ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA

SIMULACIÓN DE UNA VANET PARA UN SECTOR DEL CENTRO HISTÓRICO DE LA CIUDAD DE QUITO COMO APOYO A LA MOVILIDAD VEHICULAR

PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN ELECTRÓNICA Y TELECOMUNICACIONES

MARCO DANIEL CAIZA CHANGOLUISA
marco.caiza@epn.edu.ec

DIRECTORA: ING. MARTHA CECILIA PAREDES PHD.
cecilia.paredes @epn.edu.ec

CODIRECTOR: ING. CHRISTIAN TIPANTUÑA MSC.
christian.tipantuna @epn.edu.ec

Quito, marzo 2017

DECLARACIÓN

Yo, Marco Daniel Caiza Changoluisa, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentada para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedo mis derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

_____

Marco Daniel Caiza Changoluisa

CERTIFICACIÓN

Certificamos que el presente trabajo fue desarrollado por Marco Daniel Caiza Changoluisa, bajo nuestra supervisión.

Ing. Martha Cecilia Paredes, PhD.
DIRECTORA DEL PROYECTO

Ing. Christian Tipantuña, MSc.
CODIRECTOR DEL PROYECTO

AGRADECIMIENTOS

A mis maestros y a la Universidad por darme la oportunidad de crecer tanto humana como profesionalmente, y por brindarme conocimientos invaluables que difundiré y aplicaré en futuros proyectos para el bienestar de la sociedad.

A la PhD. Martha Cecilia Paredes por su asesoramiento, interés y valiosa colaboración a este proyecto de titulación.

Al MSc. Christian Tipantuña por su aporte en el tema y su continua colaboración en la elaboración de este documento.

DEDICATORIA

Este proyecto está dedicado a la gloria de Dios por brindarme salud, vida y motivación para ser mejor persona y profesional cada día.

Al amor, el esfuerzo y la humildad de mis amados padres, por sacrificarse por mí y no renunciar a la ilusión de que alcance un peldaño más que ellos en la vida.

A mi hermana por ser la amiga eterna que Dios me envió para recordarme siempre que soy ejemplo de vida y que nunca estaré solo.

Finalmente a mis compañeros y amigos porque muchas veces me vi de rodillas y pensé en rendirme, pero sus historias con tristezas y alegrías me motivaron a levantarme siempre.

ÍNDICE DE ABREVIATURAS

AC	Access category (categoría de acceso)
AIFSN	Arbitration InterFrame Space Number (número arbitrario entre espacios de tramas)
AU	Application Unit (unidad de aplicación)
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance (múltiple acceso con escucha de portadora y evasión de colisiones)
DSRC	Dedicated Short-Range Communications (comunicaciones especializadas de corto alcance)
EDCA	Enhanced Distributed Channel Access (mejora en la distribución de canal de acceso)
FCC	Federal Communications Commission (comisión federal de comunicaciones)
IEEE	Institute of Electrical and Electronics Engineers (instituto de ingenieros eléctricos y electrónicos)
GPS	Global Positioning System (sitema de posicionamiento global)
ITS	Intelligent Transportation Systems (sistema de transportación inteligente)
IVC	Inter-Vehicular Communication (comunicación inter vehicular)
VANET	Vehicular Ad-hoc Network (red vehicular ad-hoc)
WANET	Wireless Ad-hoc Network (red inalámbrica ad-hoc)
MANET	Mobile Ad-hoc Network (red móvil ad-hoc)
MPDU	MAC Protocol Data Unit (unidad de datos del protocolo MAC)
MSs	Set of Mobile Station (conjunto de estaciones móviles)
OBU	On Board Unit (unidad a bordo)
OFDM	Orthogonal Frequency Division Modulation (modulación ortogonal por división de frecuencia)
OMNeT++	Objective Modular Network Testbed (probador de redes modulares y objetivas)
PDU	Protocol Data Unit (unidad del protocolo de datos)
PLCP	Physical Layer Convergence Protocol (capa física para el protocolo de convergencia)
PMD	Physical Medium Dependent (dependiente del medio físico)
PPDU	Physical Layer Protocol Data Unit (unidad de datos para el protocolo de capa física)
QoS	Quality of Service (calidad de servicio)
RSU	Road Side Unit (unidades alrededor de la carretera)
SIFS	Short Interframe Space (espacio corto entre tramas)
SUMO	Simulation of Urban Mobility (simulador de movilidad urbana)
TRACI	Traffic Control Interface (interfaz para el control de tráfico)
UTC	Universal Coordinated Time (tiempo universal coordinado)
veins	VEhicles In Network Simulation (simulación de redes en vehículos)
V2V	Vehicle-to-vehicle (vehículo a vehículo)
V2I	vehicle-to-infrastructure (vehículo a infraestructura)
WAVE	Wireless Access in Vehicular Environments (acceso inalámbrico en redes vehiculares)
WBSS	WAVE Basic Service Set (conjunto de servicios básicos WAVE)

CONTENIDO

LISTA DE FIGURAS I
LISTA DE TABLAS III
LISTA DE ANEXOS IV
RESUMEN V
PRESENTACIÓN VI
CAPÍTULO 1: FUNDAMENTOS TEÓRICOS 1
1.1 INTRODUCCIÓN 1
1.2 SEGURIDAD VIAL EN LA CIUDAD DE QUITO 2
1.3 REVISIÓN DE TRABAJOS AFINES 4
1.4 REDES VANETS 6
1.4.1 JERARQUÍA DE REDES VANETS 6
1.4.2 ARQUITECTURA VANET 9
1.4.3 ESTÁNDARES PARA VANET 12
1.5 BREVE DESCRIPCIÓN DE LOS SIMULADORES EMPLEADOS 21
1.5.1 SUMO 21
1.5.2 OMNeT++ 24
1.5.3 VEINS 26
CAPÍTULO 2: DISEÑO, SIMULACIÓN E IMPLEMENTACIÓN 30
2.1 DISEÑO 30
2.1.1 REQUISITOS 30
2.1.2 DISEÑO PARA SIMULACIÓN DE MOVILIDAD 31
2.1.3 DISEÑO PARA LA RED DE COMUNICACIÓN 39
2.2 SIMULACIÓN E IMPLEMENTACIÓN 42
2.2.1 ESCENARIO 1 MOVILIDAD SIN VANET 42
2.2.2 ESCENARIO 2 MOVILIDAD CON VANET 44
CAPÍTULO 3: PRUEBAS Y RESULTADOS 50
3.1 ESCENARIO UNO 50
3.2 ESCENARIO DOS 55
CAPÍTULO 4: CONCLUSIONES Y RECOMENDACIONES 64

4.1 CONCLUSIONES 64
4.2 RECOMENDACIONES 65
REFERENCIAS 67
ANEXOS 69

LISTA DE FIGURAS

Figura 1.1- Siniestros a nivel Nacional abril- 2016 3
Figura 1.2- Estructura del transporte y volúmenes vehiculares en vías principales 4
Figura 1.3- Simulación de tráfico.....5
Figura 1.4- Simuladores VANET híbridos e integrados 6
Figura 1.5- Jerarquía de redes inalámbricas ad-hoc 6
Figura 1.6- Aplicación de redes MANETs.....7
Figura 1.7- Componentes de comunicación en el vehículo 10
Figura 1.8- Componentes de comunicación V2V y V2I 11
Figura 1.9- Comunicación entre infraestructuras.....11
Figura 1.10- Visión general de una VANET.....12
Figura 1.11- Estándares VANETs y su relación con el modelo OSI 14
Figura 1.12- Intervalos de canal.....14
Figura 1.13- Subcapas de IEEE 802.11 y sus unidades de datos 15
Figura 1.14- Espectro de banda reservado para 802.11p 17
Figura 1.15- Formato de trama PPDU en IEEE 802.11p.....19
Figura 1.16- Interfaz y herramientas de SUMO 22
Figura 1.17- Ventana de configuración en SUMO 23
Figura 1.18- Entorno de OMNeT++ 5.0.....25
Figura 1.19- Relación de los componentes de veins 26
Figura 1.20- Entorno de simulación de Veins.....28
Figura 2.1- Área de análisis del centro histórico de Quito 30
Figura 2.2- Intensidad del tráfico vehicular en el centro histórico de Quito 31
Figura 2.3- Archivos generados para SUMO.....33
Figura 2.4- Ruta 0, programada para vehículos rojos.....35
Figura 2.5- Ruta 1, programada para vehículos azules 35
Figura 2.6- Ruta 2, programada para vehículos verdes.....36
Figura 2.7- Ruta 3, programada para vehículos violetas 36
Figura 2.8- Ruta 4, programada para vehículos celestes 37
Figura 2.9- Generación de incidentes en la red de movilidad 38
Figura 2.10- Mejora en la identificación de vehículos.....39

Figura 2.11- Mapa Georeferenciado del área simulada y alcance de RSUs.....41
Figura 2.12- Visualización de mundo real en SUMO.....43
Figura 2.13- Visualización de RSUs en veins.....45
Figura 2.14- Transmisión de AirFrame11p.....47
Figura 2.15- Visualización de fuente, destinos e información de AirFrame11p.....47
Figura 2.16- Eventos que reportan los mensajes eliminados.....48
Figura 2.17- Visualización de un accidente programado.....49
Figura 3.1- Simulación de la ruta 0, escenario uno.....50
Figura 3.2- Simulación de la ruta 1, escenario uno.....51
Figura 3.3- Simulación de la ruta 2, escenario uno.....51
Figura 3.4- Simulación de la ruta 3, escenario uno.....51
Figura 3.5- Simulación de la ruta 4, escenario uno.....52
Figura 3.6- Simulación de rutas 5 y 6, escenario uno.....52
Figura 3.7- Simulación de rutas 7 y 8, escenario uno.....52
Figura 3.8- Simulación de rutas 9 y 10, escenario uno.....53
Figura 3.9- Simulación de rutas 11 y 12, escenario uno.....53
Figura 3.10- Simulación de rutas 13 y 14, escenario uno.....53
Figura 3.11- Parámetros en SUMO para el nodo Ruta0.10.....54
Figura 3.12- Comparación de ruta 0 programada y soluciones VANET.....56
Figura 3.13- Variación de la Ruta1.....57
Figura 3.14- Variación de la Ruta3.....57
Figura 3.15- Variación de la Ruta4.....58
Figura 3.16- Variación de la Ruta6.....58
Figura 3.17- Variación de la Ruta7.....59
Figura 3.18- Variación de la Ruta8.....60
Figura 3.19- Variación de la Ruta11.....60
Figura 3.20- Variación de la Ruta13.....61
Figura 3.21- Comparación de distancia.....62
Figura 3.22- Comparación de tiempos.....62
Figura 3.23- Análisis tiempo-distancia de la utilización de la VANET.....63

LISTA DE TABLAS

Tabla 1.1- Comparación de veins con otros simuladores de VANET 5
Tabla 1.2- Campos de aplicación y ejemplos de uso en VANET [11] 9
Tabla 1.3- Parámetros dependientes de la modulación [22].....16
Tabla 1.4- Parámetros de sincronización [22] 16
Tabla 1.5- Configuración para WAVE [24].....18
Tabla 2.1- Porcentaje, color e identificador de vehículos en 15 rutas.....34
Tabla 2.2- Ubicación de dispositivos RSU.....40
Tabla 3.1- Vehículos con rutas modificadas por VANET 55
Tabla 3.2- Comparación de datos de la Ruta 7 59
Tabla A.1- Siniestros por causas probables [8] 70
Tabla C.1- Distancias y tiempos para los escenarios 1 y 2 72

LISTA DE ANEXOS

ANEXO A– Siniestros por causas probables.....70
ANEXO B – Guía de usuario de OMNeT++ v5.0.....71
ANEXO C– Distancias y tiempos de simulación.....72

RESÚMEN

Uno de los puntos más críticos de la movilidad vehicular en la ciudad de Quito, capital de Ecuador, es la congestión vehicular y los accidentes de tránsito en su centro histórico, lo cual llega a ser insoportable en especial en horas pico por generar estrés, frustración y agresividad en los conductores lo que afecta la calidad de vida de los ciudadanos.

Una alternativa para mejorar el tráfico vehicular en el centro de la ciudad es el uso de una red vehicular ad-hoc (VANET, *Vehicular Ad-hoc NETwork*). Las redes VANETs han logrado gran popularidad debido a que permiten la transmisión de mensajes a lo largo de un conjunto de vehículos (nodos), a través de las carreteras, lo que ayuda a coordinar el tránsito vehicular, proporcionar rutas alternas, transmitir mensajes de emergencia, etc.

En el presente trabajo de titulación se realiza la simulación de una VANET para un área del centro histórico de Quito a fin de determinar si la red ayudará a solucionar los problemas asociados con el tráfico de esta zona de la ciudad. Para la simulación se utiliza veins, que vincula a un entorno de movilidad generado en SUMO, con un ambiente que emula la comunicación de los dispositivos de red, mediante OMNeT++. Este trabajo a más de ser un manual práctico de veins (que trabaja con los programas SUMO y OMNeT++), pretende resaltar la utilización de una VANET como una herramienta idónea para solucionar problemas de tráfico y una solución no agresiva en un área urbana.

De la simulación realizada se obtuvo que un 70% de los vehículos mejoró su movilidad en tiempo, en distancia o simultáneamente en ambos parámetros con lo que se equilibró el tráfico en el área estudiada, priorizando la eficiencia del tráfico, la seguridad y el confort del conductor.

PRESENTACIÓN

En el presente trabajo se describe el diseño y simulación de una red VANET para el centro histórico de la ciudad de Quito mediante el simulador veins que trabaja con los programas SUMO y OMNeT++. Con la finalidad de alcanzar los objetivos planificados, el documento se ha organizado de la siguiente manera:

En el Capítulo 1 se presentan los estándares relacionados con las redes VANETs, haciendo énfasis en el estándar IEEE 802.11p. Se revisan, además, los datos estadísticos que manejan los organismos encargados de la movilidad vehicular del centro histórico de Quito y trabajos similares para recopilar la información de diseño, protocolos, tecnologías, aplicaciones, requerimientos y parámetros asociados a una red VANET. Asimismo, se dan a conocer las principales características y principios de funcionamiento de las herramientas que se utilizarán para la simulación.

El Capítulo 2, parte de toda la información recopilada en los fundamentos teóricos relacionados a las redes VANETs, para presentar el diseño, simulación e implementación de la red VANET, en un área del centro histórico de la ciudad de Quito limitado por las calles: Vicente Rocafuerte, Cuenca, Esmeraldas y Guayaquil. Se presentan requerimientos y parámetros utilizados para el diseño y con base en éstos, se realiza la simulación de la VANET, además se da una breve explicación del uso de programas adicionales como: OpenStreetMap, Google Map y AutoCAD 2D necesarios para el mapeo y la georreferenciación.

En el Capítulo 3, se utilizan los resultados de la simulación de la red VANET y los datos estadísticos de la movilidad vehicular. Además, se realiza una comparación en el cual se determina sí una eventual implementación de VANET es una solución al tráfico vehicular actual y si se generan beneficios adicionales para el centro histórico de la ciudad.

Finalmente en Capítulo 4 se exponen las conclusiones y recomendaciones del proyecto.

CAPÍTULO 1: FUNDAMENTOS TEÓRICOS

1.1 INTRODUCCIÓN

El centro histórico de la ciudad de Quito es víctima del colapso vehicular dado el intenso tráfico que soporta, por ser un sector importante de la ciudad y la conexión entre las zonas sur y norte. Al considerar que el parque automotor de Quito ha rebasado los límites permitidos en áreas metropolitanas [1], se han adoptado soluciones como el pico y placa [2]; no obstante, medidas como ésta no han logrado su objetivo planteado, y por el contrario han incrementado el problema como se muestra en un artículo del diario El Universo [3], publicado en octubre del 2015, donde Ecuador se ubicó entre los 10 peores países para manejar, según el índice global de satisfacción del conductor, esto tras evaluar la experiencia de 50 millones de usuarios en 32 países y 167 ciudades.

Utilizar una solución tecnológica que cumpla con la demanda de la sociedad actual de movilidad y conectividad, en todo momento y lugar como las redes vehiculares Ad-Hoc (VANET, *Vehicular Ad-hoc NETwork*), resulta una opción atractiva para disminuir el tráfico vehicular; sin embargo, implementar una red sin una simulación previa, implica realizar inversiones para adquirir, comprar e instalar los equipos necesarios y solicitar permisos, sin visualizar claramente el comportamiento de la red ni analizar los requisitos y características.

En general, la simulación de redes de datos ha tenido un gran desarrollo, en el caso específico de VANETs es necesario el uso tanto de un simulador de red como de uno de movilidad, que trabajen en lo posible en tiempo real y con el mismo lenguaje de programación.

Veins (*vehicles in network simulation*) es un *software* de simulación híbrido, de código abierto [4], que permite la comunicación bidireccional de los simuladores SUMO (*Simulation of Urban MObility*) [5] y OMNeT++ (*Objective Modular Network Testbed*) [6]. Por un lado, SUMO permite crear el entorno de movilidad urbano, utiliza el lenguaje de programación C++ y maneja archivos que pueden ser generados sistemáticamente mediante comandos. Por otro lado, OMNeT++ se

encarga de la comunicación de la red VANET, administra el control de tiempo y el monitoreo de los paquetes transmitidos. Veins permite trabajar a estos simuladores de manera sincronizada en dos pantallas que tienen correspondencia tanto en tiempo como en el movimiento de los vehículos involucrados, de manera que resulta muy didáctico implementar una VANET y comparar sus resultados con un entorno sin la tecnología.

1.2 SEGURIDAD VIAL EN LA CIUDAD DE QUITO

En el ámbito de justicia y seguridad ciudadana del plan nacional de seguridad integral que emitió el Ministerio de Coordinación de Seguridad, sobresale la seguridad vial como un elemento indispensable, ya que las muertes por accidentes de tránsito tienen un porcentaje superior al de homicidios y asesinatos en el país [7].

"*El Plan Nacional de Seguridad Integral incorpora estrategias para revertir esta grave situación y construir una cultura de Seguridad Vial, mediante un conjunto de acciones continuas que intervienen de forma permanente en el comportamiento de los usuarios de la red vial nacional, y sobre los medios relacionados a la circulación vehicular, con la finalidad de prevenir y disminuir el número de accidentes de tránsito, así como mitigar sus consecuencias*" [7].

En la Tabla A.1 del ANEXO A, se presentan los siniestros y sus porcentajes por causas probables de mayor a menor a nivel nacional hasta abril del 2016 [8], y en los cuatro primeros lugares se encuentran:

- Conducir desatento a las condiciones de tránsito
- Superar los límites de velocidad
- No respetar las señales de tránsito
- No mantener una distancia prudencial

En la Figura 1.1, se presenta un horario con intensidad de calor para siniestros según día y hora de ocurrencia a nivel nacional, es decir que entre semana las llamadas horas pico se identifican con un tono naranja rojizo al inicio y al cierre de la jornada laboral, éstas coinciden con la mayor tasa de accidentes mencionados

en el párrafo anterior. Por otro lado, durante el fin de semana los accidentes se incrementan alrededor de las 20:00, debido a la concurrencia a sitios de afluencia masiva.

RANGO HORA	LUNES	MARTES	MIERCOLES	JUEVES	VIERNES	SÁBADO	DOMINGO	TOTAL
00:00 A 00:59	7	7	5	5	11	28	14	77
01:00 A 01:59	2	3	1	9	5	27	13	60
02:00 A 02:59	3	3	3	9	9	19	9	55
03:00 A 03:59	1	1	7	6	9	18	15	57
04:00 A 04:59	2	1	1	7	7	11	22	51
05:00 A 05:59	6	12	4	9	11	19	9	70
06:00 A 06:59	16	12	22	16	20	21	12	119
07:00 A 07:59	14	19	8	20	22	20	19	122
08:00 A 08:59	18	19	19	20	22	14	10	122
09:00 A 09:59	14	3	21	10	15	22	10	95
10:00 A 10:59	14	14	15	11	16	28	8	106
11:00 A 11:59	19	12	8	8	16	14	17	94
12:00 A 12:59	15	13	14	16	25	21	13	117
13:00 A 13:59	15	21	17	16	21	23	24	137
14:00 A 14:59	15	13	10	13	16	26	12	105
15:00 A 15:59	18	24	16	23	17	23	21	142
16:00 A 16:59	21	13	19	17	24	27	23	144
17:00 A 17:59	16	13	8	18	14	27	19	115
18:00 A 18:59	22	13	14	11	18	27	12	117
19:00 A 19:59	17	17	18	5	25	30	22	134
20:00 A 20:59	14	19	11	10	20	36	21	131
21:00 A 21:59	15	9	14	7	21	33	11	110
22:00 A 22:59	12	8	6	15	20	28	8	97
23:00 A 23:59	7	10	3	7	22	21	6	76
	303	279	264	288	406	563	350	2453

Fuente: DNCTSV, CTE, EMOV - Cuenca, Gobierno Autónomo Descentralizado Municipal de Loja, Agencia Metropolitana de Tránsito - Quito, Gobierno Autónomo Descentralizado de Ma Ambato, Municipio de Ibarra, Autoridad de Tránsito Municipal de Guayaquil.

Figura 1.1- Siniestros a nivel Nacional abril- 2016 [8]

La estructura del transporte colectivo y el volumen diario vehicular de Quito en sus principales ejes viales se muestran en la Figura 1.2, donde se observan que las zonas de color naranja, evidencian la densidad de tráfico más alto, como en el centro histórico de Quito y el antiguo aeropuerto. Los 729.900 viajes que se realizaron en el área señalada, a la fecha del estudio en el transporte público, significaron el 45% del flujo total de pasajeros, lo que proyectó al centro histórico como una de las zonas con mayor afluencia en la capital, según la página de la Secretaría de Territorio, Hábitat y Vivienda del Distrito Metropolitano de Quito [9].

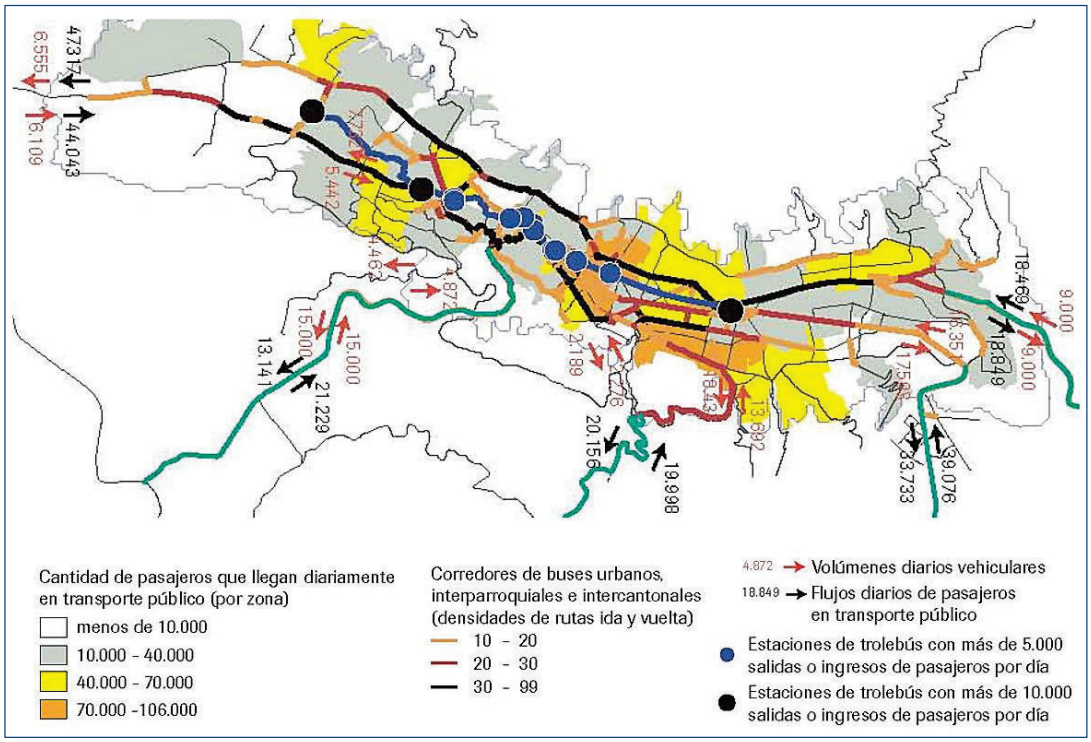


Figura 1.2- Estructura del transporte y volúmenes vehiculares en vías principales [9]

1.3 REVISIÓN DE TRABAJOS AFINES

En las siguientes líneas se realiza una revisión general de trabajos afines, para lo cual se han considerado proyectos desarrollados tanto a nivel nacional como internacional.

En la mayoría de los trabajos encontrados en la literatura, las simulaciones se realizan solamente con modelos de movilidad básicos, que consisten de una zona geográfica mínima, es decir, que solo consideran un par de intersecciones. Estos trabajos son realizados simplemente para comprobar la transmisión de datos entre vehículos y la activación de algunas alertas de colisión.

El trabajo presentado en [10], en el año 2015 por la Universidad Politécnica Salesiana (Cuenca - Ecuador), realiza un análisis de factibilidad para crear una red vehicular inteligente integrando una red Ad-Hoc con una red LTE (*Long Term Evolution*), para lo cual recopila información estadística detallada de las causas de accidentes en las principales carreteras de Ecuador. Además, el estudio incluye la descripción de varios componentes y protocolos empleados por la red VANET. El

diseño y simulación de la red se realiza por medio de los simuladores SUMO, NS2 y NS3. El área analizada está formada solamente por cuatro intersecciones (ver Figura 1.3). Los resultados que se pueden extraer son muy restringidos ya que dicho estudio se limita a la factibilidad de simulación de una VANET en estos simuladores.

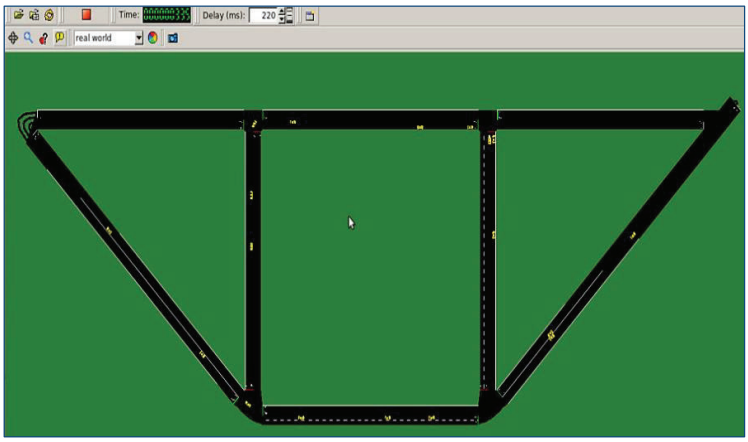


Figura 1.3- Simulación de tráfico presentado en [10]

Un proyecto realizado por la Universidad Carlos III de Madrid (España) en 2012 presenta un estudio sobre el simulador híbrido de redes vehiculares veins [11]. En dicho estudio se demuestra que los simuladores de redes vehiculares son una solución viable para tratar el tráfico en ciudades y carreteras. Además, en la tabla 1.1 se destaca que veins sobresale de otros simuladores como TraNS [12], GrooveNET [13] y MobiREAL [14], y la Figura 1.4 muestra sus componentes.

Tabla 1.1- Comparación de veins con otros simuladores de VANET

	Control tiempo	Actualización	Software libre	Hibrido
Veins	√	√	√	√
TraNS	X	X	√	√
GrooveNet	X	X	√	X
MobiREAL	√	X	X	√

Los simuladores de tipo integrado (manejan un solo mecanismo para red y movilidad), a diferencia de los híbridos que se ven obligados a simplificar la interacción de sus componentes para movilidad y red, degradando la calidad de la simulación [11]. Así, TraNS al ser comparado con veins es una de las mejores opciones por ser software libre y de tipo hibrido; sin embargo, no considera el

control de intervalos de tiempo entre sus componentes y no se ha actualizado en los últimos años.

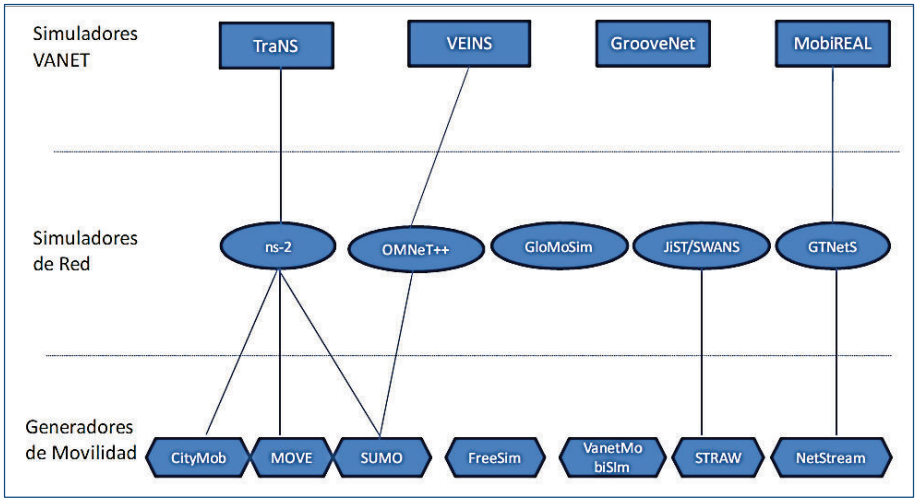


Figura 1.4- Simuladores VANET híbridos e integrados [11]

1.4 REDES VANETS

1.4.1 JERARQUÍA DE REDES VANETS

Una Red Vehicular ad-hoc[1] (VANET) es un subtipo de redes móviles MANETs (*Mobile Ad-hoc Networks*), y éstas a su vez forman parte del grupo principal de redes inalámbricas Ad-hoc WANETs (*Wireless Ad-hoc Networks*). El resumen de esta jerarquía de Redes Inalámbricas ad-hoc se presenta en la Figura 1.5.

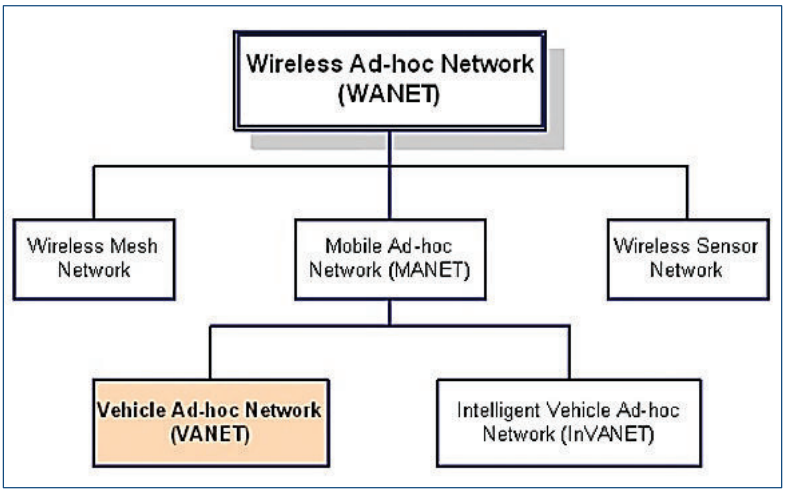


Figura 1.5- Jerarquía de redes inalámbricas ad-hoc [15]

[1] Ad-hoc: Locución latina que significa "para esto" y se refiere a algo improvisado. Es un tipo de red inalámbrica descentralizada, que no depende de infraestructura previa, brinda flexibilidad y autonomía gracias al principio de autoconfiguración [30].

Para lograr innovación en tecnología debe existir una continua investigación de WANET y de cada una de sus variaciones. Todas las ramas del árbol de redes inalámbricas, ver Figura 1.5, cuentan con un grupo de estudio dedicado a cubrir necesidades específicas.

Una MANET es una red auto configurable sin infraestructura fija, de tipo malla formada por un conjunto de estaciones móviles (MSs, *Set of Mobile Station*), conectadas mediante un medio inalámbrico y que se mueven libres e independientes en cualquier dirección dentro de un área de cobertura determinada, por lo tanto cambian con frecuencia los enlaces entre sus dispositivos con una topología de red dinámica [15].

Las aplicaciones principales de una red MANET son en el área militar, medica, de educación, para manejo de desastres y seguridad como se muestra en la Figura 1.6. Además, este tipo de redes cuenta con las siguientes características:

- Enlaces de capacidad variable
- Operación de energía finita
- Limitada seguridad en capa física

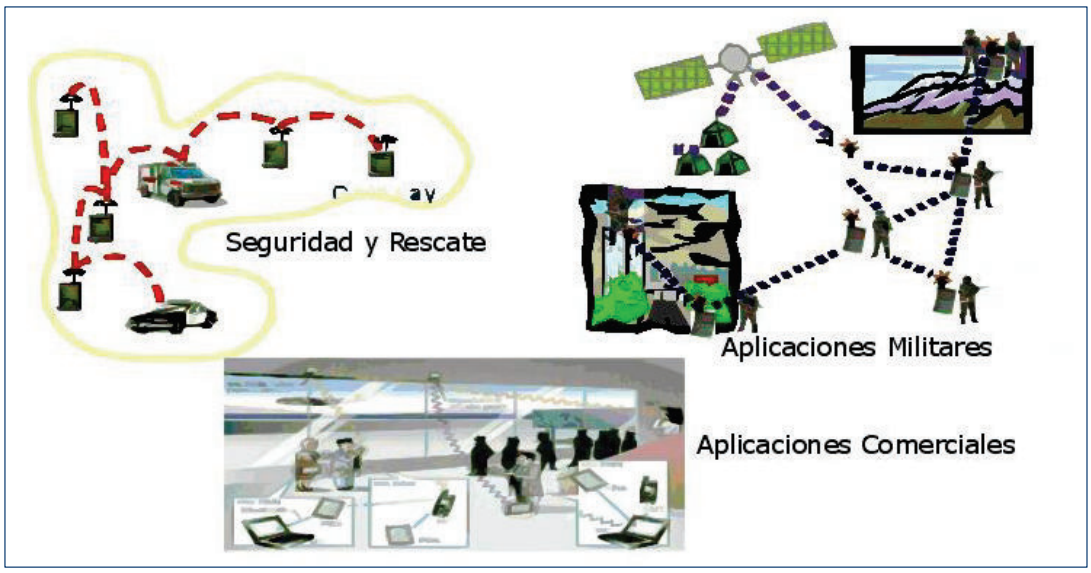


Figura 1.6- Aplicación de redes MANETs [16]

VANET es un tipo de red MANET, especializada en proveer comunicación intervehicular (IVC, *Inter-Vehicular Communication*), cuyo objetivo es mejorar

tanto la experiencia como la seguridad del usuario al conducir. A más de contar con las propiedades de las redes móviles inalámbricas ya citadas, también cuenta con sus propias características de red vehicular, que de acuerdo a [15] son:

- Sus nodos poseen alta capacidad computacional y de energía, ya que el vehículo no es un limitante ante estos factores
- Pueden ser de naturaleza distribuida y estar basada en comunicación de nodo a nodo, o de naturaleza centralizada al incluir una estación fija
- Tienen naturaleza dinámica, ya que las posiciones de los nodos móviles cambian constantemente, y varían la topología de red
- Intercambian información frecuentemente entre sus nodos para acoplarse a la naturaleza dinámica de la red
- El procesamiento de información es en tiempo real y cuentan con auto organización para comunicar sus nodos móviles en una topología de red rápidamente variable
- Poseen baja volatilidad, esto indica que el rango de comunicación entre nodos dura un periodo corto de tiempo, ya que los datos intercambiados varían rápidamente
- Proveen una comunicación con un rango entre 5 y 10 [km] sin considerar zonas de sombra
- Tienen limitaciones de movilidad (restricción de velocidad y tipo de vehículos en ciertos caminos)

La comunicación de vehículos en una VANET es la base de la tecnología, conviene subrayar que no se limita únicamente a monitorear el tráfico de vehículos en las ciudades, como muchas aplicaciones para teléfonos inteligentes lo realizan hoy en día, VANET es la base para muchas aplicaciones. A continuación, en la Tabla 1.2 se enumeran algunos campos de aplicación de las redes vehiculares como: seguridad vial, servicio público, mejora de la conducción y entretenimiento; además se nombran varios ejemplos de cada campo con su respectivo propósito, esto evidencia múltiples soluciones que se generan al aplicar esta tecnología. Para mayor información, en los trabajos de titulación presentados en [10] y [11] se dedican varias páginas a discutir en detalle de características y aplicaciones de las VANETs.

Tabla 1.2- Campos de aplicación y ejemplos de uso en VANET [11]

Parámetros	Propósito	Ejemplo
Seguridad vial	Carreteras peligrosas	Advertencia de intersecciones
	Condiciones anormales de tráfico	Advertencia de zona de obras
	Peligro de colisión	Advertencia de ángulo muerto; cambio de carril; peatones que cruzan
	Choque inminente	Frenada pre-accidente
	Aviso de incidente	Alerta tras colisión ya ocurrida. Aviso SOS.
Servicio público	Respuesta de emergencia	Alerta vehículo de emergencias, señal de anticipación
	Apoyo a las autoridades	Matrícula o licencia de conducir electrónica
Mejora de la conducción	Mejora de la conducción	Asistente de carretera o giro
	Eficiencia del tráfico	Notificación accidentes; guía de ruta y navegación; aparcamientos
Negocios y entretenimiento	Mantenimiento de vehículos	Diagnósticos remotos; avisos de seguridad
	Servicios móviles	Internet; notificaciones de puntos de interés
	Soluciones empresariales	Gestión de flotas; alquiler; seguimiento de la carga

1.4.2 ARQUITECTURA VANET

A continuación, se presentan las unidades principales de una red VANET [17].

- Unidad a bordo denominada OBU (*On Board Unit*)
- Unidad de aplicación llamada AU (*Application Unit*)
- Unidades alrededor de la carretera nombradas RSU (*Road Side Unit*)

En una VANET, cada vehículo se considera como un nodo móvil que contiene las unidades a bordo (OBU) y de aplicación (AU), tal como se muestra en la Figura 1.7. La AU en el vehículo permite adquirir información de los diferentes componentes como: GPS, direccionales, sensores de velocidad, sensores de frenado, sensores de fatiga, entre otros. Por otro lado, la OBU tiene como objetivo la comunicación inalámbrica dentro de un rango de alcance. La figura 1.7 ilustra el

nodo VANET en un entorno de redes vehiculares, el cual comparte datos de diferentes sensores e información básica del vehículo, el nodo coopera con prioridad con la información de velocidad y la posición GPS de cada OBU, para formar una red en la que se comunican eventos continuos en el trayecto de los nodos. Estos datos se procesan en la red, para dar a conocer a los vehículos del comportamiento de los equipos vecinos, empleando los distintos tipos de comunicación de una VANET, entre las que según [18] están las siguientes:

- Comunicación intravehicular
- Comunicación vehículo a vehículo (V2V)
- Comunicación de vehículo a infraestructura (V2I)
- Comunicación entre infraestructuras
- Comunicación del vehículo a la nube

Figura 1.7- Componentes de comunicación en el vehículo

En las comunicaciones V2I, las infraestructuras son nodos estáticos llamados estaciones fijas o RSUs que están ubicados estratégicamente en distintos lugares de las carreteras dentro del área de cobertura de la red. Un ejemplo de esto se muestra en la Figura 1.8.

La comunicación entre RSUs extiende la cobertura de la VANET; se comparte toda la información concentrada de la red ad-hoc, mediante su correspondiente

RSU con otras estaciones fijas similares para formar la comunicación entre infraestructuras, ver Figura 1.9, e informar los eventos dados a los vehículos que componen la red Ad-hoc vecina, previniéndoles de posibles incidentes.

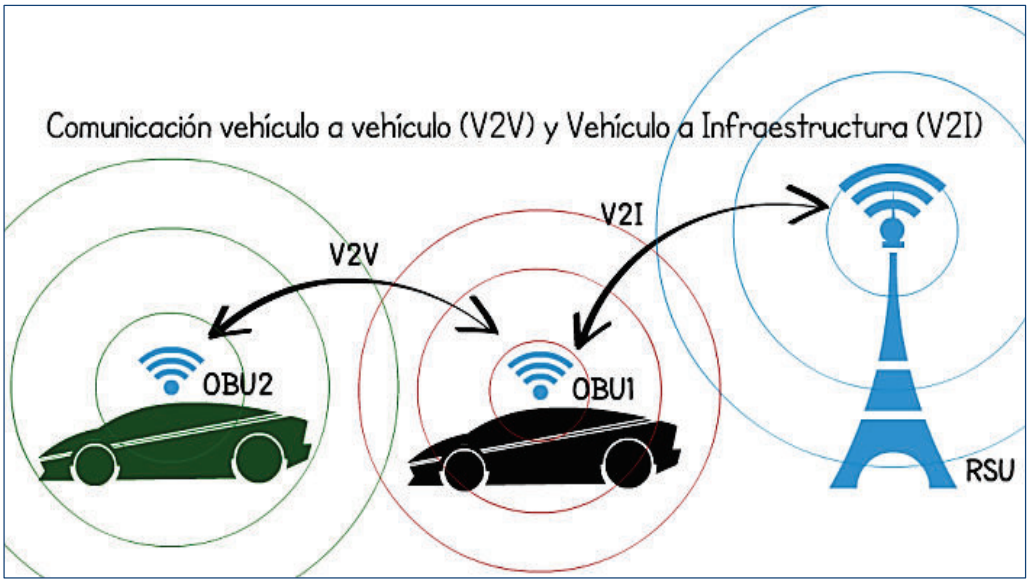


Figura 1.8- Componentes de comunicación V2V y V2I

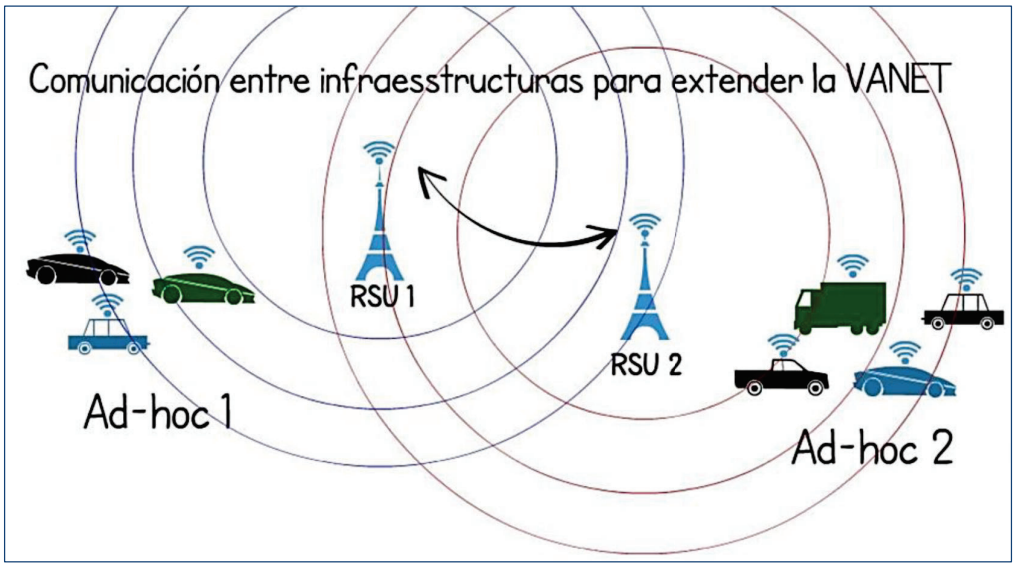


Figura 1.9- Comunicación entre infraestructuras

Con las interacciones entre los elementos de una VANET ocurren las distintas formas de comunicación como las ya mencionadas; en la Figura 1.10 se pueden observar varias interacciones que trabajan en conjunto dentro de una VANET.

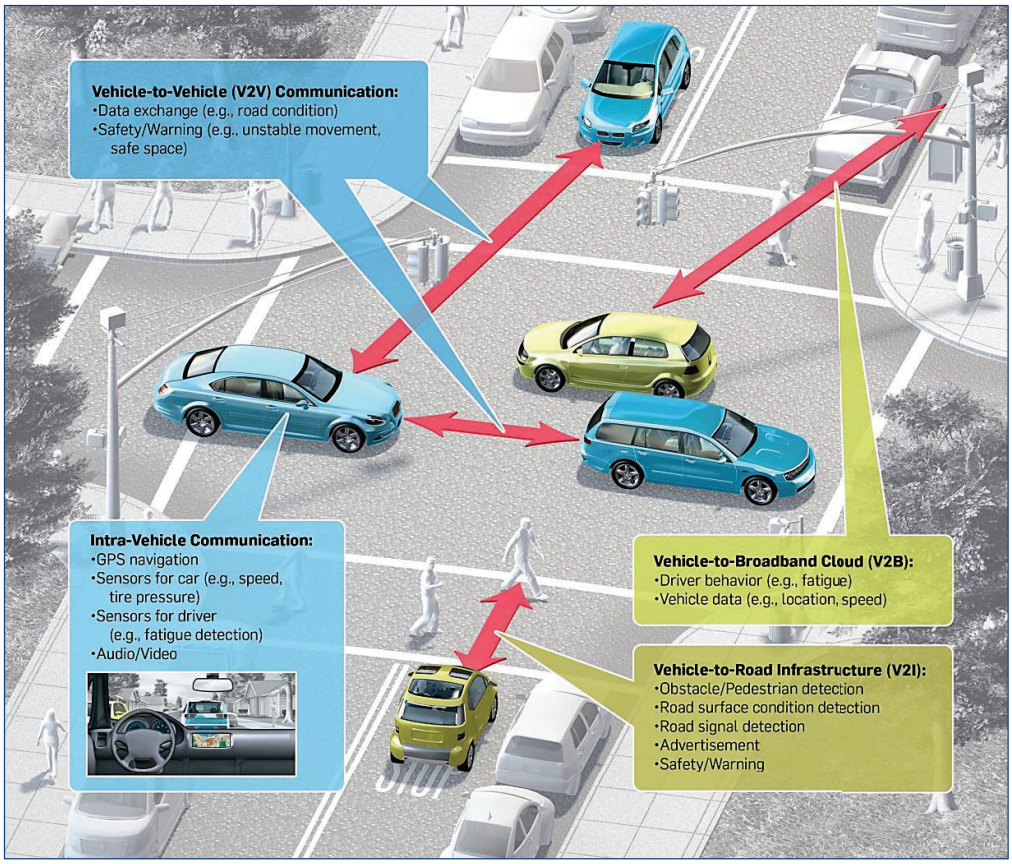


Figura 1.10- Visión general de una VANET [18]

Gracias a la comunicación de todos los dispositivos que conforman la red, se conoce sobre la disponibilidad de las vías a las que pueden acceder los vehículos, para equilibrar la carga de tráfico en las calles y descongestionar las rutas; la comunicación permite además anticipar a los usuarios de incidentes en su trayecto para prevenir nuevos siniestros, y se puede conocer la presencia de vehículos de emergencia a través del envío de señales directas en la OBU, para reforzar las señales visuales y auditivas de la sirena. Todo esto permite pensar en nuevas aplicaciones que se podrían implementar en futuros proyectos para mejorar la experiencia y seguridad del usuario.

1.4.3 ESTÁNDARES PARA VANET

Las redes inalámbricas vehiculares como la mayoría de tecnologías, tienen dos grandes desarrolladores principales en el mundo. Para Europa, los sistemas de transporte inteligente (ITS, *Intelligent Transportation Systems*) están regidos por el

Instituto de Estandarización en Telecomunicaciones Europeo (ETSI, *European Telecommunications Standards Institute*), y su arquitectura está basada en un modelo de 6 capas que utiliza protocolos propios de su región y en frecuencias de trabajo de 2.8 [GHz] y 5 [GHz] [19].

Por otro lado en América, el encargado del desarrollo y estandarización es el Instituto de Ingenieros Eléctricos y Electrónicos (IEEE, *Institute of Electrical and Electronics Engineers*), que se originó en Estados Unidos. Para el estudio de ITS, en 1999 la Comisión Federal de Comunicaciones (FCC, *Federal Communications Commission*) designó la banda de 5.9 [GHz] (5.850-5.925 [GHz]) con un ancho de banda de 75 [MHz] para los servicios de comunicaciones especializados de corto alcance DSRC (*Dedicated Short-Range Communications*), que ya se encontraban en desarrollo desde 1992. En años posteriores fue necesario la creación de un grupo de trabajo especial para el estudio de redes inalámbricas en área local IEEE 802.11, este estándar define la capa física y la parte baja de la subcapa MAC dirigida a las redes de área local inalámbrica (WLAN, *Wireless local Area Network*), como una enmienda de este estándar está el IEEE 802.11p, el cual está enfocado a entornos vehiculares. Dicha enmienda es una modificación del IEEE 802.11a, pero que trabaja con bajo *overhead*[2] y logra mayor eficiencia espectral [19] y [20].

El grupo IEEE 1609 se creó años más tarde, para encargarse de las capas superiores del acceso inalámbrico en redes vehiculares denominado (WAVE, *Wireless Access in Vehicular Environments*). En la Figura 1.11 se muestra en la equivalencia con el modelo OSI (*Open System Interconnection*) a la integración de estos estándares, encontrándose en la capa inferior a IEEE 802.11p y en las capas superiores a IEEE 1609. Existen subestándares dentro del IEEE 1609 que se encargan de cada capa superior. Para la comunicación de la subcapa MAC con capas superiores se emplea el IEEE1609.4, que es un estándar para definir las operaciones en un ambiente multicanal sobre la banda de 5.9 [GHz]; es decir, busca definir un mecanismo para que un sistema con dispositivos que se están moviendo entre varios canales, puedan encontrarse en un canal común y se sintonicen en un mismo tiempo para que puedan comunicarse [20] y [21].

[2] *Overhead*: bits extras a los de información o carga útil (*payload*), usados para control e identificación en una trama

Aplicación	Capa 7	IEEE 1609
Presentación	Capa 6	
Sesión	Capa 5	
Transporte	Capa 4	
Red	Capa 3	
Enlace	Capa 2	IEEE 802.11p
Física	Capa 1	

Figura 1.11- Estándares VANETs y su relación con el modelo OSI [20]

El canal común en el que los dispositivos se sincronizan regularmente se denomina canal de control CCH (*Control Channel*) designado así por IEEE 1609.4. Para lograr la sincronización se utiliza una fuente de tiempo común entre los dispositivos que es la señal del sistema de posicionamiento global GPS con su tiempo universal coordinado UTC (*Universal Coordinated Time*). Una vez sincronizados todos los dispositivos, el subestándar IEEE 1609.4 impone la división de tiempo en intervalos para el canal de control e intervalos para los canales de servicio SCH (*Service Channels*) como se muestra en la Figura 1.12 [20] y [21].

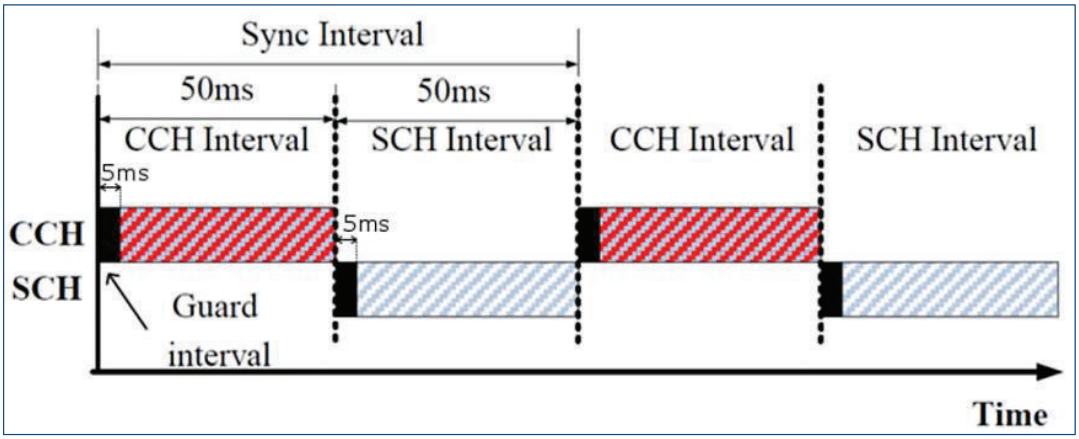


Figura 1.12- Intervalos de canal [20]

1.4.3.1 Estándar IEEE 802.11p

Las especificaciones del estándar IEEE 802.11p se centran en las dos capas inferiores del modelo de referencia OSI. A continuación, se dedican dos secciones

para una breve descripción de las principales características de la capa física y la subcapa MAC.

1.4.3.1.1 Capa Física

La capa física en la pila de protocolos IEEE 802.11 se divide en dos subcapas como se muestra en la Figura 1.13. La subcapa de función de convergencia PLCP (*Physical Layer Convergence Protocol*) indica la velocidad de los datos y la longitud de los paquetes, además prepara la unidad de datos del protocolo MAC MPDU (*MAC Protocol Data Unit*) para crear la unidad de datos para el protocolo de capa física PPDU (*Physical Layer Protocol Data Unit*). Mientras que en la parte más baja, la subcapa dependiente del medio físico PMD (*Physical Medium Dependent*) establece los métodos de transmisión y recepción con las características necesarias para usar el medio inalámbrico.

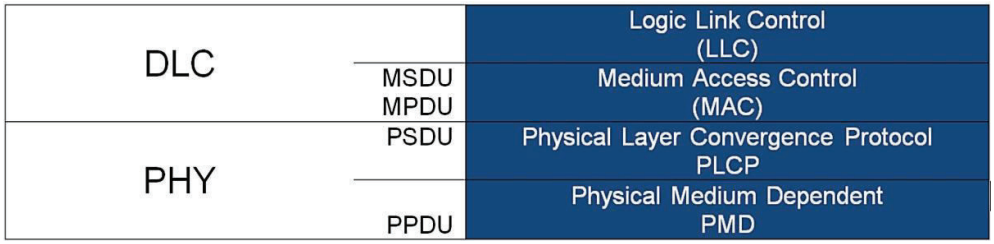


Figura 1.13- Subcapas de IEEE 802.11 y sus unidades de datos [22]

En el grupo encargado del estándar inalámbrico IEEE 802.11, los servicios de comunicaciones especializados de corto alcance DSRC son conocidos como IEEE 802.11p WAVE y su capa física es una variación de IEEE 802.11a, por lo que se utiliza OFDM (*Orthogonal Frequency Division Modulation*) con diferentes tipos de modulación como: BPSK (*Binary Phase Shift Keying*), QPSK (*Quadrature Phase-Shift Keying*), 16QAM (16 *Quadrature Amplitude Modulation*) y 64QAM (16 *Quadrature Amplitude Modulation*); además, emplea diferentes tasas de codificación convolucional (R) como se muestra en la Tabla 1.3. El estándar cuenta con 64 subportadoras de las cuales se toman 52 para transmisión, de

éstas 48 son para datos y 4 son pilotos para rastrear el *offset* o desplazamiento de frecuencia[3] y el ruido de fase[4] [23].

Tabla 1.3- Parámetros dependientes de la modulación [22]

Modulation	Coding rate (R)	Coded bits per subcarrier (N_{BPSC})	Coded bits per OFDM symbol (N_{CBPS})	Data bits per OFDM symbol (N_{DBPS})	Data rate (Mb/s) (10 MHz cannel spacing)
BPSK	½	1	48	24	3
BPSK	¾	1	48	36	4.5
QPSK	½	2	96	48	6
QPSK	¾	2	96	72	9
16-QAM	½	4	192	96	12
16-QAM	¾	4	192	144	18
64-QAM	$^2/_3$	6	288	192	24
64-QAM	¾	6	288	216	27

El estándar IEEE 802.11p para disminuir el efecto por propagación Doppler[5], cuenta con una variación de IEEE 802.11a, ya que usa canales de 10 [MHz] en lugar de los 20 [MHz] y debido a esto, varios parámetros de sincronización se duplican (ver Tabla 1.4), como por ejemplo: el intervalo de banda de guarda, la duración de símbolos OFDM y los componentes de la trama como los campos de preámbulo PLCP, el campo de señal y otros que se describen más adelante. Por el contrario, las velocidades de datos se reducen a la mitad, ofreciendo 3, 4.5, 6, 9, 12, 18, 24 y 27 [Mbps] en lugar de 6, 9, 12, 18, 24, 36, 48 y 54 [Mbps] como en el estándar IEEE 802.11a [23], cuyos valores se pueden observar en la Tabla 2.

Tabla 1.4- Parámetros de sincronización [22]

Parameter	Value (20 MHz cannel spacing)	Value (10 MHz cannel spacing)
$T_{PREAMBLE}$: PLCP preamble duration	16µs ($T_{SHORT} + T_{LONG}$)	32µs ($T_{SHORT} + T_{LONG}$)
T_{SIGNAL}: Duration of the SIGNAL BPSK − OFDM symbol	4.0µs ($T_{GI} + T_{FFT}$)	8.0µs ($T_{GI} + T_{FFT}$)
T_{GI}: GI duration	0.8µs ($T_{FFT}/4$)	1.6µs ($T_{FFT}/4$)
T_{GI2}: Training symbol GI duration	1.6µs ($T_{FFT}/2$)	3.2µs ($T_{FFT}/2$)
T_{SYM}: Symbol interval	4µs ($T_{GI} + T_{FFT}$)	8µs ($T_{GI} + T_{FFT}$)
T_{SHORT}: Short training sequence duration	8µs ($10 * T_{FFT}/4$)	16µs ($10 * T_{FFT}/4$)
T_{LONG}: Long training sequence duration	8µs ($T_{GI2} + 2 * T_{FFT}$)	16µs ($T_{GI2} + 2 * T_{FFT}$)

[3] Desplazamiento de frecuencia (offset): Es un cambio leve intencional de la frecuencia original para reducir la interferencia entre transmisores que usan el mismo canal.

[4] Ruido de fase: Son componentes extras a la portadora principal usada, estos componentes tienen frecuencias y amplitudes aleatorias que impiden distinguir la información original y se incrementan con los amplificadores del sistema [31].

[5] Doppler: Fenómeno que aumenta o disminuye la frecuencia de una onda al acercarla o alejarla de la fuente [32].

El grupo de 7 canales que tiene WAVE en la banda de (5.850 a 5.925) [GHz] se ilustra en la Figura 1.14. La parte baja de frecuencias reserva 5 [MHz] para un margen de seguridad de la banda, los canales 172 y 184 de los extremos tienen prioridad para aplicaciones de seguridad vital y pública. La frecuencia de 5.890 [GHz] corresponde al canal de control 178 (CCH, *Control Channel*), y a sus costados se encuentran los canales de servicio 174, 176, 180 y 182 (SCH, *Service Channel*), estos últimos transmiten datos para servicios y tienen la prioridad más baja, que se da después de la negociación del canal de control. Los canales 174 y 176 se combinan para tener un mayor ancho de banda (AB), dando lugar al canal 175 con 20 [MHz], y del mismo modo con 180 y 182 se forma el canal 181 de 20 [MHz] [23].

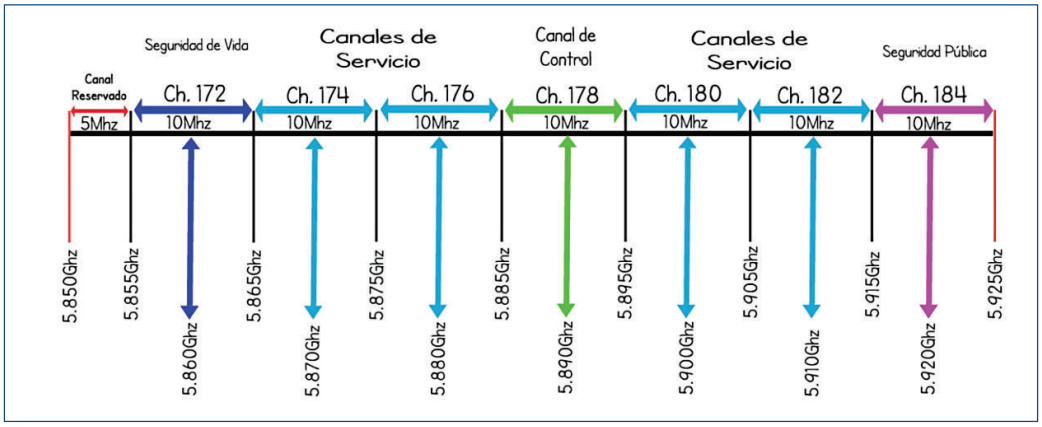


Figura 1.14- Espectro de banda reservado para 802.11p [23]

Los principales intervalos de canal corresponden a CCH y SCH como se puede observar en la Figura 1.12. El canal de control CCH no permite el tráfico IP (*Internet Protocol*), ya que está dedicado para mensajes de gestión como WSA[6] y mensajes WSMP[7] [23]. Mientras que los canales de servicio SCH son utilizados para aplicaciones en general que permiten tráfico de gestión, tráfico WSMP e IPV6 [23].

El consorcio de comunicaciones de seguridad vehicular VSCC (*Vehicular Safety Communications Consortium*) estableció una latencia[8] máxima de 100 [ms] para WAVE, dicho esto, el valor es muy alto al ser comparado con WIFI (*WIreless*

[6] WSA: Servicio de publicidad WAVE, (WSE, por sus siglas *WAVE Service Advertisement*) [23].
[7] WSMP: Mensaje corto de protocolo WAVE, (WSMP, del inglés *WAVE Short Message Protocol*) [23].
[8] Latencia: Es la suma de retardos temporales en una red debido a la demora por propagación y transmisión [32].

FIdelity) de 36 [ms] debido a que una alta latencia se refleja en la disminución del rendimiento de la red; el valor alto se asume porque en la comunicación V2V, es común la pérdida de paquetes debido a la alta densidad de nodos y el movimiento de los mismos. En la Figura 1.14 se observan los intervalos de sincronización del canal que incluyen el tiempo de guarda de 5ms, el intervalo para el canal de control con 50 [ms] y el canal de servicio que lleva el mismo periodo de tiempo [20].

En la Tabla 1.5 se observan algunos parámetros a ser configurados en WAVE, los cuales son: la longitud del espacio de canales, los bits para las ventanas de contención mínima y máxima, el espacio corto entre tramas SIFS (*Short Interframe Space*) y los anchos de banda disponibles para transmisión [24].

Tabla 1.5- Configuración para WAVE [24]

Parámetro	Valor
Longitud de ranura CCH/SCH – Intervalo de Guarda	50ms - 5ms
Ventana de Contención aCWmin y aCWmax	15 y 1023
Ranura de tiempo	13us
SIFS	32us
Ancho de Banda	3Mbit/s...27Mbit/s

El estándar IEEE 802.11p trabaja con nodos que tienen velocidad alta y un tiempo de comunicación muy corto, por lo que se emplea un nuevo conjunto de servicios básicos exclusivos para WAVE denominados WBSS (*Wave Basic Service Set*). Para simplificar la asociación de vehículos a este grupo de servicios, un proveedor transmite periódicamente un mensaje WSA en el CCH, este mensaje contiene información básica para que los usuarios se unan al BSS (*Basic Service Set*). Un mensaje WSA posee: el identificador BSSID (*Basic Service Set Identifier*), la disponibilidad de un servicio de WBSS, la información de sincronización, etc. WSA omite los procesos de escaneo, autentificación y asociación típicos, ya que está disponible todo el tiempo para mejorar la eficiencia en recepción y transmisión de datos [20].

La subcapa PLCP de la capa física en el estándar IEEE 802.11p define los formatos de la trama PPDU (*Physical Layer Protocol Data Unit*), por lo que añade

un preámbulo, una cabecera, bits para la cola (*tail*) y bits para el relleno (*padding*). Los parámetros de sincronización duplicados en comparación a IEEE 802.11 es la variación más evidente en el formato de la trama (ver Figura 1.15).

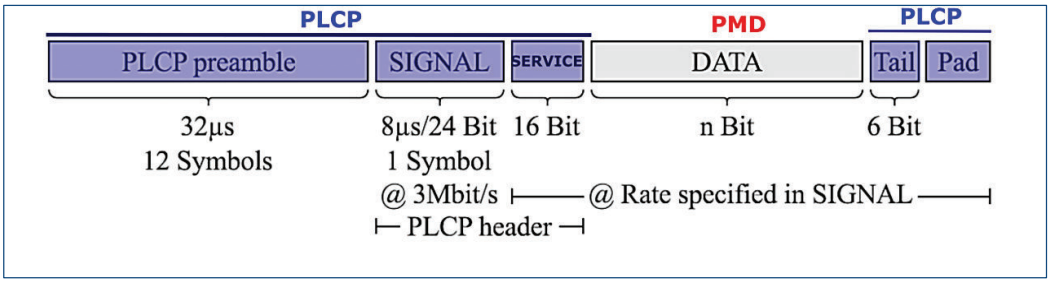


Figura 1.15- Formato de trama PPDU en IEEE 802.11p [24]

La Figura 1.15 muestra los campos que componen a la trama PPDU. El preámbulo PLCP cuenta con 12 símbolos y una duración de 32 [µs] para la sincronización entre el transmisor y receptor, además se emplea para la selección de una antena adecuada en tiempo y frecuencia. El encabezado PLCP (*header*) incluye a los campos de señal (*signal*) y servicio (*service*); el campo de señal tiene 1 símbolo OFDM con 24 bits y una duración de 8 [µs], los subcampos del campo señal son: la velocidad empleada (*rate*) en [Mbps] (4 bits), el bit para uso futuro en cero (*reserved*) (1 bit), la cantidad de bits de la trama (*lenght*) (12 bits), el bit de paridad (*parity*) para evitar corrupción de datos (1 bit) y bits en cero con que termina el código convolucional (*tail*) (6 bits) [22].

El campo de servicio que está incluido en la cabecera PLCP tiene 16 bits que se numeran del bit cero al 15, los bits del cero al seis se usan para sincronizar el decodificador en el receptor y los bits del siete al 15 se mantienen como una reserva. La cola de la trama PPDU cuenta con seis bits encerados para regresar el código convolucional a cero y mejorar el FEC (*Forward Error Correction*). Finalmente, el campo de relleno (*padding*) tiene un número de bits variable que se utilizan para extender la información sin codificar.

1.4.3.1.2 Subcapa MAC

En cuanto a la subcapa de Acceso al Medio (MAC) y a la calidad de servicios (QoS), el estándar IEEE 802.11p cuenta con una variación de IEEE 802.11e, que

emplea una función de coordinación hibrida (HCF, *Hybrid Coordination Function*) con dos mecanismos de acceso al medio mejorados, de los cuales se toma a EDCA (*Enhanced Distributed Channel Access*) definido en el 2007. Además, EDCA se requiere para tener comunicación en tiempo real, y cumplir con requisitos de QoS. Los grupos que otorgan QoS se denominan Categoría de Acceso (AC) y en orden de prioridad son: a) voz (VOI, *Voice*), b) video (VI, *Video*), c) mejor esfuerzo (BE, *Best Effort*) y d) antecedentes (BK, *Background*) [25].

EDCA es el mecanismo para redes vehiculares, es tomado como solución para el caso en que un nodo envíe varios paquetes al mismo tiempo, ya que en lugar de tener una cola por cada AC como en IEEE 802.11a por ejemplo, éste maneja una cola por la combinación de AC y tipo de canal; gracias a esto se logra una composición de ocho colas que son controladas cada una por una Función EDCA (EDCAF, *Enhanced Distributed Channel Access Function*). Cuando las tramas llegan a la subcapa MAC, se clasifican por su prioridad, y luego la diferencia de tráfico se da por parámetros de acceso al medio diferentes para cada AC, y según [24] y [25] estos son:

- El número de espacios arbitrarios entre tramas AIFSN (*Arbitration InterFrame Space Number*)
- La ventana de contención mínima y máxima *CWmin & CWmax* (*Contention Window*)
- La oportunidad de transmisión TXOP (*Transmission Opportunity*)

EDCA emplea múltiple acceso al canal con escucha de portadora y evasión de colisiones CSMA/CA (*Carrier Sense Multiple Access with Collision Avoidance*), y se basa en la transmisión dependiendo del intervalo AIFS disponible para cada AC y en un tiempo aleatorio de *Backoff*, que se utiliza para aplazar el envío de datos si el canal se encuentra ocupado [23].

En el trabajo presentado en [23] se evalúa el estándar IEEE 802.11p basada en 3 categorías: analítica, simulación y pruebas. Del análisis del modelo analítico se recomienda un rango de 200 [m] como máximo y no los 1000 [m] planteados por el estándar para tener un corto retardo y alcanzar el máximo rendimiento; además, se concluye que es posible cubrir el requerimiento de latencia menor a

100 [ms], pero no se logra cubrir la fiabilidad mayor al 99.9% que se requiere para mensajes de seguridad [23]. La evaluación por simulación considera una muestra menor a 100 nodos; en efecto, en este estudio se menciona que con más de 1000 nodos la red posiblemente se saturaría, además, plantea que cada uno de los vehículos en la red envía información, y que la distancia entre RSU vecinos es de 1000 [m] a 1500 [m] como máximo. Finalmente, las conclusiones de las pruebas son limitadas ya que la muestra es de apenas 3 vehículos, en este caso se logra alcanzar una distancia de 700 [m] con una velocidad de 3 [Mbps] y un PIRE[9] de 15,5 [dBm], además se recomienda que la altura del RSU sea mayor a la antena más alta de los vehículos en la red [23].

1.5 BREVE DESCRIPCIÓN DE LOS SIMULADORES EMPLEADOS

La simulación de una VANET requiere utilizar al menos un simulador de tráfico vehicular, un simulador de red y un vínculo entre ambos.

En el presente trabajo se ha seleccionado a SUMO como simulador del tráfico vehicular, OMNeT++ para emular a la red de comunicaciones y veins como el vínculo entre los dos programas iniciales. Para asegurar la compatibilidad de los simuladores se han seleccionado las versiones SUMO v0.25, OMNeT++ v5.0 y veins v4.2; por lo cual, cada uno de los simuladores se describe a continuación.

1.5.1 SUMO

Sumo es un simulador de movilidad urbana desarrollado en lenguaje C++, que permite crear e importar mapas, y se fundamenta en la identificación de nodos con un número que representan la latitud y longitud bajo el sistema de coordenadas WGS84[10]. Los nodos se unen mediante aristas que simbolizan las calles, las cuales se usan a posteriori para fijar el inicio, puntos intermedios y fin de un trayecto; puesto que, los vehículos programados toman una ruta determinada con el modelo de coche seguidor[11], se emplea la numeración de aristas secuenciales en el mapa para crear un camino según se desee.

---

[9] PIRE: Es la Potencia Isotrópica Equivalente Radiada de una antena.
[10] WGS84: es el Sistema Geodésico Mundial de 1984, es un sistema estándar a nivel mundial para cartografía, geoposicionamiento y navegación
[11] Coche seguidor: es un robot móvil que sigue una trayectoria regida por una línea recta

El generador de movilidad presenta una interfaz amigable, es decir, que cuenta con herramientas para el monitoreo y el control del tiempo de ejecución; por ejemplo, incluye una zona especial para generar un retardo entre 0 y 100 [ms], lo que facilita visualizar con mayor detalle el movimiento de los vehículos en las rutas establecidas. El comportamiento particular de un vehículo en el simulador, se puede observar mediante la opción de acercamiento sin necesidad de interrumpir la reproducción, además, cuenta con un sistema de múltiples ventanas que permiten visualizar al mismo tiempo el mapa general y un área seleccionada, como se observa en la Figura 1.16.

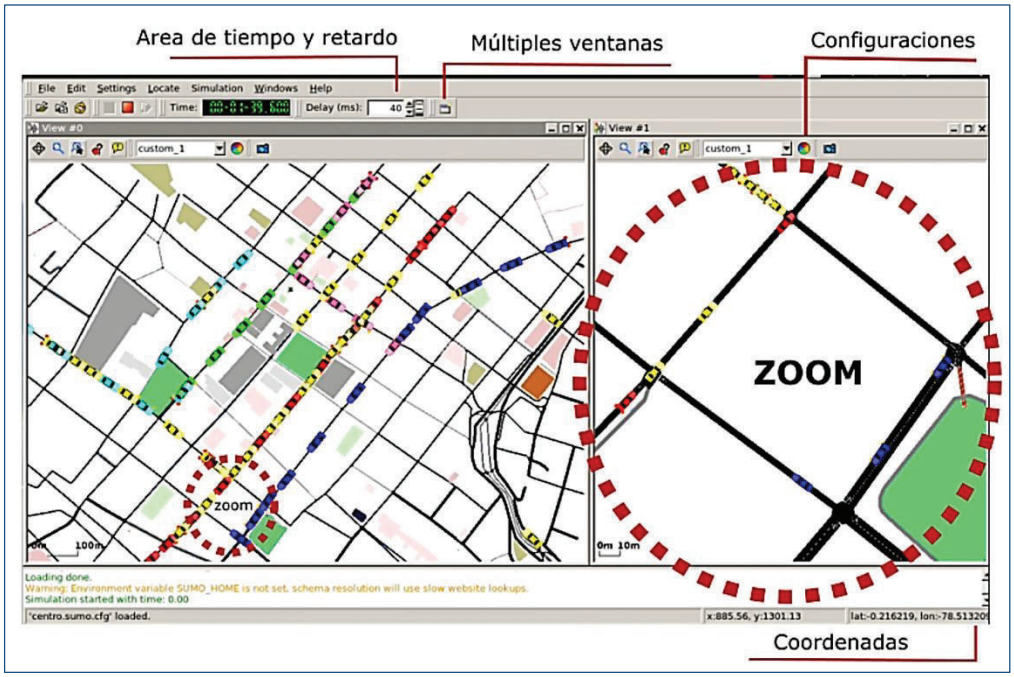


Figura 1.16- Interfaz y herramientas de SUMO

La ventana de configuración visual de SUMO permite variar parámetros como: fondo de pantalla, calles, número de vehículos, nodos, semáforos, puntos poligonizados (POIs), etc; como se muestra en la Figura 1.17. En particular, los más utilizados son las variaciones de vehículos y POIs. En consecuencia, las variaciones que se pueden realizar a los vehículos son: el tamaño en que se puede visualizar el número del nodo móvil, el color de los vehículos y la opción de mantener el tamaño del vehículo constante aún al aplicar enfoque o desenfoque.

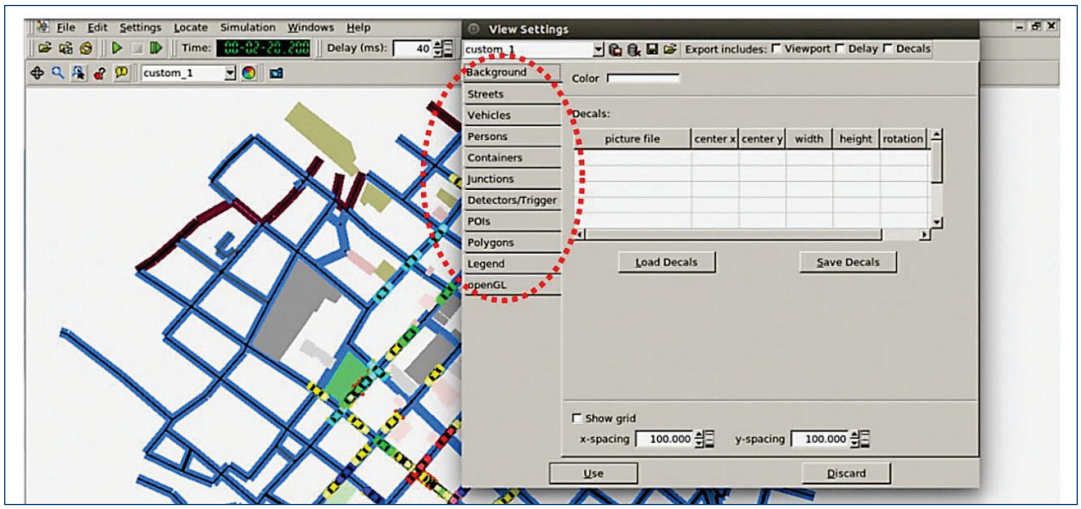


Figura 1.17- Ventana de configuración en SUMO

La información extra en el mapa como la ubicación de equipos RSU (*Road-Side Unit*) o sitios destacados de la ciudad; se puede añadir desde la sección POIs de puntos poligonizados, que ofrece la opción para visualizar nombres, identificativos y variar el color de los dispositivos fijos.

La instalación de SUMO puede realizarse desde el centro de administración de Ubuntu o mediante comandos, sin embargo, con estos métodos solamente se instalará la última versión de SUMO (v0.28). No obstante, por temas de compatibilidad con veins, es necesario trabajar con SUMO v0.25, por consiguiente, se presentan los comandos necesarios para la instalación manual y la solución de posibles errores generados.

- Descargar el archivo **sumo-src-0.25.0.zip** de la página: *https://sourceforge.net/projects/sumo/files/sumo/version%200.25.0/*
- Crear una carpeta para instalar SUMO, copiar y descomprimir el archivo .zip. Recordar que se necesita permisos de administrador:

```
$ sudo su
$ mkdir /home/sumowork
```

- Dirigirse a la carpeta creada para SUMO:

```
$ cd /home/sumowork/sumo-0.25.0
```

- Ejecutar los siguientes comandos para configurar e instalar SUMO, se debe esperar que concluya cada proceso.

```
$ ./configure
```

```
$ make
$ make install
```

- En caso de encontrar errores en el proceso de instalación es probable que la actualización del sistema operativo deseche paquetes antiguos necesarios para el simulador, para solucionarlo se ejecuta:

```
$ checkinstall
```

  Con esta instrucción se genera una versión .deb del instalador y se instalan los paquetes necesarios.

- Ejecutar el instalador .deb:

```
$ dpkg -i sumo-0.25.0.deb
```

- Finalmente, desde el terminal comprobar la instalación.

```
$ sumo –version
```

- Para ejecutar SUMO, debe ingresar la siguiente línea de comandos:

```
$ sumo –gui –c centro.sumo.cfg
```

1.5.2 OMNeT++

OMNeT++ es un simulador de redes de eventos discretos, abierto al público desde 1997, con librerías y módulos que usan conceptos de programación orientada a objetos en C++; Trabaja con módulos simples para formar módulos complejos, así pues, las topologías se simulan con un lenguaje propio de OMNET llamado NED. El simulador maneja 3 parámetros importantes: a) taza de error, b) tiempo de propagación y c) taza de transferencia de información [11].

El *framework* encargado de los diferentes protocolos de comunicación para la simulación de redes en OMNeT++ en las capas superiores de redes inalámbricas es INET; por otro lado, MIXIM es un *framework* que se encarga de la interfaz con INET además del modelado de las capas de enlace y física. Conviene subrayar, que al integrar la versión 5.0 de OMNeT++ con veins v4.2, los *frameworks* INET y MIXIM se instalan por defecto.

En la Figura 1.18 se presenta el entorno de OMNeT++ v5.0 que se describe en la guía de usuario del ANEXO B y que cuenta con los elementos necesarios para compilar un proyecto NED; En la ventana de exploración de proyectos (ubicada a la izquierda), se puede observar la carpeta veins en la que se encuentran los

archivos SUMO y algunos otros vinculados a la VANET, asimismo, el archivo **omnetpp.ini** que contiene los datos de configuración de la red.

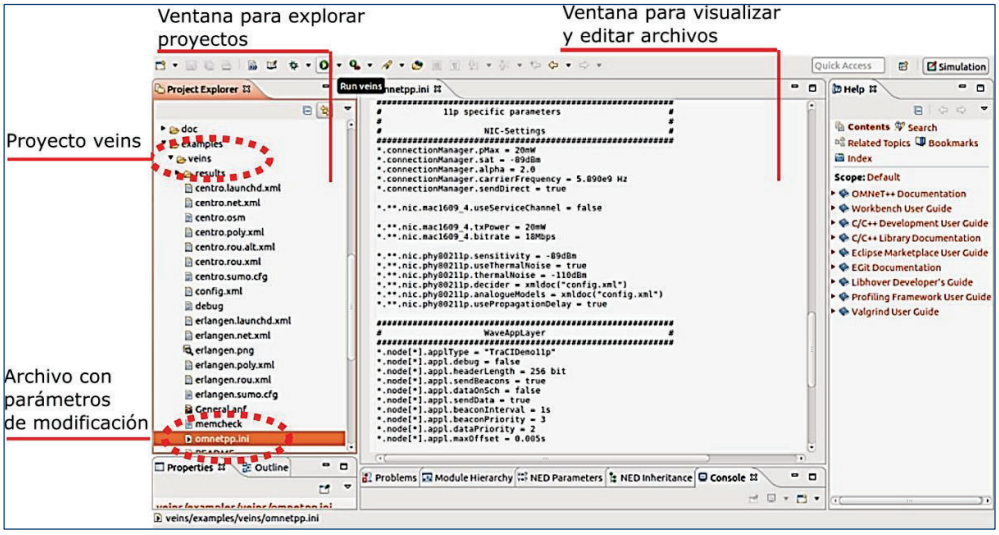


Figura 1.18- Entorno de OMNeT++ 5.0

Los requerimientos y comandos necesarios para la instalación de OMNeT++ v5.0 en el sistema operativo Ubuntu v16.04 se presentan a continuación:

- Instalar varios paquetes necesarios para el funcionamiento de OMNeT++, con el siguiente comando:

```
$ build-essential, gcc, g++, bison, flex, perl, tcl-dev, tk-dev,
blt, libxml2-dev, zlib1g-dev, openjdk-6-jre, doxygen, graphviz,
openmpi-bin, libopenmpi-dev, libpcap-dev
```

- Descargar el instalador de OMNeT++ v5.0 de la página oficial *https://omnetpp.org/omnetpp*
- Desde un terminal, crear una carpeta para la instalación del programa

```
$ mkdir omnetpp
```

- En la carpeta omnetpp creada, copiar el archivo descargado

```
$ cp omnetpp-5.0-src.tgz ../omnetpp
```

- Descomprimir el instalador

```
$ tar xvfz omnetpp-5.0-src.tgz
```

- En la carpeta de instalación descomprimida **omnetpp-5.0,** ejecutar el comando para definir las variables de entorno:

```
$ setenv
```

- Editar el archivo **bashrc** e incluir la siguiente línea de comandos

```
gedit ~/.bashrc
export PATH=$PATH:$HOME/omnetpp/omnetpp-4.6/bin
```

- Ejecutar el archivo de configuración de la instalación

```
$ ./configure
```

- En el directorio omnetpp/omnetpp-5.0, ejecutar el comando make

```
make MODE=release
```

Además, es necesario instalar algunos componentes para la interfaz gráfica de Ubuntu; los detalles de instalación para diferentes sistemas operativos se encuentran en el manual del programa de la página oficial de OMNeT++.

1.5.3 VEINS

Veins es un simulador de tipo híbrido, de código abierto y el vínculo entre SUMO y OMNeT++, fue desarrollado por el grupo de trabajo de Christoph Sommer en la Universidad de Erlangen-Nuremberg (Alemania). El presente proyecto emplea veins v4.2 que no requiere instalar por separado el *framework* MIXIM, ya que el vínculo de instalación se encuentra inmerso en la carpeta veins. La relación que mantienen los simuladores se muestra en la Figura 1.19.

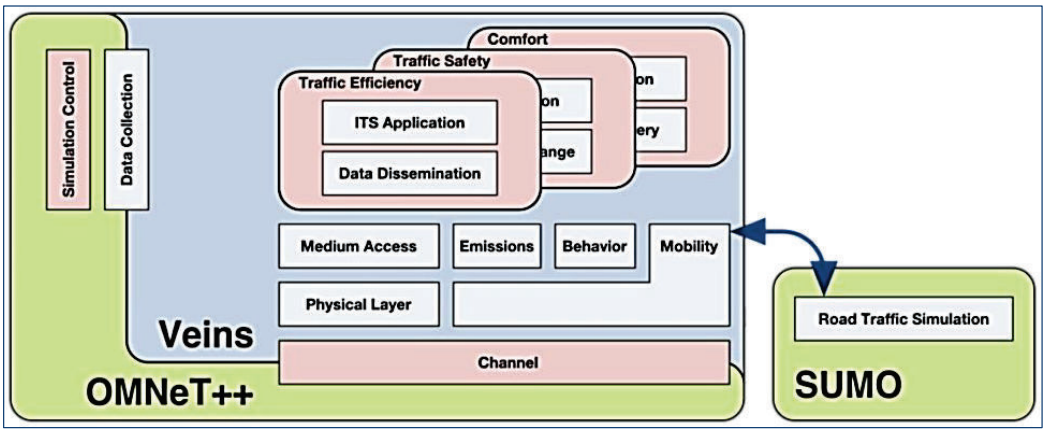


Figura 1.19- Relación de los componentes de veins [26]

Una VANET puede simularse a partir del ejemplo que veins incluye por defecto, para esto se substituyen los 8 archivos correspondientes a la movilidad de SUMO (ver Figura 2.3), y se edita uno extra (**centro.launchd.xml**) que se encuentra en

el ejemplo de veins e incluye la información para crear el vínculo con OMNeT++, tal como se indica a continuación:

```
<!-- debug config -->
<launch><copy file="centro.net.xml"/>
<copy file="centro.rou.xml"/>
<copy file="centro.poly.xml"/><copy file="centro.sumo.cfg"
type="config"/>
</launch>
```

El siguiente paso, es modificar el tiempo de SUMO para acoplarlo a veins, esto se logra al incluir en el archivo **centro.sumo.cfg** el tiempo de veins, que tiene la siguiente forma:

```
<time>
<begin value="0"/>
<end value="1000"/>
<step-length value="0.1"/>
</time>
```

Para compilar el proyecto, en la carpeta de ejemplos de veins, es necesario editar el archivo **omnetpp.ini** en el campo de *TraCI Scenario Manager parameters*, con el nombre del archivo generado para el proyecto (**centro.launchd.xml**), tal como se muestra en la siguiente línea:

```
*.manager.launchConfig = xmldoc("centro.launchd.xml")
```

Los archivos de veins que se modificaron para este proyecto se pueden descargar de la plataforma GibHub en el enlace *https://goo.gl/zYkpvV*. Asimismo, el archivo **omnetpp.ini** se adjunta en el repositorio, que además incluye secciones para configurar la interfaz con el usuario al activar o desactivar diferentes funciones y parámetros; las secciones más importantes de **omnetpp.ini** son:

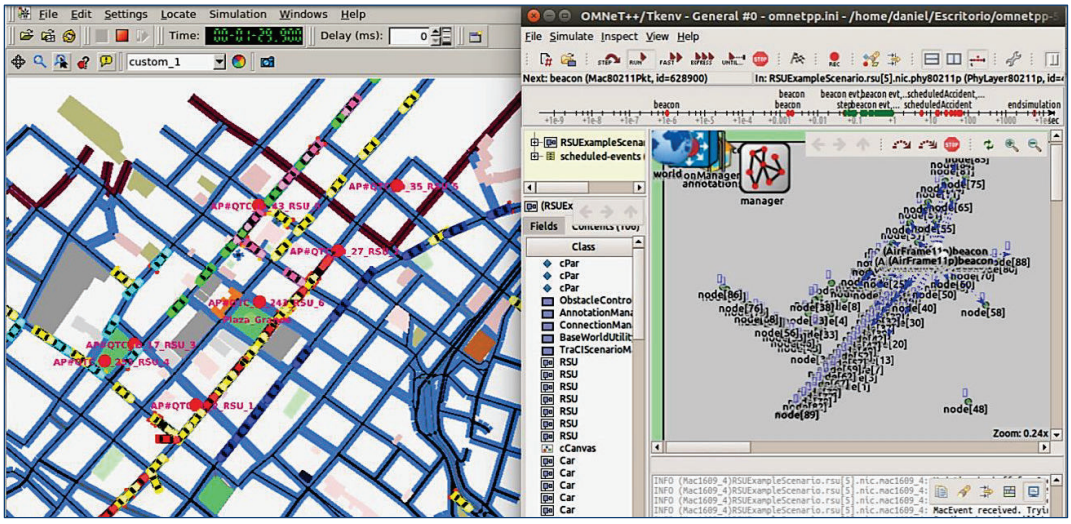
- *TraCIScenarioManager parameters:* Que administra el archivo de carga launchConfig y activa el puerto 9999 para realizar el vínculo con SUMO.
- *RSU SETTINGS:* Para configurar el número de unidades fijas (RSU) en carretera y sus posiciones en el display.
- *11p specific parameters and NIC-Settings:* Permite administrar parámetros como: potencia, frecuencia portadora, velocidad de trama, sensibilidad tanto para capa física como para subcapa MAC como se muestra a continuación:

```
*.connectionManager.pMax = 20mW
*.connectionManager.sat = -89dBm
*.connectionManager.alpha = 2.0
*.connectionManager.carrierFrequency = 5.890e9 Hz
```

```
*.connectionManager.sendDirect = true
*.**.nic.mac1609_4.useServiceChannel = false
*.**.nic.mac1609_4.txPower = 20mW
*.**.nic.mac1609_4.bitrate = 18Mbps
*.**.nic.phy80211p.sensitivity = -89dBm
*.**.nic.phy80211p.useThermalNoise = true
*.**.nic.phy80211p.thermalNoise = -110dBm
```

- *Mobility:* Para configurar la generación de accidentes

La ventana de veins se abre al compilar y ejecutar la carpeta del proyecto en la interfaz de OMNeT++. En la Figura 1.20 se presenta el entorno de simulación de veins, así pues, en la Figura 1.20a se muestra el entorno de movilidad mientras que en la Figura 1.20b se distingue la simulación de la red de datos con diferentes opciones para el control del tiempo de reproducción, así como los botones de inicio y parada, una opción de reproducción rápida en la que se omite la visualización del intercambio de paquetes entre nodos y un botón para mayor velocidad en el que la prioridad es observar la movilidad vehicular, además se dispone de una opción para reproducir la simulación hasta un evento determinado. Todos los eventos de la simulación se pueden observar en una línea de tiempo ubicada en la parte superior del interfaz.



a) Ventana de movilidad b) Ventana de comunicación de la red

Figura 1.20- Entorno de simulación de Veins

Una vez instalados los simuladores SUMO v0.25 y OMNeT++ v5.0, es necesario descargar el paquete de instalación de veins 4.0 o superior de su página oficial [26]; a continuación, el paquete se debe descomprimir en la carpeta de instalación

del simulador de red. Una vez realizado esto, se abre el simulador OMNeT++ y en su interfaz gráfico, secuencialmente se seleccionan las siguientes opciones: *File* – *Import* – *General* – *Exiting Project Into Workspace*. Finalmente se escoge el archivo de veins que contiene, a partir de la versión 4.0, a los paquetes necesarios de INET y MIXIM que se importan y compilan automáticamente.

CAPÍTULO 2: DISEÑO, SIMULACIÓN E IMPLEMENTACIÓN

En este capítulo se realiza el diseño de la VANET para el centro histórico de la ciudad de Quito. El diseño requiere consideraciones por separado, para la movilidad y la comunicación de la red. Se debe iniciar con el diseño de movilidad, para lo cual se considera una intensidad de tránsito vehicular lo más cercano a la situación real de la ciudad, luego se diseña la red de comunicación, se instalan las unidades fijas de transmisión (RSU) en el área geográfica y se implementa la red VANET para determinar el comportamiento y el rendimiento de la misma.

2.1 DISEÑO

2.1.1 REQUISITOS

En el diseño se seleccionó una zona del centro histórico de la ciudad de Quito con un perímetro aproximado de 2,4 [km]; esta dimensión resulta ser suficiente para trabajar sin comprometer el rendimiento de los simuladores, pero es lo bastante amplia para colocar un número considerable de RSUs alrededor de las vías y de los vehículos, para de esta manera simular alto tráfico y permitir a la VANET aplicar soluciones como la generación de rutas alternas. El área seleccionada está delimitada por las calles: Vicente Rocafuerte, Cuenca, Esmeraldas y Guayaquil, tal como se indica en la Figura 2.1.

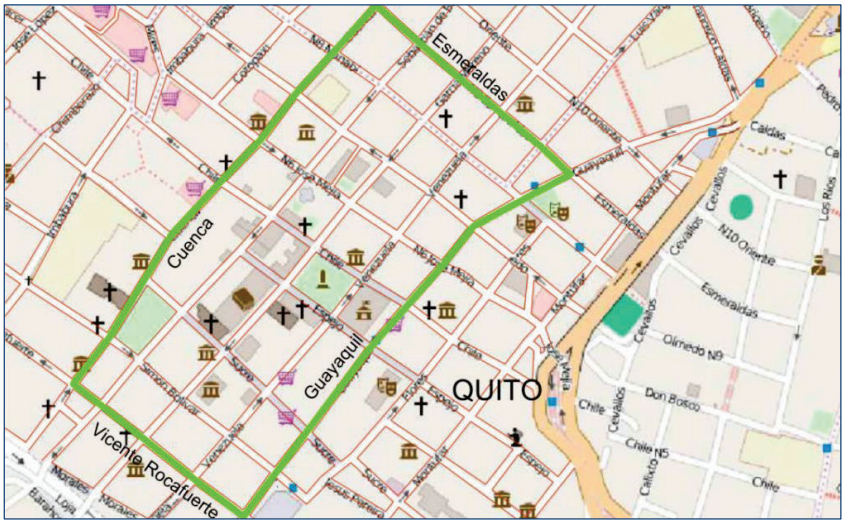


Figura 2.1- Área de análisis del centro histórico de Quito [27]

Los medios empleados para el diseño involucran hardware, software y recursos de información. Estos medios se resumen en la siguiente lista y más adelante se describe el uso de los mismos.

- Computador personal: Intel Core i5, 2.50 [GHz] y RAM de 6 [GB].
- Sistemas operativos: Windows 10 para AutoCAD y Ubuntu 16.04 para la simulación.
- Programas: SUMO v0.25, OMNeT++ v5.0 (con los *frameworks* INET, MiXiM y veins), OpenStreetMap, Google Map y AutoCAD 2D.

2.1.2 DISEÑO PARA SIMULACIÓN DE MOVILIDAD

La intensidad del tráfico de Quito, en horas pico, se obtuvo desde Google Map [28], el cual emplea un código de intensidad de calor para describir el tráfico en vivo, cuando existe congestión vehicular las vías se pintan de rojo y cuando existe un movimiento rápido se pintan de verde, tal como se ve en la Figura 2.2.

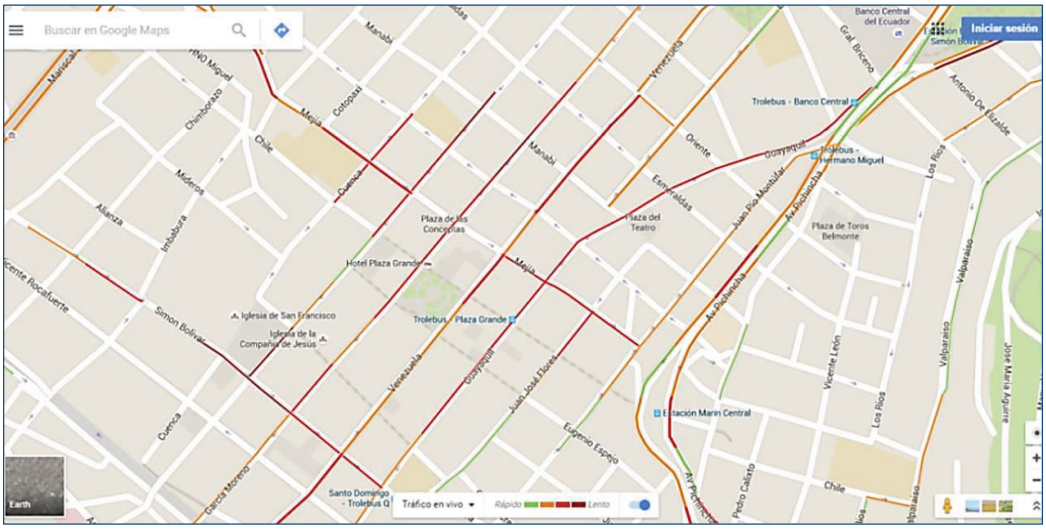


Figura 2.2- Intensidad del tráfico vehicular en el centro histórico de Quito [28]

Para crear el tránsito vehicular, SUMO permite importar mapas en lugar de generarlos desde cero; en particular, se utiliza OpenStreetMap[12] porque maneja información pública y es *open source*. Para importar el mapa, se ubica el área requerida en OpenStreetMap y se selecciona la opción exportar, con lo que se

[12] OpenStreetMap: Proyecto lanzado en 2004 para crear mapas virtuales sin licencia apoyado por colaboradores [27]

descargará automáticamente un archivo de extensión **.osm** que debe ubicarse en una carpeta de SUMO (*home/sumowork*), este archivo se transforma a la base de red para SUMO con el nombre de **centro.net.xml**, mediante la siguiente línea de comandos:

```
$ netconvert --osm-files centro.osm -o centro.net.xml
```

El siguiente paso es crear el archivo **typemap.xml** que contiene la descripción general de elementos típicos encontrados en un mapa de movilidad como: parques, edificios importantes, puntos turísticos, zonas con agua, parqueaderos, etc. Toda ésta información, incluye a los identificadores (id) que se visualizan en el mapa como un texto descriptivo para cada estructura, los nombres en inglés de los elementos en el mapa son: *water, sport, forest, natural, park, leisure, tourism, landuse, residential, commercial, industrial, shop, building, amenity, parking, military, farm, power, land, boundary.administrative, aeroway, aerialway, historic*.

Lo siguiente para importar el mapa, es describir los colores para cada elemento utilizando el código de colores RGB (*Red, Green, Blue*) y finalmente se especifican las capas de cada elemento para identificar el sobrelapamiento entre ellos; todo esto, se introduce gracias a las varias líneas de código del archivo **typemap.xml** que son un aporte de los colaboradores que experimentan con SUMO, el archivo tiene la estructura que se muestran a continuación:

```
<polygonTypes>
<polygonType id=”park” name=”park” color=”.81,.96,.79” layer=”-11”/>
<polygonTypes>
```

El archivo **centro.poly.xml** describe mediante polígonos y puntos cada uno de los elementos referenciales que existen en el entorno, gracias a información de OpenStreetMap actualizada por usuarios o colaboradores mediante GPS (*Global Positioning System*), dichas características fueron exportadas junto con el mapa **centro.osm**. El archivo **centro.poly.xml** se genera en una ventana terminal con los archivos previos de carga **centro.osm**, **centro.net.xml** y **typemap.xml**, para lo cual se ejecuta un convertidor a polígonos con la siguiente línea de código:

```
$ polyconvert --net-file centro.net.xml --osm-files centro.osm --type-
file typemap.xml -o centro.poly.xml
```

A continuación, se debe ubicar el archivo **Trips.py** (archivo de Python), copiar el archivo y su dirección, usualmente la ruta por defecto es

/usr/share/sumo/tolos/randomTrips.py, en la carpeta creada para los ficheros de SUMO, y editar para crear al menos 100 vehículos con rutas aleatorias en las calles, este procedimiento se lo realiza de la siguiente manera:

```
$ python /usr/share/sumo/tools/ randomTrips.py -n centro.net.xml -e 100 -
l
```

Luego, se debe crear un archivo con las rutas aleatorias que seguirán los vehículos mediante la siguiente línea de comandos:

```
$ python /usr/share/sumo/tools/randomTrips.py -n centro.net.xml -r
centro.rou.xml -e 100 -l
```

Todos los documentos necesarios para ejecutar el simulador de movilidad urbana se pueden descargar de la plataforma GitHub en el siguiente enlace *https://goo.gl/2zy24K*; la Figura 2.3 resume todos los archivos mencionados. Para evitar conflictos por paquetes y actualizaciones de la interfaz gráfica, se sugiere correr el simulador desde una pantalla terminal, ubicándose en la dirección del proyecto generado, con la siguiente línea de comandos:

```
$ sumo-gui -c centro.sumo.cfg
```

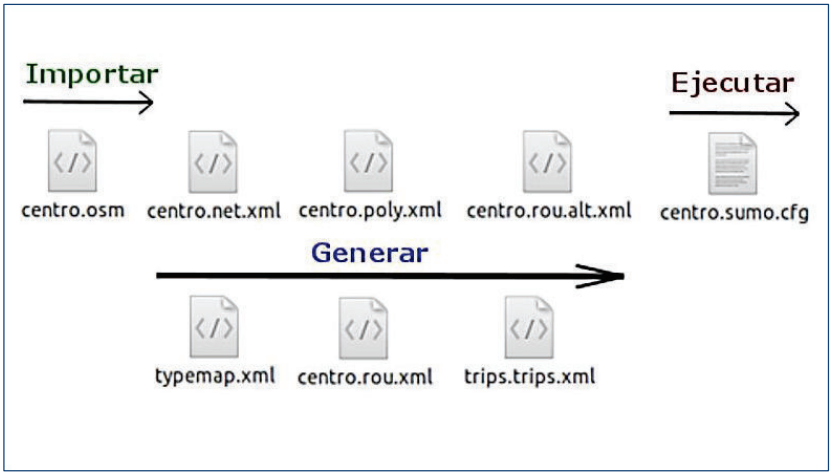


Figura 2.3- Archivos generados para SUMO

Las rutas aleatorias creadas para 100 vehículos son de gran ayuda para comprobar el buen funcionamiento del simulador en un inicio, sin embargo para emular el comportamiento de los vehículos en condiciones de tráfico reales como se muestra en la Figura 2.2, es necesario observar las calles principales con intensidad de tráfico y generar una carga de vehículos razonable para realizar el estudio. La Tabla 2.1 describe con código de colores empleado en el uso de 15

rutas, de las cuales 9 son programadas para recorrer las calles principales, es decir, se consideran estos trayectos como rutas típicas porque los usuarios las utilizan mayormente mientras que las rutas restantes son aleatorias o de bajo uso.

La programación del camino que siguen los vehículos puede realizarse de manera independiente para cada vehículo o para un grupo de vehículos. Para distribuir los números de identificación de los 100 vehículos empleados, identificados del 0 al 99, se emplea un triángulo mágico invertido, como se muestra en la Tabla 2.1. Con esta distribución se consigue que el número identificador no se repita y que los vehículos sean distribuidos en las 15 rutas, generando tráfico vehicular en toda el área de análisis; adicionalmente, cabe indicar que se omite el uso de semáforos para recrear la peor condición de congestión vehicular.

Tabla 2.1 - Porcentaje, color e identificador de vehículos en 15 rutas

Ruta	Color	Porcentaje	Identificador de vehículo													
0		14%	0	2	6	12	20	30	40	50	60	70	80	88	94	98
1		14%	1	3	7	13	21	31	41	51	61	71	81	89	95	99
2		12%		4	8	14	22	32	42	52	62	72	82	90	96	
3		12%		5	9	15	23	33	43	53	63	73	83	91	97	
4		10%			10	16	24	34	44	54	64	74	84	92		
5		10%			11	17	25	35	45	55	65	75	85	93		
6		8%				18	26	36	46	56	66	76	86			
7		8%				19	27	37	47	57	67	77	87			
8		6%					28	38	48	58	68	78				
Aleatoria		6%					29	39	49	59	69	79				

Como se puede ver en la Tabla 5, el 14% de los móviles se distribuyen en la ruta 0, es decir, 14 vehículos pintados de rojo seguirán únicamente la calle Venezuela. Para lograr este propósito es necesario editar la ruta de cada uno de los vehículos programados de acuerdo al identificador de aristas que corresponden a los bloques que son parte de esta vía, tal como se puede observar en el código de la Figura 2.4a, 13 números distintos identifican a las aristas que conforman la calle Venezuela.

El simulador de movilidad maneja la idea del coche seguidor, en consecuencia, todos los vehículos que lleven la secuencia de aristas que trazan una ruta determinada, deberán tomar el mismo camino sin desviarse como se muestra en la Figura 2.4b. De igual manera, se planifica el trayecto de las rutas restantes, con

sus correspondientes calles, líneas de programación y colores con los que serán identificados en la simulación de movilidad.

a) Programación de la ruta b) Plano georreferenciado de la ruta 0

Figura 2.4- Ruta 0, programada para vehículos rojos

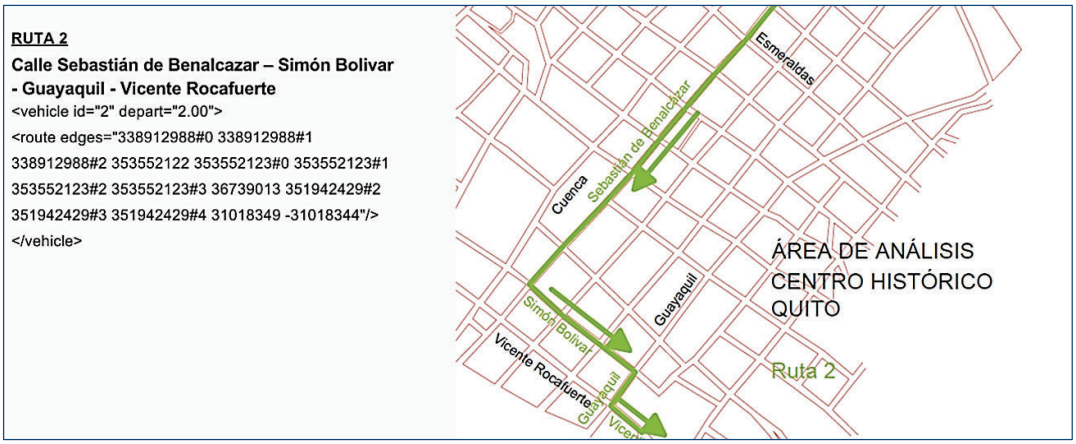
En la Figura 2.5a se observan los nombres de las vías que son parte de la ruta 1 y la secuencia de códigos de aristas que forman el trayecto, mientras que en la Figura 2.5b se muestra el mapa del área de análisis creado en AutoCAD[13], este mapa será utilizado como referencia para realizar mediciones de distancia y para comparar las rutas programadas con los resultados que arrojará el simulador.

a) Programación de la ruta 1 b) Plano georreferenciado de la ruta 1

Figura 2.5- Ruta 1, programada para vehículos azules

[13] AutoCAD: Es un software de dibujo para realizar diseños en 2D y 3D que contiene herramientas suficientes para desarrollar diferentes esquemas, mapas y planos [33].

La ruta 2 parte en la calle Sebastián de Benalcázar y culmina su trayecto en la calle Vicente Rocafuerte, tiene una carga de 12 vehículos, está identificada con el color verde y se puede observar en la Figura 2.6.



a) Programación de la ruta 2 b) Plano georreferenciado de la ruta 2

Figura 2.6- Ruta 2, programada para vehículos verdes

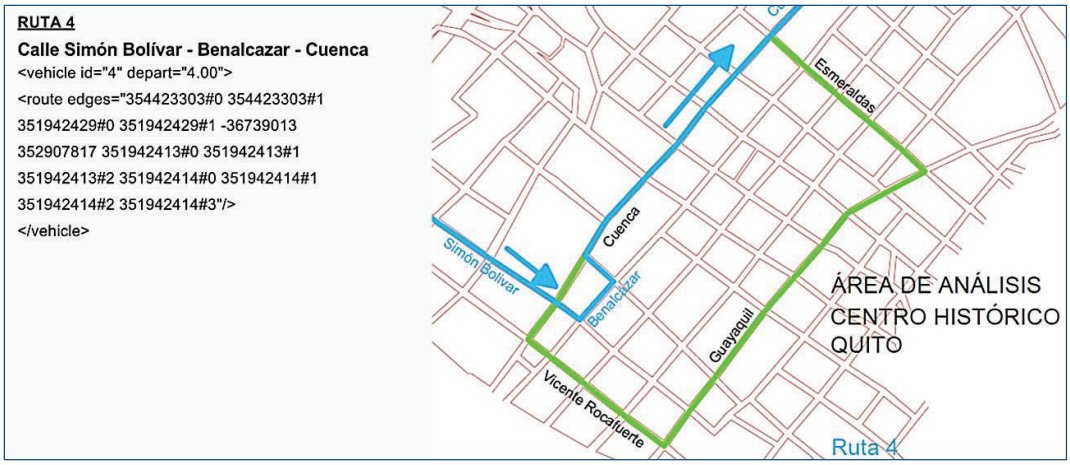
La ruta 3 al igual que la ruta 2, inicia su trayecto en la calle Sebastián de Benalcázar y lo concluye en la calle Vicente Rocafuerte (ver Figura 2.7), sin embargo, las calles intermedias son diferentes, esta variación se realiza para analizar a posteriori las soluciones que se plantean a dos rutas que cuentan con el mismo inicio y fin.

a) Programación de la ruta 3 b) Plano georreferenciado de la ruta 3

Figura 2.7- Ruta 3, programada para vehículos violetas

Al sur de la calle Sebastián de Benalcázar, frente a la iglesia de San Francisco entre las calles Simón Bolívar y Sucre existe un tramo con doble flujo de vehículos

en el que se puede apreciar un cuello de botella de tránsito. La ruta 4 incluye al tramo de doble flujo en su trayecto, ver Figura 2.8b, y le otorga una carga de 10 vehículos para observar el tráfico generado en ésta área y a posteriori la solución que planteará la VANET.



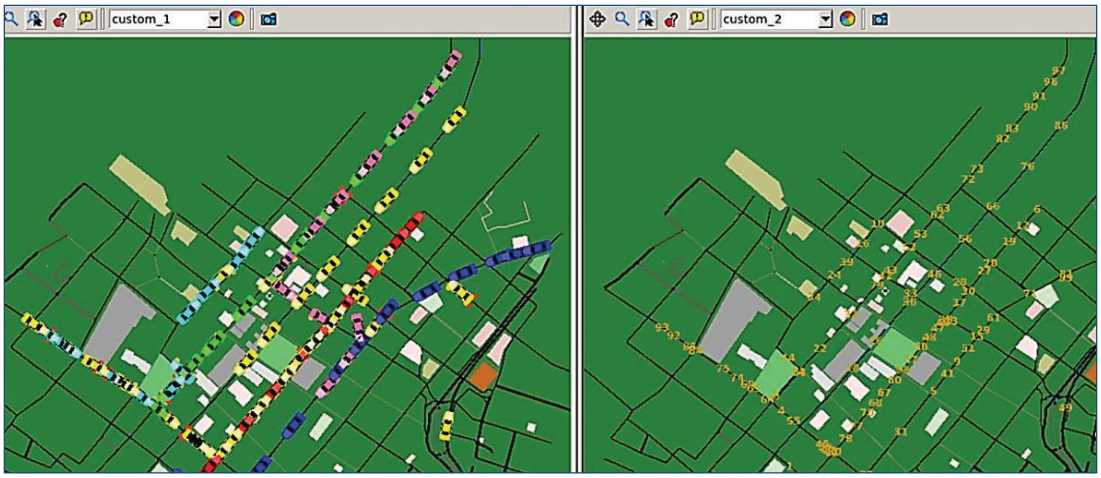
a) Programación de la ruta 4 b) Plano georreferenciado de la ruta 4

Figura 2.8- Ruta 4, programada para vehículos celestes

Como muestran los porcentajes de la Tabla 2.1, el número de vehículos que circulan por cada trayecto decrece a medida que el número de ruta aumenta, por lo cual se resaltan desde la ruta cero a la cuatro con varios colores para identificar su movimiento. Por otro lado, a partir de la ruta cinco el número de vehículos es cada vez menor y se utiliza un color por defecto que puede variarse desde la ventana de configuración de visualizaciones de SUMO. Finalmente, en la fila denominada rutas aleatorias de la Tabla 2.1, se programan 6 rutas distintas que son tomadas por los vehículos 29, 39, 49, 59, 69, 79, éstas rutas simulan por ejemplo, el comportamiento de usuarios que están: perdidos, de visita turística o simplemente quieren dar vueltas en su vehículo.

Una vez creados los 100 vehículos de muestra y las 14 rutas que recorren los mismos, se ejecuta SUMO y se configuran sus parámetros de visualización a nivel macro (visualización de la intensidad de tráfico) y micro (visualización de cada vehículo en el mapa), esto para evitar interpretar accidentes por el hecho de ver coches demasiado unidos. Los eventos que se pretenden generar con el modelo de rutas son: a) largas colas de vehículos en espera, b) bajas velocidades debido

a continuo frenado y arranque y c) un efecto cuello de botella en varias zonas como por ejemplo en la calle Benalcázar de la ruta 3 debido al doble flujo. Todos estos eventos se presentan a nivel macro en la Figura 2.9a, además, la representación numérica de cada vehículo puede observarse en la Figura 2.9b.



a) Identificación por colores b) Identificación por etiqueta

Figura 2.9- Generación de incidentes en la red de movilidad

Los 100 vehículos generados individualmente hasta el momento en el proyecto, cada uno con su número identificador y su respectivo código de ruta cumplen con el triángulo invertido de la Tabla 2.1 y con las expectativas de la simulación; sin embargo, para facilitar la identificación de vehículos en SUMO se modifica el archivo **centro.rou.xml** que contiene por defecto una programación extensa, ya que contiene líneas de código para cada uno de los 100 vehículos. Como una opción más eficiente se utiliza la emisión de vehículos mediante flujo (flow[14]), que consiste en generar 15 rutas programadas, y asociar un número de vehículos determinado a cada ruta (ver Tabla 5). Además, las rutas creadas se identifican con un *route* id que es un número entero y con un color específico. Al utilizar la función *flow se* cambia el identificador de cada vehículo de un número entero a uno de tipo flotante, donde la parte entera describe la ruta y la parte decimal inicia el conteo de los vehículos, es decir, se podría iniciar la ruta 0 desde 0.00, 0.01 o según se requiera. Las modificaciones realizadas en el archivo **centro.rou.xml** de SUMO para el caso de la ruta 0 con 14 vehículos se muestran a continuación:

[14] Flow: Es una función en el entorno de SUMO para emisiones repetidas de vehículos en serie [5]

```
<route id="route0" color="1,0,0" edges="104750584 420917636 420917638
104750573#0 104750573#1 104750573#2 104750532#0 104750532#1 104750532#2
409656671#0 409656671#1 421482710#0 421482710#1"/>
<vType id="passenger/sedan" guiShape="passenger/sedan" accel="2.0"
decel="4.0" sigma="0.5" length="2.5" minGap="3.0" maxSpeed="50" >
<carFollowing-IDM/>
</vType>
<flow id="Ruta0" type="passenger/sedan" route="route0" begin="0.00"
period="3" number="14"/>
</routes>
```

En la Figura 2.10a se observa la identificación por colores de las rutas planificadas, mientras que en la Figura 2.10b se muestra una mejora significativa en la identificación de los vehículos en el mapa, ya que en lugar de visualizar los números de la Tabla 2.1, ahora, directamente se observa el nuevo esquema de identificación; por ejemplo, el identificador Ruta0.00, indica que es el primer vehículo que circula por la ruta 0 y la etiqueta Ruta0.13 representa al último vehículo en el mismo trayecto.

a) Identificación por colores b) Identificación por etiqueta

Figura 2.10- Mejora en la identificación de vehículos

2.1.3 DISEÑO PARA LA RED DE COMUNICACIÓN

En la simulación de la VANET se ha considerado una comunicación V2I (*Vehicle to Insfrastructure*). Por lo tanto, para crear la red vehicular bastaría con dos nodos móviles que cuenten con unidades a bordo OBU (*On-Board Unit*) y que se encuentre en un rango de alcance común. Sin embargo, para trabajar con los 100 vehículos, es necesario colocar RSUs en sitios estratégicos para comunicar zonas lejanas y extender las redes ad-hoc formadas. En la VANET cada vehículo que ingresa al sector simulado, inmediatamente reporta su presencia y los eventos que se presentan en su trayecto, además, informa las condiciones de tráfico que reportan los nodos vecinos.

Para establecer las posiciones de las RSUs y además, para que la simulación sea más realista, se tomaron como referencias las ubicaciones de los AP (*Access Point*) utilizados en el proyecto de #QuitoTeConecta (servicio gratuito Wi-Fi del Distrito Metropolitano de Quito [29]), tal como se muestra en la Tabla 2.2. En efecto, el área de operación del proyecto #QuitoTeConecta en el centro histórico de Quito coincide con la zona establecida para el diseño de la VANET; además, la información del proyecto es abierta al público, cada AP se encuentra geoferenciado y también dispone de los siguientes datos:

- Número de RSU en la simulación
- Número identificador en la red #QuitoTeConecta
- La ubicación con calles en el mapa
- Las coordenadas geográficas con latitud y longitud

El archivo **Area.dwg** fue creado en AutoCAD para utilizarlo como apoyo en la selección de los RSUs. El proceso inicia partir de una copia de pantalla de la zona de análisis obtenida a través de OpenStreetMap que se usa para realizar un *site survey* de la red #QuitoTeConecta. En AutoCAD se importa la copia de pantalla, y se crean diferentes líneas y capas sobre el dibujo para desarrollar un plano realista de la zona de análisis, y con la información de georeferencias de la Tabla 6, se escala el mapa y se incluyen los APs con su correspondiente información geográfica. A posteriori, se realiza un mapeo con el plano generado en AutoCAD en el entorno gráfico de SUMO, aprovechando que las coordenadas x e y coinciden en ambos programas.

Tabla 2.2- Ubicación de dispositivos RSU

RSU	AP ID	UBICACIÓN	LAT/LONG	SUMO x,y	OMNET
0	43	Plaza Benalcazar, Benalcazar y Olmedo	**-0.2177251, -78.5119647**	**1027.82,1134.37**	**1040.28,600.88**
1	29	Museo Casa Sucre, Sucre y Venezuela	**-0.2218503, -78.5132253**	**883.67,678.09**	**891.30,1052.56**
2	27	Venezuela y Olmedo	**-0.2186639, -78.5102991**	**1209.65,1030.50**	**1236.69,715.13**
3	17	Plaza San Francisco	**-0.2206594, -78.5144967**	**742.29,817.03**	**765.48,915.97**
4	35	Museo Camilo Egas, Venezuela y Esmeraldas	**-0.2173174, -78.5090894**	**1344.94,1179.14**	**1380.53,575.58**
5	243	Chile, entre García Moreno y Venezuela	**-0.219723, -78.511905**	**1030.08,913.61**	**1066.73,826.51**
6	258	Plaza San Francisco	-0.220943, -78.515096	675.08,778.60	NO UTILIZADO
7	256	Chile y Benalcazar	-0.218578, -78.5139415	934.14,988.69	NO UTILIZADO
8	152	Trolebus Plaza Grande, Chile y Guayaquil	-0.2204609, -78.5107952	1151.61,845.50	NO UTILIZADO
9	248	Centro Cultural Metropolitano, García Moreno y Espejo	-0.220188, -78.512844	921.80,864.03	NO UTILIZADO
10	247	Espejo, entre Venezuela y Guayaquil	-0.221011, -78.511657	1046.26,771.53	NO UTILIZADO
11	28	Museo Alberto Camaño, Sucre, entre Benalcazar y García Moreno	-0.2210886, -78.5139468	788.95,753.25	NO UTILIZADO
12	9	Fundación Mariana de Jesus, Simón Bolivar y García Moreno	-0.2219683, -78.5141694	778.83,662.05	NO UTILIZADO
13	34	Plaza Santa Clara, Vicente Rocafuerte y Cuenca	-0.2220435, -78.5160148	569.49,671.14	NO UTILIZADO
14	150	Trolebus Santo Domingo, Simón Bolivar y Guayaquil	-0.2233523, -78.512705	935.72,504.95	NO UTILIZADO

En el mapa georefenciado se representan los RSUs con un alcance de aproximadamente 300 [m] de radio que se obtuvo mediante experimentación en el simulador veins, los radios se grafican para evitar sobrelapamientos innecesarios por distancias demasiado cortas entre dispositivos, se evita así redundancia de información y un incremento de latencia para los paquetes transmitidos en la red, cada vez que se actualiza la VANET en búsqueda de soluciones para los eventos dados. Como resultado se obtiene el mapa georreferenciado que se muestra en la Figura 2.11, de color rojo se presenta la ubicación de los RSUs seleccionados, en tanto que los RSUs descartados son mostrados en color verde; la imagen del mapa exportado en OpenStreetMap y los archivos de AutoCAD generados, pueden ser descargados del repositorio en *GitHub https://goo.gl/ov7Rv6*.

Figura 2.11- Mapa Georeferenciado del área simulada y alcance de RSUs

Los RSU en SUMO se añaden en el archivo **centro.poly.xml**, al incluir un punto para cada dispositivo fijo con su identificador, nombre y posición según muestra la Tabla 6. Para esta tarea se emplea una instrucción que incluye un número

identificador del punto insertado en el mapa, la leyenda con que se visualizará *AP#QTC_ID_35_RSU_4*, el código de color RGB que se le otorga y su respectiva capa de visualización; como se indica a continuación:

```
<poi id="589503716" type="AP#QTC_ID_35_RSU_4" color="255,0,0"
layer="4.00" x="1344.94" y="1179.14"/>
```

Por otro lado, para añadir los dispositivos fijos al proyecto veins, en el display que corresponde a la red de datos, se editan los archivos **RSUExampleScenario.ned** y **omnetpp.ini** que se encuentran en las fuentes de veins en el siguiente directorio: *omnetpp-5.0/veins/examples/veins/source*.

En el archivo **RSUExampleScenario.ned** simplemente se modifica el número de RSU que se requiere para reportar en el entorno de trabajo del escenario planteado, este procedimiento se lo realiza de la siguiente manera:

```
network RSUExampleScenario extends Scenario
{
      submodules:
      rsu[6]: RSU {
            @display("p=150,140;b=10,10,oval");
      }
}
```

En el archivo **omnetpp.ini** se modifica la sección *RSU SETTINGS* para reportar las coordenadas de los 6 dispositivos fijos que se utilizarán según la información de la Tabla 2.2, con el formato que se indica a continuación para cada uno:

```
*.rsu[0].mobility.x = 1040.28
*.rsu[0].mobility.y = 600.88
*.rsu[0].mobility.z = 3
```

2.2 SIMULACIÓN E IMPLEMENTACIÓN

A continuación, se presentan las simulaciones de 2 escenarios bajo las mismas condiciones de tráfico, en una misma área: un escenario sin VANET y otro en el que se implementa la red VANET a fin de comparar y analizar el desempeño de la misma.

2.2.1 ESCENARIO 1 MOVILIDAD SIN VANET

En el primer escenario se emplea únicamente SUMO y se emulan las condiciones de tráfico ordinarias del centro histórico de Quito tal como se muestra en la Figura 2.2; además, se emplean los archivos generados en SUMO que se presentan en

la Figura 2.3. Conviene subrayar, que para este escenario se utilizan las rutas y porcentajes de vehículos de la Tabla 2.1 y se emplea la función de SUMO *flow* para generar grupos de vehículos en cada ruta planteada.

Para ejecutar el programa, mediante una ventana terminal se direcciona a la carpeta que contiene el proyecto (*home/sumowork/centro*) y se ejecuta el siguiente comando:

```
$ sumo-gui -c centro.sumo.cfg
```

Una vez generada la pantalla de SUMO, se sugiere seleccionar la opción de mapa *real world* para tener una interfaz más amigable, y dado que el tiempo de simulación por defecto es muy corto y apenas de 1000 [ms], es necesario seleccionar un retardo (*delay)* de 40 [ms] o superior para visualizar el tráfico a una velocidad moderada. Además, en las opciones de configuración de visualización de vehículos, *View sentings – Vehicles – (Show vehicle name or Draw with constant size when zoomed out),* se seleccionan las opciones para visualizar el nombre de los vehículos y/o la opción de dibujarlos con un tamaño constante al variar el zoom, tal como se puede apreciar en la Figura 2.12.

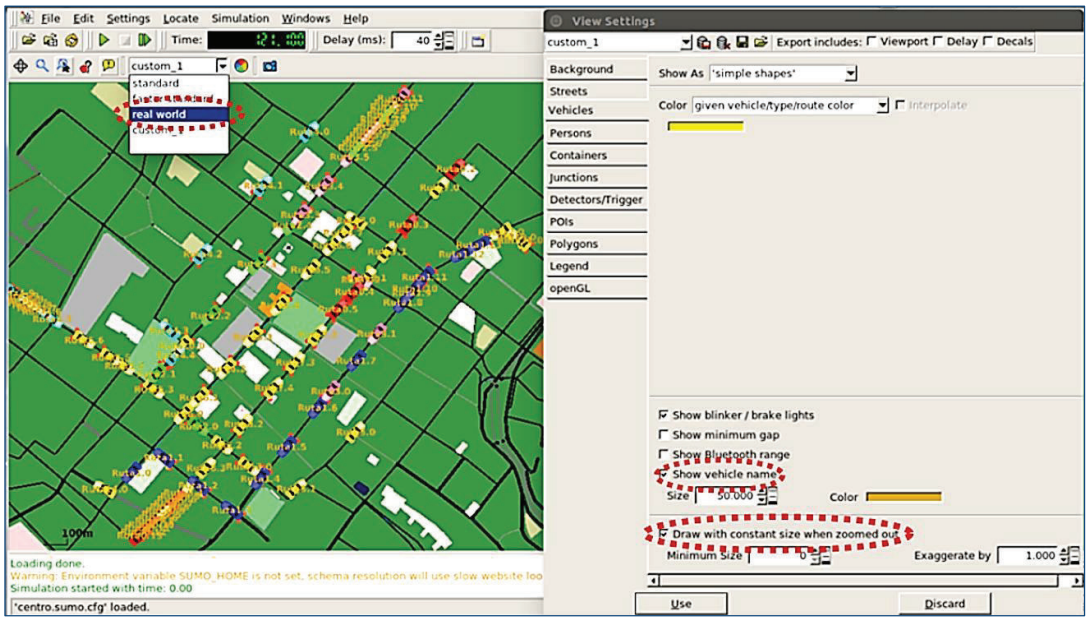


Figura 2.12- Visualización de mundo real en SUMO

2.2.2 ESCENARIO 2 MOVILIDAD CON VANET

En este escenario se plantea la misma movilidad vehicular del escenario uno, más la implementación de una red VANET que incluye a 6 RSU y una OBU por cada vehículo programado. Al ejecutar el simulador OMNeT++ se despliega una ventana que brinda la posibilidad de escoger o generar diferentes espacios de trabajo, para utilizar el simulador veins, se sugiere crear un espacio de trabajo exclusivo y no incluir otras tareas, ya que se pueden generar errores por nombres o archivos duplicados al construir el proyecto. Es importante antes de correr veins, mediante una ventana terminal habilitar en segundo plano el proceso **launchd.py** que activa el puerto 9999 para vincular la comunicación con SUMO desde OMNeT++. Para esto, el usuario debe direccionarse a la carpeta que contiene el proyecto en */home/…/omnetpp-5.0/veins* y ejecutar el siguiente comando que incluye la ruta del archivo **sumo-launchd.py**:

```
$ python /home/…/omnetpp-5.0/veins/sumo-launchd.py -vv -c sumo-gui
```

Para ejecutar veins se activa OMNeT++, se habilita el puerto 9999, se ubica la carpeta veins dentro del proyecto, se compila y se ejecuta el programa. Inmediatamente se despliega la pantalla con los controles de red, posteriormente al iniciar la simulación aparece una pantalla secundaria con SUMO. Los archivos modificados en veins para vincular SUMO, el archivo **centro.launchd.py** así como los cambios necesarios para adicionar los dispositivos fijos RSU pueden descargarse de la plataforma GitHub en el enlace *https://goo.gl/zYkpvV*.

La interfaz para la comunicación de datos cuenta con una ventana para la visualización de eventos, que permite censar todo el historial de la red y filtrar el contenido de ser necesario; ésta ventana en inicio reporta la posición de los 6 dispositivos RSU (ver Figura 2.13) e inicializa los módulos de escenario para cada uno, incluyendo la capa física y subcapa mac con las siguientes líneas de reporte:

```
RSUExampleScenario.rsu[0].appl: Initializing module
RSUExampleScenario.rsu[0].appl, stage 1
RSUExampleScenario.rsu[0].nic.phy80211p: Initializing module
RSUExampleScenario.rsu[0].nic.phy80211p, stage 1
RSUExampleScenario.rsu[0].nic.mac1609_4: Initializing module
RSUExampleScenario.rsu[0].nic.mac1609_4, stage 1
RSUExampleScenario.rsu[0].mobility: Initializing module
RSUExampleScenario.rsu[0].mobility, stage 1
```

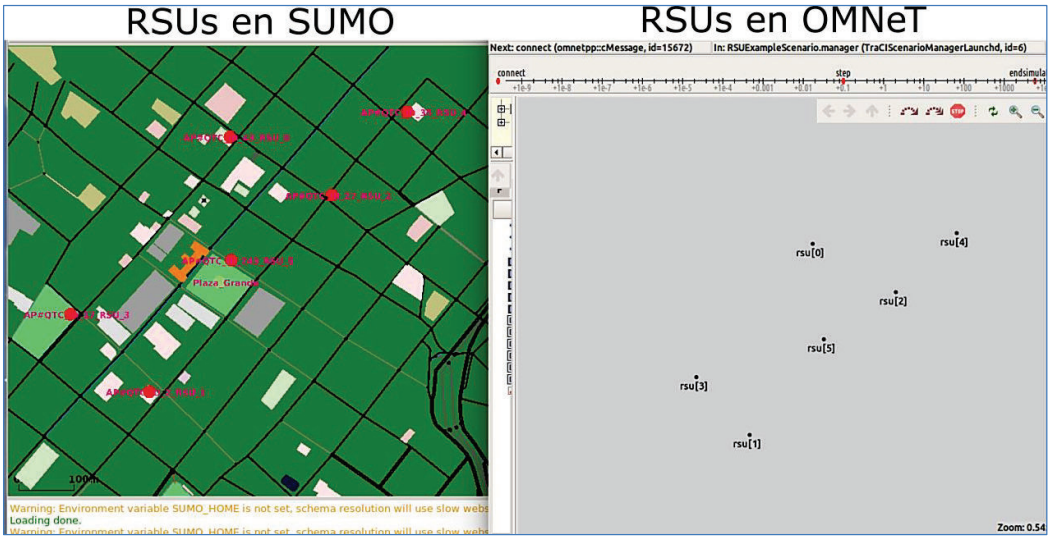


Figura 2.13- Visualización de RSUs en veins

El módulo de control de tráfico TRACI (*Traffic Control Interface*) pertenece a SUMO, emplea la información de las nodos, aristas y rutas programadas en el mapa de movilidad vehicular para reportar constantemente las condiciones de la red, cuando un vehículo ingresa en la VANET se inicializan los módulos que se indicaron para las RSU, IEEE 802.11p se encarga de establecer la frecuencia de portadora a 5.9 [Ghz] y se gestionan los canales con IEEE 1609.4 de la siguiente manera:

```
INFO (PhyLayer80211p)RSUExampleScenario.node[0].nic.phy80211p:
RSUExampleScenario.node[0].nic.phy80211p: node[0]::PhyLayer80211p:
createPathLossModel(): alpha set from config.xml to 2
INFO (PhyLayer80211p)RSUExampleScenario.node[0].nic.phy80211p:
RSUExampleScenario.node[0].nic.phy80211p: node[0]::PhyLayer80211p:
createPathLossModel(): carrierFrequency set from config.xml to 5.89e+09
INFO (PhyLayer80211p)RSUExampleScenario.node[0].nic.phy80211p:
RSUExampleScenario.node[0].nic.phy80211p: node[0]::PhyLayer80211p:
AnalogueModel "SimplePathlossModel" loaded.
INFO (PhyLayer80211p)RSUExampleScenario.node[0].nic.phy80211p:
RSUExampleScenario.node[0].nic.phy80211p: node[0]::PhyLayer80211p:
initializeSimpleObstacleShadowing(): carrierFrequency set from config.xml
to 5.89e+09
INFO (PhyLayer80211p)RSUExampleScenario.node[0].nic.phy80211p:
RSUExampleScenario.node[0].nic.phy80211p: node[0]::PhyLayer80211p:
AnalogueModel "SimpleObstacleShadowing" loaded.
INFO (PhyLayer80211p)RSUExampleScenario.node[0].nic.phy80211p:
RSUExampleScenario.node[0].nic.phy80211p: node[0]::PhyLayer80211p:
Decider "Decider80211p" loaded.
RSUExampleScenario.node[0].nic.mac1609_4: Initializing module
RSUExampleScenario.node[0].nic.mac1609_4, stage 0
INFO (Mac1609_4)RSUExampleScenario.node[0].nic.mac1609_4:
RSUExampleScenario.node[0].nic.mac1609_4: Channel turned idle: Switch: 1
INFO (Mac1609_4)RSUExampleScenario.node[0].nic.mac1609_4:
RSUExampleScenario.node[0].nic.mac1609_4: Restarting contention.
```

```
INFO (Mac1609_4)RSUExampleScenario.node[0].nic.mac1609_4:
RSUExampleScenario.node[0].nic.mac1609_4: Channel is already idle for:0
since 0.1
INFO (Mac1609_4)RSUExampleScenario.node[0].nic.mac1609_4:
RSUExampleScenario.node[0].nic.mac1609_4: I don't have any new events in
this EDCA sub system
RSUExampleScenario.node[0].veinsmobility: Initializing module
RSUExampleScenario.node[0].veinsmobility, stage 0
```

Cuando ingresa el segundo vehículo a la red, se añaden a una lista de TRACI también todas las posibles rutas que se ofrecen en caso de tener una señal de alerta en la red.

```
INFO (TraCIScenarioManagerLaunchd)RSUExampleScenario.manager: Adding
route0 to list of possible routes
INFO (TraCIScenarioManagerLaunchd)RSUExampleScenario.manager: Adding
route1 to list of possible routes
```

El reporte de TRACI cada vez que hay una variación en la red incluye al número de vehículos que inician su trayecto, los nodos que se encuentran aún en la red, los que se estacionan y aquellos que llegaron a su destino, estos datos se muestran de la siguiente manera:

```
INFO (TraCIScenarioManagerLaunchd)RSUExampleScenario.manager: Writing
TraCI message of 6 bytes
INFO (TraCIScenarioManagerLaunchd)RSUExampleScenario.manager: Reading
TraCI message of 204 bytes
INFO (TraCIScenarioManagerLaunchd)RSUExampleScenario.manager: Getting 3
subscription results
INFO (TraCIScenarioManagerLaunchd)RSUExampleScenario.manager: TraCI
reports 1 departed vehicles.
INFO (TraCIScenarioManagerLaunchd)RSUExampleScenario.manager: TraCI
reports 0 arrived vehicles.
INFO (TraCIScenarioManagerLaunchd)RSUExampleScenario.manager: TraCI
reports current time step as 1100ms.
INFO (TraCIScenarioManagerLaunchd)RSUExampleScenario.manager: TraCI
reports 0 vehicles starting to teleport.
INFO (TraCIScenarioManagerLaunchd)RSUExampleScenario.manager: TraCI
reports 0 vehicles ending teleport.
INFO (TraCIScenarioManagerLaunchd)RSUExampleScenario.manager: TraCI
reports 0 vehicles starting to park.
INFO (TraCIScenarioManagerLaunchd)RSUExampleScenario.manager: TraCI
reports 0 vehicles ending to park.
INFO (TraCIScenarioManagerLaunchd)RSUExampleScenario.manager: TraCI
reports 2 active vehicles.
```

Después de reportarse el nuevo nodo a la red, se actualiza la presencia de quienes son parte del sistema cada 0.1 segundos, sin reportar novedades mientras no llegue un nuevo vehículo a la red o surja algún mensaje de alerta. El nodo 20 (Ruta12.0) presenta una variación al reporte de los demás vehículos, con una señal de alerta que solicita al canal de comunicación se le dé acceso con una prioridad mayor con las siguientes líneas:

```
INFO (Mac1609_4)RSUExampleScenario.node[20].nic.mac1609_4: Received a
message from upper layer for channel 178 Access Category (Priority):  2
INFO (Mac1609_4)RSUExampleScenario.node[20].nic.mac1609_4: sorted packet
into queue of EDCA 0 this packet is now at position: 1
INFO (Mac1609_4)RSUExampleScenario.node[20].nic.mac1609_4: this packet is
for the currently active channel
INFO (Mac1609_4)RSUExampleScenario.node[20].nic.mac1609_4: Restarting
contention.
```

El nodo que genera la alarma se identifica con color rojo y como se puede ver a continuación, la capa física se encarga de transmitir el mensaje de alerta a los nodos cercanos mediante un **AirFrame11p**[15], éste proceso se representa en la interfaz gráfica de OMNeT++ como se muestra en la Figura 2.14.

```
** Event #427  t=42.204614  RSUExampleScenario.node[20].nic.phy80211p
(PhyLayer80211p, id=167)  on data (Mac80211Pkt, id=281)
INFO (PhyLayer80211p)RSUExampleScenario.node[20].nic.phy80211p:
node[20]::PhyLayer80211p: AirFrame encapsulated, length: 1326
```

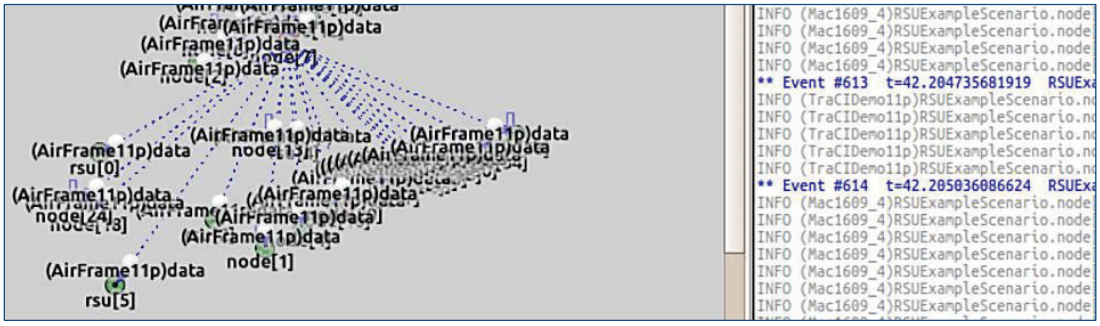


Figura 2.14- Transmisión de AirFrame11p

El envío del **AirFrame11p** se puede observar de una manera más detallada en la ventana de datos de red, al seleccionar la opción de visualizar un evento a la vez para observar el origen, los destinos e información de la transmisión como se muestra en el área sombreada de la Figura 2.15.

Event#	Time	Src/Dest	Name	Info
#7900	42.210453906574	node[9] --> node[48]	data	id=12855 kind=22003 length=166 bytes
#7900	42.210453906574	node[9] --> node[49]	data	id=12857 kind=22003 length=166 bytes
#7900	42.210453906574	node[9] --> node[52]	data	id=12859 kind=22003 length=166 bytes
#7900	42.210453906574	node[9] --> node[53]	data	id=12861 kind=22003 length=166 bytes
#7900	42.210453906574	node[9] --> node[56]	data	id=12863 kind=22003 length=166 bytes
#7900	42.210453906574	node[9] --> node[57]	data	id=12816 kind=22003 length=166 bytes
#8048	42.210659199854	node[5] --> rsu[1]	data	id=13086 kind=22003 length=166 bytes
#8048	42.210659199854	node[5] --> rsu[3]	data	id=13088 kind=22003 length=166 bytes
#8048	42.210659199854	node[5] --> rsu[5]	data	id=13090 kind=22003 length=166 bytes
#8048	42.210659199854	node[5] --> node[0]	data	id=13092 kind=22003 length=166 bytes
#8048	42.210659199854	node[5] --> node[3]	data	id=13094 kind=22003 length=166 bytes
#8048	42.210659199854	node[5] --> node[8]	data	id=13096 kind=22003 length=166 bytes
#8048	42.210659199854	node[5] --> node[9]	data	id=13098 kind=22003 length=166 bytes
#8048	42.210659199854	node[5] --> node[14]	data	id=13100 kind=22003 length=166 bytes
#8048	42.210659199854	node[5] --> node[15]	data	id=13102 kind=22003 length=166 bytes
#8048	42.210659199854	node[5] --> node[16]	data	id=13104 kind=22003 length=166 bytes
#8048	42.210659199854	node[5] --> node[18]	data	id=13106 kind=22003 length=166 bytes

Figura 2.15- Visualización de fuente, destinos e información de AirFrame11p

[15] AirFrames: representan las características físicas del marco (frecuencia, potencia, etc.). AirFrame11p encapsula los marcos 802.11p en el simulador OMNeT++ [6]

Cada nodo recibe y procesa la información del paquete **Airframe11p** como se muestra en las siguientes líneas:

```
INFO (PhyLayer80211p)RSUExampleScenario.node[34].nic.phy80211p:
node[34]::PhyLayer80211p: Received new AirFrame (AirFrame11p)data from
channel.
INFO (PhyLayer80211p)RSUExampleScenario.node[34].nic.phy80211p:
PhyLayer(SimplePathlossModel): sqrdistance is: 1811.11
INFO (PhyLayer80211p)RSUExampleScenario.node[34].nic.phy80211p:
PhyLayer(SimplePathlossModel): wavelength is: 0.0508985
INFO (PhyLayer80211p)RSUExampleScenario.node[34].nic.phy80211p:
PhyLayer(SimplePathlossModel): distance factor is: 3.49651e-06
INFO (PhyLayer80211p)RSUExampleScenario.node[34].nic.phy80211p:
PhyLayer(SimplePathlossModel): Signal contains frequency dimension: yes
INFO (PhyLayer80211p)RSUExampleScenario.node[34].nic.phy80211p:
PhyLayer(SimpleObstacleShadowing): value is: 1
INFO (PhyLayer80211p)RSUExampleScenario.node[34].nic.phy80211p: [Host 34]
- PhyLayer(Decider): Processing AirFrame...
```

Cuando los nodos de la red cuentan con la información del último evento suscitado en el sistema se identifican con color verde, y una vez actualizados todos los dispositivos involucrados, se eliminan los mensajes previos de cada nodo para no saturar la red como se puede observar en la Figura 2.16.

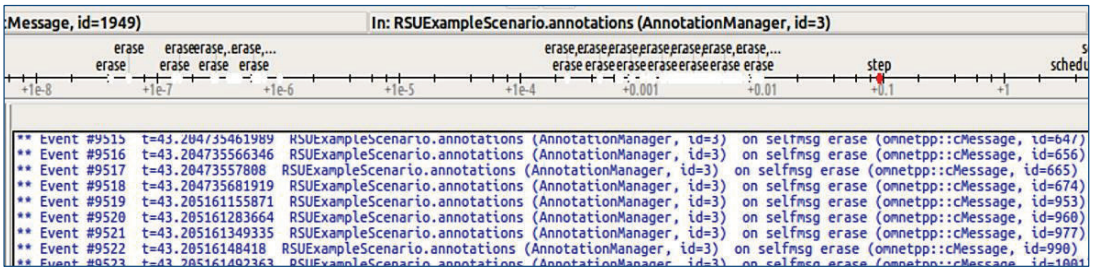


Figura 2.16- Eventos que reportan los mensajes eliminados

Con los datos de la VANET el módulo TRACI modifica las rutas de cada uno de los vehículos que pertenecen a la red, los nodos se reportan cada cierto tiempo y se incluye la nueva línea a la que son redirigidos como se puede ver en las siguientes líneas:

```
INFO (TraCIScenarioManagerLaunchd)RSUExampleScenario.manager: module
Ruta0.0 moving to 1004.77,983.012
INFO (TraCIScenarioManagerLaunchd)RSUExampleScenario.manager:
node[0]::Veins::TraCIMobility: checkIfOutside, outside=0 borderStep:
(0,0,0)
INFO (TraCIScenarioManagerLaunchd)RSUExampleScenario.manager:
updatePosition: HostMove startPos: (1004.77,983.012,1.895) direction:
(0.642612,-0.766192,0) startTime: 43.9 speed: 11.11
```

2.2.2.1 *Accidentes programados*

El simulador veins permite generar accidentes de tránsito programados en el escenario dos, como una herramienta para el estudio de VANETs, para modificar las opciones de accidentes se accede al archivo **omnetpp.ini** de veins y se configura la sección *Mobility* que cuenta con parámetros para establecer el número de accidentes programados, el tiempo en que inician y la duración que tendrán, como se observa en las siguientes líneas de código:

```
*.node[*0].veinsmobility.accidentCount = 1
*.node[*0].veinsmobility.accidentStart = 75s
*.node[*0].veinsmobility.accidentDuration = 30s
```

Los accidentes programados (*scheduled accident*) se generan a partir del tiempo establecido en la red, en ese momento, un vehículo se detiene a propósito para simular un accidente de cualquier tipo y su nodo se identifica con color rojo (ver Figura 2.17). Adicionalmente, en la ventana de visualización de eventos se genera la señal de alerta con su respectivo número de evento, tiempo en que sucede y el reporte del nodo con velocidad cero, como se muestra a continuación:

```
** Event #10434  t=50.1  RSUExampleScenario.node[0].veinsmobility
(TraCIMobility, id=46)  on selfmsg scheduledAccident (omnetpp::cMessage,
id=31)
INFO (TraCIMobility)RSUExampleScenario.node[0].veinsmobility: Writing
TraCI message of 23 bytes
INFO (TraCIMobility)RSUExampleScenario.node[0].veinsmobility: Reading
TraCI message of 7 bytes
INFO (TraCIScenarioManagerLaunchd)RSUExampleScenario.manager: module
Ruta0.0 moving to 1057.29,919.24
INFO (TraCIScenarioManagerLaunchd)RSUExampleScenario.manager:
node[0]::Veins::TraCIMobility: checkIfOutside, outside=0 borderStep:
(0,0,0)
INFO (TraCIScenarioManagerLaunchd)RSUExampleScenario.manager:
updatePosition:  HostMove  startPos: (1057.29,919.24,1.895) direction:
(0.612036,-0.79083,0) startTime: 53.8 speed: 0
```

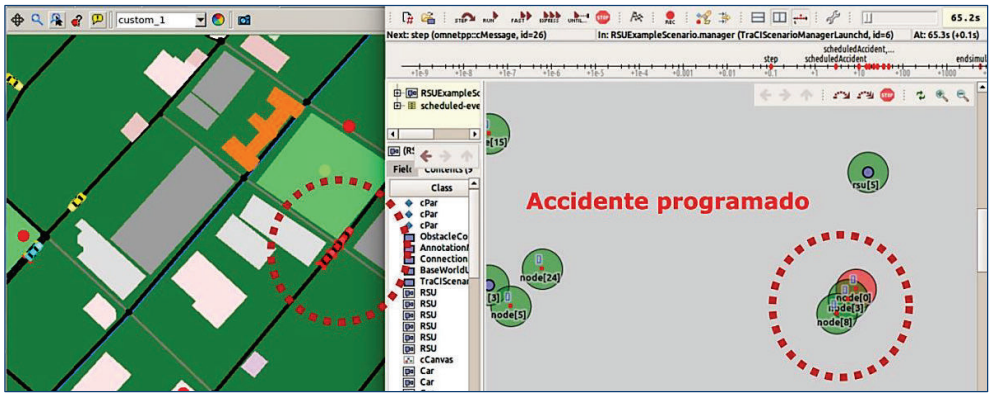


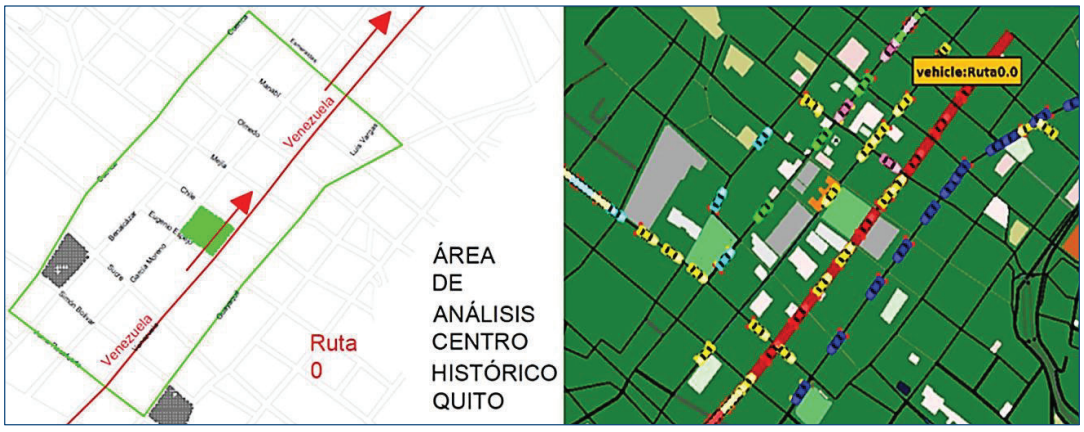
Figura 2.17- Visualización de un accidente programado

CAPÍTULO 3: PRUEBAS Y RESULTADOS

En esta sección se presentan los resultados de las simulaciones del escenario uno (sin VANET) y el escenario dos (con VANET), a fin de visualizar los cambios que se producen en las rutas y determinar el beneficio de uso de una VANET.

3.1 ESCENARIO UNO

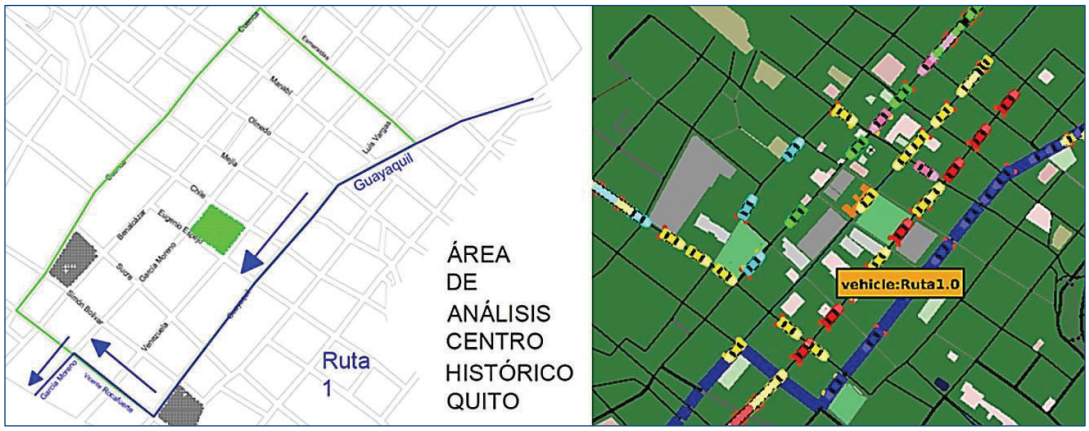
En las Figuras 3.1 a la 3.5 se presenta una comparación de la movilidad planificada en los planos de AutoCAD y el tráfico vehicular obtenido en SUMO. Es decir, en la parte a de cada una de las figuras se muestran las rutas establecidas en la etapa de diseño, las cuales se obtuvieron a partir de los planos de AutoCAD, por otro lado, en la parte b de las figuras se exponen las simulaciones de la movilidad obtenida en SUMO para cada ruta. De esta manera se indica la correspondencia de los mapas a fin de comprobar que los vehículos cumplen tanto con la ruta programada como con la identificación de colores y nombres.



a) Plano de ruta 0 programada b) Simulación de ruta 0

Figura 3.1- Simulación de la ruta 0, escenario uno

En la simulación de movilidad vehicular cabe recordar, que únicamente las 5 primeras rutas se programaron con una identificación de color, ya que éstas cuentan con un porcentaje mayor de vehículos; por otro lado, los nodos de las rutas restantes se tornan de un color en general y su identificación se da únicamente al seleccionar cada nodo y solicitar su nombre.



a) Plano de ruta 1 programada b) Simulación de ruta 1

Figura 3.2- Simulación de la ruta 1, escenario uno

a) Plano de ruta 2 programada b) Simulación de ruta 2

Figura 3.3- Simulación de la ruta 2, escenario uno

a) Plano de ruta 3 programada b) Simulación de ruta 3

Figura 3.4- Simulación de la ruta 3, escenario uno

a) Plano de ruta 4 programada b) Simulación de ruta 4

Figura 3.5- Simulación de la ruta 4, escenario uno

En la Figura 3.6 a la Figura 3.10 se observan las 10 rutas restantes del escenario uno que tienen un menor porcentaje de vehículos y no cuentan con un color identificativo, para visualizar los trayectos se activa la herramienta mostrar ruta actual de SUMO, al seleccionarla mediante un clic derecho en cada vehículo.

a) Simulación de la ruta 5 b) Simulación de la ruta 6

Figura 3.6- Simulación de rutas 5 y 6, escenario uno

a) Simulación de la ruta 7 b) Simulación de la ruta 8

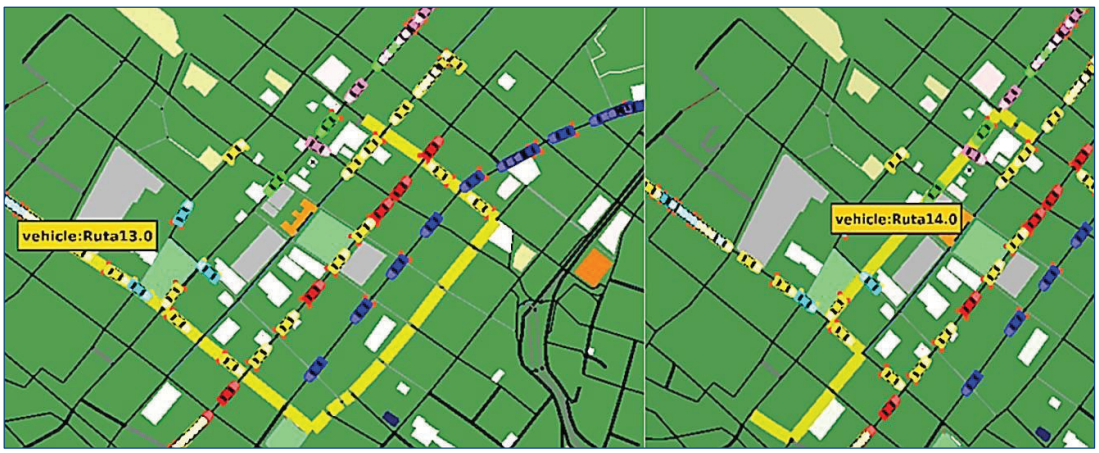
Figura 3.7- Simulación de rutas 7 y 8, escenario uno

a) Simulación de la ruta 9 b) Simulación de la ruta 10

Figura 3.8- Simulación de rutas 9 y 10, escenario uno

a) Simulación de la ruta 11 b) Simulación de la ruta 12

Figura 3.9- Simulación de rutas 11 y 12, escenario uno



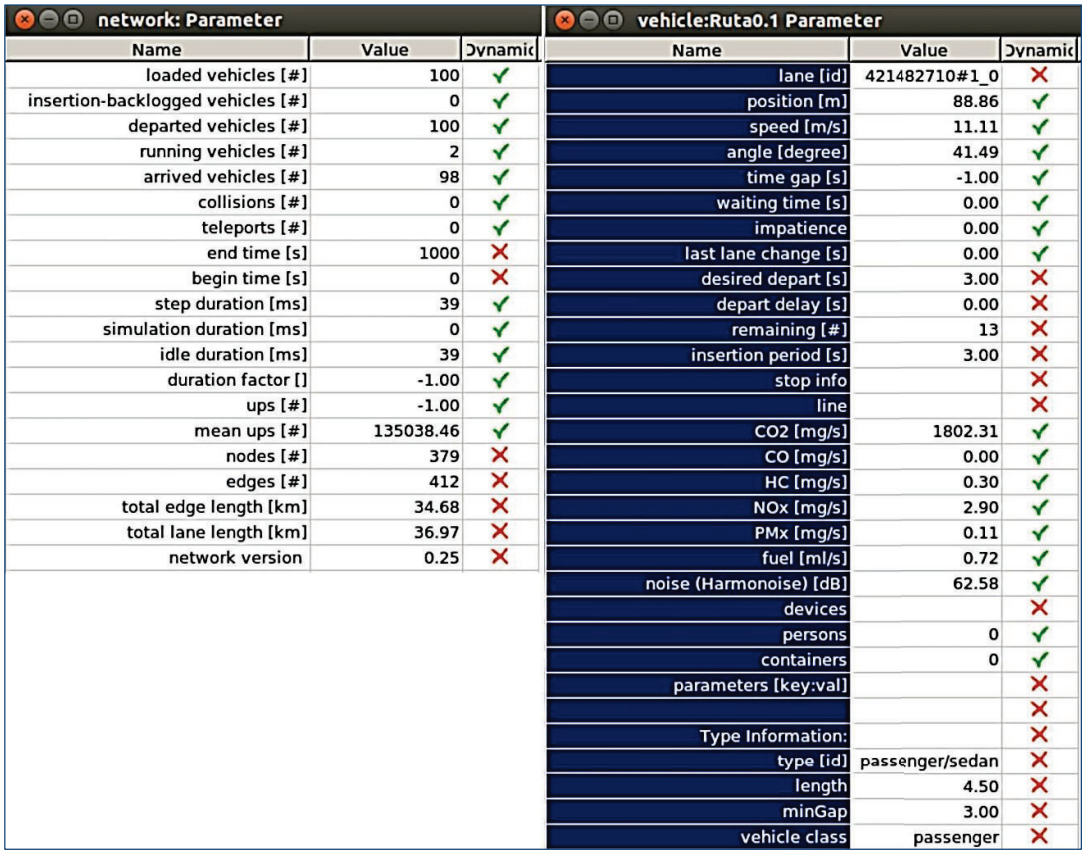
a) Simulación de la ruta 13 b) Simulación de la ruta 14

Figura 3.10- Simulación de rutas 13 y 14, escenario uno

El simulador SUMO permite observar en cualquier momento los parámetros de vehículos que se encuentran en el mapa; de los cuales los más importantes son:

el id de arista en que se encuentra el auto, la velocidad actual, la emisión de CO2 que produce, el ruido ambiental generado y la velocidad máxima a la que puede llegar el vehículo.

Por otro lado, los parámetros generales de la red en SUMO proveen información importante del entorno de movilidad como: el número de vehículos que se encuentran activos, los vehículos que están por llegar y aquellos que ya llegaron a su destino, el número de colisiones que se produjeron, el número de nodos y aristas o líneas que tiene el mapa y su longitud en kilómetros. Todos los parámetros se muestran en la Figura 3.11a y los datos específicos para el vehículo 10 en la ruta cero se observan en la Figura 3.11b.



network: Parameter

Name	Value	Dynamic
loaded vehicles [#]	100	✓
insertion-backlogged vehicles [#]	0	✓
departed vehicles [#]	100	✓
running vehicles [#]	2	✓
arrived vehicles [#]	98	✓
collisions [#]	0	✓
teleports [#]	0	✓
end time [s]	1000	✗
begin time [s]	0	✗
step duration [ms]	39	✓
simulation duration [ms]	0	✓
idle duration [ms]	39	✓
duration factor []	-1.00	✓
ups [#]	-1.00	✓
mean ups [#]	135038.46	✓
nodes [#]	379	✗
edges [#]	412	✗
total edge length [km]	34.68	✗
total lane length [km]	36.97	✗
network version	0.25	✗

vehicle:Ruta0.1 Parameter

Name	Value	Dynamic
lane [id]	421482710#1_0	✗
position [m]	88.86	✓
speed [m/s]	11.11	✓
angle [degree]	41.49	✓
time gap [s]	-1.00	✓
waiting time [s]	0.00	✓
impatience	0.00	✓
last lane change [s]	0.00	✓
desired depart [s]	3.00	✗
depart delay [s]	0.00	✗
remaining [#]	13	✗
insertion period [s]	3.00	✗
stop info		✗
line		✗
CO2 [mg/s]	1802.31	✓
CO [mg/s]	0.00	✓
HC [mg/s]	0.30	✓
NOx [mg/s]	2.90	✓
PMx [mg/s]	0.11	✓
fuel [ml/s]	0.72	✓
noise (Harmonoise) [dB]	62.58	✓
devices		✗
persons	0	✓
containers	0	✓
parameters [key:val]		✗
		✗
Type Information:		✗
type [id]	passenger/sedan	✗
length	4.50	✗
minGap	3.00	✗
vehicle class	passenger	✗

a) Parámetros de red b) Parámetros del vehículo Ruta0.10

Figura 3.11- Parámetros en SUMO para el nodo Ruta0.10

3.2 ESCENARIO DOS

El escenario dos presenta una simulación con la VANET ya implementada, el resultado más relevante en veins, gracias a la identificación por colores, es la variación de diferentes rutas para resolver problemas de movilidad. Entre los principales problemas de movilidad solventados con la VANET se tienen: disminución del tiempo de espera, producto de las largas colas en algunas intersecciones, e información disponible para evitar accidentes. Los cuadros de los vehículos que mantienen las rutas programadas se pintan con su color identificativo; mientras que en los bloques sin color se colocan los nombres de los nodos que presentan cambios en su trayecto y se detallan en la Tabla 3.1.

Tabla 3.1- Vehículos con rutas modificadas por VANET

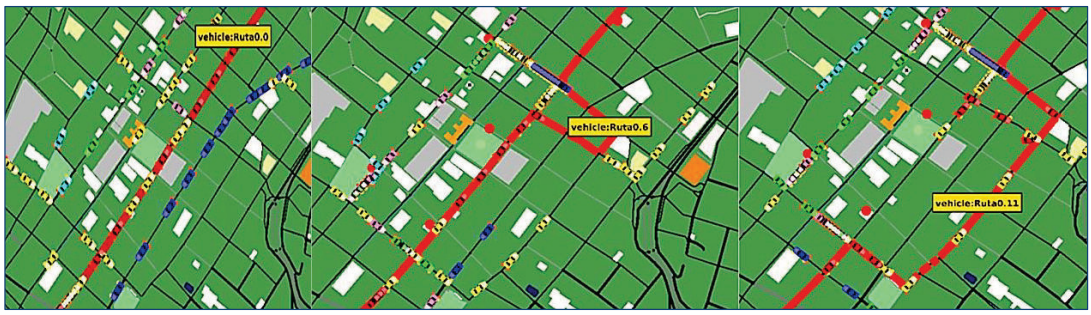
#Veh	Ruta 0	Ruta 1	Ruta 2	Ruta 3	Ruta 4	Ruta 5	Ruta 6	Ruta 7	Ruta 8	Ruta 9	Ruta 10	Ruta 11	Ruta 12	Ruta 13	Ruta1 4
0		1,00		3,00			6,00	7,00	8,0			11,00		13,00	
1		1,01		3,01	4,01		6,01	7,01	8,1						
2		1,02		3,02	4,02		6,02	7,02							
3		1,03		3,03	4,03		6,03	7,03							
4		1,04		3,04	4,04		6,04								
5		1,05		3,05			6,05								
6	0,06	1,06		3,06			6,06	7,06							
7	0,07	1,07					6,07	7,07							
8	0,08	1,08		3,08											
9	0,09	1,09		3,09											
10	0,10	1,10		3,10											
11	0.11	1,11		3,11											
12	0.12														
13	0.13														

Las variaciones de rutas se generan cuando la red detecta un posible accidente en uno de los nodos que conforman la VANET, cuando esto sucede el nodo involucrado se pinta de color rojo en la ventana de red; después, los vehículos cercanos se reportan al nodo emitiendo una alerta con: su posición, velocidad y dirección. La cobertura de la VANET se extiende, debido a que la información se difunde mediante los RSU a todos los dispositivos que se encuentran en el área de cobertura. Además, cada vez que se actualiza la VANET actúa el módulo para el control de tránsito en SUMO denominado TRACI, y utiliza su base de datos con

los caminos existentes en el mapa, para decidir mover los nodos de la red a otras líneas (vías) o mantenerlos en su ruta programada.

Los gráficos restantes que se presentan en esta sección se utilizan para comparar los trayectos programados con las soluciones VANET aplicadas, por lo que se aclara que las rutas 2, 5, 9, 10,12 y 14 no están incluidas, debido a que los vehículos que las conforman no varían su trayecto al implementar la VANET, como se muestra en la Tabla 3.1.

La Figura 3.12 presenta las modificaciones de la ruta 0 de acuerdo a la información mostrada en la Tabla 3.1. Los 6 primeros vehículos rojos (ver Figura 3.12a) siguen la trayectoria programada ya que no existe ningún evento que lo impida, las OBUs de los vehículos restantes, con id 0.06 a 0.10, reciben información de congestión en la calle Venezuela ya que su transversal José Olmedo tiene una prioridad mayor, bajo esta condición, los vehículos modifican su camino hacia la calle José Mejía, y retoman la ruta con la calle José Olmedo (Figura 3.12b). Los últimos vehículos rojos (ver Figura 3.12c), a partir del vehículo con id 0.11, reciben el informe de congestión mucho antes que el segundo grupo, por lo que toman la solución más cercana, es decir la calle Simón Bolívar, para evitar el tramo de la calle Venezuela que reporta congestión.



a) Ruta 4 programada b) Solución VANET 1 c) Solución VANET 2

Figura 3.12- Comparación de ruta 0 programada y soluciones VANET

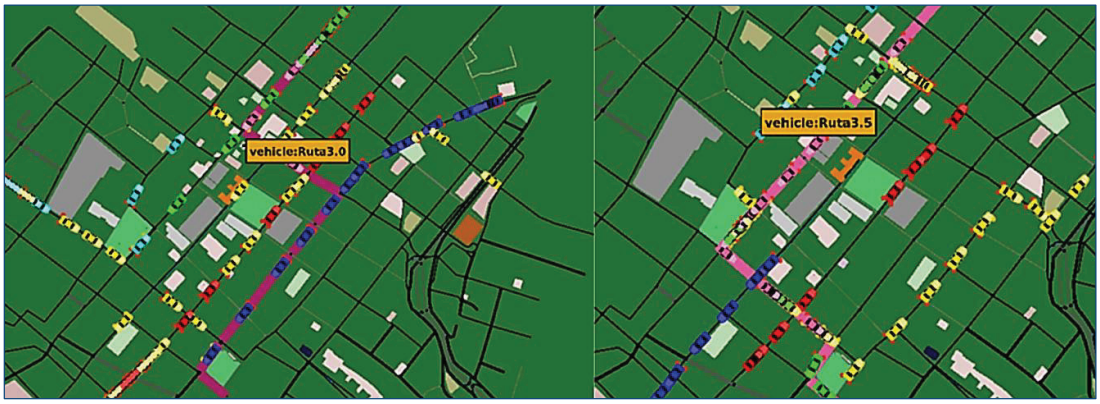
En la ruta 1 se aplica la solución VANET con una menor distancia recorrida, a partir del nodo 1.00 (ver Figura 3.13b) debido a que los vehículos que parten del sur de la calle Venezuela, presentan una cola en la intersección con la calle Simón Bolívar y se prevé que los vehículos de la ruta 1 al desembocar en la calle Vicente Rocafuerte, únicamente incrementarían la congestión vehicular ya que no

existen vías alternas que la ruta 1 pueda utilizar una vez tomada la calle Guayaquil. Una vez que el flujo de nodos disminuye en la intersección mencionada, los nodos 1.12 y 1.13 toman el camino que se programó originalmente sin afectar a la movilidad como se observa en la Figura 3.13a.

a) Ruta 1 programada b) Solución VANET

Figura 3.13- Variación de la Ruta1

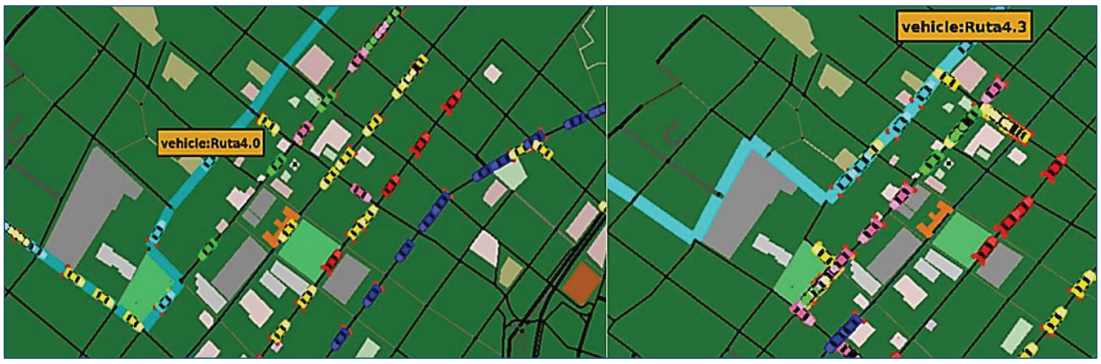
La ruta 3 emplea el trayecto de la solución VANET en todos sus nodos con una menor distancia recorrida (ver Figura 3.14b), excepto en el vehículo 3.07 que es enviado por el trayecto original (ver Figura 3.14a) como una solución para descongestionar la ruta generada por la VANET ya que el flujo de vehículos que brinda esta solución, conlleva una nueva congestión vehicular por lo que la red recalcula los problemas y genera una nueva solución o como en este caso, retoma la misma.



a) Ruta 3 programada b) Solución VANET

Figura 3.14- Variación de la Ruta3

En el caso de los vehículos que utilizan la ruta 4, como se muestra en la Figura 3.15a, el primer vehículo celeste utiliza las aristas programadas ya que en el momento de ingresar al área de análisis no encuentra ninguna alerta; sin embargo, para los siguientes OBUs se actualiza la red VANET y se ofrece una vía alternativa más corta (ver Figura 3.15b), esto da como resultado que los vehículos con id 4.01, 4.02 y 4.03 lleguen a su destino antes que el auto 4.00. A partir del vehículo con id 4.05, la VANET determina que los vehículos pueden tomar las calles programadas, lo que indica que veins no actúa como un seleccionador de rutas efectivas en tiempo o distancia, por el contrario, su prioridad es resolver problemas de tráfico cuando sea necesario y no imponer un recorrido al usuario.



b) Ruta 4 programada b) Solución VANET

Figura 3.15- Variación de la Ruta4

La Figura 3.16a muestra la ruta 6 que se programó para el escenario uno, y se plantea únicamente para compararla con la solución VANET, que se observa en la Figura 3.16b ya que todos los nodos sin excepción en el escenario dos siguen la trayectoria planteada por la red vehicular con una distancia menor recorrida.

a) Ruta 6 programada b) Solución VANET

Figura 3.16- Variación de la Ruta6

La solución VANET de la ruta 7 (ver Figura 3.17b) se aplica para seis de los ocho vehículos que la conforman, ya que disminuye el flujo en la calle Venezuela y la distancia recorrida; los nodos 7.04 y 7.05 siguen la ruta original de la Figura 3.17a.

a) Ruta 7 programada b) Solución VANET

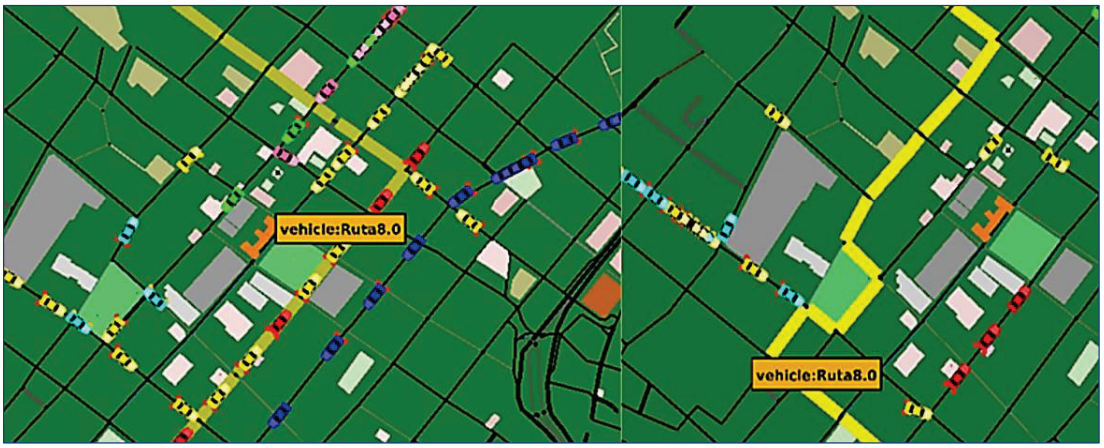
Figura 3.17- Variación de la Ruta7

En los resultados de la ruta 7 se destaca que la diferencia entre los tiempos con y sin VANET decrece progresivamente, como se muestra en la Tabla 3.2, y la progresión continúa incluso al tomar la ruta original que tiene una distancia mayor.

Tabla 3.2- Comparación de datos de la Ruta 7

Vehiculo	Tiempo	Tiempo VANET	Dif.	Distancia	Distancia VANET	Dif
7,00	161,900	146,300	-15,600	1231,64	1208,04	-23,6
7,01	173,900	153,000	-20,900	1231,64	1208,04	-23,6
7,02	190,000	166,900	-23,100	1231,64	1208,04	-23,6
7,03	196,500	172,600	-23,900	1231,64	1208,04	-23,6
7,04	201,700	177,000	-24,700	1231,64	1231,64	0
7,05	243,200	213,200	-30,000	1231,64	1231,64	0
7,06	249,700	219,000	-30,700	1231,64	1208,04	-23,6
7,07	254,200	224,600	-29,600	1231,64	1208,04	-23,6

La Figura 3.18 muestra los trayectos que toman los vehículos de la ruta 8, la solución VANET de la Figura 3.18b la toman únicamente los 2 primeros nodos para disminuir el flujo de la calle Venezuela; sin embargo, el costo en distancia es muy alto por lo que a posteriori se aplican soluciones a otras rutas y se permite al resto de nodos en la ruta 8 tomar el trayecto original (ver Figura 3.18b).



a) Ruta 8 programada b) Solución VANET

Figura 3.18- Variación de la Ruta8

La ruta 11 forma parte de los trayectos aleatorios que se planificó en el diseño como se observa en la Figura 3.19a, para vehículos que dan vueltas en el área y generan congestión, por lo que, la solución tiene una mejora clara en distancia y se aplica en la red (ver Figura 3.19b).

a) Ruta 11 programada b) Solución VANET

Figura 3.19- Variación de la Ruta11

Al igual que la ruta 11, la ruta 13 que se muestra en la Figura 3.20a tiene un trayecto extenso y con la solución VANET aplicada (ver Figura 3.20b) recorre una distancia menor, además disminuye la carga de vehículos en la calle Simón Bolívar que es utilizada por varias rutas.

a) Ruta 13 programada b) Solución VANET

Figura 3.20- Variación de la Ruta13

En la Figura 3.21 se comparan las distancias recorridas para los 100 vehículos tanto para el escenario 1, de sólo movilidad, como para el escenario 2, que incluye la VANET, de estas figuras se puede deducir que para el escenario dos un 10% de los vehículos involucrados presentan un incremento en la distancia recorrida, de éste porcentaje las distancias para los vehículos del 7 al 14 son superiores al utilizar VANET debido a que este grupo de vehículos pertenecen a la ruta cero cuyo camino programado es una línea recta (la distancia más corta entre dos puntos) y las rutas alternas para evitar tráfico involucran el uso de vías secundarias. Un caso similar se da para los vehículos 89 y 90 que pertenecen a la ruta 8, al comparar el camino original con la opción que brinda la red VANET resulta que la ruta original es la mejor opción en distancia ya que utiliza tan sólo 2 vías para llegar a su destino mientras que la alternativa planteada por la VANET cuenta con más de dos vías. En efecto, el 43% de los vehículos presenta una disminución en la distancia recorrida, además, sobresalen los vehículos 97 y 99 de las rutas 11 y 13 respectivamente, los cuales muestran una reducción significativa para llegar a su destino, ya que dichas rutas son aleatorias. Finalmente, hay que mencionar que el 47% restante de los vehículos no tienen variación en la distancia recorrida y siguen la ruta programada.

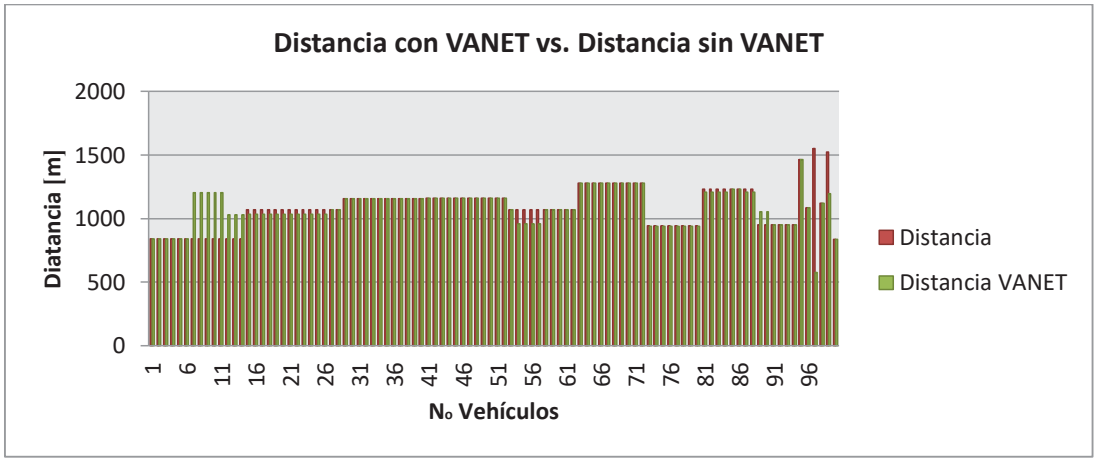


Figura 3.21- Comparación de distancia

El tiempo es uno de los parámetros que más les puede interesar a los usuarios, sin embargo, veins es un simulador que prioriza la percepción de movilidad y la seguridad del conductor, por lo que el tiempo pasa a ser un factor que puede sacrificarse a cambio de obtener un ambiente de seguridad en toda la red vehicular. En la Figura 3.22 se presenta la comparación de tiempos de los escenarios uno y dos, y es evidente el incremento en un gran porcentaje de los vehículos a los que se les aplica VANET.

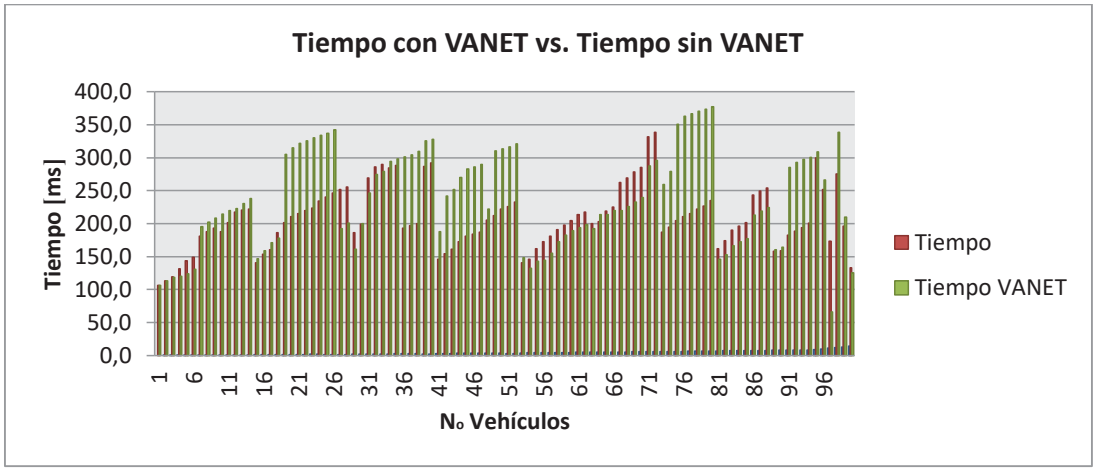


Figura 3.22- Comparación de tiempos

En la Figura 3.23 se muestran los resultados que se generan al comparar las mediciones realizadas en tiempo y distancia para los dos escenarios, más detalles se obtienen en la Tabla C.1 del ANEXO C. El simulador veins para soluciones de tráfico considera 3 parámetros principales como: la eficiencia de tráfico, la

seguridad de tráfico y el confort del conductor: En el estudio se verificó que el 70% de los vehículos presentan una reducción en tiempo, distancia o ambas y apenas un 30% no mostró ninguna mejora o incluso presentaron un incremento debido a factores como las pocas opciones de rutas alternas que tiene VANET al emplear un área reducida con varias calles peatonales. Otra razón es la prioridad que se da a la seguridad vehicular, ya que obliga a los vehículos a disminuir su velocidad en ciertas intersecciones para prevenir accidentes; y finalmente la percepción del usuario de movilidad para lo cual ofrece rutas largas en las que no prioriza tiempo ni distancia, sino más bien la necesidad del conductor de moverse.

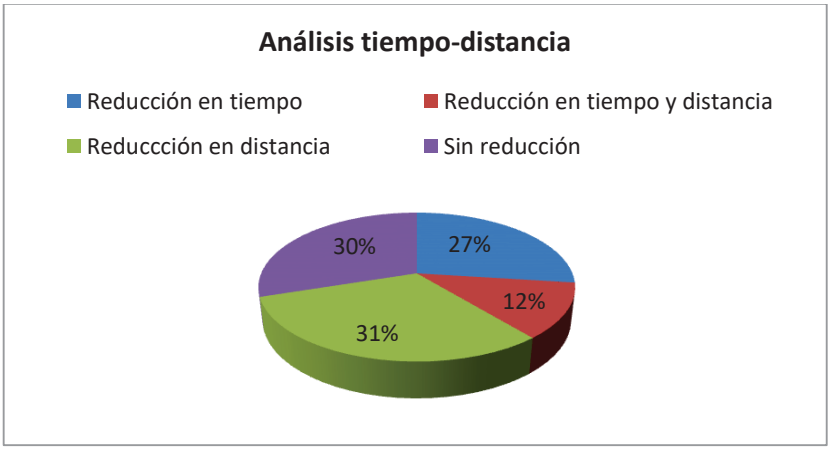


Figura 3.23- Análisis tiempo-distancia de la utilización de la VANET

CAPÍTULO 4: CONCLUSIONES Y RECOMENDACIONES

4.1 CONCLUSIONES

- Las VANETs permiten la transmisión de mensajes a lo largo de un conjunto de vehículos y a través de las carreteras, a fin de informar el estado del tránsito y alertar sobre el movimiento de los vehículos para proporcionar un sistema de trasporte inteligente; el simular una VANET permite tener una visión más clara del número de RSUs que son necesarios para cubrir un área determinada y los lugares en los que éstos se pueden ubicar.
- El uso de la banda de frecuencias de (5.850-5.925) [GHz] que establece el estándar IEEE 802.11p de manera exclusiva para redes vehiculares, evita la saturación de la banda. Los cambios en la capa física del IEEE 802.11a permite trabajar al IEEE 802.11p con canales de 10 [MHz] que se emplean para disminuir el efecto por propagación *Doppler* y trabajar con un bajo *overhead*, para lograr mayor eficiencia espectral. Además, IEEE 802.11p cuenta una variación de IEEE 802.11e en la parte baja de la subcapa MAC para trabajar con EDCA como mecanismo de acceso al medio para comunicación en tiempo real.
- Para la simulación de una VANET es necesario contar con un simulador de tránsito y otro de red, para utilizarlos de manera integrada o híbrida. De las múltiples opciones disponibles, en este trabajo se ha utilizado SUMO como simulador vehicular y OMNet++ como simulador de red, con su *framework* veins para vincularlos, ya que cuentan con interfaces amigables y se mantienen en continuo desarrollo.
- Para este trabajo se analizó una zona particular del centro histórico de Quito, la cual cuenta con mapas georeferenciados, condiciones de tráfico muy cercanas a la realidad, mismas que se pueden visualizar de forma virtual mediante aplicaciones para movilidad como Google Map y un proyecto vigente para la geoposición de los RSUs. Se aprovechó ésta información para crear un plano digital del área y una tabla con la correspondencia de las posiciones de los APs del proyecto #QuitoTeConecta con los RSUs en los simuladores de red y movilidad,

todo esto con el objetivo de contar con una simulación cuyos resultados sean útiles en una futura implementación.

- Se verificó con la simulación de movilidad vehicular en el centro histórico de Quito, que al implementar una VANET, un 70% de los vehículos obtuvo una mejora en tiempo, en distancia o en ambos parámetros. La red vehicular equilibra el tráfico en el área de cobertura, priorizando la eficiencia de tráfico, la seguridad de tráfico y el confort del conductor, todo esto para disminuir el índice de accidentes de tránsito y reducir sus posibles daños colaterales como la congestión en áreas cercanas, estrés y agresividad en los conductores ya que estos factores suelen generar más accidentes.
- El análisis de las soluciones VANET aplicadas, muestra que veins no actúa únicamente como un seleccionador de rutas efectivas en tiempo o distancia, su prioridad es resolver problemas de tráfico cuando sea necesario en toda la red, por lo que cabe aclarar que la red VANET no impone un recorrido al usuario, pero sí sugiere constantemente un mejor trayecto en el área aplicada, para mejorar su movilidad vehicular.

4.2 RECOMENDACIONES

- Se recomienda utilizar los programas SUMO v0.25 y OMNeT++ v5.0 ya que son requisitos para ejecutar el framework veins v4.0 en OMNeT++. El trabajar en el sistema operativo Ubuntu 16.04 permite un mejor acoplamiento de los simuladores y se evitan conflictos de instalación, sin embargo hay que prestar atención a las actualizaciones del sistema, ya que suelen variar las versiones de los programas e impide utilizar veins.
- Al importar el mapa para trabajar en SUMO, es necesario modificar polígonos y puntos de GPS generados para identificar sitios históricos o turísticos ya que en la simulación dificultan la identificación de vehículos, éstos elementos se encuentran en el archivo **centro.poly.xml** en la carpeta de SUMO y pueden eliminarse manualmente sin afectar a otros parámetros del simulador. En el presente trabajo se aprovecha este archivo para generar los puntos geoposicionados que identifican a los dispositivos RSUs en el mapa.

- Para el diseño de movilidad vehicular, en este documento se modificó el archivo de SUMO **centro.rou.xml** tanto para generar los vehículos de manera particular, como para programar sus rutas y generarlos en grupos mediante la función *Flow* de SUMO. Se recomienda escoger la emisión de vehículos que mejor se acople a las necesidades del estudio a desarrollarse, ya que tienen distintas características en orden e identificación que pueden aprovecharse.
- En cuanto al diseño de la red de comunicación, se recomienda crear un mapa digital geoposicionado del área estudiada para visualizar distancias y radios de alcance de las unidades RSU, además una tabla de relación entre las posiciones de los dispositivos fijos en SUMO y OMNeT++ para tener una referencia de datos y evitar confusiones de ubicación.
- Se recomienda ejecutar el simulador OMNeT++ en un espacio de trabajo exclusivo para el proyecto veins, ya que el incluir al programa vehicular en un espacio con ejemplos generales, propicia a errores por el uso de directorios y nombres duplicados en el entorno de simulación.
- El presente trabajo, además sirve de guía para la utilización de los simuladores SUMO y OMNeT++, y se recomienda emplearlo a futuro como referencia para extender su complejidad a un área geográfica mayor e incluir mayores detalles respecto a vehículos, conductores y estados de la vías.
- Para trabajos futuros se sugiere el estudio de los vehículos de emergencia en la red VANET, la generación de accidentes programados para crear un plan de soluciones a diferentes causas de congestión, y el uso de semáforos inteligentes en la red.

REFERENCIAS

[1] Diario EL NORTE, “OMM propone en Ecuador urgente reducción emisiones CO2,” *Diario EL NORTE*, 2013. [Online]. Available: http://www.elnorte.ec/ecuador/30810-omm-propone-en-ecuador-urgente-reducci%c3%b3n-emisiones-co2.html. [Accessed: 03-Oct-2016].

[2] Carrión Fernando, “Pico y placa,” *Diario Hoy*, 2010. [Online]. Available: https://works.bepress.com/fernando_carrion/360/. [Accessed: 23-Aug-2016].

[3] ELUNIVERSO.COM, “Ecuador, en el top 10 de los peores países para conducir, según Waze,” *Diario El Universo*, 2015. [Online]. Available: http://www.eluniverso.com/noticias/2015/10/02/nota/5160911/ecuador-top-10-peores-paises-conducir-segun-waze. [Accessed: 23-Aug-2016].

[4] C. Sommer, R. German, and F. Dressler, “Bidirectionally Coupled Network and Road Traffic Simulation for Improved IVC Analysis,” *IEEE Trans. Mob. Comput.*, vol. 10, no. 1, pp. 3–15, Jan. 2011.

[5] “SUMO,” 2016. [Online]. Available: http://www.sumo.dlr.de/userdoc/SUMO-GUI.html. [Accessed: 11-Nov-2016].

[6] OMNET++, “OMNeT++ Discrete Event Simulator - Home,” *OMNET++*, 2015. [Online]. Available: https://omnetpp.org/. [Accessed: 26-Aug-2016].

[7] Ministerio de Coordinación de Seguridad, “Plan Nacional de Seguridad Integral,” 2011.

[8] E. L. Oro, L. O. S. Rios, M. Santiago, S. Elena, S. D. D. E. L. O. S. Tsachilas, and Z. Chinchipe, “Comparativo Mensual Marzo 2015 - 2016 Número de Siniestros por Provincia,” 2016.

[9] DMT/EMSAT/EMOP-Q/UOST/UPGT, “Mapas tematicos. Estructura del transporte colectivo y volumenes vehiculares diarios en los ejes viales principales (2011),” 2011. [Online]. Available: http://sthv.quito.gob.ec/spirales/9_mapas_tematicos/9_10_movilidad/9_10_2_3.html. [Accessed: 06-Nov-2016].

[10] A. Palaguachi, D. Alexandra, E. Tenempaguay, and E. Estuardo, “Diseño de una red de comunicación vehicular inteligente, integrando la tecnología Ad-Hoc con LTE, para la movilidad en la zona urbana de la ciudad de Cuenca,” 2015.

[11] P. González and R. Cerezo, “Estudio del simulador de redes vehiculares veins,” p. 166, 2012.

[12] M. Piórkowski, M. Raya, A. L. Lugo, P. Papadimitratos, M. Grossglauser, and J.-P. Hubaux, “TraNS,” *ACM SIGMOBILE Mobile Computing and Communications Review*, 01-Jan-2008. [Online]. Available: http://portal.acm.org/citation.cfm?doid=1374512.1374522. [Accessed: 03-Feb-2017].

[13] Network Simulation Tools Project Team, “Groove Net Simulator | Network Simulation Tools,” 2017. [Online]. Available: https://networksimulationtools.com/groove-net-simulator-projects/. [Accessed: 03-Feb-2017].

[14] T. Higashino, “MobiREAL : realistic network simulator for MANET,” 2009. [Online]. Available: http://www.mobireal.net/. [Accessed: 03-Feb-2017].

[15] M. AL-Qutayri, C. Yeun, and F. Al-Hawi, *Computational Intelligence and Modern Heuristics*, vol. 1. InTech, 2010.

[16] “Mobile Ad hoc Networks (Routing Protocol) | Brunel University London,”

Brunel University London, 2016. [Online]. Available: http://www.brunel.ac.uk/cedps/electronic-computer-engineering/research-activities/wncc/mobile-ad-hoc-networksrouting-protocol. [Accessed: 28-Dec-2016].

[17] S. Al-Sultan, M. M. Al-Doori, A. H. Al-Bayatti, and H. Zedan, "A comprehensive survey on vehicular Ad Hoc network," *J. Netw. Comput. Appl.*, vol. 37, no. 1, pp. 380–392, 2014.

[18] M. Faezipour, M. Nourani, A. Saeed, and S. Addepalli, "Progress and challenges in intelligent vehicle area networks," *Commun. ACM*, vol. 55, no. 2, pp. 90–100, 2012.

[19] O. Orozco Sarasti and G. Llano Ramírez, "VANET APPLICATIONS FOCUSED ON ENVIRONMENTAL SUSTAINABILITY, A SYSTEMATIC REVIEW," *Cienc. e Ing. Neogranadina*, vol. 24, no. 2, pp. 111–132, 2014.

[20] J. Marousek, D. Khijniak, T. Zhang, T. Weil, and W. Xiang, "Wireless Access for Vehicular Environments {(WAVE)}," *IEEE Globecom 2008 Des. Dev. Forum*, 2008.

[21] J. Kenney, *Standards and regulations*, vol. 9. 2009.

[22] IEEE, *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*, vol. 2012, no. March. 2012.

[23] L. Miao, K. Djouani, B. J. Van Wyk, and Y. Hamam, "Evaluation and Enhancement of IEEE 802.11p Standard: A Survey," *Mob. Comput.*, vol. 1, no. 1, pp. 15–30, 2012.

[24] D. Eckhoff and C. Sommer, "A Multi-Channel IEEE 1609.4 and 802.11 p EDCA Model for the Veins Framework," *Proc. 5th ACM/ICST Int. …*, pp. 4–5, 2012.

[25] J. Mayné, "Comunicaciones Inalámbricas," pp. 1–10, 2005.

[26] C. Sommer, "Veins," 2016. [Online]. Available: http://veins.car2x.org/download/. [Accessed: 26-Nov-2016].

[27] OpenStreetMap Foundation, "OpenStreetMap," 2016. [Online]. Available: https://www.openstreetmap.org/search?query=Quito#map=17/-0.22027/-78.51296. [Accessed: 22-Nov-2016].

[28] Google, "Mapa del Centro de Quito," 2016. [Online]. Available: https://www.google.com.ec/maps/@-0.2190291,-78.5108042,16z/data=!5m1!1e1?hl=es-419. [Accessed: 26-Nov-2016].

[29] CONQUITO, "Mapa | QuitoTeConecta," 2016. [Online]. Available: http://www.quitoteconecta.gob.ec/mapa/. [Accessed: 22-Nov-2016].

[30] A. Mercado, R. Berrios F., and P. Chan Ye, "Redes inalámbricas ad hoc," pp. 1–7, 2003.

[31] V. Studio, "RF,RFIC&Microwave theory," vol. 5884, pp. 1–14, 2015.

[32] M. Rouse, "Computer Glossary, Computer Terms - Technology Definitions and Cheat Sheets from WhatIs.com - The Tech Dictionary and IT Encyclopedia," 2007. [Online]. Available: http://whatis.techtarget.com/. [Accessed: 03-Jan-2017].

[33] Autodesk, "AutoCAD," 2016. [Online]. Available: http://latinoamerica.autodesk.com/products/autocad/overview. [Accessed: 21-Nov-2016].

ANEXOS

ANEXO A– Siniestros por causas probables

Tabla A.1- Siniestros por causas probables [8]

CODIGO	CAUSAS PROBABLES	ENE	FEB	MAR	ABR	TOTAL A ABRIL -2016	%
C14	CONDUCIR DESATENTO A LAS CONDICIONES DE TRANSITO (CELULAR, PANTALLAS DE VIDEO, COMIDA, MAQUILLAJE O CUALQUIER OTRO ELEMENTO DISTRACTOR)	634	632	590	492	**2.348**	**21,97**
C09	CONDUCIR VEHICULO SUPERANDO LOS LIMITES MAXIMOS DE VELOCIDAD	408	314	262	289	**1.273**	**11,91**
C23	NO RESPETAR LAS SEÑALES REGLAMENTARIAS DE TRANSITO (PARE, CEDA EL PASO, LUZ ROJA DEL SEMAFORO, ETC)	366	300	296	248	**1.210**	**11,32**
C11	NO MANTENER LA DISTANCIA PRUDENCIAL CON RESPECTO AL VEHICULO QUE LE ANTECEDE	263	238	220	239	**960**	**8,98**
C25	NO CEDER EL DERECHO DE VIA O PREFERENCIA DE PASO A VEHICULOS	171	207	192	202	**772**	**7,22**
C06	CONDUCE BAJO LA INFLUENCIA DE ALCOHOL, SUSTANCIAS ESTUPEFACIENTES O PSICOTROPICAS Y/O MEDICAMENTOS	188	179	137	169	**673**	**6,30**
C26	NO CEDER EL DERECHO DE VIA O PREFERENCIA DE PASO AL PEATON	156	153	202	150	**661**	**6,18**
C19	REALIZAR CAMBIO BRUSCO O INDEBIDO DE CARRIL	185	160	138	138	**621**	**5,81**
C12	NO GUARDAR LA DISTANCIA LATERAL MINIMA DE SEGURIDAD ENTRE VEHICULOS	194	102	81	127	**504**	**4,72**
C16	NO TRANSITAR POR LAS ACERAS O ZONAS DE SEGURIDAD DESTINADAS PARA EL EFECTO	89	75	57	69	**290**	**2,71**
C03	CONDUCIR EN ESTADO DE SOMNOLENCIA O MALAS CONDICIONES FISICAS (SUEÑO, CANSANCIO Y FATIGA)	78	51	69	70	**268**	**2,51**
C22	ADELANTAR O REBASAR A OTRO VEHICULO EN MOVIMIENTO EN ZONAS O SITIOS PELIGROSOS TALES COMO: (CURVAS, PUENTES, TUNELES, PENDIENTES, ETC)	78	61	55	55	**249**	**2,33**
C01	CASO FORTUITO O FUERZA MAYOR (EXPLOSION DE NEUMATICO NUEVO, DERRUMBE, INUNDACION, CAIDA DE PUENTE, ARBOL, PRESENCIA INTEMPESTIVA E IMPREVISTA DE SEMOVIENTES EN LA VIA, ETC)	44	51	44	34	**173**	**1,62**
C15	DEJAR O RECOGER PASAJEROS EN LUGARES NO PERMITIDOS	25	28	41	27	**121**	**1,13**
C27	PEATON QUE CRUZA LA CALZADA SIN RESPETAR LA SEÑALIZACION EXISTENTE (SEMAFOROS O SEÑALES MANUALES)	31	34	21	21	**107**	**1,00**
C10	CONDICIONES AMBIENTALES Y/O ATMOSFERICAS (NIEBLA, NEBLINA, GRANIZO, LLUVIA)	31	17	21	33	**102**	**0,95**
C18	CONDUCIR EN SENTIDO CONTRARIO A LA VIA NORMAL DE CIRCULACION	18	17	15	26	**76**	**0,71**
C05	FALLA MECANICA EN LOS SISTEMAS Y/O NEUMATICOS (SISTEMA DE FRENOS, DIRECCION, ELECTRONICO O MECANICO)	28	12	16	15	**71**	**0,66**
C21	MALAS CONDICIONES DE LA VIA Y/O CONFIGURACION (ILUMINACION Y DISEÑO)	11	6	10	16	**43**	**0,40**
C02	PRESENCIA DE AGENTES EXTERNOS EN LA VIA (AGUA, ACEITE, PIEDRA, LASTRE, ESCOMBROS, MADEROS, ETC)	10	7	9	7	**33**	**0,31**
C08	PESO Y VOLUMEN-NO CUMPLIR CON LAS NORMAS DE SEGURIDAD NECESARIAS AL TRANSPORTAR CARGAS	7	7	9	5	**28**	**0,26**
C17	BAJARSE O SUBIRSE DE VEHICULOS EN MOVIMIENTO SIN TOMAR LAS PRECAUCIONES DEBIDAS	7	10	6	4	**27**	**0,25**
C07	TRANSITA BAJO INFLUENCIA DE ALCOHOL, SUSTANCIAS ESTUPEFACIENTES O PSICOTROPICAS Y/O MEDICAMENTOS	3	9	10	4	**26**	**0,24**
C24	NO RESPETAR LAS SEÑALES MANUALES DEL AGENTE DE TRANSITO	9	3	5	7	**24**	**0,22**
C04	DAÑOS MECANICOS PREVISIBLES	2	5	5	3	**15**	**0,14**
C20	EL CONDUCTOR QUE DETENGA O ESTACIONE VEHICULOS EN SITIOS O ZONAS QUE ENTRAÑEN PELIGRO, TALES COMO ZONA DE SEGURIDAD, CURVAS, PUENTES, TUNELES, PENDIENTES	7	1	2	3	**13**	**0,12**
C28	DISPOSITIVO REGULADOR DE TRANSITO EN MAL ESTADO DE FUNCIONAMIENTO (SEMAFORO)	1	0	0	0		
	TOTAL	**3.044**	**2.679**	**2.513**	**2.453**	**10.689**	**100,00**

Fuente: DNCTSV, CTE, EMOV - Cuenca, Gobierno Autónomo Descentralizado Municipal de Loja, Agencia Metropolitana de Tránsito - Quito, Gobierno Autónomo Descentralizado de Manta, Municipio de Ambato, Municipio de Ibarra, Autoridad de Tránsito Municipal de Guayaquil.

ANEXO B – Guía de usuario de OMNeT++ v5.0

(El contenido del anexo se presenta en el CD adjunto)

ANEXO C– Distancias y tiempos de simulación

Tabla C.1- Distancias y tiempos para los escenarios 1 y 2

Ruta	Vehículo	Tiempo	Tiempo VANET	Diferencia T	Distancia	Distancia VANET	Diferencia D
	0,0	106,100	106,300	0,200	842,42	842,42	0
	0,1	113,200	113,400	0,200	842,42	842,42	0
	0,2	119,000	118,400	-0,600	842,42	842,42	0
	0,3	131,800	119,800	-12,000	842,42	842,42	0
	0,4	143,500	123,800	-19,700	842,42	842,42	0
	0,5	149,000	130,800	-18,200	842,42	842,42	0
	0,6	181,100	195,200	14,100	842,42	1204,46	362,04
	0,7	188,000	202,200	14,200	842,42	1204,46	362,04
	0,8	192,900	208,700	15,800	842,42	1204,46	362,04
	0,9	187,800	214,300	26,500	842,42	1204,46	362,04
	0,10	201,200	219,700	18,500	842,42	1204,46	362,04
	0,11	217,500	222,700	5,200	842,42	1028,42	186
	0,12	220,400	230,100	9,700	842,42	1028,42	186
	0,13	222,300	237,800	15,500	842,42	1028,42	186
	1,0	140,400	146,700	6,300	1070,16	1032,35	-37,81
	1,1	153,500	158,800	5,300	1070,16	1032,35	-37,81
	1,2	160,300	170,800	10,500	1070,16	1032,35	-37,81
	1,3	186,400	178,100	-8,300	1070,16	1032,35	-37,81
	1,4	201,900	304,700	102,800	1070,16	1032,35	-37,81
	1,5	210,400	315,300	104,900	1070,16	1032,35	-37,81
	1,6	215,500	321,600	106,100	1070,16	1032,35	-37,81
	1,7	219,600	325,500	105,900	1070,16	1032,35	-37,81
	1,8	223,900	329,800	105,900	1070,16	1032,35	-37,81
	1,9	234,600	333,800	99,200	1070,16	1032,35	-37,81
	1,10	240,600	337,200	96,600	1070,16	1032,35	-37,81
	1,11	246,400	342,000	95,600	1070,16	1032,35	-37,81
	1,12	251,600	192,400	-59,200	1070,16	1070,16	0
	1,13	255,700	200,600	-55,100	1070,16	1070,16	0
	2,0	186,600	161,500	-25,100	1156,04	1156,04	0
	2,1	198,900	199,700	0,800	1156,04	1156,04	0
	2,2	269,200	246,700	-22,500	1156,04	1156,04	0
	2,3	286,300	274,300	-12,000	1156,04	1156,04	0
	2,4	289,700	279,100	-10,600	1156,04	1156,04	0
	2,5	284,600	294,300	9,700	1156,04	1156,04	0
	2,6	288,200	298,500	10,300	1156,04	1156,04	0
	2,7	193,500	301,500	108,000	1156,04	1156,04	0
	2,8	196,900	304,600	107,700	1156,04	1156,04	0
	2,9	200,000	309,800	109,800	1156,04	1156,04	0
	2,10	287,000	325,400	38,400	1156,04	1156,04	0
	2,11	292,200	328,100	35,900	1156,04	1156,04	0
	3,0	146,300	187,700	41,400	1159,5	1159,31	-0,19
	3,1	154,200	242,100	87,900	1159,5	1159,31	-0,19
	3,2	161,300	252,000	90,700	1159,5	1159,31	-0,19
	3,3	172,300	270,000	97,700	1159,5	1159,31	-0,19
	3,4	180,700	283,200	102,500	1159,5	1159,31	-0,19
	3,5	184,000	286,100	102,100	1159,5	1159,31	-0,19
	3,6	187,300	290,100	102,800	1159,5	1159,31	-0,19
	3,7	205,700	222,300	16,600	1159,5	1159,5	0
	3,8	211,900	310,200	98,300	1159,5	1159,31	-0,19

	3,9	222,300	313,100	90,800	1159,5	1159,31	-0,19
	3,10	225,700	316,200	90,500	1159,5	1159,31	-0,19
	3,11	232,700	321,200	88,500	1159,5	1159,31	-0,19
	4,0	140,400	148,000	7,600	1071,46	1071,46	0
	4,1	145,800	132,500	-13,300	1071,46	957,62	-113,84
	4,2	162,300	142,700	-19,600	1071,46	957,62	-113,84
	4,3	172,900	144,700	-28,200	1071,46	957,62	-113,84
	4,4	180,700	155,000	-25,700	1071,46	957,62	-113,84
	4,5	191,200	172,800	-18,400	1071,46	1071,46	0
	4,6	197,600	182,700	-14,900	1071,46	1071,46	0
	4,7	204,400	189,200	-15,200	1071,46	1071,46	0
	4,8	214,000	193,600	-20,400	1071,46	1071,46	0
	4,9	217,900	198,500	-19,400	1071,46	1071,46	0
	5,0	199,700	192,200	-7,500	1279,21	1279,21	0
	5,1	203,300	214,100	10,800	1279,21	1279,21	0
	5,2	218,700	214,100	-4,600	1279,21	1279,21	0
	5,3	224,900	219,700	-5,200	1279,21	1279,21	0
	5,4	262,800	219,600	-43,200	1279,21	1279,21	0
	5,5	269,400	226,200	-43,200	1279,21	1279,21	0
	5,6	278,100	232,400	-45,700	1279,21	1279,21	0
	5,7	285,200	239,300	-45,900	1279,21	1279,21	0
	5,8	331,700	287,800	-43,900	1279,21	1279,21	0
	5,9	338,500	295,600	-42,900	1279,21	1279,21	0
	6,0	186,900	259,500	72,600	942,81	938,34	-4,47
	6,1	194,900	279,400	84,500	942,81	938,34	-4,47
	6,2	204,700	350,400	145,700	942,81	938,34	-4,47
	6,3	210,600	362,900	152,300	942,81	938,34	-4,47
	6,4	214,900	366,700	151,800	942,81	938,34	-4,47
	6,5	222,300	370,500	148,200	942,81	938,34	-4,47
	6,6	226,400	373,700	147,300	942,81	938,34	-4,47
	6,7	235,000	377,600	142,600	942,81	938,34	-4,47
	7,0	161,900	146,300	-15,600	1231,64	1208,04	-23,6
	7,1	173,900	153,000	-20,900	1231,64	1208,04	-23,6
	7,2	190,000	166,900	-23,100	1231,64	1208,04	-23,6
	7,3	196,500	172,600	-23,900	1231,64	1208,04	-23,6
	7,4	201,700	177,000	-24,700	1231,64	1231,64	0
	7,5	243,200	213,200	-30,000	1231,64	1231,64	0
	7,6	249,700	219,000	-30,700	1231,64	1208,04	-23,6
	7,7	254,200	224,600	-29,600	1231,64	1208,04	-23,6
	8,0	158,000	160,300	2,300	950,48	1051,93	101,45
	8,1	159,300	164,000	4,700	950,48	1051,93	101,45
	8,2	182,600	285,200	102,600	950,48	950,48	0
	8,3	188,500	292,500	104,000	950,48	950,48	0
	8,4	193,800	297,300	103,500	950,48	950,48	0
	8,5	201,100	300,600	99,500	950,48	950,48	0
	9,0	299,400	309,000	9,600	1463,36	1463,36	0
	10,0	251,900	266,300	14,400	1086,28	1086,28	0
	11,0	173,500	66,000	-107,500	1551,92	574,95	-976,97
	12,0	275,600	338,300	62,700	1122,62	1122,62	0
	13,0	196,400	209,800	13,400	1522,55	1197,76	-324,79
	14,0	132,700	125,500	-7,200	837,44	837,44	0