

Trabajo Fin de Máster

Estudio de entornos de simulación en redes de vehículos

Autor

Alain Bedoya Reyes

Director

Dr. Sergio Ilarri Artigas
Departamento de Informática e Ingeniería de Sistemas
Centro Politécnico Superior
Universidad de Zaragoza

Diciembre de 2013



Departamento de
Informática e Ingeniería
de Sistemas
Universidad Zaragoza



Escuela de
Ingeniería y Arquitectura
Universidad Zaragoza

Estudio de entornos de simulación en redes de vehículos

Resumen

La última década ha sido testigo de un creciente interés en las VANETs (*Vehicular Ad-Hoc Networks*), redes formadas entre diferentes vehículos que intercambian datos utilizando tecnologías de comunicación de corto alcance, intentando ofrecer al conductor información dinámica que pueda resultarle de utilidad, brindándole seguridad y evitando riesgos potenciales al proporcionar un mayor conocimiento de las condiciones del camino.

Se han propuesto diversos sistemas de intercambio y gestión de datos para redes de vehículos. Sin embargo, para realizar pruebas de su funcionamiento es conveniente utilizar simuladores del entorno real. De esta manera los costes serían menores, se garantizaría la obtención de resultados fiables y un mayor rendimiento del producto final.

El objetivo de este trabajo es realizar un estudio de los simuladores más utilizados en el ámbito de las redes de vehículos, con el fin de contribuir a la comunidad investigadora en el momento de decidir qué simulador utilizar. Para ello se emplearán artículos realizados por diferentes investigadores y distintos software de simulación que permitan determinar características, ventajas, desventajas, usabilidad, actividad en la comunidad científica y otros criterios de evaluación que nos ayuden a la toma de decisiones en el momento de utilizar algún simulador específico.

A partir de este punto se seleccionarán los que se consideren más adecuados para realizar simulaciones de diversos tipos, incluyendo escenarios de ciudades reales, así como ejemplos propios de los simuladores, donde se darán directrices de cómo llevar cabo la simulación y qué análisis se puede realizar con los resultados obtenidos.

Palabras clave: Redes vehiculares, Simuladores de VANET, Modelos de Movilidad, Tráfico Vehicular, Sistemas de Transporte Inteligente (ITS), Comunicación entre Vehículos (V2V), Comunicación Vehículo Infraestructura (V2I).

Agradecimientos

Quiero agradecer a toda mi familia, los que están y los que se fueron, especialmente a mi esposa y mi padre, por el apoyo y ayuda brindada en todo momento y por ser un ejemplo a seguir. Muchas gracias a mi familia fuera de España que a pesar que están lejos están presentes constantemente sus voces de apoyo y cariño.

Mi agradecimiento a todos mis profesores en la Universidad de Zaragoza por los conocimientos que me han transmitido, en especial a Sergio Ilarri por darme la oportunidad de conocer el mundo de las redes vehiculares y por sus correcciones y recomendaciones a lo largo de este proyecto.

Índice

RESUMEN	I
AGRADECIMIENTOS	II
ÍNDICE.....	III
ÍNDICE DE FIGURAS	VI
ÍNDICE DE TABLAS	VII
CAPITULO 1: INTRODUCCIÓN	1
1.1. REDES AD-HOC	1
1.2. REDES DE VEHÍCULOS	2
1.2.1. CARACTERÍSTICAS DE LAS REDES DE VEHÍCULOS.....	2
1.2.2. ESTÁNDARES	3
1.2.3. ENTORNOS Y MÉTODOS DE COMUNICACIÓN	4
1.2.4 ARQUITECTURA VANET DE REFERENCIA	5
1.2.5 EJEMPLOS DE PROYECTOS DE REDES DE VEHÍCULOS	6
1.3. ESTRUCTURA DE LA MEMORIA	7
CAPÍTULO 2: SIMULADORES PARA REDES DE VEHÍCULOS.....	8
2.1. CONCEPTOS GENERALES DE SIMULADORES DE REDES DE VEHÍCULOS.....	8
2.1.1. MODELOS DE TRÁFICO	8
2.1.2. MODELOS DE MOVILIDAD	9
2.2. SIMULADORES DE TRÁFICO.....	11
2.3. SIMULADORES DE RED.....	11
2.4. SIMULADORES HÍBRIDOS.....	12
2.5. SIMULADORES INTEGRADOS	12
CAPÍTULO 3: ANÁLISIS COMPARATIVO DE SIMULADORES	13
3.1. ANÁLISIS DE ESTUDIOS COMPARATIVOS DE OTROS AUTORES	13
3.2. TABLAS COMPARATIVAS DE SIMULADORES.....	15
3.3. CONCLUSIONES DE TABLAS COMPARATIVAS.....	22
3.3.1. CONCLUSIONES DEL ANÁLISIS DE SIMULADORES DE TRÁFICO.....	23
3.3.2. CONCLUSIONES DEL ANÁLISIS DE SIMULADORES HÍBRIDOS	23
3.3.3. CONCLUSIONES DEL ANÁLISIS DE SIMULADORES INTEGRADOS	24
CAPÍTULO 4: SIMULACIÓN DE ESCENARIOS DE EJEMPLO.....	25
4.1. SIMULANDO CON SUMO.....	26

4.1.1. CONSTRUCCIÓN DE LA RED VIAL.....	27
4.1.2. IMPORTACIÓN DE MAPAS	27
4.1.3. CONSTRUCCIÓN DE LA RED DE TRÁFICO.....	28
4.1.4. PROBABILIDADES EN LAS INTERSECCIONES DE VÍAS	30
4.1.5. GENERAR EL PATRÓN DE MOVILIDAD.....	31
4.1.6. ANÁLISIS DE LAS SIMULACIONES	32
4.2. INTERCAMBIO DE DATOS UTILIZANDO UN EJEMPLO PREDEFINIDO EN VEINS.....	33
4.2.1. ESCENARIO DEL MODELO VEINS	34
4.2.2. NODOS DEL MODELO VEINS	34
4.2.3. ARCHIVO DE INICIO DEL MODELO VEINS	35
4.2.4. LANZAR LA SIMULACIÓN	35
4.2.5. ANÁLISIS DE LOS RESULTADOS	37
4.3. SIMULANDO CON OTRAS HERRAMIENTAS.....	39
4.3.1. ESCENARIO CON ROTONDA EN AIMSUN.....	39
4.3.2. SIMULANDO UN CRUCE PEATONAL CON VISSIM	41
4.4. CONCLUSIONES GENERALES DE LAS SIMULACIONES	45

CAPÍTULO 5: CONCLUSIONES Y TRABAJO FUTURO 46

5.1. CONCLUSIONES GENERALES	46
5.2. VALORACIONES PERSONALES	48
5.3. TRABAJO FUTURO.....	48
BIBLIOGRAFÍA	50

ANEXO A: DESCRIPCIÓN DE SIMULADORES DE TRÁFICO 57

A.1. INTEGRATION	57
A.2. SMARTPATH	57
A.3. VISSIM.....	58
A.4. STRAW	59
A.5. VANETMOBISIM	59
A.6. CORSIM	59
A.7. MITSIM	60
A.8. SUMO	61
A.8.1 HERRAMIENTA MOVE	63
A.9. CITYMOB.....	64

ANEXO B: DESCRIPCIÓN DE SIMULADORES DE RED 65

B.1. NS-2	65
B.2. NS-3	65
B.3. OMNET++	66
B.4. QUALNET.....	67
B.5. JiST	68
B.6. SWANS.....	68
B.7. GLOMOSIM.....	69
B.8. GTNETS	69

ANEXO C: DESCRIPCIÓN DE SIMULADORES HÍBRIDOS E INTEGRADOS

..... **71**

C.1. TRANS.....	71
C.2. MOBIREAL.....	71
C.3. VEINS.....	72
C.4. GROOVENET.....	73
C.5. AIMSUN.....	74
C.6. GROOVESIM.....	75
C.7. NCTUNS.....	76
C.8. iTETRIS.....	78

Índice de Figuras

Figura 1.1: Red ad-hoc.	1
Figura 1.2: Tipos de comunicación a) vehículo a vehículo b) vehículo a infraestructura.	4
Figura 1.3: Arquitectura de C2CC (extraída de [8]).....	5
Figura 2.1: Tabla de simuladores por categorías.	8
Figura 4.1: Características del ordenador para las simulaciones.	25
Figura 4.2: Área de simulación definida en OpenStreetMap: a) Fragmento de la ciudad de Los Ángeles (Estados Unidos), b) Fragmento de la ciudad de Roma (Italia), c) Fragmento de la ciudad de Zaragoza (España).	26
Figura 4.3: Mapa de escenarios importados en SUMO: a) Fragmento de la ciudad de los Ángeles, b) fragmento de la ciudad de Roma, c) fragmento de la ciudad de Zaragoza.	28
Figura 4.4: Interfaz gráfica de MOVE.	28
Figura 4.5: Ejemplo de cómo buscar los id de los <i>edge</i> en SUMO.	29
Figura 4.6: Flujo de vehículos: a) Los Ángeles, b) Roma, (c) Zaragoza.	29
Figura 4.7: Configuración de tipos de vehículos: a) Los Ángeles, b) Roma, c) Zaragoza.	30
Figura 4.8: Cruces definidos en MOVE.	31
Figura 4.9: Generación de archivo de movilidad del escenario de la ciudad de Los Ángeles.	31
Figura 4.10: Simulación del tráfico vehicular (a-b) Puntos de salidas de Los Ángeles, (c-d) puntos de salida de Roma, (e-f) Puntos de salida de Zaragoza.	32
Figura 4.11: Interfaz gráfica del escenario en el modelo VEINS.....	34
Figura 4.12: Nodos del modelo VEINS.....	35
Figura 4.13: Archivo <i>omnetpp.ini</i>	35
Figura 4.14: Mensajes TRACI.	36
Figura 4.15: Ampliación de zona de vehículos de SUMO y OMNeT++ (a) 6 nodos, (b) 15 nodos.....	36
Figura 4.16: Opción <i>Browse Data</i> , donde se observan los datos vectoriales.	37
Figura 4.17: Análisis de la aceleración de los nodos 1 y 2.....	37
Figura 4.18: Análisis de la velocidad del nodo 0.	38
Figura 4.19: Distancia recorrida por los 20 primeros nodos.	38
Figura 4.20: Velocidad máxima de los 10 primeros nodos.	39
Figura 4.21: Importación de mapa en AIMSUN versión limitada.	40
Figura 4.22: Creación de elementos de tráfico.	40
Figura 4.23: Área de trabajo de VISSIM.....	41
Figura 4.24: Configuración de líneas a) vehículos b) peatones.....	42
Figura 4.25: Configuración de semáforos a) grupo 1: vehículos b) grupo 2: peatones..	42
Figura 4.26: a) Entrada de vehículos b) Entrada de peatones.	43
Figura 4.27: Detectores para el cruce de peatones.	43
Figura 4.28: a) Imagen de fondo de escenario real b) imagen integrada.....	44
Figura 4.29: a) Circulación de vehículos b) Circulación de peatones.	44
Figura 4.30: Ejemplo de simulación con VISSIM.	44
Figura A.1: Arquitectura de MOVE.	64
Figura B.1: Arquitectura de JIST.	68
Figura C.1: Arquitectura de las simulaciones y emulaciones distribuidas.	77
Figura C.2: Arquitectura de ITETRIS.	78

Índice de Tablas

Tabla 1.1: Comparación de MANET y VANET.....	2
Tabla 3.1: Comparación de simuladores: a) elementos de software, b) popularidad.....	16
Tabla 3.2: Características de simuladores.	16
Tabla 3.3: Modelo de movilidad y tráfico.	17
Tabla 3.4: Soporte para micro simulación.....	18
Tabla 3.5: Capacidad de simular eventos.	18
Tabla 3.6: Elementos implementados en simuladores microscópicos.	19
Tabla 3.7: Nivel de actividad y control de versiones de los simuladores.....	19
Tabla 3.8: Publicaciones sobre simuladores en la revista <i>IEEE Transactions on Intelligent Transportation Systems</i> (2012-2013).....	20
Tabla 3.9: Análisis de los simuladores según artículos publicados recientemente.	22
Tabla 4.1: Parámetros de generación de tráfico.	27

CAPITULO 1: INTRODUCCIÓN

En este capítulo se definen los conceptos principales que se deben tener en cuenta para la comprensión de la temática principal de este trabajo: la simulación de redes de vehículos. Primeramente se dan a conocer las terminologías de redes *ad-hoc*, MANETs y VANETs caracterizando cada una de ellas y haciendo énfasis en las redes VANETs por la importancia que requieren para esta investigación. Posteriormente se definen estándares de IEEE utilizados en estas redes, así como los entornos y métodos de comunicación que se emplean para el intercambio de datos entre los elementos que conforman la red. Además, se describen los principales proyectos y consorcios a nivel mundial que desarrollan el tema de las redes vehiculares.

1.1. Redes *ad-hoc*

El término *ad-hoc* podría traducirse como “desorganizado” o “específico”. Lo primero se interpretaría por su flexibilidad ante los cambios de la red y lo segundo, cómo ante estos cambios, su infraestructura sólo tiene sentido en ese instante o situación.

Sin embargo, como una definición formal se podría decir que una red *ad-hoc* es una red inalámbrica descentralizada, es decir, que no hay un nodo central sino que consta de nodos móviles que forman una red temporal sin infraestructura externa donde cada nodo envía paquetes de datos a los demás, trabajando en común para que el paquete enviado llegue correctamente a su destino.

Estos nodos pueden estar conectados entre sí (véase Figura 1.1), siendo el protocolo de encaminamiento el responsable del intercambio de información entre los nodos en condiciones de bajo ancho de banda y constante movilidad que presentan estas redes.

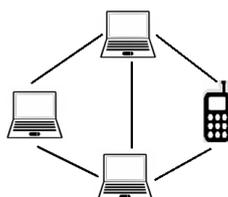


Figura 1.1: Red *ad-hoc*.

El uso de comunicaciones *multi-hop* es otra de las características de las redes *ad-hoc*, son saltos radioeléctricos que se producen en los nodos de la red, es decir, si se envía una información desde un nodo origen pasará por los nodos intermedios antes de llegar a su destino.

Las topologías dinámicas de estas redes obligan a los nodos a tener que adaptarse a los cambios de la red: el tráfico de red, la distribución de carga, la posición de los nodos (ubicación relativa en el tiempo) y ante éstos cambios tienen que auto-organizarse, determinando sus parámetros de configuración: dirección, encaminamiento, *clustering*, etc.

Uno de los principales problemas de las redes ad-hoc es la seguridad ya que son vulnerables a los ataques, donde el atacante puede hacerse pasar por un nodo de la red y desviar paquetes o destruir tablas de encadenamiento. No obstante, dada sus amplias posibilidades de utilización en diferentes entornos (donde es necesario establecer la red en un corto plazo de tiempo), la seguridad ha sido y sigue siendo tratada por los investigadores. [1-3]

Otro problema en una red ad-hoc por su constante dinamismo es la garantía de la calidad de servicio QoS (*Quality of Service*) dado los diferentes entornos donde se encuentren las redes, el ancho de banda o el número de nodos conectados pudiendo ocurrir pérdidas o retrasos en la llegada de paquetes a su destino, sin embargo, esta problemática no ha quedado exenta de las investigaciones. [4-5]

1.2. Redes de vehículos

Las redes VANETs se consideran un tipo específico de red móvil de comunicación MANET (*Mobile Ad-hoc Network*). No obstante, se diferencian en que las primeras utilizan vehículos como nodos de la red e intervienen factores como la topología de la carretera, obstáculos, semáforos, retenciones de tráfico, comportamiento de los conductores, etc. La Tabla 1.1 extraída de [6] muestra un resumen comparativo entre una VANET y una MANET.

CARACTERÍSTICAS	MANET	VANET
Coste de Producción	Barato	Costoso
Cambios en la topología de la red	Lentos	Muy Rápidos
Movilidad	Baja	Alta
Densidad de los Nodos	Escasa	Densa y variable frecuentemente
Ancho de Banda	100 kps	1000 kps
Rango	Hasta 100 m	Hasta 500 m
Duración de un Nodo	Depende de los recursos de energía	Depende del tiempo de vida del vehículo
Comunicación <i>multi-hop</i>	Disponible	No siempre disponible
Confiabilidad	Media	Alta
Patrones de Movilidad de los Nodos	Aleatorio	Regular
Esquema de direccionamiento	Basado en Atributo	Basado en localización
Adquisición de la Posición	Utilizando <i>ultrasonic</i>	Utilizando GPS, RADAR

Tabla 1.1: Comparación de MANET y VANET.

1.2.1. Características de las redes de vehículos

Las redes VANETs se caracterizan por:

- Cambios de su topología: esto se debe a la movilidad de los vehículos ya que pueden moverse de forma arbitraria o seguir un patrón determinado, por ejemplo, los trayectos en una autopista.

- Variabilidad de las velocidades de los vehículos que intervienen: los vehículos que van en la misma dirección por lo general tienen la misma velocidad, lo que implica que la comunicación entre ellos puede tener mayor duración que los vehículos que van en dirección contraria. No obstante, hay que tener en cuenta que las velocidades dependen de factores como el tránsito, las carreteras, retenciones de tráfico, etc.
- Control distribuido de la red: cada vehículo es capaz por sí mismo de procesar la información proveniente de la red sin la necesidad de una infraestructura.
- Consumo reducido de energía: debido a que se puede utilizar la batería de los vehículos.

Se han mencionado algunas de las principales características de las redes VANETs, lo que nos permite tener una idea de las ventajas que nos brindan para realizar simulaciones. En la sección 1.2.2 se comentan los estándares que rigen estas redes.

1.2.2. Estándares

El estándar IEEE 802.11 [7] es un estándar para redes de área local LAN (*Local Area Network*) y redes de área metropolitana MAN (*Metropolitan Area Network*). Se basa en la descripción de las capas física y de nivel de enlace de datos del modelo de interconexión de sistemas abiertos (ISO/IEC 7498-1), más conocido como OSI (*Open System Interconnection*). Soporta todos los estándares de red *ethernet* y se integra con cableado IEEE 802.3, utilizando puntos de acceso (*access point*). Además, garantiza la inter-conectividad y compatibilidad entre los distintos productos WLAN (*wireless local area network*), de ahí su gran importancia. Cada cierto tiempo se hacen revisiones de este estándar, la última revisión fue en el 2012.

En esta investigación nos centraremos en las categorías 802.11b y la 802.11p de este estándar.

IEEE 802.11b fue aprobada en 1999, es una modificación del estándar IEEE 802.11. Tiene una velocidad máxima de transmisión de 11 Mbps, utiliza una banda de 2.4 GHz, como método de acceso al medio el protocolo CSMA/CA (*Carrier Sense Multiple Access with Collision Avoidance*) y una modulación de la señal denominada DSSS (*Direct Sequence Spread Spectrum*) para modular la señal transmitida e intentar protegerla de factores como el ruido o el desvanecimiento de la señal con la frecuencia. Popularmente ésta tecnología se conoce como comunicación inalámbrica (*Wireless o Wifi*).

IEEE 802.11p también conocido como WAVE (*Wireless Access and Vehicular Environment*) fue aprobado en el 2010 para mejorar el desempeño de las redes basadas en CSMA/CA, aplicado a entornos vehiculares proporcionándole la base para un conjunto de aplicaciones que incluyen gestión de tráfico, seguridad en los vehículos, etc. Opera sobre las frecuencias de 5,90 GHz y 6,20 GHz y permite las comunicaciones inalámbricas seguras de V2V (*Vehicle to Vehicle*) y V2I (*Vehicle to Infrastructure*), entornos de comunicación que serán detallados en la sección 1.2.3.

IEEE 802.11p define una estructura de red compuesta por nodos: unidad abordo OBU (*On-Board Unit*), que representa un vehículo equipado con un radio de comunicación de rango corto dedicado DSRC (*Dedicated Short Range Communication*) y unidades fijas RSU (*Roadside Support Units*) que representa un dispositivo fijo con un radio DSRC ubicado en el borde de carreteras.

1.2.3. Entornos y métodos de comunicación

En las redes VANETs habitualmente se distinguen dos tipos de entornos (entorno de comunicación vehicular y entorno de comunicación de infraestructura).

Entorno de comunicación vehicular: En este tipo de entorno existen dos entidades diferentes: vehículos e infraestructura de comunicaciones, donde los vehículos están equipados con una unidad de comunicación a bordo OBU, mientras que la infraestructura de comunicaciones se forma mediante conexiones vía satélites o bien mediante puntos ubicados a lo largo de las carreteras denominados RSU. En el entorno descrito pueden existir diferentes métodos de comunicación siendo los más relevantes el vehículo a vehículo (V2V) y vehículo a infraestructura (V2I) y viceversa, que se detallan a continuación.

Comunicación vehículo a vehículo, se trata del intercambio dinámico de datos entre vehículos cercanos mediante conexión inalámbrica, lo que posibilita un gran número de oportunidades para las mejoras en la seguridad. Por ejemplo, puede utilizar servicios como el aviso de colisión, envío de mensajes entre ellos, información del clima, plazas de aparcamiento libres, evitar congestionamientos del tráfico o sistemas de navegación para activar los servicios de seguridad.

Comunicación vehículo a infraestructura, además de la comunicación entre vehículos incluye una infraestructura formada por nodos a lo largo de la carretera que brinda información a los conductores. Ver la Figura 1.2.

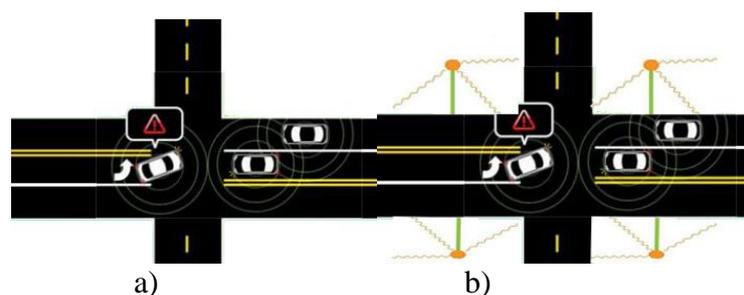


Figura 1.2: Tipos de comunicación a) vehículo a vehículo b) vehículo a infraestructura.

Entorno de comunicación de infraestructura: En este entorno se sitúan los proveedores de servicios a los que se accede a través de esta red, así como las demás entidades de gestión auxiliares como autoridades de certificación, autoridades de gestión del tráfico, etc. Estas entidades pueden establecer conexión con los vehículos a través de la infraestructura de comunicaciones del entorno de comunicación vehicular.

1.2.4 Arquitectura VANET de referencia

En la Figura 1.3 se muestra una arquitectura de C2CC que se toma como referencia cuando se habla de redes VANETs, esta clasifica tres dominios de comunicaciones: (*In-Vehicle Domain*, *Ad hoc Domain*, *Infraestructure Domain*) que se detallan a continuación.

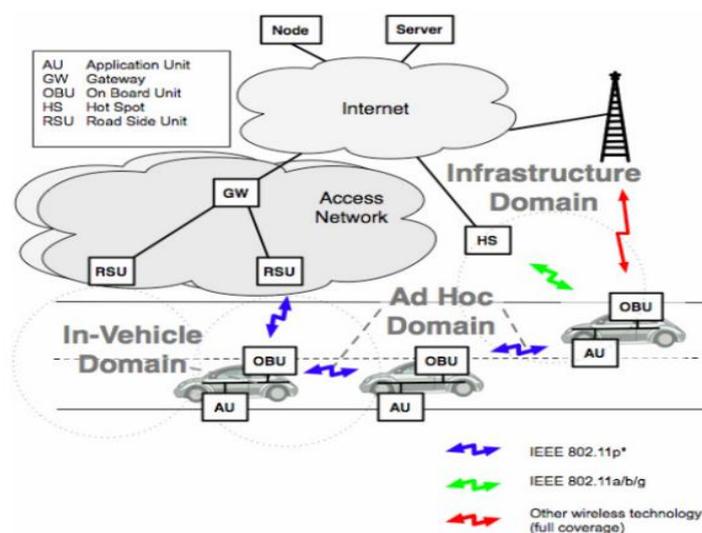


Figura 1.3: Arquitectura de C2CC (extraída de [8])

In-Vehicle Domain: es una red lógica dentro de cada vehículo, que contiene una unidad a bordo (OBU) y unidades de aplicación AUs (*Application Units*) que están compuestas por un dispositivo que ejecuta una aplicación o un conjunto de aplicaciones utilizando las capacidades de comunicación de la OBU, por ejemplo un portátil. Esta conexión puede ser inalámbrica o utilizar conexión cableada.

Ad hoc Domain: se compone de los vehículos equipados con OBUs que conectan con dispositivos (RSU) a lo largo de la carretera. Los RSUs se pueden conectar a una red de infraestructuras, que a su vez puede estar conectada a Internet, con lo cual pueden permitir a las OBUs acceder a la infraestructura. De este modo, es posible para las AUs comunicarse con cualquier servidor de internet a través de una OBU, cuando al menos una infraestructura de RSU esté disponible. Hay que destacar también que dos o más RSUs pueden comunicarse entre sí.

Los accesos a Internet por las RSUs suelen estar controlados por claves mediante la comunicación C2C, mientras que los HS (*Hot Spot*): puntos que ofrecen acceso a internet a través de red inalámbrica y un *router* conectado al proveedor de servicios de internet que puede ser público o privado) se instalan generalmente en un ambiente menos controlado. En el caso de que ni las RSUs ni los HSs puedan proporcionar acceso a Internet, las OBUs pueden utilizar la capacidad de comunicación mediante redes de radio celular GSM (*Global System for Mobile communications*), GPRS (*General Packet Radio Service*), UMTS (*Universal Mobile Telecommunications System*), HSDPA (*High Speed Downlink Packet Access*) también denominada 3G+,

WIMAX (*Worldwide Interoperability for Microwave Access*) y 4G (*four generations*) siempre y cuando estén integradas en la OBU.

Infrastructure Domain: Además de permitir la comunicación entre vehículos, introduce un conjunto de nodos centrales que informan a los vehículos sobre determinados acontecimientos de la carretera: colisión en cruces, condiciones del tráfico, ubicaciones de lugares peligrosos en la carretera (baches, hielo por una nevada), estado del tiempo, etc.; así ante la ausencia de vehículos en un carretera en caso de ocurrir algún accidente con un vehículo, este nodo central podría informar sobre la situación.

Un ejemplo de servicios basados en infraestructura es *Portal Based* [9], una plataforma de servicios telemáticos que ofrece información relacionada con el ocio y el entretenimiento de los pasajeros (información de estacionamientos de vehículos, restaurantes, hoteles, visitas turísticas, etc.). Véase el portal *Virtual City* presentado en [10].

1.2.5 Ejemplos de proyectos de redes de vehículos

Existen varios consorcios que trabajan en proyectos de desarrollo de VANET. A continuación se mencionaran algunos de los más destacados:

- VSC (*Vehicle Safety Communications*) [11] de EE.UU, fundado en el año 2002, en el que se encuentran fabricantes de coches como BMW, Ford, GM, Nissan y Toyota su objetivo es que los coches sean seguros minimizando los accidentes y la pérdida de vidas humanas en las carreteras.
- GST (*Global System for Telematics*) [12] es un proyecto que comenzó en el año 2004, financiado por la Unión Europea, que ha tenido como objetivo crear un diálogo abierto y estandarizado de los sistemas telemáticos de automóviles, definiendo una arquitectura abierta al desarrollo de aplicaciones sobre la red vial, centralizando los sistemas de comunicación del vehículo en un dispositivo no integrado al vehículo, como por ejemplo, el móvil.
- CVIS (*Cooperative Vehicle Infrastructure System*) [13], fundado en el 2006 y desarrollado por la Unión Europea, en el que participan empresas de telecomunicaciones como Vodafone y Telecom, fabricantes de vehículos como BMW y Renault y universidades como CIT (Cork Institute of Technology) en Irlanda. Sus objetivos son: permitir la comunicación entre los vehículos de distintos fabricantes y las infraestructuras viales de distintos tipos, desarrollar componentes básicos que soporten los modelos actuales de aplicaciones y servicios para conductores, empresas y otras partes interesadas; lograr la aceptación de los usuarios, la privacidad y seguridad, interoperabilidad del sistema, riesgo y responsabilidad, necesidades del orden público, modelos de negocio y costo/beneficio, así como una correcta planificación y despliegue de la implementación. Es el grupo de trabajo que mayor énfasis hace en el estudio y desarrollo de C2IS (*Car-to-Infrastructure Services*).

- En C2CC [14] desarrollan un estándar abierto en Europa para los sistemas de comunicación, fomentar la creación de un estándar de comunicación V2V en todo el mundo, promover la asignación de una banda de frecuencia exclusiva para sus comunicaciones y desarrollar estrategias y modelos de negocio para acelerar la introducción en el mercado.

Estos son los principales consorcios y grupos de trabajo líderes en el campo de redes de vehículos, no obstante, cabe destacar que cada vez se suman más investigadores al desarrollo de esta área.

1.3. Estructura de la memoria

El resto de la memoria está compuesta por los siguientes capítulos y anexos:

Capítulo 2: Se introducen los simuladores para redes de vehículos, se comentan los modelos de tráfico y movilidad que utilizan y se detallan los cuatro tipos de simuladores existentes. También se comentan algunas herramientas que se integran a los simuladores con el fin de mejorar sus especificaciones.

Capítulo 3: Se realiza una comparación de los simuladores mediante información disponible en artículos de investigación y páginas oficiales de los simuladores. Se tuvieron en cuenta características como el sistema operativo en el que pueden funcionar, si son de uso comercial o no, fechas de las últimas versiones, si tienen documentación disponible para el usuario, características de micro-simulación, tipo de comunicaciones, elementos que pueden simularse, entre otras. Al finalizar el capítulo, se proponen los simuladores que creemos más adecuados para simular redes de vehículos.

Capítulo 4: Se realizan simulaciones en determinados escenarios a los simuladores seleccionados en el capítulo 3, con el objetivo de mostrar cómo se deben configurar y ejecutar las simulaciones, así como análisis de los resultados de dichas simulaciones y comparación del rendimiento y facilidad de uso de los simuladores.

Capítulo 5: Se presentan las conclusiones del trabajo realizado y se plantean las principales líneas de trabajos futuros.

Anexo A: Se describen los simuladores de tráfico estudiados en esta memoria. Destacando sus principales características, herramientas con las que se integra, arquitectura, año y lugar de desarrollo, etc.

Anexo B: Se detallan características de los simuladores de red mencionados en el capítulo 2. Haciendo énfasis en su facilidad de integración con simuladores de tráfico, sus principales objetivos, si son o no de software libre y elementos de su funcionamiento, entre otros.

Anexo C: Se detallan los simuladores híbridos e integrados presentados en este trabajo. Mostrando sus elementos más importantes, así como sus principales características con el fin de dar una idea clara de cada uno de ellos.

CAPÍTULO 2: SIMULADORES PARA REDES DE VEHÍCULOS

Siempre es importante realizar pruebas a la hora de estudiar las características de una red antes de ser implementada, las VANETs no son la excepción, utilizar técnicas de simulación es un método común que nos permite no tener que invertir demasiado en infraestructuras y poder recrear escenarios muy similares al real para llevar a cabo las pruebas y puesta a punto del modelo final. Existen varias herramientas para llevar a cabo las simulaciones, cada una de ellas con sus características particulares y varias dependen de la integración con otros programas. En la actualidad hay gran aceptación dentro de la comunidad científica en cuanto a clasificarlas en cuatro categorías principales; simuladores de tráfico, simuladores de red, simuladores híbridos y simuladores integrados. En la Figura 2.1 se muestran las cuatro categorías y los simuladores que se estudiarán en esta memoria.

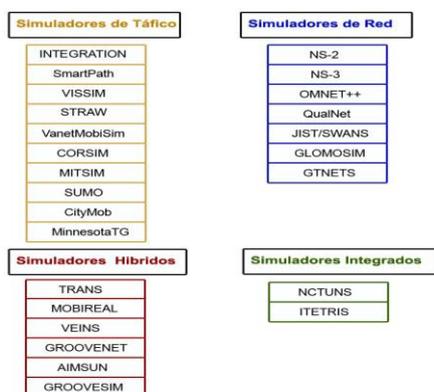


Figura 2.1: Tabla de simuladores por categorías.

2.1. Conceptos generales de simuladores de redes de vehículos

En esta sección se comentan los aspectos generales del modelado y simulación de escenarios en redes vehiculares. En primer lugar se presentarán los modelos de tráfico existentes, que representan el nivel de detalle con que se quiera realizar la simulación. En segundo lugar se exponen los tipos de modelos de movilidad y se mencionan algunos ejemplos de los más utilizados para redes de vehículos. Los modelos de movilidad tienen que ver más con la representación de los elementos del escenario, tratando de imitar el comportamiento de dichos elementos en la simulación de la red, para ello, se deben tener en cuenta aspectos como el comportamiento de los conductores, señales de tráfico, cambios de aceleración y velocidades de los vehículos, obstáculos en la carretera, entre otros.

2.1.1. Modelos de Tráfico

Los simuladores de redes de vehículos cuentan con modelos de tráfico que se encargan del nivel de detalle en la simulación y están clasificados en cuatro tipos: macroscópicos o de macro tráfico, microscópicos o de micro tráfico, mesoscópicos y sub-microscópicos, como se indica en [15].

En los modelos macroscópicos la simulación se realiza partiendo el espacio de simulación en secciones, de ahí su bajo coste computacional, al no simular el entorno real con precisión. La simulación depende de factores como la velocidad del vehículo, la densidad vehicular y el flujo de tráfico, factores que tienen que ver con la topología de la simulación. Mientras que en los modelos microscópicos, sí se logra simular la realidad con un alto nivel de precisión ya que se considera el vehículo como una unidad independiente, poniendo toda la atención a su forma de actuar en las carreteras, siendo el resultado de la simulación la suma de los comportamientos de todos los vehículos, es por ello que su coste es bastante elevado.

El simulador VanetMobiSim, es un ejemplo de modelo macroscópico, no obstante, también utiliza características de nivel microscópico, por su parte SUMO está basado en el modelo microscópico.

Los modelos sub-microscópicos se basan en el modelo microscópico y describen el funcionamiento de las distintas partes de los coches, dividiendo el mismo en subunidades, de ahí que tenga un alto coste computacional al tener que describir las características de los vehículos de forma individual. Además, realizan un seguimiento de los coches, su interacción por las carreteras y los procesos que siguen en la conducción.

Los modelos mesoscópicos representan los sistemas de transporte que analizan el comportamiento de grupos de conductores que muestran conductas similares. Estos modelos combinan propiedades de modelos macro y microscópicos, siendo la unidad de simulación el vehículo individual (como en el modelo microscópico) y los movimientos de éstos siguen una aproximación macroscópica.

Para realizar simulaciones de redes VANETs se recomiendan los modelos microscópicos, debido a que permite modelar los movimientos de cada vehículo individualmente y recrea escenarios reales con un alto nivel de complejidad.

2.1.2. Modelos de movilidad

Para simular correctamente redes de vehículos *ad-hoc* es imprescindible el uso de un modelo de movilidad que represente los nodos con precisión y que trate de imitar los movimientos de los vehículos que conforman la red, así como los cambios de velocidad, dirección, aceleración, etc. El desarrollo de los modelos de movilidad vehicular puede clasificarse en cuatro clases diferentes [16]:

Modelos sintéticos: intenta comprender el comportamiento de los movimientos de los conductores y de los coches para a partir de ellos poder desarrollar un modelo matemático y reproducir sus acciones de forma "virtual". Su principal inconveniente, es la complejidad de los modelos y la incapacidad en ocasiones de poder realizar una "copia fiel" de dichos comportamientos. Ejemplos de modelos de este tipo son:

Manhattan Mobility Model [17]: es utilizado para imitar el movimiento de los vehículos en las calles definidas por un mapa, siendo de utilidad para modelar el movimiento vehicular en áreas urbanas. El mapa se compone por calles horizontales con dos carriles dirección este y oeste y verticales con dos carriles dirección norte y sur, por donde

circula el vehículo a lo largo de toda la red. Para variar la dirección de movimiento de los vehículos se utilizan datos probabilísticos.

RWP (Random Waypoint) [18]: es un modelo bastante utilizado debido a su simplicidad, el movimiento de cada nodo es independiente del resto, eligiendo un destino de forma aleatoria se mueve desde un punto origen a un destino. Tras alcanzar el destino, se elige otro escogiéndolo uniformemente dentro del área de simulación. El principal inconveniente de este modelo es que no es capaz de modelar escenarios reales, debido a que no tiene en cuenta factores importantes como la velocidad de los vehículos, la presencia de obstáculos y las correlaciones entre nodos.

Obstacle Mobility Model [19]: fue diseñado para modelar el movimiento de los vehículos de forma tal que se parezca al entorno real. Permite al usuario definir el escenario proporcionando las dimensiones del terreno y posicionando los obstáculos (coches, edificios, etc.). Para el cálculo de los caminos potenciales que existen dado los obstáculos determinados se utiliza el Diagrama de Voronoi de vértices de los obstáculos, que proporciona caminos que se encuentran en medio de dos obstáculos adyacentes. Posteriormente se ubican los vehículos seleccionando un destino y velocidad aleatoria siguiendo el camino de distancia mínima del diagrama de Voronoi, donde el coste de cada parte del segmento del camino es su distancia euclidiana. Al llegar a su destino, el vehículo se detiene por un periodo de "descanso", luego se selecciona nuevo punto de destino, se calcula la ruta que se necesita para llegar a él y se repite el ciclo.

Modelos basados en datos: teniendo como base los patrones de movilidad de datos reales obtenidos en estudios o encuestas se pretende obtener un modelo que permita simular el comportamiento del tráfico. El modelo *Realistic Vehicular Traces* es un ejemplo de este tipo de modelos.

Realistic Vehicular Traces [20]: estas trazas se obtienen de un simulador de tráfico microscópico *multi-agente* MMTS (*multi-agent microscopic traffic simulator*) desarrollado por K. Nagel en la ETH de Zurich, ahora Universidad Técnica de Berlín, Alemania. Este simulador es capaz de simular, con un alto nivel de realismo, el tráfico público y privado sobre los mapas de las calles reales de Suiza.

Modelos basados en trazas: generan patrones de movimiento basándose en trazas de movilidad reales. Para ello se necesitan hacer varios tipos de mediciones ya que la traza de movimiento depende del tipo de vehículo, no siendo la misma la de un camión que la de un autobús o vehículo pequeño. El proyecto *Reality Mining* del *MIT Media Laboratories* [21] es un ejemplo.

Reality Mining [22]: El objetivo de este experimento fue estudiar las características de los *smart phones* que permitieron a los investigadores conocer las interacciones humanas más allá de la metodología basada en encuestas o en simulaciones. Antes de este experimento, los teléfonos móviles no eran lo suficientemente potentes como para rastrear a las personas.

Modelos basados en simuladores de tráfico: constituyen un eslabón superior frente a los otros modelos. Generan redes VANETs en base a la creación de topologías en

simuladores de tráfico. Para lograr resultados reales se debe tener en cuenta los protocolos de red que se van a utilizar y el modelo de movilidad de los vehículos.

Los modelos basados en simuladores de tráfico suponen una evolución de los anteriores y son los más utilizados en la actualidad.

2.2. Simuladores de tráfico

Se trata de simuladores especializados en el tráfico de vehículos que nos permiten modelar escenarios reales teniendo en cuenta las características (físicas y psicológicas) de los elementos que componen el flujo de tránsito como conductores, varios tipos de vehículos, tranvías, peatones y otros, así como su interacción con elementos de la vía como el número de carriles, cruces complejos, semáforos, obstáculos, líneas de transporte público, etc.

Estos simuladores permiten configurar el movimiento de los elementos en las vías con modelos de movilidad aleatorios en determinados escenarios, incluyendo aspectos como aceleración, velocidad, tiempo de parada y porcentaje de giro en un cruce, etc. Además, ofrecen la posibilidad de generar trazas de movimiento y en varios de ellos estas trazas pueden ser utilizadas por simuladores de red.

La mayoría de estos simuladores, por lo general los más actuales permiten importar mapas reales de cualquier ciudad desde diversas fuentes como TIGER (*Topologically Integrated Geographic Encoding and Referencing*) [23] y OSM (*OpenStreetMap*) [24], también se pueden generar mapas diseñados por el usuario sin restricción y con una gran capacidad de nodos dentro de la red

En este trabajo se estudian simuladores de tráfico como INTEGRATION [25], SmartPath [26], VISSIM [27], STRAW [28], VanetMobSim [29], CORSIM [30], MITSIM [31-32], SUMO [33-35], CityMob [36] y MinnesotaTG [37-39]. Información adicional de estos simuladores puede ser consultada en el Anexo A.

2.3. Simuladores de red

Los simuladores de red son aplicaciones que permite diseñar redes virtuales, ya sea entre ordenadores, enrutadores u otros periféricos que se encuentren conectados entre sí. Permiten realizar varios tipos de pruebas tales como compatibilidad de conexión, comportamiento de los equipos, seguimiento y control de tráfico de red, etc. Además, posibilita la configuración de varios parámetros de la comunicación entre dispositivos por ejemplo, enrutamiento, porcentaje de pérdida de datos, latencias, asignación de direcciones IP, cambios de tarjetas de red y otras funciones que se pueden realizar en un entorno real.

Al realizar una simulación con estas redes, uno de los errores que se pueden presentar es la utilización de patrones de movilidad deterministas cuando en la realidad el comportamiento de los vehículos es aleatorio al depender de las condiciones del camino. Por tal motivo algunos simuladores de red están equipados para permitir patrones de movilidad generados por simuladores de tráfico y cuentan con interfaces

gráficas API (*Application Programming Interface*) para establecer sesiones de conectividad.

Entre los simuladores de red analizados en esta investigación se encuentran NS-2 [40-41], NS-3 [42], OMNET++ [43-44], QualNet [45], JiST/SWANS [46], GloMoSim [47] y GTNets [48]. Estos simuladores se estudian con mayor profundidad en el Anexo B.

2.4. Simuladores híbridos

Los simuladores híbridos integran simuladores de tráfico o movilidad con simuladores de redes a través de una interfaz diseñada para ello. En este caso ambos simuladores se ejecutan en paralelo lo que permite que se comuniquen dinámicamente entre sí mediante la alteración de los patrones de movilidad basados en los flujos de red y viceversa. Lo que nos permite utilizar las mejores prestaciones de ambas herramientas, agrupando modelos de movilidad de última generación con simuladores de redes actuales y eficientes.

Estos simuladores constituyen un paso de avance a la hora de realizar simulaciones de redes de vehículos y evaluar su comportamiento. Sin embargo la principal limitación es que exige numerosos recursos computacionales debido a que los dos simuladores necesitan ejecutarse en paralelo. Aún así son los más utilizados en la actualidad ya que cuentan con muchas ventajas, flexibilidad, gran desarrollo y además los simuladores de tráfico y red que los conforman se desarrollan y actualizan de manera independiente.

Los simuladores híbridos evaluados en este proyecto son TraNS [49-50], MOBIREAL [51], VEINS [52-53], GrooveNet [54-56], AIMSUN [57] y GrooveSim [58]. Sus principales características se presentan en el Anexo C.

2.5. Simuladores integrados

Los simuladores integrados surgieron con el fin de unificar en una sola herramienta las funcionalidades de simuladores de tráfico y de red, debido a que en ocasiones la ejecución en paralelo de estos últimos consumía numerosos recursos.

Para la implementación de estos simuladores se elaboraron nuevos simuladores de red con un desarrollo limitado, donde se compensa la falta de protocolos elaborados con una colaboración directa entre la red y el generador de movilidad.

El principal objetivo de estos simuladores es tener ambos modelos trabajando e interactuando eficientemente. Sin embargo, la principal desventaja es precisamente la mala calidad del simulador de red. Además, no permiten aún incorporar mapas públicos de *Google Maps* y *OpenStreetMap*.

Teniendo en cuenta que en la actualidad la principal tendencia en el desarrollo y mejoras de simuladores de redes es la unión entre protocolos estándares y la eficiencia computacional (utilizando computación paralela y distribuida), los simuladores integrados no parecen la mejor opción para la simulación de escenarios reales. No obstante, en el Anexo C se detallan dos simuladores integrados que se encuentran en proceso de mejoras NCTUns [59-60] e iTETRIS [61].

CAPÍTULO 3: ANÁLISIS COMPARATIVO DE SIMULADORES

Existen numerosas herramientas para realizar simulaciones VANETs, muchas de ellas comentadas en el capítulo anterior. Sin embargo, la técnica de más aceptación en la actualidad para simular una VANET es utilizar un simulador de tráfico vehicular e integrarlo con un simulador de red mediante un simulador híbrido. En este capítulo se hará un análisis, comparación y evaluación de los simuladores de tráfico e híbridos teniendo en cuenta características fundamentales de cada uno de ellos, para definir los de mejores prestaciones que permitan simular una VANET en entornos reales. No se tendrán en cuenta para la comparación los simuladores de red, ya que no consideramos una prioridad estudiarlos con el mismo nivel de detalle y sí como elementos potencialmente integrables con las herramientas de movilidad y las de tipo híbrido para hacer simulaciones más realistas a nivel de comunicaciones.

Para realizar esta comparación se analizarán estudios de otros investigadores sobre el tema, así como instalación, configuración y ejecución de algunos simuladores. Como resultado se seleccionarán los que a nuestro entender mejor se adapten a la simulación de redes de vehículos en entornos reales, para posteriormente realizar simulaciones en estos escenarios que se mostrarán en el Capítulo 4 de esta memoria.

3.1. Análisis de estudios comparativos de otros autores

Las comparaciones realizadas anteriormente están basadas en una bibliografía relevante para éste trabajo. A continuación se muestran cuatro trabajos donde se realizan comparaciones entre algunos de los simuladores mencionados:

- I- Estudio comparativo de simuladores para redes ad-hoc vehiculares. [62]
- II- Simulación de movilidad vehicular para VANETs. [63]
- III- Desarrollo de un simulador VANET basado en agentes móviles. [64]
- IV- Estudio comparativo de simuladores. [65]

Cabe mencionar que algunos de estos trabajos se centran específicamente en tres simuladores como máximo, para poder realizar pruebas exhaustivas sobre ellos. Otros investigadores analizan solamente un grupo de simuladores, por ejemplo, los de tráfico, incluso en ocasiones los estudios fueron realizados hace algunos años por lo que no tienen en cuenta herramientas que han sido mejoradas en la actualidad.

Uno de los aportes de nuestra investigación es el estudio de un gran número de simuladores (25), así como herramientas adicionales que facilitan el trabajo de los mismos. Se incluyen ejemplos de simuladores de todos los grupos existentes (explicado en el Capítulo 2) y se realiza un estudio comparativo bastante completo teniendo en cuenta varios aspectos importantes a la hora de decidir cuál de ellos utilizar en determinada situación. Además se hace una propuesta de simulación al final de éste capítulo y se realizan simulaciones con ella en el Capítulo 4.

A continuación se realiza un análisis de cada uno de los trabajos mencionados:

- **Trabajo de investigación I**

Se realiza una comparación entre un buen número de simuladores, los cuales se encuentran dentro de los cuatro grupos analizados en este trabajo, Tráfico, Red, Híbridos e Integrados.

En este estudio entre los simuladores de tráfico destacan VanetMobiSim por un excelente soporte de trazas y a CityMob por su modelo de tráfico.

Entre los simuladores híbridos e integrados, los autores de este artículo concluyen que se utilizan más estos últimos y recomiendan GrooveNet y NCTUns para simulaciones VANETs, basado en una encuesta realizada a trabajos publicados entre 2000 y 2009.

A pesar de que en este artículo se ha realizado una comparación de varios simuladores para redes VANETs, hay que tener en cuenta que se realizó en el año 2009, por lo que muchos de estos simuladores presentan nuevas versiones, la gran mayoría de ellas con más ventajas que desventajas. Además, no se han considerado simuladores como VEINS, iTETRIS, AIMSUN, entre otros que se analizan en este proyecto y que en la actualidad son muy recomendados por la comunidad.

- **Trabajo de investigación II**

Se analiza a fondo el simulador de tráfico VanetMobiSim en escenarios reales, realizando una comparación con su antecesor CanuMobiSim y después de realizar varios estudios macroscópicos y microscópicos de ambos, se concluye que VanetMobiSim está claramente por encima de su antecesor incorporando aspectos de micro-simulación como señales de tráfico, múltiples carriles, adelantamientos, etc.

Apuntar que se coincide plenamente con los aportes del trabajo mencionado y que sin dudas VanetMobiSim es un gran generador de tráfico y movilidad de código abierto, pero a pesar de esto, no lo recomendamos como la principal opción para simular redes VANETs en esta memoria, debido a que se integra con el simulador de red NS-2, que tiende a dejar de utilizarse dando paso a su sucesor NS-3 o con el simulador de red QualNet que es una herramienta comercial. Además VanetMobiSim solamente ofrece mapas de Europa. Todos los simuladores mencionados en este párrafo se detallan en el **Capítulo 2** de este trabajo.

- **Trabajo de investigación III**

Se mejora un simulador de tipo integrado que se ha implementado en la Universidad de Zaragoza y se realizan comparaciones con otros simuladores similares, este simulador por su descripción es muy parecido a NCTUns pero con el aliciente de que es multiplataforma, gratuito y permite realizar simulaciones de monitorización de áreas geográficas. Incluso, para estudios futuros se propone a los investigadores realizar pruebas con éste simulador.

Según las investigaciones que se han llevado a cabo para la realización de este trabajo fin de máster, recomendamos la utilización de un simulador de tráfico y uno de red,

integrados por un simulador híbrido, ya que se pueden explotar mejor las ventajas de cada uno de ellos, además de que hasta la fecha los integrados se han desarrollado con una limitada simulación de red.

• Trabajo de investigación IV

Analizan el desempeño de tres simuladores de tráfico microscópicos: SUMO, VISSIM y TRANSIMS en un fragmento de una red urbana real de la ciudad de Poznan en Polonia.

Los resultados de esta comparación muestran que VISSIM permite un modelado más preciso, enfatizando en ofrecer un alto nivel de realismo, pero con baja velocidad de simulación. En el lado opuesto aparece TRANSIMS con un modelado a grandes rasgos que no se acerca tanto a la realidad, pero ofrece simulación de alta velocidad y soporta multiprocesamiento. Y finalmente se presenta SUMO como la alternativa, ofreciendo las siguientes ventajas: modelo *car-following*, modelado de grandes redes de vehículos, incluso redes regionales.

También se analizan estos simuladores teniendo en cuenta la facilidad de uso concluyendo que VISSIM es muy fácil de usar, ofrece una potente e intuitiva interfaz gráfica con edición de la red en 2D, así como simulación y visualización en modo 2D/3D, con la desventaja de que es una herramienta de pago. Por otra parte, tanto SUMO como TRANSIMS son software de código abierto y gratuitos que permiten a los programadores ver, analizar y modificar el código fuente, lo que permite la integración con un software externo. Los resultados comparativos de este artículo son similares a algunos de los obtenidos en este trabajo fin de máster.

3.2. Tablas comparativas de simuladores

En esta sección se realizan comparaciones de simuladores de tráfico, híbridos e integrados estudiados en este trabajo. Teniendo en cuenta aspectos importantes de su funcionamiento.

La Tabla 3.1 presenta un resumen de elementos técnicos y popularidad de los simuladores. La tabla está dividida en dos categorías (a) elementos de software y (b) actividad en la comunidad científica.

Simuladores de Tráfico	S.O.	Lenguaje de programación	GUI	Código abierto	Licencia	Popularidad de uso en trabajos de investigación	Documentación disponible	Nivel de actividad
INTEGRATION	Windows 95/98/00 Windows NT	C y BASIC	No	No	comercial	1	1	1
SmartPath	Windows 95	SHIFT	Si	Si	Gratuito	2	2	2
VISSIM	Windows	C/C++	Si	No	comercial	5	4	5
STRAW	Windows o Linux	java	Si	Si	Gratuito	3	4	3
VanetMobiSim	multiplataforma	java	Si	Si	Gratuito	4	3	3
CORSIM	Windows	C++	Si	No	(*) comercial	3	3	3
MITSIM	multiplataforma	Visual C++	Si	Si	Gratuito	3	3	2
SUMO	multiplataforma	C++	Si	Si	Gratuito	5	5	5
CityMob	multiplataforma	java	Si	Si	Gratuito	3	3	3

Simuladores Híbridos e Integrados								
TraNS	multiplataforma	java y C++	Si	Si	Gratuito	3	4	3
VEINS	multiplataforma	java	Si	Si	Gratuito	4	5	5
iTETRIS	Windows	multi-lenguaje	Si	Si	Gratuito	3	5	4
GrooveNet	Linux	C++	Si	No	Gratuito	3	4	3
AIMSUN	multiplataforma	C++ y Python	Si	No	comercial	5	4	5
NCTUns	Linux Fedora 7	C++	Si	Si	Gratuito	4	4	3

(a)

(b)

Tabla 3.1: Comparación de simuladores: a) elementos de software, b) popularidad.

(*) 50% de descuento para instituciones académicas calificadas.

La información de la mayoría de parámetros en esta tabla proviene principalmente de las páginas webs oficiales de cada simulador, en el caso de “documentación disponible y popularidad de uso” se ha tenido en cuenta la experiencia acumulada a través del largo proceso de recopilación y análisis de información sobre simuladores de esta memoria.

La Tabla 3.2 compara aspectos importantes de los simuladores como la precisión del modelo de movilidad, soporte para micro simulación, comportamientos específicos de los conductores, simulación de eventos, integración con otras herramientas y simuladores, entre otros. Más adelante en este capítulo se explicarán los parámetros cuyos valores se encuentran en la escala del 1 al 5 en esta tabla.

Simuladores de Tráfico	Precisión del Modelo de Movilidad	Soporte para micro-simulación	Comportamientos específicos	Simulación de eventos	Elementos que pueden simularse	Tipo de simulación	Herramientas de Importación de mapas	Precisión de comunicaciones inalámbricas	Integración con otros simuladores y herramientas
INTEGRATION	3	3	0	0	2	microscópica	no importa mapas reales	información no encontrada	no
SmartPath	3	2	2	1	2	microscópica	no importa mapas reales	2	no
VISSIM	5	5	4	4	5	microscópica, espacio continuo y tiempo discreto	OpenStreet Map	4	SUMO
STRAW	2	2	2	1	2	microscópica	TIGER	3	JiST/SWANS
VanetMobiSim	3	4	4	4	4	microscópica, macroscópica	GDF/Tiger	4	NS-2, GlomoSim, QualNet
CORSIM	3	4	2	3	4	microscópica	Base de datos TIGER	3	(**) FRESIM NetSim y Qualnet
MITSIM	3	4	4	3	4	microscópica	información no encontrada	3	traffic management simulator (TMS)
SUMO	4	4	4	4	4	microscópica, espacio continuo y tiempo discreto	OpenStreet Map, Tiger, incluso VISSIM	5	TraNS, VEINS, NS-2, NS-3, iTETRIS, MOVE
CityMob	4	4	4	4	4	microscópica	OpenStreet Map	4	NS-2, SUMO
Simuladores Híbridos e Integrados									
TraNS	(*)	(*)	(*)	(*)	(*)	(*)	(*)	5	SUMO, NS-2, QualNet
VEINS	(*)	(*)	(*)	(*)	(*)	(*)	(*)	5	SUMO, OMNeT++/MiXiM
iTETRIS	(*)	(*)	(*)	(*)	(*)	(*)	(*)	4	SUMO, NS-3
GrooveNet	4	4	4	4	4	microscópica, espacio continuo y tiempo discreto	OpenstreetMap	5	(****) Roadnav, GrooveSim
AIMSUN	5	5	5	5	5	microscópica, mesoscópica, macroscópica	OpenstreetMap, (***) Autocad	5	Autocad
NCTUns	3	2	2	3	2	microscópica	no importa mapas reales	5	no

Tabla 3.2: Características de simuladores.

(*) Contiene las mismas puntuaciones que el simulador SUMO, debido a que la generación de movilidad de estos simuladores se lleva a cabo por SUMO.

(**) NETSIM: aplicación interna de CORSIM que se encarga de simular escenarios de tráfico urbano.

FREESIM: aplicación interna de CORSIM que se encarga de simular escenarios de tráfico en autopistas.

(***) Roadnav: navegador disponible en Windows, Linux y Mac que permite localizar la posición de un vehículo utilizando un GPS y realiza indicaciones verbales de una dirección.

(****) Autocad: aplicación de escritorio para de diseño 2D y 3D asistido por ordenador.

A partir de este punto una parte de la información que se resume en las siguientes tablas del Capítulo 3 se ha extraído de otros artículos de investigación sobre el tema [66-70], así como los mencionados en la Sección 3.1, mientras que para el caso de simuladores como SUMO, VEINS, VISSIM y AIMSUN se han realizado simulaciones prácticas en sus respectivas interfaces de usuario, algunas en versiones de prueba para el caso de los simuladores de carácter comercial. Los resultados de dichas simulaciones se pueden observar en el Capítulo 4.

En la Tabla 3.3 se expresa detalladamente qué modelo de movilidad y flujo de tráfico utiliza cada simulador con el fin de ampliar la puntuación otorgada en la tabla 2.

Simuladores de Tráfico	Modelo de movilidad	Modelo de flujo de tráfico
INTEGRATION	modelo origen-destino	car-following, lane changing, and gap acceptance behavior
SmartPath	información no encontrada	modelo mediante asignación de tareas de ASH
VISSIM	modelo origen-destino	car-following, lane changin de Wiedemann
STRAW	Random Waypoint Model	car-following de Nagel-Schreckenberg
VanetMobiSim	Algoritmo Dijkstra (cálculo de rutas entre origen-destino)	Intelligent DriverModel with Intersection Management (IDMIM) Intelligent Driver Model with Lane Changes (IDM-LC)
CORSIM	información no encontrada	modelo de asignación de parámetros de entrada
MITSIM	modelo origen-destino	car-following, lane changin
SUMO	Algoritmo Dijkstra (cálculo de rutas entre origen-destino) y algoritmo de Equilibrio de Usuario (Dynamic User Equilibrium) Dinámico (Dynamic User Equilibrium)	car-following de Krauss
CityMob	Simple, Manhattan y Centro de Ciudad	car-following (Krauss, Wagner, Kerner e IDM)
Simuladores Híbridos e Integrados		
TraNS	(*)	(*)
VEINS	(*)	(*)
iTETRIS	(*)	(*)
GrooveNet	modelo Random waypoint y modelo origen-destino	car-following, traffic lights, lane changing , simulated GPS
AIMSUN	modelo origen-destino	Flujo de entrada y matrices OD con modelo de selección de rutas (car-following , lane changing)
NCTUns	agent-based vehicle mobility control (AVMC) vehicle mobility model by writing a simple C/C++ program	car-following, strolling model , landmark-based movement

Tabla 3.3: Modelo de movilidad y tráfico.

(*) Utiliza los mismos modelos que SUMO.

En la Tabla 3.4 se muestra el soporte para micro simulación de cada uno de los simuladores de nuestro estudio.

Simuladores de Tráfico	Tipo de conducción	(**) Visualización de la simulación	Cambios de carriles	Intersecciones señalizadas y sin señales
INTEGRATION	0	lineal,2D	si	si
SmartPath	2	lineal,2D y 3D	si	no
VISSIM	4	lineal,2D y 3D	si	si
STRAW	2	lineal,2D	no	si
VanetMobiSim	4	lineal,2D	si	si
CORSIM	3	lineal,2D	si	si
MITSIM	3	lineal,2D	si	si
SUMO	4	lineal,2D	si	si
CityMob	4	lineal,2D	si	si
Simuladores Híbridos e Integrados				
TraNS	(*)	lineal,2D	si	si
VEINS	(*)	lineal,2D	si	si
iTETRIS	(*)	lineal,2D y 3D	si	si
GrooveNet	4	lineal,2D	si	si
AIMSUN	5	lineal,2D y 3D	si	si
NCTU _{ns}	3	lineal,2D	no	si

Tabla 3.4: Soporte para micro simulación.

(*) Corresponde al mismo valor de SUMO.

(**) 2D: dos dimensiones, 3D: tres dimensiones.

En la Tabla 3.5 se observan capacidades de cada simulador para recrear eventos como accidentes, obstáculos, plazas de aparcamiento, entre otras.

Simuladores de Tráfico	Accidentes	Vehículos de emergencia	Obstáculos en la carretera	Fenómenos meteorológicos
INTEGRATION	si	no	no	no
SmartPath	no	no	no	no
VISSIM	si	si	si	información no encontrada
STRAW	no	no	no	no
VanetMobiSim	si	no	si	no
CORSIM	si	si	si	no
MITSIM	si	si	si	no
SUMO	si	si	si	no
CityMob	si	no	si	no
Simuladores Híbridos e Integrados				
TraNS	(*)	(*)	(*)	(*)
VEINS	(*)	(*)	(*)	(*)
iTETRIS	(*)	(*)	(*)	(*)
GrooveNet	si	si	si	no
AIMSUN	si	si	si	si
NCTU _{ns}	no	no	no	no

Tabla 3.5: Capacidad de simular eventos.

(*) Corresponde los mismos valores de SUMO.

En la Tabla 3.6 se estudia el nivel de detalle que nos brinda cada simulador, con el fin de tener una idea de cómo se visualizará nuestra simulación y cuántos elementos de la vida real podemos recrear en un escenario simulado.

Simuladores de tráfico	Peatones	Tranvías	Vehículos de dos ruedas	Autobús	Plazas de aparcamiento	Rotondas e intersecciones complejas	Edificios	Entornos rurales, urbanos
INTEGRATION	no	no	no	no	no	1	no	si
SmartPath	no	no	no	no	no	1	no	si
VISSIM	si	si	si	si	si	5	si	si
STRAW	no	no	no	no	no	1	no	si
VanetMobiSim	no	no	no	no	no	3	no	si
CORSIM	no	no	no	si	no	3	no	si
MITSIM	no	no	si	si	no	4	no	si
SUMO	no	no	si	si	no	4	no	si
CityMob	no	no	no	si	no	4	no	si
Simuladores Híbridos e Integrados								
TraNS	(*)	(*)	(*)	(*)	(*)	(*)	(*)	(*)
VEINS	(*)	(*)	(*)	(*)	(*)	(*)	(*)	(*)
iTETRIS	(*)	(*)	(*)	(*)	(*)	(*)	(*)	(*)
GrooveNet	no	no	no	si	no	4	si	si
AIMSUN	no	si	no	si	no	5	si	si
NCTUns	no	no	no	no	no	2	no	no

Tabla 3.6: Elementos implementados en simuladores microscópicos.

(*) Corresponde a los mismos valores de SUMO.

En la Tabla 3.7 se observa la periodicidad de las versiones y actualizaciones de cada simulador.

Simuladores de Tráfico	Fecha de la última versión	Frecuencia de actualizaciones	Soporte
INTEGRATION	versión 2.3 (4/9/2010)	3 años	no
SmartPath	versión 3.0 (1995)	no se actualiza	no
VISSIM	VISSIM 5.40 (2011)	periódica	si
STRAW	Versión 1.3 (2007)	no se actualiza	no
VanetMobiSim	Versión 1.1 (14/2/2007)	no se actualiza	no
CORSIM	TSIS TM 6.3 (08/2012)	1 año	si
MITSIM	versión 4.3 (2010)	Mejoras de usuarios sobre la versión estándar	no
SUMO	versión 0.17.0 (3/5/2013)	periódica	si
CityMob	versión 2 (07/2009)	solo 2 versiones	si
Simuladores Híbridos e Integrados			
TraNS	TraNS v1.2 (03/02/2009)	(*) no se actualiza	no
VEINS	VEINS 2.1 module framework for MiXiM (2013)	periódica	si
iTETRIS	versión 2.1 (20-08-2011)	periódica	si
GrooveNet	versión 2.0.1 (2010)	periódica	si
AIMSUN	Versión 8 (14-11-2011)	periódica	si
NCTUns	versión NCTUns 6.0 (2012)	periódica	si

Tabla 3.7: Nivel de actividad y control de versiones de los simuladores.

(*) Cita extraída de la página web de TraNS: “El desarrollo de TraNS se suspende. Por lo tanto, TraNS no admite la versión más reciente de SUMO y NS-2. En la actualidad, no podemos ofrecer ningún apoyo en relación a TraNS. Si desea realizar simulaciones VANETs con lo último en paquetes de software, consulte la versión TraNSLite”. Ver referencia en la sección 2.4.

En la Tabla 3.8 se hace un estudio de las publicaciones entre los años 2012-2013 en la revista *IEEE Transactions on Intelligent Transportation Systems* que mencionan los simuladores analizados en esta memoria.

Simuladores de Tráfico	No. Artículos	Referencias
VISSIM	4	[70][71][72][73]
STRAW	0	-
VanetMobiSim	0	-
CORSIM	1	[76]
MITSIM	0	-
SUMO	3	[78][79][80]
CityMob	0	-
Simuladores de Red		
OMNET++	1	[81]
NS-2	3	[78][82][83]
NS-3	3	[74][84][85]
Simuladores Híbridos e Integrados		
TraNS	0	-
VEINS	0	-
iTETRIS	0	-
GrooveNet	0	-
AIMSUN	1	[71]
NCTUns	1	[89]
Otros		
	8	[71][72][74][83][85][90][91][92][93]

Tabla 3.8: Publicaciones sobre simuladores en la revista *IEEE Transactions on Intelligent Transportation Systems* (2012-2013).

En la Tabla 3.9 se analiza el tema central de cada artículo y con qué objetivo utilizan algunos de los simuladores estudiados en nuestro trabajo, así como las principales ventajas de su uso. Para este análisis también se tuvieron en cuenta otras revistas importantes como *Transportation Research Part C: Emerging Technologies*.

Simuladores de Tráfico	Artículo	Problema central	Utiliza el simulador para:	Ventajas de utilización del simulador
VISSIM	[70]	Propone un algoritmo <i>Cooperative Vehicle Intersection Control (CVIC)</i> que no requiere de una señal de tráfico y se utiliza para manipular las maniobras de los vehículos individuales de manera que pueden cruzar las intersecciones con seguridad sin chocar con otros vehículos.	Para la simulación vehicular a nivel microscópica y para las optimizaciones de los algoritmos genéticos la interfaz COM de VISSIM.	Su integración con Matlab.
	[71]	En este trabajo se presenta un modelo de micro-simulación de tráfico incorporado a autopistas centrándose en los aspectos relacionados con el control de medición de rampas (RM). El modelo se enfoca en tareas conductuales, como: la cooperación de los conductores al permitir que otros conductores se incorporen al frente de ellos, ya sea por la desaceleración o el paso de los carriles adyacentes.	Se encontraron limitaciones con el rendimiento computacional, la precisión en la representación del flujo de tráfico y la dificultad de integración con gestiones de tráfico avanzado. VISSIM utiliza un tiempo de espera antes de que el vehículo intercambie con el objetivo de hacer frente a esos casos.	No encuentran ventajas de su utilización
	[72]	Se desarrolla una selección de estrategias para el control de atascos de tráfico, se evaluó el impacto de esas estrategias sobre los cambios del tráfico y se simuló procesos de formación de atascos y la disipación con el modelo de transmisión celular.	Lo proponen para utilizarlo en trabajos futuros.	Solamente lo proponen
	[73]	Proponen un sistema cooperativo de intersección para advertencia de colisión (<i>CICWS</i>) que utiliza sensores de tráfico fijos para proporcionar información de alerta a los conductores en tiempo real.	Simular el flujo de tráfico	Aprovechan la exactitud que brinda un simulador como VISSIM para simular escenarios reales.
STRAW	[75]	Presenta un análisis de simuladores de redes y simuladores de tráfico para VANET, incluyendo simuladores que permiten integrarlos.	Se compara con otros simuladores y herramientas similares como CityMob, SUMO, MOVE, VanetMobiSim, PARAMICS.	Se muestran varias ventajas e inconvenientes de su utilización
VanetMobiSim	[75]	Está descrito en la fila de STRAW	Se compara con otros simuladores y herramientas similares como CityMob, SUMO, MOVE, STRAW, PARAMICS.	Se muestran varias ventajas e inconvenientes de su utilización
CORSIM	[76]	Se presenta un método de predicción de flujo de	Simular el tráfico urbano real.	Permite predecir con precisión el flujo

		tráfico a corto plazo para las redes de señales controladas por el tráfico urbano (UTN)		de tráfico a corto plazo, incluso durante los picos de tráfico.
MITSIM	[77]	se ha tratado un enfoque multi-paso para el análisis de sensibilidad de simulación de tráfico utilizando MITSIM.	Realizar las simulaciones de tráfico en una carretera de Porto en Portugal.	Modelo avanzado de comportamiento del conductor, presenta un controlador integrado Del modelo de comportamiento.
SUMO	[78]	Propone un protocolo de direccionamiento que produce una ruta de enrutamiento con el número mínimo de nodos intermedios.	Generar los movimientos de vehículos.	Pueden simular varios escenarios reales, realizan las pruebas con un gran número de vehículos (600)
	[79]	Proporcionar un <i>framework</i> para la recolección de información del tráfico regional a partir de las solicitudes de los usuarios con el objetivo de tener una base para futuras investigaciones en el diseño e implementación de aplicaciones de control de tráfico. Además, proponen un esquema de enrutamiento centrado en los datos y personalizable por el usuario para la entrega de información de tráfico.	Modelar el movimiento y comportamiento de los vehículos en las redes de tráfico.	Permite obtener datos de tráfico dinámicos que son similares a la realidad y lograr un alto grado de rendimiento en las simulaciones de las redes.
	[80]	Se propone un <i>framework</i> para la conducción de la energía (se contabiliza el tiempo y la eficiencia energética) y planificación de rutas sensibles al contexto (se tiene acceso a los datos de tráfico en tiempo real para la estimación de costos del enrutamiento) para vehículos completamente eléctricos (<i>FEVs-fully electric vehicles</i>).	Implementar una plataforma de simulación basándose en el código fuente de SUMO	La utilización de su código fuente para otros desarrollos de plataformas de simulación de vehículos.
CityMob	[75]	Está descrito en la fila de STRAW	Se compara con otros simuladores y herramientas similares como VanetMobiSim, SUMO, MOVE, STRAW, PARAMICS.	Se muestran varias ventajas e inconvenientes de su utilización
Simuladores de Red				
OMNET++	[81]	Nodos de intersección mientras toma en cuenta la conectividad.	Realizar las simulaciones de red.	El nuevo marco aprovecha varias características de simulación de OMNeT++ y Permite simulaciones multi-hop.
NS-2	[78]	Propone un protocolo de direccionamiento que produce una ruta de enrutamiento con el número mínimo de nodos intermedios.	Evaluar el desempeño del protocolo propuesto (BAHG)	Permite la comparación de varios protocolos de enrutamiento como: GPCR y GyTAR. Tienen en cuenta la presencia de obstáculos.
	[82]	Proponen un coordinador controlador de congestión basado en MAC (CCC- MAC), que es un protocolo de acceso al medio basado en intervalos de tiempo que se encarga de los mensajes de emergencia.	Investigar el rendimiento del protocolo de acceso al medio	Su versión NS- 2.33 sigue el modelo de especificaciones de la capa física PHY de las DSRC(<i>Dedicated Short-Range Communications</i>)
	[83]	Se propone un marco para la predicción de tiempo de viaje en carreteras con las mediciones de tráfico disponibles a partir de sistemas de integración de la infraestructura del vehículo (VII).	La evaluación de diferentes protocolos de comunicación y los parámetros de red	Proporcionan un soporte completo de protocolos de comunicación y es de código abierto.
NS-3	[84]	Se presentan estrategias de gestión de información <i>Intraplatoon</i> para la operación de tráfico segura y estable. Se muestran algoritmos para mitigar los retrasos en las comunicaciones.	Probar varios escenarios de un grupo de vehículos con limitaciones de comunicación entre ellos.	Este simulador trae implementado las funcionalidades de QoS (calidad de servicio) de IEEE 802.11e con su función de acceso al medio EDCA.
	[85]	Se centran en los Sistemas de Transporte Inteligentes (ITS) como un método viable de ayudar a los problemas ambientales de los que son responsables el sector del transporte. En particular, se evalúa el impacto en términos de dióxido de carbono (CO2) de la Luz de Tráfico Virtual (VTL), que es un sistema de control de tráfico basado exclusivamente en la comunicación V2V.	Integración con el simulador de tráfico DIVERT. NS-3 se alimenta de la información de la movilidad (la posición, la velocidad y el destino de cada vehículo) e intercambia mensajes con DIVERT utilizando el protocolo VTL, viéndose afectada la movilidad individual de cada vehículo, por ejemplo, tener que detenerse por una intersección.	Se pueden utilizar varias tecnologías de radio acceso (por ejemplo, IEEE 802.11p), así como múltiples características de interfaces/canales y diversos módulos de protocolos.
	[74]	Se describe la incorporación de la comunicación V2V en VATSIM utilizando Network Simulator 3 (NS3)	simular la pila de Internet para múltiples nodos y permitir el uso de medios de canal inalámbrico o por cable con diferentes protocolos en diferentes capas de la pila	Proporciona cambio de canal, colas de prioridad y requisitos específicos de tiempo, pueden modificar su código, etc.
Simuladores Híbridos e Integrados				
TraNS	[75]	Está descrito en la fila de STRAW	Realizan una comparativa con otros simuladores similares como NCTUns y GrooveNet.	Se muestran varias ventajas e inconvenientes de su utilización
VEINS	[86]	Desarrollan un sistema que combina simuladores de tráfico y red para estudiar las estrategias y protocolos necesarios y administración de <i>platoons</i> en varios escenarios. <i>simple IVC-based platoon management</i>	Implementar todos los controladores con SUMO y la interfaz Traci utilizado por VEINS para realizar operaciones como <i>switchOnACC</i> , <i>setCACCLLeaderData</i> , <i>changeToLane (x)</i> , etc. Estos comandos simplifican la implementación de un protocolo de nivel de aplicación en el entorno de VEINS para administrar <i>platoons</i> y para evaluar su desempeño mediante indicadores como la mejora del tráfico o la robustez de las comunicaciones.	Su interfaz Traci que permita la comunicación bidireccional entre el simulador de tráfico y el de red.
iTETRIS	[87]	Se presenta en detalle la plataforma de simulación iTETRIS, se describe su arquitectura, implementación y sus nuevas funcionalidades. Por último se implementan ejemplos de simulaciones en iTETRIS.	La implementación y la evaluación de aplicaciones de gestión de carriles, líneas de autobuses, detección y congestión del tráfico, etc.	Es una plataforma de simulación de código abierto única. Arquitectura modular que permite la integración de dos simuladores de tráfico y red
	[93]	Abordan la extensión del simulador de tráfico que se utiliza en iTETRIS y los estudios de escalabilidad de simulador de redes.. Además el documento presenta la adaptación de las Comunicaciones Inteligentes para Sistemas de Transporte (ICTS)	Realiza pruebas de simulaciones y realizar estudios de simulaciones a gran escala.	Tráfico rodado en comunicaciones inalámbricas, es de código abierto, permite simulaciones a gran escala.

GrooveNet	[88]	Está descrito en la fila de STRAW	Realizan una comparativa con otros simuladores similares como NCTUns y TraNS.	Se presentan ventajas y desventajas de utilizar este simulador en el artículo.
AIMSUM	[71]	Está descrito en la fila de VISSIM	Se encontraron limitaciones con el rendimiento computacional, la precisión en la representación del flujo de tráfico y la dificultad de integración con gestiones de tráfico avanzadas. AIMSUN utiliza un tiempo máximo de espera para que el vehículo pueda intercambiar y después del intercambio, es eliminado del sistema.	No presenta ventajas pues ha sido descartado en este artículo.
NCTUns	[89]	Abordan el tema de la comunicación vehicular en las redes híbridas urbanas y presentan un esquema de enrutamiento híbrido para la difusión de datos en VANETs	Implementar protocolos de enrutamiento híbrido en NCTUns v5.0 y evaluar diferentes escenarios: incluyendo la codificación de red <i>multicast</i> y soporte a RSU.	Simular utilizando el escenario de Manhattan <i>grid</i> de 600 m x 600 m y un modelo de movilidad aleatorio.
Otros				
OARSim	[90]	Estudia el comportamiento de los conductores utilizando la tecnología de realidad aumentada (AR) para redes vehiculares, centrada en dos sistemas AR vehículo (ARV) y AR simulador (OARSIM)	OARSim: Se utiliza para simular comportamientos reales en la carretera incluyendo elementos adicionales que no son reales, para analizar el comportamiento de conductores en determinados escenarios.	Se pueden analizar comportamientos en escenarios a medida, incluyendo los elementos que interesen analizar, como edad de los conductores, vehículos de prueba, peatones, etc.
CarSim	[91]	Se presenta el diseño de un sistema de seguridad activa evitando salidas no deseadas de la carretera. El <i>framework</i> propuesto unifica la evaluación de la amenaza, la estabilidad y el control de pasajeros de los vehículos.	CarSim: se utiliza para simular el modelo de vehículos de alta fidelidad. Los autores realizaron la implementación en <i>dSPACE</i> y simularon utilizando <i>drivers</i> reales interactuando con CarSim.	Se ejecuta en sistema operativo <i>dSPACE</i>
	[92]	Se describe la implementación de un modelo de dirección del conductor que captura los mecanismos de dirección claves del conductor en base a los análisis de los datos de prueba del vehículo en el cambio de curso en el carril doble estándar (DLC).	CarSim para verificar el modelo propuesto sobre la conducción del conductor.	Opciones para ciclos cerrados de dirección, frenos, acelerador.
WLS	[93]	Se describe el enfoque de AUTOPIA, un sistema inteligente de gestión de tráfico basado en las comunicaciones V2I. Se desarrolló un algoritmo de control difuso que tiene en cuenta la distancia de seguridad de cada vehículo y el ajuste de la velocidad para evitar colisiones y un mejor flujo de tráfico.	WLS (EDCA - <i>Enhanced Distributed Channel Access</i>): se utiliza para estimar el retraso y la pérdida de paquetes y simular la estación de control de tráfico y el creciente número de vehículos.	El simulador WLS fue desarrollado para implementar exactamente EDCA. Se ha validado ampliamente por los autores y otros investigadores.
DIVERT	[85]	Se centran en los Sistemas de Transporte Inteligentes (ITS) como un método viable de ayudar a los problemas ambientales de los que son responsables el sector del transporte. En particular, se evalúa el impacto en términos de dióxido de carbono (CO2) de la Luz de Tráfico Virtual (VTL), que es un sistema de control de tráfico basado exclusivamente en la comunicación V2V.	Integración con el simulador de red NS-3; éste intercambia mensajes con DIVERT utilizando el protocolo VTL y DIVERT simula la elección del vehículo líder del grupo de vehículos, los mensajes de luz virtuales, etc. en el contexto del protocolo VTL.	Es un simulador microscópico sofisticado basado en el <i>Intelligent Driver Model</i> (IDM) con un modelo de movilidad validado. Permite la simulación de miles de vehículos con un alto grado de realismo y una amplia gama de configuraciones.
PARAMICS	[71]	En este trabajo se presenta un modelo de <i>microsimulación</i> de tráfico incorporado a autopistas centrándose en los aspectos relacionados con el control de medición de rampas (RM). El modelo se enfoca en tareas conductuales, como: la cooperación de los conductores al permitir que otros conductores se incorporen al frente de ellos, ya sea por la desaceleración o el paso de los carriles adyacentes.	PARAMICS se utilizó para comparar los datos que generaba con los datos del modelo que los autores implementaron y éste mostró mejores resultados. PARAMICS no permitió cambiar por ejemplo, algunos valores de entrada como los coeficientes de utilización de carriles para replicar exactamente los datos reales.	Trabaja con redes relativamente grandes y es un modelo utilizado para aplicaciones de tráfico.
	[72]	Está descrito en la fila de VISSIM	PARAMICS lo proponen para utilizarlo en trabajos futuros.	Solo es propuesto para futuras investigaciones.
	[83]	Está descrito en la fila de NS-2	PARAMICS: se utiliza para el modelado de tráfico además de una API adicional que permite modificar característica de los modelos.	Controla el tiempo basado en el comportamiento del conductor, representa con realismo las condiciones del tráfico
VATSIM	[74]	Está descrito en la fila de NS-3	VATSIM: para construir la red de carreteras y obtener información de la posición de los elementos.	Soporta múltiples elementos como automóviles, autobuses, semáforos, señales de alto, y las paradas de autobuses

Tabla 3.9: Análisis de los simuladores según artículos publicados recientemente.

3.3. Conclusiones de tablas comparativas

A modo de resumen de la comparación realizada en las tablas anteriores, en esta sección se destacan puntos fuertes y débiles que consideramos importantes a la hora de seleccionar los simuladores más convenientes para la realización de simulaciones en nuestro proyecto. Se realizaron comparaciones de simuladores teniendo en cuenta varios aspectos como sistema operativo, interfaz de usuario, documentación disponible,

protocolos, tipo de simulación y otros. Al finalizar recomendamos desde nuestro punto de vista la mejor opción para simular escenarios reales en VANETs.

3.3.1. Conclusiones del análisis de simuladores de tráfico

El simulador de tráfico SUMO permite modelar escenarios con un alto nivel de detalle y posibilita una fácil integración con herramientas del sector, por ejemplo, MOVE, NS-2, OPNET++, etc. SUMO también se caracteriza por su facilidad de uso, es de código abierto y se encuentra en constante desarrollo. Por estos motivos consideramos que es uno de los simuladores de tráfico más asequibles y de mejores prestaciones en la actualidad.

El simulador de tráfico STRAW tiene como desventajas que sólo brinda mapas fuentes de Estados Unidos, además de ser inferior a SUMO en cuanto al modelo de movilidad, elementos que pueden simularse, entre otras. Con lo cual consideramos que existen mejores opciones para realizar simulaciones reales.

VISSIM es una herramienta comercial, a pesar que las comparaciones sugieren que brinda muy buenas prestaciones y está bien valorada por la comunidad, sólo está disponible de forma gratuita una versión de prueba por 30 días que no permite guardar las simulaciones. En este trabajo modelamos un escenario sencillo utilizando VISSIM. Sin embargo, se recomienda realizar pruebas en escenarios complejos para evaluar su rendimiento, en caso de que se tengan los recursos necesarios.

Nuestra propuesta como simulador de tráfico es SUMO, los demás simuladores de tráfico analizados en las tablas no se recomiendan debido a que muchos no están en constante desarrollo o poseen características inferiores a él.

3.3.2. Conclusiones del análisis de simuladores híbridos

VEINS es un simulador híbrido cuya ventaja principal frente a otros simuladores es que acopla de manera bidireccional el simulador de tráfico SUMO y OMNeT++ como simulador de red. Utiliza una interfaz de comunicaciones común denominada TraCi con una arquitectura cliente-servidor donde se intercambian mensajes entre ambos simuladores sobre el protocolo TCP; el resultado se ve reflejado en cambios sobre los patrones de movilidad y sobre el funcionamiento de la red, permitiendo la simulación de VANETs en tiempo y escenarios reales. Otras ventajas de VEINS son: es de código abierto, soporta los estándares IEEE 802.11p e IEEE 1609.4, incluye operación multicanal, calidad de servicio en el acceso al canal y efectos de interferencia y ruido. También soporta la ejecución de las simulaciones en paralelo usando computación distribuida e incluye la importación de mapas reales de varias de las herramientas disponibles para este fin.

Por su parte, TraNS, integra a SUMO como simulador de tráfico y NS-2 como simulador de red. A pesar de no poder utilizar la última versión de SUMO debido a que detuvo su desarrollo, cuenta con una versión reducida llamada TraNSLite que nos permite realizar simulaciones a gran escala. Es relativamente fácil de usar para la simulación de escenarios reales y es un software de libre distribución. Además de permitir la importación de mapas reales mediante varias herramientas, incluye para

TIGER una visualización de la simulación en *Google Earth*. Otras características a señalar de TraNS son la generación de rutas aleatorias y simulación de eventos de tráfico así como el soporte para aplicaciones de seguridad y eficiencia vial. Una de sus desventajas es no poder utilizar NS-3 como simulador de red.

AIMSUN es un software de modelado de tráfico que permite modelar prácticamente cualquier escenario real al mínimo detalle lo que se considera un punto a favor en las simulaciones. Es un programa de licencia comercial con muy buenas prestaciones, posee una versión de prueba por 30 días que se utilizará para un modelado sencillo en nuestro proyecto. AIMSUN además destaca por la excepcional velocidad de sus simulaciones, asignación de tráfico estático y dinámico con las tres variantes de simulación vehicular: mesoscópica, microscópica e híbrida, todo dentro una aplicación informática única. Además permite llevar a cabo evaluaciones de las operaciones de tráfico de cualquier escala y complejidad. Su principal inconveniente para su uso en este trabajo es su elevado coste.

Nuestra propuesta basándonos en el estudio realizado, es VEINS, no obstante, si se cuenta con recursos suficientes valdría la pena realizar pruebas con la versión completa de AIMSUN.

3.3.3. Conclusiones del análisis de simuladores integrados

Comenzar apuntando que nuestra recomendación para la simulación de redes VANETs es una propuesta de simulador híbrido, debido a que se aprovechan características de varios simuladores que actualmente interactúan entre sí bidireccionalmente de forma muy rápida, con lo cual hace pensar que se utiliza una sola herramienta. No obstante, los simuladores híbridos como iTETRIS vienen despuntando bien en el área, sin embargo todavía está en desarrollo de mejoras.

iTETRIS es una plataforma de simulación de red y tráfico integrado, entre sus ventajas se encuentran: es de libre distribución, permite el acoplamiento bidireccional de los simuladores NS-3 para el intercambio de mensajes de comunicación y SUMO para la simulación del movimiento y la gestión del tráfico. También soporta aplicaciones de seguridad y eficiencia vial. Este simulador se recomienda como una opción sobre todo para usuarios de Linux.

GroveNet destaca por su arquitectura modular, lo que permite recrear escenarios en entornos vehiculares fácilmente escalables, soportando múltiples modelos de movilidad. Además, es un simulador híbrido de comunicaciones V2X, acopla bidireccionalmente los módulos de red y tráfico y soporta los estándares para VANET definidos por DSRC.

En resumen proponemos una simulación híbrida utilizando VEINS. También se realizarán simulaciones con versiones limitadas de AIMSUN y VISSIM pues a pesar de ser herramientas comerciales presentan buenas prestaciones.

CAPÍTULO 4: SIMULACIÓN DE ESCENARIOS DE EJEMPLO

El cuarto capítulo de este proyecto está dedicado a la presentación de escenarios de diferentes características seleccionados para realizar las simulaciones. Para estudiar la gran variedad de escenarios que se pueden analizar con la información recopilada, se proponen simular en la Sección 4.1, tres escenarios utilizando el simulador de tráfico SUMO.

- a) Fragmento del mapa de la ciudad de Los Ángeles, Estados Unidos.
- b) Fragmento del mapa de la ciudad de Roma, Italia.
- c) Fragmento del mapa de la ciudad de Zaragoza, España.

El resto del capítulo está organizado de la siguiente manera: la Sección 4.2 presenta una guía por pasos de cómo realizar simulaciones con el simulador híbrido VEINS, basado en el intercambio de datos utilizando un ejemplo que viene por defecto con la herramienta.

En la Sección 4.3 se simula utilizando versiones limitadas de herramientas de pago que presentan muy buenas prestaciones.

Primero se simula un escenario con intersección de tipo rotonda con AIMSUN, donde se explica cómo configurar y utilizar esta herramienta. Debido a que es una herramienta de pago sólo se realizó un sencillo ejemplo para comprobar algunas de sus prestaciones con la versión de pruebas, que no incluye el intercambio de información entre vehículos.

También en esta sección se simula un paso peatonal que cruza una vía de doble sentido utilizando semáforos. Para ello se utiliza VISSIM y se marcan las pautas a seguir para conocer los elementos de esta herramienta, así como para ejecutar simulaciones.

La instalación de los simuladores seleccionados no se detallará a fondo pues no es objetivo de este trabajo. Las simulaciones se realizarán en un ordenador portátil con las características que se muestran en la Figura 4.1.

Edición de Windows

Windows 8
© 2012 Microsoft Corporation. Todos los derechos reservados.
[Obtener más características con una nueva edición de Windows](#)

Sistema

Evaluación:	4,4 Evaluación de la experiencia en Windows
Procesador:	Intel(R) Core(TM) i3-2350M CPU @ 2.30GHz 2.30 GHz
Memoria instalada (RAM):	4,00 GB (3,89 GB utilizable)
Tipo de sistema:	Sistema operativo de 64 bits, procesador x64
Lápiz y entrada táctil:	La entrada táctil o manuscrita no está disponible para esta pantalla

[Información de soporte técnico](#)

ASUS
Inspiring Innovation. Persistent Perfection

Figura 4.1: Características del ordenador para las simulaciones.

4.1. Simulando con SUMO

En esta sección se va a demostrar que uno de los factores que influyen en los resultados de simulaciones para VANET es el escenario escogido, muchos simuladores no rinden igual dependiendo de la topología de la red, como se explica en [94]. Se van a seleccionar escenarios de tres ciudades con diferentes características mostradas en la Tabla 4.1.

Simulamos los tres escenarios seleccionados con la misma configuración utilizando SUMO, para realizar un análisis en distintos escenarios reales. Los escenarios a considerar son muy diferentes en cuanto a su arquitectura, aspectos como la longitud de las calles y regularidad del trazado, entre otras. Con lo cual nos permite realizar un análisis más profundo de SUMO en diversos escenarios.

Los escenarios escogidos fueron, Central LA, un barrio al noroeste del centro de la ciudad de Los Ángeles en Estados Unidos, otro en la ciudad de Roma en Italia al noroeste del Coliseo y un entorno de tráfico urbano del centro de la ciudad de Zaragoza en España, específicamente en los alrededores de un segmento del Paseo María Agustín. Los fragmentos seleccionados tienen una extensión de 2 km². La Figura 4.2 representa el área de simulación sobre la que se despliega el tráfico vehicular, donde se puede ir observando las características de cada fragmento.



Figura 4.2: Área de simulación definida en OpenStreetMap: a) Fragmento de la ciudad de Los Ángeles (Estados Unidos), b) Fragmento de la ciudad de Roma (Italia), c) Fragmento de la ciudad de Zaragoza (España).

En la Tabla 4.1 se muestra la configuración definida para los tres escenarios escogidos.

Parámetros	Valor
Área de simulación	2000m X 2000m
Número de vehículos	100
Tiempo de simulación	300 s
Max Velocidad Vehicular	50 km/h
Capacidad de Aceleración	1,4 m/s ²
Capacidad de Desaceleración	2 m/s ²
Grado de imperfección del conductor	0,5 (escala de 0 a 1)
Longitud del vehículo	5 metros
Modelo de Movilidad	car-following Krauss

Tabla 4.1: Parámetros de generación de tráfico.

4.1.1. Construcción de la red vial

Como primer paso, teniendo en cuenta los escenarios definidos anteriormente se procederá a generar la red vial, por la que viajarán los vehículos. Aquí juega un papel importante nuestro simulador de tráfico SUMO en el que se manejan cuatro términos principales.

- **Nodo:** Intersecciones y uniones de la red.
- **Edge:** Vías, carreteras y calles de la red.
- **Tipo:** Características de las vías, por ejemplo, número de carriles, velocidad máxima de la vía, entre otras.
- **Conexiones:** Cómo conectan los vértices para formar la red vial.

4.1.2. Importación de mapas

Seguidamente se procede a importar los mapas reales utilizando la herramienta OpenStreetMap, se ubica la región a simular y se exporta obteniendo un archivo *.osm*, en nuestro caso será: *angeles.osm*, *roma.osm* y *zaragoza.osm*. Estos archivos se colocan dentro de la carpeta *bin* de SUMO y luego se convierten utilizando la herramienta NETCONVERT perfectamente compatible con SUMO, al formato *xml*, en nuestro caso: *angeles.net.xml*, *roma.net.xml* y *zaragoza.net.xml*. A continuación se muestran los comandos a ejecutar en la consola para realizar la conversión.

```
netconvert --osm angeles.osm -o angeles.net.xml
netconvert --osm roma.osm -o roma.net.xml
netconvert --osm zaragoza.osm -o zaragoza.net.xml
```

Una vez que tenemos los archivos *xml* se abre la interfaz gráfica de SUMO que se encuentra en el directorio principal (*bin/sumo-gui.exe*). Desde aquí importamos los mapas en SUMO y queda representado como muestra la Figura 4.3.

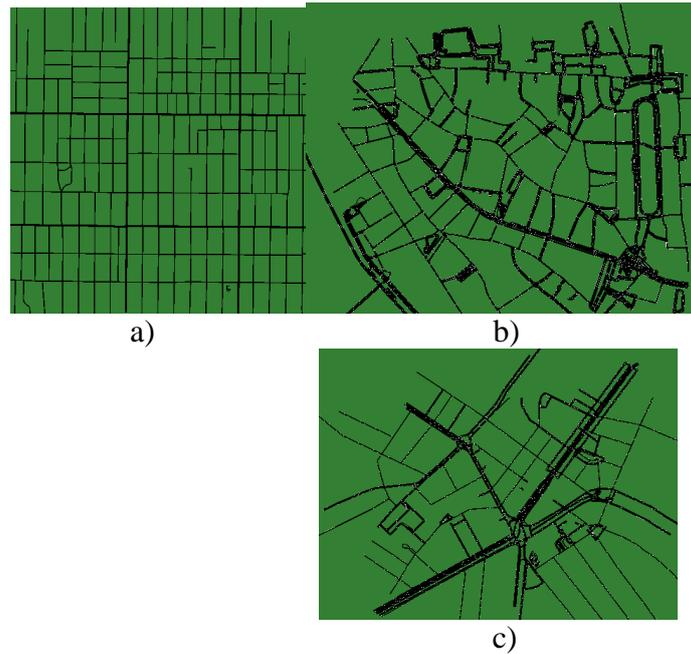


Figura 4.3: Mapa de escenarios importados en SUMO: a) Fragmento de la ciudad de los Ángeles, b) fragmento de la ciudad de Roma, c) fragmento de la ciudad de Zaragoza.

Con esto se da por terminado el proceso de generación de la red vial para la simulación.

4.1.3. Construcción de la red de tráfico

Una vez importado el mapa en SUMO se construye la red de tráfico utilizando la herramienta MOVE, que también es de código libre y permite generar patrones de tráfico para un gran número de vehículos, tomando como parámetros el lugar de salida y de llegada de los vehículos, en este paso se genera un archivo que guarda la configuración de la red de tráfico, para este caso: *angeles.flow.xml*, *roma.flow.xml*, *zaragoza.flow.xml*.

La herramienta MOVE cuya interfaz gráfica se muestra en la Figura 4.4, necesita un pequeño ajuste inicial que consiste en indicarle la ruta donde está instalado SUMO, esta configuración la encontramos en: *file/ set sumo binaries path* que tiene como valor por defecto *untitled*, aquí colocamos en nuestro caso: *C:\sumo-0.15.0\bin*.

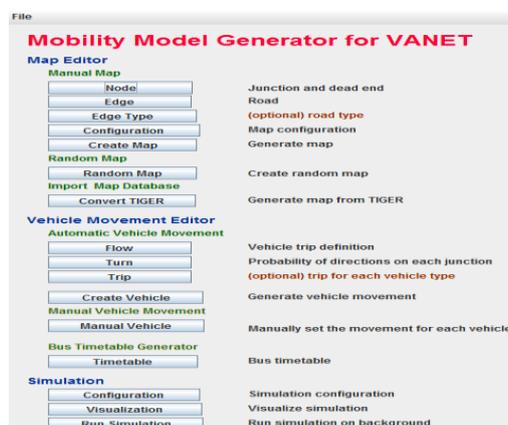


Figura 4.4: Interfaz gráfica de MOVE.

A continuación se procede en la opción *flow* a rellenar los campos para definir el flujo de vehículos:

- *ID*: Identificador del flujo, puede ser cualquier nombre o id.
- *FromEdge*: Lugar de salida de los vehículos, se obtiene al hacer clic con el botón derecho encima de carril que se desee en el mapa importado en SUMO. Ver Figura 4.5.
- *ToEdge*: Lugar de destino, se obtiene igual que el anterior.
- *Begin*: Inicio del flujo.
- *End*: Final del flujo.
- *No Vehicles*: Número de vehículos del flujo

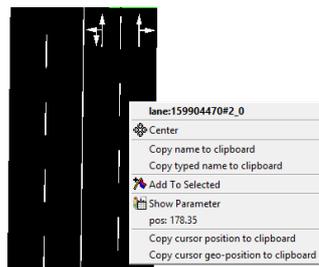


Figura 4.5: Ejemplo de cómo buscar los id de los *edge* en SUMO.

MOVE permite generar tantos puntos de entrada y salida del escenario como se estime conveniente. Para este proyecto se configurarán dos puntos de entrada y dos de salida para cada escenario a simular como se muestra en la Figura 4.6.

Vehicle Flows Definitions						
ID	From Edge	To Edge	Begin	End	No Vehicles	
0	160067418#0	159764373#36	0	300	50	
1	-173465234#25	160067418#11	0	300	50	

a)

Vehicle Flows Definitions						
ID	From Edge	To Edge	Begin	End	No Vehicles	
0	128364582#0	92559300#2	0	300	50	
1	148525611	-2815910	0	300	50	

b)

Vehicle Flows Definitions						
ID	From Edge	To Edge	Begin	End	No Vehicles	
0	124019745#0	146144473	0	300	50	
1	186909512#0	136655271#1	0	300	50	

c)

Figura 4.6: Flujo de vehículos: a) Los Ángeles, b) Roma, (c) Zaragoza.

MOVE se centra en generar los flujos, por lo que omite algunas características de *car-following* necesarias para generar escenarios más realistas, esto se puede solucionar editando a mano los valores correspondientes al tipo de vehículo *vtype* en el archivo *flow.xml* como se muestra en la Figura 4.7.

Se definieron los parámetros para cada vehículo tal cual se describieron en la Tabla 4.1 mostrada anteriormente. En cada escenario se utiliza exactamente la misma configuración, excepto que los vehículos tendrán diferente color. En la parte inferior de la figura se muestran los parámetros que se pueden editar en los vehículos.

```
</flows>
<vtype accel="1.4" decel="2.0" id="v1" length="5.0" maxspeed="50.0" sigma="0.5" color="1,0,0"/>
<flow id="0" from="160067418#0" to="159764373#36" begin="0" end="300" no="50" />
<flow id="1" from="-173465234#25" to="160067418#11" begin="0" end="300" no="50" />
</flows>

<!--
id: identificador del vehiculo
accel: capacidad de aceleración (en m/s2)
decel: capacidad de desaceleración (en m/s2)
sigma: grado de imperfección del conductor desde 0 hasta 1. Donde 0 un conductor totalmente distraído
      y 1 representa el comportamiento de un conductor perfecto.
length: longitud del vehiculo en metros.
maxspeed: máxima velocidad del vehiculo (en m/s).
color: color del vehiculo, dado por tres valores entre 0 y 1 para rojo verde y azul.
OTROS ATRIBUTOS QUE SE PUEDEN DEFINIR
tau: tiempo de reacción del conductor en segundos.
depart: tiempo de la simulación en el que el vehiculo se genera para ser introducido en la red.
-->
```

a)

```
</flows>
<vtype accel="1.4" decel="2.0" id="v1" length="5.0" maxspeed="50.0" sigma="0.5" color="0,1,0"/>
<flow id="0" from="128364582#0" to="92559300#2" begin="0" end="300" no="50" />
<flow id="1" from="148525611" to="-2815910" begin="0" end="300" no="50" />
</flows>
```

b)

```
</flows>
<vtype accel="1.4" decel="2.0" id="v1" length="5.0" maxspeed="50.0" sigma="0.5" color="0,0,1"/>
<flow id="0" from="124019745#0" to="146144473" begin="0" end="300" no="50" />
<flow id="1" from="186909512#0" to="136655271#1" begin="0" end="300" no="50" />
</flows>
```

c)

Figura 4.7: Configuración de tipos de vehículos: a) Los Ángeles, b) Roma, c) Zaragoza.

4.1.4. Probabilidades en las intersecciones de vías

En este paso se procede a darle más realismo al patrón de movilidad, indicando las probabilidades de que un vehículo tome una de las vías posibles al llegar a una intersección, para ello se crea un archivo **.turn.xml*; si este archivo se guarda vacío quiere decir que las condiciones de nuestro escenario serán totalmente aleatorias y caóticas en este sentido. Se pueden definir tantos cruces como existan en el mapa para hacerlo más realista. En este caso no se definirán cruces pues la idea es generar simulaciones lo más aleatoriamente posible y analizar cómo se desempeña SUMO en los distintos escenarios escogidos.

No obstante comentar que los cruces se pueden definir en MOVE utilizando la opción *turn* donde se definen los siguientes parámetros:

- *Begin*: Inicio del flujo.
- *End*: Fin del flujo.
- *From edge*: Carril desde donde se accede al cruce.
- *To edge*: Carril al cual se accede al dejar el cruce.
- *Percentage*: Porcentaje de vehículos que toman el cruce.

La Figura 4.8 muestra un cruce que se realizó en otras simulaciones de escenarios aislados con SUMO; este ejemplo tiene un 10% de girar a la derecha y un 90% de seguir al frente.

Begin	End	From Edge	To Edge	Percentage
0	1000	124019745#1	111847127	10
0	1000	124019745#1	136655284#0	90

Figura 4.8: Cruces definidos en MOVE.

4.1.5. Generar el patrón de movilidad

En esta sección se procede a generar el archivo de movilidad para la simulación, en nuestro caso serían: *angeles.rou.xml*, *roma.rou.xml* y *zaragoza.rou.xml*. Para este fin se utiliza la opción *create vehicle* de MOVE donde se colocan las rutas de los tres archivos creados anteriormente y que se explican en este capítulo: **.flow.xml*, **.turn.xml*, **.net.xml*, se coloca la ruta del archivo de salida y se presiona la opción *ok*. Siempre es importante verificar en la consola que no existan errores.

La Figura 4.9 muestra como quedaría la ventana final para el escenario de Los Ángeles, los otros dos escenarios son muy similares.

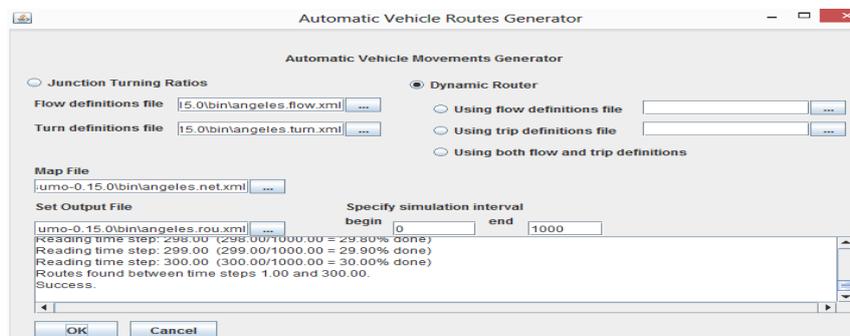
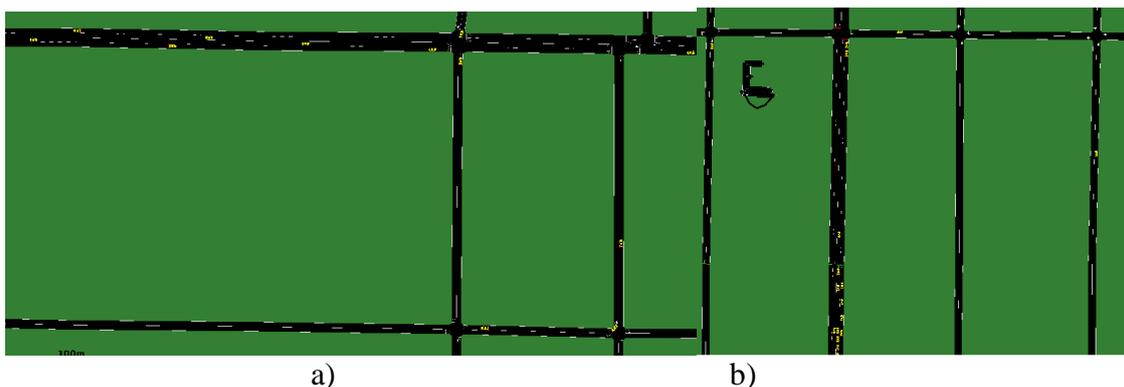


Figura 4.9: Generación de archivo de movilidad del escenario de la ciudad de Los Ángeles.

A continuación se procede a simular este archivo de movilidad utilizando la opción *configuration*, que se encuentra en la parte inferior dentro de las opciones de simulación. Aquí se configuran los directorios de los archivos del mapa **.net.xml* y el archivo de movilidad **.rou.xml*, para luego salvar la simulación en un archivo **.sumo.cfg* en nuestro caso *angeles.sumo.cfg*, *roma.sumo.cfg* y *zaragoza.sumo.cfg*.

La Figura 4.10 muestra la simulación centrada en los puntos de salida de los tres escenarios escogidos al final del tiempo de simulación, para estos casos 300 segundos.



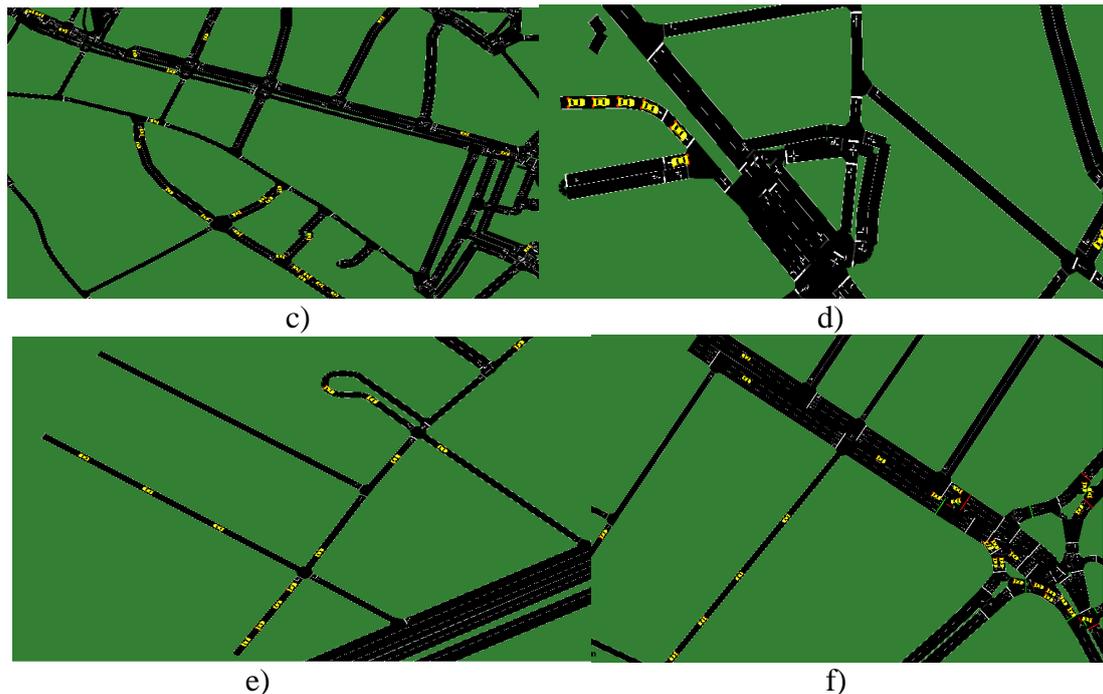


Figura 4.10: Simulación del tráfico vehicular (a-b) Puntos de salidas de Los Ángeles, (c-d) puntos de salida de Roma, (e-f) Puntos de salida de Zaragoza.

4.1.6. Análisis de las simulaciones

Se pueden analizar varios aspectos de las simulaciones realizadas, por ejemplo, en calles de la ciudad de Los Ángeles que son más anchas, largas y con menor número de intersecciones que las de Roma que son exactamente lo contrario, los vehículos tienen menos dificultades para llegar al final del recorrido, como se observa en las figuras anteriores la ciudad de Roma tiene un gran número de vehículos en zonas aledañas al origen de la simulación, no siendo así en el caso de Los Ángeles. La ciudad de Zaragoza se encuentra en un término medio en este sentido.

Otro aspecto importante es que SUMO puede sufrir un gran número de atascos en ciudades de tipo Roma, a no ser que se configure un mayor intervalo de salida entre vehículos, debido a que las calles son muy cortas y con gran número de intersecciones con señales de tráfico y donde los conductores deben parar o reducir bastante su velocidad. En ciudades tipo Los Ángeles sucede lo contrario por lo que existen menos congestiones en el tráfico. En ciudades como Zaragoza el comportamiento está en un término medio, en zonas de muchas intersecciones y calles cortas puede adoptar un comportamiento similar al de Roma y en caso contrario al de los Ángeles.

Para la transmisión de datos en estos escenarios se recomienda el estudio de [62], donde se habla del envío de mensajes en los distintos tipos de ciudades. En ciudades irregulares como Roma se dificulta mucho la transmisión de datos, debido a las interferencias de objetos en la vía como edificios, entre otros. Mientras que en ciudades más regulares como Los Ángeles y Manhattan los datos se distribuyen rápidamente y con mayor facilidad a través de la red y es mucho menor el número de vehículos que quedan incomunicados. Además clasifican un gran número de ciudades en los tres grupos desarrollados en este trabajo, y demuestran que los resultados de simulaciones en ciudades del mismo grupo son muy similares.

En cuanto al tiempo que costaría definir cada escenario al máximo nivel de detalle, es evidente que costaría mucho más diseñar ciudades del tipo Roma, por las características comentadas anteriormente.

A modo de resumen en esta sección se ha indagado en el mundo de la simulación del tráfico rodado, concretamente con la herramienta SUMO, enmarcada dentro de las líneas principales que definen a un generador de movilidad de gran potencia y facilidad de uso, que nos brinda muchas posibilidades de recrear escenarios de diversas características, así como un gran número de situaciones y eventos que se observan a menudo en las carreteras y ciudades reales. Esta herramienta tiene mucho que ofrecer, ya que se encuentra en continuo desarrollo y está siendo utilizada en gran cantidad de proyectos a nivel mundial.

Concluyendo esta sección se recomienda SUMO como candidato entre las herramientas de licencia gratuita para seguir siendo explorado y modelar escenarios de distintos índoles que sean de interés dentro del estudio de las redes de vehículos. Además de su integración con varios simuladores de red para obtener resultados bastantes cercanos a la realidad. En la Sección 4.2 se explicará mediante un ejemplo su integración con el simulador de red OMNeT++ mediante el simulador híbrido VEINS.

4.2. Intercambio de datos utilizando un ejemplo predefinido en VEINS

En esta sección se realizarán simulaciones sobre una red de datos, para ello se va utilizar SUMO y el simulador de red OMNeT++. En lugar de crear un modelo real desde el principio se utilizarán los modelos de ejemplo del *framework* VEINS, específicamente el modelo VEINS que se encuentra en la carpeta *mixim/examples*, ya que este modelo se especializa en redes *wireless* y VANETs, además cuenta con un módulo de aplicación que permite interactuar con SUMO mediante mensajes.

Las comunicaciones en ambos simuladores almacenarán todos aquellos comandos que lleguen en cada paso de la simulación mediante ejecución sincronizada. Para cada unidad de tiempo OMNeT++ enviará los comandos almacenados a SUMO y una vez finalizado este paso, SUMO enviará una serie de comandos con la posición de todos los vehículos de vuelta a OMNeT++. Cuando éste reciba la información se realizará el movimiento de los distintos nodos en función de la información de movilidad y una vez procesada se ejecuta el siguiente paso de la simulación. Ante las nuevas condiciones de movilidad los nodos reaccionarán generando nuevos comandos que serán enviados a SUMO y el proceso continúa sucesivamente con la recepción y el tratamiento de dicha información por el simulador de tráfico.

El proceso se lleva a cabo haciendo uso de un protocolo muy sencillo de petición-respuesta mediante una interfaz de control de tráfico TRACI con una arquitectura cliente (SUMO) - servidor (OMNeT++) para ayudar la ejecución de la comunicación. El módulo cliente hace uso de un dominio desarrollado en *Python* denominado *sumo-launchd*, por lo que se debe tener instalado en el ordenador el intérprete de *Python* que se dedica a escuchar los requerimientos del servidor y actúa como proxy entre ambos, con esto se evita lanzar SUMO manualmente antes de cada simulación de OMNeT++,

ya que *sumo-launchd* se encarga del manejo de los puertos y de los archivos temporales simplificando la ejecución de la simulación, creando y eliminando según se necesiten instancias relativas a SUMO. Concretamente comienza una nueva copia de la simulación de SUMO para la conexión requerida por el simulador de red.

Se escogerá el escenario que viene por defecto con el modelo mencionado, no obstante, también se puede utilizar este modelo para un escenario real modificando algunas clases, módulos y archivos *.ned* y apuntando a los directorios de la simulación en SUMO. Debido a que el objetivo de este trabajo es realizar simulaciones sobre las herramientas más destacadas y no centrarnos específicamente en un simulador, para este caso utilizaremos este ejemplo de OMNeT++.

El modelo VEINS, como la mayoría de los modelos de OMNeT++ incluyen dos vistas, la de interfaz gráfica y la del código. Tiene tres archivos principales que definen el escenario, nodos y archivos de inicio:

- *highway.ned*
- *car.ned*
- *omnetpp.ini*

4.2.1. Escenario del modelo VEINS

En la Figura 4.11 se muestra la interfaz gráfica del escenario para el modelo VEINS, que cuenta con cuatro componentes:

1. *world*: define parámetros globales como el área de la red.
2. *connectionManager*: comprueba la conexión entre dos dispositivos activos.
3. *anotations*: Gestiona anotaciones desde un documento *.xml*.
4. *obstacles*: permite definir obstáculos al modelo.

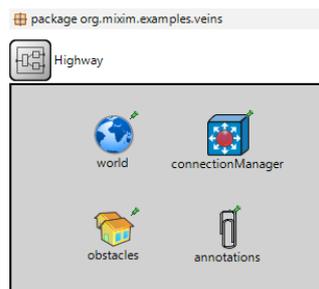


Figura 4.11: Interfaz gráfica del escenario en el modelo VEINS.

4.2.2. Nodos del modelo VEINS

En la Figura 4.12 se muestra la vista gráfica de los nodos en el modelo VEINS que también cuenta con cuatro componentes:

1. *appl*: Se encarga del nivel de aplicación de las comunicaciones entre vehículos, por ejemplo, mensajes.
2. *arp*: Se ocupa de asociar la dirección MAC con la dirección IP.

3. *nic*: Cuenta con dos módulos simples, encargado de las etapas de transmisión y del acceso al medio.
4. *mobility*: Conecta con SUMO para recibir actualizaciones de la posición y estado de los nodos y se encarga de controlar la movilidad.

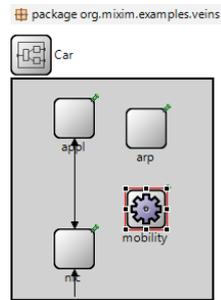


Figura 4.12: Nodos del modelo VEINS.

4.2.3. Archivo de inicio del modelo VEINS

Por último se explicará el archivo de inicio *omnetpp.ini* que se encarga de guardar la información de la simulación como el tiempo, los obstáculos y demás configuraciones de este modelo y de cada módulo que en él se encuentra como se observa en la Figura 4.13.

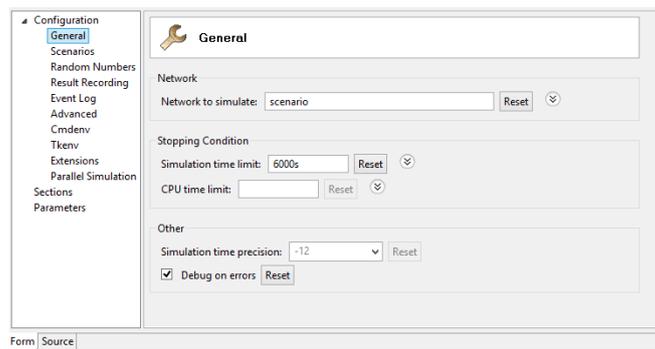


Figura 4.13: Archivo *omnetpp.ini*.

4.2.4. Lanzar la simulación

Se lanza la simulación desde OMNeT++ en el explorador de proyectos de la GUI, luego se hace clic derecho en el archivo *omnetpp.ini* del proyecto VEINS, posteriormente se selecciona *Run As/OMNeT++ Simulation* donde se abren simultáneamente las GUI de OMNeT++ y SUMO y se ejecuta la simulación en ambas.

En el momento que se lanza la simulación en SUMO, que estaba en modo de espera, recibirá los primeros mensajes de sincronización de OMNeT++ a través de TRACI como muestra la Figura 4.14.

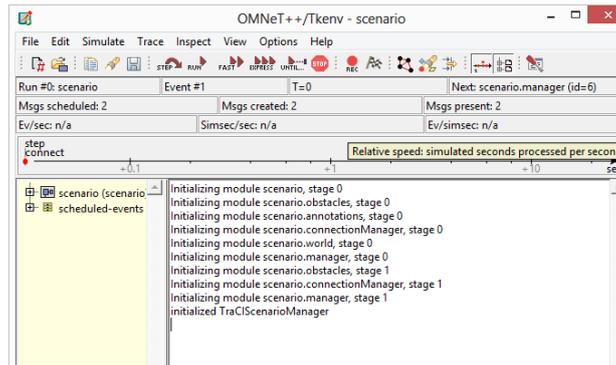
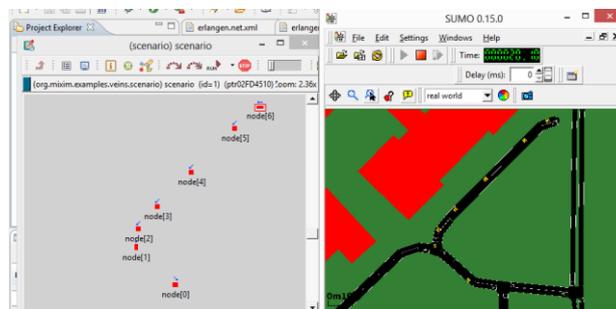
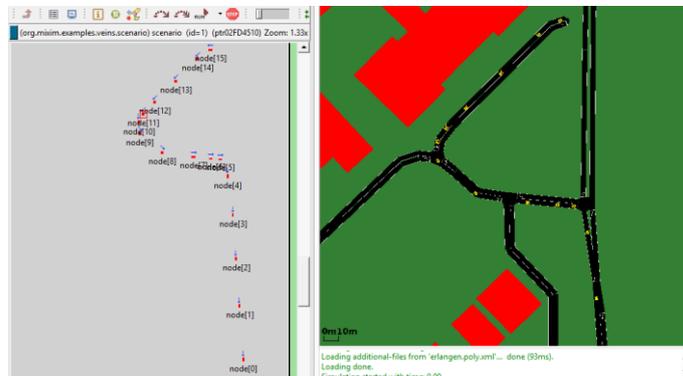


Figura 4.14: Mensajes TRACI.

A continuación se muestra visualmente una ampliación de la simulación en ambas herramientas y cómo va evolucionando con 6 y con 15 nodos.



a)



b)

Figura 4.15: Ampliación de zona de vehículos de SUMO y OMNeT++ (a) 6 nodos, (b) 15 nodos.

En la Figura 4.15 se puede observar que están perfectamente sincronizadas las dos herramientas mediante el envío de mensajes entre ambas y el movimiento de cada uno de los nodos, para una mejor visualización se tomaron fragmentos ampliados, pero este ejemplo cuenta con 195 nodos. Al terminar la simulación podemos analizar los datos concretos que se han cargado de los ficheros de resultados.

4.2.5. Análisis de los resultados

Interpretar los resultados de la simulación se puede tornar un proceso complejo dependiendo de qué variable en concreto se desee estudiar. OMNeT++ muestra los resultados en valores escalares, vectoriales y como histogramas, como se puede observar en la Figura 4.16. En el proyecto se creará un directorio llamado *results* que guarda estos resultados. A partir de estos archivos, se pueden aplicar filtros para extraer la información que resulte relevante según los objetivos trazados. En el caso escalar: distancia recorrida, movilidad, etc. y para los resultados vectoriales: velocidad, aceleración, etc.

Estos datos se guardan en un fichero **.anf* de modo que no hay que ejecutar la simulación siempre que se quieran analizar los resultados. Simplemente indicando al simulador cuál es el fichero se generarán los diagramas o gráficos.

Folder	File name	Config na...	R	Run id	Module	Name	Count
/mixim/examples/veins/results/	nodebug-0.vec	nodebug	0	nodebug-0-20130924-22:18:30-3400	scenario.node[0].mobility	posx	2734
/mixim/examples/veins/results/	nodebug-0.vec	nodebug	0	nodebug-0-20130924-22:18:30-3400	scenario.node[0].mobility	posy	2734
/mixim/examples/veins/results/	nodebug-0.vec	nodebug	0	nodebug-0-20130924-22:18:30-3400	scenario.node[0].mobility	speed	2733
/mixim/examples/veins/results/	nodebug-0.vec	nodebug	0	nodebug-0-20130924-22:18:30-3400	scenario.node[0].mobility	acceleration	2732
/mixim/examples/veins/results/	nodebug-0.vec	nodebug	0	nodebug-0-20130924-22:18:30-3400	scenario.node[0].mobility	co2emission	2732
/mixim/examples/veins/results/	nodebug-0.vec	nodebug	0	nodebug-0-20130924-22:18:30-3400	scenario.node[1].mobility	posx	2753
/mixim/examples/veins/results/	nodebug-0.vec	nodebug	0	nodebug-0-20130924-22:18:30-3400	scenario.node[1].mobility	posy	2753
/mixim/examples/veins/results/	nodebug-0.vec	nodebug	0	nodebug-0-20130924-22:18:30-3400	scenario.node[1].mobility	speed	2752
/mixim/examples/veins/results/	nodebug-0.vec	nodebug	0	nodebug-0-20130924-22:18:30-3400	scenario.node[1].mobility	acceleration	2751

Figura 4.16: Opción *Browse Data*, donde se observan los datos vectoriales.

Primero se mostrarán algunos ejemplos de gráficos sobre los resultados vectoriales como el análisis de la aceleración, utilizando como referencia los dos primeros nodos identificados con distintos colores, como muestra la leyenda, para este ejemplo el resto de nodos poseen una aceleración constante exactamente igual que los dos primeros. Ver la Figura 4.17.

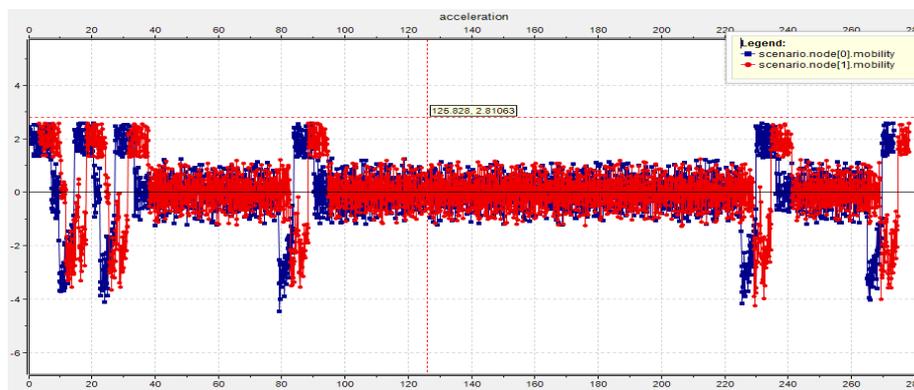


Figura 4.17: Análisis de la aceleración de los nodos 1 y 2.

La Figura 4.18 muestra la velocidad del nodo 0 en todo el trayecto de la simulación. Se puede observar un comportamiento normal aumentando la velocidad hasta alcanzar un

máximo que se mantiene constante hasta llegar a algún cruce, donde reduce la velocidad con paradas y así sucesivamente hasta llegar al destino final.

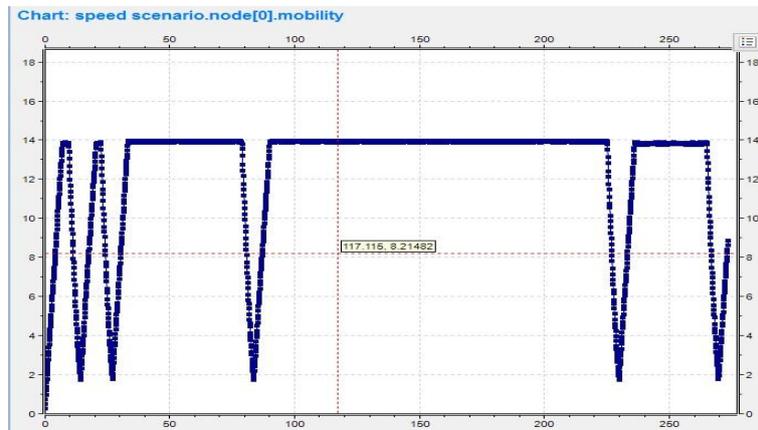


Figura 4.18: Análisis de la velocidad del nodo 0.

Otros aspectos interesantes para el análisis son los resultados escalares, a continuación sigue un ejemplo gráfico de la distancia recorrida en metros (*m*) por cada nodo en un tiempo determinado de la simulación, para este caso se puede observar que todos los nodos siguen el mismo recorrido. En la Figura 4.19 se observa la distancia recorrida por los 20 primeros nodos en un tiempo de 300 segundos. No obstante, ésta gráfica sería más interesante para redes de vehículos con características caóticas donde cada nodo tiene un movimiento aleatorio.



Figura 4.19: Distancia recorrida por los 20 primeros nodos.

Otro aspecto que se suele tener en cuenta cuando se analiza una red vehicular, es la velocidad máxima desarrollada por cada nodo, en este ejemplo en concreto es constante de 14 m/s para todos los vehículos como se muestra en la Figura 4.20, utilizando de ejemplo los 10 primeros nodos.

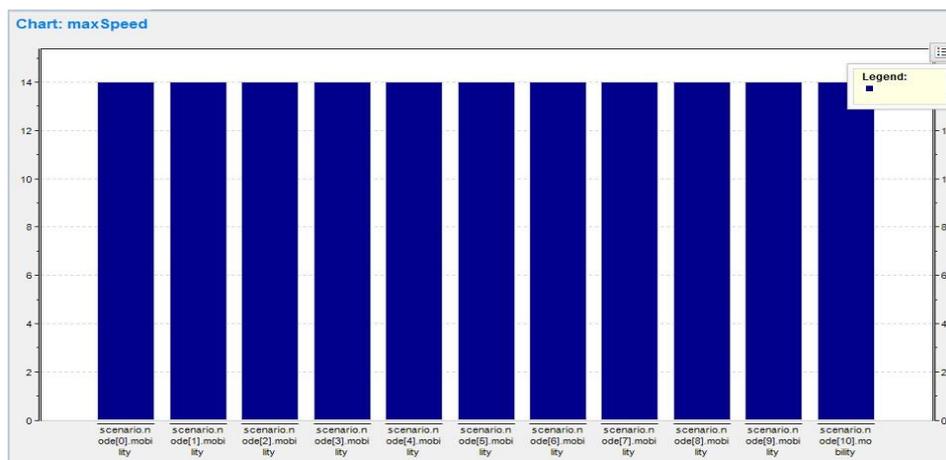


Figura 4.20: Velocidad máxima de los 10 primeros nodos.

Para finalizar, a modo de resumen, se han visto varios aspectos para realizar simulaciones con VEINS: definir el escenario y patrón de movilidad con SUMO, el intercambio de mensajes utilizando OMNeT++ con muchas de sus funcionalidades extras. Luego se muestra cómo mediante el sistema TRACI ambos simuladores se ejecutan en paralelo y bidireccionalmente acoplados. También se ha dejado un abanico amplio de opciones para que los lectores puedan crear sus propios proyectos en entornos reales, uno de los objetivos de este trabajo, donde se han presentado ejemplos de creación, ejecución y análisis de las simulaciones de redes de vehículos.

4.3. Simulando con otras herramientas

En esta sección se muestran simulaciones realizadas con versiones de pruebas de herramientas de grandes prestaciones como AIMSUN y VISSIM. Se realizaron pequeñas simulaciones de escenarios sencillos con el fin de explotar sus ventajas de modelado y simulación de redes VANETs. Sin embargo, nos ha sido imposible ir más allá debido al alto coste de sus licencias y las grandes limitaciones de sus versiones gratuitas.

4.3.1. Escenario con rotonda en AIMSUN

Se llevará a cabo una simulación de tráfico centrada en la intercepción de tipo rotonda y sus zonas aledañas, que aparece en el fragmento seleccionado de la ciudad de Zaragoza, con el objetivo de realizar simulaciones con AIMSUN para comprobar sus características de micro-simulación, muy aceptadas por la comunidad, así como su rendimiento en un escenario real. Este simulador a pesar de ser de pago en teoría presenta muy buenas prestaciones, por lo que se ha decidido simular utilizando una versión limitada de 30 días.

En primer lugar se ha obtenido un mapa real de la zona de simulación utilizando la herramienta OpenStreetMap, para luego importarlo en el simulador AIMSUN representado en la Figura 4.21, además del mapa importado nos muestra una vista total de la herramienta donde se puede apreciar en la columna derecha que tiene en cuenta muchos aspectos importantes para la micro-simulación y que permite definir propiedades particulares a cada escenario.

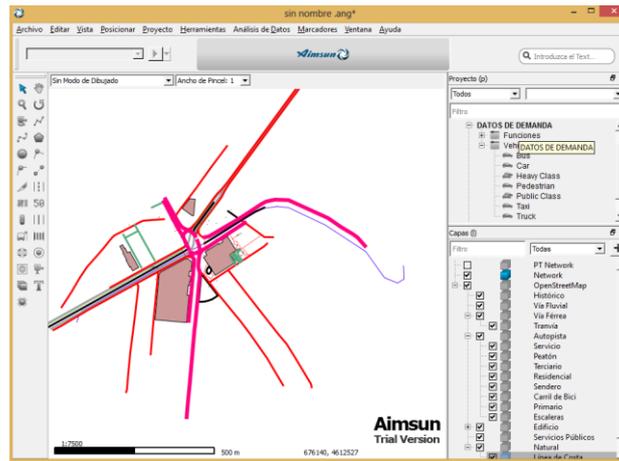


Figura 4.21: Importación de mapa en AIMSUN versión limitada.

Una vez que tenemos el mapa se comienza a construir la red vial centrada en la rotonda que se encuentra en el centro del mapa, creando elementos como secciones, intersecciones (con o sin semáforos), centroides, vehículos, líneas, entre otros. Para nuestro ejemplo se definen 11 secciones, el número de carriles de cada una de ellas y una rotonda que las une como se muestra en la Figura 4.22.

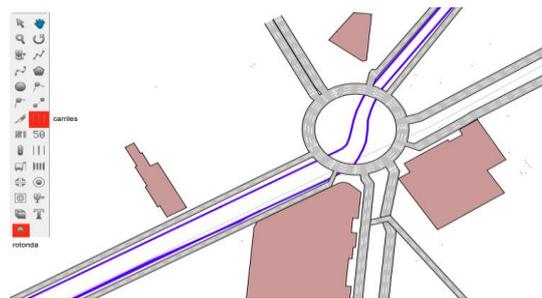


Figura 4.22: Creación de elementos de tráfico.

Después de construir la infraestructura se procede a introducir los parámetros que definirán la circulación en las vías. El simulador AIMSUN permite realizar esta tarea de dos formas diferentes en función de los datos que se dispongan. La primera es mediante una Matriz O/D que indica el número de vehículos que viajan entre los centroides, es decir, los puntos de entrada y salida del sistema y la segunda, es mediante los flujos de vehículos conocido como tabla de estados de tráfico con datos de flujo de entrada de vehículos al sistema y porcentajes de giro en cada uno de los cruces.

Una vez calibrada la Matriz O/D podemos realizar diversas pruebas: cambios de tiempos o estructuras semafóricas, reordenación del tráfico, controles de acceso, pruebas con paneles de señalización variable, estrategias, condiciones del tráfico y problemas que pudieran surgir como dobles filas, cierres de carril, etc.

Después de estudiar de forma práctica el funcionamiento de la herramienta AIMSUN para modelado de escenarios reales a nivel microscópico, considerado como uno de los mejores a nivel mundial en este aspecto debido a la cantidad de parámetros que logra configurar, se ha decidido no llevar a cabo la parte de creación de flujo de tráfico y

movilidad y la de comunicación debido a que la versión gratuita no nos permite estas opciones.

4.3.2. Simulando un cruce peatonal con VISSIM

Para mostrar el funcionamiento del simulador de tráfico VISSIM se realiza una simulación de un escenario con un cruce peatonal sencillo que pasa por dos carriles en sentidos opuestos y luego se mostrará un ejemplo más complejo que viene por defecto con VISSIM.

La versión que se utilizará es la *PVT-VISSIM 6.00-04 Demo*, esta versión es completamente funcional pero no permite guardar las simulaciones por lo que se hace muy difícil trabajar en proyectos grandes. En la versión 6 de VISSIM se han realizado cambios notables con respecto a versiones anteriores, desde el modelado de la red hasta los formatos en que se guardan las simulaciones. Estos aspectos se irán mencionando a lo largo de esta sección.

En la Figura 4.23 se muestra la nueva interfaz gráfica de VISSIM en su versión limitada. Aquí se puede apreciar a la izquierda el menú de creación de objetos, en el centro el área de construcción de la red y debajo las listas (elementos que nos ayudarán en la simulación).

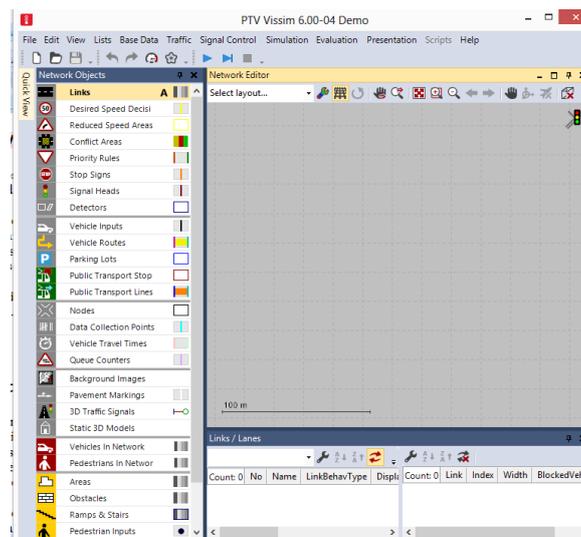


Figura 4.23: Área de trabajo de VISSIM.

Para crear un nuevo escenario lo primero que se añadirán son las líneas o calles, para ellos seleccionamos la opción *links* en el menú de la izquierda. Al crear la línea aparece una nueva ventana para definir sus características como el número de carriles, ancho de carriles, tipos de elementos que circulan, etc. Véase la Figura 4.24. En nuestro ejemplo configuramos dos líneas verticales de un solo carril en sentidos opuestos para los vehículos y una línea horizontal de doble sentido para el paso de peatones.

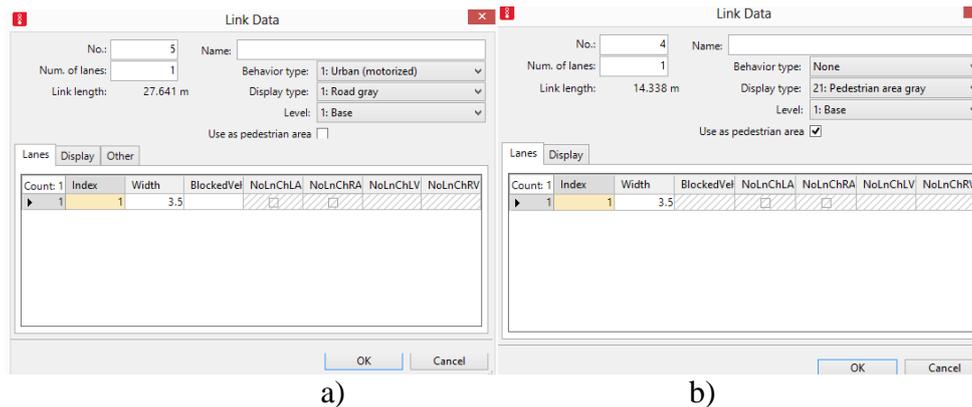


Figura 4.24: Configuración de líneas a) vehículos b) peatones.

Para crear una señal de tráfico de tipo semáforo se selecciona la opción *signal heads* y se crea dentro del carril la señal, aparecerá una ventana de configuración del semáforo donde se puede definir grupos de semáforos, tipo de semáforo, para qué elementos debe actuar, la frecuencia de cambio, etc. Para nuestro ejemplo se crearán dos grupos a) para todos los vehículos y b) para los peatones, cada grupo de señales contiene dos semáforos. Véase la Figura 4.25.

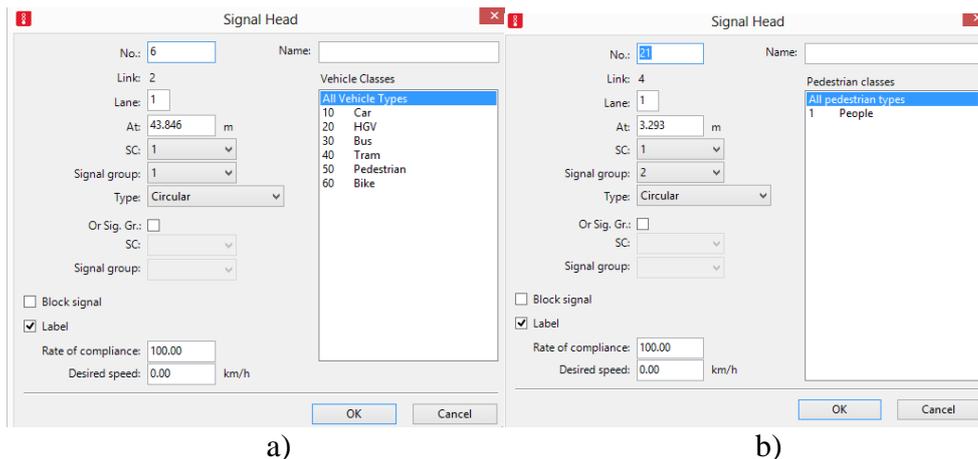
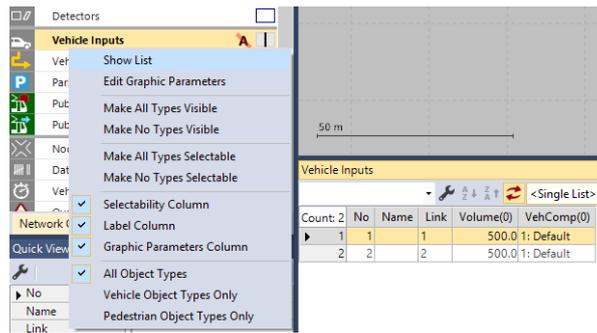


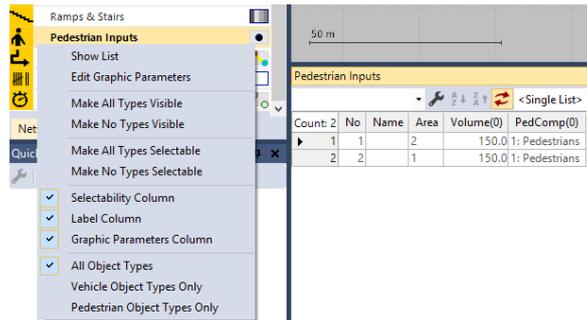
Figura 4.25: Configuración de semáforos a) grupo 1: vehículos b) grupo 2: peatones.

La creación del tráfico vehicular se ejecuta en la opción *vehicle inputs* con botón derecho y luego en la opción *show list*, donde se muestran los parámetros de entrada de vehículos a la red.

La entrada de peatones se ejecuta de la misma forma pero en la opción *pedestrian inputs*. La configuración de este ejemplo se muestra en la Figura 4.26.



a)



b)

Figura 4.26: a) Entrada de vehículos b) Entrada de peatones.

Para el paso peatonal incluimos dos detectores, en la opción *detectors*. Al utilizar los detectores se coloca una barra de parada para los peatones y se define la longitud, posición, tipo de detector, etc. En la Figura 4.27 se puede observar la configuración para nuestro ejemplo.

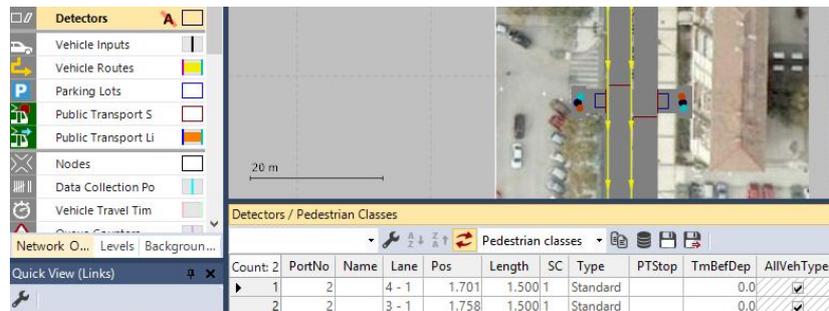


Figura 4.27: Detectores para el cruce de peatones.

También incluimos una imagen de fondo de un mapa real con las características de nuestro ejemplo mediante la opción *background imag*, luego en las opciones del menú de contexto se puede definir la escala. En la Figura 4.28 se observa a) la imagen de fondo y b) la imagen integrada al escenario.

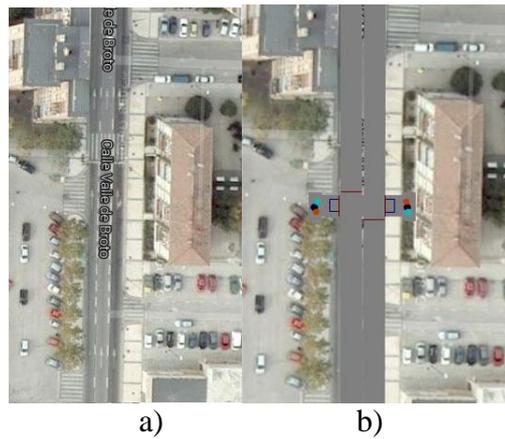


Figura 4.28: a) Imagen de fondo de escenario real b) imagen integrada.

A continuación mostramos una captura de imagen de la simulación donde se muestra a) un instante de tiempo donde están circulando los vehículos y b) momento en que circulan los peatones. Véase la Figura 4.29.

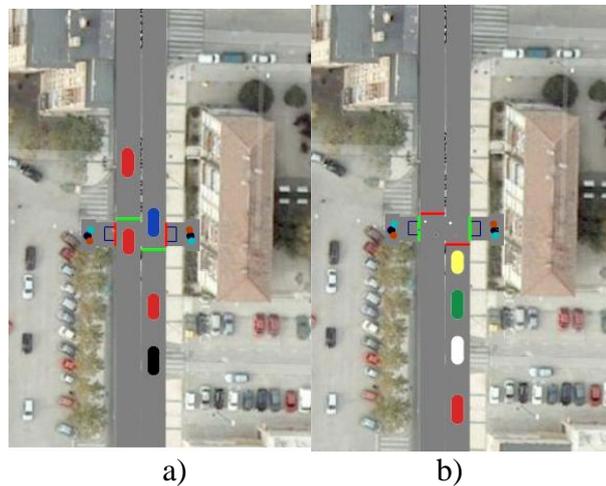


Figura 4.29: a) Circulación de vehículos b) Circulación de peatones.

Para finalizar la Figura 4.30 muestra un ejemplo que viene por defecto con la versión de pruebas de VISSIM, que representa un escenario algo más complejo, con intersecciones, líneas con varios carriles, varios tipos de vehículos, etc.



Figura 4.30: Ejemplo de simulación con VISSIM.

Después de realizar estas prácticas de simulación con VISSIM podemos afirmar que es una herramienta muy competente para el modelado de escenarios reales. Al igual que AINSUM es una herramienta de pago lo que limita bastante su utilización en proyectos de investigación. En esta memoria sólo se muestra un ejemplo sencillo debido a que no permite guardar escenarios en su versión limitada a pesar de ser completamente funcional, lo que supone realizar la simulación sin poder cerrar la herramienta.

4.4. Conclusiones generales de las simulaciones

Todos los simuladores utilizados en este capítulo, mantuvieron un buen rendimiento en los escenarios escogidos. En cuanto a facilidad de uso SUMO ha destacado sobre los demás. A continuación se detallan algunas ventajas y desventajas obtenidas de las simulaciones.

Para la simulación de elementos AIMSUN y VISSIM toman una ligera ventaja sobre SUMO, ya que tienen en cuenta más elementos para ejecutar la simulación, por ejemplo, tranvías, peatones, etc. Además de realizar un mejor modelado de intersecciones complejas y rotondas. No obstante, con SUMO se pueden definir escenarios en menor tiempo debido a que importa el mapa con los carriles definidos y se configura únicamente el modelo de movilidad, mientras que en AIMSUN y VISSIM hay que crear las vías, las intersecciones, señales y cada elemento que se desee simular. Además la versión gratuita de AIMSUN es muy limitada.

Las simulaciones utilizando VISSIM muestran que es un simulador de tráfico muy potente, que permite configurar los más mínimos detalles de un escenario real. Sin embargo, posee una curva de aprendizaje lenta, toma bastante tiempo modelar un escenario al detalle y es una herramienta comercial donde la versión de pruebas no permite guardar proyectos.

Las simulaciones de red con OMNET++ resultaron satisfactorias sobre todo destacar su integración bidireccional con SUMO mediante VEINS. No obstante, es una herramienta complicada de utilizar debido al gran número de elementos que pueden configurarse.

En definitiva, consideramos que a día de hoy VEINS es una de las mejores opciones no comerciales para realizar simulaciones en una VANET. Como nota adicional mencionar que existen numerosas posibilidades de cómputo y escenarios de simulación que nos ofrece una herramienta tan completa como VEINS, con lo cual se recomienda seguir realizando pruebas en diversos escenarios y elementos reales en la carretera.

Apuntar que al realizar pruebas con VEINS utilizando escenarios reales: Roma, Los Ángeles y Zaragoza, produjo errores cerrando las interfaces gráficas de SUMO y OPNET++ debido al tamaño y complejidad de los escenarios y a las bajas características del portátil en el que se realizaron las simulaciones.

CAPÍTULO 5: CONCLUSIONES Y TRABAJO FUTURO

La motivación de este trabajo nos conduce a conocer las redes VANETs y demostrar que las simulaciones juegan un papel importante en las mismas, así como poner a disposición de la comunidad científica un estudio comparativo teórico-práctico de las herramientas más utilizadas en la simulación de redes de vehículos.

Una vez finalizada la investigación se presentan las conclusiones obtenidas y las líneas de trabajo que se han abierto a partir de su desarrollo.

5.1. Conclusiones generales

En este trabajo se ha presentado un acercamiento al mundo de las simulaciones de redes de vehículos. Además, se han valorado varios de los simuladores existentes comparando sus principales características. Finalmente, se realizan simulaciones prácticas en escenarios reales con los simuladores más destacados de nuestro estudio.

En definitiva, se han cumplido los objetivos propuestos y se publica un estudio completo de los simuladores más utilizados en la actualidad, lo que hace más sencillo el proceso de seleccionar adecuadamente una herramienta para realizar simulaciones de redes vehiculares. A continuación se presentan algunos puntos a tener en cuenta de esta tesis de fin de máster.

Sobre las redes de vehículos:

- Estas redes son un factor imprescindible para mejorar la seguridad, el entretenimiento y la información en las carreteras, debido a que permiten el intercambio de información de forma inalámbrica en tiempo real. Pueden ofrecer servicios como información sobre accidentes, congestión de tráfico, eventos de masas, rutas alternativas en caso de algún problema en la vía, inclemencias del tiempo, entre otros. Además, los conductores pueden tomar decisiones en tiempo real de acuerdo con la información recolectada y provista por los nodos de la red.
- El estándar 802.11p mejora el desempeño de las redes vehiculares y presenta las características necesarias para el acceso inalámbrico vehicular de forma segura en un rango de aproximadamente 350 metros.
- La importancia de estas redes aumentará significativamente en los próximos años, debido a que ofrecen grandes ventajas y que los vehículos estarán equipados con comunicación V2V.

En cuanto a los simuladores estudiados:

- La utilización de escenarios reales y modelos de movilidad le aporta un gran nivel de realismo a la simulación. No obstante, los resultados a día de hoy se pueden mejorar, debido a que todavía simuladores gratuitos (SUMO, VEINS,

etc.), no incluyen obstáculos en la carretera, que interfieren en las comunicaciones reales, en el caso de VEINS está en proceso de implementación.

- Muchos simuladores analizados en este trabajo, incluso los que se proponen para realizar simulaciones VANETs, se pueden instalar en las plataformas más comunes y utilizadas en la actualidad como *Windows* y *Linux*, lo que brindan una mejor adaptación a los usuarios y demuestra que son herramientas con el soporte necesario para ser utilizadas en la actualidad.
- La mayoría de los simuladores en la actualidad utilizan interfaces gráficas GUI, lo que es una gran ventaja con respecto a la utilización de líneas de comandos y permiten configurar una red vehicular fácilmente, por ejemplo: MOVE y GNED.
- La modelación de tráfico en AIMSUN y VISSIM para simular redes vehiculares es un proceso muy similar al escenario real en nuestras calles y carreteras, debido a que introduciendo los datos reales de la vía se obtienen simulaciones con resultados exactos de entornos reales.
- El principal aporte de VEINS es el acoplamiento bidireccional entre el buen modelo de movilidad que brinda una herramienta como SUMO y la óptima simulación del canal inalámbrico de OMNET++, brindando una mejora notable sobre herramientas desacopladas.
- Se recomienda a los lectores el uso de las versiones utilizadas en esta memoria que hasta la fecha son las últimas desarrolladas para cada simulador. Sin embargo, las herramientas de código abierto suelen evolucionar muy rápido, debido al gran número de desarrolladores que trabajan en la implementación de mejoras y nuevas funcionalidades, por lo que en un futuro se deben analizar nuevas versiones.

Finalmente sobre las simulaciones:

- La velocidad que alcanza un vehículo depende del número de intersecciones que existan en el escenario simulado debido a que estos obligan a reducir la velocidad, impidiendo que aumente indefinidamente.
- Que un vehículo reciba un gran número de mensajes implica que su trayecto puede ser una ruta crítica dentro de la red simulada.
- Cuanto mayor sea el número de vehículos de la red más amplia es la cobertura de la misma. No obstante, también el canal está más ocupado lo que puede llevar a perder información, con lo cual, es necesario que ambos aspectos estén proporcionalmente relacionados. El desempeño de la red disminuirá acorde a la cantidad de nodos que intervengan por lo que se debe implementar un protocolo de enrutamiento que permita una mejor fiabilidad en el sistema.

A modo de resumen este trabajo ha establecido las bases para el estudio de las simulaciones de redes VANETs. Sin embargo, no se trata de un trabajo cerrado, sino

que está abierto a nuevas líneas de trabajo futuro para realizar pruebas más complejas y consistentes sobre nuestra propuesta.

5.2. Valoraciones personales

Primero decir que al ser mi primera incursión en la investigación científica veía muy complicado conseguir los objetivos propuestos, debido a que tenía que investigar sobre dos grandes campos, por un lado las redes vehiculares y por otro el funcionamiento de las simulaciones, lo que muchas veces no se hace fácil debido a que son temáticas diferentes y muy amplias, donde existe una gran cantidad de artículos y proyectos de investigación.

Además, me ha resultado difícil encontrar ayudas y tutoriales de algunos simuladores, con lo cual, me he visto obligado algunas veces a realizar simulaciones probando un gran número de elementos del simulador para entender sus finalidades. También en ocasiones el portátil utilizado generó fallos en su rendimiento al intentar realizar pruebas más complejas.

Otro aspecto que me supuso algunos problemas fueron las versiones de pago de varias herramientas, por lo que tenía que probar versiones gratuitas que casi siempre eran muy limitadas y prácticamente sin documentación disponible.

Sin embargo, cuando me fui adentrando en el tema de las simulaciones de redes vehiculares me encontré con una grata experiencia personal, resolviendo problemas que surgían y sobre todo al realizar simulaciones prácticas en las herramientas e ir consiguiendo resultados satisfactorios.

Una vez finalizado el trabajo y haber cumplido sus objetivos resulta gratificante observar los resultados obtenidos, tanto en la parte técnica y práctica, como en el estudio de otros trabajos interesantes sobre el tema.

En fin me quedo con los conocimientos aprendidos y la experiencia obtenida a lo largo de este trabajo que sin dudas, me ha aportado mucho en lo personal y profesional.

5.3. Trabajo futuro

El desarrollo de esta tesis de fin de máster ha abierto varias líneas de investigación, algunas de ellas serán abordadas por el autor de este trabajo como próximos pasos a seguir en la investigación. A continuación se detallan estas líneas.

- Estudiar en profundidad simuladores como AIMSUN, iTETRIS y NCTUNS, con el objetivo de realizar comparaciones entre resultados prácticos de simulaciones complejas en determinados escenarios. Además destacar ventajas y desventajas de cada uno de ellos, lo que podría ayudar a mejorar sus funcionalidades.
- Investigar y realizar pruebas sobre simuladores que permitan incluir obstáculos en las simulaciones: edificios, construcciones, etc., ofreciendo un mayor nivel de realismo a las comunicaciones dentro de un escenario real.

- Para futuros trabajos se recomienda tomar el tiempo necesario para implementar cada elemento del escenario al detalle antes de ejecutar la simulación, de lo contrario se pueden obtener resultados muy variables en las simulaciones.
- Trabajar en base a desarrollar un instalador unificado para algunos simuladores como VEINS, ya que el proceso de instalación de tres simuladores (SUMO, OPNET++, VEINS), además de las herramientas integradas como MOVE por separado suele tornarse algo complejo.
- Realizar simulaciones utilizando ordenadores independientes con diferentes simuladores instalados e intentar aprovechar la computación distribuida, con el fin de implementar una simulación compleja introduciendo trazas reales para aumentar el realismo.
- Al realizar simulaciones con la solución SUMO, OPNET, VEINS siempre que se ejecuta una simulación diferente es necesario volver a cargar nuevos escenarios y configuraciones en los simuladores. Para facilitar este proceso se podría implementar un sistema que permita lanzar varias simulaciones a la vez con una sola llamada.
- Intentar seguir insertando el tema de redes vehiculares en las universidades mediante actividades como talleres de motivación, asignaturas de posgrado, seminarios, etc., para que los estudiantes puedan familiarizarse con el tema que poco a poco se va convirtiendo en una necesidad para la mejora de la seguridad ciudadana.

Por último, junto con mi director de TFM, vamos a preparar un artículo de investigación con los resultados obtenidos en esta investigación para ser enviado a alguna de las principales revistas del tema, poniendo así a disposición de la comunidad científica un estudio completo y actual de las simulaciones en redes de vehículos (Título provisional: “*A Study on Simulation for Vehicular Networks*”).

Bibliografía

- [1] A. L. Sandoval Orozco, J. García Matesanz, L. J. García Villalba, J. D. Márquez Díaz and T.-H. Kim. “Security Issues in Mobile Ad Hoc Networks”. *International Journal of Distributed Sensor Networks*. 2012. Vol. 2012. pp 1-6.
- [2] Peng Li. “Study on the Wireless Ad Hoc Network Secure Routing Based on the SAR Technology”. *Computer and Information Science*. 2011. Vol.4, No.3. pp 151-156.
- [3] Wenjia Li and Anupam Joshi. “Security Issues in Mobile Ad hoc networks (A Survey)”. *The 17 th White House Papers Graduate Research In Informatics at Sussex*. 2004. pp 1-23.
- [4] María Canales, José Ramón Gállego, Ángela Hernández-Solana, Antonio Valdovinos. “QoS provision in mobile ad hoc networks with an adaptive cross-layer architecture”. *Wireless Networks*. 2009. Vol.15, No.8. pp 1165-1187.
- [5] Ash M, Oivind K. “Quality of Service in mobile ad hoc networks: a survey”. *Journal of Ad Hoc and Ubiquitous Computing*. 2010. Vol.6, No.2. pp 75-98.
- [6] Arzoo Dahiya , Dr.R.K.Chauhan. “A Comparative study of MANET and VANET Environment”. *Journal of computing*. 2010. Vol. 2. pp 1-3.
- [7] IEEE 802.11, “IEEE Standard for Information technology-Telecommunications and information exchange between systems Local and metropolitan area networks-Specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications”. 2012. pp 1-2793.
- [8] Baldessari, Roberto, Bödecker, Bert, et al. “CAR 2 CAR Communication Consortium – Manifesto”. *Overview of the C2C-CC System*. 2007. pp 1-94.
- [9] Eichler, S., Schroth, C., and Eberspächer, J. “Car-to-Car Communication”. *In Proceedings of the VDE-Kongress - Innovations for Europe*. 2006. pp 1-6.
- [10] R. Bogenberger, W. Kellerer, T. Kosch, T. Reicher, C. Schwingenschlögl, P. Sties, M. Wagner. “Virtual City Portal – A Multi-Network Personal Information System for Automobile Users”. *In proc. Multiradio Multimedia Communications - Communication Technology for Vehicles*. 2003. pp 1-9.
- [11] Michael Shulman and Richard Deering. “Vehicle Safety Communications in the United States”. *Technical report 07-0010, Ford Motor Company and General Motors Corporation*. 2007. pp 1-12.
- [12] GST (*System for Telematics*). Disponible en: <http://www.ertico.com/gst-website>. Última consulta: 13/06/2013.
- [13] CVIS (*Cooperative Vehicle Infrastructure System*). Disponible en: <http://www.cvisproject.org>. Última consulta: 21/08/2013.

- [14] Car 2 Car Communication Consortium. Disponible en: <http://www.car-to-car.org/>. Última consulta: 05/08/2013.
- [15] Serge P. Hoogendoorn, Piet H. L. Bovy. "State-of-the-art of Vehicular Traffic Flow Modelling". *Systems and Control Engineering*. 2001. Vol. 215, No. 4, pp. 1-46.
- [16] J. Harri, F. Filali, C. Bonnet. "Mobility models for vehicular ad hoc networks: a survey and taxonomy. *Communications Surveys & Tutorials, IEEE*. 2009. Vol.11, No.4. pp 19-41.
- [17] B. Ramakrishnan. "Analysis of Manhattan mobility model without RSUs". *IOSR Journal of Computer Engineering (IOSRJCE)*. 2013. Vol. 9, No. 5. pp 82-90.
- [18] E. Casilari, y A. Triviño "Análisis de la estabilidad de modelos de movilidad en simulaciones de redes ad hoc". *JITEL '05*. 2005. pp. 33-40.
- [19] A. Jardosh, E.M. Belding-Royer, K.C. Almeroth, and S. Suri, "Towards realistic mobility models for mobile Ad hoc networks", *9th Intl. Conf. on Mobile Computing and Networking*. 2003. pp 217-229.
- [20] Realistic Vehicular Traces. Disponible en: <http://www.lst.inf.ethz.ch/research/ad-hoc/car-traces/> Última consulta: 03/09/2013.
- [21] MIT Human Dynamics Laboratory. Disponible en: <http://hd.media.mit.edu/> Última consulta: 03/09/2013.
- [22] N. Eagle, A. Pentland. "Reality mining: sensing complex social systems". *Journal Personal and Ubiquitous Computing*. 2006. Vol. 10, No. 4. pp 255-268.
- [23] TIGER databes. Disponible en: <http://www.census.gov/geo/www/tiger/>. Última consulta: 09/10/2013.
- [24] OpenStreetMap: The FreeWikiWorld Map, (2008). Disponible en: <http://www.openstreetmap.org/>. Última consulta: 09/10/2013.
- [25] INTEGRATION. Release 2.30 for Windows. "User's Guide - Volume I. Fundamental Model Features". 2010. pp 1-63.
- [26] PATH (*Partners for Advanced Transportation Technology*). Disponible en: <http://www.path.berkeley.edu/>. Última consulta: 10/07/2013.
- [27] VISSIM. Disponible en: <http://vision-traffic.ptvgroup.com/es/productos/ptv-vissim/>. Última consulta: 10/10/2013.
- [28] STRAW (*Street Random Waypoint*) Disponible en: <http://www.aqualab.cs.northwestern.edu/projects/STRAW/index.php>. Última consulta: 10/06/2013.
- [29] VanetMobiSim. Disponible en: <http://vanet.eurecom.fr/>. Última consulta: 01/09/2013.

- [30] CORSIM. Disponible en: <http://mctrans.ce.ufl.edu/featured/tsis/Version6/corsim.htm>. Última consulta: 02/09/2013.
- [31] MITSIM. Disponible en: <http://its.mit.edu/software/mitsimlab>. Última consulta: 02/09/2013.
- [32] Ben-Akiva, M., A. Davol, T. Toledo, H. N. Koutsopoulos, W. Burghout, I. Andréasson, T. Johansson, C. Lundin. "MITSIMLab for Stockholm: Enhancements, Calibration and Validation". *GFK Research Report Stockholm*. 2000. pp 5-64.
- [33] Krajzewicz, Daniel und Erdmann, Jakob und Behrisch, Michael und Bieker, Laura. "Recent Development and Applications of SUMO". *International Journal On Advances in Systems and Measurements*. 2012. Vol. 5, No. 3 y 4. pp 128-138.
- [34] SUMO Disponible en: <http://sumo-sim.org/>. Última consulta: 25/10/2013.
- [35] Feliz Kristianto F., Hai Z., Lan K. "Rapid Generation of Realistic Mobility Models for VANET". *Wireless Communications and Networking Conference*. 2007. pp 1-2.
- [36] Grupo de Redes de Computadores. Universidad Politécnica de Valencia. Disponible en: <http://www.grc.upv.es/Software/citymob.html>. Última consulta: 25/01/2013.
- [37] MinnesotaTG: Web-based U.S. Road Traffic Generator. Disponible en: <http://mntg.cs.umn.edu>. Última consulta: 21/05/2013.
- [38] Berlin Mod Disponible en: <http://dna.fernuni-hagen.de/secondo/BerlinMOD/BerlinMOD.html>. Última consulta: 21/05/2013.
- [39] Thomas Brinkhoff: Network-based Generator of Moving Objects. Disponible en: <http://iapg.jade-hs.de/personen/brinkhoff/generator/>. Última consulta: 3/08/2013.
- [40] NS-2 (*The Network Simulator*). Disponible en: <http://www.isi.edu/nsnam/ns/>. Última consulta: 14/10/2013.
- [41] REAL 5.0 Overview. S. Keshav Cornell University. Disponible en: <http://www.cs.cornell.edu/skeshav/real/overview.html>. Última consulta: 08/10/2013.
- [42] NS-3. Disponible en: <http://www.nsnam.org/>. Última consulta: 27/09/2013.
- [43] OMNeT++. Disponible en: <http://omnetpp.org/>. Última consulta: 14/10/2013.
- [44] MiXiM project. Disponible en: <http://mixim.sourceforge.net/>. Última consulta: 14/10/2013.
- [45] QualNet SCALABLE Network Technologies. Disponible en: <http://web.scalable-networks.com/content/qualnet>. Última consulta: 05/07/2013.
- [46] JiST/SWANS (*Java in Simulation Time/Scalable Wireless Ad hoc Network Simulator*). Disponible en: <http://jist.ece.cornell.edu/>. Última consulta: 11/07/2013.

- [47] Jorge Nuevo. “A comprehensible GloMoSim tutorial”. *INRS Université do Québec*. 2004. pp 1-33.
- [48] GTNetS (*The Georgia Tech Network Simulator*-), 2008. Disponible en: <http://www.ece.gatech.edu/research/labs/MANIACS/GTNetS/>. Última consulta: 15/07/2013.
- [49] TraNS Disponible en: <http://lca.epfl.ch/projects/trans/>. Última consulta: 14/10/2013.
- [50] Piorkowski M, Raya M, Lezama A, Papadimitratos P, Grossglauser M, and Pierre J. “TraNS: realistic joint traffic and network simulator for VANETs”. *ACM -Mobile Computing and Communications Review*. 2008. Vol. 12, No.1. pp 31-33.
- [51] MOBIREAL. Disponible en: <http://www.mobireal.net/>. Última consulta: 25/09/2013.
- [52] VEINS. Disponible en: <http://VEINS.car2x.org/>. Última consulta: 25/10/2013.
- [53] Christoph Sommer, Student Member, IEEE, Reinhard German, and Falko Dressler, Senior Member, IEEE. “Bidirectionally Coupled Network and Road Traffic Simulation for Improved IVC Analysis”. *IEEE Transactions on mobile computing*. 2011. Vol. 10, No. 1. pp. 1-13.
- [54] GrooveNet. Disponible en: <http://mlab.seas.upenn.edu/research/automotive-architecture/groovenet/>. Última consulta: 21/07/2013.
- [55] Mangharam R., Weller D., Rajkumar R., Mudalige P. and Bai F. “GrooveNet: A Hybrid Simulator for Vehicle-to-Vehicle Networks. School of Engineering and Applied Science”. *Real-Time and Embedded Systems Lab*. 2006. pp 1-9.
- [56] Biblioteca QT. Disponible en: [http://es.wikipedia.org/wiki/Qt_\(biblioteca\)](http://es.wikipedia.org/wiki/Qt_(biblioteca)). Última consulta: 2/11/2013.
- [57] AIMSUN. Disponible en: <http://www.aimsun.com/wp/>. Última consulta: 25/10/2013.
- [58] Mangharam R., Weller D., Stancil D., Rajkumar R., Parikh J. “GrooveSim: A Topography-Accurate Simulator for Geographic Routing in Vehicular Networks”. *2nd ACM International Workshop on Vehicular Ad Hoc Networks*. 2005. pp 1-10.
- [59] NCTUns 6.0 Network Simulator and Emulator. Disponible en: <http://nsl10.csie.nctu.edu.tw/>. Última consulta: 27/09/2013.
- [60] S.Y. Wang and Y.M. Huang. “NCTUNS Distributed Network Emulator”. *Internet Journal*. 2012. Vol. 4, No. 2. pp 61-94.
- [61] Rondinone M, Maneros J., Krajzewicz D., Bauza R., Cataldi P., Hrizi F., Gozalvez J., Kumar V., Röckl M., Lin L., Lazaro O., Leguay J., Härrri J., Vaz S., Lopez Y., Sepulcre M., Wetterwald M., Blokpoel R., Cartolano F. “iTETRIS: A modular

simulation platform for the large scale evaluation of cooperative ITS applications”. *Vehicular Technology Conference (VTC 2010-Spring)*. 2010. pp 1-5

[62] Francisco J. Martinez, Chai Keong Toh, Juan-Carlos Cano, Carlos T. Calafate and Pietro Manzoni. “A survey and comparative study of simulators for vehicular ad hoc networks”. *Wireless Communications and Mobile Computing*. 2009. Vol. 11, No. 7. pp 813–828.

[63] Marco Fiore, Jerome Harri, Fethi Filali, and Christian Bonnet. “Vehicular Mobility Simulation for VANETs”. In *Proceedings of the 40th Annual Simulation Symposium*. 2007. pp 301-309.

[64] Eduardo L García, Sergio Ilarri Artigas. “Simulador de VANETs para evaluar técnicas de gestión de datos basadas en agentes móviles”. *Proyecto fin de carrera*. 2010. Disponible en: <http://zaguan.unizar.es/record/5307?ln=es>. Última consulta: 30/10/2013.

[65] Michal Maciejewski. “A comparison of microscopic traffic flow simulation systems for an urban area”. *Transport problems*. 2010. Vol. 5 No. 4. pp 1-8.

[66] Stefan Joerer, Falko Dressler and Christoph Sommer. “Comparing Apples and Oranges Trends in IVC Simulations”. *Vehicular Internetworking*. 2012. pp 27-32.

[67] Gustavo Marfia, Giovanni Pau, Enzo De Sena, et al. “Evaluating Vehicle Network Strategies for Downtown Portland: Opportunistic Infrastructure and the Importance of Realistic Mobility Models”. *Proceedings of the 1st international MobiSys workshop on Mobile opportunistic networking*. 2007. pp 47-51.

[68] Aamir Hassan. “VANET Simulation Master’s Thesis in Electrical Engineering”. *School of Information Science, Computer and Electrical Engineering*. 2009. pp 25-35.

[69] Taufiq Ismail. “Comparision of the two micro-simulation software AIMSUN & SUMO for highway traffic modelling”. *Technology Intelligent Transport System*. 2012. pp 39-58.

[70] Joyoung Lee, Byungkyu Park. “Development and Evaluation of a Cooperative Vehicle Intersection Control Algorithm Under the Connected Vehicles Environment”. *IEEE Transactions on Intelligent Transportation Systems*. 2012. Vol.13, No.1. pp 81-90.

[71] Jalal Al-Obaedi, Saad Yousif. “Microsimulation Model for Motorway Merges With Ramp-Metering Controls”. *IEEE Transactions on Intelligent Transportation Systems*. 2012. Vol.13, No.1. pp 296-306.

[72] Jiancheng Long, Ziyou Gao, Penina Orenstein, Hualing Ren. “Control Strategies for Dispersing Incident-Based Traffic Jams in Two-Way Grid Networks”. *IEEE Transactions on Intelligent Transportation Systems*. 2012. Vol.13, No.2. pp 469-481.

[73] Jeong-Ah Jang, Keechoo Choi, Hanbyeog Cho. “A Fixed Sensor-Based Intersection Collision Warning System in Vulnerable Line-of-Sight and/or Traffic-

Violation-Prone Environment”. *IEEE Transactions on Intelligent Transportation Systems*. 2012. Vol.13, No.4. pp 1880-1890.

[74] Scott Biddlestone, Keith Redmill, Radovan Miucic, Ümit Özgüner. “An Integrated 802.11p WAVE DSRC and Vehicle Traffic Simulator With Experimentally Validated Urban (LOS and NLOS) Propagation Models”. *IEEE Transactions on Intelligent Transportation Systems*. 2012. Vol.13, No.4. pp 1792-1802.

[75] Syed A. Hussain and A. Saeed. “An Analysis of Simulators for Vehicular Ad hoc Networks”. *World Applied Sciences Journal*. 2013. Vol. 23, No. 8. pp 1044-1048.

[76] Qing-Jie Kong, Yanyan Xu, Shu Lin, Ding Wen, Fenghua Zhu, Yuncai Liu. “UTN-Model-Based Traffic Flow Prediction for Parallel-Transportation Management Systems”. *IEEE Transactions on Intelligent Transportation Systems*. 2013. Vol.14, No.3. pp 1541-1547.

[77] Biagio Ciuffo, Carlos Lima Azevedo. “Un enfoque multi-paso para el análisis de sensibilidad global de modelos de simulación de tráfico complejas: Aplicación a la modelo MITSIM”. *Transportation Research Board*. 2012.

[78] Pratap Kumar Sahu, Eric Hsiao-Kuang Wu, Jagruti Sahoo, Mario Gerla. “BAHG: Back-Bone-Assisted Hop Greedy Routing for VANET's City Environments”. *IEEE Transactions on Intelligent Transportation Systems*. 2013. Vol.14, No. 1. pp 199-213.

[79] Jin Zhou, C. L. Philip Chen, Long Chen, Wei Zhao. “A User-Customizable Urban Traffic Information Collection Method Based on Wireless Sensor Networks”. *IEEE Transactions on Intelligent Transportation Systems*. 2013. Vol.14, No.3. pp 1119-1128.

[80] Yan Wang, Jianmin Jiang, Tingting Mu. “Context-Aware and Energy-Driven Route Optimization for Fully Electric Vehicles via Crowdsourcing”. *IEEE Transactions on Intelligent Transportation Systems*. 2013. Vol.14, No.3. pp 1331-1345.

[81] Sandra Céspedes Umaña, Ning Lu, Xuemin Shen: VIP-WAVE. “On the Feasibility of IP Communications in 802.11p Vehicular Networks”. *IEEE Transactions on Intelligent Transportation Systems*. 2013. Vol.14, No. 1. pp 82-97.

[82] Jagruti Sahoo, Eric Hsiao-Kuang Wu, Pratap Kumar Sahu, Mario Gerla. “Congestion-Controlled-Coordinator-Based MAC for Safety-Critical Message Transmission in VANETs”. *IEEE Transactions on Intelligent Transportation Systems*. 2013. Vol.14, No.3. pp 1423-1437.

[83] Yongchang Ma, Mashrur Chowdhury, Adel Sadek, Mansoureh Jeyhani. “Integrated Traffic and Communication Performance Evaluation of an Intelligent Vehicle Infrastructure Integration (VII) System for Online Travel-Time Prediction”. *IEEE Transactions on Intelligent Transportation Systems*. 2012. Vol.13, No.3. pp 1369-1382.

[84] Pedro Fernandes, Urbano Nunes. “Platooning With IVC-Enabled Autonomous Vehicles: Strategies to Mitigate Communication Delays, Improve Safety and Traffic

Flow”. *IEEE Transactions on Intelligent Transportation Systems*. 2012. Vol.13, No.1. pp 91-106.

[85] Michel Ferreira, Pedro M. d'Orey. “On the Impact of Virtual Traffic Lights on Carbon Emissions Mitigation”. *IEEE Transactions on Intelligent Transportation Systems*. 2012. Vol.13, No.1. pp 284-295.

[86] Michele Segata, Falko Dressler, Renato Lo Cigno, Mario Gerla. “ACM MobiCom 2012 Poster: A Simulation Tool for Automated Platooning in Mixed Highway Scenarios”. *Mobile Computing and Communications Review archive*. 2012. Vol. 16, No. 4. pp 46-49.

[87] Michele Rondinone, Julen Maneros, Daniel Krajzewicz, et al. “iTETRIS: A modular simulation platform for the large scale evaluation of cooperative ITS applications”. *Simulation Modelling Practice and Theory*. 2013. Vol. 34. pp 99-125.

[88] Vineet Kumar, Lan Lin, Daniel Krajzewicz, Fatma Hrizi, Oscar Martinez, Javier Gozalvez, Ramon Bauza. “iTETRIS: Adaptation of ITS Technologies for Large Scale Integrated Simulation”. *Vehicular Technology Conference*. 2010. pp 1-5.

[89] Di Wu, Yuan Zhang, Lichun Bao, Amelia C. Regan. “Location-Based Crowdsourcing for Vehicular Communication in Hybrid Networks”. *IEEE Transactions on Intelligent Transportation Systems*. 2013. Vol.14, No. 2. pp 837-846.

[90] Khaled F. Hussain, Essam Radwan, Ghada S. Moussa. “Augmented Reality Experiment: Drivers' Behavior at an Unsignalized Intersection”. *IEEE Transactions on Intelligent Transportation Systems*. 2013. Vol.14, No. 2. pp 608-617.

[91] Andrew Gray, Mohammad Ali, Yiqi Gao, et al. “A Unified Approach to Threat Assessment and Control for Automotive Active Safety”. *IEEE Transactions on Intelligent Transportation Systems*. 2013. Vol.14, No.3. pp 1490-1499.

[92] Han-Shue Tan, Jihua Huang. “Experimental Development of a New Target and Control Driver Steering Model Based on DLC Test Data”. *IEEE Transactions on Intelligent Transportation Systems*. 2012. Vol.13, No.1. pp 375-384.

[93] Vicente Milanés, Jorge Villagra, Jorge Godoy, Javier Simo, Joshué Pérez, Enrique Onieva. “An Intelligent V2I-Based Traffic Management System”. *IEEE Transactions on Intelligent Transportation Systems*. 2012. Vol.13, No.1. pp 49-58.

[94] Manuel Fogué, Martínez Domínguez F, Garrido Picazo, P. “Design and evaluation of a traffic safety system based on vehicular networks for the generation of intelligent vehicles”. *Tesis doctoral Universidad de Zaragoza*. 2012. Cap 6. pp 105-125.