aceleracion_obs (float t_reaccion, float tiempo_simulacion, float velocidad_fin)* : Busca el obstáculo que influye al vehículo y obtiene la aceleración que debe aplicar un vehículo cuando se ve influido por un obstáculo que se encuentra en su mismo carril.
- *float Deceleracion_parcial (float At, float aminoracion, float velocidad_fin)* : Devuelve el valor de la deceleración que debe aplicar un vehículo para llegar a una velocidad determinada.
- *void Obtener_aceleracion_semaf (float At, float tiempo_simulacion, Lista_inf<Vehiculo> Lista_vehiculos, float velocidad_fin)* : Obtiene la aceleración que debe aplicar un vehículo cuando se ve influido por un semáforo.
- *void Obtener_aceleracion_aparca (float At, float pos_ap)* : Obtiene la aceleración que debe aplicar un vehículo cuando se ve influido por tener que realizar un aparcamiento.
- *float Aceleracion_influencia (float posicion_delante, float velocidad_delante, float At)* : Obtiene la aceleración de un vehículo cuando se conoce la posición y velocidad del vehículo que le precede e influye.
- *float Aceleracion_obstaculo (float distancia_fin)* : Obtiene la aceleración que debe aplicar un vehículo cuando conoce la distancia a la que se encuentra el obstáculo que le influye.
- *void Obtener_distancia_influencia (float At, unsigned int v_max, float velocidad_delante, float d_max)* : Obtiene la distancia de seguridad y la de influencia del vehículo.

- *void Obtener_distancia_influencia (float At, unsigned int v_max, float velocidad_delante, float d_max, float TR, float longi)* : Obtiene la distancia de seguridad y la de influencia de un vehículo teniendo en cuenta el tiempo de reacción del conductor y la longitud del vehículo precedente.
- *void Obtener_aceleracion_influencia (float At, Vehiculo veh)* : Obtiene la aceleración del vehículo cuando *veh* se encuentra por delante de él, aunque puede que quizás no le afecte.
- *void Obtener_aceleracion_senal_influido (float At, float posicion_n, bool regulado, float velocidad_fin)* : Obtiene la aceleración de un vehículo cuando está influido por una señal.
- *void Obtener_posicion (float At)* : Obtiene cuál será la posición del vehículo en el siguiente instante de simulación, teniendo en cuenta cuál es la velocidad durante ese período de tiempo.
- *void Obtener_velocidad (float At)* : Obtiene cuál será la velocidad del vehículo en el siguiente instante de simulación, teniendo en cuenta cuál es la aceleración durante ese período de tiempo.
- *void Escribir_datos_vehiculo (FILE *F1, float tiempo_simulacion)* : Escribe los datos relativos al vehículo en un fichero que contiene información sobre todos los vehículos en cada instante de tiempo.
- *void Escribir_datos_vehiculo (float tiempo_simulacion)* : Escribe los datos relativos al vehículo en ese instante de simulación en un fichero específico para cada uno de los vehículos.
- *void Escribir_plantilla (void)* : Escribe la cabecera del fichero correspondiente a los datos del vehículo.
- *void Escribir_plantilla (FILE* F1)* : Escribe la cabecera correspondiente al fichero en el que se guardan los datos de todos los vehículos.
- *bool Influencia_aparcamiento (unsigned int tramo_destino, float posicion_ap, float* dist)* : Calcula la distancia hasta la posición a la que debe llegar en el tramo destino y determina si ese aparcamiento es el que influye al vehículo.
- *void Escribir_datos_generacion (float t)* : Escribe los datos relativos a la generación del vehículo en el fichero correspondiente al tramo en el que se produce la generación.
- *void Actualiza_datos (void)* : Convierte los datos de posición, velocidad y aceleración futuras correspondientes al vehículo en los presentes.
- *void Obtener_aceleracion_aparcamiento (float At, float velocidad_fin, float posicion_ap, unsigned int indice_ap_masivo)* : Obtiene la aceleración de un vehículo cuando se ve influido por un aparcamiento masivo.
- *float Obtener_distancia_objeto (Lista_inf<Vehiculo> Lista_vehiculos, float tiempo_simulacion)* : Obtiene la distancia al objeto (vehículo u obstáculo) que influye al vehículo.

5.2.3.3.1.2 Clase "Tipo_vehiculo"

Los objetos de esta clase contienen la información concerniente a los distintos tipos de vehículos que circulan por el viario.



Figura 5.40. Clase "Tipo_vehiculo"

Atributos de la clase "Tipo_vehiculo":

- *tipo* : Código del tipo de vehículo del que se trata.
- *nombre* : Nombre que se corresponde con el tipo de vehículo, pudiendo ser UTILITARIO o MERCANCIAS.
- *aceleracion_max* : Máxima aceleración que pueden desarrollar los vehículos de este tipo.
- *deceleracion_max* : Máxima deceleración que pueden desarrollar los vehículos de este tipo.
- *longitud* : Longitud en metros de este tipo de vehículos.
- *peso* : Peso en kilogramos de este tipo de vehículos.
- *porcentaje* : Tanto por ciento del total de vehículos que circulan por el viario que pertenecen a este tipo de vehículos.

5.2.3.3.1.3 Clase "Tipo_conductor"

Esta clase contiene la información relativa al tipo de conductor que lleva cada vehículo. El tipo de conductor influye notablemente sobre el comportamiento de los vehículos, por lo que es muy importante caracterizar bien cada tipo de conductor.

Atributos de la clase "Tipo_conductor":

- *tipo* : Código que identifica al tipo de conductor.
- *nombre* : Nombre correspondiente al tipo de conductor, puede ser Novato, Experimentado, Agresivo o Defensivo.
- *t_reaccion* : Tiempo que tarda en reaccionar el conductor ante algún evento inesperado.

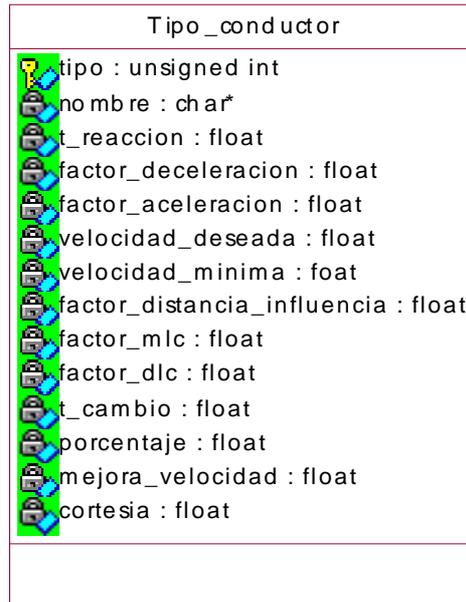


Figura 5.41. Clase "Tipo_conductor"

- *factor_deceleracion* : Factor que interviene en el cálculo de la deceleración que debe aplicar el vehículo en un instante.
- *factor_aceleracion* : Factor que interviene en el cálculo de la aceleración de un vehículo en flujo libre.
- *velocidad_deseada* : Velocidad que desearía llevar un vehículo que es conducido por este tipo de conductores.
- *velocidad_minima* : Velocidad mínima a la que quieren circular los conductores de ese tipo.
- *factor_distancia_influencia* : Factor que influye a la hora de calcular la distancia a partir de la cual se puede realizar un cambio de carril obligatorio.
- *factor_mlc* : Factor involucrado en el cálculo del gap necesario para realizar un cambio de carril obligatorio.
- *factor_dlc* : Factor involucrado en el cálculo del gap necesario para realizar un cambio de carril por mejora.
- *t_cambio* : Tiempo que tarda un vehículo conducido por este tipo de conductores en efectuar una maniobra de cambio de carril.
- *porcentaje* : Tanto por ciento de los vehículos que son conducidos por el tipo de conductor correspondiente.
- *mejora_velocidad* : Porcentaje de aceleración en el que debe aumentar la aceleración actual del vehículo para considerar conveniente la realización de un cambio de carril por mejora.

- *cortesia* : Probabilidad de dejar paso a un vehículo para que se sitúe en su carril, a pesar de no cumplirse el gap mínimo.

5.2.3.3.1.4 Clase "Configuracion_Microscopico"

Esta clase se emplea para configurar determinados parámetros empleados en el simulador microscópico.

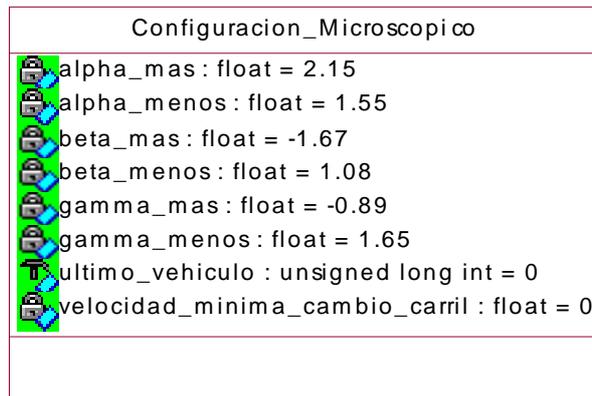


Figura 5.42. Clase "Configuracion_Microscopico"

Atributos de la clase "Configuracion_Microscopico":

- *alpha_mas* : Parámetro que interviene en el cálculo de la aceleración de un vehículo cuando el vehículo que le influye circula a mayor velocidad.
- *alpha_menos* : Parámetro que interviene en el cálculo de la aceleración de un vehículo cuando el vehículo que le influye circula a menor velocidad.
- *beta_mas* : Parámetro que interviene en el cálculo de la aceleración de un vehículo cuando el vehículo que le influye circula a mayor velocidad.
- *beta_menos* : Parámetro que interviene en el cálculo de la aceleración de un vehículo cuando el vehículo que le influye circula a menor velocidad.
- *gamma_mas* : Parámetro que interviene en el cálculo de la aceleración de un vehículo cuando el vehículo que le influye circula a mayor velocidad.
- *gamma_menos* : Parámetro que interviene en el cálculo de la aceleración de un vehículo cuando el vehículo que le influye circula a menor velocidad.
- *ultimo_vehiculo* : Código del último vehículo introducido en el simulador.
- *velocidad_minima_cambio_carril* : Mínima velocidad en metros por segundo que debe llevar un vehículo para realizar un cambio de carril.

5.2.3.3.1.5 Clase "Obstaculo"

Esta clase representa los distintos obstáculos que aparecen en el viario, imposibilitando la circulación por su posición en el carril, o carriles, que ocupan.

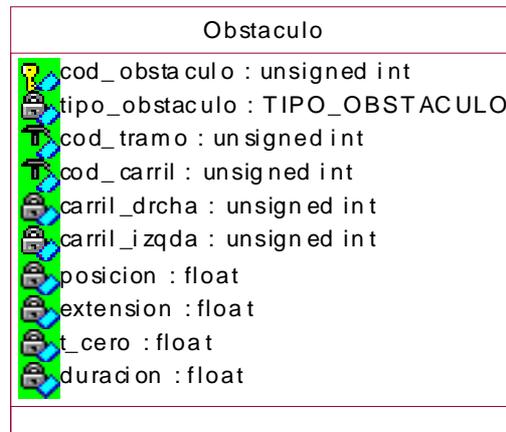


Figura 5.43. Clase "Obstaculo"

Atributos de la clase "Obstaculo":

- *cod_obstaculo* : Código identificativo del obstáculo.
- *tipo_obstaculo* : Indica el tipo de obstáculo de que se trata. Puede tomar uno de los siguientes valores: APARCANDO si se trata de un vehículo que está aparcando, NO_APARCANDO si es un vehículo que no está aparcando o CONSTANTE si es un obstáculo que se encuentra en su posición de manera permanente.
- *cod_tramo* : Tramo en el que se encuentra el obstáculo.
- *cod_carril* : Carril en el que se encuentra el obstáculo.
- *carril_drcha* : Número de carriles hacia la derecha por los que se extiende el obstáculo, a partir del que se considera el carril en el que está.
- *carril_izqda* : Número de carriles hacia la izquierda por los que se extiende el obstáculo, a partir del que se considera el carril en el que está.
- *posicion* : Posición relativa al tramo en la que comienza el obstáculo.
- *extension* : Longitud del obstáculo.
- *t_cero* : Instante de tiempo en el que aparece el obstáculo.
- *duracion* : Duración en segundos que tiene el obstáculo.

5.2.3.3.1.6 Clase "Parada"

Con esta clase se representan las distintas paradas que deben realizar los vehículos a lo largo de sus recorridos.



Figura 5.44. Clase "Parada"

Atributos de la clase "Parada":

- *tramo_origen* : Tramo en el que comienzan los vehículos que deben realizar la parada correspondiente.
- *codigo_carril* : Carril en el que se encuentra la parada.
- *codigo_tramo* : Tramo en el se debe efectuar la parada.
- *posicion* : Posición relativa al tramo en la que se encuentra la parada.
- *siguiente* : Índice de la siguiente parada que habrá de realizar el vehículo.
- *duracion_aparcado* : Duración en segundos que tiene la parada o aparcamiento que se ha de realizar.

5.2.3.3.2 Diagrama de clases del paquete "Simulador Microscópico"

El diagrama de clases correspondiente a este paquete de datos se muestra en la siguiente figura, en la que se aprecia que los elementos primordiales del simulador son los vehículos que circulan por el viario.

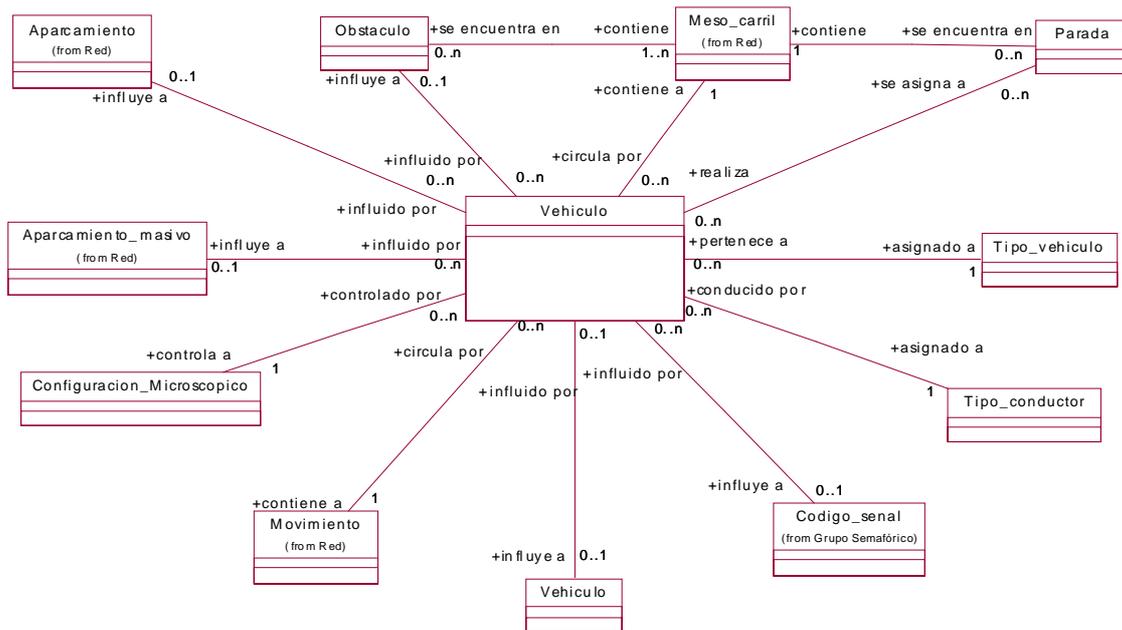


Figura 5.45. Diagrama de clases "Simulador Microscópico"

5.2.3.4 Paquete de datos "Generación Vehículos"

Este paquete contiene toda la información correspondiente al procedimiento de generación de nuevos vehículos en el simulador microscópico. Éste es uno de los aspectos fundamentales que pretende cubrir el presente proyecto, por lo que será expuesto con especial detalle.

Antes de pasar a ver las distintas clases que forman parte de este paquete de datos, cabe recordar que la generación de vehículos se va a realizar empleando funciones de generación trapezoidales afectadas por ciertas funciones de distribución aleatorias, por lo que ambos aspectos han de ser tenidos en cuenta en este apartado.

5.2.3.4.1 Descripción de las clases del paquete "Generación Vehículos"

5.2.3.4.1.1 Clase "Aleatorio"

Esta clase contiene la información necesaria para la generación de números aleatorios. Además, es la clase padre de las siguientes clases que se verán a continuación, las cuales representan las distintas funciones de distribución que podrán seguir los números pseudoaleatorios que se generen.

Cabe destacar que los atributos *numero* y *numero1* no son claves primarias de la clase, sino que son atributos protegidos, ya que se trata de una clase que es padre de otras clases.

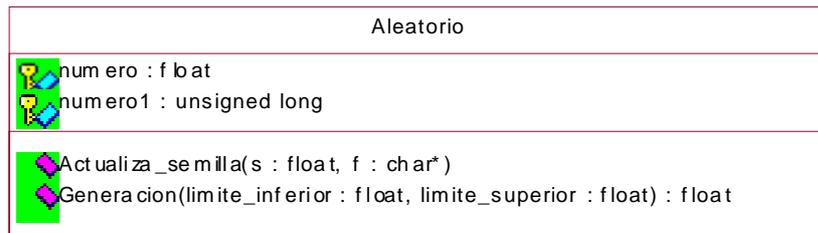


Figura 5.46. Clase "Aleatorio"

Atributos de la clase "Aleatorio":

- *numero* : Número aleatorio que se ha generado.
- *numero1* : Número empleado como semilla para la siguiente generación aleatoria. Está relacionado con el último número aleatorio que ha sido generado.

Métodos de la clase "Aleatorio":

- *void Actualiza_semilla (float s, char* f)* : Este método cambia la semilla que se emplea para la generación de números aleatorios, tomándola de un fichero, o bien, tomando la que se le pasa como parámetro de la función.
- *float Generacion (float limite_inferior, float limite_superior)* : Devuelve un número aleatorio generado entre los límites que se le indican como parámetros.

5.2.3.4.1.2 Clase "Exponencial"

Esta clase es hija de la clase "Aleatorio", además, dispone de los datos y funciones necesarios para la generación de números según una función de distribución exponencial de una media determinada.

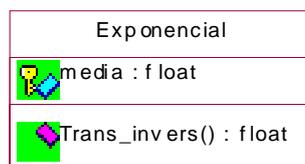


Figura 5.47. Clase "Exponencial"

Atributo de la clase "Exponencial":

- *media* : Valor de la media de la distribución exponencial.

Método de la clase "Exponencial":

- *float Trans_invers (void)* : Devuelve un número aleatorio generado según una función de distribución exponencial.

5.2.3.4.1.3 Clase "Normal"

Esta clase es hija de la clase "Aleatorio", además, dispone de los datos y funciones necesarios para la generación de números según una función de distribución normal de cierta media y varianza.

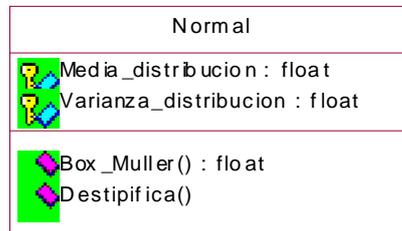


Figura 5.48. Clase "Normal"

Atributos de la clase "Normal":

- *Media_distribucion* : Valor de la media de la distribución normal.
- *Varianza_distribucion* : Valor de la varianza de la distribución normal.

Métodos de la clase "Normal":

- *float Box_Muller (void)* : Devuelve un número aleatorio que corresponde a una distribución normal de media 0 y varianza 1.
- *void Destipifica ()* : Convierte el número generado con la función anterior en uno correspondiente a la distribución de media y varianza deseadas.

5.2.3.4.1.4 Clase "Poisson"

Esta clase es hija de la clase "Aleatorio", además, dispone de los datos y funciones necesarios para la generación de números según una función de distribución Poisson de una media determinada.

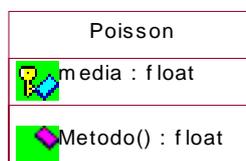


Figura 5.49. Clase "Poisson"

Atributo de la clase "Poisson":

- *media* : Valor de la media de la distribución Poisson.

Método de la clase "Poisson":

- *float Metodo ()* : Este método devuelve un número que responde a una función de distribución de Poisson de la media indicada.

5.2.3.4.1.5 Clase "Triangular"

Esta clase es hija de la clase "Aleatorio", además, dispone de los datos y funciones necesarios para la generación de números según una función de distribución triangular de unos parámetros determinados.

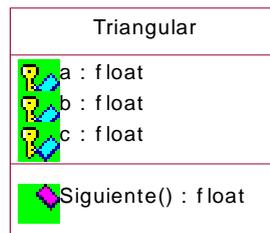


Figura 5.50. Clase "Triangular"

Atributos de la clase "Triangular":

- *a* : Número en el que comienza a ser distinto de cero la función de distribución triangular.
- *b* : Número en el que la función triangular alcanza su máximo (moda de la distribución).
- *c* : Número en el que la función triangular vuelve a ser cero.

Método de la clase "Triangular":

- *float Siguiente ()* : Método que devuelve un número generado según una función de distribución triangular correspondiente a los parámetros *a*, *b* y *c*.

5.2.3.4.1.6 Clase "Angulo"

Esta clase contiene la información necesaria para la definición de ángulos, así como distintas operaciones que se pueden realizar con los mismos. Estas operaciones serán muy importantes a la hora de trabajar con los trapecios que definen las funciones de generación de vehículos.

En esta clase, además del habitual operador de igualdad, también se han sobrecargado los operadores de suma y resta, tanto para sumar objetos de la clase "Angulo" como para sumar ángulos en grados. Asimismo, los operadores de comparación mayor y menor que, también han sido sobrecargados.

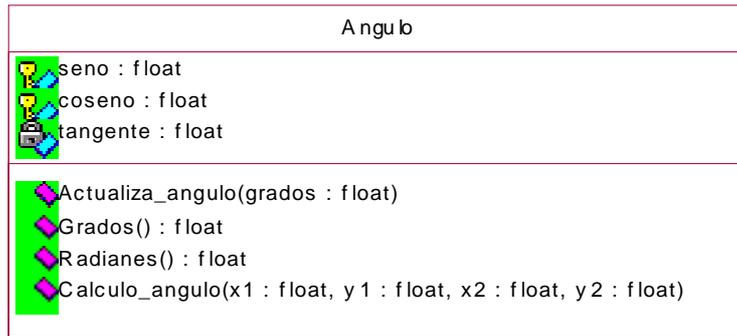


Figura 5.51. Clase "Angulo"

Atributos de la clase "Angulo":

- *seno* : Valor del seno del ángulo.
- *coseno* : Valor del coseno del ángulo.
- *tangente* : Valor de la tangente del ángulo.

Métodos de la clase "Angulo":

- *void Actualiza_angulo (float grados)* : Esta función obtiene el valor del seno, el coseno y la tangente de un ángulo que se le pasa como parámetro (en grados).
- *float Grados (void)* : Devuelve el valor en grados del ángulo correspondiente.
- *float Radianes (void)* : Devuelve el valor en radianes del ángulo correspondiente.
- *void Calculo_angulo (float x1, float y1, float x2, float y2)* : Calcula los parámetros correspondientes del ángulo que forma con la horizontal un segmento definido por los puntos que se pasan como parámetros.

5.2.3.4.1.7 Clase "Trapezio"

En esta clase se encuentran los datos necesarios para definir los trapezios que representan las funciones de generación de vehículos. Para ayudar a comprender el significado de los distintos parámetros del trapezio, se presenta a continuación una figura en la que se representa dicho trapezio.

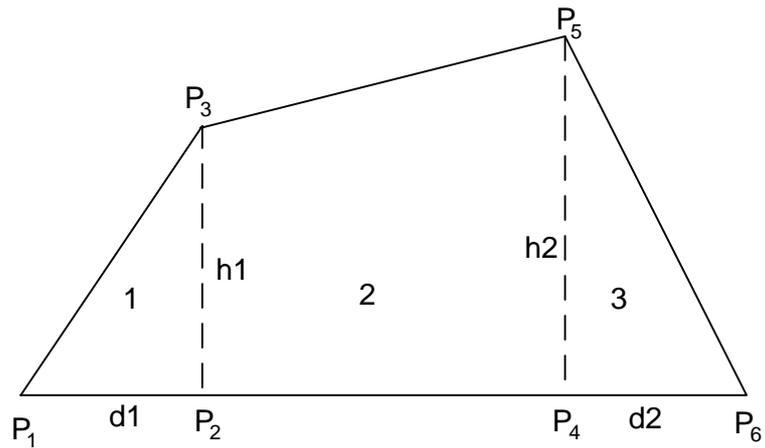


Figura 5.52. Parámetros del trapecio

Trapezio	
	desfase : float
	d1 : float
	d2 : float
	d : float
	h1 : float
	h2 : float
	angulo1 : Angulo
	angulo2 : Angulo
	area : float
	funcion_gen : Tipo_estadistica
	cod_trapezio : unsigned int
	normal : Normal
	exponencial : Exponencial
	poisson : Poisson
	triangular : Triangular
	Calcular_angulos()
	Area(sg_actual : float, Incr_t : float, varianza : float*) : float
	Calcula_altura_zona1(sg : float) : float
	Calcula_altura_zona2(sg : float) : float
	Calcula_altura_zona3(sg : float) : float
	Calcula_area() : float
	Obtencion_vehiculos(sg_actual : float, Incr_t : float, semilla : unsigned long) : float

Figura 5.53. Clase "Trapezio"

Atributos de la clase "Trapezio":

- *desfase* : Es el número de segundos que existen de desfase desde que comienza la simulación hasta que comienza el trapecio, es decir, hasta que se empiezan a generar vehículos. Gráficamente se puede interpretar como el tiempo transcurrido hasta que llega al punto P₁.

- *d1* : Es el tiempo que transcurre desde que comienza el trapecio hasta que se alcanza la altura *h1*, tal y como se puede apreciar en la figura 5.52.
- *d2* : Tiempo que tarda en producirse la bajada de la función trapezoidal desde la altura *h2* hasta que acaba el trapecio.
- *d* : Duración en segundos del trapecio entero, es decir, desde el punto P_1 hasta el P_2 .
- *h1* : Altura en la que se produce el primer cambio de pendiente en la forma de la función trapezoidal.
- *h2* : Altura en la que se produce el segundo cambio de pendiente en la forma de la función trapezoidal.
- *angulo1* : Ángulo formado por el segmento P_1 - P_3 con la horizontal.
- *angulo2* : Ángulo formado por el segmento P_6 - P_5 con la horizontal.
- *area* : Área total del trapecio.
- *funcion_gen* : Tipo de función estadística por la que se rige el trapecio para calcular el número de vehículos a generar. Puede tomar los valores NADA, NORMAL, POISSON, EXPONENCIAL o TRIANGULAR.
- *cod_trapecio* : Código que se emplea para identificar al trapecio.
- *normal* : Función de distribución normal que se empleará en el caso de que *funcion_gen* tome el valor NORMAL.
- *poisson* : Función de distribución de Poisson que se empleará en el caso de que *funcion_gen* tome el valor POISSON.
- *exponencial* : Función de distribución exponencial que se empleará en el caso de que *funcion_gen* tome el valor EXPONENCIAL.
- *triangular* : Función de distribución triangular que se empleará en el caso de que *funcion_gen* tome el valor TRIANGULAR.

Atributos de la clase "Trapecio":

- *void Calcular_angulos (void)* : Esta función calcula los ángulos del trapecio basándose en el resto de parámetros del mismo.
- *float Area (float sg_actual, float Incr_t, float* varianza)* : Esta función devuelve el área de la sección del trapecoide correspondiente al intervalo de tiempo que se va a simular.
- *float Calcula_altura_zona1 (float sg)* : Devuelve la altura del trapecoide en un punto perteneciente a la zona 1.
- *float Calcula_altura_zona2 (float sg)* : Devuelve la altura del trapecoide en un punto perteneciente a la zona 2.
- *float Calcula_altura_zona3 (float sg)* : Devuelve la altura del trapecoide en un punto perteneciente a la zona 3.

- *float Calcula_area (void)* : Esta función es la encargada de calcular el área total del trapecio en función de sus parámetros.
- *float Obtencion_vehiculos (float sg_actual, float Incr_t, unsigned long semilla)* : Devuelve el área de la sección del trapecio correspondiente al intervalo de tiempo que se va a simular, ya modificada por la función de distribución correspondiente al propio trapecio.

5.2.3.4.1.8 Clase "Generacion_pendiente"

Con la definición de objetos de esta clase, se pretende almacenar la información referente a generaciones de vehículos que no se han podido llevar a cabo en el momento deseado. Esto es debido a que en el momento en el que se debía realizar la generación, ya existía un vehículo en una posición cercana a donde debía aparecer el nuevo vehículo. Esto lleva a que dicha generación se retrase hasta el instante en el que pueda ser realizada.

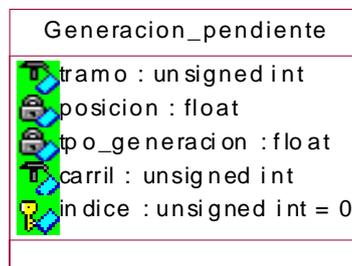


Figura 5.54. Clase "Generacion_pendiente"

Atributos de la clase "Generacion_pendiente":

- *tramo* : Código del tramo en el que se desea realizar la generación del vehículo.
- *posicion* : Posición de dicho tramo en la que aparecerá el vehículo.
- *tpo_generacion* : Instante de simulación en el que se intentó llevar a cabo la generación de manera fallida.
- *carril* : Carril en el que se desea llevar a cabo la generación del vehículo.
- *indice* : Este índice sirve para diferenciar generaciones pendientes que han de tener lugar en la misma posición de un mismo tramo.

5.2.3.4.2 Diagrama de clases del paquete "Generación Vehículos"

La relación entre las distintas clases que componen este paquete de datos se puede apreciar en la siguiente figura:

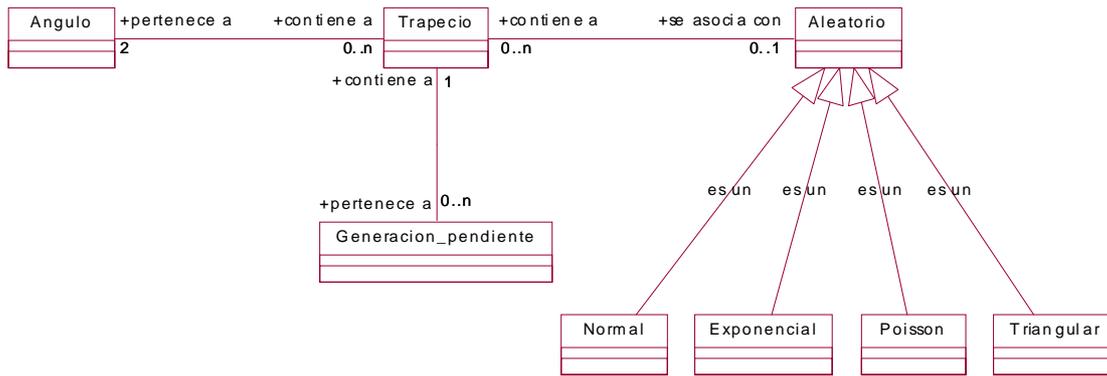


Figura 5.55. Diagrama de clases "Generación Vehículos"

5.2.3.5 Paquete de datos "Estadísticas"

Este paquete de datos contiene únicamente a una clase, sin embargo, dadas las relaciones de esta clase con otras de los paquetes ya vistos con anterioridad y la importancia de las estadísticas en este proyecto, resulta conveniente estudiarlas en un paquete de datos independiente.

5.2.3.5.1 Clase "Elemento_estadistico"

Los elementos estadísticos son estructuras de datos que permiten almacenar la información relativa a las distintas estadísticas que se van a extraer del comportamiento de los componentes del simulador microscópico.



Figura 5.56. Clase "Elemento_estadistico"

Atributos de la clase "Elemento_estadistico":

- *tipo* : Tipo de estadística que se almacena en el elemento estadístico. Puede tomar el valor: CAM_CAR para cambios de carril, CAM_VEH para cambios de carril de cada vehículo, VEH_CAR para vehículos de cada carril, VEH_GEN para la generación de vehículos, POS_GEN para las posiciones de generación de los vehículos, POS_APA para las posiciones de aparcamiento, TPO_SIS para el tiempo en el sistema de los vehículos,

TPO_TRA para el tiempo en los tramos de los vehículos y VEH_SEM para los vehículos detenidos en los semáforos.

- *entidad* : Código de la entidad a la que se refiere el elemento estadístico.
- *secundario* : Este atributo es empleado para poder referirse a elementos que necesitan ser referenciados mediante dos códigos, por ejemplo, para referirse a un carril en concreto, hace falta conocer el código del carril y también el del tramo en el que se encuentra.
- *valor* : Valor del elemento estadístico.
- *elementos* : Número de elementos que han sido considerados ya de dicho estadístico.
- *media* : Media de los valores considerados en el estadístico.
- *suma* : Suma de los valores de los elementos considerados en el elemento estadístico.

5.2.3.5.2 Diagrama de clases del paquete "Estadísticas"

En este diagrama se muestra de dónde toman los datos los diferentes tipos de elementos estadísticos comentados anteriormente.

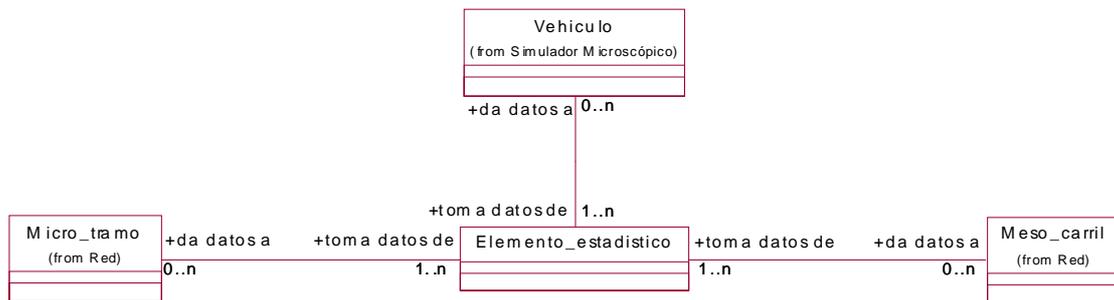


Figura 5.57. Diagrama de clases "Estadísticas"

5.2.3.6 Paquete de datos "Listas"

Los componentes de este paquete permiten la agrupación de muchos elementos de una misma clase en una lista. Esta forma de almacenar la información resulta muy eficiente, ya que permite el acceso a cualquier elemento de la lista de manera inmediata, sin más que realizar un búsqueda a través de la misma.

5.2.3.6.1 Clase "Lista_inf"

Esta clase posee una peculiaridad respecto al resto de las clases vistas hasta el momento, ya que se trata de una clase parametrizada, también conocidas como *plantillas* o *templates*. La existencia del parámetro, al que se le denomina *Y*, permite que se puedan construir listas de elementos pertenecientes a cualquier clase (aunque una misma lista sólo puede contener elementos correspondientes a la clase que indica *Y*).

Al igual que en las clases ya vistas, es necesaria la sobrecarga de los operadores de comparación e igualdad, ya que son fundamentales para la búsqueda de elementos. También es importante resaltar que la gestión de la memoria que necesitan estas listas es llevada a cabo a través de los métodos implementados a tal efecto.

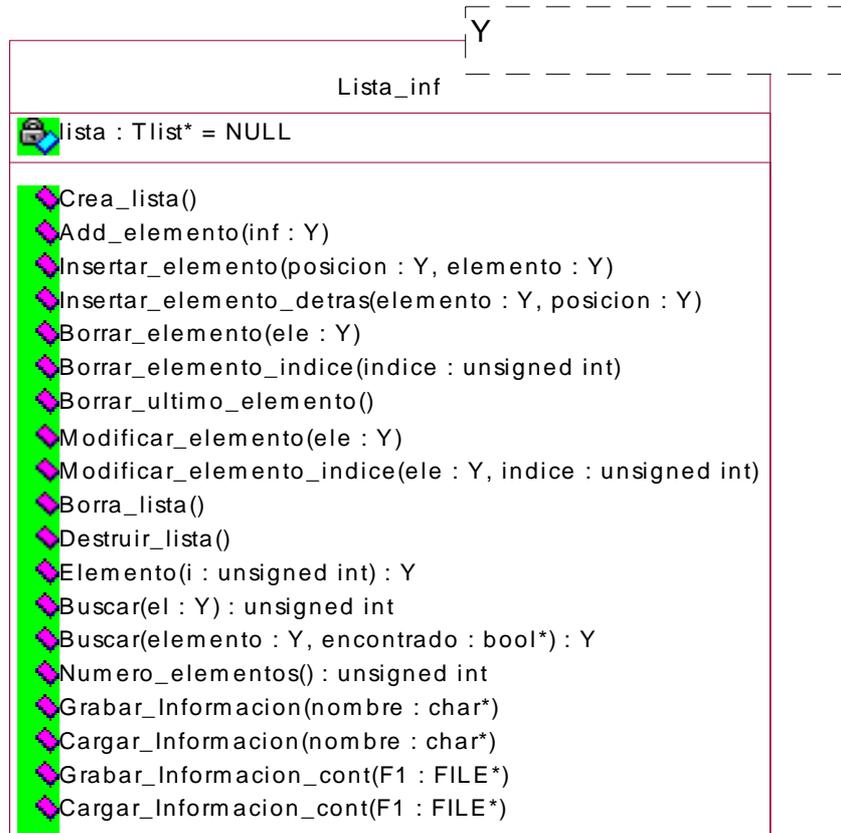


Figura 5.58. Clase parametrizada "Lista_inf"

Atributo de la clase "Lista_inf":

- *lista* : Puntero a los objetos que van a formar parte de la lista.

Métodos de la clase "Lista_inf":

- *void Crea_lista (void)* : Reserva la memoria necesaria para la lista.
- *void Add_elemento (Y inf)* : Añade el elemento *inf* al final de la lista.
- *void Insertar_elemento (Y posicion, Y elemento)* : Inserta el elemento *elemento* en la posición que ocupa el elemento *posicion*.
- *void Insertar_elemento_detras (Y elemento, Y posicion)* : Inserta el elemento *elemento* detrás de la posición que ocupa el elemento *posicion*.
- *void Borrar_elemento (Y ele)* : Borra el elemento *ele* de la lista.

- *void Borrar_elemento_indice (unsigned int indice)* : Borra el elemento de la lista que tiene por índice el indicado en el parámetro.
- *void Borrar_ultimo_elemento (void)* : Borra el último elemento de la lista.
- *void Modificar_elemento (Y ele)* : Busca en la lista el elemento que a efectos del operador == sea igual a *ele* y lo sustituye por éste.
- *void Modificar_elemento_indice (Y ele, unsigned int indice)* : Sustituye el elemento de la posición *indice* por *ele*.
- *void Borra_lista (void)* : Borra todos los elementos de la lista.
- *void Destruir_lista (void)* : Libera la memoria reservada para la lista.
- *Y Elemento (unsigned int i)* : Devuelve el elemento que ocupa la posición *i*.
- *unsigned int Buscar (Y el)* : Devuelve el índice de la posición que ocupa *el* en la lista, o -1 si no existe.
- *Y Buscar (Y elemento, bool* encontrado)* : Busca *elemento* en la lista, si lo encuentra, lo devuelve como resultado de la función y activa *encontrado* a un valor verdadero, si no lo encuentra pone *encontrado* a un valor falso.
- *unsigned int Numero_elementos (void)* : Devuelve el número de elementos de la lista.
- *void Grabar_Informacion (char* nombre)* : Graba la información de la lista en un fichero binario situado donde indica *nombre*.
- *void Cargar_Informacion (char* nombre)* : Carga la información en la lista desde un fichero binario situado donde indica *nombre*.
- *void Grabar_Informacion_cont (FILE* F1)* : Graba la información contenida en la lista en el fichero binario *F1*.
- *void Cargar_Informacion_cont (FILE* F1)* : Carga en la lista la información contenida en el fichero binario *F1*.

5.2.3.6.2 Diagrama de clases del paquete "Listas"

Dado que este paquete sólo contiene una clase, su diagrama de clases se reduce a ver con qué clases de otro paquetes se relaciona la clase "Lista_inf". Esta clase se relaciona con todas aquellas clases que le sirven como parámetro para indicar qué tipo de elementos contiene la lista.

Ya que existe una gran cantidad de clases que sirven como parámetro de la plantilla, únicamente se va a representar en el diagrama su relación con la clase "Vehiculo", manteniendo idéntica relación con el resto de clases siguientes: "Nodo", "Elemento_estadistico", "Micro_tramo", "Obstaculo", "Tipo_vehiculo", "Parada", "Tipo_conductor", "Aparcamiento_masivo", "Senal_semaforica", "Trapeccio", "Fase", "Generacion_pendiente", "Meso_carril" e "Intermedio".



Figura 5.59. Diagrama de clases "Listas"

6 Diseño del Simulador Microscópico de Tráfico

6.1 Introducción

El simulador microscópico de tráfico ha sido implementado utilizando un lenguaje de programación orientado a objetos como es C++. Más concretamente se ha empleado el entorno de programación que proporciona la herramienta C++ Builder, la cual suministra un entorno de programación visual que facilita en gran medida el empleo de un lenguaje como C++, que se caracteriza por tener una gran cantidad de reglas y términos diferentes.

En este apartado se va a realizar una descripción muy breve de lo que es la programación orientada a objetos y de aquellos componentes fundamentales que forman parte de la misma.

La programación orientada a objetos (POO) se define como una técnica o estilo de programación que utiliza objetos como bloque esencial de construcción. Un *objeto* es una unidad que contiene datos y las funciones que operan sobre esos datos. Estos objetos se corresponden con los elementos que debe utilizar el programa, como pueden ser los tramos o los vehículos en el caso del simulador.

Una *clase* es una colección de objetos que poseen características y operaciones comunes. En una clase se encuentra contenida toda la información necesaria para crear nuevos objetos pertenecientes a esa misma clase.

Los objetos solamente pueden ser accedidos y manipulados por las operaciones que previamente se hayan definido sobre su clase. Para evitar que dichos objetos sean utilizados en otras operaciones, se utiliza una técnica denominada *encapsulación*, que esconde los datos y sólo permite acceder a ellos de forma controlada.

Otros conceptos de interés en la POO son el *polimorfismo* que permite utilizar un mismo nombre para un tipo genérico de acciones, y la *herencia*, mediante la cual un objeto puede adquirir las propiedades de otro objeto.

6.2 Diagramas de flujo del simulador microscópico de tráfico

El diseño del simulador va ser representado mediante los diagramas de flujo correspondientes a la implementación en C++ de las funciones principales que intervienen en él. El significado de los símbolos utilizados en los diagramas de flujo es el que se indica en el Apéndice A. Asimismo, la notación empleada para referirse a los distintos parámetros de los vehículos es la que se definió en el apartado 2.2.

En dichos diagramas son utilizados objetos pertenecientes a las clases que se han visto en el apartado 5, correspondiente al diseño en UML. La forma de acceder a dichos objetos es a través de sus atributos y métodos convenientes.

Principalmente se van a analizar aquellos procedimientos relativos a los 3 puntos fundamentales del proyecto, es decir, los modelos cambios de carril, generación de vehículos y estadísticas. Sin embargo, estos modelos carecen de significado por sí

mismos, por lo que es necesario enmarcarlos dentro de lo que es el simulador microscópico de tráfico al completo.

La estructura básica del simulador microscópico de tráfico viene determinada por el diagrama de flujo de la figura 6.1.

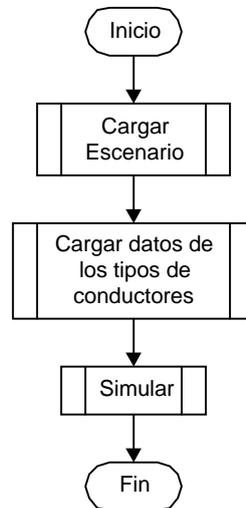


Figura 6.1. Estructura general del simulador

El primer paso consiste en cargar el escenario que va a ser objeto de la simulación. Para ello, se comienza por determinar el valor de los parámetros correspondientes al simulador microscópico, aquellos que se vieron en el apartado 5.2.3.3.1.4. A continuación, se procede a la lectura de los ficheros de texto que contienen los datos referentes a tipos de vehículos, tipos de conductores, tramos, movimientos, giros, incompatibilidades, obstáculos, aparcamientos y paradas. También se aprovecha esta carga del escenario para generar la cabecera del fichero de texto que contendrá los datos de todos los vehículos durante la simulación y para crear las listas correspondientes a los semáforos, estadísticas y generación de vehículos.

Entrando en aspectos más específicos de este proyecto, en este primer paso de cargar el escenario, se aprovecha para generar las formas trapezoidales que determinan las funciones de generación de vehículos, para ello, hay que indicar el valor de todos los parámetros vistos en la correspondiente clase. Dichos trapecios se almacenan en la lista de trapecios correspondiente.

En lo concerniente a las estadísticas, además de crear las listas correspondientes a todos los tipos explicados en el apartado 4, se crea un elemento estadístico por cada uno de los carriles de cada tramo, destinado a contener el número de vehículos que circulan por él. También se crea otro elemento estadístico para cada tramo, que se usan para indicar el número de vehículos generados en cada instante de tiempo en dicho tramo. Todos estos elementos estadísticos se añaden a sus listas correspondientes.

Volviendo a la figura 6.1, el segundo paso (relacionado con la carga de los parámetros de los distintos tipos de conductores) es opcional. En el caso de no llevarse a cabo, se emplean los valores que fueron utilizados en la última simulación. Para la introducción de nuevos valores de los parámetros se utiliza el interfaz mostrado en la figura 6.2, en el cuál se aprecian algunos de dichos parámetros.

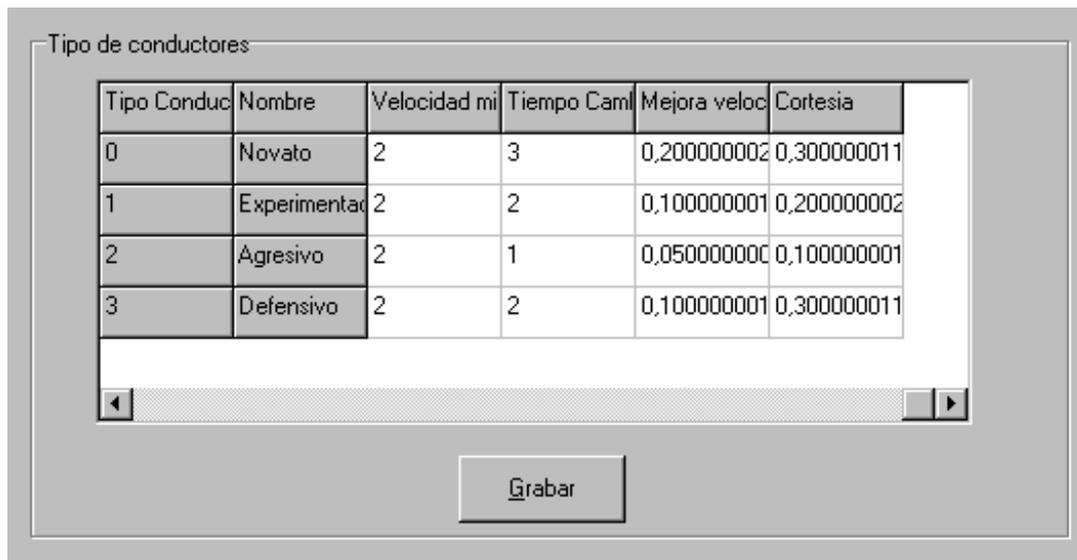


Figura 6.2. Interfaz para modificar los parámetros de los tipos de conductores

En la anterior figura se aprecia que se han introducido los nuevos parámetros relacionados con el nuevo modelo de cambio de carril, como son el tiempo empleado por cada tipo de conductor en efectuar una maniobra de cambio de carril, el tanto por ciento de mejora de velocidad que se debe dar para que un vehículo lleve a cabo un cambio de carril por mejora o el porcentaje de probabilidad de que un conductor de cada tipo deje realizar a otro vehículo un cambio de carril por cortesía.

Cuando se pulsa el botón *Grabar* en el interfaz mostrado, la lista que contiene los datos correspondientes a los tipos de conductores es actualizada para que contenga los valores que se le hayan indicado.

El último de los bloques de la figura 6.1 se corresponde con la simulación propiamente dicha y constituye el núcleo principal del simulador. En este bloque se lleva a cabo la simulación del escenario que ha sido cargado y responde al diagrama de flujo que se muestra en la figura 6.3.

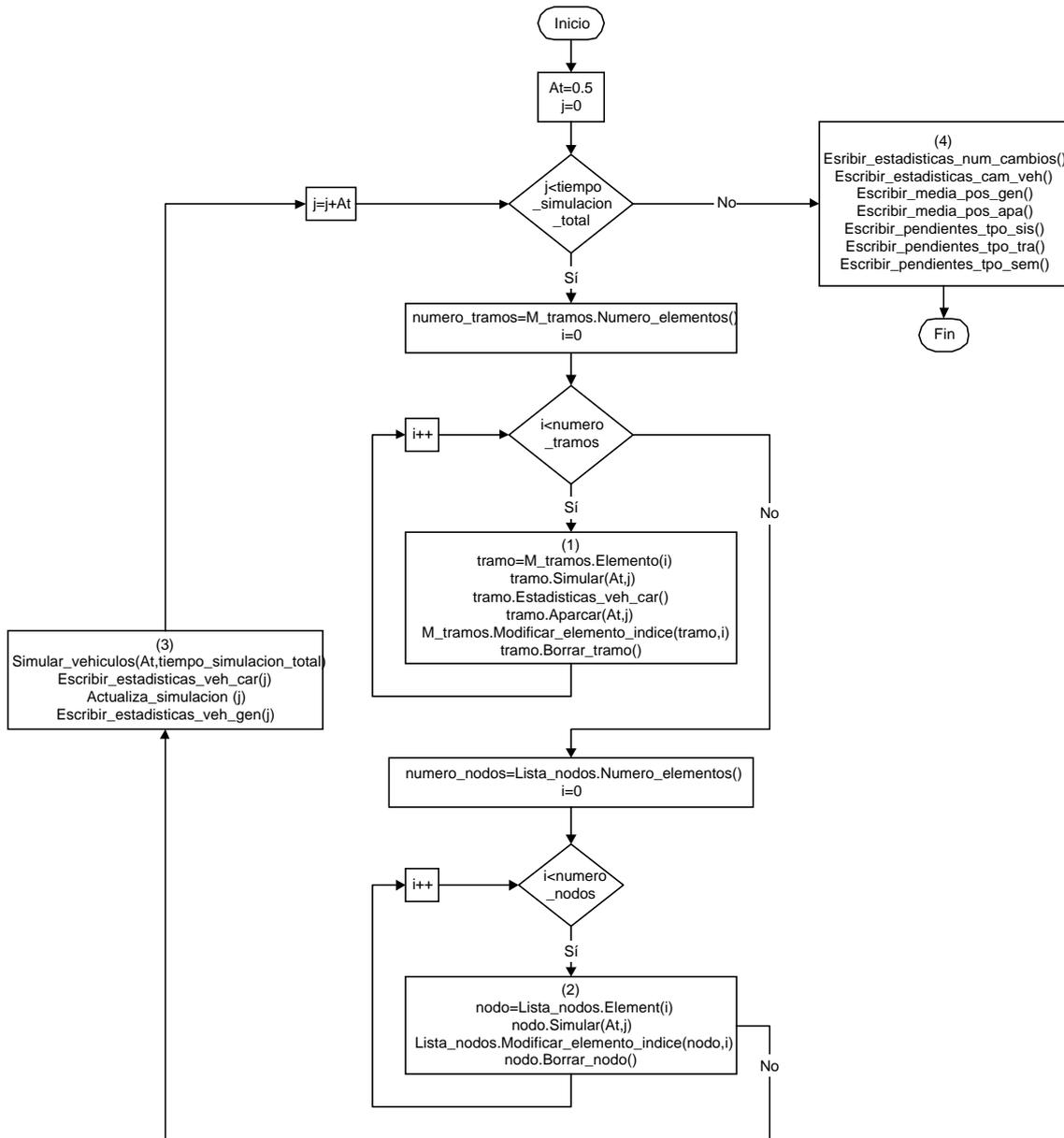


Figura 6.3. Procedimiento general de simulación

Observando la figura 6.3 se aprecia que la simulación comienza con la determinación del valor del intervalo de tiempo que se va a utilizar para determinar los instantes de simulación, que en este caso se toma como 0.5 segundos. A continuación comienza un bucle en el que se van a repetir una serie de operaciones hasta que se llegue al instante final de simulación.

En cada instante de simulación, se va a comenzar por simular todos los tramos existentes en el viario (1) y a continuación se simulan los nodos (2). Una vez se hayan simulado todos los tramos y nodos, se pasa a un nuevo bloque (3), en el cual se actualizan distintos valores correspondientes a la simulación y se escriben algunas estadísticas referidas al instante de simulación. Una vez que se ha alcanzado el tiempo total que dura la simulación, se procede a la escritura de otras estadísticas globales de la simulación (4).

En los siguientes apartados se estudian con detalle cada uno de los bloques señalados en el diagrama de flujo de la figura 6.3.

6.2.1 Simulación de tramos

La simulación de los tramos que componen el viario es llevada a cabo mediante las funciones indicadas en el bloque (1) de la figura 6.3.

Simular

La primera de ellas consiste en simular un tramo y responde al diagrama de flujo mostrado en la figura 6.4.

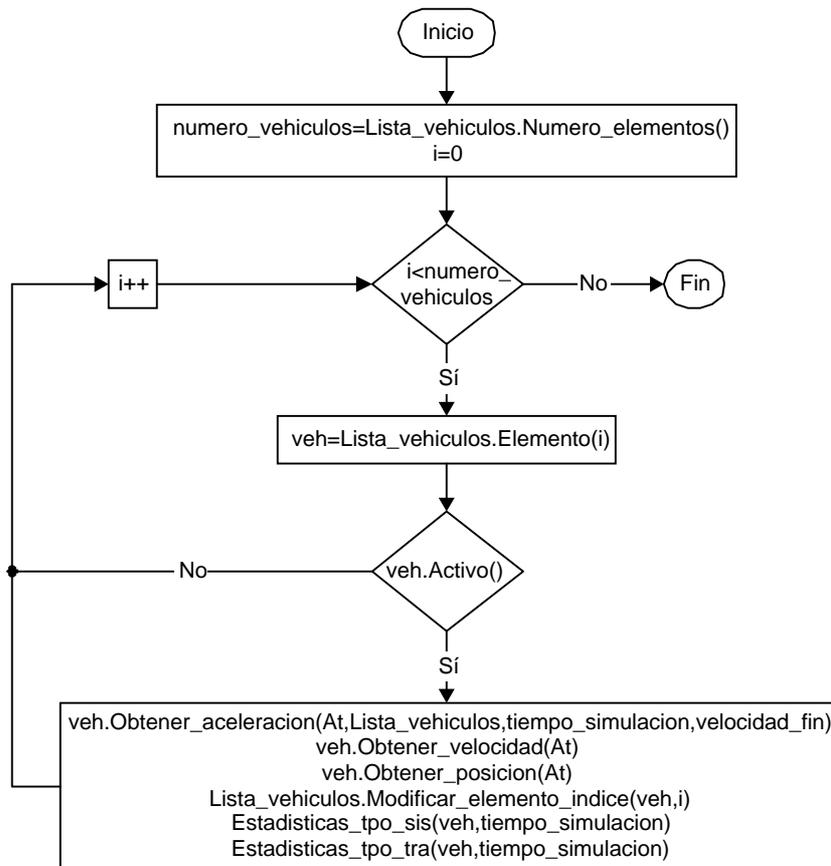


Figura 6.4. Simular tramo

Como se observa en la figura 6.4, el proceso de simular un tramo consiste en obtener la aceleración, velocidad y posición correspondientes a todos los vehículos que se encuentran activos en el tramo. Las funciones que obtienen estos valores no son analizadas, ya que se salen del alcance de este proyecto. Una vez obtenidos dichos parámetros, se pasa a obtener algunas estadísticas de los vehículos del tramo.

La primera de las estadísticas se corresponde con el tiempo que llevan en el sistema cada uno de los vehículos que están en el tramo. Su diagrama de flujo se muestra a continuación en la figura 6.5.

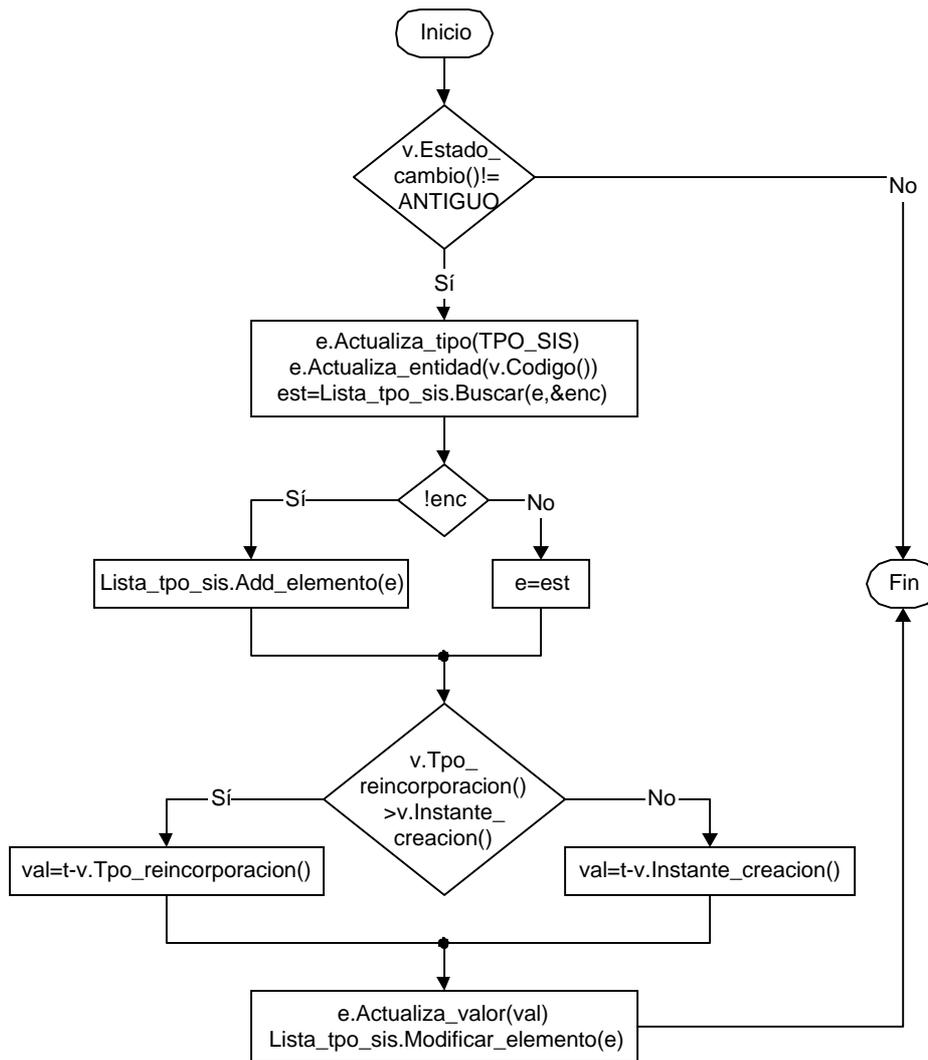


Figura 6.5. Estadísticas de tiempo en el sistema

En esta figura se observa que para tener en cuenta el tiempo que un vehículo lleva en el sistema, la primera condición que debe darse es que el vehículo se encuentre activo en el sistema, ya que de lo contrario, se considera que ya el vehículo no pertenece al mismo. A continuación se busca en la lista correspondiente a las estadísticas de tiempo en el sistema de los vehículos, aquel elemento estadístico que se corresponde con el vehículo en cuestión, en el caso de no existir dicho elemento, se añade a la lista. Por último, el valor del estadístico se actualiza al tiempo transcurrido desde que el vehículo se creó, o bien, desde que el vehículo se reincorporó a la circulación tras haber estado aparcado.

Como consecuencia de esto, el valor de las estadísticas de tiempo en el sistema, consideran el tiempo que el vehículo lleva circulando desde que se incorporó al tráfico por última vez, es decir, no se acumulan los tiempos que haya podido estar circulando el vehículo antes de realizar un aparcamiento.

La otra estadística que se contempla en la simulación del tramo se corresponde con el tiempo que lleva cada vehículo circulando por el tramo en cuestión. El diagrama de flujo de dicha estadística se aprecia en la figura 6.6.

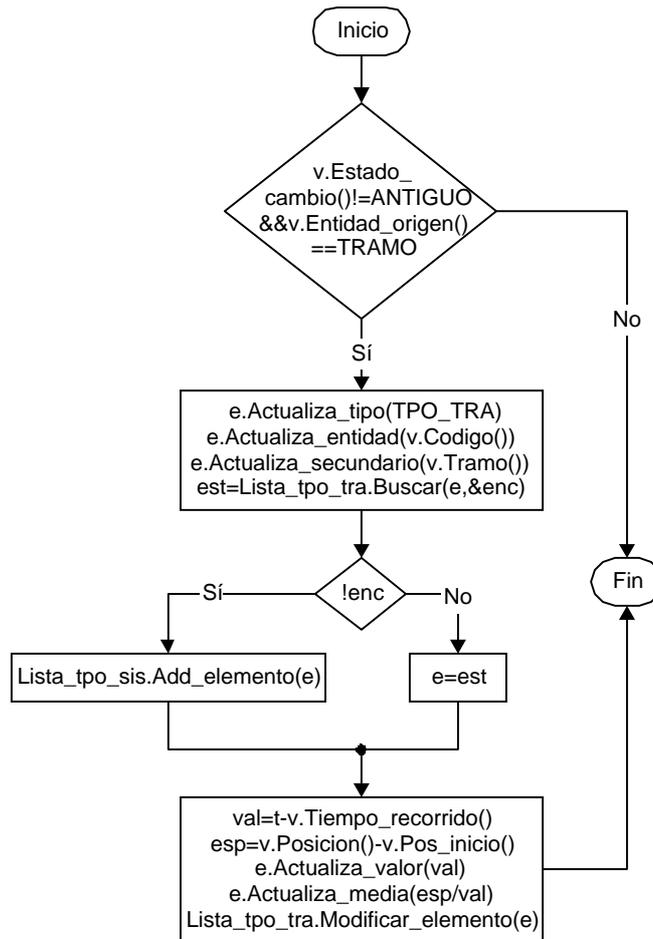


Figura 6.6. Estadísticas de tiempo en el tramo

Para controlar las estadísticas correspondientes al tiempo que lleva un vehículo en el tramo, lo primero que hay que comprobar es que el vehículo se encuentre activo y que efectivamente se encuentre en un tramo. A continuación, se busca en la lista de las estadísticas correspondientes al tiempo en un tramo de un vehículo, el elemento estadístico correspondiente a ese vehículo en ese tramo. En el caso de no encontrarse, se añade el elemento a la estadística. Por último, se actualiza el valor del estadístico al tiempo que ha transcurrido desde que el vehículo inició su recorrido por el tramo.

El atributo *media* de este elemento estadístico es aprovechado para almacenar la velocidad media que ha llevado el vehículo en su recorrido por el tramo, ya que es inmediato conocer el espacio recorrido por el mismo a partir de la posición en la que se incorporó en el tramo. Realizando la división de dicho espacio entre el tiempo que lleva en el tramo, se obtiene dicha velocidad media.

Estadísticas_veh_car

Una vez finalizada la simulación del tramo, la siguiente función que se lleva a cabo (ver bloque (1) de la figura 6.3) consiste en obtener las estadísticas asociadas al número de vehículos que circulan en ese instante por cada uno de los carriles del tramo. El diagrama de flujo de dicha función se presenta en la figura 6.7.

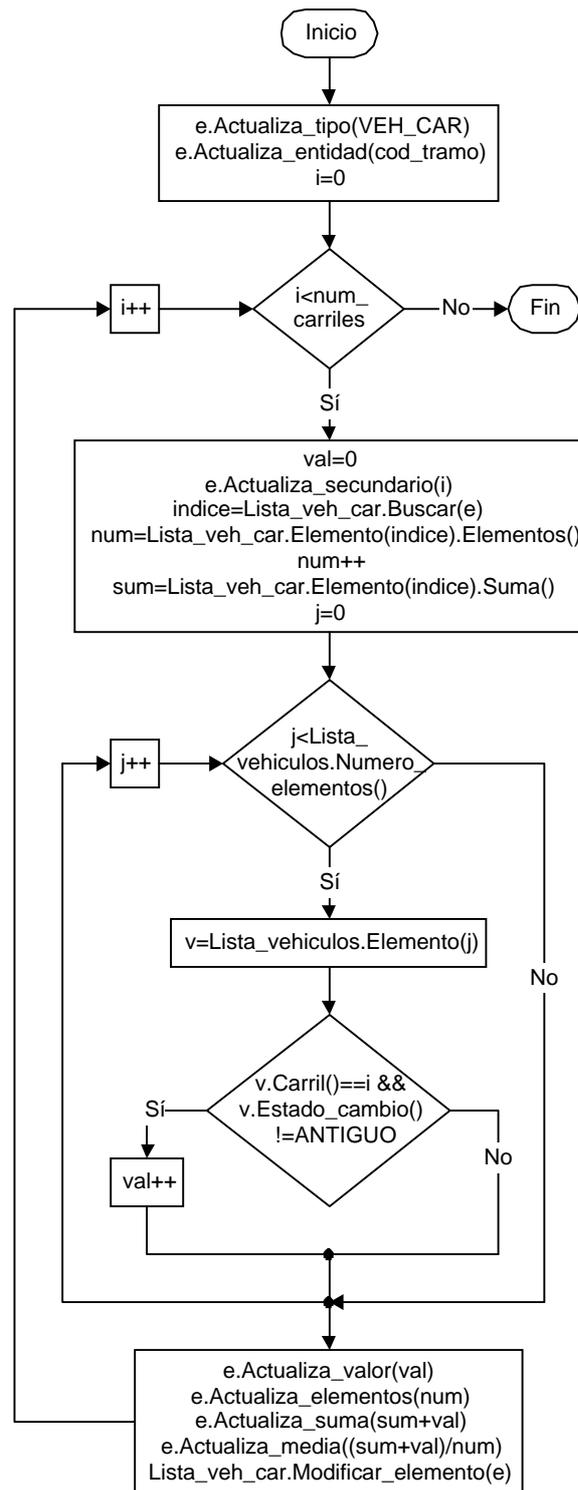


Figura 6.7. Estadísticas de vehículos por carril

El procedimiento mostrado en la figura 6.7 comienza con un bucle en el que se van a recorrer todos los carriles de los que está compuesto el tramo. Para cada uno de ellos, se busca en la lista de estadísticas de vehículos por carril el elemento estadístico correspondiente y se aumenta en una unidad el atributo que indica el número de veces que se ha considerado dicho elemento. A continuación, se cuentan el número de vehículos que hay en el carril y no son vehículos ficticios, es decir, sólo se consideran aquellos vehículos que son reales, no los creados para producir el efecto de desdoble en

los cambios de carril. El valor del elemento estadístico se actualiza al número de vehículos contados. Asimismo, se actualiza el valor del número medio de vehículos que circulan por ese carril, para lo que se tienen en cuenta el número total de vehículos que se han acumulado y el número de veces que se ha realizado este algoritmo.

Aparcar

La siguiente función del bloque (1) de la figura 6.3 se corresponde con la realización de los aparcamientos en el tramo. El procedimiento que lleva a cabo un vehículo para comprobar si debe realizar un aparcamiento no forma parte de los objetivos que este proyecto debe cubrir. Sin embargo, hay algunos aspectos de dicha maniobra que están relacionados con los cambios de carril a realizar por los vehículos y con las estadísticas, los cuales sí van a ser abordados.

Cuando un vehículo se encuentra a una distancia determinada del lugar en el que debe realizar un aparcamiento (40 metros en el simulador) y no está en un carril que le permita acceder al mismo, debe llevar a cabo una maniobra de cambio de carril para desplazarse hacia dicho carril.

Antes de iniciar dicha maniobra, debe comprobar que dispone de espacio suficiente para realizarla con seguridad, para lo que deben existir en el nuevo carril unas distancias delantera y trasera determinadas en las que no haya ningún obstáculo (estas distancias se han considerado iguales a 5 metros en el simulador). Este algoritmo responde al diagrama de flujo mostrado en la figura 6.8.

En la figura 6.8 se observa que el procedimiento para aceptar el gap a la hora de situarse en el carril apropiado para realizar un aparcamiento se divide en dos pasos. Por un lado, se estudian los vehículos existentes en el tramo, en el momento en el que se encuentre alguno que está situado en el carril hacia el que se quiere desplazar el vehículo y a una distancia inferior que la deseada, ya sea por delante o por detrás del vehículo, el gap queda rechazado. En el caso de que se recorra la lista de vehículos completa y no se encuentre ningún vehículo que cumpla lo anterior, se pasa a analizar los obstáculos. Para los obstáculos, hay que verificar que en la distancia que se ha determinado no haya ningún obstáculo que ocupe el carril deseado por el vehículo en el instante de tiempo en el que desea cambiar de carril. Si no se encuentra ni vehículo ni obstáculo que impidan el cambio de carril, el vehículo puede proceder a la maniobra.

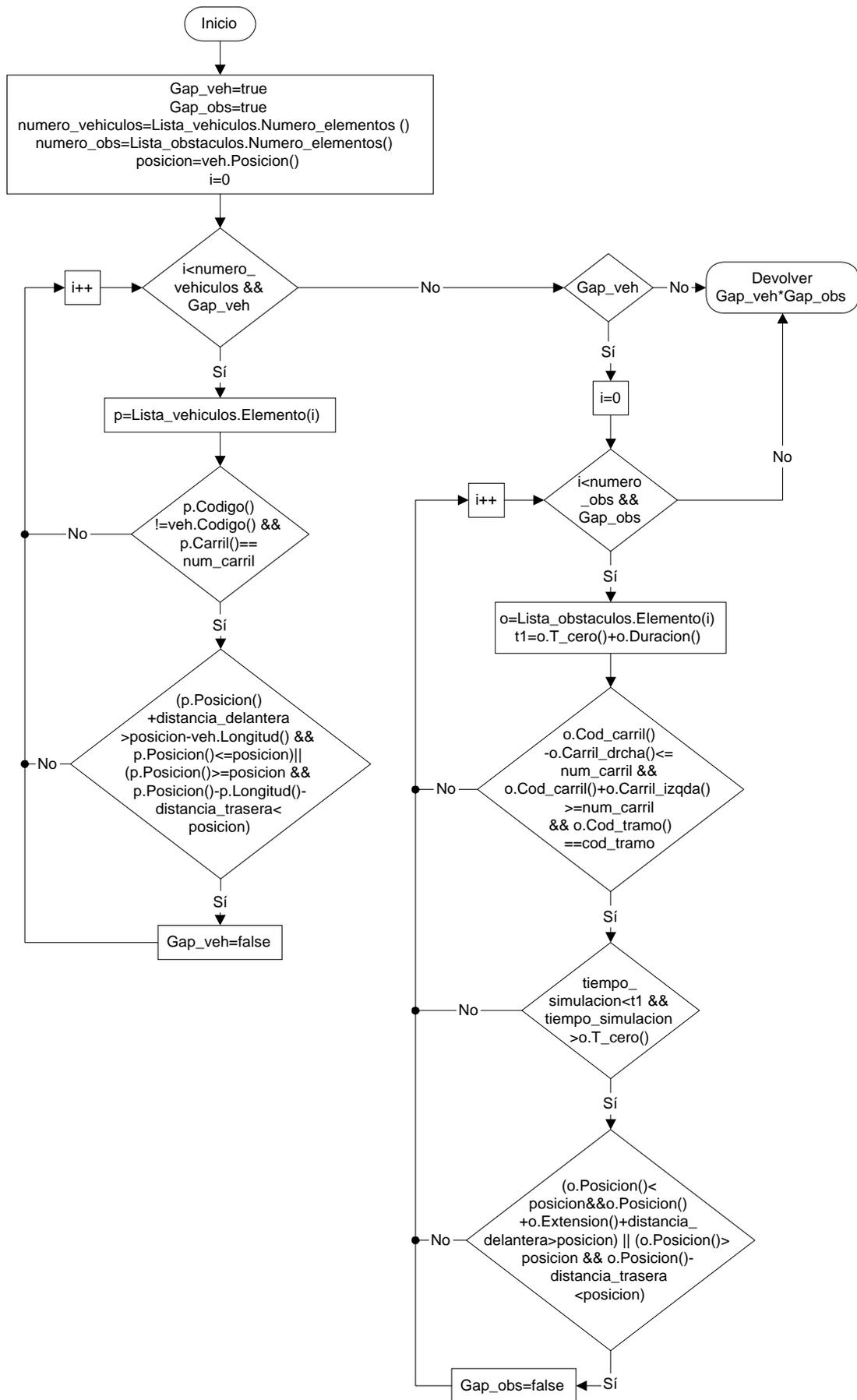


Figura 6.8. Aceptación de gap en cambio de carril para aparcamiento

Una vez aceptado este gap, el procedimiento para el cambio de carril del vehículo es totalmente análogo al que se sigue en cualquier otro cambio de carril de carril obligatorio, por lo que se analiza en el correspondiente apartado.

También cabe destacar, que cuando un vehículo aparca, se escriben en el fichero correspondiente sus estadísticas referentes al tiempo total en el sistema y en el tramo del vehículo, ya que durante un tiempo va a encontrarse inactivo. Además, se crea la estadística correspondiente a la posición de aparcamiento del vehículo, que sigue el diagrama de flujo mostrado en la figura 6.9.

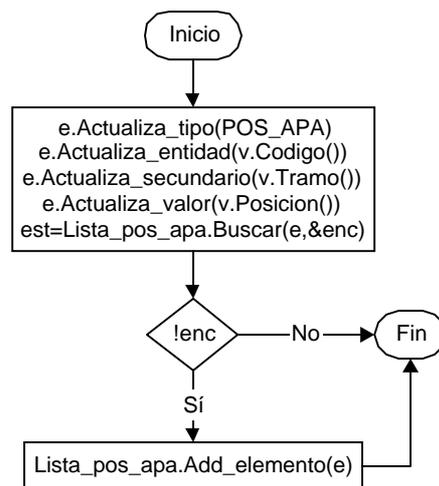


Figura 6.9. Estadísticas de posiciones de aparcamiento

Como se observa en la figura 6.9, el procedimiento es muy simple, simplemente consiste en buscar en la lista de estadísticas correspondiente si existe el elemento que se quiere añadir, el cual contiene los datos del vehículo, tramo y posición del aparcamiento. En el caso de que no se encuentre, se añade a la lista.

También cabe destacar respecto a este tipo de estadísticas que son escritas en el fichero correspondiente en el mismo momento en el que se produce el aparcamiento. La cabecera de dicho fichero se escribe sólo la primera vez que se accede a esta función.

Una vez visto este proceso de aparcamiento, lo único que resta por realizar para finalizar el bloque (1) de la figura 6.3, consiste en modificar la lista de tramos para contemplar las modificaciones que se han realizado en el tramo correspondiente.

6.2.2 Simulación de nodos

La simulación de los nodos pertenecientes al escenario se realiza mediante el bloque de funciones (2) indicado en la figura 6.3. En esta simulación de los nodos, lo que se hace es obtener la posición, velocidad y aceleración de los distintos vehículos que se encuentran en el nodo. A continuación se obtiene la distancia de influencia del vehículo. Para finalizar, se modifica la lista de vehículos con los nuevos datos calculados y se escriben los datos de los vehículos en los ficheros correspondientes.

No se va a entrar en la forma de realizar estas acciones, ya que no es materia de este proyecto. Sin embargo, baste lo expuesto anteriormente para hacerse una idea del funcionamiento general del simulador.

6.2.3 Actualización de la simulación

En este apartado se van a explicar las funciones que conforman el bloque (3) de la figura 6.3. La misión fundamental de este bloque consiste en realizar la actualización de la simulación, teniendo en cuenta las nuevas posiciones, velocidades y aceleraciones de los vehículos calculadas en los apartados 6.2.1 y 6.2.2.

La primera de las funciones que se ejecuta en dicho bloque (*Simular_vehiculos*) se emplea para realizar la simulación de los aparcamientos masivos que forman parte del escenario.

A continuación, se procede a la escritura de las estadísticas correspondientes al número de vehículos por carril en el fichero adecuado. Este proceso consiste simplemente en escribir en un fichero el código de cada uno de los tramos y carriles, seguido del número de vehículos que circulan por los mismos.

La función más importante de este bloque es la denominada *Actualiza_simulacion*, cuyo diagrama de flujo se muestra en la figura 6.10.

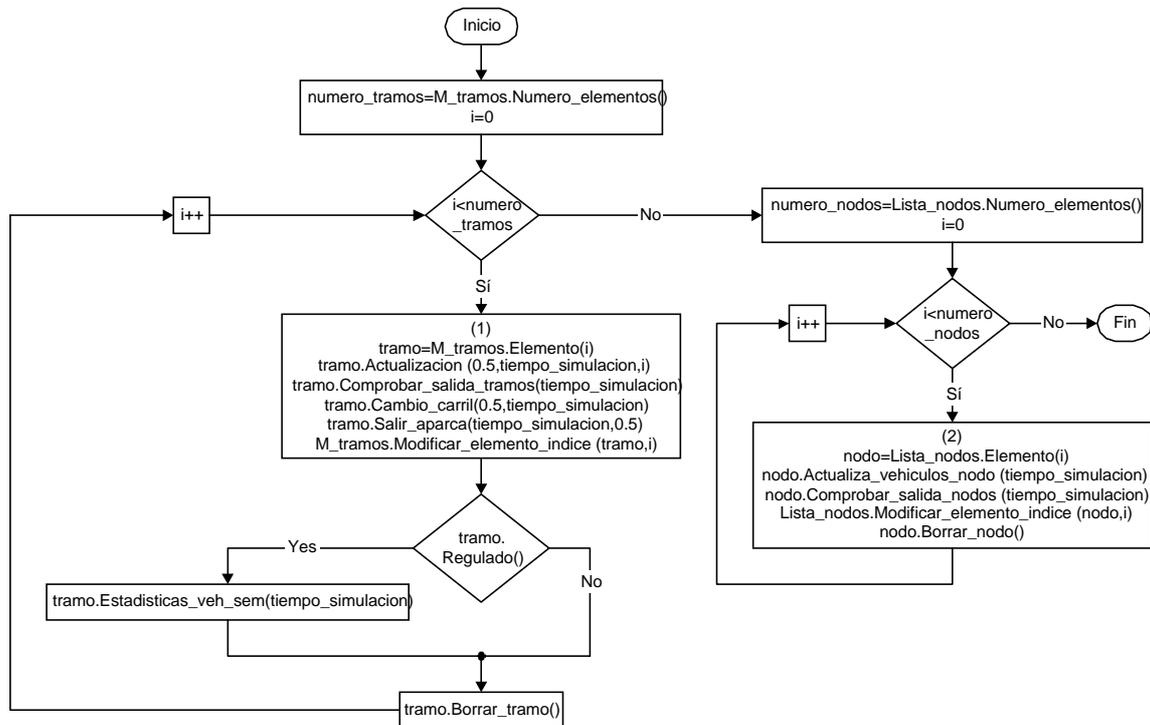


Figura 6.10. Actualiza simulación

En la figura 6.10 se aprecia que la actualización de la simulación se produce en primer lugar para los tramos de la red, mediante el conjunto de funciones del bloque (1), y posteriormente para los nodos, mediante el bloque (2) de funciones. En los siguientes apartados se van a analizar todas estas funciones.

6.2.3.1 Actualización de un tramo

En este apartado se estudia la primera de las funciones que tienen lugar en el bloque (1) de la figura 6.10. En dicha función se analizan todos los vehículos activos y que no estén cambiando de entidad que se encuentran circulando por el tramo. Los vehículos que cambian de entidad, se actualizan para que ya no lo estén, ya que ahora se encuentran en un tramo.

La primera actualización realizada, es pasar los parámetros de posición, velocidad y aceleración futuras a presentes. También se actualiza el atributo de la distancia recorrida por el vehículo, teniendo en cuenta la diferencia entre la posición futura y la presente, esta actualización es muy importante para las estadísticas.

A continuación, se obtiene la distancia de influencia del vehículo, para lo que es necesario conocer la deceleración máxima del vehículo y la velocidad que lleva el vehículo que le precede.

El siguiente paso consiste en comprobar si el vehículo debe entrar en un aparcamiento masivo. En el caso de que así sea, se produce una salida del vehículo del tramo y del carril correspondiente, ya que entra en el aparcamiento, por lo tanto se llama a las funciones de estadísticas de tiempo en el sistema y en el tramo vistas en las figuras 6.5 y 6.6.

Por último, queda un paso de gran importancia para este proyecto, como es la generación de nuevos vehículos en el tramo. Esta generación se produce siempre que la función de generación asociada al tramo sea distinta de 0, es decir, que se deban generar vehículos en el tramo. En ese caso se sigue el procedimiento explicado en el apartado 6.2.3.2.

6.2.3.2 Generar vehículos

En la figura 6.11 se muestra el procedimiento general empleado para generar nuevos vehículos en el escenario de simulación.

El primer paso a la hora de generar un vehículo es cargar el trapecio que se va a usar en la generación. En la figura 6.11 se ha supuesto que siempre se carga el primero de la lista de trapecios, aunque esto puede depender del tramo. La función estadística que va a modificar la forma del trapecio es la propia del tramo.

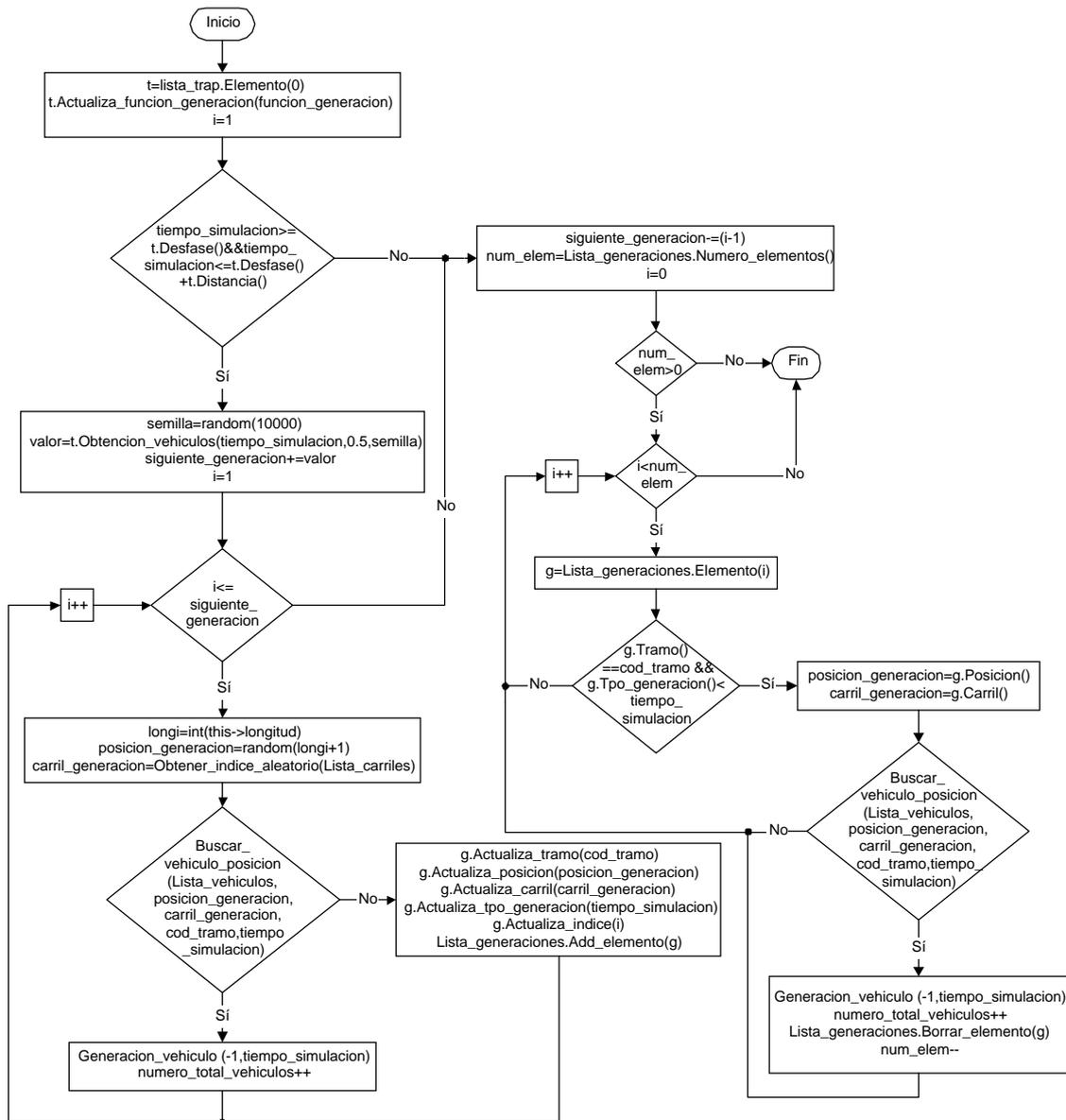


Figura 6.11. Procedimiento general de generación de vehículos

Obtención del número de vehículos a generar

Para que se generen nuevos vehículos, el instante de simulación debe estar comprendido entre los límites que marcan el comienzo y el final del trapecio, para que el número de vehículos sea distinto de cero. A continuación, se obtiene el número de vehículos que han de generarse y se suman a la parte fraccionaria que haya sobrante de generaciones anteriores. El número de vehículos a generar se obtiene mediante el procedimiento indicado en la figura 6.12.

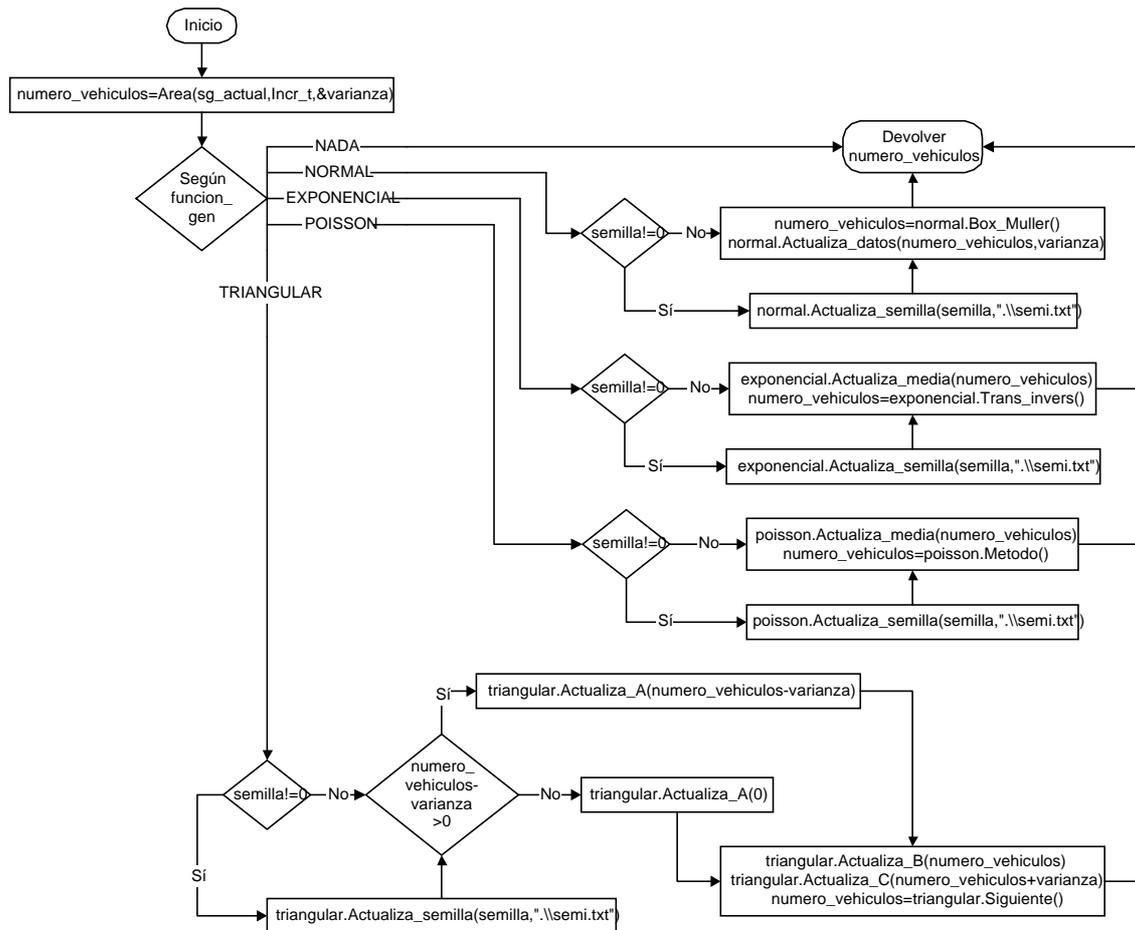


Figura 6.12. Obtención de vehículos

La obtención del número de vehículos que se deben generar comienza con el cálculo del área de la sección de trapezoide correspondiente al intervalo de tiempo a simular. Este cálculo se realiza según el diagrama de flujo de la figura 6.13, en el que se tiene en cuenta la zona del trapezoide en la que se encuentre el intervalo de tiempo de simulación actual. Tal y como se observa en dicha figura, el cálculo del área y la varianza en cada una de las secciones del trapezoide se corresponde con el procedimiento ya explicado en el apartado 3.4.1.

El cálculo de las alturas de los puntos del trapezoide pertenecientes a las distintas zonas, se realiza mediante los algoritmos representados en la figura 6.14. En dichas funciones, se hace referencia a los ángulos característicos del trapezoide vistos en el apartado 5.2.3.4.1.7.

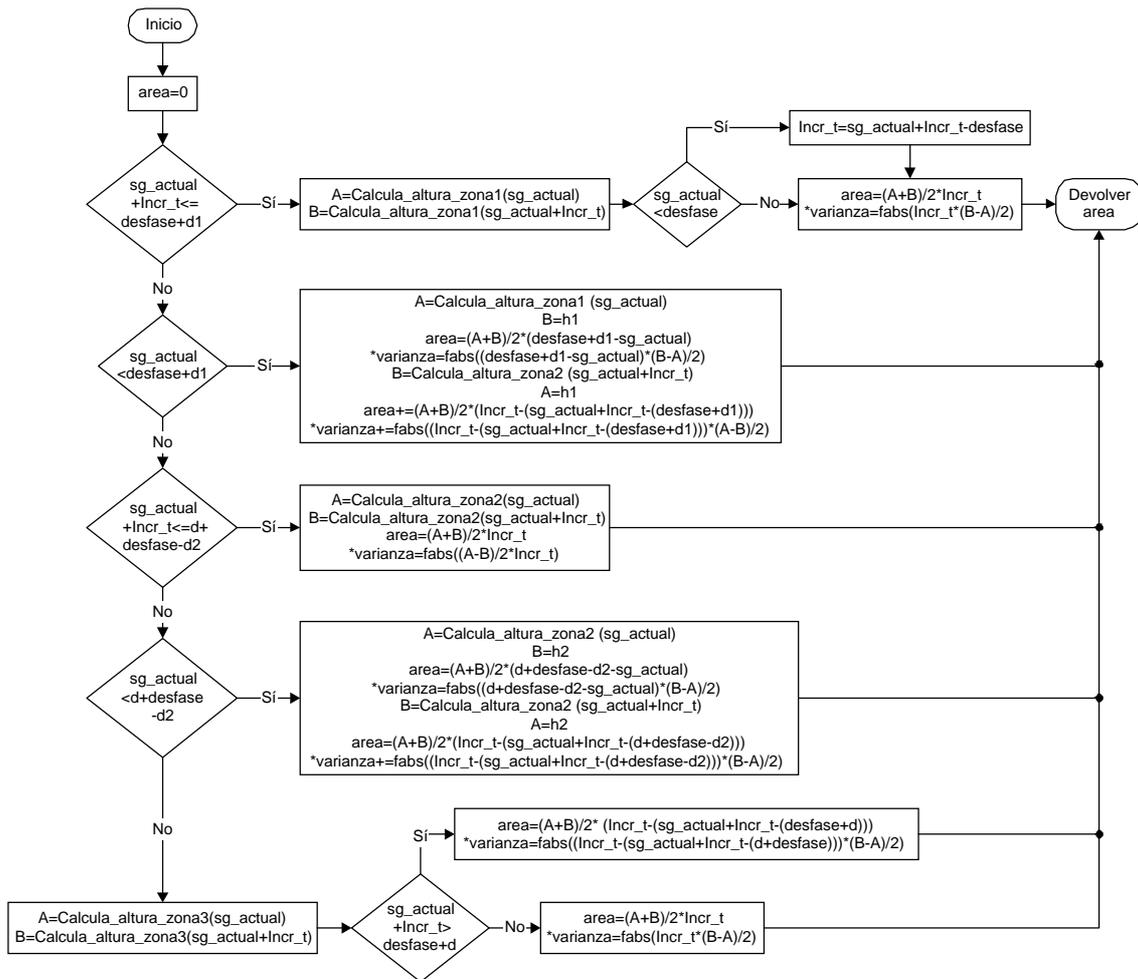


Figura 6.13. Cálculo del área de una sección del trapezoide

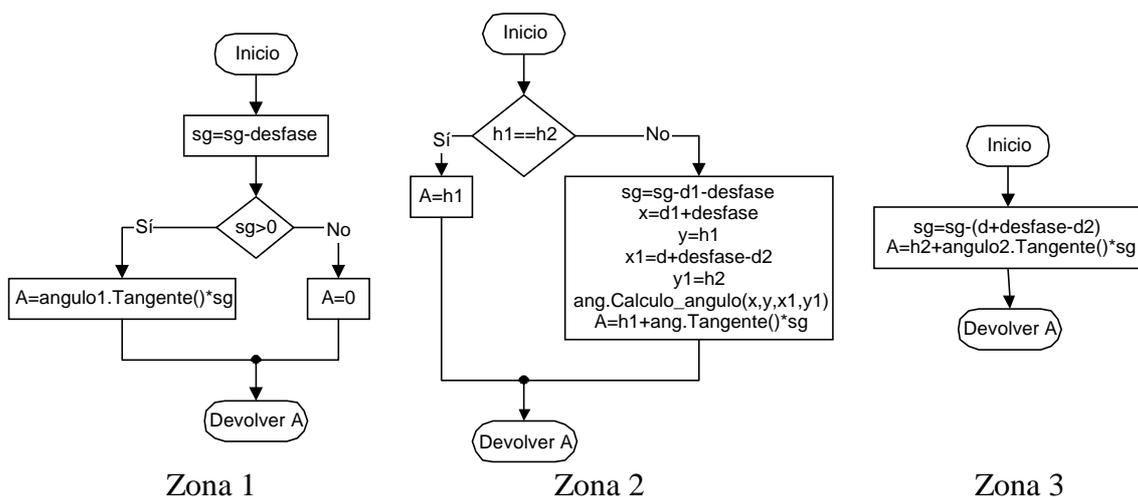


Figura 6.14. Cálculo de la altura de un punto según zona del trapezoide

Volviendo a la figura 6.12, una vez que se ha obtenido el área, dicho valor se afecta por la función estadística asociada al trapezoido. Si no tiene ninguna función asociada, el número de vehículos a generar es directamente el valor del área. En caso contrario, el área es empleada como valor medio de la función estadística correspondiente y en el caso de la normal, también se utiliza la varianza obtenida. Una

vez que se conocen dichos parámetros (media y varianza) se genera un número aleatorio según la distribución correspondiente, para lo que se hace uso de las funciones de generación de números pseudoaleatorios comentadas en el apartado 3.3.

En la generación de estos números pseudoaleatorios se emplea como semilla un valor que se pasa como parámetro a la función de obtención de vehículos, aunque también puede tomarse de un fichero de texto, o seguir usando la misma semilla anterior, en el caso de que se le pase una semilla de valor cero.

Posición de generación del vehículo

Volviendo sobre la figura 6.11, se observa que una vez conocido el número de vehículos que hay que generar se comienza la generación de cada uno de ellos. El primer parámetro de dicha generación que hay que conocer es la posición del tramo en la que se van a generar.

En el diagrama de flujo, se muestra que la posición de generación del vehículo se determina mediante un número aleatorio desde 0 hasta la longitud del tramo. Sin embargo, también existe la posibilidad de que existan unas posiciones fijas en las que se pueden generar vehículos. De hecho, ésta última opción es la elegida en las pruebas realizadas en el apartado 7.

El carril en el que se deben generar los vehículos también puede depender de la posición en la que se generen, o como se muestra en la figura 6.11, se puede elegir un carril cualquiera del tramo de forma aleatoria.

Para que el vehículo se pueda generar en la posición y carril deseados, es necesario que se cumplan ciertas distancias de seguridad con el resto de vehículos. El procedimiento de comprobación de dicha distancia se muestra en la figura 6.15.

Como se observa en dicha figura, se comienza analizando todos los vehículos que se encuentran en el mismo carril donde se quiere generar el nuevo vehículo. La generación es rechazada (o pospuesta), *Salida=true*, cuando se encuentra un vehículo a una distancia inferior a dos veces la longitud de un vehículo por delante de la posición de generación, o un vehículo a una distancia inferior a la distancia de seguridad por detrás de dicha posición de generación.

Una vez que se han analizado todos los vehículos y la generación es realizable, se pasan a analizar los obstáculos. La generación será rechazada cuando en una distancia comprendida entre 3 metros por delante y por detrás de la posición de generación se encuentre un obstáculo que esté presente en ese instante y que obstruya el carril deseado para introducir el nuevo vehículo.

En el caso de que la generación del vehículo sea rechazada por imposibilidad de realizarla en esa posición, es almacenada en una lista de generaciones pendientes para poder llevarla a cabo en el momento que sea posible.

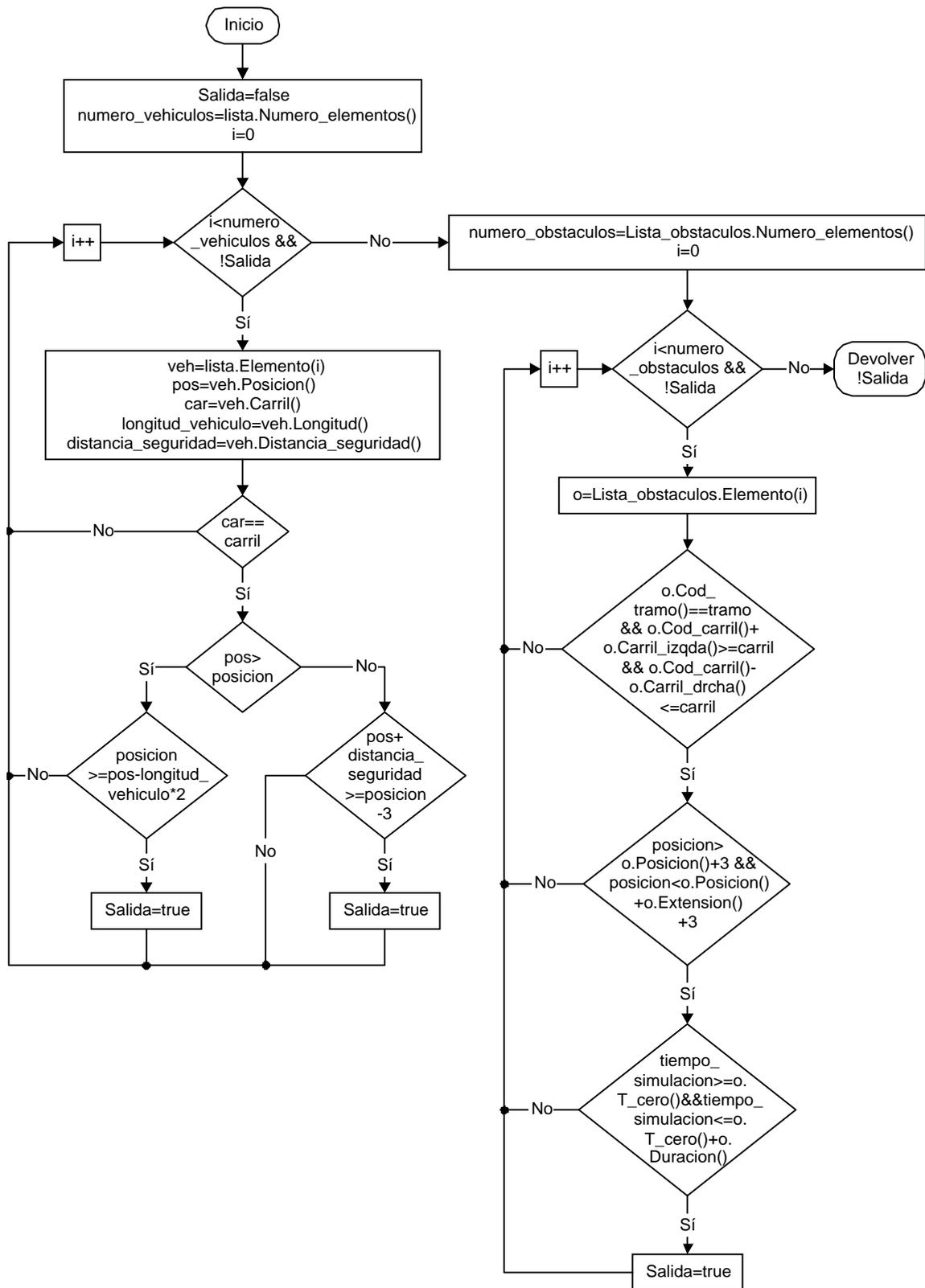


Figura 6.15. Comprobación de posición de generación de vehículos

Generación del vehículo

Cuando la posición en la que se debe generar el nuevo vehículo ha sido aceptada, se procede a la generación propiamente dicha. En primer lugar, se le asigna aleatoriamente un tipo de vehículo y de conductor de forma que se cumplan los porcentajes deseados para cada uno de estos tipos.

A continuación se crea el vehículo en el tramo y posición deseados con una velocidad inicial nula. El código del nuevo vehículo es el siguiente al último vehículo que fue creado. El instante de creación del vehículo, lógicamente se actualiza al instante de simulación en el que se esté. El resto de atributos del vehículo son inicializados a sus valores por defecto. Además se actualizan las influencias que han sido modificadas por la introducción del nuevo vehículo en el tramo. Asimismo, se considera el número de vehículos que existen ahora en el carril para determinar el posible déficit o superávit de vehículos que pueda haber en el carril.

Para la decisión del tramo de destino del vehículo, se busca aleatoriamente un giro que conecte el tramo actual con otro tramo, pero de forma que se mantengan los porcentajes del número de vehículos que deben llevar a cabo cada uno de los giros.

Una vez generado el vehículo con los parámetros anteriores se añade en la correspondiente lista de vehículos del tramo y se llevan a cabo los procedimientos asociados a las estadísticas de generación.

La primera de estas estadísticas es la asociada con el número de vehículos que son generados en cada tramo en cada instante. El diagrama de flujo del proceso necesario para controlar esta estadística se muestra en la figura 6.16, en la que se observa que el proceso consiste en buscar el elemento estadístico correspondiente al tramo y aumentar en una unidad su atributo *suma* y *valor*.

El *valor* de dicho elemento se inicializa a 0 en cada instante de simulación, por lo que contiene el número de vehículos generados en el tramo en ese mismo instante. El atributo *suma* sólo se pone a cero al comienzo de la simulación, por lo que contiene el número total de vehículos generados en ese tramo.

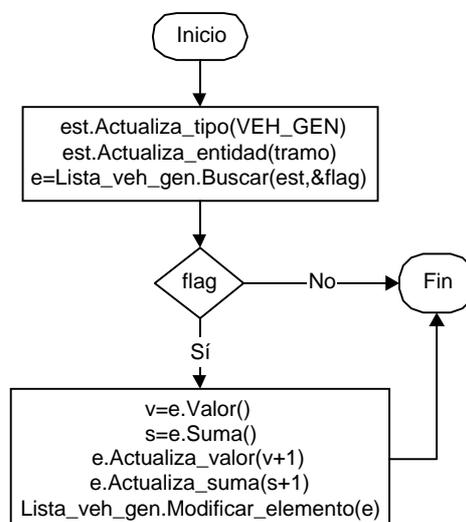


Figura 6.16. Estadísticas de generación de vehículos

La siguiente estadística se corresponde con las posiciones de generación de los vehículos. Su diagrama de flujo se muestra en la figura 6.17. Este algoritmo lo que hace es buscar el elemento estadístico de posiciones de generación de ese tramo, si no existe, lo crea y actualiza sus valores con la posición de generación, si ya existe, realiza las actualizaciones necesarias para poder conocer en cada instante la posición media del tramo en la que se generan los vehículos.

Las estadísticas de posiciones de generación se escriben en su fichero de texto correspondiente cada vez que se genera un vehículo. La cabecera de dicho fichero se escribe la primera vez que se crea un vehículo, mientras que las posiciones medias de generación se escriben una vez que finaliza la simulación.

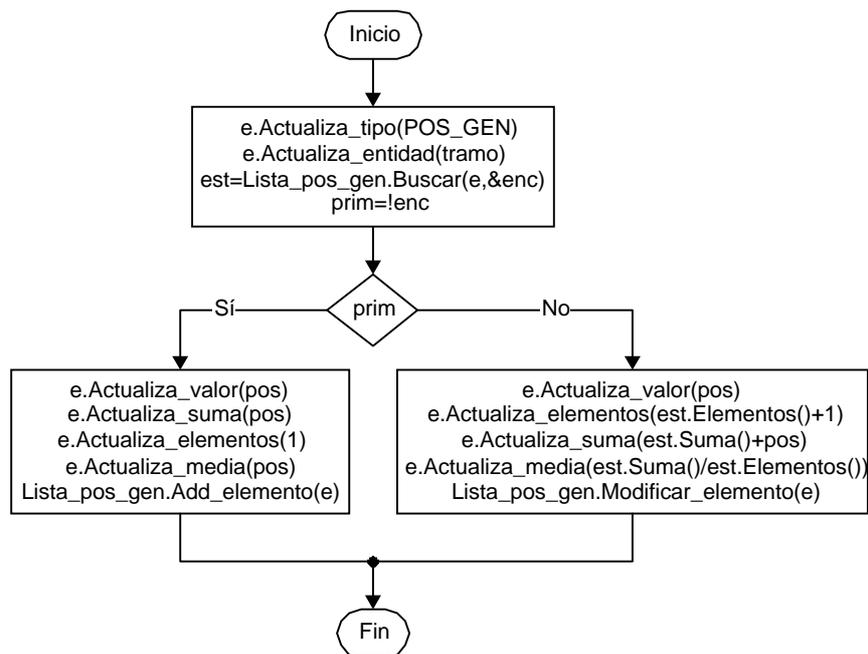


Figura 6.17. Estadísticas de posición de generación de vehículos.

Generación de vehículos pendientes

Para finalizar con el procedimiento de generación de vehículos, resta por analizar la generación de los vehículos que no han podido ser generados en su momento, debido a la incompatibilidad de la posición de generación con otros vehículos ya existentes (parte derecha del diagrama de flujo de la figura 6.11).

Para llevarla a cabo, se busca en la lista de generaciones pendientes, aquellas que se refieren al tramo en cuestión y que se han añadido a la lista en un instante de tiempo anterior al actual. Una vez localizadas, se comprueba si pueden ser llevadas a cabo en ese instante, es decir, se vuelve a comprobar que no existan ni vehículos ni obstáculos que impidan dicha generación.

Aquellas generaciones que siguen sin poder realizarse, continúan en la lista de generaciones pendientes, mientras que las que pueden llevarse a cabo, se eliminan de la lista y se genera el vehículo correspondiente.

6.2.3.3 Comprobación de salidas de tramos

La siguiente función que forma parte del bloque (1) de la figura 6.10 hace referencia a la comprobación de si hay vehículos que hayan terminado su recorrido a lo largo del tramo y deban salir del mismo.

Para ello, se comprueban todos los vehículos del tramo para buscar aquellos cuya posición ha sobrepasado la distancia total del tramo. Aquellos vehículos que cumplan esta condición son eliminados del tramo, por lo que se actualiza la influencia del vehículo que le sucedía en su carril y se marca al propio vehículo como que está cambiando de entidad. Además, hay que quitar una unidad al número de vehículos que circulan por el tramo y carril por el que lo hacía dicho vehículo.

Además, se llevan a cabo las estadísticas relativas al tiempo en el tramo del vehículo, y si no tiene más tramos destino, también la estadística de tiempo en el sistema. Para la estadística correspondiente a la velocidad media, se tiene en cuenta la posición final del vehículo en el tramo.

En el caso de que el vehículo deba dirigirse a un nuevo tramo, hay que calcular cuál es el giro que debe llevar a cabo para llegar hasta su nuevo tramo.

6.2.3.4 Cambios de carril en el tramo

Continuando con las funciones del bloque (1) de la figura 6.11, se llega al procedimiento que realiza el cambio de carril de los vehículos del tramo. Este procedimiento va a ser estudiado detalladamente y su diagrama de flujo general se muestra en la figura 6.18.

En este procedimiento se analizan todos los vehículos que circulan por el tramo. La primera comprobación consiste en si el vehículo está realizando ya una maniobra de cambio de carril o no. Si no la está haciendo, hay que ver si es deseable que la inicie, en caso de que ya esté cambiando de carril se comprueba si ya es momento de que la acabe (parte derecha de la figura 6.18).

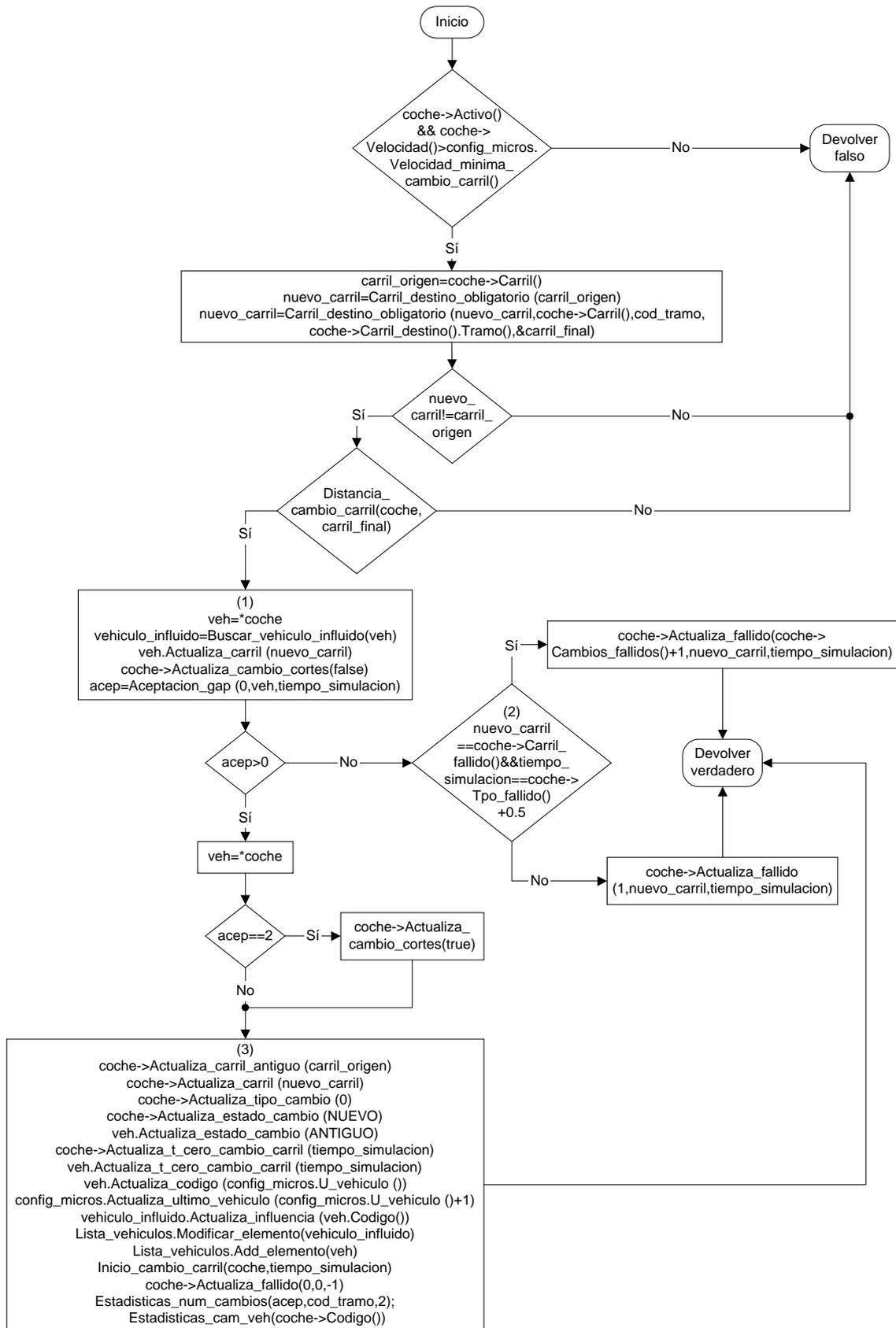


Figura 6.19. Cambio de carril obligatorio

Una vez que esto ha sido comprobado, se calcula el carril de destino obligatorio hacia el que el vehículo puede necesitar desplazarse, a continuación se comprueba si existe una distancia suficiente para poder realizar la maniobra con seguridad. En el caso de que todos estos requisitos se cumplan, se inicia el cambio de carril. A continuación se muestran detalladamente todos estos pasos.

Carril de destino para mantener porcentajes en los carriles

El cálculo del carril de destino de un vehículo en un cambio obligatorio se descompone en dos pasos. En este punto se va a estudiar el primero de ellos, consistente en la obtención de un nuevo carril cuando el número de vehículos que circulan por el carril actual del vehículo es superior al debido. Este proceso se muestra en la figura 6.20.

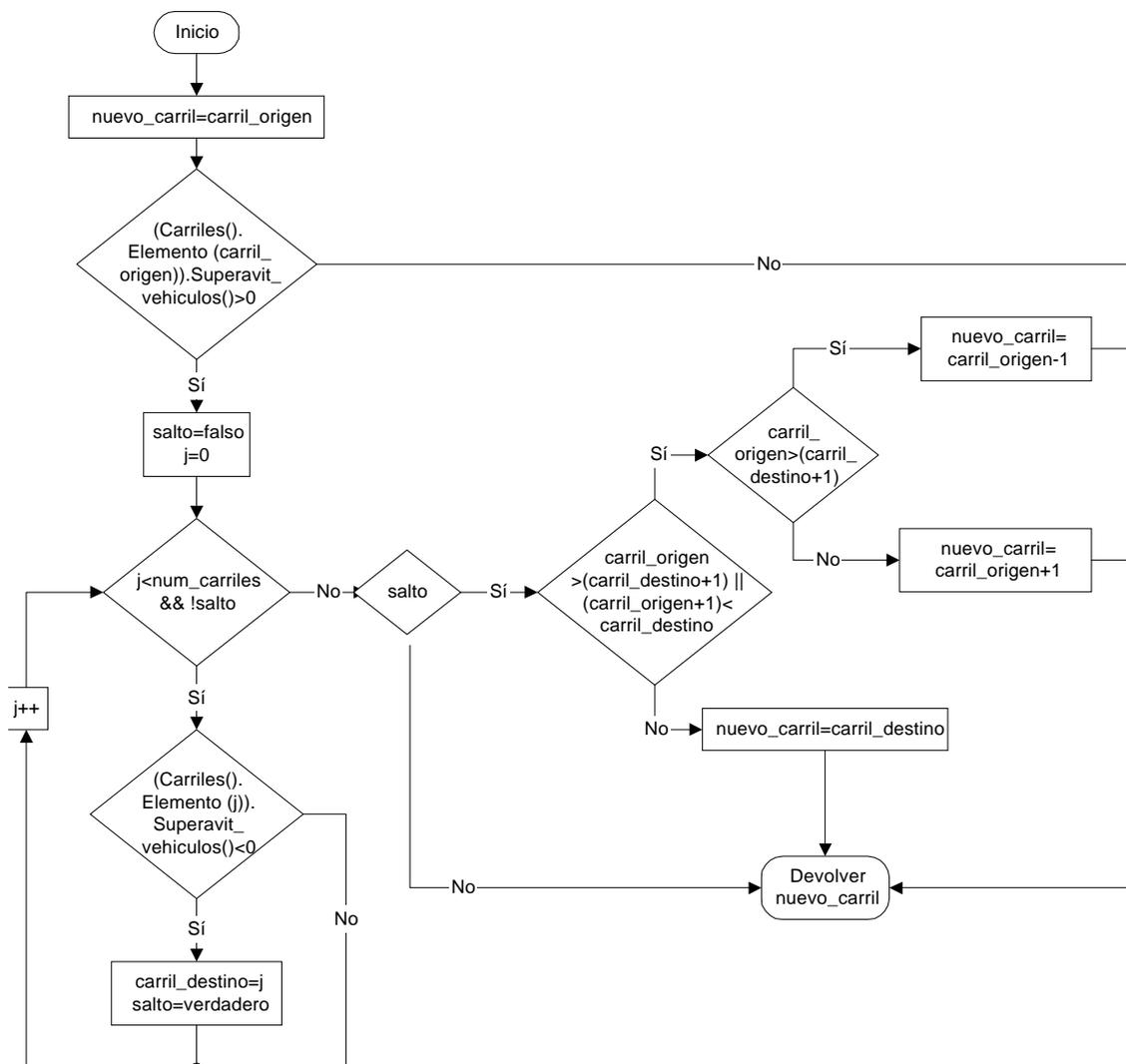


Figura 6.20. Carril de destino obligatorio para mantenimiento de porcentajes

El procedimiento comienza comprobando que en el carril actual del vehículo existe superávit de vehículos, es decir, que existen más de los que debe haber para mantener el porcentaje de vehículos deseado en el carril. En el caso de que no haya superávit, el vehículo puede continuar por el carril actual.

En el caso de que exista superávit de vehículos en el carril, se busca entre los demás carriles del tramo alguno en el que haya déficit de vehículos. Una vez encontrado, éste es el carril de destino hacia el que se debe desplazar el vehículo.

Cuando el carril actual del vehículo y el deseado están separados por algún otro carril intermedio, el carril de destino será el adyacente al actual en la dirección donde se encuentre el carril deseado.

Carril de destino obligatorio para realización del giro deseado

La segunda parte del procedimiento de búsqueda del carril de destino en un cambio obligatorio, consiste en buscar aquel carril del tramo que permita que el vehículo pueda realizar el giro que desee al llegar a la finalización del tramo. Este procedimiento de búsqueda se muestra en la figura 6.21 y necesita conocer el número del carril que devuelve la primera parte del proceso.

En la primera estructura *if*, lo que se comprueba es si existe el nodo de destino del vehículo, ya que de no existir, el carril deseado es directamente el obtenido para el mantenimiento de los porcentajes, ya que el vehículo no necesita ir a ningún sitio en concreto.

En el caso de que el vehículo sí deba ir a algún tramo en concreto, se comprueba si circulando por el carril que se ha obtenido en el paso anterior es posible incorporarse al movimiento que le permite alcanzar el tramo deseado. Si es así, el carril obtenido para mantener el superávit será el carril deseado definitivo.

Si no fuera así, es decir, se tiene que desplazar a un carril diferente para llegar al tramo deseado, se empieza a recorrer todos los carriles del tramo, comprobando para cada uno de ellos si es posible llegar al destino a partir de él. El carril de destino deseado será el más próximo al actual que le permita realizar el giro hasta el nuevo tramo por el que debe circular.

Cuando se ha obtenido un carril diferente al que se obtuvo en el primer paso (*salto=true* en la figura 6.21), se comprueba si el nuevo carril es adyacente al actual, y en el caso de que no sea así, el nuevo carril definitivo será el adyacente al actual en la dirección donde está el carril deseado.

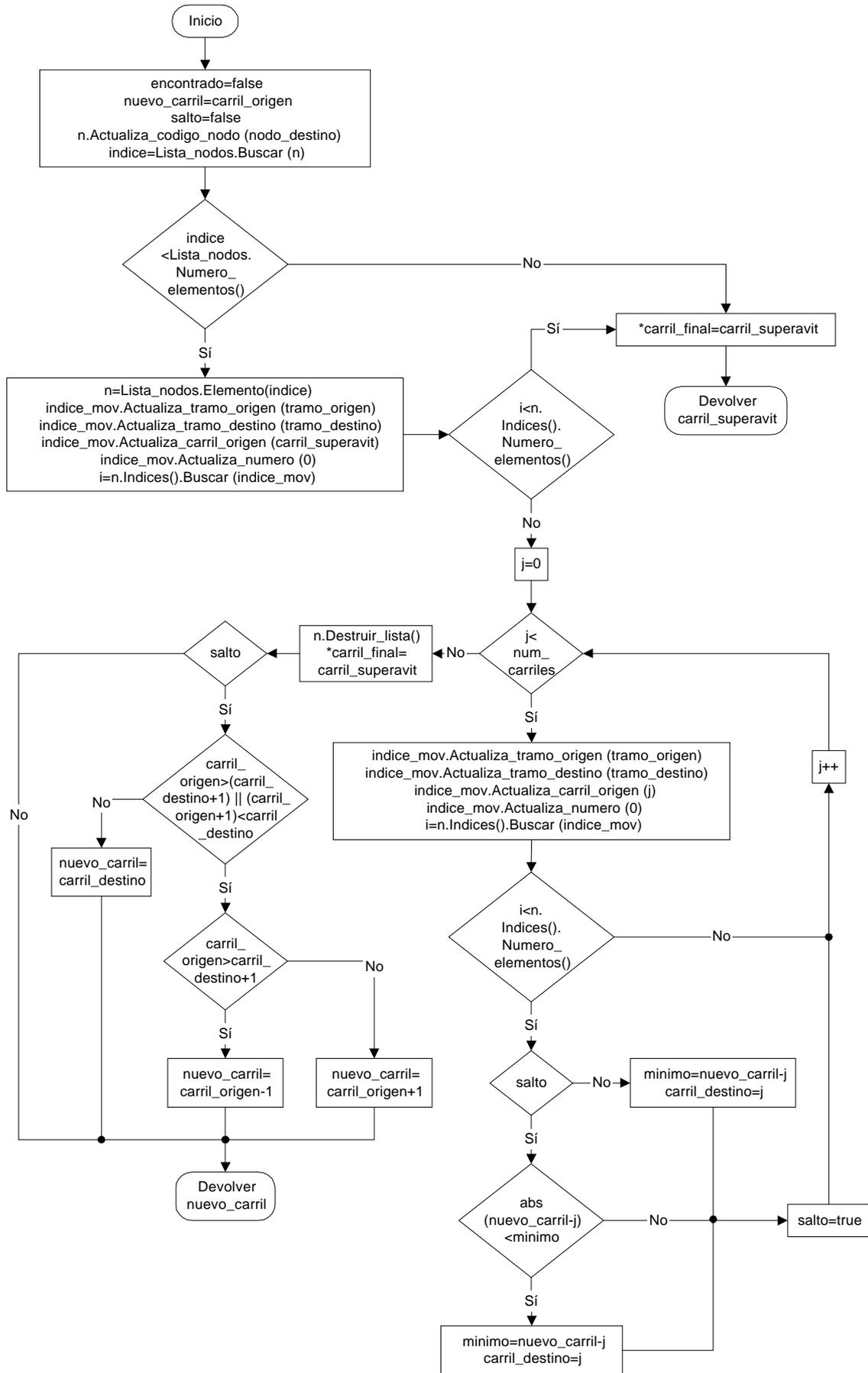


Figura 6.21. Carril de destino obligatorio para giro deseado

Distancia de cambio de carril obligatorio

Siguiendo el diagrama de flujo de la figura 6.19, se comprueba que si el nuevo carril obtenido en los procesos anteriores es diferente del carril actual de vehículo, es decir, si hay que realizar una maniobra de cambio de carril obligatorio, el siguiente paso consiste en verificar si el vehículo se encuentra a la distancia mínima necesaria del fin del tramo para realizar este tipo de cambio. Esta comprobación se lleva a cabo siguiendo el diagrama de flujo de la figura 6.22.

Esta distancia a partir de la cual se pueden realizar cambios de carril obligatorios depende del carril final hacia el que se tiene que desplazar el vehículo, de forma que si el vehículo debe hacer más de un cambio de carril para llegar hasta su destino, la distancia de cambio de carril obligatorio aumenta.

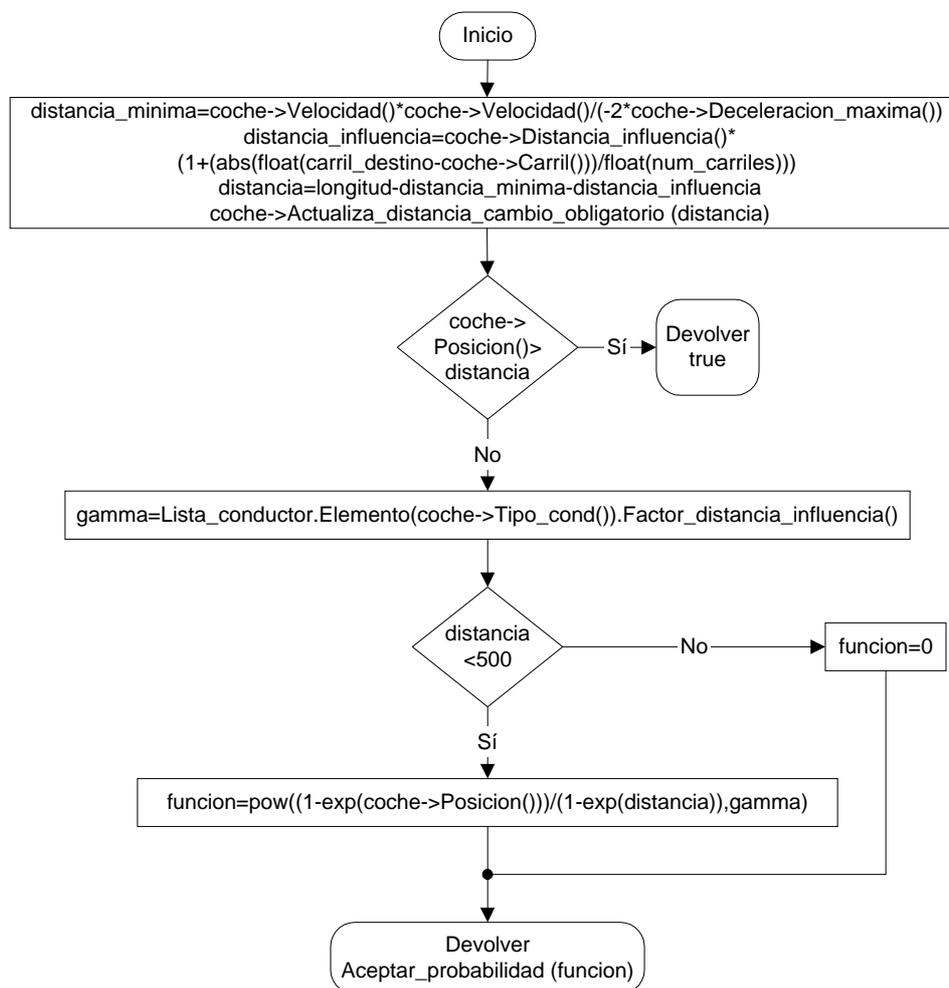


Figura 6.22. Distancia de cambio de carril obligatorio

En el algoritmo de comprobación de la distancia de cambio de carril obligatorio, lo primero es implementar la expresión vista para esta distancia en el apartado 2.4.2.1, lo cual se realiza en el primer bloque de la figura 6.22. Una vez obtenida esta distancia, se comprueba si el vehículo está a una distancia inferior a ella del final del tramo. Si es así, la distancia de cambio de carril obligatorio es aceptada.

En el caso de que no sea así, existe una segunda oportunidad para la realización del cambio de carril obligatorio siempre que la distancia desde el vehículo hasta el final del tramo sea inferior a 500 metros. Esta segunda oportunidad consiste en una cierta función de probabilidad dependiente de la posición del vehículo y de la distancia teórica de cambio de carril obligatorio.

Aceptación de gap

Cuando la distancia de cambio de carril obligatorio se acepta, el último paso previo al inicio de la maniobra de cambio de carril es la comprobación de la existencia de una distancia suficiente para realizar el cambio con seguridad. Esta comprobación se realiza en el bloque (1) de la figura 6.19.

Lo primero es hacer una copia del vehículo que desea cambiar de carril y situarla en el nuevo carril, lo que va a servir para comprobar cuál sería la situación en el nuevo carril en el caso de que finalmente el vehículo realizara el cambio de carril. A continuación se procede a la llamada de una función que se encarga de aceptar o no el gap disponible para la realización del cambio de carril. El diagrama de flujo de esta función se muestra en la figura 6.23.

Para comenzar, hay que determinar si es necesario estudiar el gap trasero (*anterior*), el delantero (*posterior*), ninguno o ambos. Para ello, se busca en primer lugar el vehículo al que va a influir en el nuevo carril (vehículo sucesor). En el caso de que dicho vehículo exista, hay que estudiar la distancia entre el vehículo que quiere cambiar de carril y su vehículo sucesor (*veh_n0*). Si no se encuentra ningún vehículo sucesor en el tramo, hay que buscarlo en el movimiento anterior, y si lo hubiera, éste sería el vehículo sucesor *veh_n0*.

A continuación, se determina la influencia a la que se ve sometido el vehículo en el nuevo carril. Si dicha influencia es un vehículo, hay que estudiar la distancia hasta dicho vehículo precedente (*veh_n2*). Si la influencia que se encuentra no es la de un vehículo, se pasa a buscar el vehículo más cercano en los movimientos que siguen al carril. En el caso de que se encuentre, éste es el vehículo *veh_n2*, y su posición es la que resulta de sumar la longitud total del tramo con su posición dentro del movimiento.

En el caso de que se encuentre un vehículo precedente pero no uno sucesor, sólo se estudia el gap delantero. Si sólo se encuentra el vehículo sucesor, se estudia el gap trasero. Si se encuentran tanto el vehículo precedente como el sucesor se estudian los dos gaps, y si no se encuentra ninguno de los dos vehículos, el gap es inmediatamente aceptado, ya que dispone de todo el espacio necesario para realizar el cambio de carril.

Antes de ver en qué casos son aceptadas las distancias existentes con los vehículos para poder realizar el cambio de carril, cabe decir que como se aprecia en la figura 6.23, cuando la influencia que encuentra el vehículo por detrás en el nuevo carril es un obstáculo que aún no ha superado totalmente, no se puede realizar el cambio de carril. Así mismo, cuando el obstáculo se encuentra por delante del vehículo en el nuevo carril, debe estar a una distancia al menos igual a la distancia de seguridad para que pueda realizarse la maniobra.

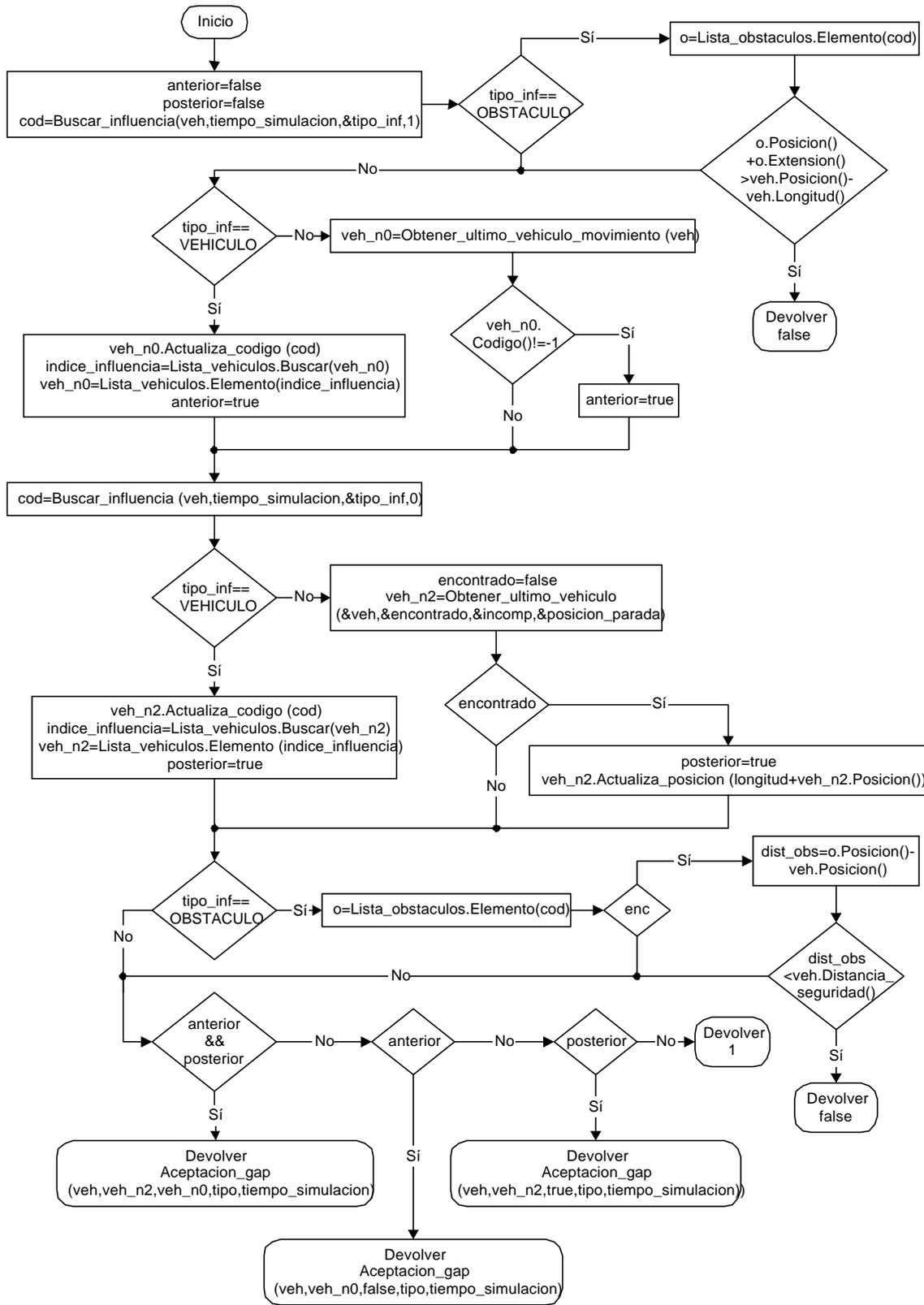


Figura 6.23. Proceso de aceptación del gap

En la figura 6.24 se muestra el algoritmo empleado para determinar si el gap existente es suficiente o no. En dicho algoritmo se emplea un parámetro de carácter booleano llamado *delantero*, que toma el valor verdadero cuando se pretende estudiar el gap delantero y el valor falso cuando se estudia el gap trasero. En el caso de que se deseen estudiar ambos gaps se hacen dos llamadas a este algoritmo, se comienza estudiando el gap delantero y en el caso de que se acepte se pasa a estudiar el trasero.

Además, también se emplea el parámetro *tipo*, que indica el tipo de cambio de carril para el que se quiere comprobar el gap. En los cambios de carril obligatorios, dicho parámetro toma el valor 0, mientras que en los cambios de carril por mejora, su valor es igual a 1.

El comienzo de la comprobación del gap consiste en determinar si se trata de un gap delantero o trasero, para actuar de una forma u otra. En el caso del gap delantero se determina la distancia entre el vehículo que desea cambiar de carril y su precedente, si dicha distancia es inferior a la distancia de seguridad del vehículo que quiere cambiar de carril, el gap es inmediatamente rechazado.

Si dicha distancia es superior a la distancia de seguridad y también mayor que la distancia de seguridad, el vehículo puede cambiar de carril con total seguridad, por lo que el gap es aceptado.

Cuando la distancia entre ambos vehículos se encuentra entre los límites anteriormente mencionados, es necesario recurrir a una función de probabilidad que depende de un factor del propio vehículo, de las distancias de seguridad e influencia, y del gap existente. La expresión de esta función de probabilidad es diferente para el caso de un cambio de carril obligatorio o por mejora, y sus expresiones fueron vistas en los apartados 2.4.2.3.2 y 2.4.3.3 respectivamente, y cuya implementación en el simulador se observa en la figura 6.24. En el caso de que dicha probabilidad sea aceptada, el gap también lo será, y el cambio podrá llevarse a efecto.

En el caso del gap trasero, la distancia que hay que estudiar es la existente entre el vehículo que desea cambiar de carril y su sucesor en el carril. Al igual que en el caso del gap delantero, si la distancia entre los vehículos es inferior a la distancia de seguridad, el gap es rechazado, y si es superior a la distancia de influencia, el gap se acepta de inmediato.

Cuando la distancia se encuentra entre la de seguridad y la de influencia, se sigue un proceso idéntico al del gap delantero, aunque con distintas expresiones, como se observa en la figura 6.24. Además, para que la función de probabilidad tome un valor distinto de cero, es necesario que los parámetros $z1$ y $z2$ sean mayores que cero.

Si este gap trasero es rechazado, aún existe una posibilidad de que se pueda realizar el cambio de carril. Para ello, es necesario que se acepte el denominado *gap de cortesía*, que depende no sólo de la distancia existente entre el vehículo y su sucesor, sino también del número de intentos que ha hecho el vehículo para realizar ese cambio de carril y de la predisposición del vehículo sucesor para dejar espacio al vehículo para que lo realice.

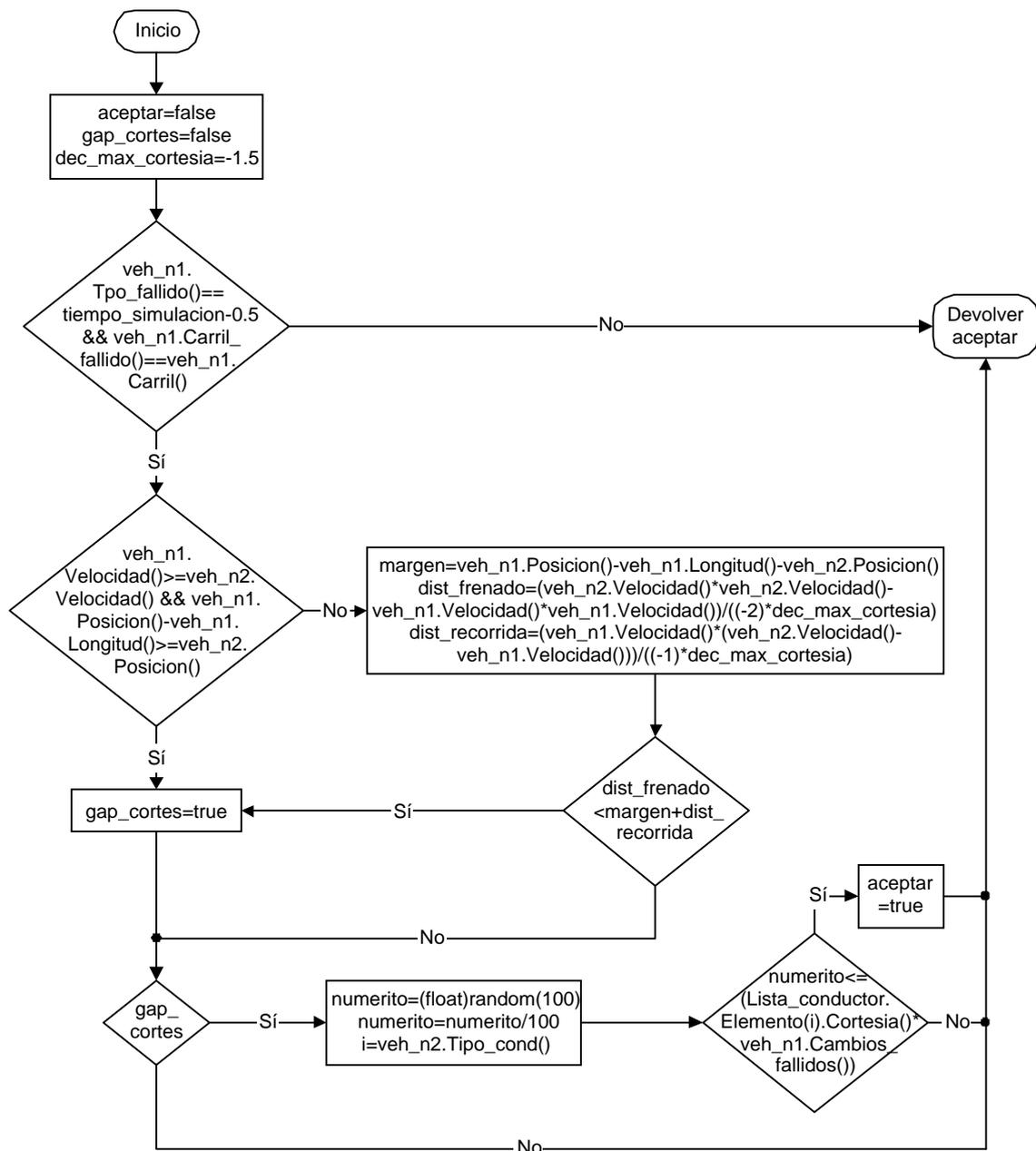


Figura 6.25. Aceptación del gap de cortesía

Para poder llevar a cabo un cambio de carril por cortesía, es necesario que ese cambio se haya intentado ya con anterioridad y no se haya podido realizar, de ahí viene el hecho de que la primera comprobación sea que el último intento de cambio fallido haya sido en el instante de tiempo anterior y hacia el mismo carril donde se quiere cambiar ahora, en caso contrario el gap de cortesía es rechazado.

A continuación se comprueba si la distancia entre el vehículo y su sucesor es suficiente (*gap_cortes* en la figura 6.25). Si el vehículo que desea cambiar de carril circula a una velocidad superior a la que lo hace su sucesor, y se encuentra completamente por delante de él, el *gap_cortes* es aceptado. En caso contrario, es necesario comprobar si en la distancia existente entre ambos vehículos, es posible que el vehículo sucesor frene hasta alcanzar la velocidad del que desea cambiar de carril.

El último paso antes de aceptar que se puede realizar un cambio de este tipo, consiste en aceptar una función de probabilidad determinada por el producto del número de intentos fallidos de realizar el cambio por parte del vehículo, por un parámetro que determina la probabilidad de que el vehículo sucesor permita realizar el cambio de carril, esta probabilidad depende del tipo de conductor que dirija al vehículo sucesor.

Si esta función de probabilidad toma un valor inferior a un número generado aleatoriamente, el gap de cortesía es aceptado y, por lo tanto, se puede efectuar el cambio de carril.

Actualización de cambios fallidos

Volviendo sobre la figura 6.19, se observa que cuando el gap es rechazado, hay que actualizar éste último cambio como el último que se ha intentado pero no se ha podido realizar.

Para ello se comprueba la condición (2) de dicha figura, en la que se mira si el último intento fallido que se tiene registrado para ese vehículo se corresponde con el mismo carril hacia el que se quiere desplazar ahora y en el instante de tiempo inmediatamente anterior. Si es así, se incrementa en una unidad el número de intentos fallidos, si no lo es, se actualiza el último cambio fallido al carril deseado y tiempo de simulación actuales, indicando que es el primer intento fallido.

Actualización de parámetros para inicio de cambio de carril

Antes de llevar a cabo el proceso de inicio de cambio de carril, es necesario llevar a cabo la actualización de una serie de parámetros del vehículo que va a cambiar de carril, así como la creación de un nuevo vehículo.

En primer lugar, se realiza una copia exacta del vehículo (*veh* en la figura 6.19) que desea cambiar de carril, y si el cambio se lleva a cabo gracias a la cortesía del vehículo precedente, es necesario marcar el atributo *gap_cortes* del vehículo como cierto.

A continuación, se entra en el bloque de acciones indicado como (3) en la figura 6.19. Para comenzar, el carril por el que circulaba el vehículo antes del cambio, pasa a ser su *carril_antiguo*, siendo su nuevo carril aquel hacia el que cambia. El tipo de cambio se marca como obligatorio y el estado del cambio es *NUEVO* para el vehículo que cambia de carril y *ANTIGUO* para la copia del vehículo que permanece en el carril antiguo. El tiempo de inicio de la maniobra de cambio de carril de ambos vehículos es el instante de simulación actual. El código del vehículo ficticio (el que se deja en el carril antiguo) es el valor que marque el último vehículo de la configuración microscópica, incrementando dicho valor en una unidad. Además, se actualiza el último cambio fallido para que se sepa que el último intento de cambio ha sido exitoso.

Para finalizar, hay que buscar el vehículo que estaba siendo influido por el vehículo que ha cambiado de carril y que ahora se ve influido por el vehículo ficticio. El vehículo ficticio es añadido a la lista y el que se ve influido por él, es modificado en la misma.

Una vez terminadas las actualizaciones anteriores, y continuando con el bloque (3) de la figura 6.19, se lleva a cabo el proceso de inicio del cambio de carril, consistente en una serie de actualizaciones que se comentan a continuación:

- Búsqueda de la nueva influencia del vehículo en el nuevo carril y actualización a ella.
- Actualización del nuevo tipo de influencia.
- Actualización de la influencia del vehículo que antes se veía influido por la nueva influencia del vehículo, y que ahora se ve influido por el propio vehículo.
- Actualización del número de vehículos que circulan por los carriles del tramo, considerando este último cambio.
- Modificación de la lista de vehículos del tramo, para considerar estas últimas modificaciones.

Estadísticas de cambios de carril

Para finalizar con el cambio de carril obligatorio, sólo resta por comentar las estadísticas que se actualizan cuando finaliza con éxito un cambio de carril de este tipo, correspondientes a las últimas instrucciones del bloque (3) de la figura 6.19.

La primera de las estadísticas se corresponde con el número de cambios de carril de cada tipo que se llevan a cabo. El diagrama de flujo empleado para contabilizar estas estadísticas se muestra en la figura 6.26, donde el parámetro *secund* vale 1 para los cambios de carril por mejora y 2 para los cambios obligatorios, además se emplea el valor de *acep* devuelto por la función de aceptación de gap.

El procedimiento de esta estadística consiste en crear un elemento estadístico del tipo *CAM_CAR* cuyo atributo *secundario* tome el valor 1, 2 o 3, según el cambio sea por mejora, obligatorio o por cortesía, respectivamente. El atributo *entidad* toma el valor del código del tramo.

A continuación se busca en la lista correspondiente ese elemento estadístico, si no existe, se añade a la lista con el *valor* 1, y si existe, se incrementa dicho atributo en una unidad. De esta forma, se tienen almacenados el número de cambios de cada tipo en cada uno de los tramos, para poder ser escritos en un fichero de texto al finalizar la simulación.

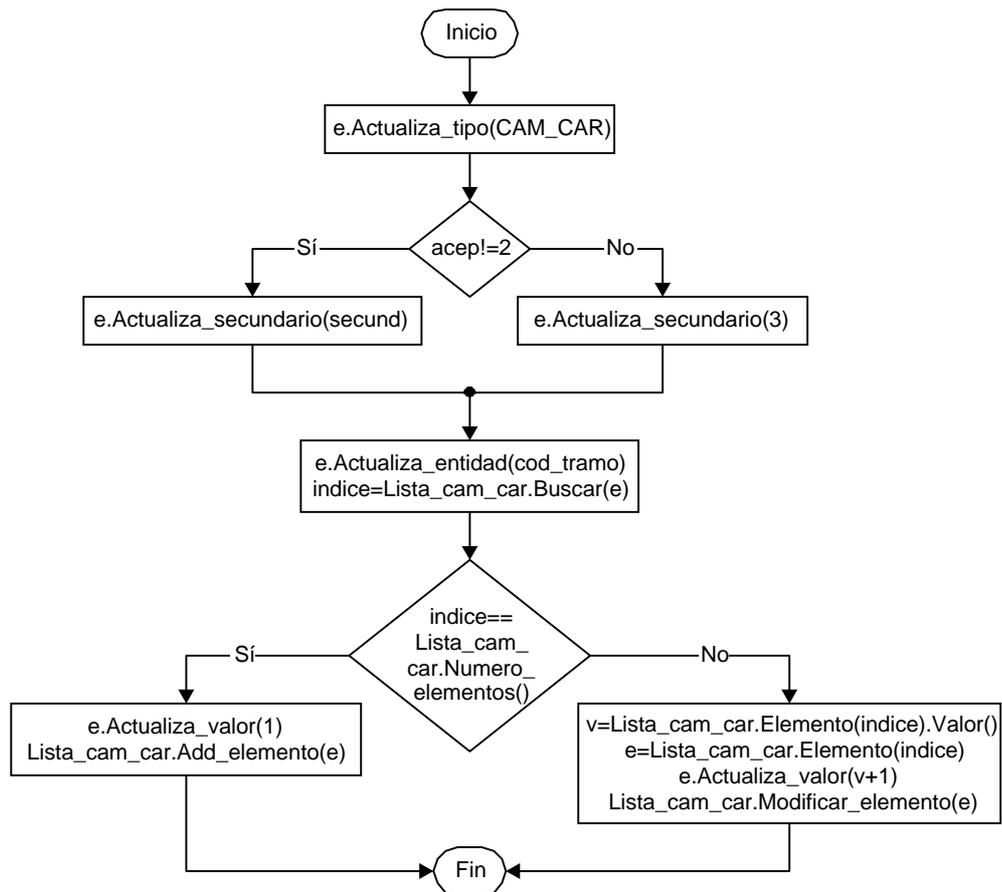


Figura 6.26. Estadísticas de cambios de carril

La otra estadística hace referencia al número de cambios de carril que efectúa cada uno de los vehículos durante la simulación. El diagrama de flujo del procedimiento que controla esta estadística se muestra en la figura 6.27.

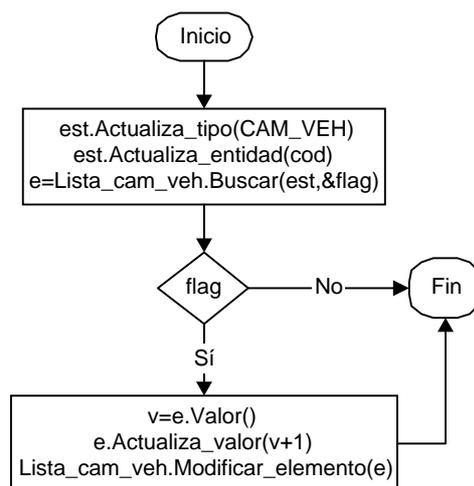


Figura 6.27. Estadísticas de cambios de carril por vehículo

Como se observa en la figura, este proceso consiste simplemente en buscar el elemento estadístico correspondiente al vehículo en la lista, y si se encuentra, se incrementa en una unidad su valor. En principio, si no hay problemas, debe existir ese

elemento, ya que cada vez que se crea un nuevo vehículo, se crea su elemento estadístico correspondiente.

6.2.3.4.2 Cambio de carril por mejora

Continuando con el diagrama de flujo de la figura 6.18, se observa que la comprobación de si un vehículo necesita realizar un cambio de carril por mejora, y la realización en su caso, tiene lugar cuando el valor devuelto por la función de cambio de carril obligatorio es *false*, y además, el vehículo que se encuentra por delante de él (en el caso de que lo haya) no esté realizando otra maniobra de cambio de carril.

El procedimiento de un cambio de carril por mejora se muestra en el diagrama de flujo de la figura 6.28.

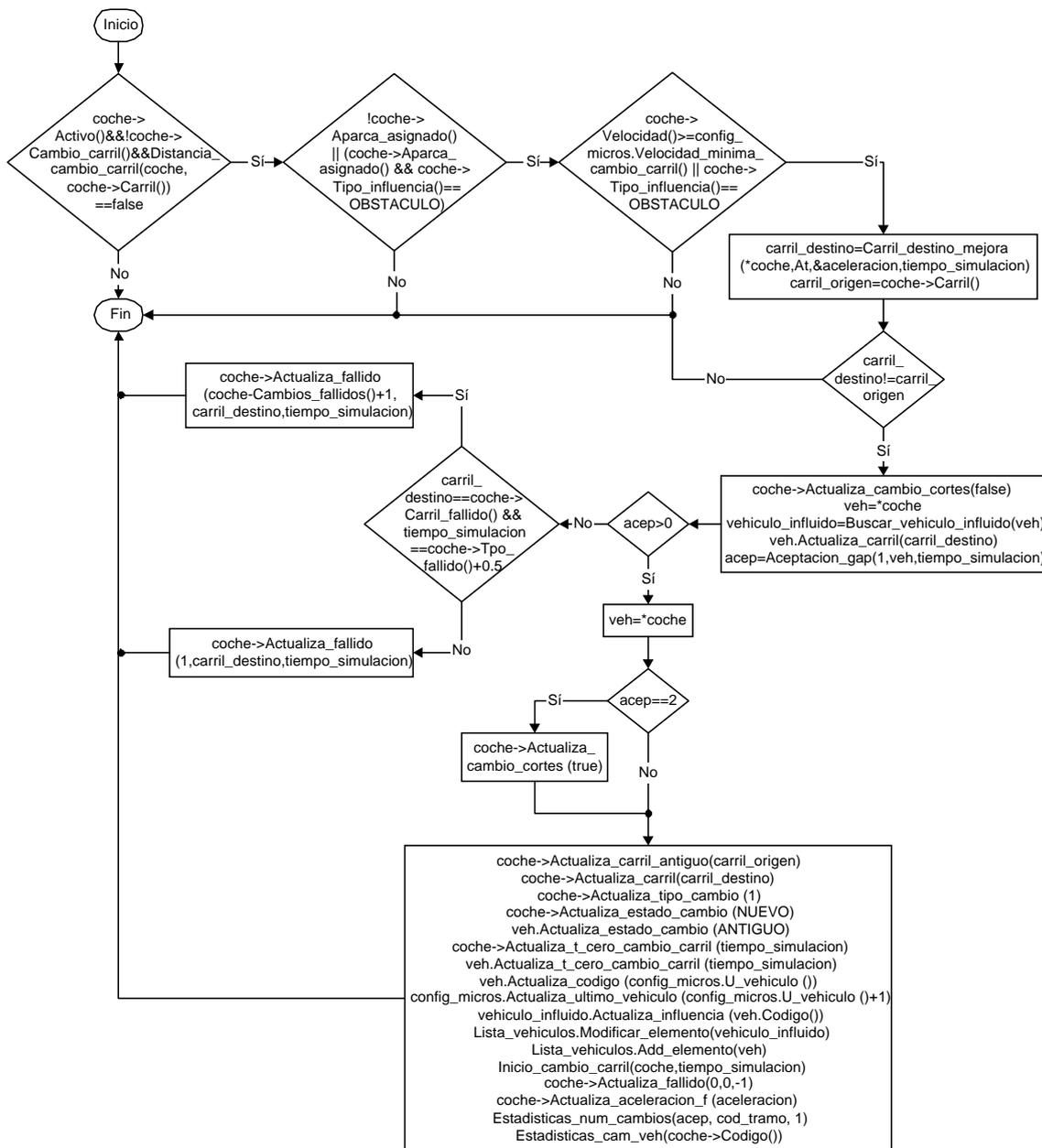


Figura 6.28. Cambio de carril por mejora

El cambio de carril por mejora sigue un proceso totalmente análogo al obligatorio. Las únicas diferencias reseñables son las condiciones previas que se deben dar para llevarlo a efecto y el proceso de búsqueda del carril de destino. Además, se emplean las expresiones del gap para este tipo de cambio, expuestas en el apartado 2.4.3.3 y que ya han sido vistas en la figura 6.24. Lógicamente, cuando se trata de actualizar el tipo de cambio de carril, ahora se pone el valor correspondiente al cambio por mejora, el resto de actualizaciones son idénticas a las del cambio obligatorio.

Condiciones previas

Estas condiciones se corresponden con las 3 estructuras *if* de la parte superior de la figura 6.28, y deben cumplirse para que se pueda realizar una maniobra de cambio de carril por mejora.

Estas condiciones se pueden resumir de la siguiente manera, para que un vehículo intente un cambio de carril por mejora debe estar activo, no estar cambiando de carril previamente, no estar a la distancia de cambio de carril obligatorio, no tener un aparcamiento asignado, o si lo tiene, que esté influido por un obstáculo y llevar una velocidad superior a la mínima necesaria para cambiar de carril (excepto en el caso de que esté influido por un obstáculo).

Búsqueda del carril de destino por mejora

Este procedimiento se encarga de encontrar el carril destino hacia el que un vehículo quiere desplazarse para la realización de un cambio de carril por mejora. El diagrama de flujo de dicho proceso se muestra en la figura 6.29.

El proceso de búsqueda del carril de destino por mejora consiste en recorrer todos los carriles y comprobar en cada uno de ellos si aumentaría la aceleración del vehículo al circular por ellos. Al finalizar ese bucle, en *carril_destino* está contenido el número del carril en el más mejorarían las condiciones de circulación del vehículo.

A continuación se comprueba si el carril de destino se encuentra a la derecha o a la izquierda del actual, para desplazarse un solo carril en esa dirección. El hecho de llamar nuevamente a la función *Mejora_condiciones* es porque de esa manera se actualiza la aceleración futura a su valor apropiado en el nuevo carril.

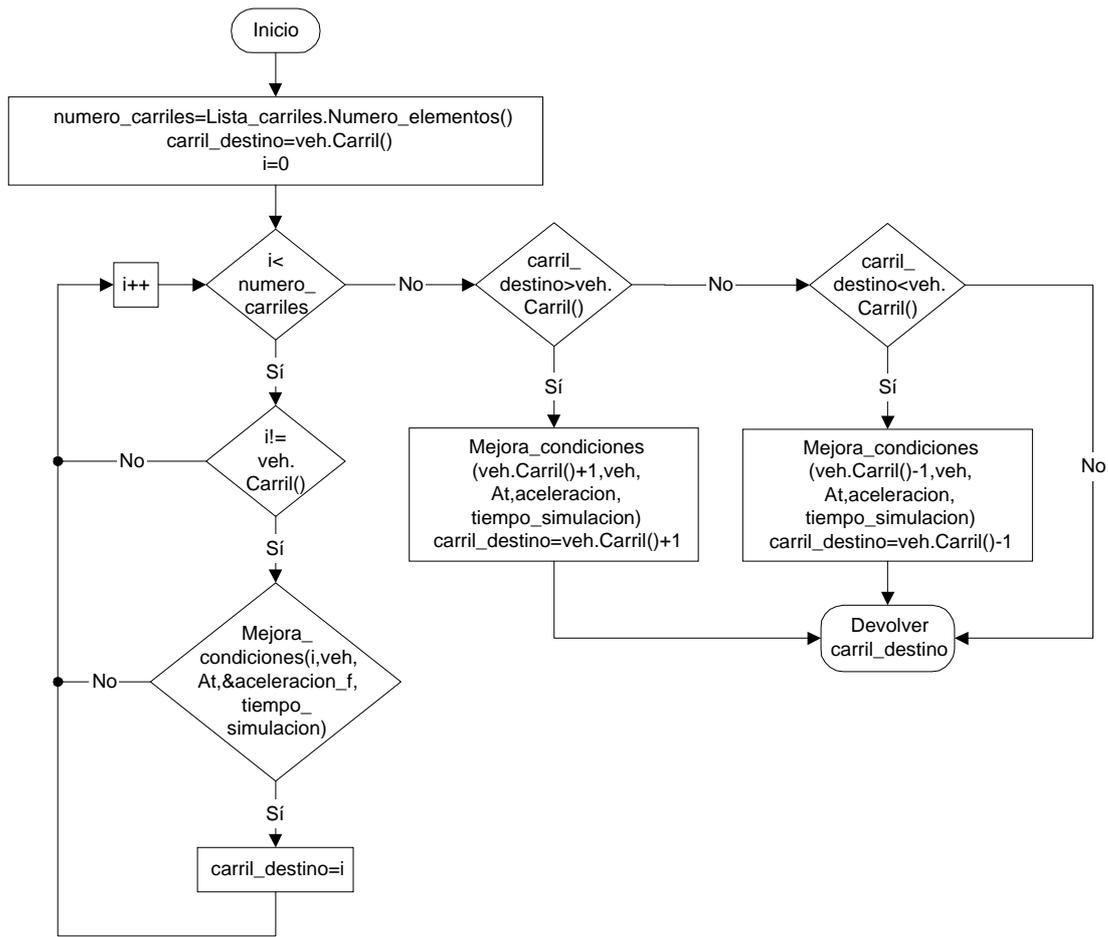


Figura 6.29. Carril de destino por mejora

En la figura 6.30 se muestra el diagrama de flujo de esta función que comprueba si mejoran las condiciones o no.

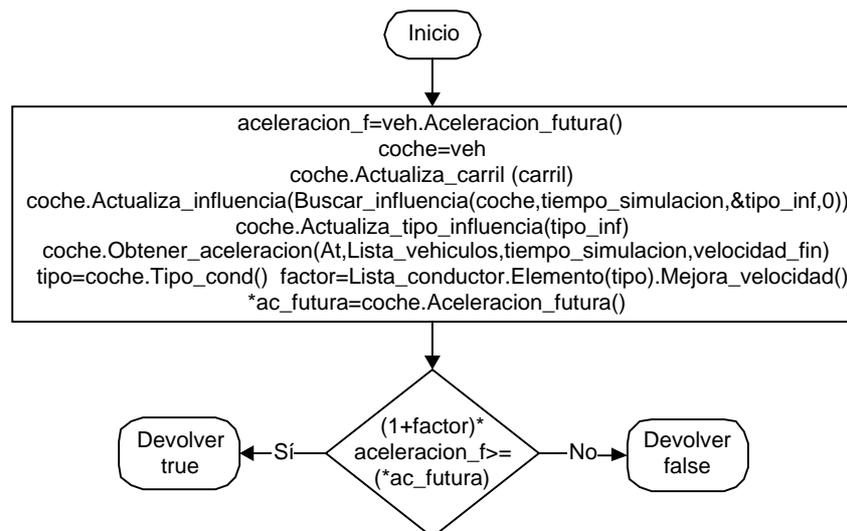


Figura 6.30. Mejora de condiciones

Como se observa en la figura, se comienza por introducir una copia del vehículo en el carril donde se desea estudiar si mejoran las condiciones. A continuación, se busca

cuál sería la influencia del vehículo en dicho carril y se calcula la aceleración futura teniendo en cuenta dicha influencia.

Si dicha aceleración futura mejora en un cierto porcentaje el valor de la aceleración que tendría si se quedara en el mismo carril, se determina que sí mejoran las condiciones en el nuevo carril. El tanto por ciento en que debe aumentar la aceleración para considerar que se produce una mejora de las condiciones suficiente para iniciar un cambio de carril, depende del tipo de conductor que lleva el vehículo.

6.2.3.4.3 Finalización de un cambio de carril

En este apartado se va a estudiar cuándo se da por finalizada una maniobra de cambio de carril y qué procedimiento se sigue para finalizarla. Esto se corresponde con la parte derecha del diagrama de flujo de la figura 6.18.

Este proceso se inicia cuando se encuentra un vehículo cuyo estado de cambio es *NUEVO* o *ANTIGUO* y además, el tiempo de simulación ya ha superado el momento en que inició el cambio de carril más el tiempo que dura la maniobra. Este tiempo predeterminado de duración de la maniobra de cambio de carril es un parámetro dependiente del tipo de conductor.

En el caso de que el vehículo sea *NUEVO*, es decir, que sea el vehículo que circula por el nuevo carril, el procedimiento consiste simplemente en actualizar su estado de cambio a *NO_CAMBIANDO* y modificar el correspondiente elemento de la lista de vehículos.

Cuando se trata de un vehículo ficticio (*ANTIGUO*) se busca el vehículo al que influía, se le actualiza la influencia a aquella que tiene el propio vehículo ficticio, se modifica el elemento de la lista, y por último se elimina el vehículo ficticio de la lista, reduciéndose en una unidad el número total de vehículos del tramo.

6.2.3.5 Salida de aparcamientos

Este apartado se corresponde con la última de las funciones del bloque (1) de la figura 6.10, y tiene que ver con la posible salida de sus aparcamientos de los vehículos que se encuentran en el tramo.

En relación con este proyecto, cabe decir, que para que un vehículo salga de su aparcamiento debe comprobar previamente si existe espacio suficiente para que su incorporación al carril tenga lugar con la suficiente seguridad. Para ello, se emplean las funciones de aceptación de gap vistas en el apartado 6.2.3.4.1.

Sin embargo, estas maniobras de incorporación al nuevo carril, no son consideradas como cambios de carril, por lo que no tienen duración temporal ni el resto de características de los cambios de carril.

Cuando un vehículo sale de un carril, se produce la actualización de algunos atributos del mismo que son de gran utilidad para las estadísticas, como son el tiempo de reincorporación, el tiempo de recorrido o la posición de inicio del recorrido en el tramo.

6.2.3.6 Estadísticas de semáforos

Continuando con el diagrama de flujo de la figura 6.10, una vez finalizado el bloque (1), se pasa a actualizar las estadísticas relacionadas con los semáforos, siempre y cuando el tramo esté regulado por uno de ellos.

El diagrama de flujo de la función que recoge los datos correspondientes a estas estadísticas y los procesa, se muestra en la figura 6.31. Estas estadísticas contienen el número de vehículos detenidos en el semáforo que regula el tramo en cada instante, así como el tiempo que permanece detenido en el semáforo cada uno de los vehículos que han tenido que parar a causa de él.

En este algoritmo, el atributo *valor* de los elementos estadísticos asociados a los vehículos en cada semáforo, indica el último instante de simulación en el que se ha comprobado que el vehículo indicado por *entidad* está parado en el semáforo del tramo indicado en *secundario*. Mientras que el atributo *media* indica el primer instante en el que se observó que el vehículo estaba parado a causa del semáforo.

Para comenzar el procedimiento se realiza un bucle en el que se recorren todos los vehículos del tramo, comprobando si tienen velocidad cero, están influidos por un semáforo y no acaban de ser generados. Los vehículos que cumplen estas tres condiciones, son aquellos que se encuentran detenidos justo por delante del semáforo, sin ningún otro vehículo por delante de ellos.

Para cada uno de estos vehículos, se incrementa en una unidad el número de vehículos detenidos ante el semáforo y se crea su elemento estadístico correspondiente relacionado con los semáforos, en cuyo atributo *valor* tiene el instante de simulación actual (este atributo no es comprobado para considerar que dos elementos estadísticos son iguales). A continuación, se busca en la lista de estadísticas asociada ese elemento, si no se encuentra, se introduce en la lista y se pone en su atributo *media* el instante de simulación actual. Si es encontrado, se modifica la lista para que ese elemento tenga el atributo *valor* igual al tiempo de simulación.

Seguidamente, se busca el vehículo que es influido por el primer vehículo de un carril parado en un semáforo, comprobando si dicho vehículo existe, tiene también una velocidad igual a cero y no ha sido creado en ese instante, es decir, si también se encuentra parado a causa del semáforo. Si no es así, se pasa a buscar otro vehículo que sea el primero de su carril parado ante el semáforo.

Si el vehículo influido cumple las condiciones anteriores, se incrementa en una unidad el número de vehículos detenidos en el semáforo, y se busca en la lista ese elemento. Si se encuentra el vehículo, se actualiza *valor* al instante de simulación, mientras que si no se encuentra, dicho elemento es añadido a la lista, con el contenido de *media* igual al instante de simulación.

Este vehículo influido, pasa a ser el vehículo del que se estudia su influencia, y se repite el proceso hasta que se encuentre que el vehículo influido no está detenido ante el semáforo.

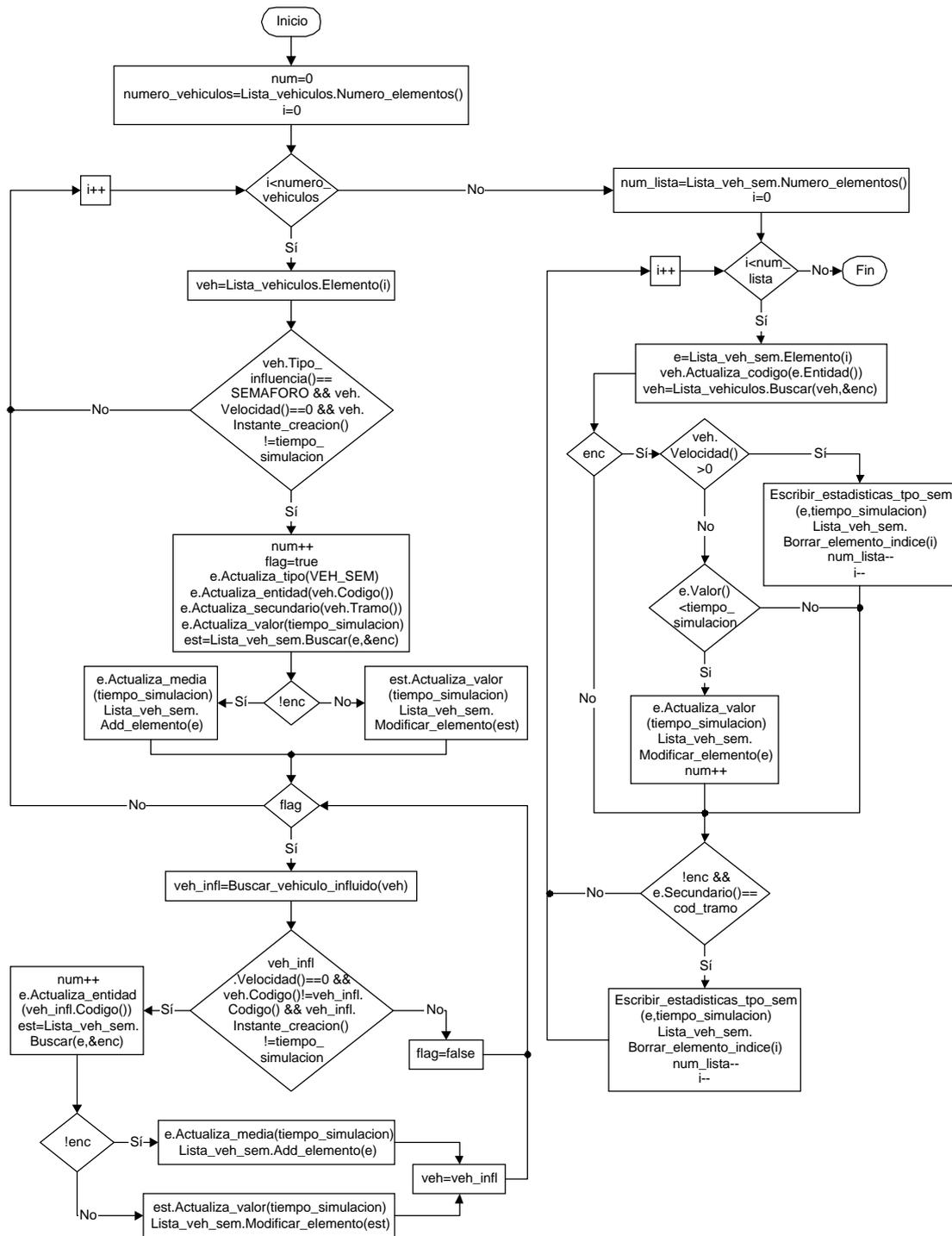


Figura 6.31. Estadísticas de vehículos en semáforos

Una vez finalizado el proceso de añadir nuevos vehículos a la lista de vehículos detenidos en el semáforo, se pasa a sacar de la lista aquellos que ya han arrancado tras haber estado detenidos en el semáforo. Este proceso de eliminación de elementos de la lista, se corresponde con la parte derecha del diagrama de flujo de la figura 6.31.

Para ello, se comienza recorriendo la lista de este tipo de elementos estadísticos, y buscando en la lista de vehículos el correspondiente al atributo *entidad* de cada elemento estadístico. Si dicho vehículo no se encuentra en la lista (porque ya ha salido del tramo) y estaba en el tramo que se encuentra en estudio en ese instante, hay que

sacarlo también de la lista de vehículos detenidos en el semáforo, no sin antes escribir en su fichero de texto correspondiente el tiempo que ha estado detenido en el semáforo.

Si el vehículo buscado se encuentra, hay que comprobar si la velocidad del mismo es distinta de cero. Si es distinta de cero, significa que el vehículo ha dejado de estar parado en el semáforo, por lo que se elimina de la lista de estadísticas correspondiente y se escriben sus datos en un fichero. Si la velocidad es igual a cero, se comprueba si *valor* es menor que el tiempo de simulación (significa que es un vehículo que ya estaba en la lista, pero que aún no ha sido actualizado) y entonces se actualiza a dicho valor, incrementando en una unidad el número de vehículos detenidos en el semáforo. Si *valor* es igual al tiempo de simulación, el elemento ya ha sido actualizado, por lo que se pasa al siguiente elemento de la lista.

Las estadísticas correspondientes al número de vehículos detenidos en el semáforo del tramo en cada instante de tiempo, son escritas al finalizar todo el proceso anterior en cada instante de simulación.

6.2.3.7 Actualización de nodos

Para finalizar con el proceso de actualización indicado en la figura 6.10, queda por ver el bloque de acciones (2), que se ejecuta para cada uno de los nodos que componen el viario.

En relación con este proyecto, el único aspecto de interés es la actualización del instante del comienzo del recorrido en un tramo por parte de un vehículo, y su posición de inicio de recorrido cuando un vehículo sale de un movimiento para incorporarse a un tramo. Estos datos son muy importantes para llevar correctamente las estadísticas relacionados con el tiempo y velocidad media de los vehículos en un tramo.

Además, cuando un vehículo se incorpora a un tramo, es necesario saber a qué carril lo ha hecho para actualizar los valores de déficit o superávit de vehículos en los carriles del tramo.

6.2.4 Escritura de estadísticas finales

Para finalizar con el simulador, sólo queda por ver el bloque (4) de la figura 6.3, que está relacionado con la escritura en los ficheros correspondientes de aquellas estadísticas que no han podido ser escritas antes, ya que necesitan de todos los datos que se generan durante la simulación. Estas estadísticas son:

- Número de cambios de carril totales de cada tipo.
- Número de cambios de carril realizados por cada vehículo.
- Media de las posiciones de generación de los vehículos en los diferentes tramos.
- Media de las posiciones de aparcamiento de los vehículos en cada tramo.
- Tiempo que llevan en el sistema los vehículos que no han concluido su recorrido cuando finaliza la simulación.

- Tiempo que llevan en sus tramos correspondientes los vehículos que no han terminado su recorrido por el viario cuando acaba la simulación.
- Tiempo que llevan detenidos en los semáforos los vehículos que aún no han arrancado cuando acaba la simulación.

6.3 Interfaz de usuario

En este apartado se explica cómo se comunica el usuario con el Simulador Microscópico de Tráfico. Esta comunicación se lleva a cabo a través de un interfaz gráfico (correspondiente al formulario generado con C++ Builder) y con una serie de ficheros de texto de entrada y salida, de los cuales se examinan sus formatos en los apartados 6.3.2 y 6.3.3 respectivamente.

6.3.1 Interfaz gráfico

El interfaz gráfico permite que el usuario se comuniqué con el simulador a través de un formulario generado con C++ Builder. Este interfaz se muestra en la figura 6.32 y está formado por 3 botones.

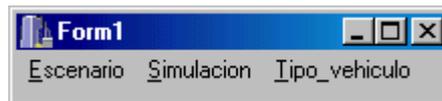


Figura 6.32. Interfaz gráfico

Escenario

Al pulsar sobre este botón, se lee y se carga toda la información contenida en los ficheros de entrada al simulador, que contienen los distintos elementos del escenario que se desea simular. Estos ficheros serán analizados minuciosamente en el apartado 6.3.2.

Tipo_vehiculo

Cuando se pulsa este botón, aparece la rejilla mostrada en la figura 6.2. En dicha rejilla, se pueden modificar los valores que toman los parámetros de los distintos tipos de conductores. Para ello, no hay más que modificar los valores de los parámetros y pulsar el botón *Grabar*.

Los parámetros que pueden ser modificados son: tiempo de reacción, factor de deceleración, factor de aceleración, velocidad deseada, factor de distancia de influencia, porcentaje de vehículos con ese tipo de conductor, velocidad mínima, duración de la maniobra de cambio de carril, factor de mejora de velocidad y probabilidad de cortesía. El significado de cada uno de estos parámetros fue convenientemente explicado en el apartado 5.2.3.3.1.3.

Simulación

Si se presiona este botón, comienza a ejecutarse la simulación del escenario que ha sido previamente cargado. En el caso de que no se haya pulsado el botón *Tipo_vehiculo*, los valores que toman los parámetros de los diferentes tipos de vehículos durante la simulación son los mismos que tuvieron en la última simulación, y que se han cargado a través de un fichero de texto.

6.3.2 Ficheros de entrada

La mayor parte de los datos necesarios para la ejecución de una simulación son importados por el simulador a través de los diferentes ficheros de texto. Estos ficheros de texto contienen los datos relacionados con el escenario que se desea simular. A continuación, se explica el formato usado para la representación de los datos en cada uno de los ficheros de texto.

Tramos

La información que hace referencia a los distintos tramos existentes en el viario se encuentra recogida en el fichero *ftramos*. A continuación se muestra un ejemplo para la información referida a un tramo. En el fichero anterior, existe una estructura análoga a la del ejemplo para cada uno de los tramos del escenario.

```
0 363.25 2 3 15.00 10.00 10.00 5.00 -1 1
3 33
3 33
3 33
```

Los números que aparecen en la primera fila tienen el siguiente significado: código del tramo, longitud del tramo, código del nodo de destino del tramo, número de carriles en el tramo, velocidad máxima a la que se puede circular en el tramo, distancia fija al final del tramo a partir de la cual sólo se realizan cambios obligatorios, distancia al final del tramo a partir de la cual se considera lo que ocurre en el nodo siguiente, distancia de no influencia, código del semáforo que regula la circulación por el tramo y código de la función de generación empleada en el tramo.

El código de la función de generación es igual a 0 cuando no se generan vehículos en ese tramo, mientras que la correspondencia de los códigos con las diferentes funciones estadísticas es la siguiente: 1 para la distribución normal, 2 para la Poisson, 3 para la función de distribución exponencial y 4 para la triangular.

Después de esta primera línea, aparecen tantas líneas como carriles tiene el tramo, cada una de las cuales presenta dos números. El primero indica el código de la señal que regula el carril, y el segundo, el porcentaje de vehículos que han de circular por dicho carril.

Señales

Para cada nodo existente en el escenario de simulación, existe un fichero de texto que contiene la información correspondiente a la caja semafórica que regula dicha

intersección. A continuación se muestra un ejemplo del fichero *N_2.caj*, que contiene la caja semafórica para el nodo 2.

```
//Codigo nodo
2
//Numero estructuras
1
//Ciclo
62
//Indica si los tiempos estan en porcentajes o segundos (P,T)
T
//Indica si los intermedios son posteriores o anteriores (P,T)
P
//codigo estructura
1
//Numero fases
3
//Indicamos el tiempo de la fase y los intermedios que tiene
9 2 2 1 2
//Indicamos el tiempo de la fase y los intermedios que tiene
9 3 1 1 2
//Indicamos el tiempo de la fase y los intermedios que tiene
16 3 2 2 4 3
//Numero senales
8
1 BBBBDCBBBBBBBBBB
2 DDDDDDDCCBBBBBBB
3 CCCBBBBBBBBBDDDD
4 BBBBBBBBDDDBBBBB
5 HHBBBBBDDDDDDDD
6 BBBBBBBBDDDDHBB
7 BBBBDDDDDDHBBBBB
8 DDDDDHBBBBBDDDD
```

En primer lugar, se indica el código del nodo que regula, el número de estructuras que contiene la caja y el tiempo que dura el ciclo de la señal. A continuación se indica si los tiempos están expresados en porcentajes o en segundos y si los tiempos intermedios son anteriores o posteriores a la señal correspondiente.

Seguidamente, se indica el código de la estructura y el número de fases que contiene. Para cada una de las fases, se indica el tiempo que dura tanto la fase en sí misma, como los intermedios correspondientes.

Para finalizar, se indica el número de señales presentes en la caja y para cada una de ellas se indica su código y la secuencia de códigos de color que va tomando la señal semafórica correspondiente.

Movimientos

La información que hace referencia a los movimientos en los que se encuentran desagregados los giros que se pueden realizar en los nodos del viario, se encuentra contenida en el fichero *fmov*. A continuación se muestra, a modo de ejemplo, parte de uno de estos ficheros.

```
2 0 18
0
0 0 6 0 9.33 3
1
0 1 2 10.07 3
3
1 2 1 10.71 3
...
```

La primera línea contiene tres números, que indican el código del nodo, el índice en la lista de señales correspondiente a la señal que lo regula y el número de movimientos existentes en el nodo.

Para cada uno de los movimientos, existen a continuación dos líneas. La primera de ellas contiene un único número, que indica el tipo de movimiento de que se trata, siendo igual a 0 para movimientos carril-carril, 1 para movimientos carril-movimiento, 2 en el caso de movimientos movimiento-movimiento y 3 para los carril-carril.

La segunda de las líneas asociada a cada movimiento contiene 6 números, que indican respectivamente: código del tramo origen, código del carril origen, código del tramo destino, código del carril destino, longitud del movimiento y código de la señal que regula el movimiento.

Conexiones entre movimientos

Los distintos movimientos se encuentran conectados entre sí y la información referente a dichas conexiones se encuentra en el fichero *fcon*, del cual se muestra a continuación un ejemplo de una parte de él.

```
2 0 6 0 0
0 -1
2 0 4 1 1
1 2
6 -1
2 0 2 1 1
1 4
2 -1
...
```

Cada una de las conexiones de movimientos permite que un vehículo se desplace desde un carril concreto de un tramo hasta otro carril determinado de otro tramo. La primera línea de cada una de estas conexiones contiene 5 números que representan el código del nodo, el código del tramo origen, el código del tramo destino, el código del carril origen y el código del carril destino.

A continuación, hay tantas filas como movimientos conformen la conexión. Cada una de estas filas contiene 2 números, el primero indica el índice del movimiento en la lista, y el segundo, el índice del siguiente movimiento de la conexión (o -1 en el caso de ser el último movimiento de la conexión).

Giros

La información correspondiente a los giros que se pueden realizar en los nodos del viario se encuentra recogida en el fichero *fgiros*. A continuación se muestra un ejemplo para el caso de un único nodo, si hubiera más, se repetiría el formato para cada uno de los nodos existentes.

```
2 9
0 6 35.00 1
0 2 35.00 2
0 4 35.00 -1
3 6 35.00 4
3 1 35.00 5
3 4 35.00 -1
5 1 35.00 7
5 6 35.00 8
5 2 35.00 -1
```

La primera de las líneas contiene 2 números que indican el código del nodo y el número de giros que se definen en él.

Para cada uno de los giros existe una línea que contiene 4 números con los siguientes significados: código del tramo origen, código del tramo destino, porcentaje de vehículos del tramo origen que toman ese giro e índice del siguiente giro que tiene el mismo tramo origen.

Incompatibilidades entre movimientos

La información referida a las incompatibilidades existentes entre los distintos movimientos de un nodo se encuentra recogida en el fichero *finc*. A continuación se muestra un ejemplo de este fichero.

```
2 3
0 0 0 12 1 -1 7 3 1 6
0 0 0 15 -1 -1 2 3 1 6
0 0 12 15 -1 1 4 1 6 3
```

En la primera línea aparece el código del nodo y el número de incompatibilidades entre movimientos que se definen en él.

Seguidamente, aparece una línea por cada incompatibilidad que se vaya a definir. En dicha línea hay 10 números con los siguientes significados: índice del movimiento origen, índice del movimiento destino, código del movimiento origen, código del movimiento destino, índice en la lista de incompatibilidades de la siguiente incompatibilidad, índice de la anterior incompatibilidad, posición del movimiento origen en la que empieza el conflicto, posición del movimiento origen en la que se debe detener un vehículo para evitar el conflicto, posición del movimiento destino en la que empieza el conflicto y posición del movimiento destino en la que se debe detener un vehículo para evitar el conflicto.

Obstáculos

Los obstáculos que van a existir en el viario a lo largo de la simulación y que no son debidos a vehículos que están aparcando, se definen en el fichero *fobstaculos*.

Por cada uno de los obstáculos existentes va a haber una línea en dicho fichero con la siguiente información: código del obstáculo, tipo de obstáculo, código del tramo en el que se encuentra, código del carril en el que está, número de carriles a la derecha hacia los que se extiende, número de carriles a la izquierda hacia los que se extiende, posición del tramo en la que se encuentra, extensión en metros del obstáculo, instante de aparición del obstáculo y duración en segundos del obstáculo.

Aparcamientos

La información relativa a los aparcamientos disponibles en el viario se encuentra contenida en el fichero *faparcamientos*. A continuación se muestra un ejemplo de ese fichero:

```
2 11 0 3 300 0 0 0
2 12 0 3 200 0 0 0
2 13 0 3 210 0 0 0
...
```

El fichero *faparcamientos* contiene una línea por cada aparcamiento que se define, en la cual hay 8 números con el siguiente significado: código del tramo, código del aparcamiento, código del carril, tipo de aparcamiento, posición del aparcamiento, si está ocupado y código del vehículo que lo ocupa si lo hubiera.

Paradas

Las paradas que deben realizar los vehículos de mercancías se determinan en el fichero *fparadas*.

Para cada una de las paradas definidas hay una línea en dicho fichero con la siguiente información: código del tramo, posición de la parada, código del carril, índice de la siguiente parada y duración de la parada.

Aparcamientos masivos

La información relativa a los aparcamientos masivos existentes en el viario se encuentra contenida en el fichero *faparcamientos_masivo*. A continuación se muestra un ejemplo de este fichero:

```
3 0 0 150.0 0 2 6.0 6.0 10
```

Por cada uno de los aparcamientos masivos que se definan hay una línea en el fichero con los siguientes datos: código del tramo, código del carril, código del aparcamiento, posición del aparcamiento, si está ocupado totalmente, número de plazas del aparcamiento, velocidad de entrada al aparcamiento, velocidad de salida y longitud de la rampa de entrada.

Ficheros binarios

Existen una serie de ficheros que contienen la información en formato binario. Entre ellos se encuentra el fichero *t_vehiculos.dat* que contiene la información relacionada con los distintos tipos de vehículos que van a circular sobre el viario, así como *t_conductor.dat* que contiene la información relacionada con los tipos de conductores.

6.3.3 Ficheros de salida

Toda la información de salida generada por el simulador se proporciona mediante ficheros de texto que deben ser analizados y procesados convenientemente para tener dicha información en el formato más adecuado.

A continuación se va a explicar el contenido de los distintos ficheros de salida generados y el formato de la información representada en ellos.

micro.sal

Este fichero contiene el resultado fundamental de la simulación, ya que indica la situación de todos los vehículos presentes en el viario en cada instante de tiempo. A modo de ejemplo, se exponen a continuación algunas de las líneas de este fichero:

Instante	Situacion	Cod_veh	Tramo	Nodo	Mov	Aceleracion	Velocidad	Posicion	Carril	D_influencia	D_seguridad
2.50	TRAMO	0	6	0	0	1.80	0.90	249.45	1	1.00	1.00
3.00	TRAMO	0	6	0	0	1.80	1.80	250.35	1	8.96	1.46
3.00	TRAMO	1	6	0	0	1.80	0.90	144.45	0	1.00	1.00
3.50	TRAMO	0	6	0	0	1.80	2.70	251.70	1	9.54	2.04
3.50	TRAMO	1	6	0	0	1.80	1.80	145.35	0	8.96	1.46
4.00	TRAMO	0	6	0	0	0.92	3.16	253.28	1	9.92	2.42
4.00	TRAMO	1	6	0	0	1.80	2.70	146.70	0	9.54	2.04
4.50	TRAMO	2	0	0	0	1.80	0.90	40.45	2	1.00	1.00

Por cada uno de los vehículos activos en el escenario en cada instante de simulación se escribe una línea en el fichero con el siguiente significado: instante de simulación, situación en tramo en movimiento, código del vehículo, código del tramo, código del nodo, código del movimiento, aceleración del vehículo, velocidad del vehículo, posición del vehículo, código del carril, distancia de influencia y distancia de seguridad del vehículo.

vehiculoN.dat

Durante la simulación se genera un fichero por cada uno de los vehículos que circulan por el viario, en el que se recoge la información relativa al vehículo en cada instante. Las siguientes líneas son un ejemplo de parte de este fichero:

Instante	Cod_veh	Tramo	Nodo	Mov	Aceleracion	Velocidad	Posicion	Carril	D_influencia	D_seguridad	D_mlc	Tramo
2.50	0	6	0	0	1.80	0.90	249.45	1	1.00	1.00	415.12	2
3.00	0	6	0	0	1.80	1.80	250.35	1	8.96	1.46	415.00	2
3.50	0	6	0	0	1.80	2.70	251.70	1	9.54	2.04	406.69	2

En cada instante de simulación se escribe una línea en el fichero con la siguiente información: instante de simulación, código del vehículo, código del tramo, código del nodo, código del movimiento, aceleración del vehículo, velocidad del vehículo, posición del vehículo, código del carril, distancia de influencia, distancia de seguridad,

distancia de cambio de carril obligatorio y código del siguiente tramo por el que debe circular el vehículo.

TramoN.dat

Para cada uno de los tramos del viario se genera un fichero que contiene la información relativa a los distintos cambios de carril que se producen en él. A continuación se muestra parte de uno de estos ficheros:

Tipo	Instante	Tramo	Vehiculo	Posicion	Origen	Destino	Cortesia	Gap_del	Gap_tra
Obl.	11.50	7	17	248.70	1	0		0.000	83.670
Obl.	17.00	7	31	255.18	0	1		3.011	47.720
Obl.	17.00	7	39	245.17	0	1		7.006	37.715
Mej.	19.50	7	12	207.69	0	1	C	8.337	2.989

Para cada cambio de carril producido en el tramo, se escribe una línea en el fichero con la siguiente información: tipo de cambio (obligatorio o por mejora), instante del cambio, código del tramo, código del vehículo que realiza el cambio, posición del vehículo, código del carril origen, código del carril destino, indicación de si el cambio es por cortesía, gap delantero y gap trasero.

Cuando en las columnas *Gap_del* y *Gap_tra* aparece el valor 0, quiere decir que no hay influencia de ningún tipo ni por delante del vehículo ni por detrás del mismo en el nuevo carril hacia el que se desplaza.

GeneracionN.dat

Existe un fichero por cada tramo en el que se produce generación de vehículos. El formato es el siguiente:

4.00	0	2	2	3	0
6.00	0	0	5	3	0
7.00	0	2	7	5	0

Por cada vehículo que se genera en el tramo, se escribe una línea en el fichero en la que se indica: instante de generación, código del tramo, código del carril, código del vehículo, código del tramo destino y código del carril destino.

Cortesia.dat

En este fichero se almacena la información que hace referencia a los cambios de carril por cortesía que se producen en el viario. A continuación se muestra un ejemplo de este fichero:

Tipo	Instante	Tramo	Vehiculo	Posicion	Origen	Destino	Veh_detras	Distancia	Intentos
Mej.	19.50	7	12	207.69	0	1	63	2.989487	3
Obl.	27.50	6	72	410.04	0	1	121	0.694153	6
Mej.	28.00	0	45	315.21	2	1	83	1.334290	3

Cada línea del fichero se corresponde con un cambio de carril por cortesía, en la que se indica: tipo del cambio de carril, instante del cambio, código del tramo, código del vehículo que cambia de carril, posición del cambio, carril del origen del cambio, carril destino del cambio, código del vehículo que permite el cambio, distancia entre ambos vehículos y número de intentos fallidos que ha realizado anteriormente el vehículo antes de llevar a cabo el cambio de carril.

cam_veh.dat

Este fichero contiene el número de cambios de carril que realiza cada uno de los vehículos que circulan por el viario. Ejemplo:

Vehiculo Cambios	
0	1
1	2
2	2
3	0
4	5

El primer número de cada fila representa el código del vehículo, y el segundo el número total de cambios realizados por el vehículo a lo largo de la simulación. En la última fila del fichero se muestra el número medio de cambios que realiza un vehículo a lo largo de una simulación completa.

num_cambios.dat

En este fichero se almacena la información relativa al número total de cambios de carril que se producen durante la simulación. A continuación se muestra un fichero de este tipo:

Tramo	Mejora	Obligatorio	Cortesía	Total
0	25	8	2	35
1	0	0	0	0
2	58	20	1	79
3	2	30	0	32
4	10	0	0	10
5	60	9	1	70
6	22	6	1	29
7	24	2	1	27
8	0	0	0	0
9	0	0	0	0
TOTAL	201	75	6	282

Este fichero contiene una línea para cada uno de los tramos del viario, en la que se indican: número de cambios por mejora, número de cambios obligatorios, número de cambios por cortesía y número total de cambios.

La última línea del fichero muestra los mismos datos que las líneas anteriores, pero para el total de tramos.

pos_apa.dat

Las posiciones en las que se producen los aparcamientos de los vehículos se almacenan en este fichero. Ejemplo:

Instante	Vehiculo	Tramo	Posicion
68.0	42	5	270.0
68.5	45	5	260.0
69.0	0	2	300.0

En cada línea se muestra la información relativa a un aparcamiento realizado por un vehículo, dicha información es: instante del aparcamiento, código del vehículo, código del tramo y posición del aparcamiento.

media_pos_apa.dat

Este fichero contiene una línea por cada tramo en los que se producen aparcamientos, mostrando el código del tramo, el número total de aparcamientos realizados en el tramo y la posición media en la que se han producido esos aparcamientos. Ejemplo:

Tramo	Total aparcamientos	Posicion media
2	12	254.166667
5	11	248.181818

pos_gen.dat

En este fichero se escribe una línea por cada vehículo generado, en la que se indica: instante de generación, código del vehículo, código del tramo, código del carril y posición de generación. Ejemplo:

Instante	Vehiculo	Tramo	Carril	Posicion
2.0	0	0	1	234.0000
3.0	1	0	2	13.0000
3.5	2	0	0	168.0000
...				

Al final del fichero se incluye una línea más por cada uno de los tramos en los que se generan vehículos, en la cual se indica la posición media en la que se han generado los vehículos en cada tramo.

veh_gen.dat

La información que hace referencia al número de vehículos generados en cada instante en cada uno de los tramos del viario se encuentra contenida en este fichero, del que a continuación se muestra algunas líneas:

Instante	Tramo0	Tramo1	Tramo2	Tramo3	Tramo4	Tramo5	Tramo6	Tramo7	Tramo8	Tramo9
0.0	0	0	0	0	0	0	0	0	0	0
0.5	0	0	0	0	0	0	0	0	0	0
1.0	0	0	0	0	0	0	0	0	0	0
1.5	0	0	0	0	0	0	0	0	0	0
2.0	0	0	0	0	0	0	1	0	0	0
2.5	0	0	0	0	0	0	1	0	0	0
3.0	0	0	0	0	0	0	0	0	0	0
3.5	0	0	0	0	0	0	0	0	0	0
4.0	1	0	0	0	0	0	0	1	0	0
4.5	0	0	0	0	0	0	0	0	0	0
5.0	0	0	0	0	0	0	0	0	0	0
5.5	1	0	0	0	0	0	0	0	0	0
6.0	0	0	0	0	0	0	0	1	0	0
6.5	1	0	0	0	0	0	0	0	0	0
7.0	0	0	0	0	0	0	0	1	0	0
7.5	1	0	0	0	0	0	1	0	0	0
8.0	0	0	0	0	0	0	2	1	0	0

Cada línea se corresponde con un instante de simulación, en el que se señala el número de vehículos que se han generado en ese instante en cada uno de los tramos del viario.

Al final del fichero se incluyen dos líneas adicionales, en las que se indica el número total de vehículos generados en cada tramo y el número medio de vehículos que se generan en cada tramo en cada instante de tiempo.

tpo_sem.dat

Las estadísticas correspondientes a los tiempos que los vehículos se encuentran detenidos a causa de los semáforos se almacenan en este fichero. A continuación se muestran algunas líneas de un fichero de este tipo:

Vehiculo	Tramo	Inicio	Final	Tiempo
25	0	21.0	29.5	8.5
75	0	24.5	29.5	5.0
60	0	27.5	29.5	2.0
5	0	22.0	30.0	8.0
0	6	29.5	30.0	0.5

Por cada vehículo que se detiene en un semáforo, se escribe una línea en el fichero, en el que se indican el código del vehículo, el código del tramo, el instante en el que se detiene a causa del semáforo, el instante en el que deja de estar parado ante ese semáforo y el tiempo total que ha permanecido detenido en el semáforo.

veh_semN.dat

En estos ficheros se muestra el número de vehículos detenidos en el semáforo del tramo *N* en cada instante de tiempo. Ejemplo:

Instante Vehiculos	
0.0	0
0.5	0
1.0	0
	...
20.5	0
21.0	1
21.5	1
22.0	2

tpo_sis.dat

Este fichero recoge las estadísticas del tiempo que los vehículos permanecen activos en el sistema. A continuación se muestra un ejemplo de este fichero.

Vehiculo	Comienzo	Final	Tiempo
33	12.0	60.0	48.0
66	17.5	62.5	45.0
124	25.0	65.0	40.0

Cada línea del fichero contiene 4 números que representan: el código del vehículo, instante en el que entra en el sistema, instante en el que sale del sistema y tiempo total que ha permanecido en el mismo.

tpo_tra.dat

El tiempo que permanece un vehículo en un tramo se encuentra recogido en este fichero, que muestra 6 valores en cada línea: código del vehículo, código del tramo, instante de comienzo del recorrido del vehículo por el tramo, instante en el que finaliza el recorrido del vehículo por el tramo y tiempo total que el vehículo ha permanecido en el tramo. Ejemplo:

Vehiculo	Tramo	Comienzo	Final	Tiempo
6	0	6.5	29.5	23.0
23	0	10.5	29.5	19.0
0	6	2.0	29.5	27.5

vel_med.dat

En este fichero se almacenan las estadísticas correspondientes a la velocidad media que ha llevado un vehículo durante su desplazamiento por un tramo. A continuación se muestran algunas de las líneas correspondientes a este fichero:

Vehiculo	Tramo	Espacio	Tiempo	Velocidad
6	0	42.120	23.0	1.83130
23	0	36.951	19.0	1.94478
0	6	167.936	27.5	6.10675

Como se observa, cada línea se corresponde con el recorrido de un vehículo por un tramo, en la que se indican: código del vehículo, código del tramo, espacio recorrido en el tramo, tiempo de permanencia en el tramo y velocidad media desarrollada por el vehículo en el tramo durante ese tiempo.

veh_car.dat

En este fichero se refleja el número de vehículos que han circulado por cada uno de los carriles de cada uno de los tramos del viario en cada instante de simulación. A continuación se muestran algunas de las líneas de este fichero:

Instante	Tramo	Carril	Vehiculos
0.0	0	0	0
0.0	0	1	0
		...	
83.5	6	0	14
83.5	6	1	13
83.5	6	2	14
83.5	7	0	19
83.5	7	1	15
		...	
Media	0	0	3.381667
Media	0	1	4.655000
Media	0	2	5.193333
Media	1	0	0.000000

En este ejemplo se aprecia que en cada línea del fichero hay 4 valores que representan: instante de simulación, código del tramo, código del carril y número de vehículos que circulan por el mismo. Además, al final del fichero se incluye una línea para cada carril de todos los tramos en las que se indica el número medio de vehículos a lo largo de la simulación.

7 Pruebas del Simulador Microscópico de Tráfico

7.1 Introducción

El paso final en la realización de este proyecto consiste en la realización de una serie de pruebas del código del Simulador Microscópico de Tráfico. Estas pruebas se han dividido en dos tipos:

- **Pruebas de validación.** Este primer conjunto de pruebas tiene por objetivo la validación y comprobación de la correcta implementación de los nuevos modelos de cambio de carril, generación de vehículos y estadísticas. Así como de verificar la integración de dichos modelos en el código del simulador. Para ello, se analizan los ficheros de texto generados tras una simulación y se comparan con los resultados teóricos esperados que se preveían en el modelo.
- **Pruebas del sistema.** Este conjunto de pruebas pretende la obtención de los resultados de aplicar el simulador sobre distintos escenarios reales pertenecientes a la ciudad de Sevilla.

El formato de todos los ficheros de texto a los que se hace referencia en estas pruebas, ya ha sido convenientemente explicado en los apartados 6.3.1 y 6.3.2.

7.2 Pruebas de validación

Antes de comentar las pruebas de validación de los distintos modelos, cabe decir que aquí sólo se muestran algunas de las pruebas realizadas que se consideran más representativas, ya que durante todo el proceso de desarrollo del código del simulador han sido innumerables las pruebas de este tipo que se han realizado para la depuración de dicho código.

El escenario empleado para la realización de este tipo de pruebas, ha sido el mismo que en la *Prueba básica* descrito en el apartado 7.3.1, sin embargo, este aspecto no es trascendente para las pruebas de validación, ya que se hace un estudio local del comportamiento de determinados vehículos o de determinados instantes, pero no del escenario en su conjunto.

Los parámetros de los diferentes tipos de conductor introducidos mediante el interfaz de usuario de la figura 6.2, y que se han empleado en todas las pruebas, se muestran en la siguiente tabla:

Tipo de conductor	Tiempo de reacción	Factor de deceleración	Factor de aceleración	Velocidad deseada	Factor de distancia de influencia
Novato	2	0.035	0.25	15	0.18
Experimentado	1.5	0.035	0.25	15	0.18
Agresivo	1	0.035	0.25	15	0.18
Defensivo	2	0.035	0.25	15	0.18

Tipo de conductor	Porcentaje	Velocidad mínima	Tiempo de cambio de carril	Factor de mejora de velocidad	Probabilidad de cortesía
Novato	10	2	3	0.2	0.3
Experimentado	60	2	2	0.1	0.2
Agresivo	15	2	1	0.05	0.2
Defensivo	15	2	2	0.1	0.3

Tabla 7.1. Parámetros de los diferentes tipos de conductores

7.2.1 Modelo de cambio de carril

En este apartado se pretende comprobar que los aspectos más importantes descritos en el modelo de cambio de carril han sido correctamente implementados.

Duración temporal de un cambio de carril

El primer aspecto a considerar es la duración temporal fija y predefinida que tienen las maniobras de cambio de carril que llevan a cabo los vehículos. Para su comprobación, nos basamos en el análisis de un cambio de carril ocurrido en el tramo 0 del escenario usado en estas pruebas.

Toda la información correspondiente a los cambios de carril ocurridos en este tramo se encuentra en el fichero *Tramo0.dat*, del cual se muestra a continuación parte de su contenido.

Tipo	Instante	Tramo	Vehículo	Posición	Origen	Destino	Cortesía	Gap_del	Gap_tra
Obl.	16.50	0	5	337.61	0	1		11.851	0.000
Mej.	16.50	0	33	266.76	0	1		67.846	0.000
Mej.	18.00	0	27	247.72	2	1		22.226	101.371
Mej.	21.00	0	7	97.73	2	1		53.730	0.000
Mej.	21.00	0	60	346.83	1	0		0.000	49.376

Concretamente, se va a analizar el cambio de carril realizado por el vehículo 27 desde el carril 2 hasta el 1 en el instante 18.0 de simulación. Para ello, se presenta a continuación algunas de las líneas del fichero *micro.sal* que van a mostrar la duración temporal de este cambio de carril.

Instante	Situación	Cod_veh	Tramo	Nodo	Mov	Aceleración	Velocidad	Posición	Carril	D_influencia	D_seguridad
18.00	TRAMO	27	0	0	0	-0.48	5.14	247.72	2	12.81	5.31
18.50	TRAMO	27	0	0	0	0.50	5.39	250.42	1	10.82	3.32
18.50	TRAMO	61	0	0	0	-0.37	4.96	250.20	2	11.59	4.09
19.00	TRAMO	27	0	0	0	0.47	5.62	253.23	1	10.80	3.30
19.00	TRAMO	61	0	0	0	-0.30	4.81	252.60	2	10.99	3.49
19.50	TRAMO	27	0	0	0	0.44	5.84	256.15	1	10.78	3.28
19.50	TRAMO	61	0	0	0	-0.22	4.70	254.95	2	10.44	2.94
20.00	TRAMO	27	0	0	0	0.42	6.05	259.18	1	10.75	3.25
20.00	TRAMO	61	0	0	0	-0.14	4.63	257.27	2	9.95	2.45
20.50	TRAMO	27	0	0	0	0.40	6.25	262.30	1	10.73	3.23

Los resultados obtenidos muestran cómo el vehículo 27 pasa a circular por el carril número 1 en el instante de simulación 18.5, justamente en ese instante aparece sobre el viario un nuevo vehículo ficticio de código 61 que circula por el carril 2 en la misma posición que ocuparía el vehículo 27 si no hubiera cambiado de carril.

Ambos vehículos circulan durante un tiempo de 2 segundos prácticamente en paralelo, con la única diferencia debida a las diferentes velocidades que se pueden desarrollar en cada carril. Es decir, durante esos 2 segundos es como si el vehículo 27 ocupara los dos carriles, por lo que ésta es la duración de la maniobra de cambio de carril.

Al llegar al instante de simulación 20.5, el vehículo 61 desaparece del viario, por lo que se da por finalizada la maniobra de cambio de carril. La duración temporal del cambio de carril ha sido de 2 segundos, que se corresponde con un conductor experimentado o defensivo.

Aumento de la velocidad en un cambio de carril por mejora

Para demostrar cómo aumenta la velocidad de un vehículo cuando realiza un cambio de carril por mejora, se analizan las mismas líneas del fichero *micro.sal* empleadas para mostrar la duración temporal del cambio de carril.

En dichas líneas se aprecia que cuando el vehículo 27 circulaba por el carril 2 lo hacía a una velocidad de 5.14 m/s y que se encontraba aplicando una aceleración negativa, es decir, el vehículo estaba frenando.

De haber continuado por dicho carril, la velocidad del vehículo habría evolucionado tal y como la del vehículo 61, es decir, habría descendido hasta una velocidad de 4.63 m/s en el instante 20. Sin embargo, se observa que la velocidad del vehículo 27 en ese mismo instante es de 6.05 m/s, por lo que la mejora que ha experimentado su velocidad al cambiar de carril es evidente. Esta evolución de ambas velocidades se aprecia en la figura 7.1.

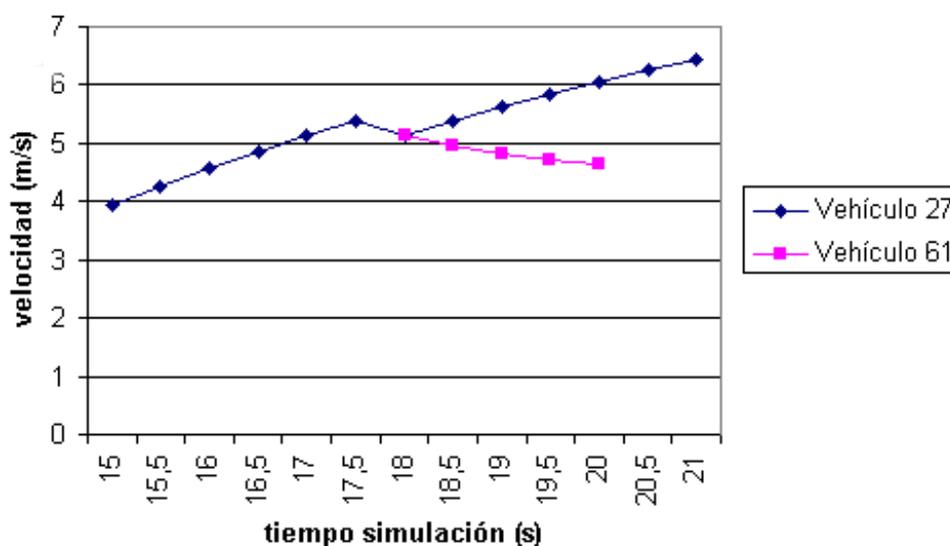


Figura 7.1. Incremento de velocidad tras cambio de carril por mejora

La gráfica de la figura 7.1 describe la evolución de la velocidad del vehículo 27 antes y después de cambiar de carril. Se aprecia que la velocidad en el nuevo carril es superior a la que habría llevado de permanecer en el mismo carril, que es la que lleva el vehículo ficticio 61.

Distancias con vehículos precedentes y sucesores

En este punto se va a comprobar que las distancias mostradas en los ficheros *TramoN.dat* entre el vehículo que va a cambiar de carril y aquellos que van a ser sus futuros vehículos precedente y sucesor en el nuevo carril son las correctas.

El cambio elegido para mostrar estas distancias es el mismo que está resaltado en el fichero *Tramo0.dat* del apartado de la duración temporal de un cambio de carril. A continuación, se muestran los datos correspondientes a todos los vehículos que circulan por el tramo 0 en el instante 18 de simulación, que es cuando el vehículo 27 decide realizar el cambio de carril, extraídos del fichero *micro.sal*.

Instante	Situación	Cod_veh	Tramo	Nodo	Mov	Aceleracion	Velocidad	Posicion	Carril	D_influencia	D_seguridad
18.00	TRAMO	2	0	0	0	0.24	8.19	119.94	2	16.47	8.97
18.00	TRAMO	5	0	0	0	-0.72	6.39	347.70	1	12.69	5.19
18.00	TRAMO	7	0	0	0	0.29	7.41	74.31	2	17.23	9.73
18.00	TRAMO	9	0	0	0	0.32	7.11	205.98	0	15.43	7.93
18.00	TRAMO	13	0	0	0	0.34	6.79	318.95	2	13.17	5.67
18.00	TRAMO	14	0	0	0	-1.72	3.66	359.68	2	10.41	2.91
18.00	TRAMO	15	0	0	0	0.38	6.44	190.24	0	14.42	6.92
18.00	TRAMO	20	0	0	0	0.42	6.05	175.90	2	10.58	3.08
18.00	TRAMO	25	0	0	0	-1.37	3.62	358.86	1	10.37	2.87
18.00	TRAMO	27	0	0	0	-0.48	5.14	247.72	2	12.81	5.31
18.00	TRAMO	33	0	0	0	0.64	4.45	272.95	1	11.33	3.83
18.00	TRAMO	34	0	0	0	0.57	4.87	51.89	2	11.89	4.39
18.00	TRAMO	36	0	0	0	0.62	4.58	150.46	2	11.50	4.00
18.00	TRAMO	37	0	0	0	0.67	4.27	127.17	0	11.11	3.61
18.00	TRAMO	42	0	0	0	0.92	3.16	281.28	0	9.92	2.42
18.00	TRAMO	45	0	0	0	1.80	2.70	262.70	2	9.54	2.04
18.00	TRAMO	46	0	0	0	0.27	7.69	348.93	0	16.94	9.44
18.00	TRAMO	47	0	0	0	-0.06	2.97	271.32	0	9.76	2.26
18.00	TRAMO	50	0	0	0	1.80	1.80	143.35	1	8.96	1.46
18.00	TRAMO	51	0	0	0	1.80	1.80	103.35	2	8.96	1.46
18.00	TRAMO	56	0	0	0	1.80	0.90	308.45	1	1.00	1.00
18.00	TRAMO	57	0	0	0	1.80	0.90	305.45	2	1.00	1.00

Tanto el vehículo 27 como los vehículos que van a ser su sucesor (vehículo 50) y su precedente (vehículo 33) en el nuevo carril (carril 1), han sido marcados en negrita. Observando las posiciones en las que se encuentran los vehículos y teniendo en cuenta que la longitud de cada uno de ellos es de 3 metros, se obtienen las distancias especificadas en el fichero *Tramo0.dat*.

Cambios de carril obligatorios

Para ilustrar con un ejemplo los cambios de carril obligatorios, se ha elegido el realizado por el vehículo 5 en el instante de simulación 16.5, que es el primero que aparece en el fichero *Tramo0.dat* ya presentado en el punto correspondiente a la duración temporal de los cambios de carril.

Una vez que este vehículo haya finalizado su desplazamiento por el tramo 0, va a pasar a incorporarse al tramo 2, para ello, tiene que circular por los movimientos 1 y 2 del nodo 2. Sin embargo, para enlazar con dichos movimientos, es necesario que circule

por el carril 1 del tramo 0 (ver figura 7.8), por lo que necesita realizar una maniobra de cambio de carril que le lleve al carril adecuado.

En el fichero *vehiculo5.dat*, del que se extraen a continuación algunas líneas, se muestra todo el recorrido que hace dicho vehículo. En particular, se han resaltado aquellos momentos en los que el vehículo realiza una transición entre carriles, entre un carril y un movimiento, o entre movimientos.

Instante	Cod_veh	Tramo	Nodo	Mov	Aceleracion	Velocidad	Posicion	Carril	D_influencia	D_seguridad	D_mlc	Tramo
16.00	5	0	0	0	0.32	7.11	333.98	0	15.72	8.22	340.94	3
16.50	5	0	0	0	0.30	7.26	337.61	0	16.04	8.54	340.30	3
17.00	5	0	0	0	-0.48	7.03	341.12	1	12.78	5.28	339.68	3
17.50	5	0	0	0	-0.54	6.75	344.50	1	12.62	5.12	339.68	3
18.00	5	0	0	0	-0.72	6.39	347.70	1	12.69	5.19	339.68	3
18.50	5	0	0	0	-1.00	5.90	350.64	1	12.59	5.09	339.68	3
19.00	5	0	0	0	-1.36	5.22	353.25	1	12.16	4.66	339.68	3
19.50	5	0	0	0	-1.75	4.34	355.42	1	11.43	3.93	347.20	3
20.00	5	0	0	0	-2.03	3.33	357.09	1	10.64	3.14	349.12	3
20.50	5	0	0	0	-1.89	2.38	358.28	1	10.07	2.57	351.03	3
21.00	5	0	0	0	-1.29	1.74	359.15	1	9.81	2.31	352.37	3
21.50	5	0	0	0	-1.09	1.19	359.74	1	8.70	1.20	353.01	3
22.00	5	0	0	0	-2.39	0.00	359.74	1	1.00	1.00	354.34	3
22.50	5	0	0	0	0.00	0.00	359.74	1	1.00	1.00	362.25	3
23.00	5	0	0	0	0.00	0.00	359.74	1	1.00	1.00	362.25	3
23.50	5	0	0	0	0.00	0.00	359.74	1	1.00	1.00	362.25	3
24.00	5	0	0	0	0.00	0.00	359.74	1	1.00	1.00	362.25	3
24.50	5	0	0	0	0.00	0.00	359.74	1	1.00	1.00	362.25	3
25.00	5	0	0	0	0.00	0.00	359.74	1	1.00	1.00	362.25	3
25.50	5	0	0	0	0.00	0.00	359.74	1	1.00	1.00	362.25	3
26.00	5	0	0	0	0.00	0.00	359.74	1	1.00	1.00	362.25	3
26.50	5	0	0	0	0.00	0.00	359.74	1	1.00	1.00	362.25	3
27.00	5	0	0	0	0.00	0.00	359.74	1	1.00	1.00	362.25	3
27.50	5	0	0	0	0.00	0.00	359.74	1	1.00	1.00	362.25	3
28.00	5	0	0	0	0.00	0.00	359.74	1	1.00	1.00	362.25	3
28.50	5	0	0	0	0.00	0.00	359.74	1	1.00	1.00	362.25	3
29.00	5	0	0	0	0.00	0.00	359.74	1	1.00	1.00	362.25	3
29.50	5	0	0	0	0.00	0.00	359.74	1	1.00	1.00	362.25	3
30.00	5	0	0	0	1.80	0.90	360.19	1	1.00	1.00	362.25	3
30.50	5	0	0	0	1.80	1.80	361.09	1	8.96	1.46	362.13	3
31.00	5	0	0	0	1.39	2.50	362.34	1	9.39	1.89	353.82	3
31.50	5	0	0	0	0.68	2.83	363.76	1	9.65	2.15	352.97	3
32.00	5	0	2	1	0.69	3.18	2.10	1	11.45	3.95	352.97	3
32.50	5	0	2	1	0.69	3.52	3.86	1	11.95	4.45	352.97	3
33.00	5	0	2	1	0.69	3.86	5.79	1	12.48	4.98	352.97	3
33.50	5	0	2	1	0.69	4.21	7.90	1	13.05	5.55	352.97	3
34.00	5	0	2	1	0.69	4.55	10.17	1	13.65	6.15	352.97	3
34.50	5	0	2	2	0.69	4.89	2.55	1	14.28	6.78	352.97	3
35.00	5	0	2	2	0.69	5.24	5.17	1	14.95	7.45	352.97	3
35.50	5	0	2	2	0.69	5.58	7.96	1	15.65	8.15	352.97	3
36.00	5	0	2	2	0.69	5.92	10.92	1	16.39	8.89	352.97	3
37.00	5	2	2	2	0.69	6.61	6.65	1	9.80	2.30	394.72	-1
37.50	5	2	2	2	-0.13	6.54	9.92	1	14.61	7.11	400.07	-1

Cambios de carril por cortesía

En este punto se va a analizar un cambio de carril por cortesía, concretamente el que lleva a cabo el vehículo 57 en el instante de tiempo 28, cuya información se aprecia en el fichero *Cortesia.dat*, del que se muestran las primeras líneas a continuación:

Tipo	Instante	Tramo	Vehiculo	Posicion	Origen	Destino	Veh_detras	Distancia	Intentos
Mej.	19.50	7	12	207.69	0	1	63	2.989487	3
Mej.	26.00	7	8	140.81	0	1	105	3.531693	2
Obl.	27.50	6	84	412.84	2	1	122	3.489746	6
Obl.	28.00	0	57	353.70	2	1	33	11.025024	4
Mej.	32.50	7	6	206.39	2	1	153	3.355011	5
Mej.	33.00	7	87	67.83	1	0	174	1.478817	2

El primer punto de interés que se debe analizar es que el cambio del carril 2 al carril 1 se ha producido tras 4 intentos fallidos, es decir, el vehículo 57 ya intentó realizar esta maniobra en los instantes de simulación 26, 26.5, 27 y 27.5, sin lograr llevarlo a cabo.

El cambio de carril se produce en el instante de simulación 28 cuando el vehículo que sería su sucesor en el nuevo carril (vehículo 33) le permite el cambio de carril. La distancia entre ambos vehículos en ese instante es de 11.02 metros, la cuál, teniendo en cuenta las velocidades a las que circulan ambos vehículos no permite aceptar el gap habitual, como se demuestra a continuación.

Instante	Situacion	Cod_veh	Tramo	Nodo	Mov	Aceleracion	Velocidad	Posicion	Carril	D_influencia	D_seguridad
28.00	TRAMO	33	0	0	0	0.24	8.14	339.68	1	17.96	10.46
28.00	TRAMO	57	0	0	0	-1.81	3.73	353.70	2	10.49	2.99

Se observa cómo la distancia entre ambos vehículos en el momento de inicio del cambio de carril está comprendida entre los límites dados por la distancia de influencia y la de seguridad, y muy próxima a la primera de ellas, esto hace que la probabilidad de rechazar este cambio de carril sea muy alta, como así ha sucedido.

Sin embargo, esos 11.02 metros son una distancia más que suficiente para que el vehículo 33 pueda frenar hasta alcanzar la velocidad del vehículo 57 antes de colisionar con él, es decir, se cumple la condición del gap de cortesía vista en el apartado 2.4.2.3.3.

En este ejemplo concreto de cambio de carril por cortesía, debido a que la velocidad del vehículo sucesor es superior a la del vehículo que cambia de carril, el vehículo 33 (sucesor) incluso se ve forzado a frenar para permitir el paso al vehículo 57. La evolución de las velocidades de ambos vehículos se aprecia en la figura 7.2.

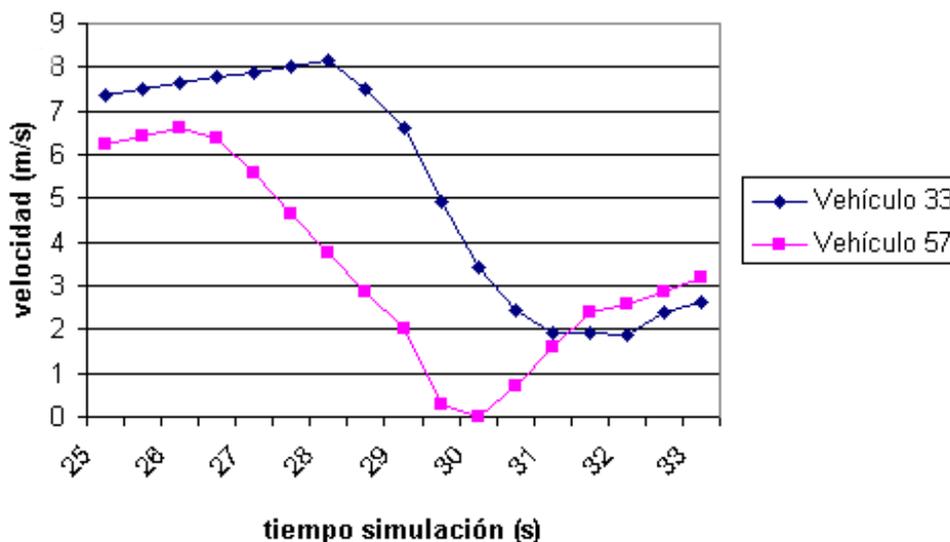


Figura 7.2. Evolución de velocidades en cambio de carril por cortesía

En esta gráfica comparativa de velocidades, se aprecia que a partir del momento en el que el vehículo 57 cambia de carril (instante 28), el vehículo 33 se ve obligado a disminuir su velocidad, hasta que logra situarse a una distancia que le permita circular con comodidad.

Cambios de carril para la realización de aparcamientos

Otro de los aspectos que merece la pena verificar cómo se lleva a cabo es el proceso que sigue un vehículo para desplazarse desde el carril por el que circula hasta aquel en el que se encuentra el aparcamiento que tiene asignado. Estos cambios están considerados dentro de los cambios obligatorios.

Como ejemplo, se van a tomar las maniobras de cambio de carril que realiza el vehículo 0 en el tramo 2 para aparcar en la posición 290 del carril 0 de ese tramo, cuando estaba circulando previamente por el carril número 2. Estos cambios de carril quedan reflejados en el fichero *Tramo2.dat*.

Tipo	Instante	Tramo	Vehiculo	Posicion	Origen	Destino	Cortesia	Gap_del	Gap_tra
Mej.	60.00	2	115	8.45	0	1	15.094	5.887	
Mej.	62.00	2	96	10.69	0	1	6.445	8.127	
Obl.	63.00	2	0	251.07	2	1	5.185	53.149	
Obl.	65.50	2	0	277.03	1	0	0.000	70.184	
Obl.	65.50	2	19	221.53	1	0	52.501	14.683	
Obl.	68.00	2	23	252.45	2	1	61.696	37.140	

En la figura 7.3 se muestra la evolución de la posición del vehículo en el tramo, así como en el carril por el que se desplaza. En ella se representa cómo el vehículo 0 cambia del carril 2 al 0 sin apenas detenerse en el carril 1. También cabe destacar cómo a pesar de que el tiempo empleado en ambos cambios es el mismo (2 segundos), el espacio recorrido durante el primero de ellos es mayor, ya que también lo es la velocidad del vehículo en esos momentos.

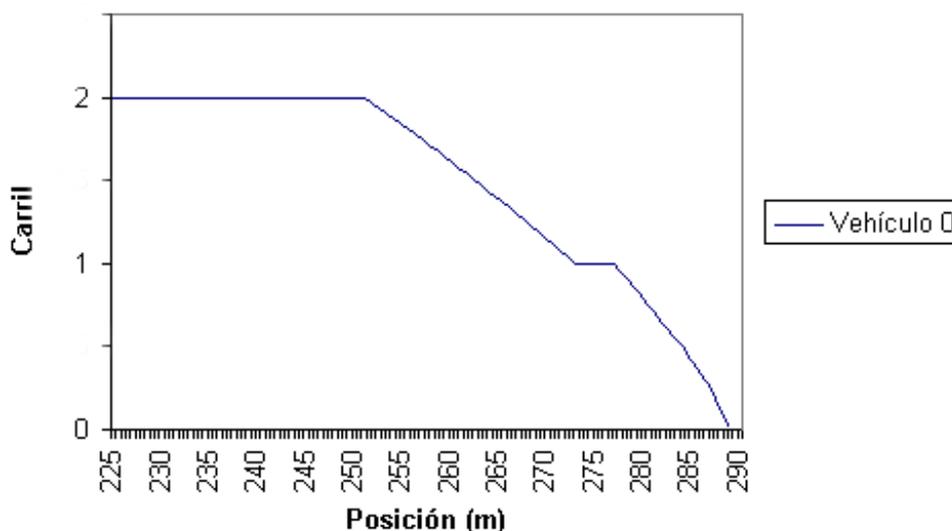


Figura 7.3. Cambios de carril para aparcamiento

7.2.2 Función de generación de vehículos

En este apartado se van a validar aquellas operaciones que son necesarias para la generación de nuevos vehículos. La generación de números pseudoaleatorios según una determinada función de densidad de probabilidad ya fue comprobada en el apartado

3.3.5, mientras que la generación de posiciones de generación se realiza según unos números aleatorios, por lo que no es necesario validarla.

Cálculo del área de la sección del trapecoide

Para comprobar que la generación de vehículos según una función trapezoidal es correcta, se ha realizado una prueba simple en la que se escriben en un fichero los valores del área de la sección correspondiente a cada intervalo de tiempo en los que se divide un trapecoide de la forma mostrada en la figura 7.4.

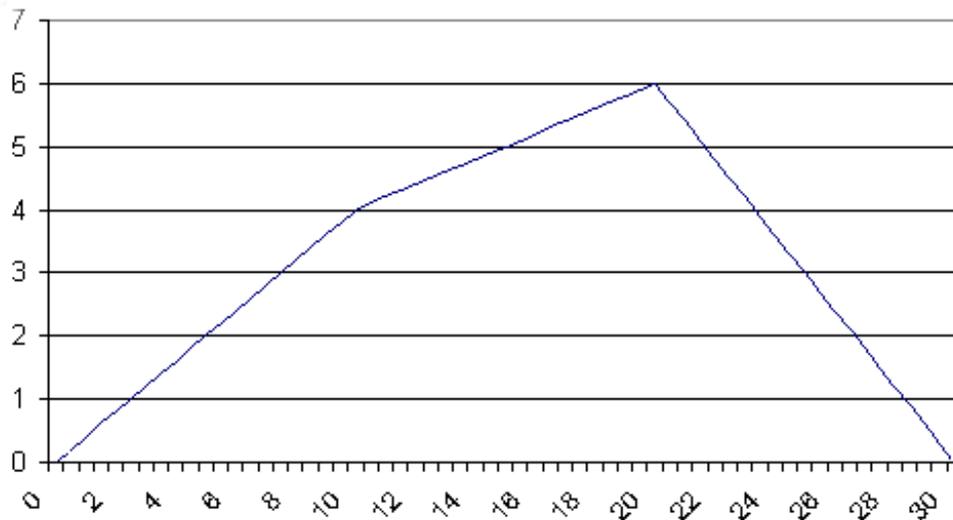


Figura 7.4. Función de generación trapezoidal

El área total de este trapecoide es de 100 unidades, por lo que la suma de los trozos de área que se van calculando en cada intervalo de tiempo debe ser igual a esta cantidad. En el diagrama de barras de la figura 7.5 se observa el valor del área del trozo de trapecoide correspondiente a cada intervalo de tiempo. Se aprecia cómo la forma es idéntica a la figura 7.4, mientras que la altura de cada barra es justamente la mitad, ya que la base de cada barra es 0.5 segundos.

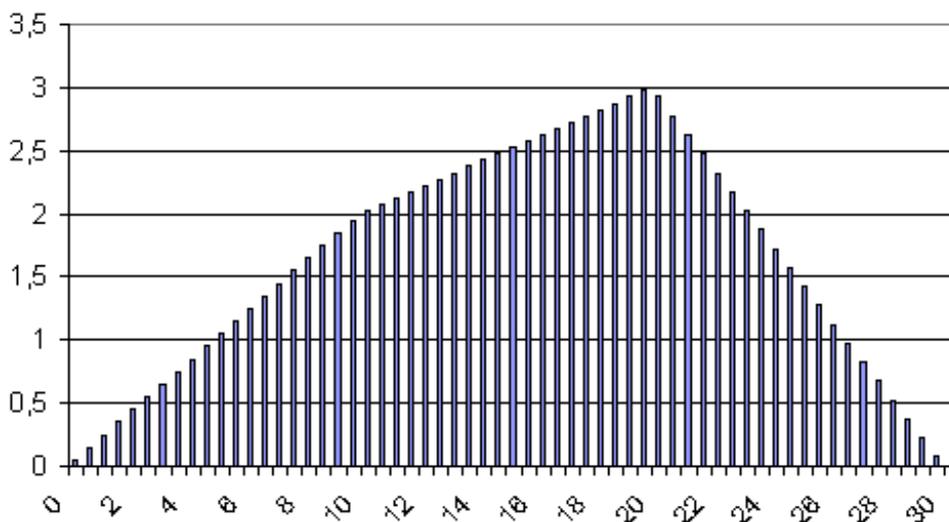


Figura 7.5. Área de la sección de trapecoide en cada intervalo

Generación de vehículos pendientes

El otro aspecto que merece la pena validar en la generación de vehículos es el relacionado con aquellos vehículos que no han podido ser introducidos en el viario en su momento de generación. Para ello, realizamos una prueba empleando el trapecio del punto anterior para generar vehículos en el tramo 0 en posiciones aleatorias.

Si todos los vehículos se pudieran generar en su intervalo de tiempo correspondiente, a partir del segundo 30 no se generarían más vehículos, sin embargo esto no es así. En el fichero *pos_gen.dat* se hace referencia a generaciones de vehículos posteriores a ese instante, concretamente se va a analizar la generación del vehículo 127 en el instante 32.5 en la posición 157 del carril 2 (marcado en negrita en el siguiente fragmento del fichero).

Instante	Vehiculo	Tramo	Carril	Posicion
30.5	123	0	2	109.0000
30.5	124	0	1	324.0000
32.5	127	0	2	157.0000
32.5	128	0	1	88.0000
33.0	130	0	1	59.0000

Realmente, ese vehículo debía haber sido generado en el instante 23 de simulación, sin embargo, debido a incompatibilidades con otros vehículos que circulaban por ese carril en ese instante, no fue posible generarlo. A continuación se muestran los vehículos que circulaban por el carril 2 del tramo 0 en el instante 23 (extraídos del fichero *micro.sal*):

Instante	Situacion	Cod_veh	Tramo	Nodo	Mov	Aceleracion	Velocidad	Posicion	Carril	D_influencia	D_seguridad
23.00	TRAMO	0	0	0	0	-1.77	3.02	361.39	2	9.80	2.30
23.00	TRAMO	6	0	0	0	0.37	4.73	127.85	2	11.70	4.20
23.00	TRAMO	10	0	0	0	0.38	6.45	192.50	2	10.33	2.83
23.00	TRAMO	11	0	0	0	-3.50	1.97	243.66	2	9.94	2.44
23.00	TRAMO	13	0	0	0	-0.11	6.41	182.92	2	14.37	6.87
23.00	TRAMO	14	0	0	0	-0.58	5.69	214.18	2	12.70	5.20
23.00	TRAMO	19	0	0	0	0.48	3.75	98.95	2	10.51	3.01
23.00	TRAMO	21	0	0	0	-0.18	5.76	260.85	2	10.48	2.98
23.00	TRAMO	22	0	0	0	-0.60	3.91	83.06	2	10.36	2.86
23.00	TRAMO	23	0	0	0	-0.23	5.11	147.25	2	10.05	2.55
23.00	TRAMO	27	0	0	0	0.22	5.05	137.79	2	12.15	4.65
23.00	TRAMO	32	0	0	0	0.64	3.37	90.78	2	10.12	2.62
23.00	TRAMO	36	0	0	0	0.43	4.44	48.84	2	11.32	3.82
23.00	TRAMO	37	0	0	0	-0.46	4.41	310.00	2	10.06	2.56
23.00	TRAMO	39	0	0	0	-0.06	4.29	41.44	2	11.13	3.63
23.00	TRAMO	48	0	0	0	-0.56	4.43	156.10	2	11.27	3.77
23.00	TRAMO	59	0	0	0	0.29	4.16	106.37	2	10.97	3.47
23.00	TRAMO	62	0	0	0	0.50	5.38	333.15	2	12.34	4.84
23.00	TRAMO	64	0	0	0	0.50	5.38	273.15	2	12.60	5.10
23.00	TRAMO	69	0	0	0	0.31	4.49	114.63	2	11.38	3.88
23.00	TRAMO	71	0	0	0	0.57	4.87	59.89	2	11.89	4.39
23.00	TRAMO	73	0	0	0	0.67	4.28	317.55	2	11.12	3.62
23.00	TRAMO	74	0	0	0	0.55	2.98	289.39	2	9.77	2.27
23.00	TRAMO	78	0	0	0	0.82	3.57	227.06	2	10.76	3.26
23.00	TRAMO	80	0	0	0	0.82	3.57	296.06	2	10.32	2.82
23.00	TRAMO	82	0	0	0	0.92	3.16	166.28	2	9.92	2.42
23.00	TRAMO	86	0	0	0	1.80	1.80	246.35	2	8.96	1.46
23.00	TRAMO	91	0	0	0	1.80	0.90	72.45	2	1.00	1.00

En este fichero se aprecia que el vehículo 48 circula en ese instante por la posición 156.1, lo que quiere decir que si se introdujera el vehículo 127 en la posición 157, ambos vehículos se solaparían por lo que no es posible realizar dicha generación.

En el instante 32.5, la situación en el carril es la siguiente:

Instante	Situacion	Cod_veh	Tramo	Nodo	Mov	Aceleracion	Velocidad	Posicion	Carril	D_influencia	D_seguridad
32.50	TRAMO	6	0	0	0	0.30	5.70	175.54	2	13.14	5.64
32.50	TRAMO	10	0	0	0	0.30	5.20	249.62	2	12.37	4.87
32.50	TRAMO	11	0	0	0	0.36	6.62	282.32	2	14.75	7.25
32.50	TRAMO	13	0	0	0	-0.13	4.97	241.90	2	12.03	4.53
32.50	TRAMO	14	0	0	0	0.38	5.65	259.19	2	13.06	5.56
32.50	TRAMO	19	0	0	0	0.37	6.51	147.47	2	11.58	4.08
32.50	TRAMO	21	0	0	0	0.28	5.92	312.74	2	13.50	6.00
32.50	TRAMO	22	0	0	0	0.31	5.54	126.14	2	12.88	5.38
32.50	TRAMO	23	0	0	0	0.08	6.36	196.97	2	14.28	6.78
32.50	TRAMO	27	0	0	0	0.25	6.07	186.33	2	13.77	6.27
32.50	TRAMO	32	0	0	0	0.37	6.00	136.25	2	13.64	6.14
32.50	TRAMO	37	0	0	0	-0.27	4.45	360.21	2	11.33	3.83
32.50	TRAMO	39	0	0	0	0.46	5.67	85.37	2	10.78	3.28
32.50	TRAMO	48	0	0	0	-0.11	6.40	207.69	2	10.04	2.54
32.50	TRAMO	53	0	0	0	0.41	5.24	73.05	2	12.43	4.93
32.50	TRAMO	63	0	0	0	-0.09	5.33	230.25	2	10.03	2.53
32.50	TRAMO	64	0	0	0	0.21	6.25	324.31	2	14.09	6.59
32.50	TRAMO	69	0	0	0	0.26	5.40	163.47	2	12.67	5.17
32.50	TRAMO	74	0	0	0	-0.03	6.38	338.09	2	9.63	2.13
32.50	TRAMO	78	0	0	0	0.37	6.15	269.81	2	13.91	6.41
32.50	TRAMO	80	0	0	0	-0.49	6.06	348.67	2	11.92	4.42
32.50	TRAMO	82	0	0	0	-0.26	5.97	219.57	2	10.53	3.03
32.50	TRAMO	86	0	0	0	0.30	7.26	297.61	2	12.03	4.53
32.50	TRAMO	123	0	0	0	0.92	3.16	113.28	2	9.92	2.42
32.50	TRAMO	126	0	0	0	-0.78	4.42	104.24	2	10.87	3.37

En este fichero se aprecia que el vehículo que sería el sucesor del 127 en el carril, sería el 19, que se encontraría a una distancia superior a la distancia de seguridad, mientras que el vehículo predecesor sería el 69, que estaría a una distancia superior a 3 metros del nuevo vehículo. Por lo tanto, se dan las condiciones adecuadas para que el vehículo 127 sea generado en el instante 32.5.

7.2.3 Estadísticas

En este apartado se va a verificar que los datos que son almacenados en los ficheros de las distintas estadísticas del sistema son correctos, para lo que se va a analizar un ejemplo de cada uno de ellos. El escenario y la función de generación empleados para estas pruebas son los mismos que los del apartado 7.2.2, por lo que se aprovechan algunos de los ficheros ya expuestos en dicho apartado.

cam_veh

Este fichero contiene el número de cambios de carril que realiza cada vehículo, como ejemplo vemos los dos primeros vehículos señalados en dicho fichero:

Vehiculo	Cambios
0	2
1	3

Los dos cambios de carril que realiza el vehículo 0, del 1 al 2 en el instante 5.5 y del 2 al 1 en el 24, se pueden apreciar en el fichero *vehiculo0.dat*:

Instante	Cod_veh	Tramo	Nodo	Mov	Aceleracion	Velocidad	Posicion	Carril	D_influencia	D_seguridad	D_mlc	Tramo
5.00	0	0	0	0	-0.41	3.36	241.74	1	11.00	3.50	351.11	5
5.50	0	0	0	0	0.78	3.75	243.62	2	10.51	3.01	350.63	5
23.50	0	0	0	0	-1.49	2.27	362.52	2	9.24	1.74	352.14	5
24.00	0	0	0	0	-1.25	1.65	363.35	1	8.89	1.39	353.27	5

Los 3 cambios del vehículo 1 se aprecian en su correspondiente fichero:

Instante	Cod_veh	Tramo	Nodo	Mov	Aceleracion	Velocidad	Posicion	Carril	D_influencia	D_seguridad	D_mlc	Tramo
15.50	1	0	0	0	-0.04	4.59	74.06	2	9.95	2.45	349.85	3
16.00	1	0	0	0	0.57	4.87	76.50	1	10.28	2.78	350.30	3
25.00	1	0	0	0	0.06	3.79	114.67	1	10.56	3.06	351.71	3
25.50	1	0	0	0	0.70	4.14	116.74	0	10.95	3.45	350.64	3
184.50	1	5	2	0	-2.37	4.94	231.57	0	11.98	4.48	856.11	-1
185.00	1	5	2	0	0.52	5.20	234.17	1	12.36	4.86	859.86	-1

num_cambios

La estadística que representa el número de cambios de cada tipo que se producen en cada tramo, se ha validado contando dicho número de cambios en cada fichero *TramoN.dat*. Sin embargo, debido a la longitud de dichos ficheros y a que no aportan nada nuevo, se ha considerado conveniente no exponer sus contenidos. No obstante, dichos valores son correctos.

pos_apa

Para verificar que las posiciones en las que se producen los aparcamientos se corresponden con las indicadas en este fichero, se va a examinar la evolución del primero de los vehículos que aparecen en él.

Instante	Vehiculo	Tramo	Posicion
68.0	9	5	280.0
70.5	20	5	260.0
75.0	8	5	300.0

El vehículo 9 tiene la siguiente evolución a lo largo del tiempo:

Instante	Cod_veh	Tramo	Nodo	Mov	Aceleracion	Velocidad	Posicion	Carril	D_influencia	D_seguridad	D_mlc	Tramo
67.00	9	5	2	0	-3.50	5.54	276.21	0	12.89	5.39	851.62	-1
67.50	9	5	2	0	-3.50	3.79	278.11	0	10.56	3.06	858.04	-1
268.50	9	5	2	0	1.09	1.09	280.82	0	8.67	1.17	862.71	-1
269.00	9	5	2	0	1.80	1.99	281.81	0	9.07	1.57	866.48	-1

En este fichero se aprecia que el vehículo 9 desaparece del viario en la posición 280 aproximadamente del tramo 5 en el instante 68 de simulación, lo que coincide con los datos del aparcamiento que realiza. Además, se comprueba que unos 200 segundos más tarde se vuelve a reincorporar al tramo en la misma posición, lo que quiere decir que sale del aparcamiento.

pos_gen

En este fichero se recogen las posiciones de generación de los vehículos, para analizarlo, se estudia el mismo vehículo que en el punto de *Generación de vehículos pendientes* del apartado 7.2.2, es decir, el vehículo 127.

En ese fichero, se ha visto que el vehículo 127 se genera en el instante 32.5 en la posición 157 del carril 2 del tramo 0, cosa que es totalmente compatible con la primera del fichero *vehiculo127.dat* en la que se indica que en el instante siguiente el vehículo se encuentra en la posición 157.45 de ese mismo tramo y carril.

Instante	Cod_veh	Tramo	Nodo	Mov	Aceleracion	Velocidad	Posicion	Carril	D_influencia	D_seguridad	D_mlc	Tramo
33.00	127	0	0	0	1.80	0.90	157.45	2	1.00	1.00	362.25	3
33.50	127	0	0	0	1.80	1.80	158.35	2	8.96	1.46	362.13	3
34.00	127	0	0	0	1.80	2.70	159.70	2	9.54	2.04	353.82	3
...												

veh_gen

La prueba de este fichero se va a realizar para los instantes de simulación 15.5 y 16, en los que se deben haber generado 2 y 3 nuevos vehículos en el tramo 0 respectivamente.

Instante	Tramo0	Tramo1	Tramo2	Tramo3	Tramo4	Tramo5	Tramo6	Tramo7	Tramo8	Tramo9
15.0	4	0	0	0	0	0	0	0	0	0
15.5	2	0								
16.0	3	0								
16.5	1	0	0	0	0	0	0	0	0	0
...										

La verificación de que esto es ciertamente así, se obtiene al analizar las siguientes líneas del fichero *Generacion0.dat*:

...					
15.00	0	0	43	3	0
15.00	0	0	44	3	0
15.00	0	1	45	5	0
15.00	0	0	46	3	0
15.50	0	0	47	5	0
15.50	0	2	48	3	0
16.00	0	0	51	5	0
16.00	0	1	52	5	0
16.00	0	1	53	3	0
16.50	0	1	54	3	0
...					

tpo_sem

En este fichero se muestra el tiempo que permanecen detenidos los vehículos debido a los semáforos que regulan la circulación en los distintos tramos. Concretamente se va a analizar el tiempo que permanece detenido el vehículo 0 delante del semáforo del tramo 0.

Vehiculo Tramo Inicio Final Tiempo

0	0	24.5	29.5	5.0
44	0	24.5	29.5	5.0
3	0	25.0	30.0	5.0
25	0	26.0	30.0	4.0

...

Para comprobar que eso es realmente así, se estudia el fichero *vehiculo0.dat*, en el que se aprecia que dicho vehículo está detenido desde el instante 24.5 hasta el 29.5, y que inmediatamente después de arrancar pasa a circular por el siguiente nodo, lo que indica que, en efecto, ha estado detenido a causa del semáforo.

Instante Cod_veh Tramo Nodo Mov Aceleracion Velocidad Posicion Carril D_influencia D_seguridad D_mlc Tramo

23.50	0	0	0	0	-1.49	2.27	362.52	2	9.24	1.74	352.14	5
24.00	0	0	0	0	-1.25	1.65	363.35	1	8.89	1.39	353.27	5
24.50	0	0	0	0	-3.30	0.00	363.35	1	1.00	1.00	353.27	5
25.00	0	0	0	0	0.00	0.00	363.35	1	1.00	1.00	353.27	5
25.50	0	0	0	0	0.00	0.00	363.35	1	1.00	1.00	362.25	5
26.00	0	0	0	0	0.00	0.00	363.35	1	1.00	1.00	362.25	5
26.50	0	0	0	0	0.00	0.00	363.35	1	1.00	1.00	362.25	5
27.00	0	0	0	0	0.00	0.00	363.35	1	1.00	1.00	362.25	5
27.50	0	0	0	0	0.00	0.00	363.35	1	1.00	1.00	362.25	5
28.00	0	0	0	0	0.00	0.00	363.35	1	1.00	1.00	362.25	5
28.50	0	0	0	0	0.00	0.00	363.35	1	1.00	1.00	362.25	5
29.00	0	0	0	0	0.00	0.00	363.35	1	1.00	1.00	362.25	5
29.50	0	0	0	0	1.80	0.90	363.80	1	1.00	1.00	362.25	5
30.00	0	0	2	1	1.80	1.80	1.45	1	1.00	1.00	362.25	5
30.50	0	0	2	1	1.80	2.70	2.80	1	10.67	3.17	362.25	5

...

veh_semN

En este fichero se almacena el número de vehículos que hay detenidos en el semáforo correspondiente al tramo N en cada instante de simulación. En particular, se va a realizar el estudio del número de vehículos detenidos en el semáforo del tramo 0 en el instante de tiempo 27.5.

Instante Vehiculos

26.5	4
27.0	4
27.5	6
28.0	6
28.5	7

...

A continuación se muestran los vehículos que se encuentran detenidos en el tramo 0 en ese instante (extraídos del fichero *micro.sal*), bien porque estén influidos por el semáforo que se encuentran en rojo, o porque el vehículo de delante lo está. Se puede ver que son 6 los vehículos en esta situación.

Instante	Situacion	Cod_veh	Tramo	Nodo	Mov	Aceleracion	Velocidad	Posicion	Carril	D_influencia	D_seguridad
27.50	TRAMO	0	0	0	0	0.00	0.00	363.35	2	1.00	1.00
27.50	TRAMO	3	0	0	0	0.00	0.00	363.34	1	1.00	1.00
27.50	TRAMO	9	0	0	0	-0.58	0.00	355.97	0	1.00	1.00
27.50	TRAMO	25	0	0	0	0.00	0.00	359.74	0	1.00	1.00
27.50	TRAMO	44	0	0	0	0.00	0.00	363.34	0	1.00	1.00
27.50	TRAMO	77	0	0	0	-2.56	0.00	359.52	1	1.00	1.00

En estas líneas se observa cómo los vehículos frenan unos muy próximos de los otros al llegar al semáforo, pero sin llegar a colisionar entre ellos.

tpo_sis

Para ilustrar con un ejemplo el tiempo que permanece un vehículo en el sistema, se va a considerar el vehículo 2 que es el segundo que aparece en este fichero:

Vehiculo	Comienzo	Final	Tiempo
3	4.0	63.0	59.0
2	3.5	64.0	60.5
62	17.5	66.5	49.0

...

Para comprobar que efectivamente el vehículo 2 permanece esos 60.5 segundos en el sistema, se analiza su fichero correspondiente:

Instante	Cod_veh	Tramo	Nodo	Mov	Aceleracion	Velocidad	Posicion	Carril	D_influencia	D_seguridad	D_mlc	Tramo
4.00	2	0	0	0	1.80	0.90	168.45	0	1.00	1.00	362.25	3
4.50	2	0	0	0	1.80	1.80	169.35	0	8.96	1.46	362.13	3
...												
63.00	2	3	2	2	0.14	10.25	152.46	1	23.50	16.00	378.04	-1
63.50	2	3	2	2	0.14	10.32	157.61	1	23.70	16.20	377.63	-1

En este fichero se observa que el primer y último dato relativo a este vehículo corresponden con los segundos 4 y 63.5, por lo que efectivamente, los datos ofrecidos en el fichero *tpo_sis.dat* son ciertos.

tpo_tra

En este caso, nos vamos a centrar en el tiempo que dura el recorrido del vehículo 0 por el tramo 0, que se corresponde con los datos de la primera línea de este fichero:

Vehiculo	Tramo	Inicio	Final	Tiempo
0	0	2.0	29.5	27.5
3	0	4.0	29.5	25.5
44	0	15.0	29.5	14.5

...

Estudiando el fichero correspondiente a los datos del vehículo 0, se aprecia que la estadística anterior es correcta.

Instante	Cod_veh	Tramo	Nodo	Mov	Aceleracion	Velocidad	Posicion	Carril	D_influencia	D_seguridad	D_mlc	Tramo
2.50	0	0	0	0	1.80	0.90	234.45	1	1.00	1.00	362.25	5
3.00	0	0	0	0	1.80	1.80	235.35	1	8.96	1.46	362.13	5
...												
29.00	0	0	0	0	0.00	0.00	363.35	2	1.00	1.00	362.25	5
29.50	0	0	0	0	1.80	0.90	363.80	2	1.00	1.00	362.25	5
30.00	0	0	2	3	1.80	1.80	1.45	2	1.00	1.00	362.25	5
30.50	0	0	2	3	1.80	2.70	2.80	2	10.67	3.17	362.25	5

...

Estos datos también sirven para ilustrar un ejemplo del fichero *vel_med.dat* que contiene las velocidades medias de los vehículos en su recorrido por un tramo, ya que se usan los mismos datos contenidos en este fichero, realizando la división entre la distancia recorrida y el tiempo empleado para ello, que en este caso es de 4.72 m/s.

veh_car

La última de las estadísticas que va a ser validada es la correspondiente al número de vehículos que circulan en cada instante por cada uno de los carriles. En particular, se van a ver los vehículos que circulan por cada uno de los carriles del tramo 0 en el instante de simulación 9.

Instante	Tramo	Carril	Vehiculos
8.5	9	2	0
9.0	0	0	5
9.0	0	1	5
9.0	0	2	6
9.0	1	0	0

Para comprobar que estos datos son ciertos, se muestran a continuación las líneas del fichero *micro.sal* correspondientes al tramo 0 en el tiempo de simulación igual a 9, comprobando que efectivamente son 5 los vehículos que circulan por el carril 0, 5 por el 1 y 6 por el 2.

Instante	Situacion	Cod_veh	Tramo	Nodo	Mov	Aceleracion	Velocidad	Posicion	Carril	D_influencia	D_seguridad
9.00	TRAMO	0	0	0	0	0.46	5.73	260.99	2	13.19	5.69
9.00	TRAMO	1	0	0	0	0.47	5.62	35.96	2	11.80	4.30
9.00	TRAMO	2	0	0	0	0.50	5.38	188.15	0	12.22	4.72
9.00	TRAMO	3	0	0	0	0.53	5.13	269.46	1	12.27	4.77
9.00	TRAMO	4	0	0	0	0.62	4.58	92.46	1	11.46	3.96
9.00	TRAMO	6	0	0	0	0.67	4.27	57.17	2	11.99	4.49
9.00	TRAMO	7	0	0	0	0.74	3.94	131.03	1	11.25	3.75
9.00	TRAMO	8	0	0	0	0.82	3.57	222.06	0	10.28	2.78
9.00	TRAMO	9	0	0	0	0.92	3.16	258.28	0	9.92	2.42
9.00	TRAMO	10	0	0	0	0.92	3.16	119.28	1	9.92	2.42
9.00	TRAMO	11	0	0	0	1.80	2.70	172.70	2	9.54	2.04
9.00	TRAMO	12	0	0	0	1.80	2.70	257.70	1	9.54	2.04
9.00	TRAMO	13	0	0	0	1.80	1.80	101.35	2	8.96	1.46
9.00	TRAMO	14	0	0	0	1.80	1.80	150.35	0	8.96	1.46
9.00	TRAMO	15	0	0	0	1.80	0.90	170.45	0	1.00	1.00
9.00	TRAMO	16	0	0	0	1.80	0.90	26.45	2	1.00	1.00

7.3 Pruebas del Sistema

En este apartado se realizan 2 pruebas del simulador sobre escenarios reales de la ciudad de Sevilla. Ahora lo que interesa es estudiar el comportamiento del conjunto de vehículos que circulan por el viario, es decir, las estadísticas globales de todos los vehículos, aunque cada uno de ellos sea tratado de forma independiente, ya que es un simulador microscópico.

Es en este tipo de pruebas donde se pueden apreciar los posibles problemas que puedan existir en la circulación, y por lo tanto, proponer las posibles soluciones que puedan mejorar estas dificultades.

7.3.1 Prueba básica

Esta primera prueba tiene lugar en un escenario compuesto por 5 nodos y 7 tramos de la zona correspondiente a los alrededores del Prado de San Sebastián, concretamente a la zona señalada en color azul en la figura 7.6, donde todos los tramos

son bidireccionales excepto el correspondiente a la calle San Fernando. De estos nodos, sólo el nodo central (nodo 2) está regulado por semáforos.

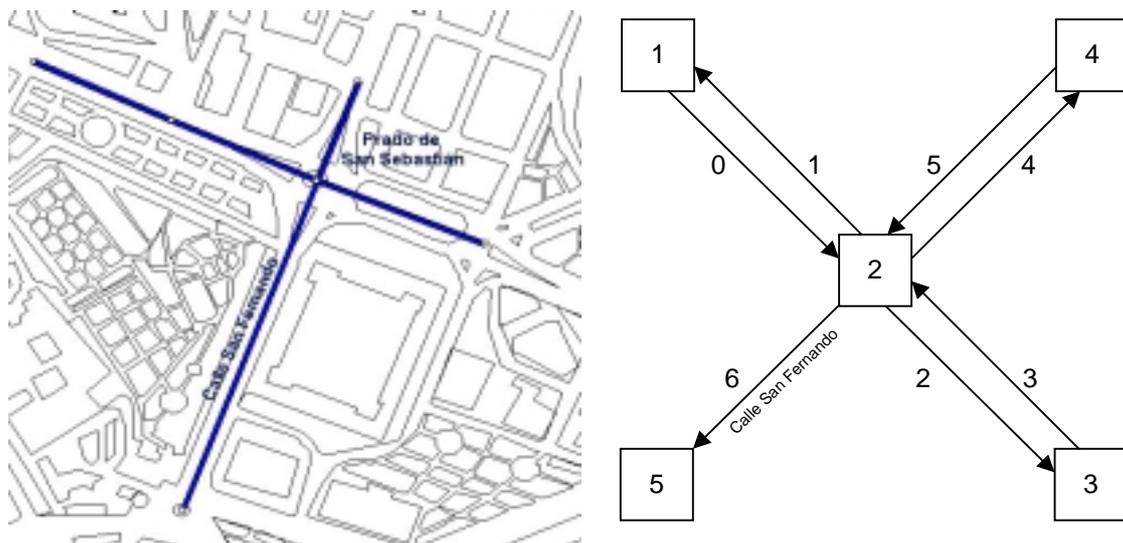


Figura 7.6. Viario de la prueba básica

Los vehículos son generados únicamente en aquellos tramos que tienen como destino el nodo central (nodo 2), es decir en los tramos 0, 3 y 5, y la posición en la que se generan dichos vehículos es el comienzo del tramo. Por esta razón, la función de generación de vehículos sólo toma valores distintos de cero en esos tramos.

En las siguientes tablas se muestra la información correspondiente al código del tramo, longitud, nodo de destino, número de carriles y función de generación de vehículos para cada uno de los tramos del viario, asimismo, para cada uno de los carriles del tramo se muestra el código de la señal que regula la circulación por el carril y el porcentaje de vehículos que deben circular por cada uno de ellos.

Tramo	Longitud	Nodo destino	Carriles	Función generación
0	363.25	2	3	1

Carril	Señal	Porcentaje
0	3	33
1	3	33
2	3	33

Tramo	Longitud	Nodo destino	Carriles	Función generación
1	363.25	1	3	0

Carril	Señal	Porcentaje
0	-1	33
1	-1	33
2	-1	33

Tramo	Longitud	Nodo destino	Carriles	Función generación
2	416.12	3	3	0

Carril	Señal	Porcentaje
0	-1	33
1	-1	33
2	-1	33

Tramo	Longitud	Nodo destino	Carriles	Función generación
3	416.12	2	3	2

Carril	Señal	Porcentaje
0	3	33
1	3	33
2	3	33

Tramo	Longitud	Nodo destino	Carriles	Función generación
4	264.42	4	3	0

Carril	Señal	Porcentaje
0	-1	33
1	-1	33
2	-1	33

Tramo	Longitud	Nodo destino	Carriles	Función generación
5	264.42	2	3	3

Carril	Señal	Porcentaje
0	2	33
1	2	33
2	2	33

Tramo	Longitud	Nodo destino	Carriles	Función generación
6	875.32	5	2	0

Carril	Señal	Porcentaje
0	-1	50
1	-1	50

Tabla 7.2. Tramos del escenario básico

El resto de parámetros de los tramos son idénticos para todos ellos, tales como velocidad máxima en el tramo de 15 metros por segundo o distancia para empezar a considerar lo que ocurre en el nodo siguiente de 10 metros.

Como se aprecia en las tablas anteriores, las funciones estadísticas empleadas en los tramos donde hay generación de vehículos es una normal para el tramo 0, una Poisson para el tramo 3 y una exponencial para el tramo 5.

En todos estos tramos se emplea la misma forma trapezoidal para la generación de vehículos. Este trapecio tiene un área de 100 unidades, por lo que éste es el número aproximado de vehículos que se generan en cada tramo. La forma del trapecio se observa en la figura 7.7.

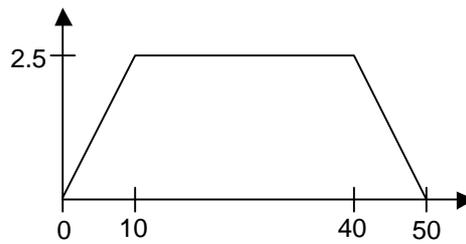


Figura 7.7. Forma trapezoidal de generación en los tramos 0, 3 y 5

Los giros que se pueden realizar en el interior del nodo 2, así como los distintos movimientos en los que se descomponen, se muestran en la figura 7.8. En dicha figura también puede apreciarse la distribución de los tramos.

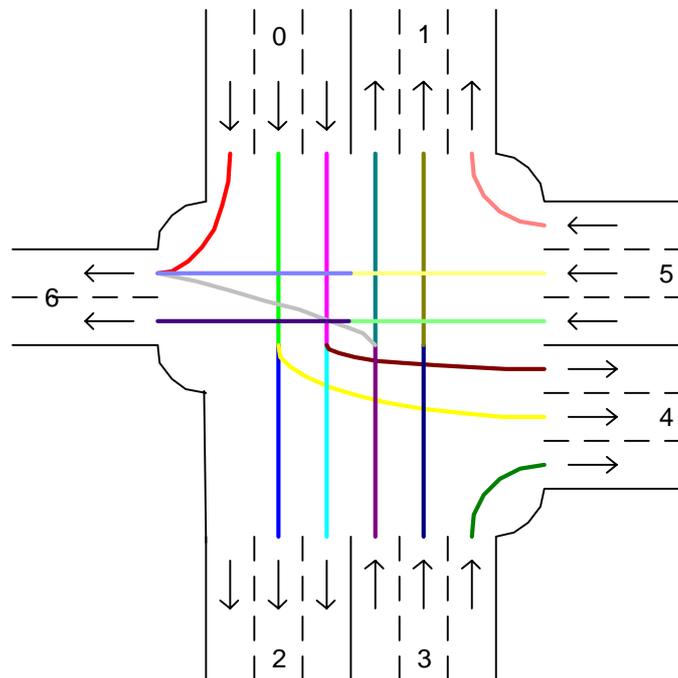


Figura 7.8. Giros y movimientos del nodo 2

A continuación, se muestran unas tablas en las que se caracterizan los movimientos de la figura anterior, mostrando cuál es el origen y el destino de los movimientos, así como su longitud. Se presenta una tabla para cada uno de los diferentes tipos de movimientos existentes en el escenario.

- Movimientos carril-carril:

Movimiento	Tramo origen	Carril origen	Tramo destino	Carril destino	Longitud
0	0	0	6	0	9.33
7	3	0	4	0	5.73
13	5	0	1	0	5.51

Tabla 7.3. Movimientos carril-carril del escenario básico

- Movimientos carril-movimiento:

Movimiento	Tramo origen	Carril origen	Movimiento destino	Longitud
1	0	1	2	10.07
3	0	2	4	9.53
8	3	1	9	9.86
10	3	2	11	9.69
14	5	1	15	8.44
16	5	2	17	9.02

Tabla 7.4. Movimientos carril-movimiento del escenario básico

- Movimientos movimiento-carril:

Movimiento	Movimiento origen	Tramo destino	Carril destino	Longitud
2	1	2	1	10.71
4	3	2	2	11.02
5	3	4	4	13.08
6	1	4	1	16.55
9	8	1	1	10.18
11	10	1	2	10.70
12	10	6	0	14.08
15	14	6	0	11.99
17	16	6	1	11.76

Tabla 7.5. Movimientos movimiento-carril del escenario básico

Por otro lado, la caja semafórica que regula la circulación en el tramo 2, se muestra en la figura 7.9, en la que se aprecian las 8 señales que conforman dicha caja y las 3 fases con sus correspondientes tiempos intermedios en las que se encuentran divididas cada una de ellas. El significado de los distintos colores que pueden tomar las señales es el que ya se indicó en la figura 5.27.

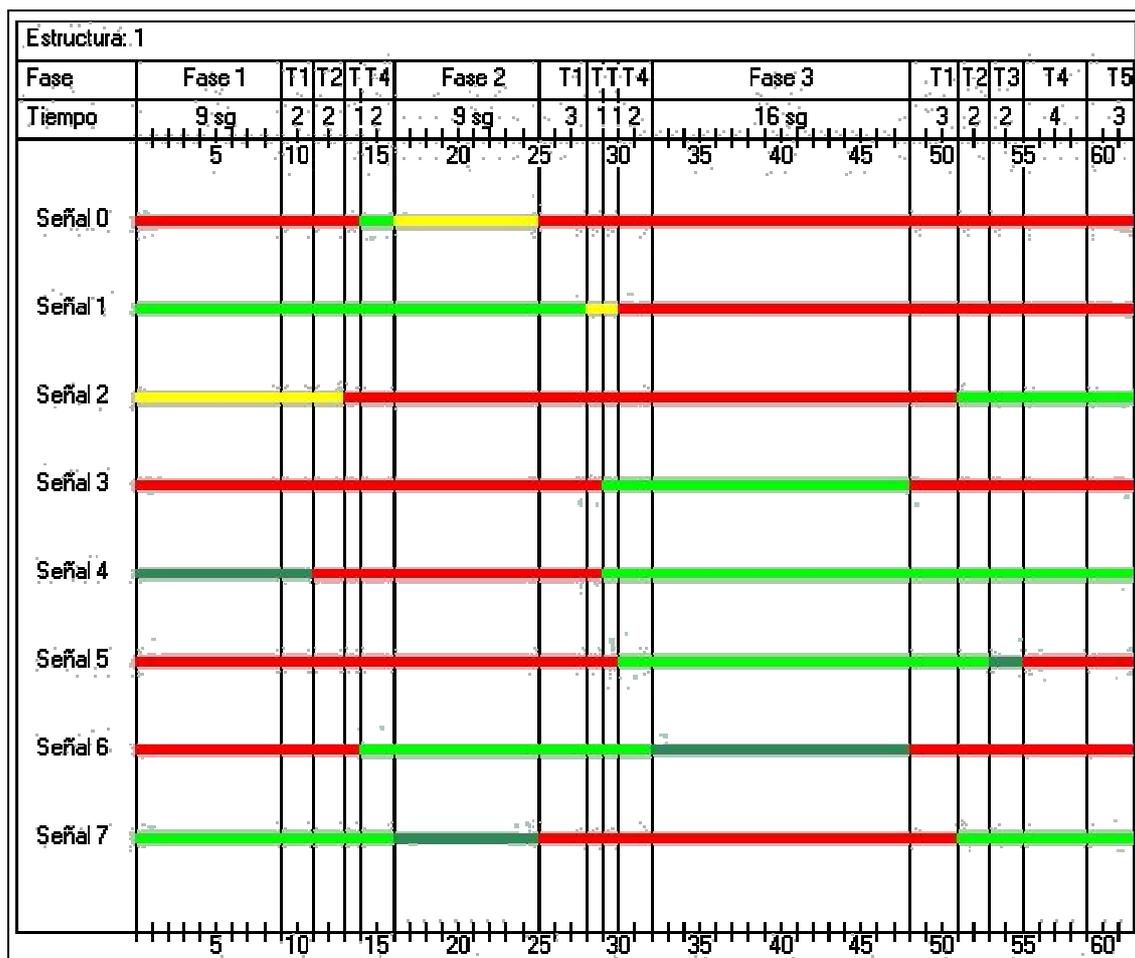


Figura 7.9. Caja semafórica del nodo 2.

Para acabar de especificar el escenario de simulación, falta por decir que los vehículos pueden llevar a cabo los aparcamientos en las posiciones que van de la 200 a la 300 de los tramos 2 y 5, existiendo 10 metros entre un aparcamiento y el siguiente.

Seguidamente se analizan los resultados obtenidos de realizar una simulación de 300 segundos de duración con un intervalo entre dos instantes de simulación consecutivos de 0.5 segundos.

Generación de vehículos

La primera estadística que es objeto de estudio es el número de vehículos generados en cada uno de los tramos que tiene una función de generación distinta de cero. En el tramo 0 este número es igual a 100, lo que coincide exactamente con el área del trapecoide de la función de generación, mientras que en el tramo 3 se generan 97 vehículos y en el tramo 5 se alcanzan los 110 vehículos generados, lo que muestra una pequeña desviación respecto al número teórico, debido a las funciones de distribución estadísticas que modifican la función trapecoidal.

En la figura 7.10 se muestra el número de vehículos generados en estos tramos en cada uno de los instantes de simulación. En dichas gráficas se aprecia que la evolución del número de vehículos generados a lo largo del tiempo se aproxima a la forma trapecoidal mostrada en la figura 7.7 en cuanto a que el mayor número de

vehículos (líneas negras más juntas en la figura) son generados durante el intervalo de tiempo en el que la función trapezoidal es distinta de 0.

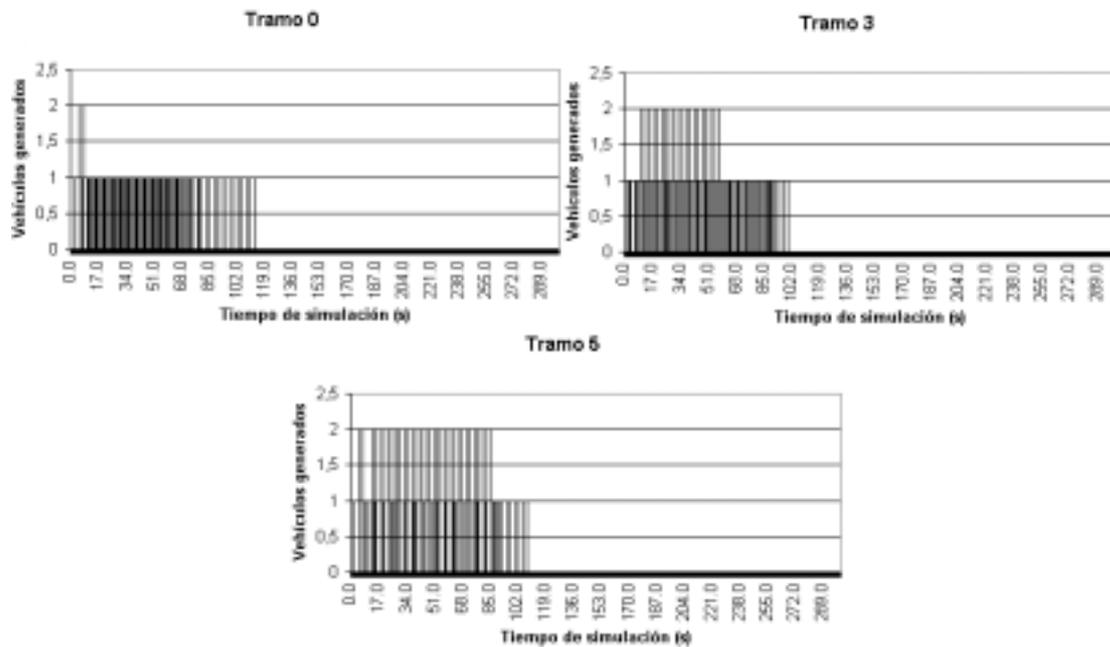


Figura 7.10. Vehículos generados en cada instante de tiempo

Las desviaciones de estos resultados respecto al trapecio teórico son debidas a la aplicación de las funciones de distribución estadísticas y a la imposibilidad de generar ciertos vehículos en el momento deseado, a causa de incompatibilidad con otros vehículos ya existentes.

Una de estas diferencias con el resultado teórico es la generación de vehículos más allá del instante 50 de simulación, esto es debido a que hay vehículos que debían haberse generado con anterioridad, pero por incompatibilidad con otros vehículos existentes en el tramo no fue posible.

En la figura 7.11 se representa el número de vehículos que se han generado en cada uno de los carriles de los 3 tramos en los que existe dicha generación.

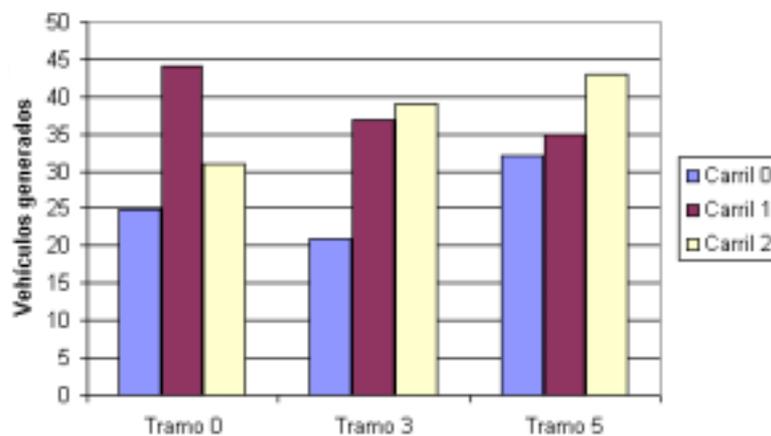


Figura 7.11. Generación de vehículos por carriles

Cambios de carril

En primer lugar se van a analizar algunos de los cambios de carril sucedidos en algunos tramos y justificar por qué se producen. Estos cambios se clasifican en función del tipo al que pertenezcan, y si son por cortesía o no.

En la siguiente tabla se muestran algunos de los cambios de carril obligatorios que se producen en el tramo 5.

Tipo de cambio	Instante	Vehículo	Tramo	Posición	Carril origen	Carril destino
Obligatorio	29.0	1	5	197.88	2	1
Obligatorio	35.0	5	5	235.15	0	1

Tabla 7.6. Ejemplos de cambios de carril obligatorios en el escenario básico

Entre ellos se aprecia el que realiza el vehículo 5 en el instante 35, que pasa del carril 0 al 1, ya que el carril 0 le obliga a girar hacia el tramo 1, mientras que en el carril 1 puede continuar hacia el carril 0 del tramo 6, como es su intención. En la siguiente tabla se muestra la evolución de este vehículo, que pasa del carril 0 al 1 y posteriormente al tramo 6.

Instante	Vehículo	Tramo	Posición	Carril
35.0	5	5	235.15	0
35.5	5	5	240.37	1
59.0	5	6	7.08	0

Tabla 7.7. Evolución del recorrido del vehículo 5

A continuación se muestran algunos cambios de carril por mejora que tienen lugar en el tramo 0.

Tipo de cambio	Instante	Vehículo	Tramo	Posición	Carril origen	Carril destino
Mejora	118.0	271	0	290.02	2	1
Mejora	120.0	287	0	282.26	1	0
Mejora	229.0	316	0	330.94	1	2

Tabla 7.8. Ejemplos de cambios de carril por mejora en el escenario básico

Para ilustrar la mejora en las condiciones de circulación que sufre el vehículo 271 al pasar del carril 2 al 1 y el vehículo 316 al cambiar del 1 al 2, en la figura 7.12 se muestran dos gráficas con la evolución de la velocidad de los vehículos durante el recorrido por sus correspondientes carriles.

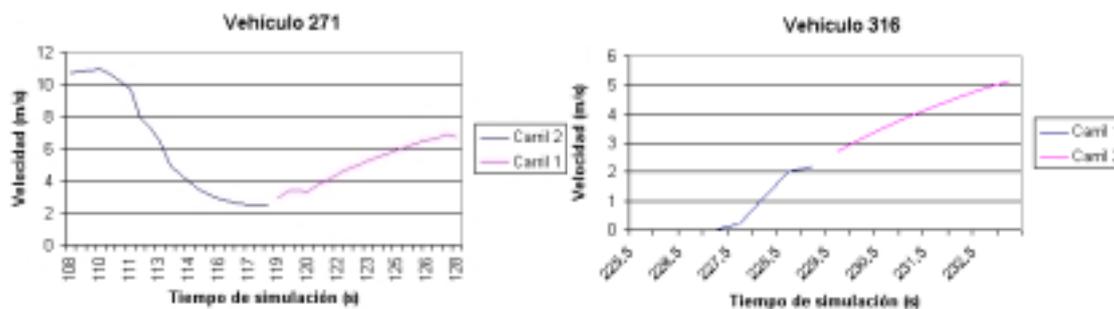


Figura 7.12. Evolución de velocidad en cambio de carril por mejora

En cuanto a los cambios de carril por cortesía, en la siguiente tabla se muestran los datos correspondientes al único cambio de este tipo que se ha producido en el escenario básico de simulación:

Tipo de cambio	Instante	Vehículo	Tramo	Posición	Carril origen	Carril destino	Vehículo trasero	Gap trasero	Intentos
Cortesía	113.0	257	3	321.54	2	1	264	10.41	3

Tabla 7.9. Cambio de carril por cortesía en el escenario básico

En la figura 7.13 se muestran dos gráficas representativas del cambio de carril por cortesía que realiza el vehículo 257 en el instante 113 y que le permite llevarlo a cabo el vehículo 264.

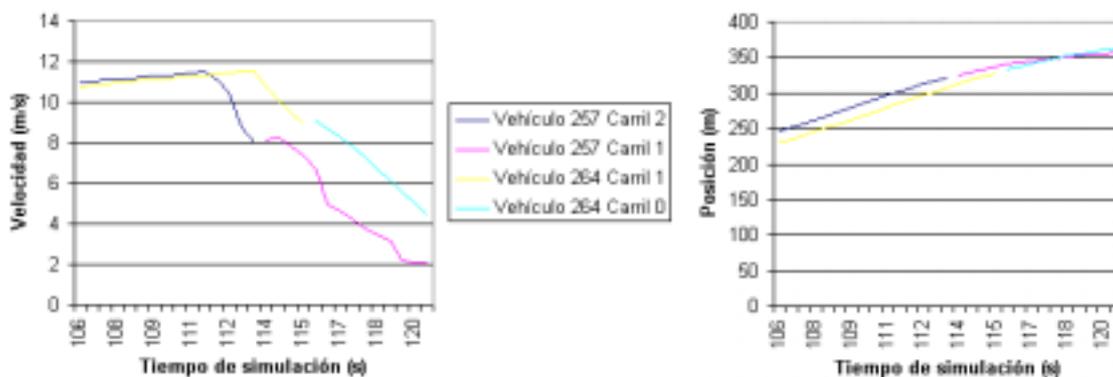


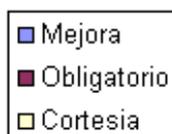
Figura 7.13. Ejemplo de cambio de carril por cortesía

En la primera de ellas se muestra cómo el vehículo 257 aumenta ligeramente su velocidad al desplazarse al carril 1, mientras que el vehículo 264 tiene que frenar para permitir esa maniobra.

En la segunda gráfica de la figura 7.13 se representa la posición de los vehículos en cada instante de tiempo, observándose que la maniobra se realiza a pesar del poco espacio existente entre los vehículos, y que el vehículo 264 frena para aumentar esta distancia. También se aprecia que una vez finalizado el cambio de carril por cortesía del vehículo 257, el vehículo 264 cambia al carril 0 para poder mejorar sus condiciones de circulación.

En los gráficos circulares de la figura 7.14 se muestran los porcentajes de cambios de carril de cada tipo que se producen en los diferentes tramos.

La leyenda correspondiente a dichos gráficos circulares es la siguiente:



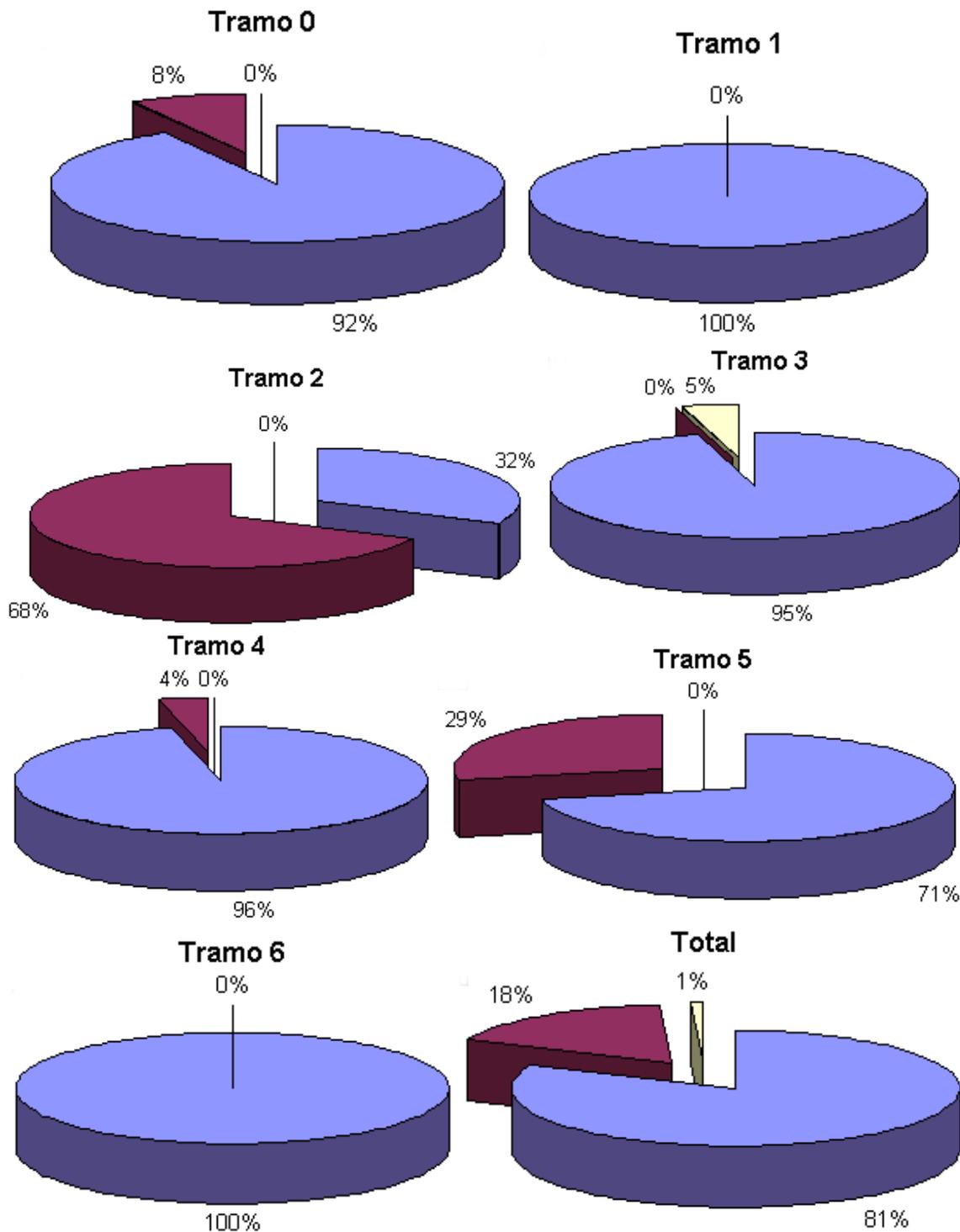


Figura 7.14. Porcentajes de tipos de cambios de carril

Los cambios de carril obligatorios tienen lugar más cerca del final del tramo que los cambios por mejora. Esto se aprecia en la figura 7.15 en la que se representan los instantes y posiciones en los que se producen los diferentes tipos de cambios de carril ocurridos en el tramo 5, que tiene una longitud de 264.42 metros.

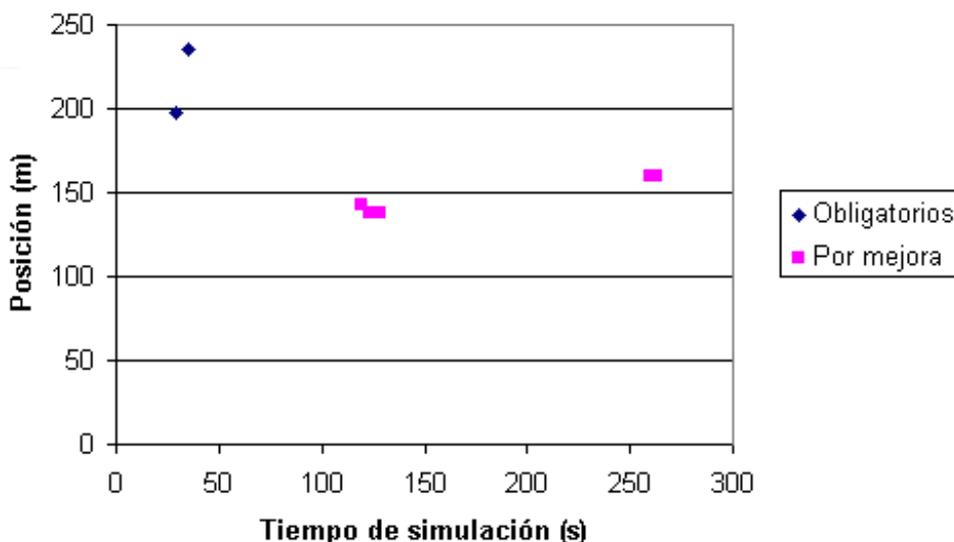


Figura 7.15. Cambios de carril ocurridos en el tramo 5

Una excepción a esta regla de que los cambios de carril obligatorios se producen al final del tramo, la constituyen los cambios de carril destinados a situar un vehículo en el carril adecuado para realizar un aparcamiento. Esto es lo que ocurre en el tramo 2, donde la mayor parte de los cambios de carril obligatorios son por aparcamientos, por lo que se producen en posiciones cercanas al lugar donde éstos se encuentran.

Otra de las estadísticas relacionadas con los cambios de carril es el número de cambios de carril que lleva a cabo cada vehículo durante su recorrido por el viario. En la figura 7.16 se muestra en un gráfico el número de cambios de carril que realizan todos los vehículos (sin incluir los vehículos ficticios). La línea roja indica el número medio de cambios de carril por vehículo.

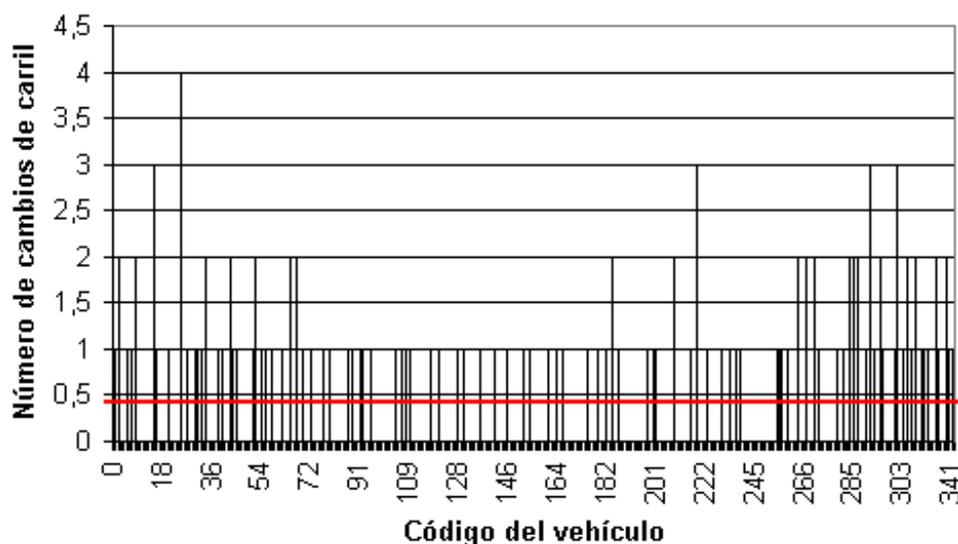


Figura 7.16. Número de cambios de carril por vehículo

A modo de ejemplo se muestran en la siguiente tabla los 2 cambios de carril que realiza el vehículo 284. Este vehículo comienza circulando por el carril 1 del tramo 3 y en el instante 122.5 pasa al carril 0 para aumentar su velocidad. Al llegar al nodo 2, gira

a la derecha y pasa al carril 0 del tramo 4 (ver figura 7.8). En el instante 227 decide cambiar al carril 1 para aumentar su velocidad.

Tipo de cambio	Instante	Tramo	Vehículo	Posición	Carril origen	Carril destino
Mejora	122.5	3	284	331.24	1	0
Mejora	227.0	4	284	31.09	0	1

Tabla 7.10. Cambios de carril del vehículo 284

Estadísticas genéricas

La evolución del número de vehículos que circulan por cada tramo a lo largo del tiempo de simulación se muestra en la figura 7.17.

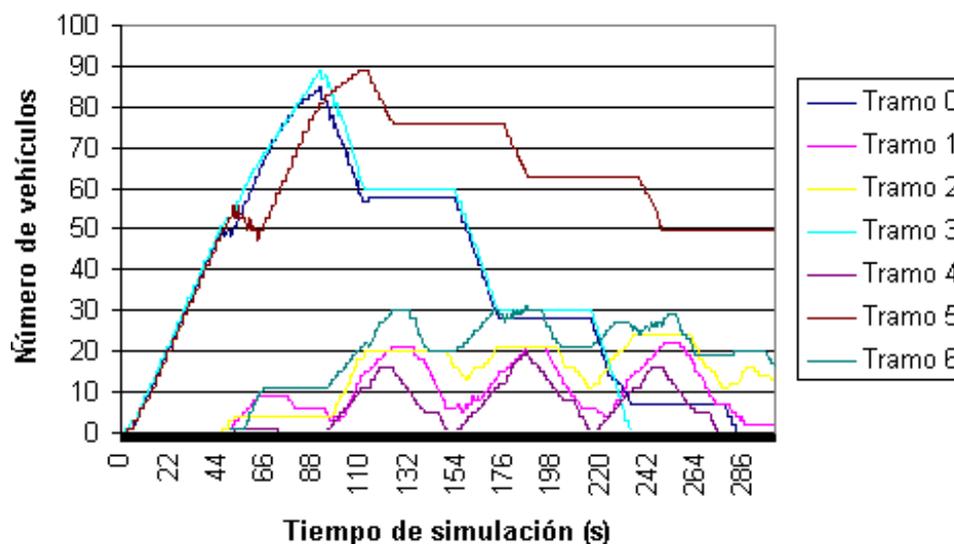


Figura 7.17. Número de vehículos por tramo

En la figura 7.17 se aprecia que en los tramos con función de generación (0, 3 y 5), la subida en el número de vehículos se corresponde con el ascenso de la función trapezoidal, mientras que dicho número va descendiendo de forma escalonada, conforme se pone en verde el semáforo que regula el tramo y permite a los vehículos salir del mismo.

En el resto de tramos, los vehículos existentes provienen de alguno de los tramos anteriores, por lo que el número de vehículos comienza a aumentar cuando salen los vehículos de los tramos 0, 3 y 5.

En la figura 7.18 se muestra en un diagrama de columnas el número medio de vehículos que circulan por cada tramo.

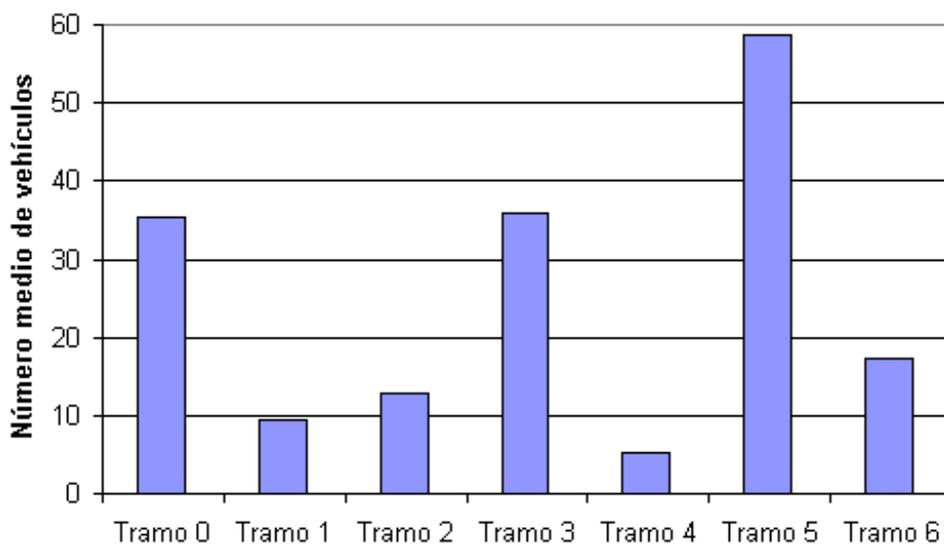


Figura 7.18. Número medio de vehículos por tramo

También se puede obtener el número de vehículos que circulan por cada uno de los carriles de los tramos. En la figura 7.19 se muestra el número de vehículos que circulan por cada carril de los tramos 3 y 6 en cada instante de tiempo, así como el número total de vehículos que circulan por dichos tramos.

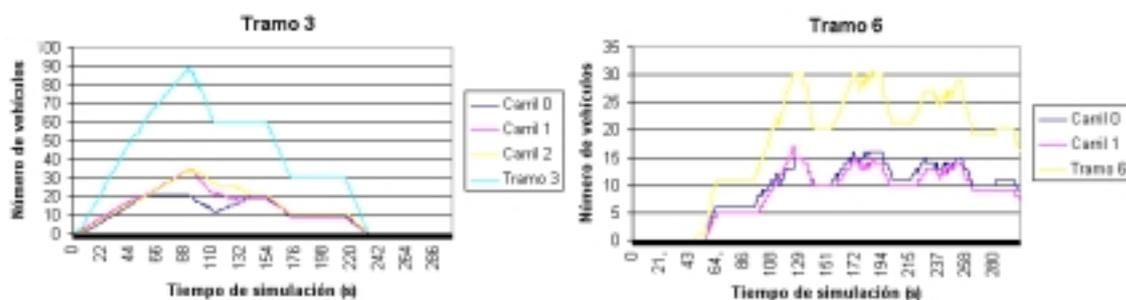


Figura 7.19. Distribución de vehículos por carriles en el tramo 0

En esta figura se aprecia que aproximadamente el 33% del total de los vehículos que circulan por el tramo 3, lo hacen por cada uno de sus carriles, mientras que en el tramo 6, los vehículos se reparten aproximadamente al 50% entre sus 2 carriles. Es decir, existe aproximadamente el mismo número de vehículos en cada carril, lo cual era una de las condiciones que impusimos al escenario.

A continuación se van a analizar las estadísticas relacionadas con los semáforos. Para ello, se comienza por mostrar en la figura 7.20 el número de vehículos que se encuentran detenidos en cada instante en cada uno de los semáforos.

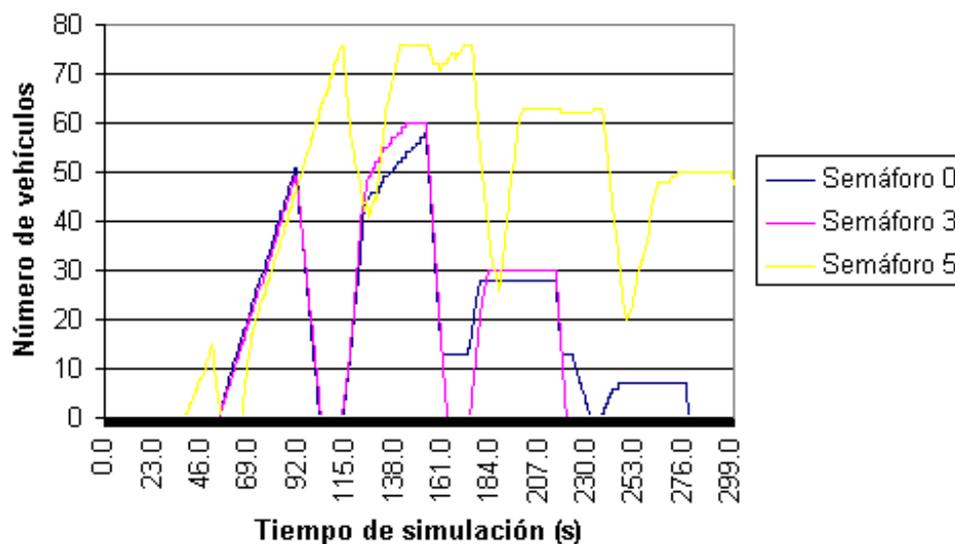


Figura 7.20. Vehículos detenidos en los semáforos

En esta figura se aprecia que el número de vehículos detenidos en los semáforos reguladores de los tramos 0 y 3 presentan una evolución muy similar, esto se debe a que el número de vehículos que circulan por ambos tramos es aproximadamente el mismo y los momentos de cambios de la señal semafórica coinciden, ya que ambos tramos están regulados por la misma señal.

Sin embargo, la señal semafórica que regula la circulación en el tramo 5 está en contrafase con las anteriores, ya que, como se aprecia en la figura 7.20, cuando el número de vehículos detenidos en los semáforos de los tramos 0 y 3 deja de crecer al ritmo que llevaba cuando estaba en rojo, en el semáforo del tramo 5 comienza a aumentar el número de vehículos detenidos.

Respecto a esta misma gráfica, también cabe destacar que en los primeros instantes de simulación no existe ningún vehículo detenido en los semáforos, ya que como se ha comentado, los vehículos son generados al principio del tramo, por lo que es necesario que transcurra un cierto tiempo antes de que lleguen al final del tramo, donde están situados los semáforos.

Dado que la generación de vehículos en estos tramos se detiene al alcanzarse un determinado instante de simulación, en los instantes finales no hay vehículos detenidos en los semáforos de los tramos 0 y 3, ya que todos los vehículos han salido de esos tramos. Sin embargo, en el semáforo del tramo 5 aún existe una gran cantidad de vehículos detenidos, esto se debe a que el tiempo de permanencia de la señal en verde es muy pequeño, no dando tiempo a que salgan todos los vehículos que estaban detenidos. Esto lleva a pensar en la conveniencia de aumentar el tiempo en verde de esta señal.

En los semáforos de los tramos 0 y 3 también hay vehículos que deben esperar más de un ciclo de la señal del semáforo, lo que indica que la circulación en estos tramos también es bastante densa.

En la figura 7.21 se muestra el número medio de vehículos detenidos en cada instante en cada semáforo, observándose que dicho número es mayor en el semáforo 5.

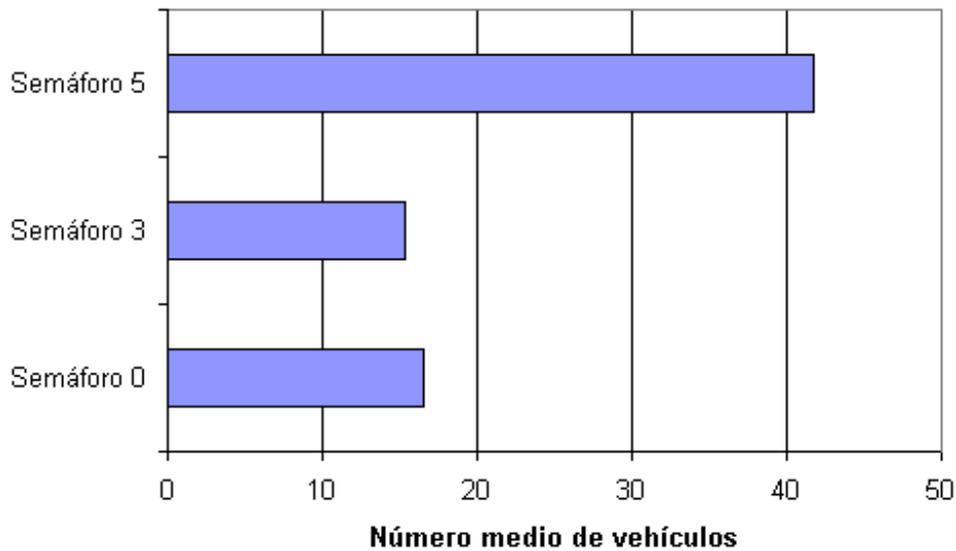


Figura 7.21. Número medio de vehículos detenidos en cada semáforo

Otro factor interesante relacionado con los semáforos es el tiempo medio que permanece detenido un vehículo en cada uno de los semáforos (contabilizando únicamente aquellos que deben detenerse en los semáforos), este *tiempo medio de espera en semáforo* se muestra en la figura 7.22.

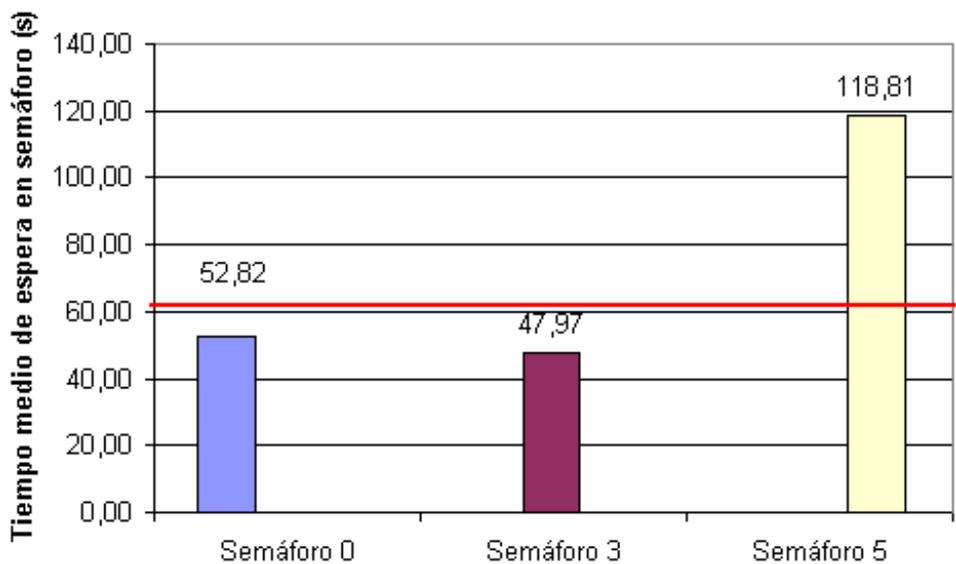


Figura 7.22. Tiempo medio de un vehículo detenido en un semáforo

En esta figura se aprecia que para el semáforo 5, el tiempo medio de espera de los vehículos es casi el doble que el tiempo de ciclo del semáforo, que tiene un valor de 62 y está indicado por una línea roja en la figura. Esto implica que la mayor parte de los vehículos que se detienen en el semáforo deben esperar más de lo que dura un ciclo de la señal del semáforo, como ya se indicó con anterioridad.

Otro dato de interés es el número de vehículos que se tienen que detener en el semáforo de cada uno de los tramos respecto del total de los vehículos que son generados en él. Esta estadística se muestra en la figura 7.23, en la que se aprecia que la

mayor parte de los vehículos tienen que detenerse en el semáforo del tramo en el que se generan.

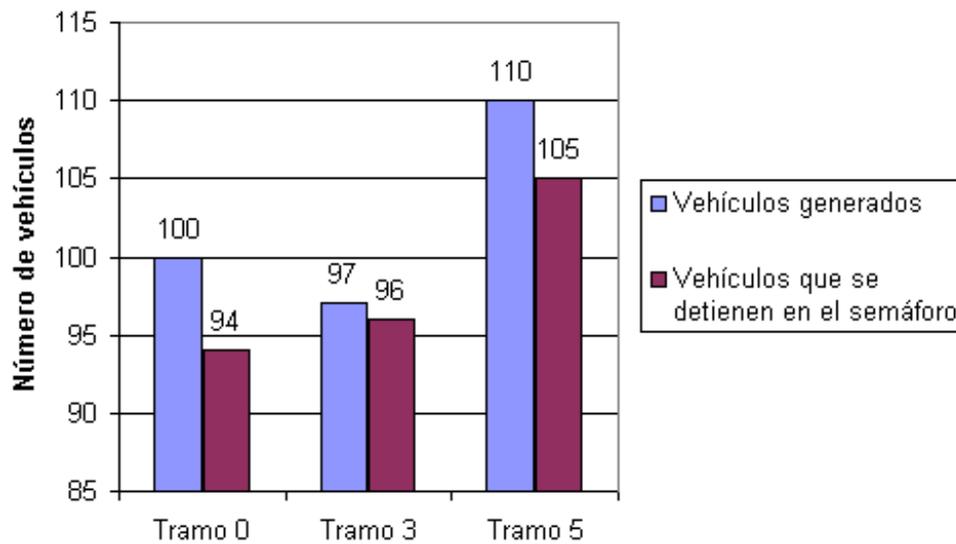


Figura 7.23. Número total de vehículos que se detienen en los semáforos

La siguiente estadística contemplada es el tiempo que permanece un vehículo en cualquier tramo desde que comienza su recorrido por él hasta que sale del mismo o realiza un aparcamiento. Calculando este valor para todos los vehículos en cada tramo por el que circula y obteniendo el valor medio para todos los tramos, se obtienen los resultados mostrados en la figura 7.24.

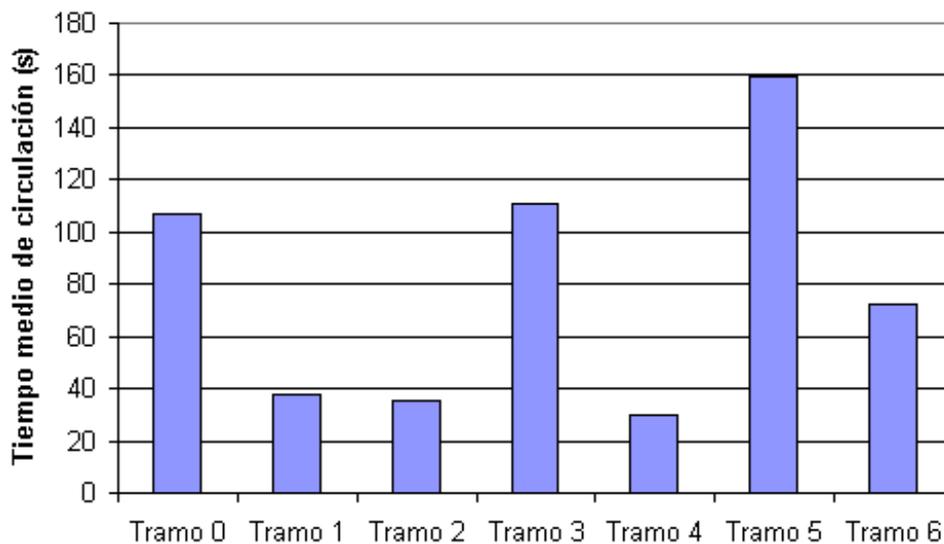


Figura 7.24. Tiempo medio de un vehículo en cada tramo

Como se observa en la figura, el tramo 5 es el que más tardan en recorrer los vehículos, debido al problema comentado de la escasa duración del semáforo en verde. A continuación se encuentran los tramos 0 y 3, que también están regulados por semáforos, y el tramo 6, que es el de mayor longitud. Finalmente, los tramos que son recorridos más rápidamente son el 1, 2 y 4, que no tienen semáforos y tienen una longitud media.

En la figura 7.25 se muestran las velocidades medias desarrolladas por los vehículos en los diferentes tramos. Se observa que son inversamente proporcionales a los tiempos medios de la figura 7.24, con excepción del tramo 6, que a pesar de tener una velocidad media alta, también tiene un tiempo medio de recorrido elevado, debido a la gran longitud que tiene.

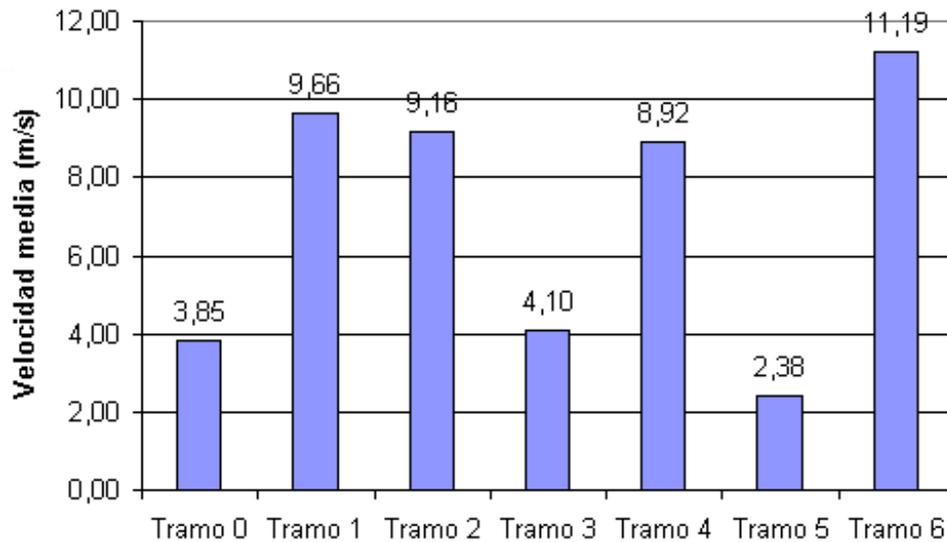


Figura 7.25. Velocidades medias en los tramos

Es lógico que en los tramos terminales del viario se alcancen las velocidades medias más altas, ya que los vehículos pueden salir del tramo sin tener que frenar a causa de un semáforo o para incorporarse a determinado movimiento.

La última estadística que falta por comentar es el tiempo en el sistema de los vehículos. Calculando este tiempo para todos los vehículos que circulan por el viario y obteniendo su media, el resultado obtenido es de 172.15 segundos. En este tiempo se contemplan tanto el tiempo que los vehículos se encuentran circulando por los tramos, como por los movimientos, además del tiempo que están detenidos en los semáforos.

7.3.2 Prueba extendida

Para la realización de la prueba extendida se emplea un escenario formado por el escenario básico al que se le añaden nuevos nodos y tramos, hasta tener los 12 nodos y 20 tramos que se muestran en las figuras 7.26 y 7.27. De los 12 nodos, hay 4 que son regulados por semáforos, concretamente los nodos 2, 3, 5 y 10.

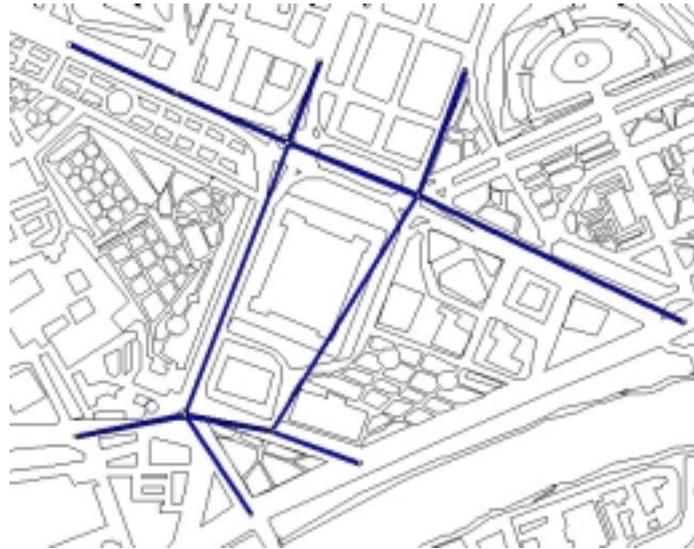


Figura 7.26. Viario del escenario extendido

La representación del viario extendido mediante un grafo dirigido se muestra en la figura 7.27.

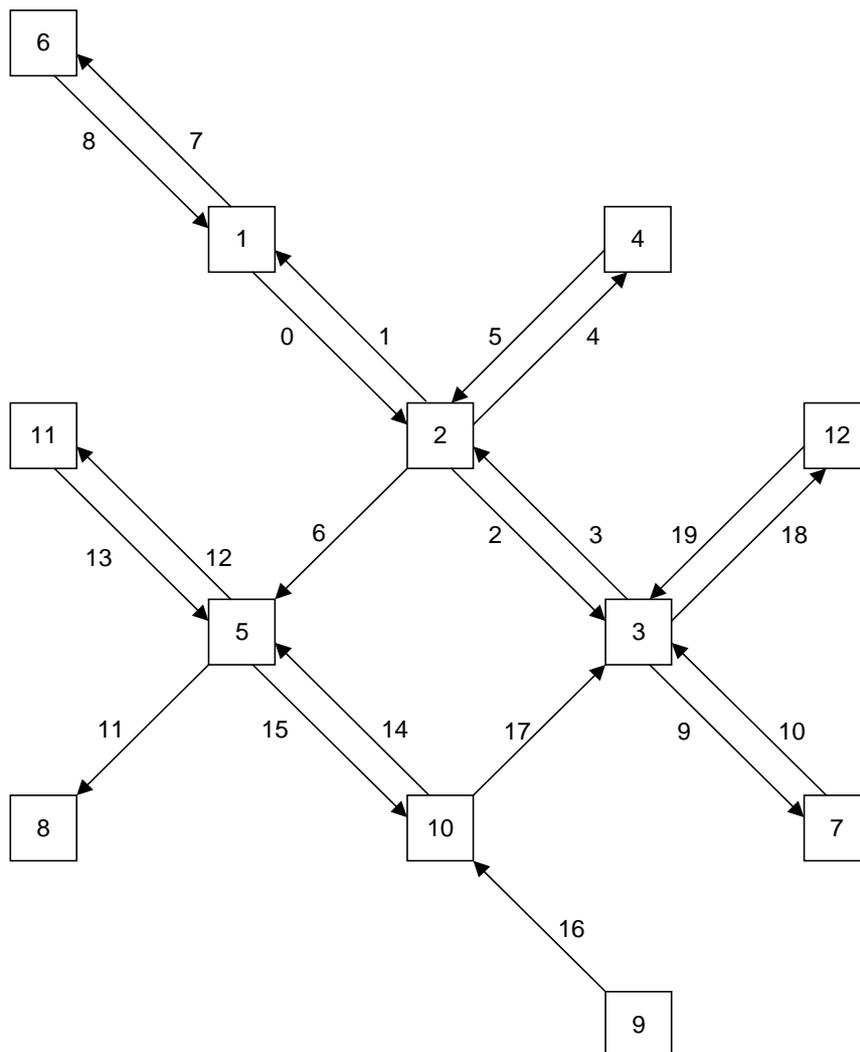


Figura 7.27. Grafo representativo del viario extendido

La información correspondiente a los diferentes tramos se muestra en las siguientes tablas:

Tramo	Longitud	Nodo destino	Carriles	Función generación
0	363.25	2	3	0

Carril	Señal	Porcentaje
0	3	33
1	3	33
2	3	33

Tramo	Longitud	Nodo destino	Carriles	Función generación
1	363.25	1	3	0

Carril	Señal	Porcentaje
0	-1	33
1	-1	33
2	-1	33

Tramo	Longitud	Nodo destino	Carriles	Función generación
2	416.12	3	3	0

Carril	Señal	Porcentaje
0	-1	33
1	-1	33
2	-1	33

Tramo	Longitud	Nodo destino	Carriles	Función generación
3	416.12	2	3	0

Carril	Señal	Porcentaje
0	3	33
1	3	33
2	3	33

Tramo	Longitud	Nodo destino	Carriles	Función generación
4	264.42	4	3	0

Carril	Señal	Porcentaje
0	-1	33
1	-1	33
2	-1	33

Tramo	Longitud	Nodo destino	Carriles	Función generación
5	264.42	2	3	1

Carril	Señal	Porcentaje
0	2	33
1	2	33
2	2	33

Tramo	Longitud	Nodo destino	Carriles	Función generación
6	875.32	5	2	0

Carril	Señal	Porcentaje
0	-1	50
1	-1	50

Tramo	Longitud	Nodo destino	Carriles	Función generación
7	347.83	6	3	0

Carril	Señal	Porcentaje
0	-1	33
1	-1	33
2	-1	33

Tramo	Longitud	Nodo destino	Carriles	Función generación
8	347.83	1	3	2

Carril	Señal	Porcentaje
0	-1	33
1	-1	33
2	-1	33

Tramo	Longitud	Nodo destino	Carriles	Función generación
9	879.13	7	2	0

Carril	Señal	Porcentaje
0	-1	50
1	-1	50

Tramo	Longitud	Nodo destino	Carriles	Función generación
10	879.13	3	2	3

Carril	Señal	Porcentaje
0	-1	50
1	-1	50

Tramo	Longitud	Nodo destino	Carriles	Función generación
11	361.12	8	3	0

Carril	Señal	Porcentaje
0	-1	33
1	-1	33
2	-1	33

Tramo	Longitud	Nodo destino	Carriles	Función generación
12	326.88	11	2	0

Carril	Señal	Porcentaje
0	-1	50
1	-1	50

Tramo	Longitud	Nodo destino	Carriles	Función generación
13	326.88	5	2	4

Carril	Señal	Porcentaje
0	0	50
1	0	50

Tramo	Longitud	Nodo destino	Carriles	Función generación
14	266.06	5	2	0

Carril	Señal	Porcentaje
0	0	50
1	0	50

Tramo	Longitud	Nodo destino	Carriles	Función generación
15	266.06	10	2	0

Carril	Señal	Porcentaje
0	2	50
1	2	50

Tramo	Longitud	Nodo destino	Carriles	Función generación
16	276.93	10	3	1

Carril	Señal	Porcentaje
0	0	33
1	0	33
2	1	33

Tramo	Longitud	Nodo destino	Carriles	Función generación
17	830.60	3	3	0

Carril	Señal	Porcentaje
0	2	33
1	2	33
2	2	33

Tramo	Longitud	Nodo destino	Carriles	Función generación
18	402.61	12	2	0

Carril	Señal	Porcentaje
0	-1	50
1	-1	50

Tramo	Longitud	Nodo destino	Carriles	Función generación
19	402.61	3	2	2

Carril	Señal	Porcentaje
0	2	50
1	2	50

Tabla 7.11. Tramos del escenario extendido

En la siguiente tabla se muestran los tramos en los que se produce generación de vehículos, indicando el tipo de función estadística empleada a tal efecto. También cabe decir que todos los vehículos son generados al principio del tramo correspondiente.

Tramo	Función estadística
5	Normal
8	Poisson
10	Exponencial
13	Triangular
16	Normal
19	Poisson

Tabla 7.12. Relación entre tramos y funciones estadísticas

La función trapezoidal empleada para la generación de vehículos se muestra en la figura 7.28, y tiene un área total de 70 unidades, lo que quiere decir que se generan aproximadamente 70 vehículos al principio de cada uno de estos tramos.

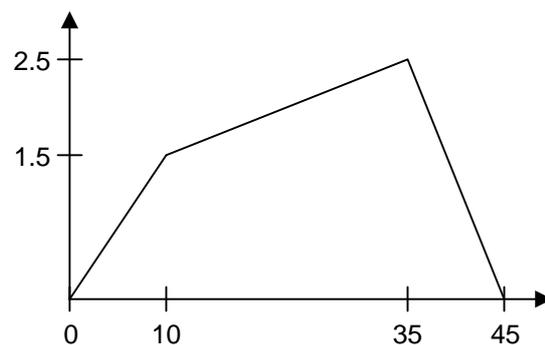


Figura 7.28. Función trapezoidal de generación de vehículos en escenario extendido

Los giros y movimientos definidos en el nodo 2 ya fueron mostrados en la figura 7.8, mientras que su caja semafórica correspondiente se mostró en la figura 7.9. En las siguientes figuras se muestran estos datos correspondientes a los nodos 3, 5 y 10.

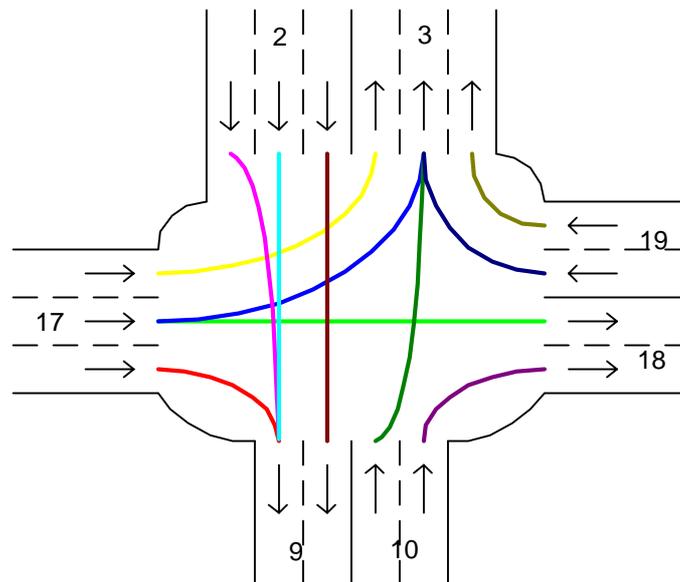


Figura 7.29. Giros y movimientos en el nodo 3

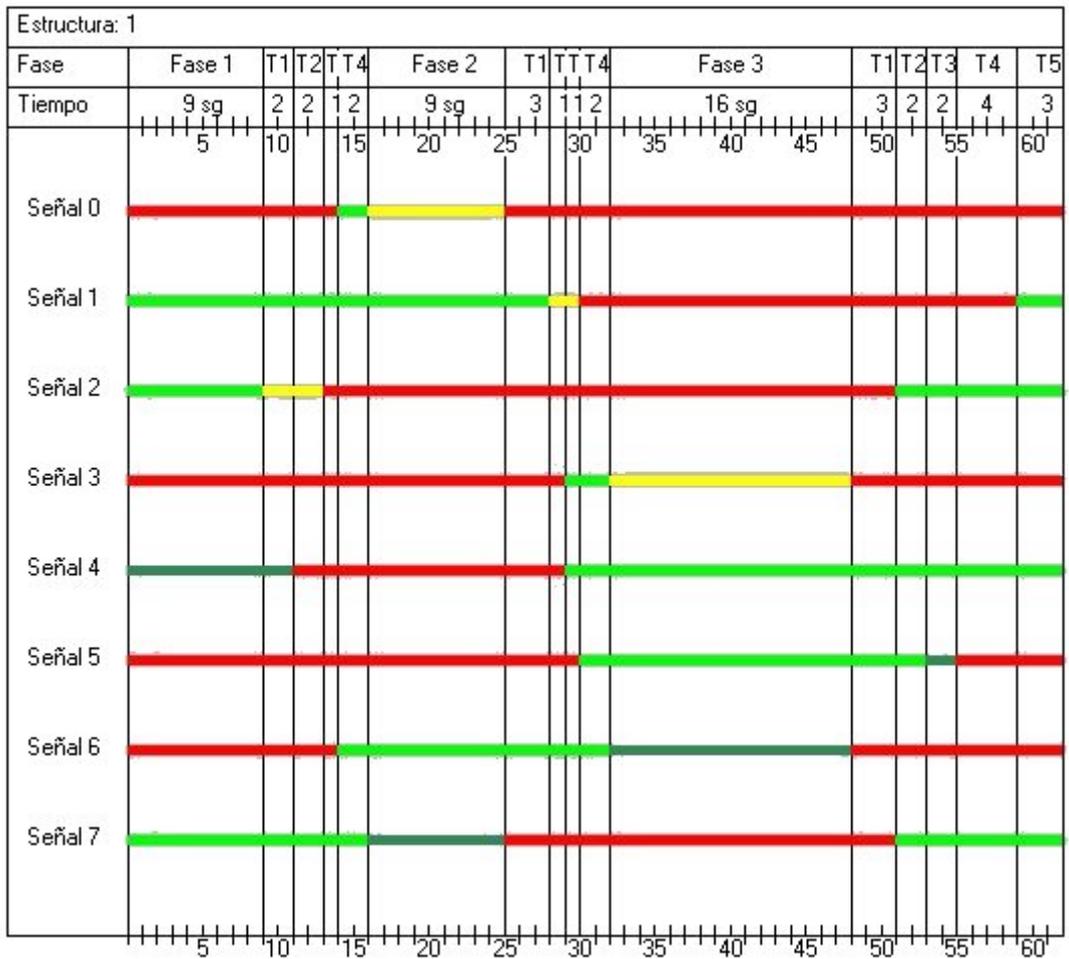


Figura 7.30. Caja semafórica del nodo 3

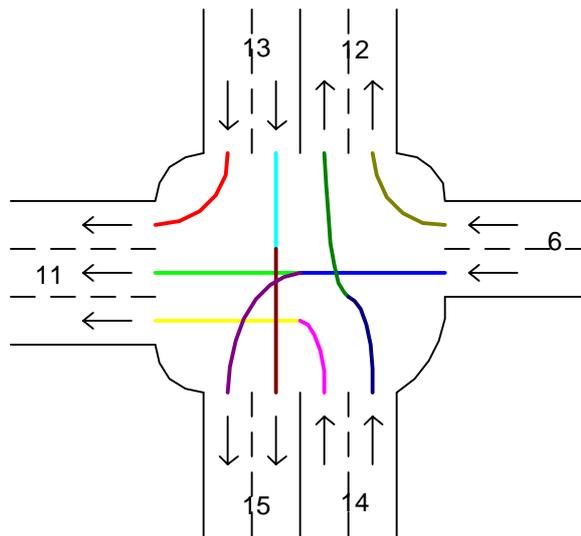


Figura 7.31. Giros y movimientos en el nodo 5



Figura 7.32. Caja semafórica del nodo 5

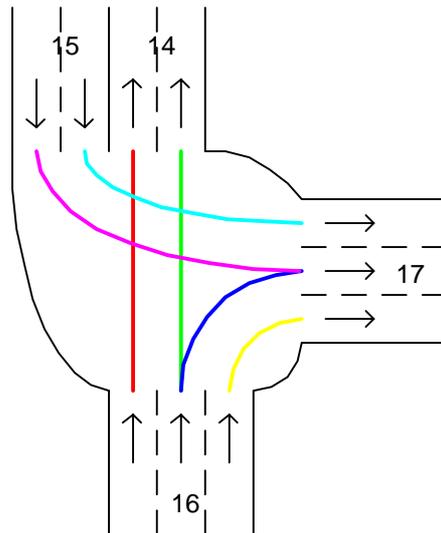


Figura 7.33. Giros y movimientos en el nodo 10

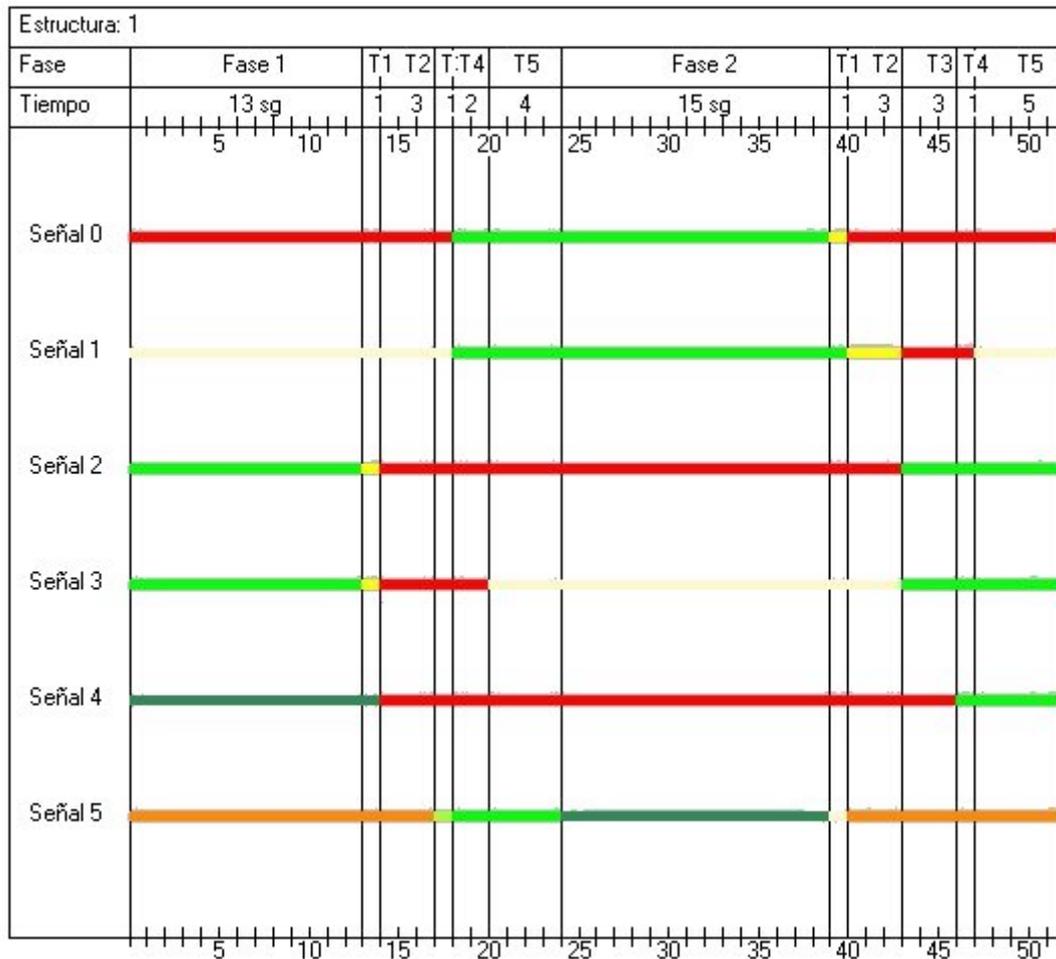


Figura 7.34. Caja semafórica del nodo 10

Respecto a los parámetros correspondientes a los distintos tipos de conductores, cabe decir que toman los mismos valores que fueron indicados en la tabla 7.1, a excepción de la velocidad deseada por cada tipo de conductor, cuyos valores (en metros por segundo) se muestran en la tabla 7.13.

Tipo de conductor	Velocidad deseada
Novato	12
Experimentado	14
Agresivo	15
Defensivo	13

Tabla 7.13. Velocidades deseadas por los distintos tipos de conductores

A continuación se muestran los resultados de las distintas estadísticas obtenidas para una simulación de duración 200 segundos.

Generación de vehículos

El número teórico de vehículos a generar en cada uno de los tramos de entrada al escenario es 70, sin embargo, debido a las distribuciones estadísticas que afectan a la función trapezoidal, el número de vehículos generados en cada uno de estos tramos es ligeramente distinto, como se muestra en la tabla 7.14.

Tramo	Vehículos generados
5	69
8	59
10	56
13	69
16	70
19	76

Tabla 7.14. Número de vehículos generados en cada tramo

El número de vehículos generados en cada instante de tiempo en cada uno de estos tramos se aprecia en la figura 7.35.

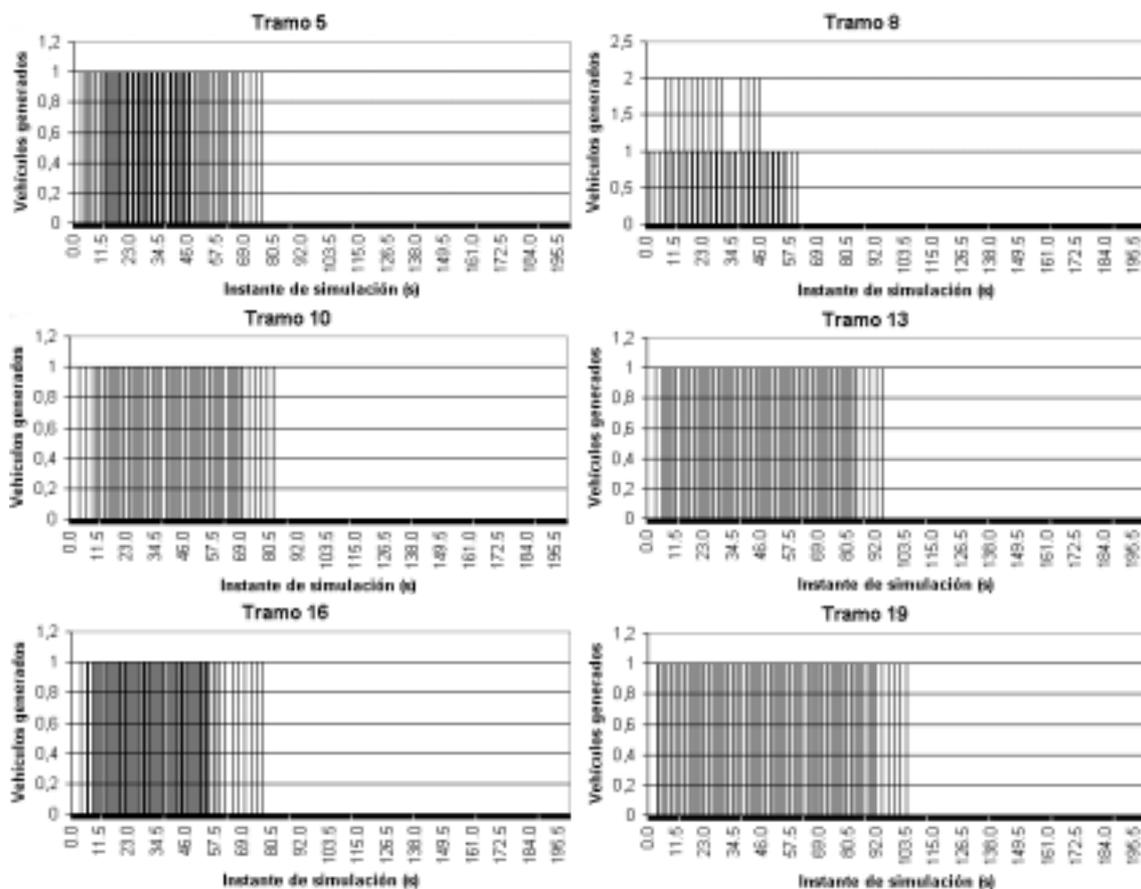


Figura 7.35. Vehículos generados en cada instante de tiempo

En estas figuras no se aprecia con claridad el hecho de que la función de generación siga una forma trapezoidal, ya que debido a la pequeña altura del trapezoide, suele generarse un único vehículo en cada instante. No obstante, sí se aprecia que en los instantes de tiempo correspondientes a la zona central del trapezoide, las líneas que indican la generación de un vehículo se encuentran más juntas, lo que muestra que se sigue dicha función.

Al igual que se comentó en la prueba básica, la generación de vehículos más allá del tiempo en el que el trapezoide acaba, es debido al aplazamiento de aquellas generaciones que no pudieron realizarse en su debido momento.

En la figura 7.36 se muestra el número de vehículos generados en cada uno de los carriles de los tramos en los que existe una función de generación de vehículos.

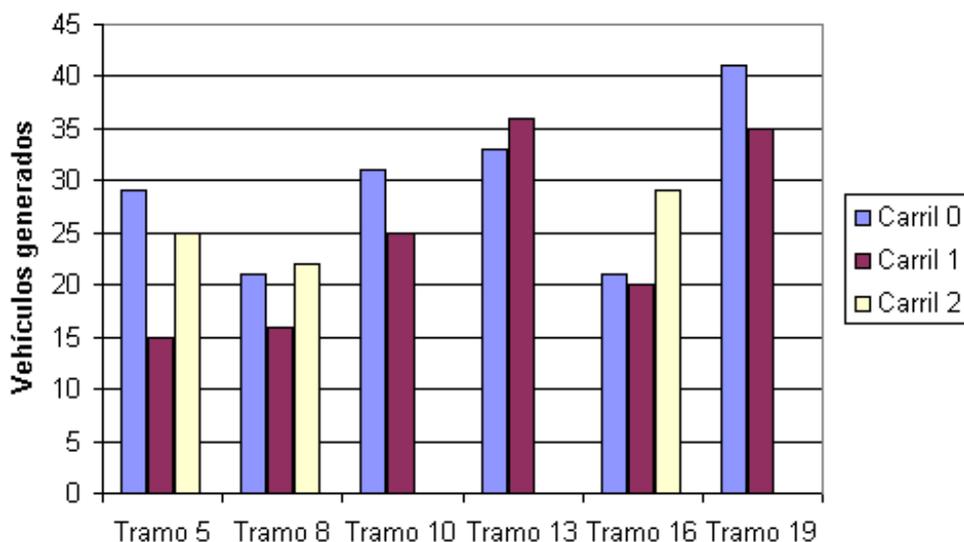


Figura 7.36. Vehículos generados por carril

Cambios de carril

El primer tipo de cambios de carril que se analizan son los obligatorios. En la tabla 7.15 se muestran algunos de los cambios de este tipo que tienen lugar en el tramo 5 del viario.

Tipo de cambio	Instante	Vehículo	Tramo	Posición	Carril origen	Carril destino
Obligatorio	31.0	1	5	213.66	2	1
Obligatorio	33.5	1	5	240.59	1	0
Obligatorio	35.5	14	5	218.98	2	1
Obligatorio	38.5	34	5	208.37	0	1

Tabla 7.15. Ejemplos de cambios de carril obligatorios

En particular, se aprecia que el vehículo 1 realiza dos cambios de carril obligatorios consecutivos, del carril 2 al 1 y del 1 al 0. Esto es debido a que el vehículo 1 debe dirigirse hacia el tramo 1, giro que únicamente es posible realizar desde el carril 0 (ver figura 7.8). En la tabla 7.16 se muestra la posición de dicho vehículo en determinados instantes de la simulación:

Instante	Vehículo	Tramo	Posición	Carril
31.0	1	5	213.66	2
31.5	1	5	218.98	1
33.5	1	5	240.59	1
34.0	1	5	245.74	0
54.5	1	1	4.81	0

Tabla 7.16. Evolución del vehículo 1

En cuanto a los cambios de carril por mejora, en la tabla 7.17 se muestran algunos de los cambios de este tipo que han tenido lugar en el tramo 1.

Tipo de cambio	Instante	Vehículo	Tramo	Posición	Carril origen	Carril destino
Mejora	63.0	79	1	10.43	0	1
Mejora	69.0	108	1	31.06	0	1
Mejora	69.5	138	1	12.24	0	1
Mejora	72.5	168	1	11.46	0	1

Tabla 7.17. Ejemplos de cambios de carril por mejora

Para reflejar la mejora en la velocidad que conlleva la realización de un cambio de este tipo, en la figura 7.37 se muestra la evolución de la velocidad del vehículo 108 antes y después de efectuar el cambio de carril. En ella se aprecia que en el momento en el que la velocidad del vehículo comienza a descender, el vehículo 108 decide cambiar al carril 1, para que su velocidad pueda seguir aumentando.

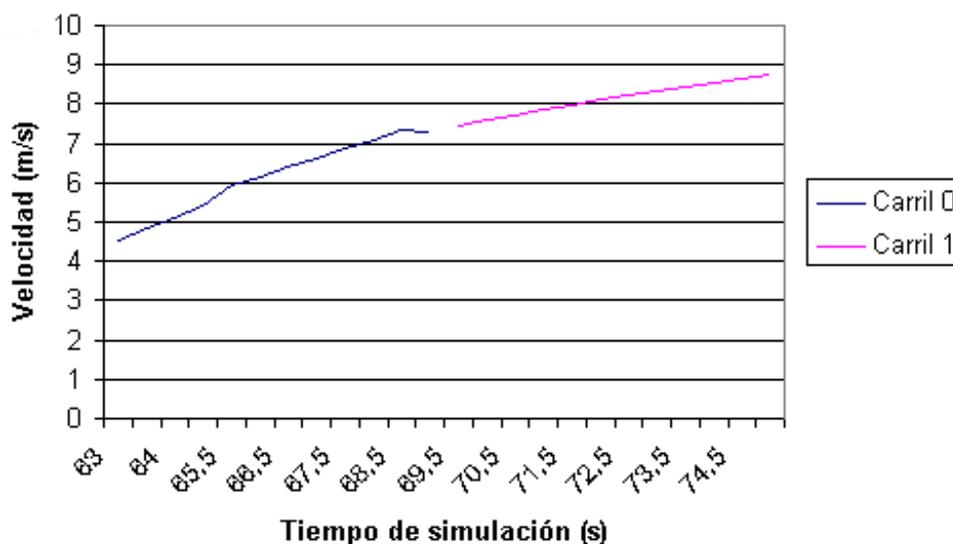


Figura 7.37. Evolución de la velocidad del vehículo 108 en cambio de carril por mejora

Para ilustrar los cambios de carril por cortesía ocurridos en el transcurso de la simulación del escenario extendido, en la tabla 7.18 se muestran los dos cambios de este tipo que han tenido lugar en el tramo 17.

Tipo de cambio	Instante	Vehículo	Tramo	Posición	Carril origen	Carril destino	Vehículo trasero	Gap trasero	Intentos
Cortesía	97.0	175	17	51.84	0	1	186	9.66	13
Cortesía	137.0	282	17	34.69	0	1	495	8.40	9

Tabla 7.18. Ejemplos de cambios de carril por cortesía

En particular se va a analizar el primero de los cambios indicados en la tabla, es decir, el que realiza el vehículo 175 con el consentimiento del 186. Para ello, en la figura 7.38 se muestra la evolución de las posiciones y velocidades de ambos vehículos en el tiempo.

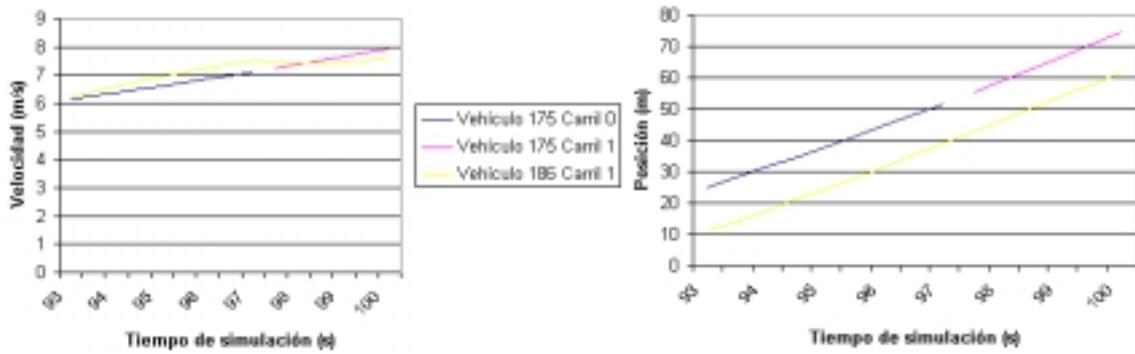
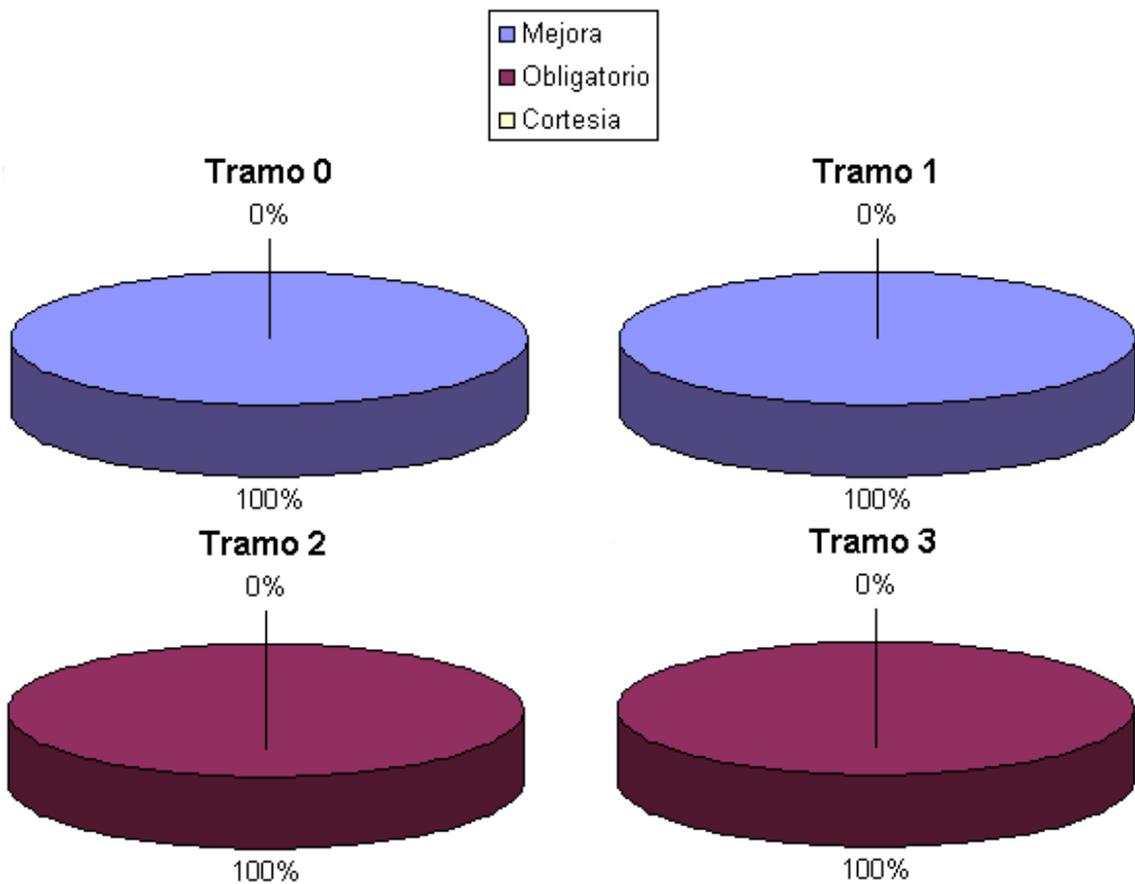
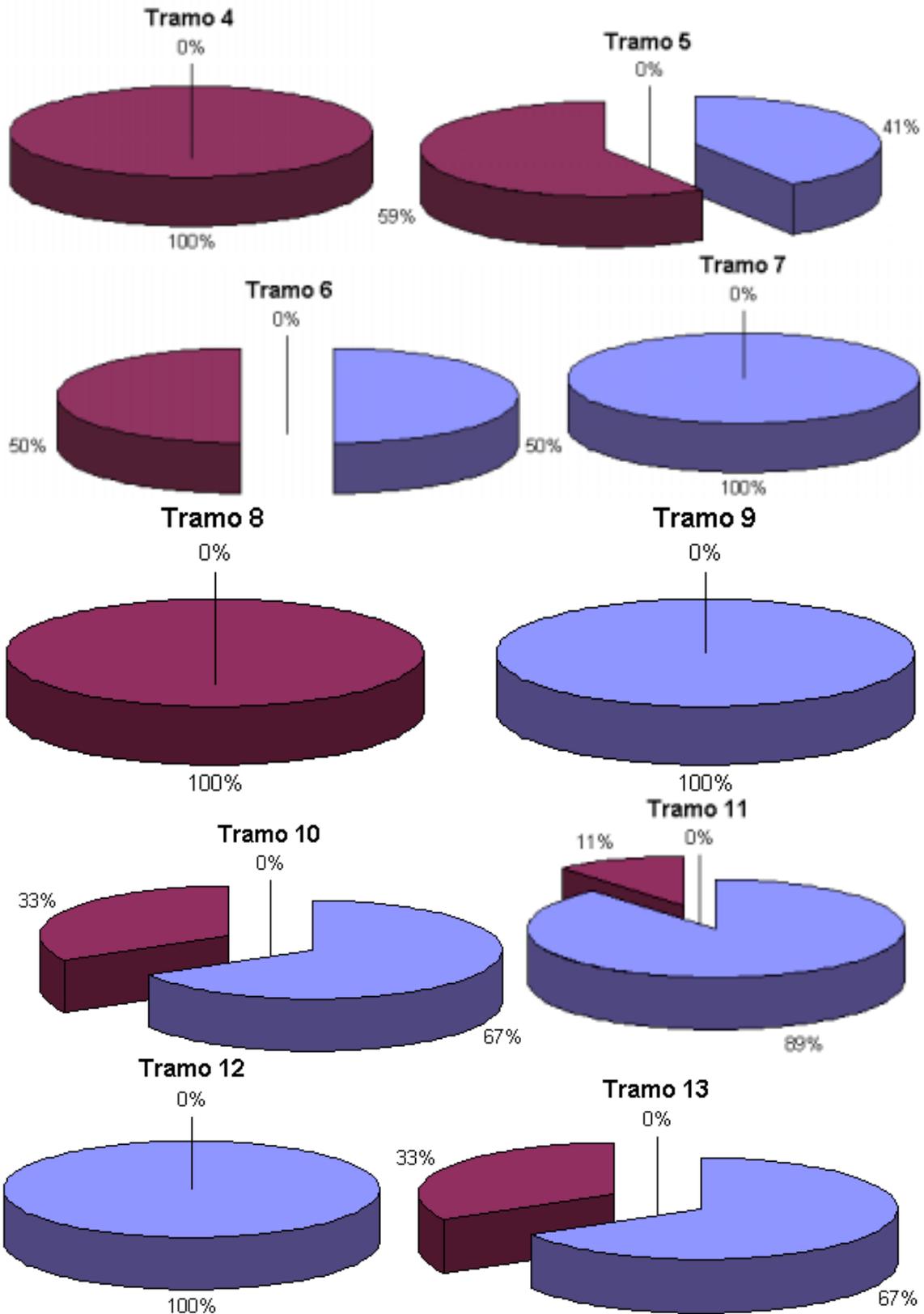


Figura 7.38. Ejemplo de cambio de cortesía

Observando la figura 7.38 se aprecia que el vehículo 186 debe reducir ligeramente su velocidad cuando el vehículo 175 cambia de carril para que aumente la distancia entre ellos y se pueda realizar la maniobra de cambio con seguridad.

Los porcentajes en los que se producen los distintos tipos de cambio de carril en cada tramo y en el viario extendido completo, se muestran en la figura 7.39.





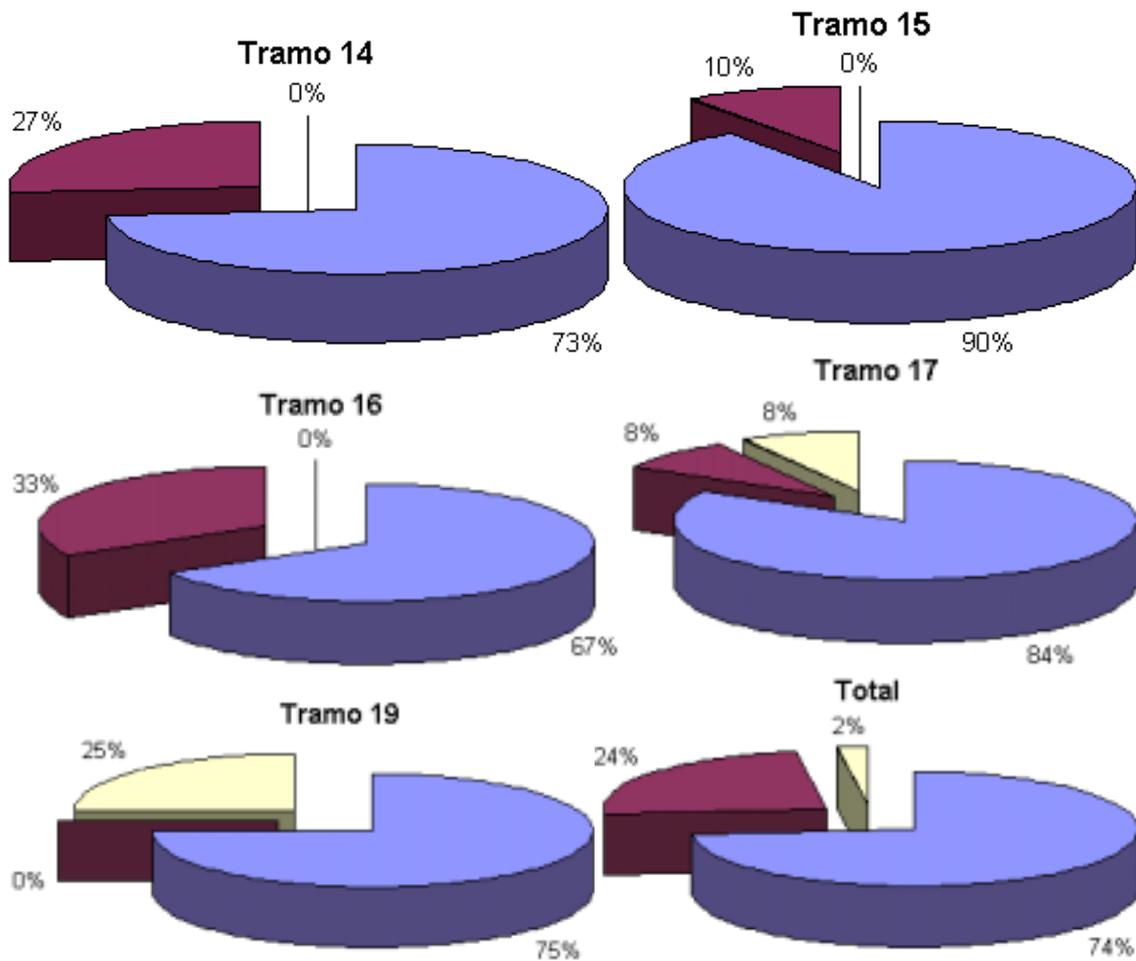


Figura 7.39. Porcentajes de tipos de cambio de carril

El hecho de que los cambios de carril obligatorios se produzcan en posiciones más avanzadas del tramo que los cambios por mejora, queda claramente de manifiesto en el caso del tramo 14 (de 266.06 metros de longitud) mostrado en la figura 7.40, en la que se aprecia que todos los cambios de carril por mejora (a excepción de uno) tienen lugar al comienzo del tramo, mientras que los obligatorios se producen en la parte final del tramo.

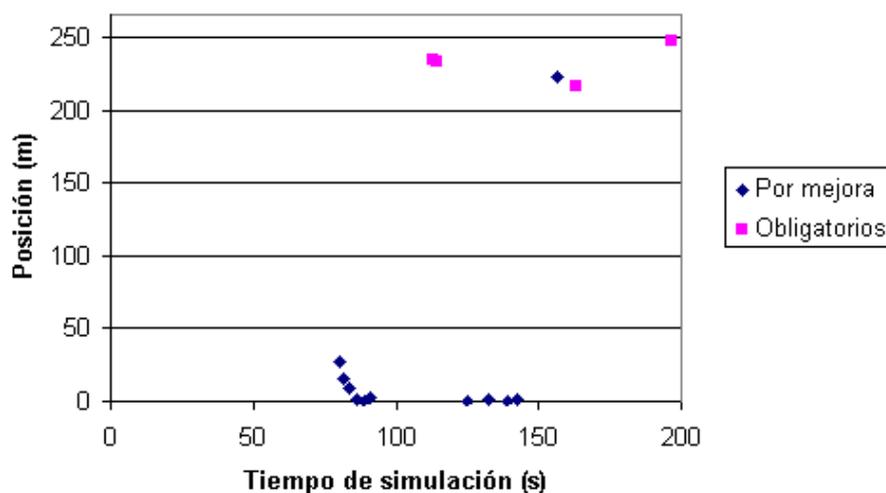


Figura 7.40. Posiciones de realización de cambios de carril en el tramo 14

El número medio de cambios de carril que realiza un vehículo en el transcurso de la simulación es de 0.32. En la gráfica de la figura 7.41 se representa el número de cambios de carril que realiza cada uno de los vehículos (no ficticios) presentes en la simulación, así como una línea roja que indica el valor medio antes comentado.

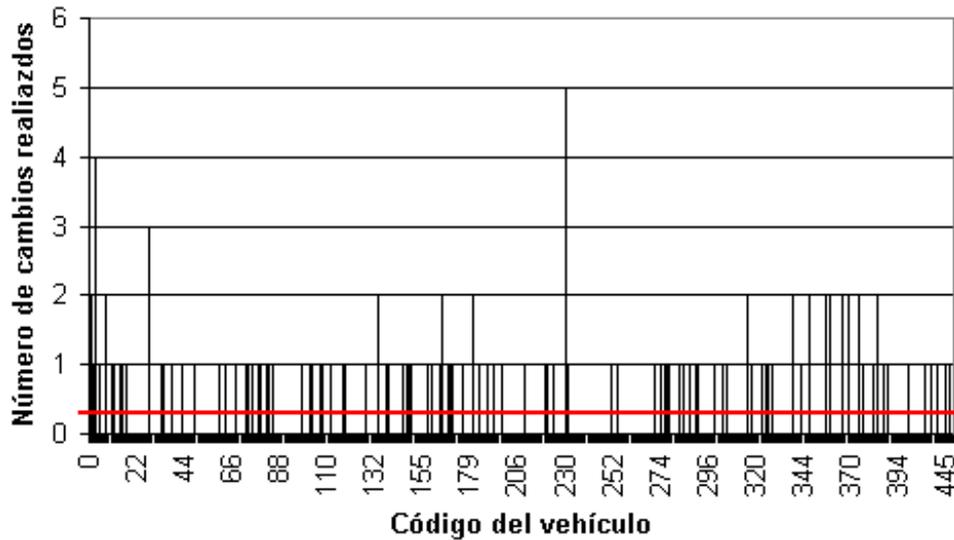


Figura 7.41. Número de cambios de carril por vehículo

Estadísticas genéricas

La evolución del número de vehículos que circulan en cada instante de simulación por cada uno de los tramos del viario extendido se muestra en la gráfica de la figura 7.42.

En ella se aprecia cómo en los tramos donde existe generación de vehículos, dicho número comienza creciendo según la función trapezoidal, para después disminuir conforme los vehículos van abandonando el tramo donde se generan y se incorporan al siguiente tramo de su itinerario.

De esta forma, se aprecia por ejemplo que cuando el número de vehículos que circulan por el tramo 5 comienza a descender, empiezan a circular por el tramo 1, ya que éste es uno de los tramos hacia los que se pueden dirigir los vehículos que circulan por el tramo 5.

Un caso especial de esta situación lo constituyen los tramos 8 y 0, ya que todos los vehículos que se generan en el tramo 8 pasan posteriormente al tramo 0, por lo que las pendientes de subida del número de vehículos en ambos tramos son idénticas.

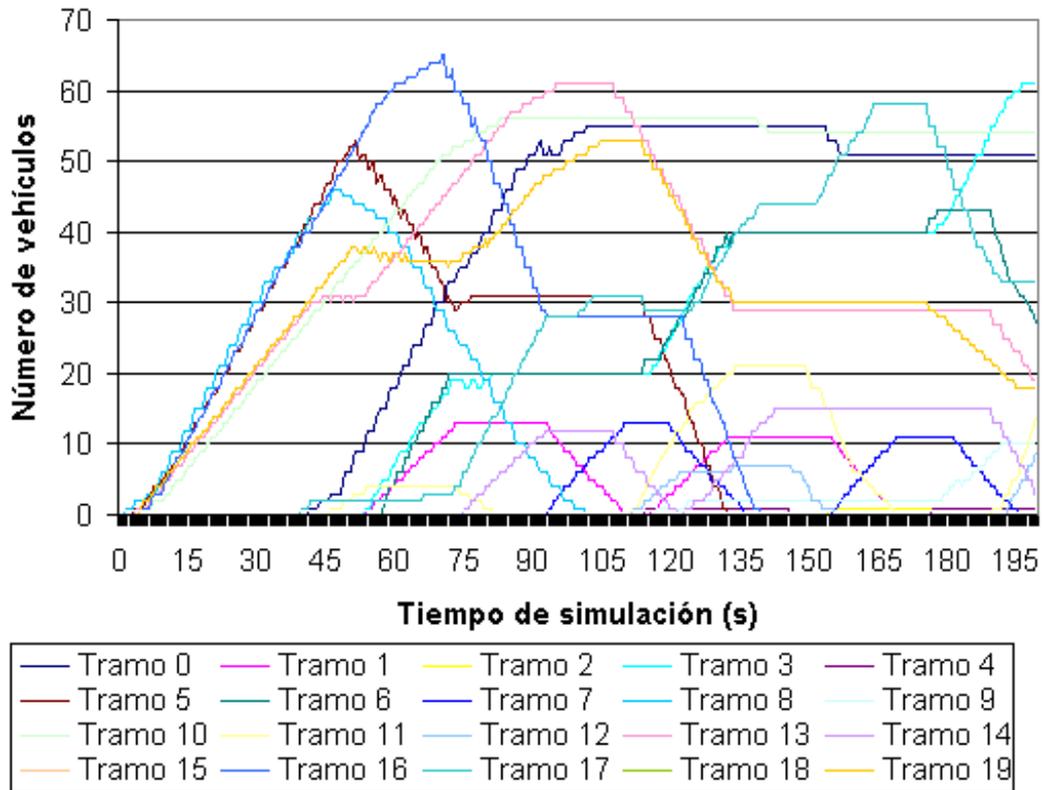


Figura 7.42. Número de vehículos por tramo

El número medio de vehículos que circulan por cada uno de los tramos, se muestra en el diagrama de columnas de la figura 7.43. En ella se aprecia que el tramo en el que hay más vehículos como media es el tramo 10, ya que en él se produce generación de vehículos y el semáforo que regula su circulación hace que los vehículos pasen mucho tiempo esperando en él. Por contra, por el tramo 18 no llega a circular ningún vehículo.

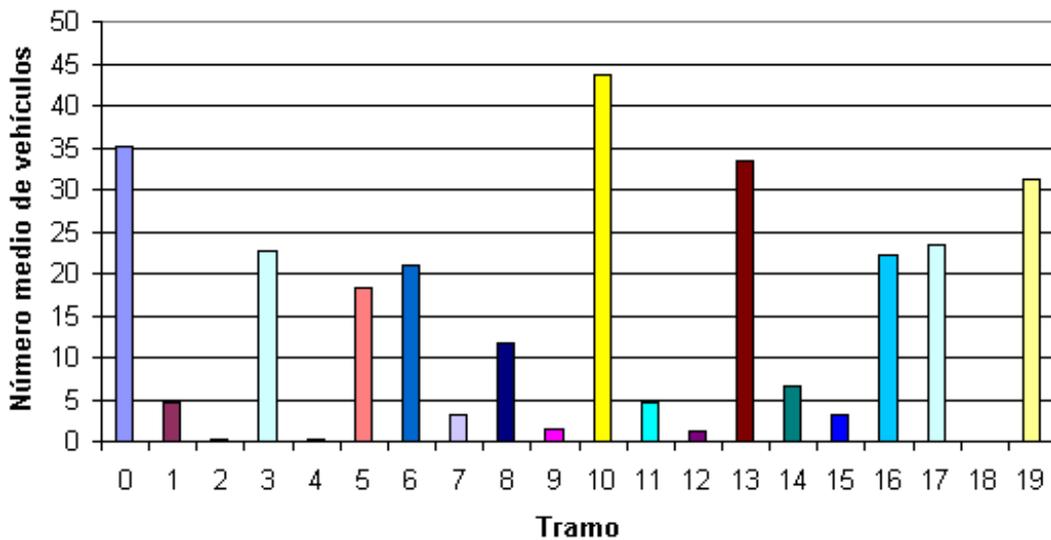


Figura 7.43. Número medio de vehículos por tramo

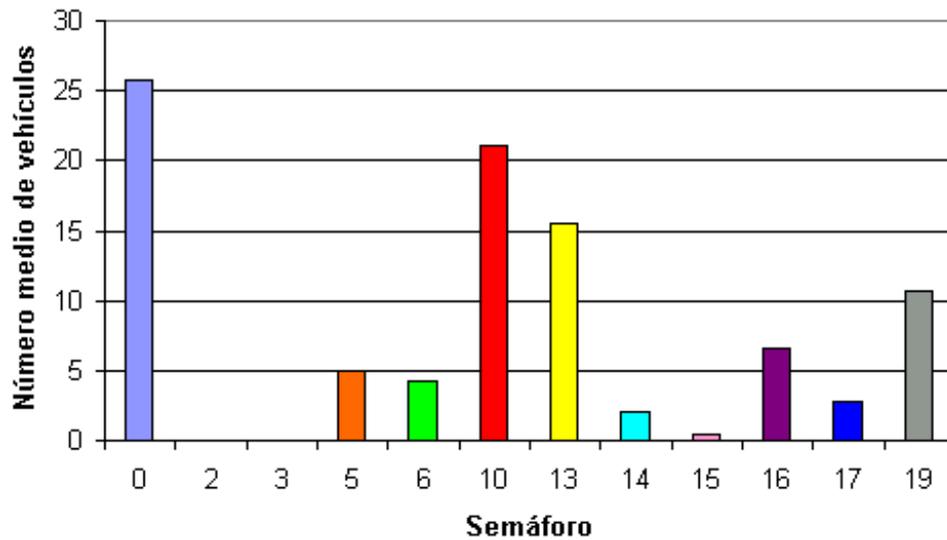


Figura 7.45. Número medio de vehículos en los semáforos

La siguiente estadística que se va a mostrar en la figura 7.46 se corresponde con el tiempo medio de permanencia de los vehículos en cada uno de los tramos del escenario durante el tiempo de simulación.

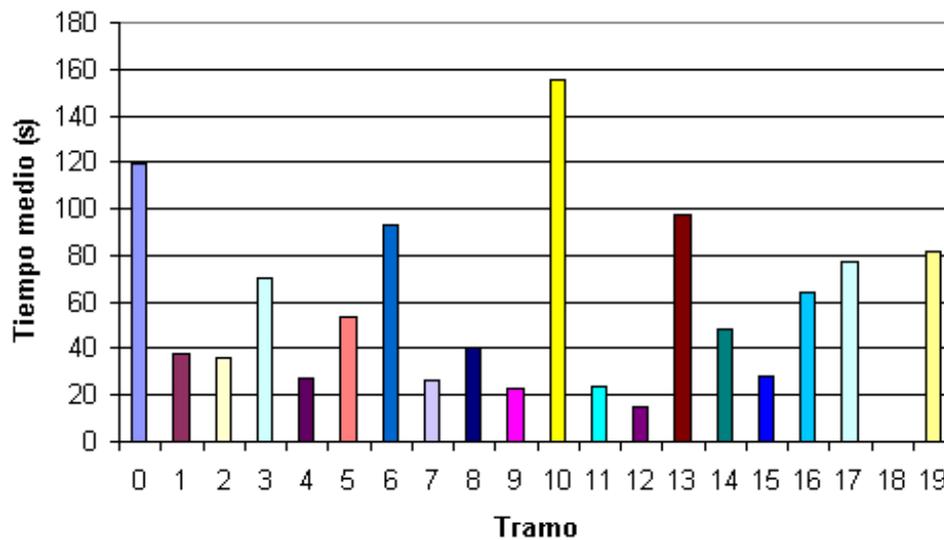


Figura 7.46. Tiempo medio de los vehículos en los tramos

Al igual que se comentó en la prueba básica, la velocidad media desarrollada por los vehículos en los tramos es inversamente proporcional al tiempo que permanecen en ellos. Sin embargo, también influyen otros factores como son la longitud del tramo y la existencia o no de un semáforo que regule la circulación en él. En la figura 7.47 se muestran los valores de estas velocidades medias.

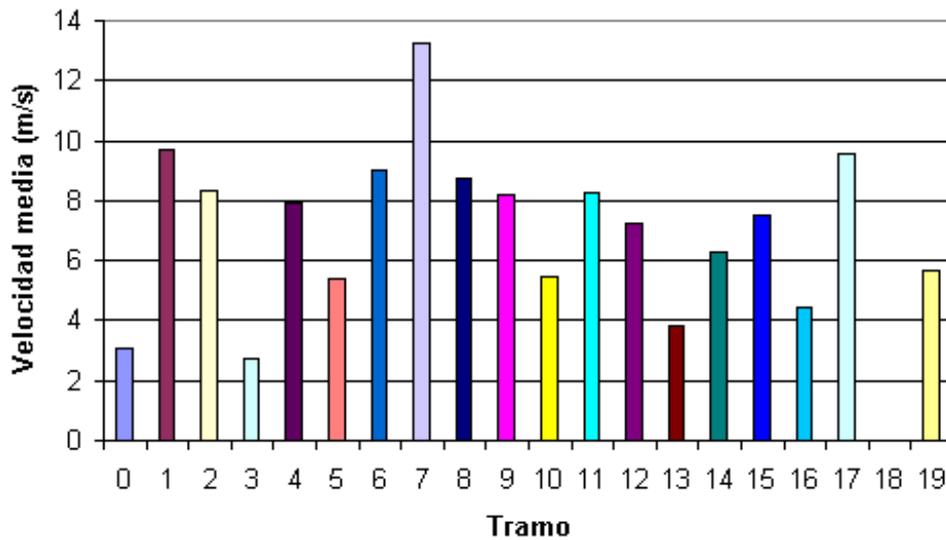


Figura 7.47. Velocidad media en los tramos

La última de las estadísticas se corresponde con el tiempo que permanecen los vehículos activos en el sistema. En la figura 7.48 se muestra el tiempo que están en el sistema todos los vehículos, así mismo, se indica con una línea roja el valor medio de dichos tiempos, que resulta ser igual a 148.55 segundos.

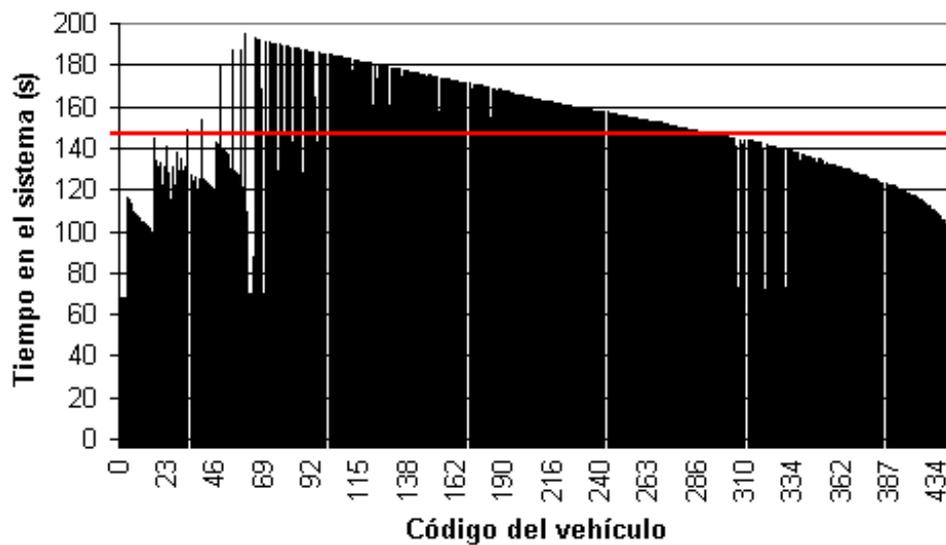


Figura 7.48. Tiempo en el sistema de los vehículos

8 Conclusiones y extensiones

En este apartado se realiza un resumen del trabajo realizado en el proyecto y se proponen posibles extensiones al mismo.

8.1 Conclusiones

El objeto fundamental del proyecto consistía en la realización de una simulación microscópica de tráfico. De todos los componentes del simulador que determinan el comportamiento de los vehículos en el viario, el proyecto se ha centrado en los modelos de cambio de carril y de generación de vehículos, así como del desarrollo de un módulo de estadísticas.

A continuación se comentan brevemente las modificaciones incluidas en los nuevos modelos desarrollados, así como el proceso que se ha seguido para su realización.

Modelo de cambio de carril

El nuevo modelo de cambio de carril desarrollado introduce notables mejoras respecto a los utilizados con anterioridad en el simulador. Entre estas novedades se encuentran la introducción de una duración temporal predeterminada para la realización de la maniobra en función del tipo de conductor que la lleve a cabo, posibilidad de realización de cambios de carril por cortesía, comprobación del cumplimiento de un cierto factor de mejora de velocidad antes de realizar un cambio de carril por mejora y otras muchas más.

La implementación de estas mejoras y su integración en el Simulador Microscópico de Tráfico ya existente, han conllevado que el comportamiento del simulador sea mucho más aproximado a la realidad.

Modelo de generación de vehículos

En cuanto al nuevo modelo de generación de vehículos, el procedimiento empleado se basa en el uso de funciones de generación trapezoidales modificadas por ciertas distribuciones estadísticas. Esto permite introducir un cierto grado de aleatoriedad a la hora de generar los vehículos, pero manteniendo siempre la tendencia indicada por la función trapezoidal.

Módulo de estadísticas

El módulo de estadísticas introducido ha permitido la obtención de datos de gran importancia para el usuario que realiza la simulación. Estos datos, tales como el número y tipo de los cambios de carril que han sucedido, número de vehículos generados, colas formadas en los semáforos, etc., proporcionan al usuario una visión mucho más global del comportamiento del simulador.

Diseño de los modelos

En el desarrollo de estos modelos se ha empleado la metodología orientada a objetos UML (*Unified Modeling Language*) que permite una descripción muy detallada del sistema mediante una serie de diagramas, en concreto se han realizado diagramas de casos de uso, de paquetes y de clases.

Estos modelos han sido implementados empleando un lenguaje de programación orientado a objetos como es C++. Los diagramas de flujo de las funciones más importantes han sido detallados en el apartado 6.

Realización de pruebas

Finalmente, para comprobar la correcta implementación y funcionamiento de los nuevos modelos desarrollados, se han realizado una serie de pruebas de validación y de funcionamiento del sistema completo en escenarios reales de la ciudad de Sevilla.

8.2 Extensiones

A continuación se enumeran una serie de posibles ampliaciones de los modelos que podrían acometerse para mejorar el sistema:

- En el modelo de cambio de carril se podrían considerar agentes externos como puede ser la climatología. También puede resultar interesante introducir nuevos tipos de vehículos como autobuses o motocicletas que tendrían diferente comportamiento a la hora de realizar una maniobra de cambio de carril. Así mismo, se podrían introducir diferentes tipos de carriles (reservados, de aceleración...) que influirían en las maniobras de los vehículos.
- Para la generación de vehículos se podrían emplear puntos de medida, o puntos de medida combinados con las funciones estadísticas ya utilizadas en el modelo desarrollado.
- En cuanto al módulo de estadísticas, una ampliación interesante puede ser la representación gráfica de los valores obtenidos durante la simulación, ya que la situación actual hace que los datos obtenidos sean procesados de manera independiente una vez finalizada la simulación.
- Las pruebas del sistema en escenarios reales podrían ser comparadas con datos obtenidos mediante la realización de una prueba de campo en el propio viario. Esto obligaría a considerar una gran cantidad de nuevos parámetros que hasta ahora eran obviados.

Apéndice A. Notación de los diagramas de flujo

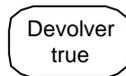
Este primer apéndice contiene la explicación del significado de todos los símbolos y operadores empleados en los diagramas de flujo.

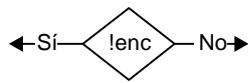
Componentes de un diagrama

I

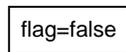
 Indica el comienzo de un diagrama de flujo.

 Indica el final de una función en la que no hay valor de retorno.

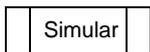
 Señala el final de una función, indicando el valor devuelto por la misma.



Representa la toma de una decisión, si el valor contenido dentro del rombo es cierto, se continúa por la rama marcada con el *Sí*, y en caso contrario por la del *No*.

 Implica un proceso que lleva asociado un cambio en algún dato.

 Señala la dirección en la que circula el flujo de información.

 Indica la llamada a una subrutina.

Operadores lógicos

Son los encargados de realizar operaciones lógicas sobre los datos y suelen aparecer a la hora de evaluar expresiones para tomar una decisión. Estos operadores son los siguientes:

- `&&` : Operador AND (y).
- `||` : Operador OR (o).
- `!` : Operador NOT (no).

Operadores aritméticos

Realizan operaciones aritméticas sobre los datos a los que afectan:

- `+, -, *, /` : Suma, resta, producto o división de datos.
- `++` : Incrementa en una unidad el término situado a su izquierda.
- `--` : Decrementa en una unidad el término situado a su izquierda.

Operadores de relación

Son empleados para comparar dos expresiones:

- `==` : Igual a.

- \neq : Distinto de.
- $<$: Menor que.
- \leq : Menor o igual que.
- $>$: Mayor que.
- \geq : Mayor o igual que.

Operador de asignación

$=$: Es empleado para asignar a la expresión que se encuentra a la izquierda de él, el valor de la expresión que está a su derecha.

Apéndice B. Bibliografía

Alberdi Causse, M. Diseño e implementación de un sistema de asignación de transporte público. Departamento de Organización Industrial y Gestión de Empresas. Proyecto fin de carrera, 2002.

Ahmed, K. I., M. E. Ben-Akiva, H. N. Koutsopoulos and R. G. Mishalani. Models of freeway lane changing and gap acceptance behavior. In J. Lesort (Ed.), *Transportation and Traffic Theory*, pp. 501-515. Pergamon. 1996.

Barrios Muñiz, V.A. Métodos Estadísticos en la Ingeniería: Inferencia. Dpto. de Organización Industrial y Gestión de empresas. Universidad de Sevilla. 1997.

Booch G., Rumbaugh J., Jacobson I. El Lenguaje unificado de modelado. Rational Software Corporation. Addison Wesley. 2000.

Charte F. Programación con C++ Builder. Anaya Multimedia, 2000.

CORSIM User Guide. Technical Report Version 1.0, Federal Highway Administration, US-DOT, McLean, Virginia, 1996.

Duncan, G. Car-Following, Lane-Changing and Junction Modelling. Paramics Technical Report, 1997.

Ferrer J.L. AIMSUN2, Version 2.1 User's manual, preliminary version. Universitat Politecnica de Catalunya. 1996.

García de Jalón J., Rodríguez J.I., Sarriegui J.M., Brazález A. Aprende C++ como si estuviera en primero. Escuela Superior de Ingenieros Industriales de San Sebastián, Universidad de Navarra. 1998.

Gipps. A model for the structure of lane changing decisions. *Transportation Research*, Vol. 20B(5):403-414, 1986.

Gogolla, M. UML for the impatient. University of Bremen, Computer Science Department.

Hoover S. V., Perry R.F. *Simulation: A problem solving approach*. Addison-Wesley Publishing Company. 1989.

Kazi Iftekhar Ahmed. Modeling Drivers' Acceleration and Lane Changing Behavior. Ph. D. thesis, Massachusetts Institute of Technology. 1999.

Kernighan B.W., Ritchie D.M. El lenguaje de programación C. Prentice-Hall Hispanoamericana. Segunda edición. 1991.

Racero Moreno, J. Técnicas de simulación y planificación de transporte para el estudio de problemas de tráfico en entornos urbanos y metropolitanos. Trabajo de investigación. Dpto. de Organización Industrial y Gestión de empresas. Escuela Superior de Ingenieros. Universidad de Sevilla. 2001.

Rumbaugh, J. Object-Oriented Modeling. and Design, Prentice Hall, Inc., 1991

Yang, Q. A Simulation Laboratory for Evaluation of Dynamic Traffic Management Systems. Ph. D. thesis, Massachusetts Institute of Technology. 1997.

Apéndice C. Índice de figuras

<i>Figura 1.1. Representación de nodos y tramos del viario en un grafo</i>	3
<i>Figura 1.2. Movimientos en el interior de una intersección</i>	3
<i>Figura 2.1. Notación cambio carril</i>	8
<i>Figura 2.2. Diagrama de cambio de carril del modelo de Gipps</i>	12
<i>Figura 2.3. Representación del GAP</i>	20
<i>Figura 2.4. Modelo de fusión de Kazi</i>	24
<i>Figura 2.5. Estructura cambio carril obligatorio</i>	25
<i>Figura 2.6. Procedimiento General de Cambio de Carril</i>	28
<i>Figura 2.7. Distancia mínima para cambio de carril obligatorio</i>	29
<i>Figura 2.8. Probabilidad de seleccionar un cambio de carril obligatorio</i>	30
<i>Figura 2.9. Búsqueda de carril para cambio obligatorio</i>	32
<i>Figura 2.10. Probabilidad de aceptar gap en cambio de carril obligatorio</i>	34
<i>Figura 2.11. Función de aceptación de gap delantero</i>	35
<i>Figura 2.12. Aceptación de distancia de seguridad en cambio de carril</i>	36
<i>Figura 2.13. Cambio de carril por cortesía cuando $v_n \geq v_{n-1}$</i>	37
<i>Figura 2.14. Comprobación de distancia en cambio de carril por cortesía</i>	38
<i>Figura 2.15. Probabilidad de aceptar cambio de cortesía</i>	39
<i>Figura 2.16. Probabilidad de aceptación de gap en cambios por mejora</i>	41
<i>Figura 2.17. Aceptación de gap delantero en cambio por mejora</i>	42
<i>Figura 2.18. Creación de un vehículo ficticio al inicio de un cambio de carril</i>	43
<i>Figura 3.1. Función de generación trapezoidal</i>	45
<i>Figura 3.2. Función trapezoidal simple</i>	47
<i>Figura 3.3. Distribución normal</i>	48
<i>Figura 3.4. Distribución exponencial</i>	49
<i>Figura 3.5. Distribución triangular</i>	51
<i>Figura 3.6. Distribución de Poisson de media 5</i>	52
<i>Figura 3.7. Generación de números según distribución de Poisson</i>	52
<i>Figura 3.8. Forma trapezoidal de la función de generación</i>	56
<i>Figura 3.9. Cálculo de área: zona 1, caso A</i>	57
<i>Figura 3.10. Cálculo de área: zona 1, caso B</i>	57
<i>Figura 3.11. Cálculo de área: zonas 1 y 2</i>	58
<i>Figura 3.12. Cálculo de área: zona 2</i>	59
<i>Figura 3.13. Cálculo de área: zonas 2 y 3</i>	60
<i>Figura 3.14. Cálculo de áreas: zona 3, caso 1</i>	60
<i>Figura 3.15. Cálculo de áreas: zona 3, caso 2</i>	61
<i>Figura 3.16. Notación empleada en la generación de vehículos</i>	63
<i>Figura 5.1. Notación de actores</i>	73
<i>Figura 5.2. Notación de casos de uso</i>	73
<i>Figura 5.3. Notación de relaciones</i>	73
<i>Figura 5.4. Casos de uso de nivel 0</i>	73
<i>Figura 5.5. Casos de uso de nivel 1</i>	74
<i>Figura 5.6. Casos de uso de nivel 2. Modelo de Cambio de Carril</i>	75
<i>Figura 5.7. Casos de uso de nivel 2. Modelo de Generación de Vehículos</i>	76

Figura 5.8. Casos de uso de nivel 2. Estadísticas	77
Figura 5.9. Casos de uso de nivel 3. Comprobación de Distancia Disponible.....	78
Figura 5.10. Diagrama de paquetes del simulador.....	80
Figura 5.11. Ejemplo de clase	83
Figura 5.12. Ejemplo de asociación.....	84
Figura 5.13. Clase "Nodo"	85
Figura 5.14. Clase "Micro_tramo"	87
Figura 5.15. Clase "Giro"	92
Figura 5.16. Representación de movimientos	93
Figura 5.17. Clase "Movimiento"	94
Figura 5.18. Clase "Codigo_movimiento"	95
Figura 5.19. Clase "Índice_movimiento"	95
Figura 5.20. Clase "In_movimiento"	96
Figura 5.21. Clase Meso_carril	97
Figura 5.22. Clase "Codigo_carril"	97
Figura 5.23. Clase "Aparcamiento"	98
Figura 5.24. Clase "Aparcamiento_masivo"	99
Figura 5.25. Clase "Codigo_aparcamiento"	100
Figura 5.26. Diagrama de clases "Red"	101
Figura 5.27. Posibles estados de la señal semafórica.....	102
Figura 5.28. Ejemplo de Caja Semafórica	102
Figura 5.29. Estructura, fases y ciclo.....	103
Figura 5.30. Clase "Estructura".....	103
Figura 5.31. Clase "Codigo_estructura"	104
Figura 5.32. Clase "Codigo_senal"	105
Figura 5.33. Clase "Fase"	105
Figura 5.34. Clase "Codigo_fase".....	106
Figura 5.35. Clase "Intermedio"	106
Figura 5.36. Clase "Codigo_intermedio"	106
Figura 5.37. Clase "Senal_semaforica"	107
Figura 5.38. Diagrama de clases "Grupo Semafórico"	108
Figura 5.39. Clase "Vehiculo"	110
Figura 5.40. Clase "Tipo_vehiculo"	115
Figura 5.41. Clase "Tipo_conductor"	116
Figura 5.42. Clase "Configuracion_Microscopico".....	117
Figura 5.43. Clase "Obstaculo"	118
Figura 5.44. Clase "Parada".....	119
Figura 5.45. Diagrama de clases "Simulador Microscópico"	120
Figura 5.46. Clase "Aleatorio"	121
Figura 5.47. Clase "Exponencial"	121
Figura 5.48. Clase "Normal".....	122
Figura 5.49. Clase "Poisson"	122
Figura 5.50. Clase "Triangular"	123
Figura 5.51. Clase "Angulo"	124
Figura 5.52. Parámetros del trapecio	125
Figura 5.53. Clase "Trapecio"	125
Figura 5.54. Clase "Generacion_pendiente"	127
Figura 5.55. Diagrama de clases "Generación Vehículos"	128
Figura 5.56. Clase "Elemento_estadistico".....	128
Figura 5.57. Diagrama de clases "Estadísticas".....	129

<i>Figura 5.58. Clase parametrizada "Lista_inf"</i>	130
<i>Figura 5.59. Diagrama de clases "Listas"</i>	132
<i>Figura 6.1. Estructura general del simulador</i>	134
<i>Figura 6.2. Interfaz para modificar los parámetros de los tipos de conductores</i>	135
<i>Figura 6.3. Procedimiento general de simulación</i>	136
<i>Figura 6.4. Simular tramo</i>	137
<i>Figura 6.5. Estadísticas de tiempo en el sistema</i>	138
<i>Figura 6.6. Estadísticas de tiempo en el tramo</i>	139
<i>Figura 6.7. Estadísticas de vehículos por carril</i>	140
<i>Figura 6.8. Aceptación de gap en cambio de carril para aparcamiento</i>	142
<i>Figura 6.9. Estadísticas de posiciones de aparcamiento</i>	143
<i>Figura 6.10. Actualiza simulación</i>	144
<i>Figura 6.11. Procedimiento general de generación de vehículos</i>	146
<i>Figura 6.12. Obtención de vehículos</i>	147
<i>Figura 6.13. Cálculo del área de una sección del trapezoide</i>	148
<i>Figura 6.14. Cálculo de la altura de un punto según zona del trapezoide</i>	148
<i>Figura 6.15. Comprobación de posición de generación de vehículos</i>	150
<i>Figura 6.16. Estadísticas de generación de vehículos</i>	151
<i>Figura 6.17. Estadísticas de posición de generación de vehículos</i>	152
<i>Figura 6.18. Procedimiento general de cambio de carril</i>	154
<i>Figura 6.19. Cambio de carril obligatorio</i>	155
<i>Figura 6.20. Carril de destino obligatorio para mantenimiento de porcentajes</i>	156
<i>Figura 6.21. Carril de destino obligatorio para giro deseado</i>	158
<i>Figura 6.22. Distancia de cambio de carril obligatorio</i>	159
<i>Figura 6.23. Proceso de aceptación del gap</i>	161
<i>Figura 6.24. Comprobación de gap</i>	163
<i>Figura 6.25. Aceptación del gap de cortesía</i>	164
<i>Figura 6.26. Estadísticas de cambios de carril</i>	167
<i>Figura 6.27. Estadísticas de cambios de carril por vehículo</i>	167
<i>Figura 6.28. Cambio de carril por mejora</i>	168
<i>Figura 6.29. Carril de destino por mejora</i>	170
<i>Figura 6.30. Mejora de condiciones</i>	170
<i>Figura 6.31. Estadísticas de vehículos en semáforos</i>	173
<i>Figura 6.32. Interfaz gráfico</i>	175
<i>Figura 7.1. Incremento de velocidad tras cambio de carril por mejora</i>	190
<i>Figura 7.2. Evolución de velocidades en cambio de carril por cortesía</i>	193
<i>Figura 7.3. Cambios de carril para aparcamiento</i>	194
<i>Figura 7.4. Función de generación trapezoidal</i>	195
<i>Figura 7.5. Área de la sección de trapezoide en cada intervalo</i>	195
<i>Figura 7.6. Viario de la prueba básica</i>	203
<i>Figura 7.7. Forma trapezoidal de generación en los tramos 0, 3 y 5</i>	205
<i>Figura 7.8. Giros y movimientos del nodo 2</i>	205
<i>Figura 7.9. Caja semafórica del nodo 2</i>	207
<i>Figura 7.10. Vehículos generados en cada instante de tiempo</i>	208
<i>Figura 7.11. Generación de vehículos por carriles</i>	208
<i>Figura 7.12. Evolución de velocidad en cambio de carril por mejora</i>	209
<i>Figura 7.13. Ejemplo de cambio de carril por cortesía</i>	210
<i>Figura 7.14. Porcentajes de tipos de cambios de carril</i>	211

<i>Figura 7.15. Cambios de carril ocurridos en el tramo 5</i>	212
<i>Figura 7.16. Número de cambios de carril por vehículo</i>	212
<i>Figura 7.17. Número de vehículos por tramo</i>	213
<i>Figura 7.18. Número medio de vehículos por tramo</i>	214
<i>Figura 7.19. Distribución de vehículos por carriles en el tramo 0</i>	214
<i>Figura 7.20. Vehículos detenidos en los semáforos</i>	215
<i>Figura 7.21. Número medio de vehículos detenidos en cada semáforo</i>	216
<i>Figura 7.22. Tiempo medio de un vehículo detenido en un semáforo</i>	216
<i>Figura 7.23. Número total de vehículos que se detienen en los semáforos</i>	217
<i>Figura 7.24. Tiempo medio de un vehículo en cada tramo</i>	217
<i>Figura 7.25. Velocidades medias en los tramos</i>	218
<i>Figura 7.26. Viario del escenario extendido</i>	219
<i>Figura 7.27. Grafo representativo del viario extendido</i>	219
<i>Figura 7.28. Función trapezoidal de generación de vehículos en escenario extendido</i>	223
<i>Figura 7.29. Giros y movimientos en el nodo 3</i>	223
<i>Figura 7.30. Caja semafórica del nodo 3</i>	224
<i>Figura 7.31. Giros y movimientos en el nodo 5</i>	224
<i>Figura 7.32. Caja semafórica del nodo 5</i>	225
<i>Figura 7.33. Giros y movimientos en el nodo 10</i>	225
<i>Figura 7.34. Caja semafórica del nodo 10</i>	226
<i>Figura 7.35. Vehículos generados en cada instante de tiempo</i>	227
<i>Figura 7.36. Vehículos generados por carril</i>	228
<i>Figura 7.37. Evolución de la velocidad del vehículo 108 en cambio de carril por mejora</i>	229
<i>Figura 7.38. Ejemplo de cambio de cortesía</i>	230
<i>Figura 7.39. Porcentajes de tipos de cambio de carril</i>	232
<i>Figura 7.40. Posiciones de realización de cambios de carril en el tramo 14</i>	232
<i>Figura 7.41. Número de cambios de carril por vehículo</i>	233
<i>Figura 7.42. Número de vehículos por tramo</i>	234
<i>Figura 7.43. Número medio de vehículos por tramo</i>	234
<i>Figura 7.44. Vehículos detenidos en los semáforos</i>	235
<i>Figura 7.45. Número medio de vehículos en los semáforos</i>	236
<i>Figura 7.46. Tiempo medio de los vehículos en los tramos</i>	236
<i>Figura 7.47. Velocidad media en los tramos</i>	237
<i>Figura 7.48. Tiempo en el sistema de los vehículos</i>	237

Apéndice D. Índice de tablas

<i>Tabla 3.1. Contraste de la distribución normal</i>	54
<i>Tabla 3.2. Contraste de la distribución exponencial</i>	54
<i>Tabla 3.3. Contraste de la distribución triangular</i>	55
<i>Tabla 3.4. Contraste de la distribución de Poisson</i>	55
<i>Tabla 7.1. Parámetros de los diferentes tipos de conductores</i>	189
<i>Tabla 7.2. Tramos del escenario básico</i>	204
<i>Tabla 7.3. Movimientos carril-carril del escenario básico</i>	205
<i>Tabla 7.4. Movimientos carril-movimiento del escenario básico</i>	206
<i>Tabla 7.5. Movimientos movimiento-carril del escenario básico</i>	206
<i>Tabla 7.6. Ejemplos de cambios de carril obligatorios en el escenario básico</i>	209
<i>Tabla 7.7. Evolución del recorrido del vehículo 5</i>	209
<i>Tabla 7.8. Ejemplos de cambios de carril por mejora en el escenario básico</i>	209
<i>Tabla 7.9. Cambio de carril por cortesía en el escenario básico</i>	210
<i>Tabla 7.10. Cambios de carril del vehículo 284</i>	213
<i>Tabla 7.11. Tramos del escenario extendido</i>	222
<i>Tabla 7.12. Relación entre tramos y funciones estadísticas</i>	223
<i>Tabla 7.13. Velocidades deseadas por los distintos tipos de conductores</i>	226
<i>Tabla 7.14. Número de vehículos generados en cada tramo</i>	227
<i>Tabla 7.15. Ejemplos de cambios de carril obligatorios</i>	228
<i>Tabla 7.16. Evolución del vehículo 1</i>	228
<i>Tabla 7.17. Ejemplos de cambios de carril por mejora</i>	229
<i>Tabla 7.18. Ejemplos de cambios de carril por cortesía</i>	229