



INVESTIGACIÓN X

PROFUNDIZACIÓN ____

MODELO ARQUITECTÓNICO PARA LA INTERCONECTIVIDAD DE REDES VANET UTILIZANDO INTERNET DE LAS COSAS

Autor

Lewys Correa Sánchez

Tutor

Octavio José Salcedo Parra

Ph. D. En Informática

Ph. D. En Estudios políticos

UNIVERSIDAD DISTRITAL FRANCISCO JOSÉ DE CALDAS
Maestría en Ciencias de la Información y las Comunicaciones
Énfasis en Teleinformática
Bogotá, Colombia
junio de 2020

A mamá y papá, por supuesto.

Y a mis hermanas.

Contenido

RESUMEN.....	11
PALABRAS CLAVE	12
ABSTRACT.....	12
KEYWORDS.....	13
INTRODUCCIÓN.....	14
1. PROBLEMA DE INVESTIGACIÓN	16
1.1 PLANTEAMIENTO DEL PROBLEMA.....	16
1.2 JUSTIFICACIÓN	17
1.3 OBJETIVOS	19
1.3.1 OBJETIVO GENERAL	19
1.3.2 OBJETIVOS ESPECÍFICOS.....	19
1.4 RESULTADOS ESPERADOS.....	20
2. APROXIMACIÓN AL ESTADO DEL ARTE.....	21
3. DISEÑO.....	26
3.1 SCRUM	26
3.1 MODELO DE REFERENCIA IOT GENÉRICO	28
3.1.1 MODELO ABSTRACTO IOT.....	29
3.2 DISEÑO FÍSICO DE UNA RED VANET	31
3.2.1 PROTOCOLOS PARA REDES VANET.....	32
3.3 DISEÑO FÍSICO CON UNA RED VANET INTERCONECTA CON IOT	33
3.3.1 PROTOCOLOS PARA IOT	34
3.4 ARQUITECTURA PROPUESTA.....	35
3.4.1 CAPA CLIENTE	36

3.4.2	CAPA DE CONEXIÓN	36
3.4.3	CAPA DE LA NUBE	36
3.6	HERRAMIENTAS DE SIMULACIÓN	38
3.6.1	SIMULADORES DE TRÁFICO	38
3.6.2	SIMULADORES DE RED	40
3.6.3	COMPARACIÓN DE SIMULADORES.....	44
4.	SIMULACIÓN DE UNA RED VANET INTERCONECTADA CON IOT	46
5.	DISCUSIÓN DE RESULTADOS OBTENIDOS	81
5.1	CONTRASTE CON TRABAJOS RELACIONADOS.....	96
6.	CONCLUSIONES, TRABAJOS FUTUROS Y RECOMENDACIONES.....	100
6.1	CONCLUSIONES.....	100
6.2	TRABAJOS FUTUROS	102
6.3	APORTES DE LA INVESTIGACIÓN.....	102
6.4	RECOMENDACIONES	102
7.	REFERENCIAS.....	104
	ANEXOS.....	111
	ANEXO 1: MODELO DE SIMULACIÓN PARA RED VANET TRADICIONAL	111
	ANEXO 2: MODELO DE SIMULACIÓN PARA RED VANET INTERCONECTADA CON IOT	115
	ANEXO 3: RESULTADOS DE LA SEGUNDA SERIE DE SIMULACIONES	119

Lista de Tablas

Tabla 1: Sprint 1	27
Tabla 2: Sprint 2	27
Tabla 3: Sprint 3	27
Tabla 4: Sprint 4	28
Tabla 5: Sprint 5	28
Tabla 6: Sprint 6	28
Tabla 7: Sprint 7	28
Tabla 8: Sprint 8	28
Tabla 9: Características de enlace de datos inalámbricos para el protocolo 802.11.p	32
Tabla 10: Comparación simuladores de red.....	45
Tabla 11: Parámetros de las simulaciones de una red Vanet	47
Tabla 12: Parámetros de las simulaciones de una red Vanet interconectada con IoT.....	49
Tabla 13: Pacject Delivary Ratio de las simulaciones realizadas con una red Vanet Tradicional	81
Tabla 14:Pacject Delivary Ratio de las simulaciones realizadas con una red Vanet interconecta con IoT	82
Tabla 15: Promedio End-End delay de las simulaciones realizadas con una red Vanet tradicional	88
Tabla 16: Promedio End-End delay de las simulaciones realizadas con una red Vanet interconecta con IoT	88
Tabla 17: Packet Loss de las simulaciones realizadas con una red Vanet tradicional.....	89
Tabla 18: Packet Loss de las simulaciones realizadas con una red Vanet interconecta con IoT.....	89
Tabla 19: Packet Loss Ratio de las simulaciones realizadas con una red Vanet tradicional	93
Tabla 20: Packet Loss Ratio de las simulaciones realizadas con una red Vanet interconecta con IoT.....	93
Tabla 21: Pacject Delivary Ratio de las segundas simulaciones realizadas con una red Vanet Tradicional.....	119
Tabla 22:Pacject Delivary Ratio de las simulaciones realizadas con una red Vanet interconecta con IoT	119

Tabla 23: Promedio End-End delay de las segundas simulaciones realizadas con una red Vanet tradicional	125
Tabla 24: Promedio End-End delay de las segundas simulaciones realizadas con una red Vanet interconecta con IoT	126
Tabla 25: Packet Loss de las segundas simulaciones realizadas con una red Vanet tradicional	126
Tabla 26: Packet Loss de las segundas simulaciones realizadas con una red Vanet interconecta con IoT	126
Tabla 27: Packet Loss Ratio de las segundas simulaciones realizadas con una red Vanet tradicional	129
Tabla 28: Packet Loss Ratio de las segundas simulaciones realizadas con una red Vanet interconecta con IoT	130

Lista de Figuras

Figura 1: El IoT puede considerarse una red de redes	18
Figura 2: Internet of Things "nació" entre 2008 y 2009	18
Figura 3: Modelo abstracto IoT.....	29
Figura 4: Red Vanet tradicional	31
Figura 5: Red Vanet interconectada con IoT	33
Figura 6: Arquitectura propuesta de red Vanet interconecta con IoT	37
Figura 7: Sumo	38
Figura 8 Estructura del simulador VanetMobiSim	39
Figura 9 Estructura en diagrama de bloques simulador ns3	40
Figura 10 Estructura en diagrama de bloques simulador QUALNET	41
Figura 11 Estructura en diagrama de bloques simulador JiST	42
Figura 12 Ejemplo de simulación en OMNET++	43
Figura 13 Estructura en diagrama de bloques simulador ns2	44
Figura 14: Obtención del mapa para la zona 1.....	50
Figura 15: Conversión del mapa de la zona 1 a redes de carreteras.....	51
Figura 16: Importación de polígonos adicionales del mapa de la zona 1	51
Figura 17: Ejecución del programa randomTrips.py para la simulación de la zona 1	52
Figura 17: Creación del fichero zona1.sumo.cfg	52
Figura 18: Ejecución del programa sumo-gui para la simulación de la zona 1	53
Figura 20: Mapa en ejecución de la zona 1.....	53
Figura 21: Ejecución del programa sumo -c para la simulación de la zona 1	54
Figura 22: Ejecución del programa traceExporter.py para la simulación de la zona 1.....	54
Figura 23: Ejecución del programa cbrgen.tcl para la generación de tráfico de la zona 1	55
Figura 24: Ejecución de la simulación para la zona 1 de una red Vanet.....	55
Figura 25: Ejecución de la simulación para la zona 1 de una red Vanet interconectada con IoT.....	56
Figura 26: Obtención del mapa para la zona 2.....	56
Figura 27: Conversión del mapa de la zona 2 a redes de carreteras.....	57
Figura 28: Importación de polígonos adicionales del mapa de la zona 2	57
Figura 29: Ejecución del programa randomTrips.py para la simulación de la zona 2	58
Figura 30: Creación del fichero zona2.sumo.cfg	58

Figura 31: Ejecución del programa sumo-gui para la simulación de la zona 2	59
Figura 32: Mapa en ejecución de la zona 2.....	59
Figura 33: Ejecución del programa sumo -c para la simulación de la zona 2	60
Figura 34: Ejecución del programa traceExporter.py para la simulación de la zona 2.....	60
Figura 35: Ejecución del programa cbrgen.tcl para la generación de tráfico de la zona 2.	61
Figura 36: Ejecución de la simulación para la zona 2 de una red Vanet.....	61
Figura 37: Ejecución de la simulación para la zona 2 de una red Vanet interconectada con IoT.....	62
Figura 38: Obtención del mapa para la zona 3.....	62
Figura 39: Conversión del mapa de la zona 3 a redes de carreteras	63
Figura 40: Importación de polígonos adicionales del mapa de la zona 3	63
Figura 41: Ejecución del programa randomTrips.py para la simulación de la zona 4	64
Figura 42: Creación del fichero zona3.sumo.cfg	64
Figura 43: Ejecución del programa sumo-gui para la simulación de la zona 3	65
Figura 44: Mapa en ejecución de la zona 3.....	65
Figura 45: Ejecución del programa sumo -c para la simulación de la zona 3	66
Figura 46: Ejecución del programa traceExporter.py para la simulación de la zona 3.....	66
Figura 47: Ejecución del programa cbrgen.tcl para la generación de tráfico de la zona 3.	67
Figura 48: Ejecución de la simulación para la zona 3 de una red Vanet.....	67
Figura 49: Ejecución de la simulación para la zona 3 de una red Vanet interconectada con IoT.....	68
Figura 50: Obtención del mapa para la zona 4.....	68
Figura 51: Conversión del mapa de la zona 4 a redes de carreteras	69
Figura 52: Importación de polígonos adicionales del mapa de la zona 4.....	69
Figura 53: Ejecución del programa randomTrips.py para la simulación de la zona 4	70
Figura 54: Creación del fichero zona4.sumo.cfg	70
Figura 55: Ejecución del programa sumo-gui para la simulación de la zona 4	71
Figura 56: Mapa en ejecución de la zona 4	71
Figura 57: Ejecución del programa sumo -c para la simulación de la zona 4	72
Figura 58: Ejecución del programa traceExporter.py para la simulación de la zona 4.....	72
Figura 59: Ejecución del programa cbrgen.tcl para la generación de tráfico de la zona 4.	73
Figura 60: Ejecución de la simulación para la zona 4 de una red Vanet.....	73

Figura 61: Ejecución de la simulación para la zona41 de una red Vanet interconectada con IoT.....	74
Figura 62: Obtención del mapa para la zona 5.....	74
Figura 63: Conversión del mapa de la zona 5 a redes de carreteras.....	75
Figura 64: Importación de polígonos adicionales del mapa de la zona 5.....	75
Figura 65: Ejecución del programa randomTrips.py para la simulación de la zona 5.....	76
Figura 66: Creación del fichero zona5.sumo.cfg.....	76
Figura 67: Ejecución del programa sumo-gui para la simulación de la zona 5.....	77
Figura 68: Mapa en ejecución de la zona 5.....	77
Figura 69: Ejecución del programa sumo -c para la simulación de la zona 5.....	78
Figura 70: Ejecución del programa traceExporter.py para la simulación de la zona 5.....	78
Figura 71: Ejecución del programa cbrgen.tcl para la generación de tráfico de la zona 5.....	79
Figura 72: Ejecución de la simulación para la zona 5 de una red Vanet.....	79
Figura 73: Ejecución de la simulación para la zona 5 de una red Vanet interconectada con IoT.....	80
Figura 74: PDF de la simulación para la Zona 1.....	82
Figura 75: PDF de la simulación para la Zona 2.....	83
Figura 76: PDF de la simulación para la Zona 3.....	83
Figura 77: PDF de la simulación para la Zona 4.....	84
Figura 78: PDF de la simulación para la Zona 5.....	84
Figura 79: Throughput de generación de bits para la Zona 1.....	85
Figura 80: Throughput de generación de bits para la Zona 2.....	86
Figura 81: Throughput de generación de bits para la Zona 3.....	86
Figura 82: Throughput de generación de bits para la Zona 4.....	87
Figura 83: Throughput de generación de bits para la Zona 5.....	87
Figura 84: Packet Loss de la simulación para la Zona 1.....	90
Figura 85: Packet Loss de la simulación para la Zona 2.....	90
Figura 86: Packet Loss de la simulación para la Zona 3.....	91
Figura 87: Packet Loss de la simulación para la Zona 4.....	91
Figura 88: Packet Loss de la simulación para la Zona 5.....	92
Figura 89: Packet Loss Ratio de la simulación para la Zona 1.....	93
Figura 90: Packet Loss Ratio de la simulación para la Zona 2.....	94
Figura 91: Packet Loss Ratio de la simulación para la Zona 3.....	94

Figura 92: Packet Loss Ratio de la simulación para la Zona 4.....	95
Figura 93: Packet Loss Ratio de la simulación para la Zona 5.....	95
Figura 94: The packet delivery ratio versus el número de vehículos en la red.....	96
Figura 95: PDF de la simulación 2 para la Zona 1	120
Figura 96: PDF de la simulación 2 para la Zona 2	120
Figura 97: PDF de la simulación 2 para la Zona 3	121
Figura 98: PDF de la simulación 2 para la Zona 4	121
Figura 99: PDF de la simulación 2 para la Zona 5	122
Figura 100: Throughput de generación de bits de la simulación 2 para la Zona 1	123
Figura 101: Throughput de generación de bits de la simulación 2 para la Zona 2	123
Figura 102: Throughput de generación de bits de la simulación 2 para la Zona 3.....	124
Figura 103: Throughput de generación de bits de la simulación 2 para la Zona 4.....	124
Figura 104: Throughput de generación de bits de la simulación 2 para la Zona 5.....	125
Figura 105: Packet Loss de la segunda simulación para la Zona 1	127
Figura 106: Packet Loss de la segunda simulación para la Zona 2	127
Figura 107: Packet Loss de la segunda simulación para la Zona 3	128
Figura 108: Packet Loss de la segunda simulación para la Zona 4	128
Figura 109: Packet Loss de la segunda simulación para la Zona 5	129
Figura 110: Packet Loss Ratio de la segunda simulación para la Zona 1	130
Figura 111: Packet Loss Ratio de la segunda simulación para la Zona 2.....	131
Figura 112: Packet Loss Ratio de la segunda simulación para la Zona 3.....	131
Figura 113: Packet Loss Ratio de la segunda simulación para la Zona 4.....	132
Figura 114: Packet Loss Ratio de la segunda simulación para la Zona 5.....	132

RESUMEN

A diario se escucha en las noticias y se lee en la prensa escrita acerca del caos vehicular, accidentes de tránsito o paso restringido de las vías principales de las grandes y medianas ciudades, lo cual afecta considerablemente el tiempo de movilidad de las personas. Al verse afectado el tiempo de movilidad, servicios que resultan ser importantes para la ciudadanía como ambulancias, policías, bomberos, entre otros, se ven retrasados al momento de necesitarse. Por otro lado, las personas que se transportan en vehículos particulares pierden tiempo en carreteras y autopistas.

Las redes ad-hoc vehiculares, Vanet (vehicular ad-hoc network), entre sus propósitos buscan incrementar la seguridad en las carreteras, la eficiencia del transporte y reducir el impacto ambiental. En este tipo de redes los vehículos actúan como nodos y en ocasiones existe una estación base en carretera. En estas redes hay dos tipos de comunicaciones: vehículo a vehículo y de vehículo a la estación base.

La tecnología de Internet de las Cosas pretende la interconexión de los objetos y cosas cotidianas con y por medio de internet. Esta tecnología es aplicable a múltiples áreas, como la educación, salud, ciudades, hogares, entre otras. En este trabajo se propone el diseño de un modelo arquitectónico que permita la interconectividad entre estas dos tecnologías. En particular, se presenta un enfoque sistemático para diseñar la arquitectura haciendo uso de la metodología Scrum. Luego se valida el modelo de arquitectura aplicándolo a cinco zonas de alta congestión de la ciudad de Bogotá (Colombia), comparando los resultados con arquitecturas propuestas en la literatura científica. El objetivo del modelo de arquitectura es proporcionar una base para la implementación de redes Vanet. Para validar el modelo, se implementaron una serie de simulaciones en NS2 (Network Simulator). Estas simulaciones permitieron demostrar la validez del modelo de arquitectura para ser aplicada en la vida real.

La evaluación de métricas como Packet Delivery Ratio (PDF), Throughput, End-End delay, Packet Loss y Packet Loss Ratio fueron usadas para hacer la evaluación y comparación del modelo planteado de una red Vanet interconectada con IoT versus una red Vanet tradicional. Con estas métricas se puede observar que el Packet Delivery Ratio de las 5

zonas interconectadas con IoT fue menor con respecto a una red Vanet tradicional. Con el resto de métricas también se obtuvieron resultados favorables que validan el modelo planteado.

Se realizan una serie de recomendaciones y se enumeran las conclusiones que la investigación arrojó, así como trabajos futuros.

PALABRAS CLAVE

Internet de las cosas, Redes Vanet, Interconectividad, Modelo arquitectónico.

ABSTRACT

The news is heard daily and the written press is read about vehicular chaos, traffic accidents or restricted passage of the main roads of large and medium-sized cities, considerably affecting the vehicle's mobility time. When mobility time is affected, services of the utmost importance for citizens, such as ambulances, police, firefighters, among others, are delayed when needed, and at the same time people who are transported in private vehicles lose time on roads and highways.

Ad-hoc vehicular networks, known as Vanet networks, seek to increase road safety, transport efficiency and at the same time reduce environmental impact. In this type of network, vehicles act as nodes and sometimes there is a road base station. In these networks there are two types of communications: vehicle to vehicle and vehicle to the base station.

The Internet of Things technology aims to interconnect everyday objects and things with and through the internet. This technology is applicable to multiple areas, such as education, health, cities, homes, among others. This paper proposes the design of an architectural model that allows interconnectivity between these two technologies. In particular, a

systematic approach to design the architecture using the Scrum methodology is presented. Then the architecture model is validated by applying it to three areas of high congestion in the city of Bogotá, Colombia, comparing the results with architectures proposed in the scientific literature. The objective of the architecture model is to provide a basis for the implementation of Vanet networks. To validate the model, a series of simulations were implemented in NS2 (Network Simulator). These simulations allow to demonstrate the validity of the architecture model to be applied in real life.

The evaluation of metrics such as Packet Delivery Ratio (PDF), Throughput, End-End delay, Packet Loss and Packet Loss Ratio were used to evaluate and compare the proposed model of a Vanet network interconnected with IoT versus a traditional Vanet network. With these metrics it can be seen that the Packet Delivery Ratio of the 5 zones interconnected with IoT was lower compared to a traditional Vanet network. With the other metrics, favorable results were also obtained that validate the proposed model.

A series of recommendations are made and the conclusions that the investigation threw are enumerated, as well as future works.

KEYWORDS

Architectural model, Interconnectivity, Internet of things, NS2, Scrum, Vanet.

INTRODUCCIÓN

Con los avances que surgieron a partir de la rueda, el hombre basó su movilización en bicicletas, motocicletas y vehículos automotores, pero el crecimiento acelerado de estos últimos en las ciudades lleva a la congestión vehicular de las principales vías y autopistas, generando así una problemática sin solución aparente porque solo se nota la implementación de políticas de restricción del uso de los vehículos y/o motivar a usar al transporte público y medios alternativos como las bicicletas. Las sociedades no pueden detenerse y desestimar el uso de los vehículos particulares, sino que deben buscar nuevas alternativas o mejorar las soluciones y propuestas existentes.

Los avances tecnológicos alcanzados en los últimos años dan indicios que permiten vislumbrar que el futuro de la tecnología va a reemplazar el concepto de combinar computadoras, sensores y redes para monitorear y controlar diferentes dispositivos, que ha predominado durante décadas bajo el esquema de la computación tradicional, La popularización de internet y los avances en las telecomunicaciones, como los sensores embebidos en objetos cotidianos con identificadores y dispositivos de posicionamiento, han permitido la concepción de sistemas que tienen la capacidad de capturar información de su entorno y de influir en el desencadenando acciones (Montes & Clemente, 2016)

Este concepto resulta ser fundamental dentro del paradigma Internet de las Cosas donde muchos objetos en el medio pueden cooperar entre ellos para alcanzar objetivos en común. La fortaleza principal de IoT radica en el impacto sobre los aspectos cotidianos de los potenciales usuarios (Atzori et al., 2010). De hecho, este concepto de conectar las cosas que rodean la vida del hombre a Internet se está arraigando cada vez más y se espera que aproximadamente en dos años la cantidad de dispositivos que estarán conectados ascienda a 50 mil millones y en siete años a 100 mil millones.

En este trabajo se propone un modelo arquitectónico que permita mejorar el desempeño del tráfico vehicular controlado por redes Vanet integradas al Internet de las Cosas. El modelo arquitectónico se generó a partir de un proceso de estudio de diferentes propuestas, iniciativas y protocolos relacionados con las Redes Vanet y el internet de las cosas, en la

literatura científica, logrando así desarrollar un diseño y su posterior implementación, y validarlo por medio de simulaciones en el software especializado de redes: NS2.

1. PROBLEMA DE INVESTIGACIÓN

1.1 PLANTEAMIENTO DEL PROBLEMA

A lo largo de la historia se ha demostrado que la evolución de la sociedad está ligada a la forma en cómo se moviliza el hombre. La invención de la rueda se puede considerar como una de las creaciones más importantes en la historia de la humanidad, ya que dio paso al éxito de distintas civilizaciones y revoluciones y en la actualidad es pieza fundamental de los medios de transporte terrestre e inclusive en el transporte aéreo.

Hoy en día, el hombre se moviliza en vehículos automotores, motocicletas y bicicletas. Pero el crecimiento acelerado del número de los vehículos ciudades ha llevado a la congestión vehicular en las principales vías y autopistas e inclusive al colapso de estas. Creando un problema sin solución aparente más allá de restringir el uso del carro particular o motivar a usar al transporte público o medios alternativos. En la actualidad, se puede decir que la movilidad engloba diferentes aspectos, que van desde la planeación y gestión de las ciudades, estado de las vías, sistemas de transporte público hasta la cantidad de vehículos que tiene una ciudad.

Según cálculos de la Secretaria Distrital de Ambiente para el año 2017 en Bogotá hubo 2.182.578 vehículos (Secretaría Distrital de Ambiente, 2018). Las grandes y medianas ciudades a la fecha presentan problemas con el tráfico vehicular, lo cual dificulta la movilización de los ciudadanos, reduciendo el tiempo de respuestas de servicios como: policía, ambulancias y carros de emergencias, por ejemplo la velocidad promedio de vehículos particulares en Bogotá en horas pico no superó los 20 km/h para el año 2018 (Alcaldía Mayor de Bogota, 2018) considerando esta, una velocidad baja para las necesidades del hombre moderno.

Teniendo en cuenta el continuo crecimiento de vehículos en carreteras de las grandes y medianas ciudades y los avances de la tecnología con las distintas ventajas que estas ofrecen, se hizo necesario integrar las nuevas tecnologías al servicio del control vehicular. (Sanchez & Cobo, 2014)

Una solución relacionada con las nuevas tecnologías para darle solución a esta problemática fueron las Redes Vanet, un tipo de red de comunicación que utiliza a los vehículos como nodos de la red, donde los vehículos disponen de más y mejor información sobre el estado del tráfico, y pueden acceder a servicios y datos que mejoran la comodidad del viaje para los pasajeros, integrándola con las bondades del Internet de las cosas (IoT), una tecnología basada en una capa de conexión digital para objetos de la vida cotidiana, es decir del mundo real; donde estos tienen como función la transmisión de datos a través de una red, sin la necesidad de que exista la relación o interacción entre persona – persona o persona – computador.

Y al controlar la situación planteada se reduciría la congestión vehicular, permitiendo así mejorar el tiempo de movilización.

1.2 JUSTIFICACIÓN

Debido al gran tamaño del parque automotriz de ciudades como Bogotá y la simplicidad de las vías es indispensable que los vehículos sean inteligentes para así tener armonía en las carreteras. Las Redes Vanet tratan de abordar esta problemática, pero es una tecnología en desarrollo y de alto costo. Por lo tanto, se hace necesario que estas estén al alcance de todos los vehículos por medio de la tecnología que lo está cambiando todo, el internet de las cosas. A lo largo de la historia, el hombre ha creado inventos que le han ayudado en su paso por la Tierra, como: el fuego, la escritura y la rueda; fundamentales para poder dominar al resto de las especies. Pero esto se puede notar que Internet es una de las creaciones más recientes de suma importancia y potente en la historia de la humanidad.

Dada la importancia que tiene IoT representando la próxima evolución del Internet, donde se supone un enorme avance en la capacidad para recopilar, analizar y distribuir datos que se pueden convertir en información y en conocimiento; la interconexión con redes Vanet ayudaría a obtener un mejor desempeño en la actualidad.

IoT se compone de un conjunto disperso de redes diseñadas a la medida del usuario final. Por ejemplo, los vehículos cuentan con diversas redes para controlar el funcionamiento del motor, la seguridad, los sistemas de comunicaciones, etc. A medida que evoluciona el IoT, estas redes y muchas otras, se conectarán y contarán con mayores funciones de seguridad,

análisis y gestión, como se puede observar en la Figura 1. Esto permitirá que el IoT pueda contribuir a que las personas consigan sus objetivos.

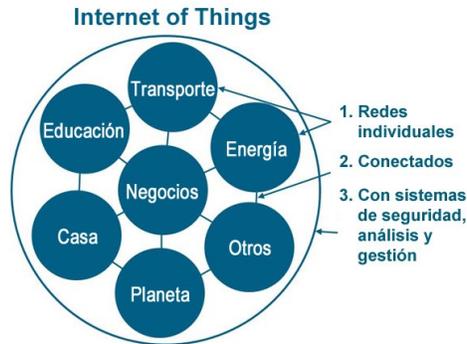


Figura 1: El IoT puede considerarse una red de redes
Fuente: (Cisco IBSG, 2011)

En 2003, había aproximadamente 6300 millones de personas en el planeta y 500 millones de dispositivos conectados a Internet (U.S. Census Bureau, 2012). Pero el crecimiento explosivo de los smartphones y tablets elevó el número de dispositivos conectados a Internet a 12.500 millones en 2010, mientras que la población mundial llegó a los 6800 millones, lo que significa que el número de dispositivos conectados por persona era de más de uno (1,84). En la Figura 2 se puede observar la irrupción de los dispositivos conectados a Internet con respecto a la población mundial. Dentro de estos dispositivos, se encuentra el poder implementar redes Vanet. Dado que las Redes Vanet permiten establecer comunicación en tiempo real entre vehículos con el fin de prevenir accidentes de tránsito y mejorar la seguridad de los usuarios.

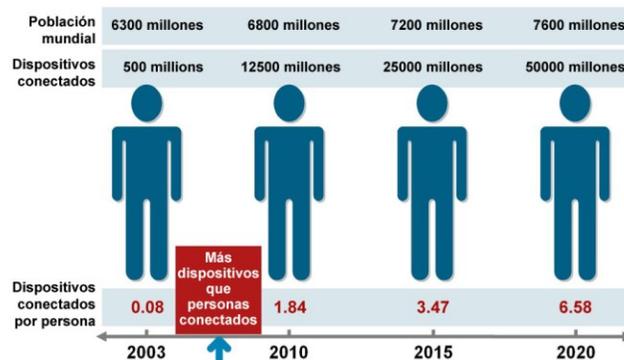


Figura 2: Internet of Things "nació" entre 2008 y 2009
Fuente: (Cisco IBSG, 2011)

1.3 OBJETIVOS

1.3.1 OBJETIVO GENERAL

- Diseñar un modelo arquitectónico que permita mejorar el desempeño del tráfico vehicular controlado por redes Vanet integradas al Internet de las Cosas.

1.3.2 OBJETIVOS ESPECÍFICOS

- Evaluar propuestas y protocolos necesarios para diseñar el modelo arquitectónico en IoT basados en W3C¹ para la interconectividad de la comunicación de redes Vanet en IoT.
- Evaluar modelos formales que soporten el modelo arquitectónico para la interconectividad de redes Vanet por medio de IoT
- Diseñar un modelo conceptual de la arquitectura de la interoperabilidad de redes Vanet e IoT.
- Implementar y simular un prototipo usando la arquitectura de la interoperabilidad de redes Vanet e IoT.

¹ W3C: World Wide Web Consortium, consorcio internacional que genera recomendaciones y estándares que aseguran el crecimiento de la World Wide Web a largo plazo.

1.4 RESULTADOS ESPERADOS

El impacto del proyecto se presentará a nivel local, nacional e internacional, ya que se pretende hacer una contribución al diseño de las soluciones Vanet interconectadas con IoT, además del fortalecimiento de la capacidad investigativa del grupo Internet Inteligente.

Resultados esperados:

- El desarrollo de un modelo de arquitectura de redes Vanet que permita la interconexión con la tecnología IoT enfocada en proporcionar una alternativa de solución para su asegure su aplicabilidad en un amplio rango de situaciones del dominio de redes Vanet por medio de IoT.
- Un modelo referencial de Vanet interconectado con IoT.
- Validación del modelo de arquitectura de redes Vanet interconectado con IoT por medio de simulaciones en software especializado.

2. APROXIMACIÓN AL ESTADO DEL ARTE

En esta sección se desarrolla el primer paso de la metodología de diseño del modelo arquitectura, donde se hace una investigación del estado del arte de trabajos relacionados con redes Vanet, protocolos de enrutamiento de redes Vanet y protocolos.

Un informe realizado por Statista, señala que el número de vehículos tanto de servicio público como de servicio particular, utilizados a nivel mundial para el año 2015 fue 1,282 millones de vehículos(Organisation Internationale des Constructeurs d'Automobiles (OICA), 2017) y para año 2050 se espera que esta cifra aumente hasta 2, 500 millones(Voelcker, 2011). Este crecimiento significativo del número de vehículos, indudablemente genera que la congestión del tráfico en las carreteras aumente (Al-Sultan et al., 2014). En otro informe realizado por INRIX y *Centre for Economics and Business Research* (Centro de investigación Economía y Empresarial) reveló que las congestiones de tráfico vehicular le cuestan a los ciudadanos americanos 124 mil millones de dólares en pérdidas directas e indirectas, y para el año 2030 se espera que sean alrededor de 186 mil millones de dólares (INRIX, 2014).

El aumento significativo del número de vehículos conectados a la Internet de las Cosas (IoT) ha hecho que aparezcan nuevos requerimientos para las redes vehiculares Ad-Hoc (VANETs) tales como: comunicaciones inter-vehiculares, infraestructura-vehículos, internet-vehículos, vehículos-dispositivos personales e intra-vehiculares (Contreras et al., 2017). Por lo cual las redes VANET quedan limitadas a la hora de procesar, analizar y evaluar toda la información recolectada por los mismos vehículos y también de dispositivos móviles y sensores. Dados estos inconvenientes, los vehículos deben convertirse en "cosas" inteligentes equipados con: una plataforma multi-sensor, un conjunto de tecnologías de comunicación, conectividad basada en IP y con conexión directa o indirecta a otros vehículos y todos los dispositivos de su ambiente. Por estas razones el concepto de Vanet se debe convertir en Internet de Vehículos (IoV). IoV integra: la creación de redes vehiculares y la inteligencia vehicular (Yang et al., 2014) haciendo referencia a vehículos, seres humanos, componentes de la infraestructura de transporte y a un conjunto de dispositivos que se encontrarían dentro del ambiente(Nitti et al., 2014)(Yang et al., 2014).

Una investigación de mercado realizada en los Estados Unidos arrojó que loV podría crear un total de 400 mil nuevos empleos en ese país (Mai et al., 2011), siendo así loV todo un mundo tecnológico por explorar y con grandes expectativas económicas a nivel mundial, porque además de los puestos de trabajos, se estima que con esta nueva tecnologías se realicen ventas alrededor de 115.26 billones de euros para el año 2020 (Strategy & Formely Booz & Company, n.d.).

En (Dandala et al., 2017), los autores muestran la aceptación que ha tenido el IoT para la gestión del tráfico, más sin embargo resaltan que es una infraestructura que se encuentra en construcción. Dentro del trabajo realizado encontraron ventajas como la toma de decisiones a través de algoritmos pre-incorporados, dentro de estas se encuentran: control del tráfico, detección de personas en las vías, respuestas de emergencias, detección de accidentes, entre otras. A estas ventajas se le suman algunas desventajas tales como: la seguridad en las comunicaciones inalámbricas, donde los hackers pueden aprovechar las vulnerabilidades del sistema como también fallas en la red por la carga que pueden generar muchos vehículos en las carreteras.

En (Golestan et al., 2016), los autores crean una arquitectura de 3 niveles. El primer nivel o capa contiene todos los sensores del vehículo que recogen datos ambientales, los cuales detectan eventos específicos como patrones de conducción, situación de los vehículos, condiciones ambientales y otros. En el segundo nivel, corresponde a la capa de comunicación, la cual soporta distintos modos de comunicación inalámbricas como V2V (vehículo a vehículo), V2I (vehículo a infraestructura), V2P (vehículo a peatón) y V2S (vehículo a sensor); y la tercera capa incluye herramientas estadísticas para el procesamiento y almacenamiento de los datos generados. Con esta capa se busca tomar decisiones basadas en diferentes sistemas y tecnologías como big data, WSN (wireless sensor network), cloud computing, entre otras.

En (Luo et al., 2018), los autores investigaron la captación y distribución de contenido en 5G-VANET. Presentando una arquitectura jerárquica basada en la computación de borde para una distribución eficiente de datos vehiculares de gran volumen, donde los recursos de computación, comunicación, almacenamiento y control se distribuyen en cualquier lugar a lo largo de la carretera desde la nube a las cosas así, terminan los recursos se distribuyéndose en todos los nodos. También proponen un esquema de captación previa

de múltiples factores y múltiples lugares para cumplir con el rápido cambio de topología y el tráfico desequilibrado y utilizan un enfoque basado en la teoría de grafos para resolver la distribución de contenido transformándola en un problema de conjunto independiente ponderado máximo.

En (Qi et al., 2018) afirman que las redes celulares de quinta generación (5G) tienen el potencial de proporcionar comunicaciones de alta capacidad y baja latencia para vehículos en entornos altamente móviles y que la integración de Vanet basadas en clúster con redes celulares 5G es beneficiosa para ahorrar recursos de espectro escasos, prevenir la congestión de la red y reducir la pérdida de paquetes. Y como es un problema difícil encontrar un algoritmo de agrupamiento efectivo que tenga una alta estabilidad y se adapte a las VANET dinámicas presentaron un algoritmo de agrupación con conciencia social habilitado para SDN en el sistema 5G-VANET, que explota un modelo de predicción de patrón social (es decir, rutas futuras de vehículos) en Para mejorar la estabilidad de los grupos.

En (Chang et al., 2016) proponen un nuevo esquema de acceso al canal, denominado acceso múltiple con detección de portador basado en la primera fecha límite (EDF-CSMA). EDF-CSMA basado en EDCA ajusta dinámicamente la prioridad de la transmisión en tiempo real para evitar colisiones e introduce una política de control de admisión de acuerdo con las limitaciones de tiempo para proporcionar QoS garantizada en entornos multicanal. Los investigadores llevaron a cabo un modelo analítico para estudiar y comparar la utilización del canal de EDF-CSMA y el método de acceso controlado al canal (HCCA) de la función de coordinación híbrida (HCF) con reconocimiento de QoS y el resultado mostró que EDF-CSMA mejora el 60 por ciento de la utilización del canal.

En (L. Xie et al., 2019) afirman que el moderno sistema de transporte inteligente brinda no solo nuevas oportunidades para los servicios de Internet de las Cosas (IoT) vehiculares, sino también nuevos desafíos para las redes ad-hoc vehiculares (VANET) y que se necesita un esquema de seguridad práctico y confiable para manejar la gestión de la confianza y al mismo tiempo preservar la privacidad del usuario. Los investigadores además abordaron el problema de seguridad y privacidad en el sistema de transporte y el entorno de IoT vehicular en 5G-VANET habilitado para SDN, donde los resultados de la simulación muestran que los nodos o mensajes vehiculares maliciosos se pueden detectar de forma exitosa mientras

que la sobrecarga y el impacto en el rendimiento de la red son aceptables para escenarios a gran escala.

En (Huang et al., 2017) afirman que la difusión de información en VANET se centra principalmente en la propagación de mensajes de conciencia cooperativa y mensajes de notificación ambiental descentralizados en el canal de control, y que la propagación de múltiples fuentes en los entornos multicanal generará un conflicto de retransmisión de múltiples fuentes (MRC) que conduce a una mayor latencia y una menor velocidad de propagación de información (IPS). Los investigadores proponen un algoritmo de contención de retransmisión basado en probabilidad de éxito (SPRCA) para mitigar el impacto de MRC en entornos de canal WAVE. SPRCA presenta una métrica vital, probabilidad de éxito, para que cada vehículo determine si se une al conjunto de candidatos de retransmisión de cierta fuente y ejecuta el algoritmo de contención de retransmisión.

En (Xia et al., 2018) confirman que en las Vanet el enrutamiento es un problema difícil debido a la tipología de red que cambia rápidamente causada por la alta movilidad en ambos extremos de la transmisión . Por lo que propusieron un semáforo codicioso y un protocolo de enrutamiento consciente de la cola (GTLQR) que considera conjuntamente la conectividad de la calle, la calidad del canal, la distancia relativa y el retraso en la cola para aliviar la pérdida de paquetes causada por la agrupación de vehículos en la intersección y equilibrar el tráfico carga entre vehículos. El protocolo propuesto supera a TLRC y GLSR-L en términos de relación de entrega de paquetes y retraso de extremo a extremo.

Los investigadores en (Seliem et al., 2018) propusieron un protocolo de enrutamiento que utiliza los drones de infraestructura para impulsar las comunicaciones VANET para lograr un retraso mínimo en la entrega de paquetes de vehículo a dron y también una expresión de forma cerrada para la distribución de probabilidad del retraso de entrega de paquetes de vehículo a dron en una carretera de dos vías y concluyeron que existe una relación entre la densidad del dron, la densidad y la velocidad del vehículo y el retraso en la entrega de paquetes de vehículo a dron.

En (Y. Xie et al., 2018) los investigadores propusieron un conjunto de metodologías de modelado y optimización de unidifusión 802.11p para determinar los parámetros óptimos de una Vent sin monitorear continuamente los vehículos cercanos. Esto lo lograron

integrando un modelo de tráfico urbano estocástico en el análisis y luego realizando una optimización de capa cruzada para cada nodo de red para reducir las colisiones de paquetes. El rango de transmisión óptimo y el tamaño de la ventana de contención en diferentes ubicaciones se derivan en función del perfil de velocidad espacio-temporal y se dan a conocer a los vehículos que ingresan. Y estos ayudan a los vehículos a configurar su potencia de transmisión y la velocidad correspondiente al ingresar a un segmento de carretera.

En (Wu et al., 2020) presentaron un novedoso protocolo de enrutamiento para Vanet denominado enrutamiento basado en el tráfico y el aprendizaje Q (QTAR) asistido por RSU. Combinando las ventajas del enrutamiento geográfico con la información estática de la hoja de ruta, QTAR aprende la información del tráfico del segmento de carretera basada en el algoritmo Q-learning. En QTAR, una ruta de enrutamiento consiste en múltiples segmentos de ruta de conexión de alta confiabilidad seleccionados dinámicamente que permiten que los paquetes lleguen a su destino de manera efectiva. Los resultados experimentales de simulación detallados demuestran que QTAR supera a los protocolos de enrutamiento de tráfico existentes, en términos de relaciones de entrega de paquetes promedio 7.9% y 16.38% más altas que las de enrutamiento de tráfico confiable (RTAR) y enrutamiento codicioso de tráfico codicioso (GyTAR) en escenarios de alta densidad vehicular y 30.96% y 46.19% menos de demoras promedio de extremo a extremo con respecto a RTAR y GyTAR en escenarios de baja densidad vehicular, respectivamente.

En (Tan & Chung, 2020) los investigadores propusieron un esquema de autenticación segura y gestión de claves. En el esquema de autenticación sin certificado aplica la clave de sesión independiente para cada vehículo para evitar interferencias. E hicieron uso de blockchain para la construcción de claves del grupo V2V, lo que hace se proporcione un acuerdo de membresía de grupo en tiempo real con actualización eficiente de la clave de grupo. Los investigadores buscaron adoptar un nuevo modelo de sistema Vanet con una infraestructura informática de vanguardia para ofrecer una capacidad informática y de almacenamiento adecuada en comparación con la estructura Vanet tradicional.

3. DISEÑO

En este capítulo se desarrolló el diseño de la red Vanet interconectada por IoT. Para esto se hizo uso de la metodología rápida Scrum. Se presenta el diseño físico de una red Vanet y el diseño físico de una red Vanet interconectada por medio de IoT. Y finalmente se realiza una mención de las herramientas de simulación, dentro de estas los simuladores de tráfico y los simuladores de red y una comparación entre ellos.

3.1 SCRUM

Scrum es un marco de trabajo ágil. Su usabilidad es más común en trabajos de tecnología que impliquen desarrollo de aplicaciones y puesta en producción, no obstante, eso no es excusa para implementarlo en otros tipos de trabajo como son las investigaciones, manejo de empresas, entre otros (Scrum.org, 2018).

Se busca con Scrum realizar una buena planeación y estrategia de ejecución de las actividades a desarrollar en las etapas de un proyecto; con el fin de enfocarse en fracciones pequeñas que generen valor y que al final conforman un todo, esto ayuda a identificar posibles errores que se estén cometiendo y que se puedan solucionar antes de que se vea afectado todo el proyecto. En Scrum se usa como unidad de medida los sprints, en estos se especifica cada necesidad en historias de usuario que se irán trabajando en lo que dure el sprint.

Por metodología un Sprint no puede durar más de 4 semanas y lo que no se pueda atender en un Sprint deberá ser atendido en el Sprint siguiente. Scrum también hace uso de algunas reuniones con el fin de buscar que el equipo de trabajo sea autogestionable, las reuniones son el sprint plannig, sprint daily, refinamiento y retrospectiva. No siempre se hace uso de ellas ya que la metodología scrum permite tomar lo que sea necesario y ser aplicado a cada proyecto. Por ello en este caso solo se hizo uso de la planeación y muestra de resultados obtenidos.

La etapa de planeación arrojó los *sprints* que se observan desde la Tabla 1 a la Tabla 8.

Duración: 4 semanas		Historias de usuario	Análisis y revisión
Sprint 1	Caracterización de las iniciativas, propuestas y protocolos necesarios para el diseño de un modelo arquitectónico	Como dueño del proyecto necesito realizar un estado del arte en bases de datos indexadas sobre redes vanet para tener conocimiento de que se ha realizado con redes vanet a la fecha	El acceso a bases de datos indexadas, para esto se realizó la consulta bibliográfica con el usuario institucional del estudiante y/o desde el proxy de la Universidad Distrital
		Como dueño del proyecto debo seleccionar y organizar los modelos y/o propuestas para así obtener datos relevantes que contribuyan a la investigación	Idioma en que se encontraron los artículos fue en inglés para esto se tradujeron
		Como dueño del proyecto debo identificar las características principales de los modelos estudiados para analizar sus ventajas y desventajas a la hora de construir el nuevo modelo	Selección de las características principales

Tabla 1: Sprint 1

Duración: 4 semanas		Historias de usuario	Análisis y revisión
Sprint 2	Diseño de un modelo conceptual	Como dueño del proyecto debo definir los objetivos del modelo conceptual para así delimitar el alcance del modelo arquitectónico	La no existencia de una metodología para la creación de modelos arquitectónicos en redes, se debió hacer uso de modelos tradicionales como el Modelo OSI de la ISO
		Como dueño del proyecto necesito clasificar los componentes para el diseño del modelo arquitectónico para poder obtener la interconectividad entre redes Vanet e IoT	Estudiar los componentes de redes Vanet para que la elección de estos no afectara el comportamiento de las futuras redes

Tabla 2: Sprint 2

Duración: 3 semanas		Historias de usuario	Análisis y revisión
Sprint 3	Diseño de un modelo conceptual	Como dueño del proyecto necesito identificar las relaciones entre los componentes o agentes del modelo arquitectónico para obtener los mejores elementos de redes Vanet e IoT	Compatibilidad entre los elementos de redes vanet e IoT, para esto se debió identificar la capa en la cual trabajan cada uno
		Como dueño del proyecto debo hacer la representación del modelo conceptual para que sea una guía en la construcción del modelo arquitectónico	Se hizo el diseño del modelo a partir de los estudios previos

Tabla 3: Sprint 3

Duración: 4 semanas		Historias de usuario	Análisis y revisión
Sprint 4	Implementación de la arquitectura del modelo final de interoperabilidad	Como dueño del proyecto debo clasificar los resultados del primer Sprint	Capítulo 1 y 2 del proyecto
		Como dueño del proyecto tengo que determinar los requisitos y funciones del modelo	Diseño de la red. Ver Figura 5.

Tabla 4: Sprint 4

Duración: 4 semanas		Historias de usuario	Análisis y revisión
Sprint 5	Implementación de la arquitectura del modelo final de interoperabilidad	Como dueño del proyecto necesito elegir una metodología de desarrollo	Metodología de desarrollo Scrum

Tabla 5: Sprint 5

Duración: 3 semanas		Historias de usuario	Análisis y revisión
Sprint 6	Simulación y contraste de la implementación realizada	Como dueño del proyecto necesito validar el modelo diseñado para esto debo utilizar una herramienta de simulación de redes	Estudio de herramientas de simulación mediante un cuadro comparativo y elección de la que se acercara a las condiciones del proyecto

Tabla 6: Sprint 6

Duración: 4 semanas		Historias de usuario	Análisis y revisión
Sprint 7	Simulación y contraste de la implementación realizada	Como dueño del proyecto debo identificar las ventajas y limitaciones del modelo arquitectónico de interoperabilidad	Realización de simulaciones y análisis de los resultados de estas

Tabla 7: Sprint 7

Duración: 2 semanas		Historias de usuario	Análisis y revisión
Sprint 8	Simulación y contraste de la implementación realizada	Como dueño del proyecto debo confrontar el desempeño del modelo propuesto con otros ya existentes	Discusión de resultados

Tabla 8: Sprint 8

3.1 MODELO DE REFERENCIA IOT GENÉRICO

Los modelos de referencia dan la base para la comprensión del dominio de IoT, modelando los conceptos y sus relaciones. El modelo ayuda a tener en cuenta las propiedades conocidos o inferidas de la arquitectura IoT. Se debe aclarar que con el término referencia

se busca indicar que la arquitectura propuesta se use como fuente para desarrollar una arquitectura IoT específica y concreta.

3.1.1 MODELO ABSTRACTO IOT

A finales del milenio anterior, el termino IoT ha sido usado en muchos lugares y en muchas formas. Son distintas las definiciones de IoT que se han introducido de forma independiente tanto por individuos como por compañías (Postscapes, 2018).

IoT es un tema que se encuentra en auge, existen diferentes áreas de investigación y organizaciones interesadas en la temática. Ya que son muchos los dispositivos que contienen microchips. Por esta razón, existen distintas definiciones de IoT.

Una definición clara y objetiva de IoT es resumida en la siguiente expresión: (Mcewen & Cassimally, 2014) (Guevara Riveros, 2018)

Objeto físico + controladores, sensores y actuadores + Internet = IoT

Con la anterior expresión se construye el siguiente gráfico.

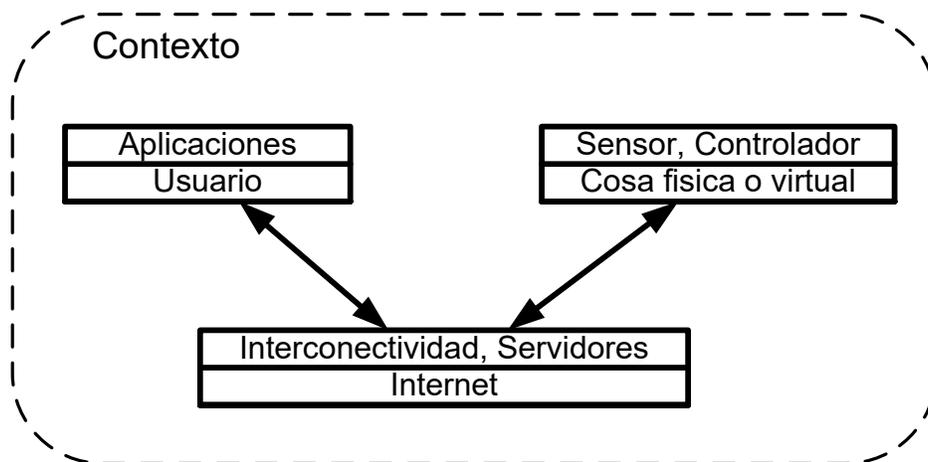


Figura 3: Modelo abstracto IoT
Fuente: (Guevara Riveros, 2018)

Contexto

El contexto es el ambiente en el que el usuario inserta la tecnología IoT y la Cosa. El ambiente puede ser físico o virtual, en el que en el caso de ambientes físicos se pueden citar ambientes públicos, privados, de uso común o particulares de personas u objetos como carros, robots, animales, etc.; para ambientes virtuales, se pueden citar sitios, juegos de realidad aumentada, redes sociales, entre otros ambientes en que personas u objetos pueden estar insertados. Por otro lado, el contexto se encuentra directamente relacionado con la Cosa, porque forma parte del contexto, es decir, es el objeto o persona “objetivo” de la recolección de datos del Dispositivo perteneciente a la Tecnología IoT.

Usuario / Aplicaciones

La capa está dirigida a los usuarios y aplicaciones como los son las herramientas de apoyo a la toma de decisiones. Los actores finales dentro del contexto son los usuarios porque estos visualizan o utilizan la representación de los datos producidos por la Tecnología IoT. Los usuarios pueden ser tanto humanos, objetos o sistemas informáticos, que utilizan los datos proporcionados por la tecnología IoT, como es el caso de hogares inteligentes que usan la información de dispositivos para realizar acciones como el control de iluminación, temperatura o humedad. El usuario puede interactuar con Cosas con diversos medios como computadoras portátiles, teléfonos inteligentes, por medio de las aplicaciones correspondientes.

Cosa física o virtual

Con la cosa física o virtual se busca la recolección de datos y da la posibilidad de ser un humano, un objeto, ambiente o sistema de información. Hay diferentes tipos de cosas disponibles. Los objetos al ser pequeños van a tener limitaciones en cuanto al poder de cómputo y recursos. La abstracción del hardware debe permitir reconocer todos estos dispositivos destinados a leer las diversas Cosas.

3.2 DISEÑO FÍSICO DE UNA RED VANET

Las redes vanets se fundamentan en dos tipos de nodos: vehículos e infraestructuras (Sánchez Sánchez, 2017). Cada uno equipado con diferentes sistemas de comunicación. El vehículo, debe incorporar una unidad de abordo (On-Board Unit - OBU), este dispositivo dispone de un sistema inalámbrico de comunicaciones de corto alcance que permite la formación de redes ad-hoc.

Las infraestructuras, por norma general, son nodos que se encuentran al lado de la carretera (Road-Side Unit, RSU) que se usa para conectar los vehículos a la infraestructura de red e inclusive expandir a lo largo de un área de servicios o aplicaciones dentro de la propia red las conexiones inalámbricas. En la Figura 4 se observa el diseño de una red Vanet tradicional.

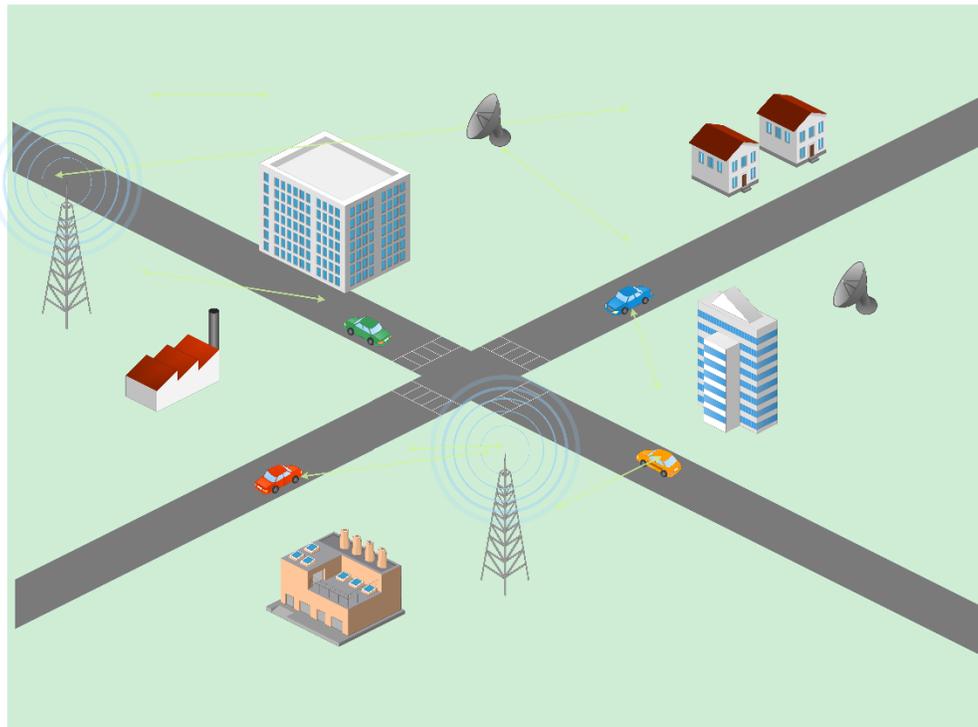


Figura 4: Red Vanet tradicional

Fuente: Autor. Generada por medio del programa *conceptDraw DIAGRAM*

3.2.1 PROTOCOLOS PARA REDES VANET

AODV

El protocolo de enrutamiento Ad hoc *On-Demand Distance Vector* (AODV) se diseñó para ser utilizado por nodos móviles en una red ad hoc. Este protocolo ofrece una rápida adaptación a condiciones de enlace dinámico, bajo procesamiento y sobrecarga de memoria, baja utilización de la red y determina rutas de unidifusión a destinos dentro de la red ad hoc. Utiliza números de secuencia de destino para garantizar la libertad del bucle en todo momento (incluso frente a la entrega anómala de mensajes de control de enrutamiento), evitando problemas (como "contar hasta el infinito") asociados con los protocolos clásicos de vectores de distancia. (IETF, 2018)

IEEE 802.11.p

El estándar IEEE 802.11.p define las características de la capa física y Mac necesarios para operar dentro un entorno vehicular. (López & Guijarro, 2018)

- Capa física (PHY): Emplea la transmisión multiportadora OFDM (orthogonal Frequency Division Multiplexing). En la siguiente tabla se muestran las características del canal físico.

<i>Característica del enlace de datos inalámbrico</i>	<i>Valor 802.11p</i>
Tasa de bit	3-27 Mbps
Rango de comunicación	< 1000m
Potencia de transmisión máxima	760mW (US) / 2W (EU)
Ancho de banda del canal	10 MHz
Espectro asignado	75 MHz (US) / 30 MHz (EU)
Banda de frecuencia	5,86 GHz - 5,92 GHz

Tabla 9: Características de enlace de datos inalámbricos para el protocolo 802.11.p

- Capa Mac: El propósito de la capa Mac es establecer los mecanismos de acceso al canal de comunicación, de manera que un conjunto de estaciones pueda compartir eficientemente el medio inalámbrico. El estándar IEEE 802.11p define el uso de CSMA/CA (Carrier Sense Multiple Access/Collision Avoidance). También aspectos como la transmisión, la probabilidad de recepción de paquetes, el tiempo de acceso al canal, el control de congestión y la priorización de los mensajes entre otros.

3.3 DISEÑO FÍSICO CON UNA RED VANET INTERCONECTA CON IOT

En la red Vanet interconectada con IoT se puede afirmar que es una red vehicular heterogénea, donde además de los elementos mencionados en la sección inmediatamente anterior se encuentran: dispositivos personales tales como: celulares inteligentes, gps (sistema de posicionamiento global) y también se encuentran los sensores, sensores propios de los vehículos y de dispositivos inteligentes. En la Figura 5 se observa una red Vanet interconectada con IoT.

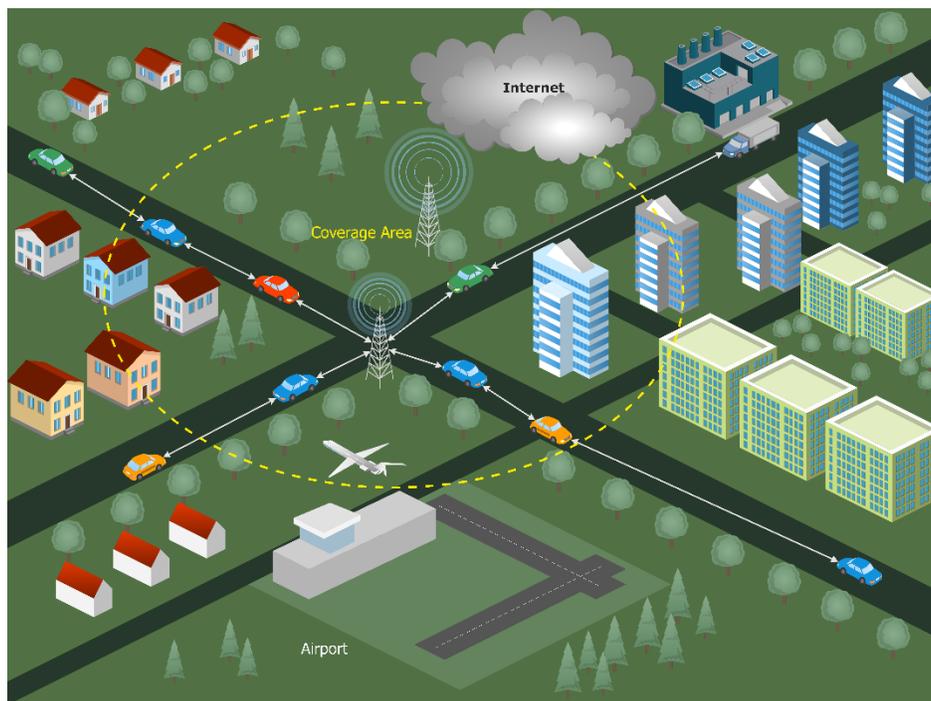


Figura 5: Red Vanet interconectada con IoT

Fuente: Autor. Generada por medio del programa *conceptDraw DIAGRAM*

3.3.1 PROTOCOLOS PARA IOT

IEEE 802.11.ah

IEEE 802.11ah es un estándar de Wi-Fi que ha sido diseñado para utilizar las bandas ISM libres de licencia por debajo de un GHz. La propagación de radio en estas frecuencias significa que las señales pueden viajar distancias más grandes y esto abre oportunidades para el uso de Internet de las Cosas, donde los sensores y los nodos de control pueden ubicarse más separados. (Electronics Notes, 2017)

Las bandas en estas frecuencias son mucho más pequeñas que las bandas de 2,4 GHz y 5 GHz que normalmente se usan para Wi-Fi y esto limita las velocidades de datos que pueden enviarse a través de los enlaces. Para acomodar los diferentes aspectos de Wi-Fi sub GHz, se ha desarrollado una nueva capa física y MAC para permitir las comunicaciones en estas frecuencias, aunque a una velocidad menor que la que se puede lograr para las variantes de Wi-Fi de alta velocidad de la corriente principal. (IEEE, 2016)

- 802.11ah channelization: El ancho del canal básico es de 1MHz, aunque es posible unir dos canales adyacentes para formar un canal de 2 MHz para proporcionar una mayor capacidad de rendimiento de datos. Hay canales más amplios disponibles, el más amplio en los EE. UU. Es de 16 MHz para la banda ISM de 902 - 928 MHz. Nuevamente, esto utiliza el mismo método de enlace de canales adoptado para 802.11n y 11ac. Se pueden utilizar anchos de canal de 1, 2, 4, 8 y 16 MHz.
- 802.11ah PHY / radio interface: 802.11ah utiliza la multiplexación por división de frecuencia ortogonal, OFDM para proporcionar el esquema de modulación para la señal. Sin embargo, hay dos categorías en las que se puede dividir la capa física 802.11ah PHY:
 - Ancho de banda de canal de 1 MHz: este modo de operación está dirigido principalmente a aquellas aplicaciones que requieren un rango extendido. El ancho de banda más estrecho y las velocidades de datos más lentas permiten acomodar señales con potencias de señal más bajas. Por lo general, estas aplicaciones pueden estar destinadas a aplicaciones IoT o M2M en las que se pueden requerir ráfagas cortas de datos, normalmente a una velocidad de datos baja.

- Anchos de banda de 2 MHz y superiores: este modo utiliza anchos de banda de 2, 4, 8 o 16 MHz. Nuevamente usa OFDM y un diseño basado en una décima velocidad de reloj de 802.11ac, es decir, una longitud de símbolo diez veces mayor que en 802.11ac. MIMO también se usa dentro de 802.11ah también en este modo.
- 802.11ah MAC: El Control de acceso a medios o la capa MAC presenta una serie de elementos mejorados para proporcionar soporte para un gran número de estaciones, ahorro de energía y similares:
 - Soporte para una gran cantidad de estaciones: los puntos de acceso 802.11 asignan identificadores, llamados Identificadores de Asociación ID, AID, a las estaciones que se asocian con el AP. Para los sistemas que no son 802.11ah, el número máximo de identificadores que se pueden asignar es 2007, pero con el uso de 802.11ah para posiblemente aplicaciones IoT o M2M, este número podría excederse.
 - Ahorro de energía: el ahorro de energía es un problema creciente, especialmente para IEEE 802.11ah que se usará para muchas aplicaciones IoT y M2M. Muchos de los nodos remotos necesitarán funcionar con baterías y estos deben poder funcionar durante semanas o incluso años sin reemplazo.
 - Mejoras de rendimiento: para poder hacer el mejor uso del ancho de banda disponible, se han realizado una serie de mejoras para garantizar que los datos se transporten de la manera más eficiente posible.

3.4 ARQUITECTURA PROPUESTA

La arquitectura propuesta presenta un enfoque de 3 capas con el fin de simplificar la funcionalidad de los componentes. Las capas que presenta esta arquitectura son:

- Capa cliente
- Capa de conexión
- Capa de la nube

3.4.1 CAPA CLIENTE

Es la capa inferior del modelo, esta capa es la encargada de cubrir las comunicaciones intra vehiculares e inter vehiculares. Esta compuesto por sensores IoT responsables de la posición, velocidad, proximidad, presión de los neumáticos, ruido, niveles de contaminación, obstáculos delanteros y laterales, detección de colisiones. Los vehículos también tendrán consigo Unidad a bordo (OBU) que permitirá procesar los datos recopilados de los sensores disponibles en el vehículo para estimar la condición actual del vehículo y de la carretera.

3.4.2 CAPA DE CONEXIÓN

En esta capa de conexión se proporciona una comunicación V2V pero también se busca dar soporte para una comunicación V2I (vehículo a infraestructura), deseable para dar un salto de una red Vanet tradicional a una red Vanet de próxima generación, en este caso interconectada con elementos IoT. En esta capa se encuentran también las unidades de carretera (RSU) desplegadas a lo largo de la carretera. Estas unidades realizan la conexión entre la capa cliente y la capa de la nube.

3.4.3 CAPA DE LA NUBE

La capa de nube, es la capa responsable de habilitar los servicios y aplicaciones para la red Vanet interconectada con IoT. Esta capa ofrece servicios basadas en la nube, como lo son almacenamiento masivo, virtualización y también interacciones en tiempo real entre diferentes entidades de red.

En la Figura 6 se puede observar gráficamente la arquitectura planteada.

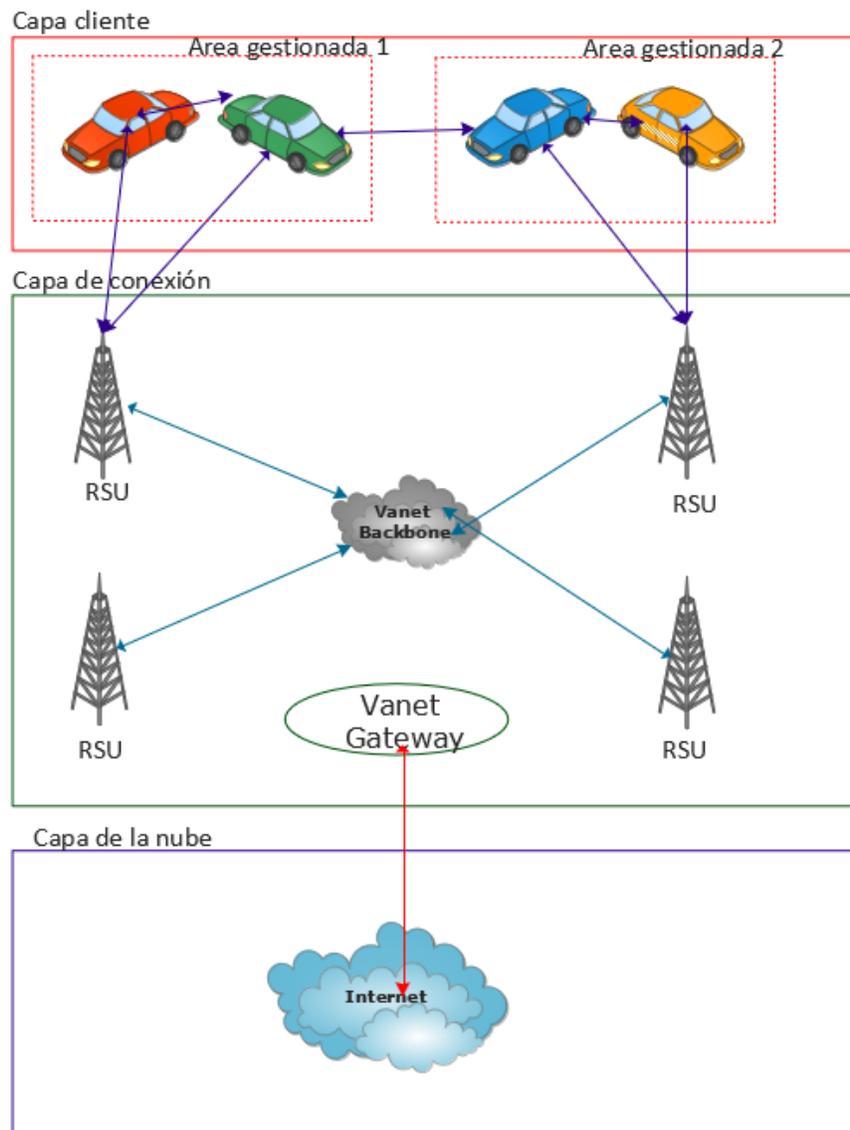


Figura 6: Arquitectura propuesta de red Vanet interconecta con IoT
Fuente: Autor. Generada por medio del programa *conceptDraw DIAGRAM*

3.6 HERRAMIENTAS DE SIMULACIÓN

Para el cumplimiento de los objetivos del proceso de investigación se indagaron varias opciones de software para la simulación, a continuación, se procede a mostrar una selección de las herramientas encontradas:

3.6.1 SIMULADORES DE TRÁFICO

SUMO

Simulation of Urban Movility(*SUMO – Simulation of Urban MObility*, 2017), simulador de tráfico gratuito desarrollado por DLR (Dutsche Gesellschaft für Luft und Raumfahrt) en lenguaje C++. Este simulador de tráfico permite simular el movimiento de los vehículos teniendo en cuenta variables como peatones y transporte público. También tiene en cuenta los carriles de las vías, cambio de carril de los vehículos, y el comportamiento de estos en los mismos cruces.



Figura 7: Sumo

Fuente: (*SUMO – Simulation of Urban MObility*, 2017)

VanetMobiSim

VanetMobiSim es una extensión del entorno de simulación de movilidad CANU (CanuMobiSim), un marco flexible para el modelado de movilidad de usuarios. (VanetMobiSim, 2018). implementado en Java como una aplicación independiente, permitiendo su utilización herramienta de simulación dedicada a redes, por ejemplo, NS-2

o Qualnet. En este simulador se puede destacar el uso de modelos de movilidad macroscópicos y microscópicos.

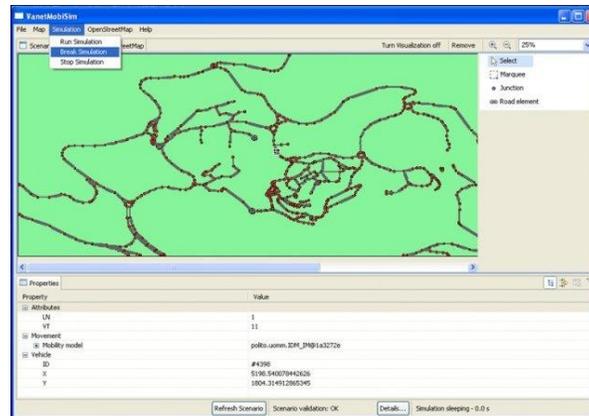


Figura 8 Estructura del simulador VanetMobiSim
Fuente: (VanetMobiSim, 2018)

STRAW

STRAW (Street RAndom Waypoint)(Choffnes & Bustamante, 2006), fue desarrollado por AquaLab. Este simulador es capaz de asumir el comportamiento real de los vehículos con gran detalle gracias a que hace uso de un modelo de movilidad de estudio de tráfico en ciudades de Estados Unidos.

BonnMotion

Es un software gratuito e implementado por la Universidad de Bonn y la Universidad de Osnabrück en Alemania(BonnMotion, 2016). Este simulador permite analizar escenarios de movilidad, siendo uno de los más extendidos para el estudio de redes MANET. La principal ventaja es la interoperabilidad de los escenarios de tráfico creados con simuladores como ns-2, ns3 o MiXim.

3.6.2 SIMULADORES DE RED

NETWORK SIMULATOR 3 (NS3)

Es la tercera versión de un simulador de redes de eventos discretos, es un software de distribución gratuita bajo la licencia GNU GPLv2 el núcleo de su operación está en lenguaje C++ y las instrucciones se generan en Python (Hafizah et al., 2016).

Es ampliamente usado para simular redes de internet con la ventaja de soportar simulaciones de Wi-Fi, Wimax y LTE (Nsnam, 2018)

Permite poder realizar emulaciones de los diferentes equipos que se usan en una red.

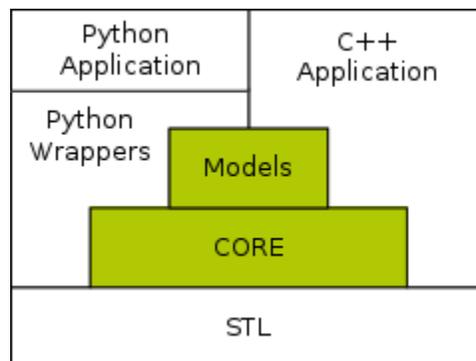


Figura 9 Estructura en diagrama de bloques simulador ns3
Fuente: (Nsnam, 2018)

QUALNET

Este es un software para simulación de protocolos fuertemente usado en simulaciones inalámbricas, aunque también presenta las librerías correspondientes a redes cableadas las instrucciones se ingresan mediante PARSEC el cual determina la cantidad de eventos discretos. (Hafizah et al., 2016)

Permite una interfaz gráfica tridimensional en donde se puede planear, probar y monitorear actividades de diferentes redes según sea la necesidad (scalable-networks, 2017).

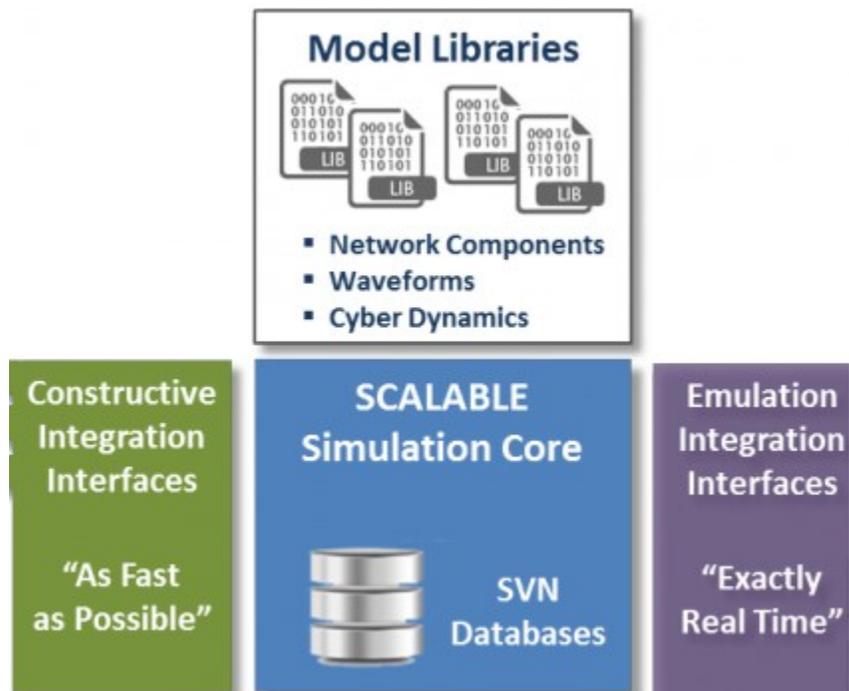


Figura 10 Estructura en diagrama de bloques simulador QUALNET
Fuente: (scalable-networks, 2017)

JiST

De sus siglas en inglés: "Java en Tiempo de simulación " es una herramienta de software que permite la implementación de simulaciones de redes en el lenguaje de programación Java, se utiliza mayormente a la vez con SWANS1.

En este software se pueden encontrar diferentes tipos de objetos que representan los elementos de red tales como los nodos, la simulación se forma por invocaciones de métodos entre estos objetos (Weingärtner et al., 2009).

Para lograr ejecutar las simulaciones, JiST carga las clases de manera personalizada que reescribe dinámicamente el código (jist, 2017).

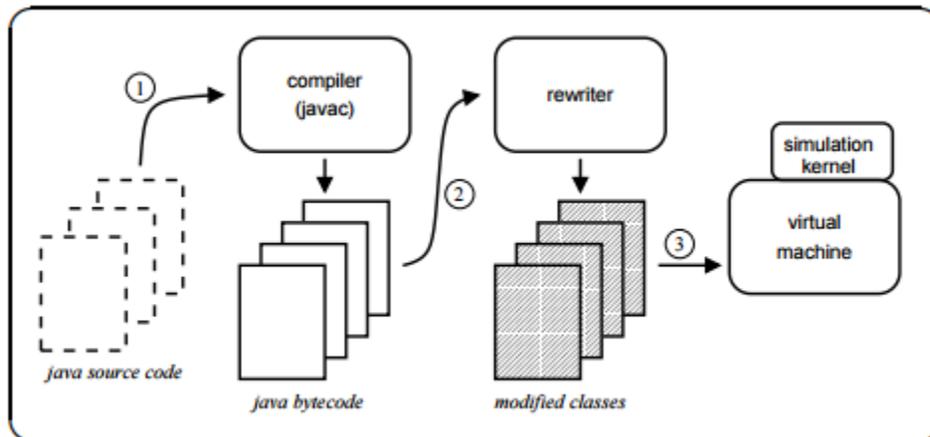


Figura 11 Estructura en diagrama de bloques simulador JiST
Fuente: (jist, 2017)

OMNET++

Es un simulador compuesto por módulos y basado en objetos, está escrito en C++ y permite realizar simulaciones en tiempo real de redes de telecomunicaciones y la integración con bases de datos, además de que soporta otros lenguajes de programación como C# o Java (Surati et al., 2017).

También presenta utilidades en simulación de sistemas de TI, teoría aplicada de colas, diversas arquitecturas de hardware y procesos de negocio.

También presenta una serie de modelos que presentan protocolos P2P. También soporta Java pero en este caso es necesario el uso de la extensión JSimpleModule (omnetpp, 2018).

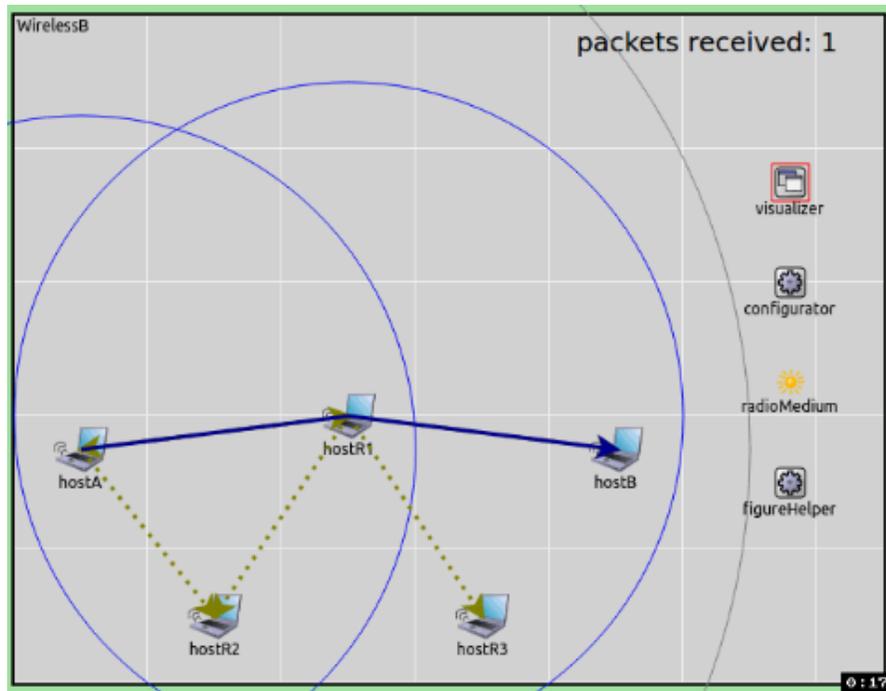


Figura 12 Ejemplo de simulación en OMNET++
Fuente:(omnetpp, 2018)

NETWORK SIMULATOR 2 (NS2)

Ns2 (Network Simulator 2) es un simulador de eventos discretos sobre redes LAN, WAN, MAN, Wi-Fi el cual permite el uso e implementación de diferentes protocolos (Nsnam, 2017). Este simulador permite realizar simulaciones teniendo en cuenta parámetros como: la forma de propagación, los efectos de los rebotes de la señal y los protocolos del estándar IEEE 802.11. Lo cual permite obtener un alto grado de detalle en la simulación de las Vanets.

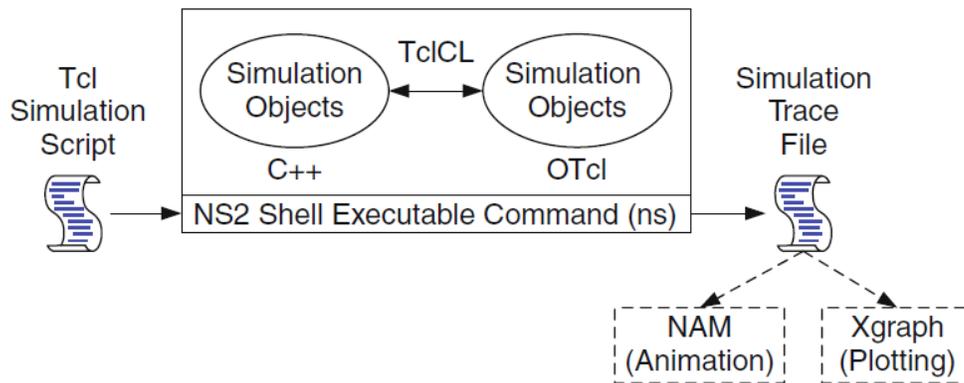


Figura 13 Estructura en diagrama de bloques simulador ns2
Fuente: (Nsnam, 2017)

3.6.3 COMPARACIÓN DE SIMULADORES

Luego de ser estudiadas las herramientas de simulación más importantes para la implementación de Vanet, se hace una comparación de estos en la Tabla 10.

En cuanto a los simuladores de tráfico, el simulador BonnMotion, es la herramienta que menos beneficios ofrece para simulaciones de escenarios realistas ya que inicialmente este fue un simulador para redes Manet. Por lo que su diseño no se acentúa en las necesidades de simulaciones para Vanet. Con respecto al resto de simuladores, SUMO es la mejor solución, ya que es una herramienta que permite obtener un mayor grado de detalle a la hora de simular redes Vanet.

Respecto a los simuladores de red, la principal ventaja de OMNet++ es su estructura modular aportando así una gran extensibilidad a los proyectos que se implementan sobre él, entendiendo de forma adecuada a redes Vanet. Los simuladores NS-3 y SWANS++ ofrecen escenarios aceptables en el momento de simular, pero cada uno presenta desventajas tales como, NS-3 no es compatible con extensiones desarrolladas en NS-2 y SWANS++ no permite la interacción entre el modelo de la red y el simulador de red, lo que reduce la capacidad de crear una red vehicular cercana a la realidad. Por otro lado, el simulador NS-2 tiene gran soporte para proyectos académicos e investigativos, y la pila de protocolos de la familia IEEE 802.11.

Características	Simuladores de trafico			
	SUMO	BonnMotion	STRAW	VanetMobiSim
Licencia	Free	Free	Free	Free
Lenguaje	C++	Java	-	Java
Plataforma	Windows, GNU/Linux	Windows, GNU/Linux	-	Windows, GNU/Linux
Compatibilidad simulador de red	Ns-2, GloMoSim, Qualnet	Ns-2, Ns-3, GloMoSim, Qualnet	Ns-2, GloMoSim, Qualnet	Ns-2, GloMoSim, Qualnet
Mapas reales	Si	Si	Si	Si
Vías con varios carriles	Si	No	Si	Si
Sentido del trafico	Si	No	Si	Si
Peatones	Si	No	No	No
Tipos de vehículos	Si	No	No	No
Intersecciones	Si	No	-	Si
Señalización	Si	No	Si	Si

Tabla 10: Comparación simuladores de red

Teniendo en cuenta el enfoque de este trabajo de investigación sobre redes Vanet que usan protocolo AODV y protocolos de redes IoT se hace uso del simulador NS2.

Una vez se ha desarrollado el modelo se procede a realizar las simulaciones para comprobar que cumpla con lo esperado.

4. SIMULACIÓN DE UNA RED VANET INTERCONECTADA CON IOT

En el presente capítulo se exponen las simulaciones realizadas de la red Vanet tradicional y de la red Vanet interconectada con IoT. Para la red Vanet tradicional se hace uso del protocolo *AODV* como protocolo de enrutamiento y del estándar *IEEE 802.11.p* para la capa física y la capa Mac. Para la red Vanet interconectada con IoT nuevamente se hace uso del protocolo *AODV* como protocolo de enrutamiento y para la capa física y la capa Mac se hace uso del estándar *IEEE 802.11.ah*.

Por limitaciones de la plataforma *openstreetmap.org* donde el límite de nodos que se pueden exportar de los mapas es de 50000 nodos. Se realizan las simulaciones en 5 zona de la ciudad de Bogotá que se relación entre sí por ser fronteras. Estas zonas son:

- Zona 1: Barrios Unidos
- Zona 2: Chapinero
- Zona 3: Engativá
- Zona 4: Usaquén
- Zona 5: Teusaquillo

Según cifras de la Secretaría Distrital de Planeación la densidad población de estas 5 zonas es de 11503 hab/km² (Gaitán Victoria et al., 2018)

A cada zona se le realizó la simulación de una red Vanet tradicional y una red Vanet con IoT.

Las conexiones de tráfico aleatorias CBR se crean por medio del programa *cbrgen.tcl*, el cual es un generador de escenarios de tráfico. Como parámetros se establece el modo de conexión (-type) como *cbr*, el número de nodos (-nn), el número de semillas (-seed), el número de conexiones (-nc) y la tasa (-rate). Estos resultados son concatenados a un fichero que luego son llamados en la simulación.

Los parámetros de las simulaciones para cada zona de una red Vanet se muestran en la Tabla 11.

	Zona 1	Zona 2	Zona 3	Zona 4	Zona 5
Tipo de canal	Wireless Channel				
Modelo de radiopropagación	Two-ray ground				
Tipo de interfaz de red	Wireless Phy				
Tipo de MAC	802.11	802.11	802.11	802.11	802.11
Tipo de cola de interfaz	Pri Queue				
Tipo de capa de enlace	LL	LL	LL	LL	LL
Modelo de antena	OmniAntenna	OmniAntenna	OmniAntenna	OmniAntenna	OmniAntenna
Paquete máximo en ifq	50	50	50	50	50
Cantidad de nodos móviles	99	94	82	92	91
Protocolo de enrutamiento	AODV	AODV	AODV	AODV	AODV
Coordenada en x de la topología	6803	3554	6859	6572	5818
Coordenada en y de la topología	12783	14378	7723	18220	6884
Tiempo de simulación (segundos)	300	300	300	300	300

Tabla 11: Parámetros de las simulaciones de una red Vanet

Los parámetros de las simulaciones para cada zona de una red Vanet interconectada con IoT se muestran en la Tabla 12.

		Zona 1	Zona 2	Zona 3	Zona 4	Zona 5
Interfaz de red	Sensibilidad de la interfaz inalámbrica (dBm)	3,9810717055349694e-13	3,9810717055349694e-13	3,9810717055349694e-13	3,9810717055349694e-13	3,9810717055349694e-13
	Pt (dBm)	0,1	0,1	0,1	0,1	0,1
	Frecuencia (dBm)	5.9e+9	5.9e+9	5.9e+9	5.9e+9	5.9e+9
	Ruido (dBm)	1.26e-13	1.26e-13	1.26e-13	1.26e-13	1.26e-13

Pérdida predeterminada del circuito de radio (dBm)	1.0	1.0	1.0	1.0	1.0	
Sensibilidad del monitor de potencia	3.981071705534985e-18	3.981071705534985e-18	3.981071705534985e-18	3.981071705534985e-18	3.981071705534985e-18	
Duración del encabezado (dBm)	0.000040	0.000040	0.000040	0.000040	0.000040	
Basic Modulation Scheme (dBm)	0	0	0	0	0	
Preamble Capture Switch (dBm)	1	1	1	1	1	
Data Capture Switch (dBm)	1	1	1	1	1	
SINR Preamble Capture (dBm)	3,1623	3,1623	3,1623	3,1623	3,1623	
SINR Data Capture (dBm)	10,0	10,0	10,0	10,0	10,0	
Trace dist (dBm)	1e6	1e6	1e6	1e6	1e6	
PHY DBG (dBm)	0	0	0	0	0	
Tipo de MAC	Ancho del canal (minimo) (dBm)	15	15	15	15	
	Ancho del canal (maximo) (dBm)	1023	1023	1023	1023	
	Slot Time	0.000052	0.000052	0.000052	0.000052	0.000052
	SIFS	0.0000160	0.0000160	0.0000160	0.0000160	0.0000160
	Short Retry Limit	7	7	7	7	7
	Long Retry Limit	4	4	4	4	4
	Header Duration	0.000040	0.000040	0.000040	0.000040	0.000040
	Symbol Duration	0.000008	0.000008	0.000008	0.000008	0.000008

	Basic Modulation Scheme	0	0	0	0	0
	RTS Threshold	2346	2346	2346	2346	2346
	MAC DBG	0	0	0	0	0
Parametros generales	Tipo de canal	Wireless Channel				
	Modelo de radiopropagación	Two-ray ground				
	Tipo de cola de interfaz	Pri Queue				
	Tipo de capa de enlace	LL	LL	LL	LL	LL
	Modelo de antena	OmniAntenna	OmniAntenna	OmniAntenna	OmniAntenna	OmniAntenna
	Paquete máximo en ifq	50	50	50	50	50
	Cantidad de nodos móviles	99	94	82	92	91
	Protocolo de enrutamiento	AODV	AODV	AODV	AODV	AODV
	Coordenada en x de la topología	6803	3554	6859	6572	5818
	Coordenada en y de la topología	12783	14378	7723	18220	6884
	Tiempo de simulación (segundos)	300	300	300	300	300

Tabla 12. Parámetros de las simulaciones de una red Vanet interconectada con IoT

En el ANEXO 3: RESULTADOS DE LA SEGUNDA SERIE DE SIMULACIONES se muestran los resultados para una nueva serie de simulaciones donde se ha variado la cantidad de nodos móviles a 150 y el tiempo de simulación a 150 segundos, para cada una de las cinco zonas.

Simulación zona 1 de red Vanet tradicional

Obtener el mapa en la plataforma *openstreetmap.org*. Se crea un fichero con la extensión *.osm*. Este es un archivo XML creado en el formato OpenStreetMap (OSM), que se utiliza para guardar información de mapas de calles; guarda los datos en forma de "nodos" (puntos), "formas" (conexiones) y "relaciones" (propiedades de calles y objetos, como etiquetas).

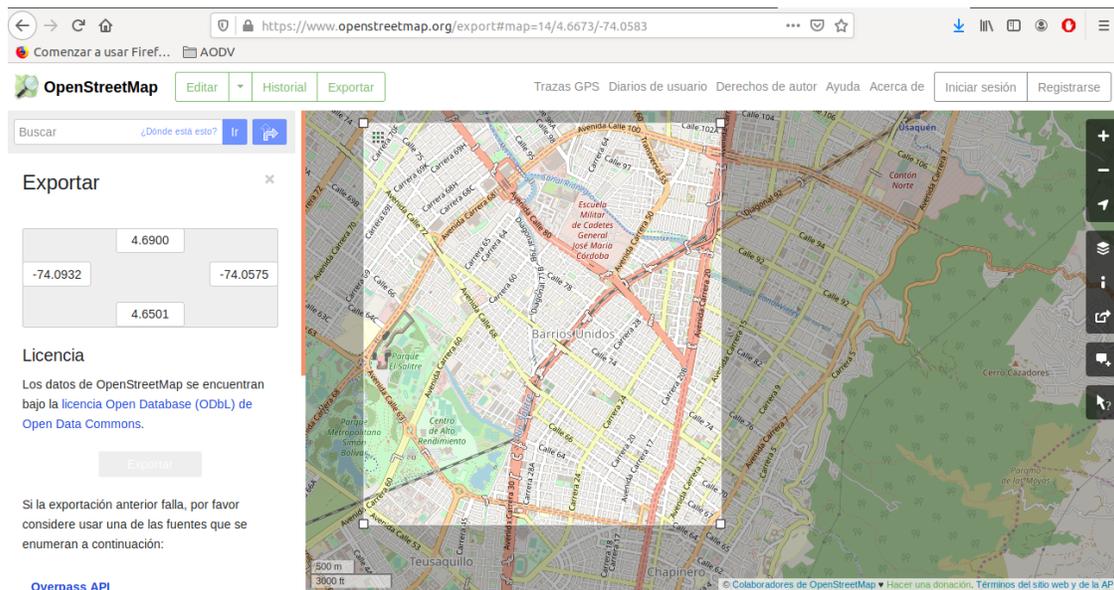


Figura 14: Obtención del mapa para la zona 1

Fuente: Autor. Generada por medio de la plataforma *openstreetmap.org*

El límite de nodos que se puede obtener de la plataforma son de 50000 nodos.

Una vez obtenidos los mapas se procede a convertir las carreteras digitales a redes de carreteras que puedan ser usadas por el simuladores Ns-2. Para estos se usa el programa *netconvert* del simulador Sumo. Esta conversión se crea un fichero con extensión *.net.xml*.

```
lewys@lewys-300E4C-300E5C-300E7C: ~/Escritorio/zona 1
Archivo Editar Ver Buscar Terminal Ayuda
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 1$ netconvert --osm-files zon
a1.osm -o zona1.net.xml
Warning: Environment variable SUMO_HOME is not set, using built in type maps.
Warning: Environment variable SUMO_HOME is not set, schema resolution will use s
low website lookups.
Warning: Discarding unknown compound 'cycleway.shared_lane' in type 'cycleway.sh
ared_lane|highway.residential' (first occurrence for edge '24546872#0').
Warning: Discarding unknown compound 'cycleway.track' in type 'cycleway.track|hi
ghway.residential' (first occurrence for edge '24690430').
Warning: Discarding unusable type 'waterway.canal' (first occurrence for edge '28
159886').
Warning: Discarding unusable type 'waterway.stream' (first occurrence for edge '5
8074589').
Warning: Discarding unusable type 'waterway.river' (first occurrence for edge '58
390754#0').
Warning: Discarding unknown compound 'cycleway.track' in type 'cycleway.track|hi
ghway.primary' (first occurrence for edge '89626693').
Warning: Discarding unknown compound 'cycleway.track' in type 'cycleway.track|hi
ghway.tertiary' (first occurrence for edge '89626722').
Warning: Discarding unknown compound 'cycleway.track' in type 'cycleway.track|hi
ghway.secondary' (first occurrence for edge '89875093').
Warning: Discarding unusable type 'waterway.artificial' (first occurrence for edg
e '228762649').
Warning: Discarding unknown compound 'cycleway.track' in type 'cycleway.track|hi
```

Figura 15: Conversión del mapa de la zona 1 a redes de carreteras
Fuente: Autor. Generado por medio de la consola de Ubuntu 18.04.2.

Los datos del mapa .osm no solo contienen la red de carreteras sino también una amplia gama de polígonos adicionales, como edificios y ríos. Estos polígonos pueden importarse usando el programa *polyconvert* y luego agregarse a una configuración de sumo-gui. Esta conversión crea el fichero *zona1.poly.xml*.

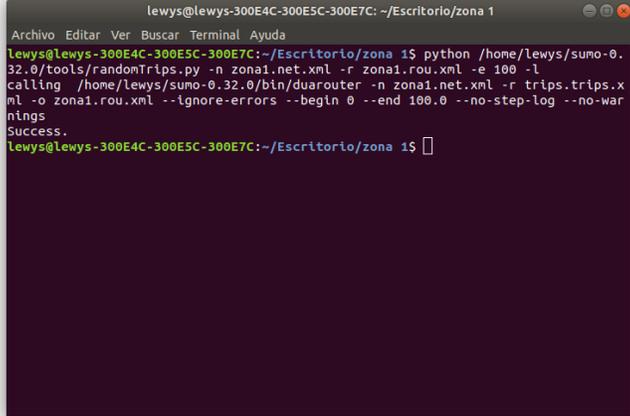
```
lewys@lewys-300E4C-300E5C-300E7C: ~/Escritorio/zona 1
Archivo Editar Ver Buscar Terminal Ayuda
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 1$ polyconvert --osm-files zo
na1.net.xml --type-file osmPolyconvert.typ.xml -o zona1.poly.xml
Warning: Environment variable SUMO_HOME is not set, schema resolution will use s
low website lookups.
Warning: Environment variable SUMO_HOME is not set, schema resolution will use s
low website lookups.
Warning: Environment variable SUMO_HOME is not set, schema resolution will use s
low website lookups.
Warning: Environment variable SUMO_HOME is not set, schema resolution will use s
low website lookups.
Success.
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 1$
```

Figura 16: Importación de polígonos adicionales del mapa de la zona 1
Fuente: Autor. Generado por medio de la consola de Ubuntu 18.04.2.

Con el programa *randomTrips.py* se genera un conjunto de viajes aleatorios para una red determinada. Lo hace eligiendo el borde de origen y de destino de manera uniforme al azar o con una distribución modificada como se describe a continuación.

Crea los archivos: *zona1.rou.xml*, *zona1.rou.alt.xml* y *zona1.trips.trips.xml*.

En el fichero *.rou.xml* se definen los vehículos, tipos de vehículos y rutas. Y en el fichero *trips.trips.xml* se hacen definiciones de viaje para el DUAROUTER SUMO.



```
lewys@lewys-300E4C-300E5C-300E7C: ~/Escritorio/zona 1
Archivo Editar Ver Buscar Terminal Ayuda
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 1$ python /home/lewys/sumo-0.32.0/tools/randomTrips.py -n zona1.net.xml -r zona1.rou.xml -e 100 -l calling /home/lewys/sumo-0.32.0/bin/duarouter -n zona1.net.xml -r trips.trips.xml -o zona1.rou.xml --ignore-errors --begin 0 --end 100.0 --no-step-log --no-warnings
Success.
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 1$
```

Figura 17: Ejecución del programa *randomTrips.py* para la simulación de la zona 1
Fuente: Autor. Generado por medio de la consola de Ubuntu 18.04.2.

Para observar el mapa de la zona 1 convertido se debe crear un archivo de configuración *zona1.sumo.cfg*, donde se importan los archivos *zona1.net.xml* y *zona1.poly.xml*.



```
zona1.sumo.cfg
~/Escritorio/zona 1
Guardar
<configuration>
<input>
<net-file value="zona1.net.xml"/>
<route-files value="zona1.rou.xml"/>
<additional-files value="zona1.poly.xml"/>
</input>
<time>
<begin value="0"/>
<end value="100"/>
<step-length value="0.1"/>
</time>
</configuration>
Texto plano Anchura del tabulador: 8 Ln 12, Col 17 INS
```

Figura 18: Creación del fichero *zona1.sumo.cfg*
Fuente: Autor. Generado por medio de gedit de Ubuntu 18.04.2.

Para ver el mapa de la zona 1 se ejecuta el programa *sumo-gui* y el parámetro es el fichero *zona1.sumo.cfg*.

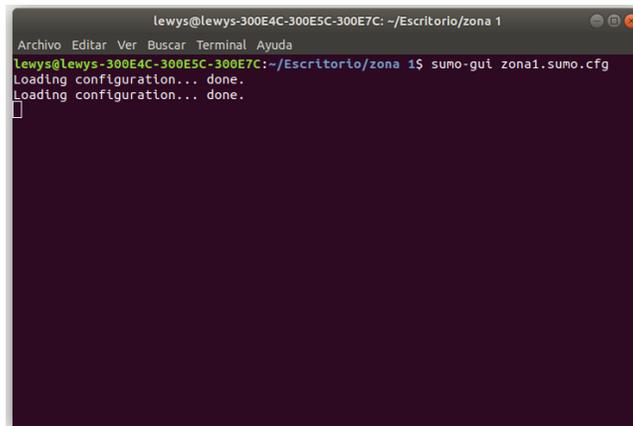


Figura 19: Ejecución del programa sumo-gui para la simulación de la zona 1
Fuente: Autor. Generado por medio de la consola de Ubuntu 18.04.2.

El mapa en ejecución de la zona 1 se puede observar en la Figura 20.

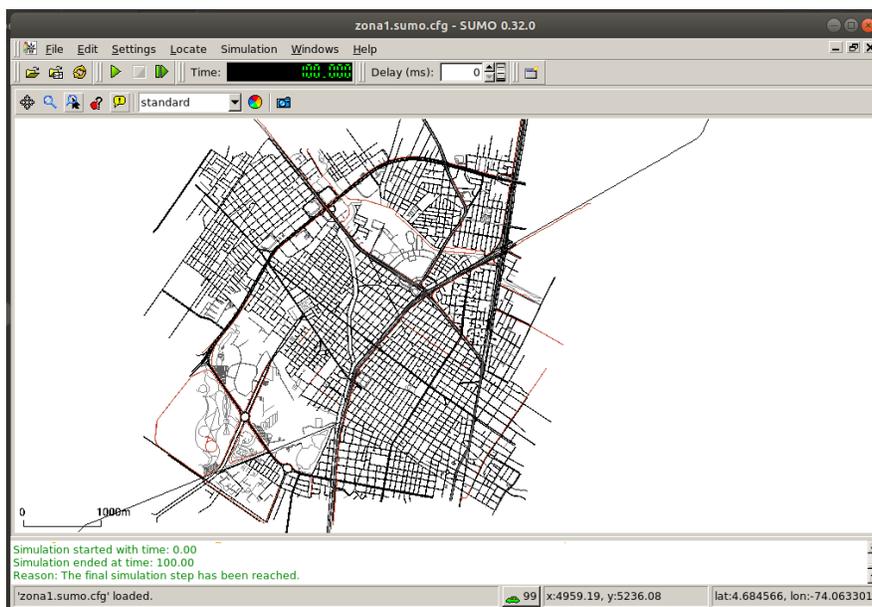
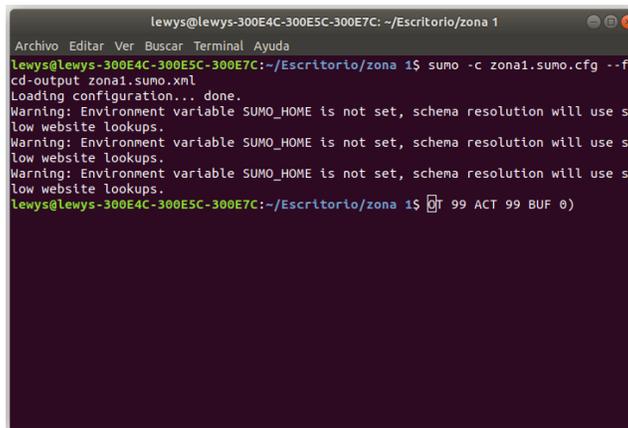


Figura 20: Mapa en ejecución de la zona 1
Fuente: Autor. Generado por medio de la herramienta sumo-0.32.0.

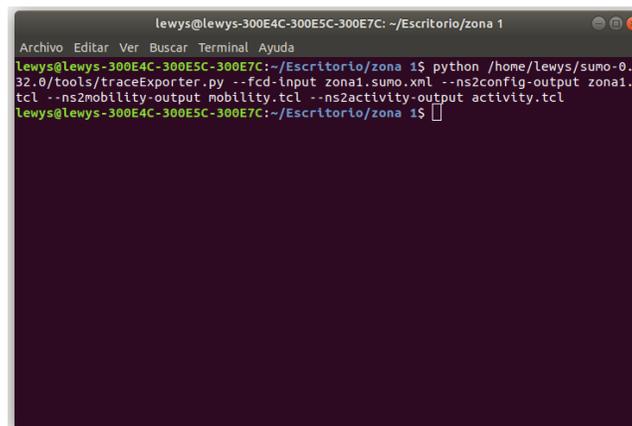
Con el programa sumo -c se crea el fichero *zona1.sumo.xml* que contiene información acerca de los vehículos como el tipo, la posición y la velocidad.



```
lewys@lewys-300E4C-300E5C-300E7C: ~/Escritorio/zona 1
Archivo Editar Ver Buscar Terminal Ayuda
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 1$ sumo -c zona1.sumo.cfg --f
cd-output zona1.sumo.xml
Loading configuration... done.
Warning: Environment variable SUMO_HOME is not set, schema resolution will use s
low website lookups.
Warning: Environment variable SUMO_HOME is not set, schema resolution will use s
low website lookups.
Warning: Environment variable SUMO_HOME is not set, schema resolution will use s
low website lookups.
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 1$ ^T 99 ACT 99 BUF 0)
```

Figura 21: Ejecución del programa `sumo -c` para la simulación de la zona 1
Fuente: Autor. Generado por medio de la consola de Ubuntu 18.04.2.

Con el programa `traceExporter.py` y con los parámetros `ns2config-output` se procede a crear el archivo básico de configuración de la red. Con el parámetro `ns2mobility-output` se genera la movilidad de la red y con el parámetro `ns2activity-output` se genera la actividad de la red.



```
lewys@lewys-300E4C-300E5C-300E7C: ~/Escritorio/zona 1
Archivo Editar Ver Buscar Terminal Ayuda
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 1$ python /home/lewys/sumo-0.
32.0/tools/traceExporter.py --fcd-input zona1.sumo.xml --ns2config-output zona1.
tcl --ns2mobility-output mobility.tcl --ns2activity-output activity.tcl
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 1$
```

Figura 22: Ejecución del programa `traceExporter.py` para la simulación de la zona 1
Fuente: Autor. Generado por medio de la consola de Ubuntu 18.04.2.

En el fichero `zona1.tcl` que se creó con el programa `ns2config-output` se procede a escribir el código de configuración de la red para la simulación.

Ejecución del programa *cbrgen.tcl* para la generación de tráfico de la zona 1.

```
lewys@lewys-300E4C-300E5C-300E7C: ~/Escritorio/ns-allinone-2.35/ns-2.35/indep-utils/cmu-...  
Archivo Editar Ver Buscar Terminal Ayuda  
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/ns-allinone-2.35/ns-2.35/indep-uti  
ls/cmu-scen-gen$ ns cbrgen.tcl -type cbr -nn 99 -seed 1.0 -nc 99 -rate 4.0 > tr  
fficzonal  
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/ns-allinone-2.35/ns-2.35/indep-uti  
ls/cmu-scen-gen$
```

Figura 23: Ejecución del programa *cbrgen.tcl* para la generación de tráfico de la zona 1
Fuente: Autor. Generado por medio de la consola de Ubuntu 18.04.2.

Una vez construida la simulación para la zona 1 se ejecuta con el programa *ns*.

```
lewys@lewys-300E4C-300E5C-300E7C: ~/Escritorio/zona 1  
Archivo Editar Ver Buscar Terminal Ayuda  
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 1$ ns zona1.tcl  
num_nodes is set 99  
warning: Please use -channel as shown in tcl/ex/wireless-mitf.tcl  
INITIALIZE THE LIST xListHead  
Starting Simulation...  
SORTING LISTS ...DONE!  
channel.cc:sendUp - Calc highestAntennaZ_ and distCST_  
highestAntennaZ_ = 1.5, distCST_ = 550.0  
NS EXITING...  
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 1$
```

Figura 24: Ejecución de la simulación para la zona 1 de una red Vanet
Fuente: Autor. Generado por medio de la consola de Ubuntu 18.04.2.

Simulación zona 1 de red Vanet interconectada con IoT

Ejecución de la simulación de una red Vanet con IoT para la zona 1.

```
lewys@lewys-300E4C-300E5C-300E7C: ~/Descargas/simulacion
Archivo Editar Ver Buscar Terminal Ayuda
lewys@lewys-300E4C-300E5C-300E7C:~/Descargas/simulacion$ ns map.tcl
num_nodes is set 47
warning: Please use -channel as shown in tcl/ex/wireless-mitf.tcl
INITIALIZE THE LIST xListHead
Starting Simulation...
SORTING LISTS ...DONE!
channel.cc:sendUp - Calc highestAntennaZ_ and distCST_
highestAntennaZ_ = 1.5, distCST_ = 550.0
NS EXITING...
```

Figura 25: Ejecución de la simulación para la zona 1 de una red Vanet interconectada con IoT
Fuente: Autor. Generado por medio de la consola de Ubuntu 18.04.2.

Simulación zona 2 de red Vanet tradicional

Obtener el mapa en la plataforma *openstreetmap.org* para la simulación de la zona 2.

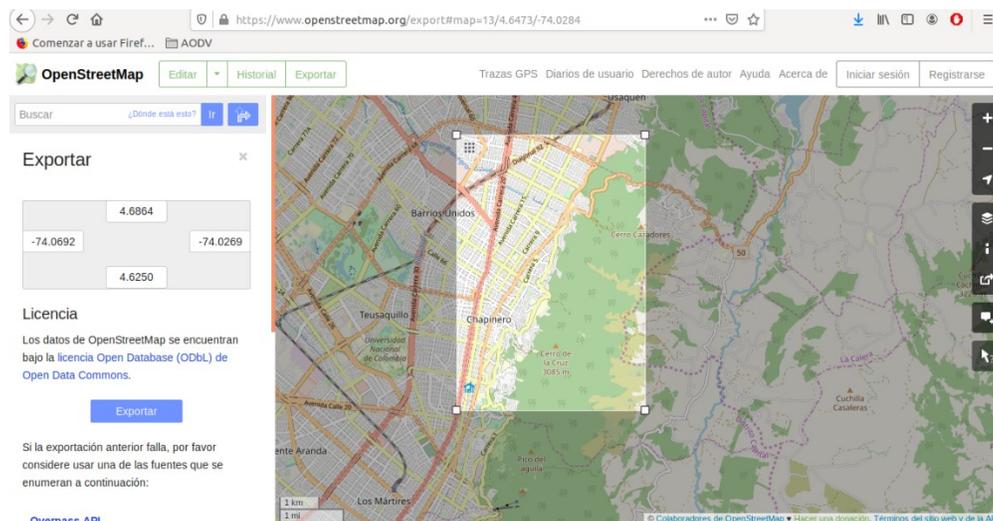
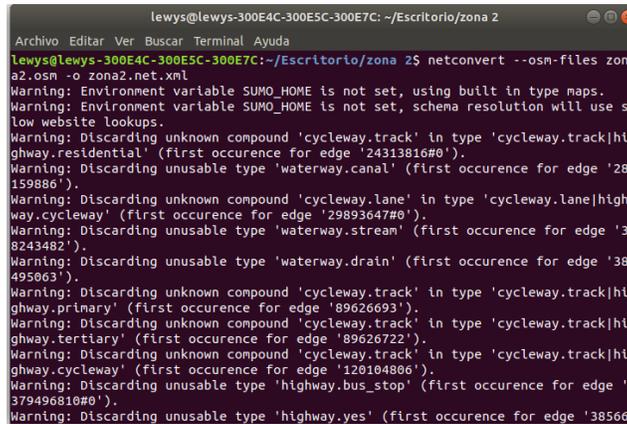


Figura 26: Obtención del mapa para la zona 2
Fuente: Autor. Generada por medio de la plataforma *openstreetmap.org*

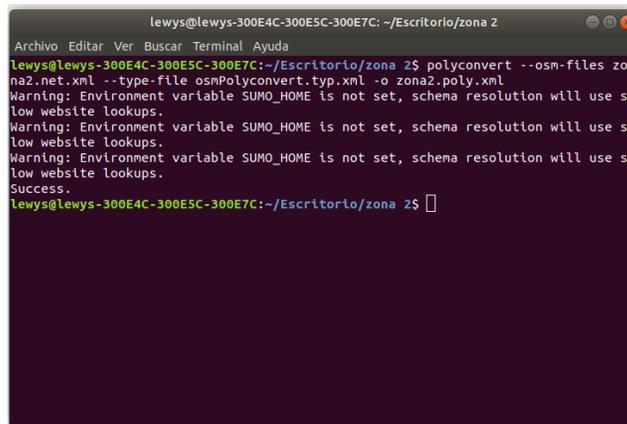
Conversión de carreteras digitales del mapa de la zona 2 a redes de carreteras por medio del programa *netconvert* del simulador Sumo.



```
lewys@lewys-300E4C-300E5C-300E7C: ~/Escritorio/zona 2
Archivo Editar Ver Buscar Terminal Ayuda
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 2$ netconvert --osm-files zona2.osm -o zona2.net.xml
Warning: Environment variable SUMO_HOME is not set, using built in type maps.
Warning: Environment variable SUMO_HOME is not set, schema resolution will use slow website lookups.
Warning: Discarding unknown compound 'cycleway.track' in type 'cycleway.track|highway.residential' (first occurrence for edge '24313816#0').
Warning: Discarding unusable type 'waterway.canal' (first occurrence for edge '28159886').
Warning: Discarding unknown compound 'cycleway.lane' in type 'cycleway.lane|highway.cycleway' (first occurrence for edge '29893647#0').
Warning: Discarding unusable type 'waterway.stream' (first occurrence for edge '38243482').
Warning: Discarding unusable type 'waterway.drain' (first occurrence for edge '38495063').
Warning: Discarding unknown compound 'cycleway.track' in type 'cycleway.track|highway.primary' (first occurrence for edge '89626693').
Warning: Discarding unknown compound 'cycleway.track' in type 'cycleway.track|highway.tertiary' (first occurrence for edge '89626722').
Warning: Discarding unknown compound 'cycleway.track' in type 'cycleway.track|highway.cycleway' (first occurrence for edge '120104806').
Warning: Discarding unusable type 'highway.bus_stop' (first occurrence for edge '379496810#0').
Warning: Discarding unusable type 'highway.yes' (first occurrence for edge '38566
```

Figura 27: Conversión del mapa de la zona 2 a redes de carreteras
Fuente: Autor. Generado por medio de la consola de Ubuntu 18.04.2.

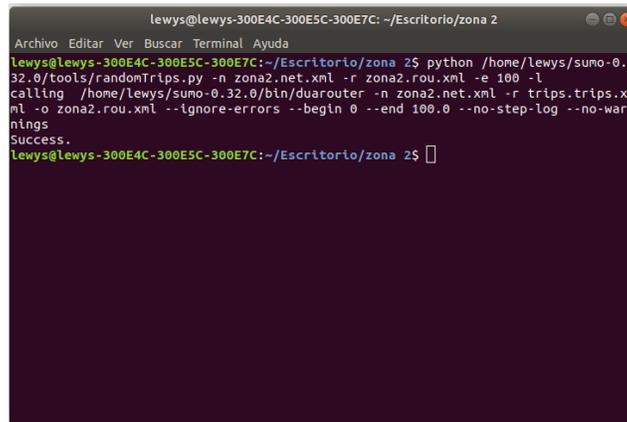
Importación de polígonos adicionales del mapa de la zona 2 por medio del programa *polyconvert* y creación del fichero *zona2.poly.xml*.



```
lewys@lewys-300E4C-300E5C-300E7C: ~/Escritorio/zona 2
Archivo Editar Ver Buscar Terminal Ayuda
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 2$ polyconvert --osm-files zona2.net.xml --type-file osmPolyconvert.typ.xml -o zona2.poly.xml
Warning: Environment variable SUMO_HOME is not set, schema resolution will use slow website lookups.
Warning: Environment variable SUMO_HOME is not set, schema resolution will use slow website lookups.
Warning: Environment variable SUMO_HOME is not set, schema resolution will use slow website lookups.
Success.
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 2$
```

Figura 28: Importación de polígonos adicionales del mapa de la zona 2
Fuente: Autor. Generado por medio de la consola de Ubuntu 18.04.2.

Ejecución del programa *randomTrips.py* para la simulación de la zona 2



```
lewys@lewys-300E4C-300E5C-300E7C: ~/Escritorio/zona 2
Archivo Editar Ver Buscar Terminal Ayuda
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 2$ python /home/lewys/sumo-0.
32.0/tools/randomTrips.py -n zona2.net.xml -r zona2.rou.xml -e 100 -l
calling /home/lewys/sumo-0.32.0/bin/duarouter -n zona2.net.xml -r trips.trips.x
ml -o zona2.rou.xml --ignore-errors --begin 0 --end 100.0 --no-step-log --no-war
nings
Success.
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 2$
```

Figura 29: Ejecución del programa *randomTrips.py* para la simulación de la zona 2
Fuente: Autor. Generado por medio de la consola de Ubuntu 18.04.2.

Creación del archivo de configuración *zona2.sumo.cfg* e importación de los archivos *zona2.net.xml* y *zona2.poly.xml*.



```
zona2.sumo.cfg
Guardar
<<configuration>
  <input>
    <net-file value="zona2.net.xml"/>
  </input>
  <route-files value="zona2.rou.xml"/>
  <additional-files value="zona2.poly.xml"/>
</configuration>
<time>
  <begin value="0"/>
  <end value="100"/>
  <step-length value="0.1"/>
</time>
</configuration>
```

Figura 30: Creación del fichero *zona2.sumo.cfg*
Fuente: Autor. Generado por medio de gedit de Ubuntu 18.04.2.

Ejecución del programa *sumo-gui* con parámetro *zona2.sumo.cfg* para observar el mapa convertido de la zona 2.

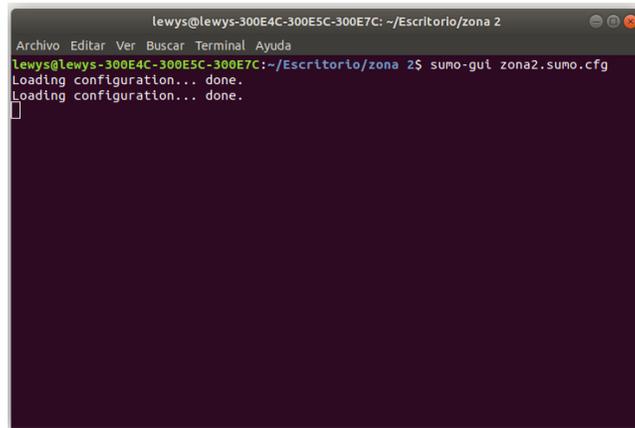


Figura 31: Ejecución del programa *sumo-gui* para la simulación de la zona 2
Fuente: Autor. Generado por medio de la consola de Ubuntu 18.04.2.

El mapa en ejecución de la zona 2 se puede observar en la Figura 32.

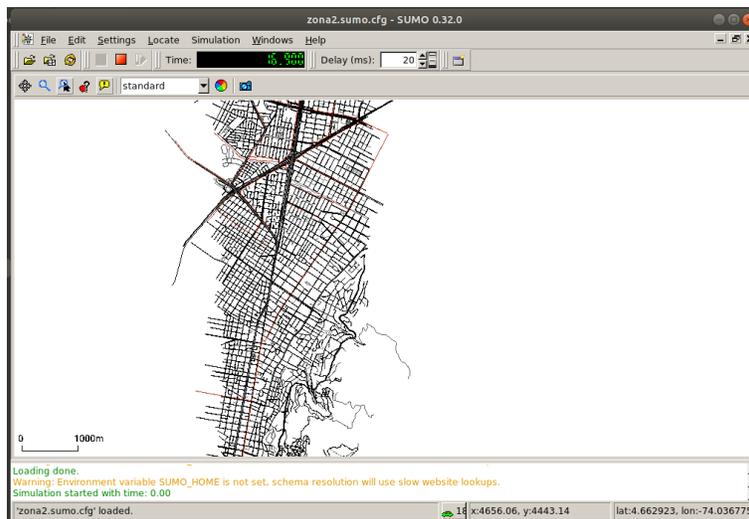
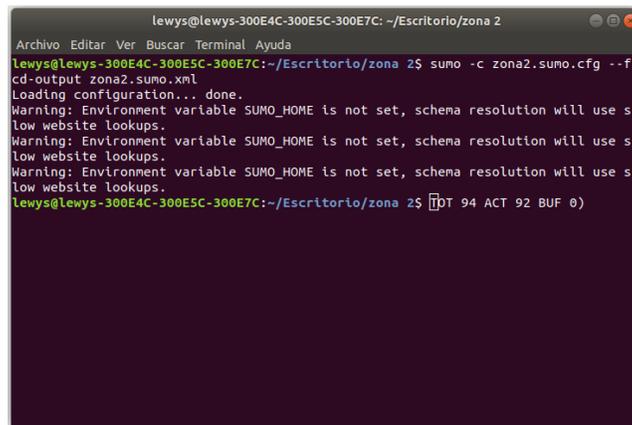


Figura 32: Mapa en ejecución de la zona 2
Fuente: Autor. Generado por medio de la herramienta sumo-0.32.0.

Creación del fichero *zona2.sumo.xml* con el programa `sumo -c`.

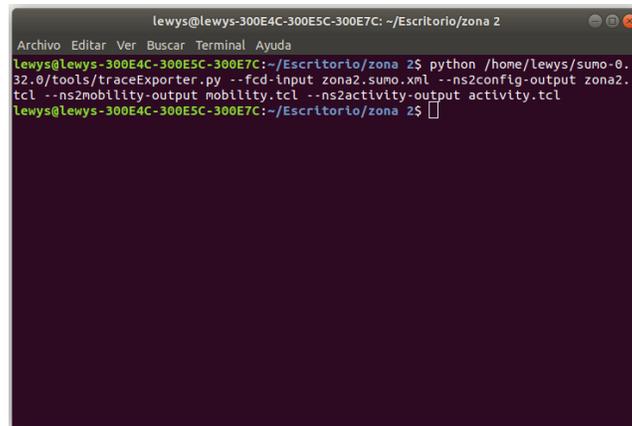


```
lewys@lewys-300E4C-300E5C-300E7C: ~/Escritorio/zona 2
Archivo Editar Ver Buscar Terminal Ayuda
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 2$ sumo -c zona2.sumo.cfg --f
cd-output zona2.sumo.xml
Loading configuration... done.
Warning: Environment variable SUMO_HOME is not set, schema resolution will use s
low website lookups.
Warning: Environment variable SUMO_HOME is not set, schema resolution will use s
low website lookups.
Warning: Environment variable SUMO_HOME is not set, schema resolution will use s
low website lookups.
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 2$ [t] 94 ACT 92 BUF 0)
```

Figura 33: Ejecución del programa `sumo -c` para la simulación de la zona 2
Fuente: Autor. Generado por medio de la consola de Ubuntu 18.04.2.

Con el programa `traceExporter.py` y con los parámetros `ns2config-output` se procede a

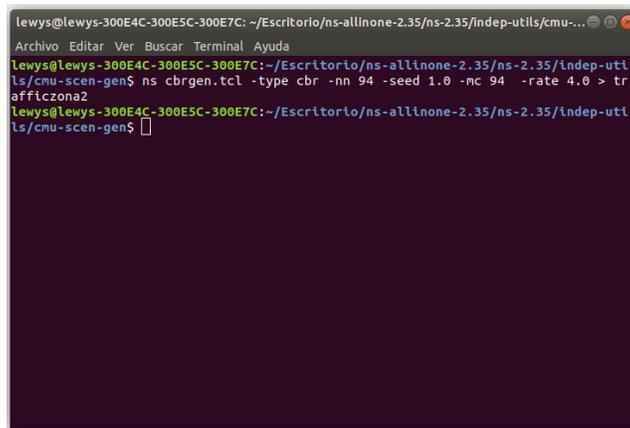
Creación del archivo básico de configuración de la red, del archivo de movilidad de la red y del archivo de actividad de la red para la simulación de la zona 2.



```
lewys@lewys-300E4C-300E5C-300E7C: ~/Escritorio/zona 2
Archivo Editar Ver Buscar Terminal Ayuda
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 2$ python /home/lewys/sumo-0.
32.0/tools/traceExporter.py --fcd-input zona2.sumo.xml --ns2config-output zona2.
tcl --ns2mobility-output mobility.tcl --ns2activity-output activity.tcl
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 2$
```

Figura 34: Ejecución del programa `traceExporter.py` para la simulación de la zona 2
Fuente: Autor. Generado por medio de la consola de Ubuntu 18.04.2.

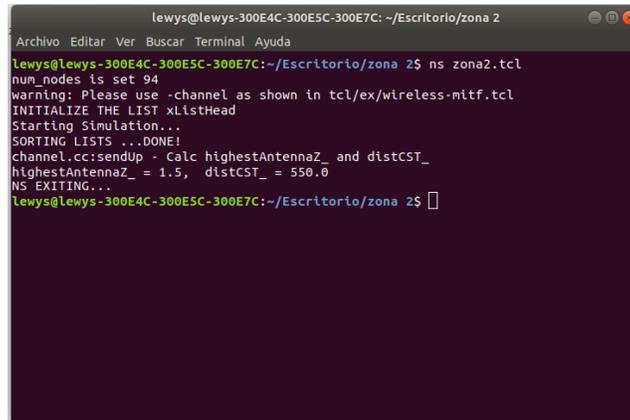
Ejecución del programa *cbrgen.tcl* para la generación de tráfico de la zona 2.



```
lewys@lewys-300E4C-300E5C-300E7C: ~/Escritorio/ns-allinone-2.35/ns-2.35/indep-utils/cmu-...
Archivo Editar Ver Buscar Terminal Ayuda
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/ns-allinone-2.35/ns-2.35/indep-uti
ls/cmu-scen-gen$ ns cbrgen.tcl -type cbr -nn 94 -seed 1.0 -nc 94 -rate 4.0 > tr
fficzona2
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/ns-allinone-2.35/ns-2.35/indep-uti
ls/cmu-scen-gen$
```

Figura 35: Ejecución del programa *cbrgen.tcl* para la generación de tráfico de la zona 2
Fuente: Autor. Generado por medio de la consola de Ubuntu 18.04.2.

Ejecución de la simulación para la zona 2 creada el fichero *zona2.tcl*.

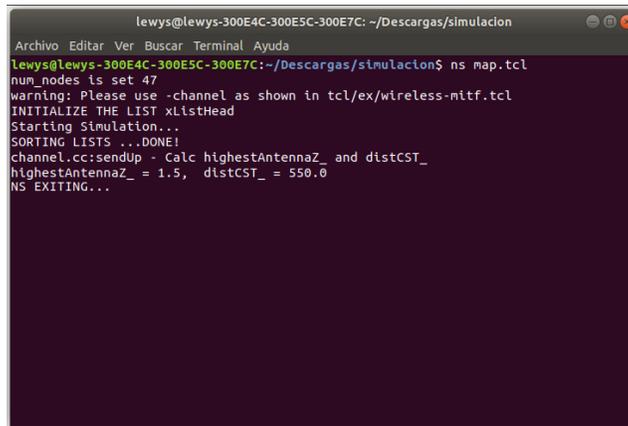


```
lewys@lewys-300E4C-300E5C-300E7C: ~/Escritorio/zona 2
Archivo Editar Ver Buscar Terminal Ayuda
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 2$ ns zona2.tcl
num_nodes ls set 94
warning: Please use -channel as shown in tcl/ex/wireless-mitf.tcl
INITIALIZE THE LIST xListHead
Starting Simulation...
SORTING LISTS ...DONE!
channel.cc:sendUp - Calc highestAntenna_ and distCST_
highestAntennaZ_ = 1.5, distCST_ = 550.0
NS EXITING...
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 2$
```

Figura 36: Ejecución de la simulación para la zona 2 de una red Vanet
Fuente: Autor. Generado por medio de la consola de Ubuntu 18.04.2.

Simulación zona 2 de red Vanet interconectada con IoT

Ejecución de la simulación de una red Vanet con IoT para la zona 2.



```
lewys@lewys-300E4C-300E5C-300E7C: ~/Descargas/simulacion
Archivo Editar Ver Buscar Terminal Ayuda
lewys@lewys-300E4C-300E5C-300E7C:~/Descargas/simulacion$ ns map.tcl
num_nodes is set 47
warning: Please use -channel as shown in tcl/ex/wireless-mitf.tcl
INITIALIZE THE LIST xListHead
Starting Simulation...
SORTING LISTS ...DONE!
channel.cc:sendup - Calc highestAntennaZ_ and distCST_
highestAntennaZ_ = 1.5, distCST_ = 550.0
NS EXITING...
```

Figura 37: Ejecución de la simulación para la zona 2 de una red Vanet interconectada con IoT Fuente: Autor. Generado por medio de la consola de Ubuntu 18.04.2.

Simulación zona 3 de red Vanet tradicional

Obtener el mapa en la plataforma *openstreetmap.org* para la simulación de la zona 3.

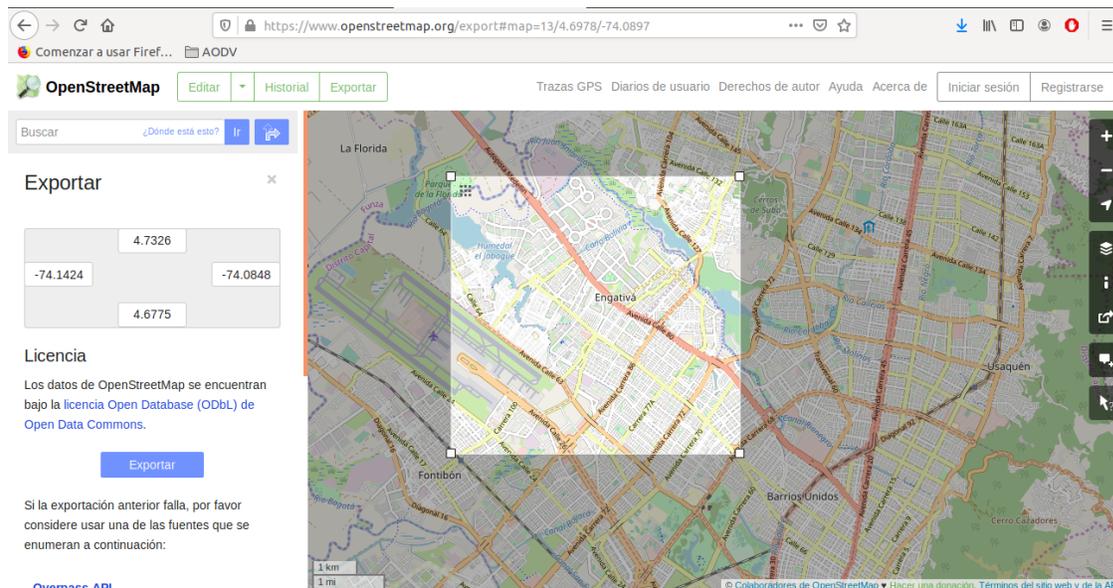
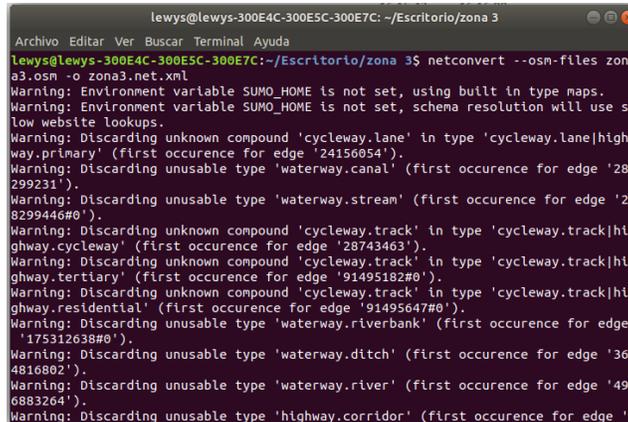


Figura 38: Obtención del mapa para la zona 3 Fuente: Autor. Generada por medio de la plataforma *openstreetmap.org*

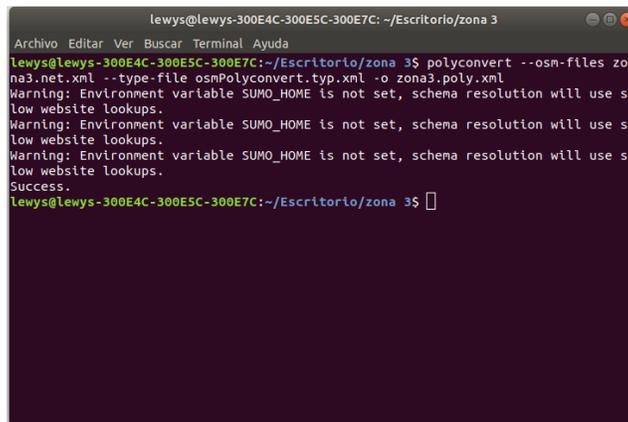
Conversión de carreteras digitales del mapa de la zona 3 a redes de carreteras por medio del programa *netconvert* del simulador Sumo.



```
lewis@lewis-300E4C-300E5C-300E7C: ~/Escritorio/zona 3
Archivo Editar Ver Buscar Terminal Ayuda
lewis@lewis-300E4C-300E5C-300E7C:~/Escritorio/zona 3$ netconvert --osm-files zon
a3.osm -o zona3.net.xml
Warning: Environment variable SUMO_HOME is not set, using built in type maps.
Warning: Environment variable SUMO_HOME is not set, schema resolution will use s
low website lookups.
Warning: Discarding unknown compound 'cycleway.lane' in type 'cycleway.lane|high
way.primary' (first occurrence for edge '24156054').
Warning: Discarding unusable type 'waterway.canal' (first occurrence for edge '28
299231').
Warning: Discarding unusable type 'waterway.stream' (first occurrence for edge '2
8299446#0').
Warning: Discarding unknown compound 'cycleway.track' in type 'cycleway.track|hi
ghway.cycleway' (first occurrence for edge '28743463').
Warning: Discarding unknown compound 'cycleway.track' in type 'cycleway.track|hi
ghway.tertiary' (first occurrence for edge '91495182#0').
Warning: Discarding unknown compound 'cycleway.track' in type 'cycleway.track|hi
ghway.residential' (first occurrence for edge '91495647#0').
Warning: Discarding unusable type 'waterway.riverbank' (first occurrence for edge
'17531263#0').
Warning: Discarding unusable type 'waterway.ditch' (first occurrence for edge '36
4816802').
Warning: Discarding unusable type 'waterway.river' (first occurrence for edge '49
6883264').
Warning: Discarding unusable type 'highway.corridor' (first occurrence for edge '
```

Figura 39: Conversión del mapa de la zona 3 a redes de carreteras
Fuente: Autor. Generado por medio de la consola de Ubuntu 18.04.2.

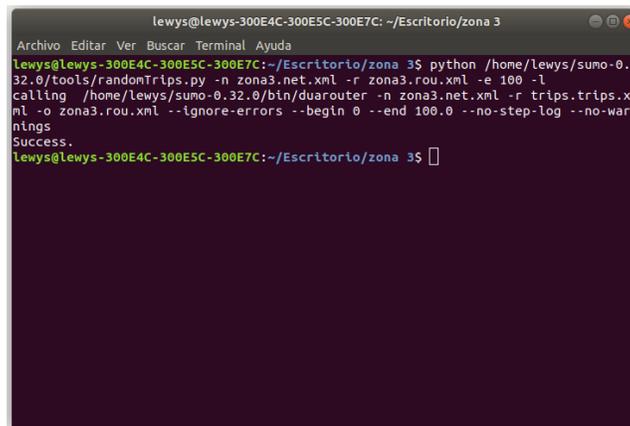
Importación de polígonos adicionales del mapa de la zona 3 por medio del programa *polyconvert* y creación del fichero *zona3.poly.xml*.



```
lewis@lewis-300E4C-300E5C-300E7C: ~/Escritorio/zona 3
Archivo Editar Ver Buscar Terminal Ayuda
lewis@lewis-300E4C-300E5C-300E7C:~/Escritorio/zona 3$ polyconvert --osm-files zo
na3.net.xml --type-file osmPolyconvert.typ.xml -o zona3.poly.xml
Warning: Environment variable SUMO_HOME is not set, schema resolution will use s
low website lookups.
Warning: Environment variable SUMO_HOME is not set, schema resolution will use s
low website lookups.
Warning: Environment variable SUMO_HOME is not set, schema resolution will use s
low website lookups.
Warning: Environment variable SUMO_HOME is not set, schema resolution will use s
low website lookups.
Success.
lewis@lewis-300E4C-300E5C-300E7C:~/Escritorio/zona 3$
```

Figura 40: Importación de polígonos adicionales del mapa de la zona 3
Fuente: Autor. Generado por medio de la consola de Ubuntu 18.04.2.

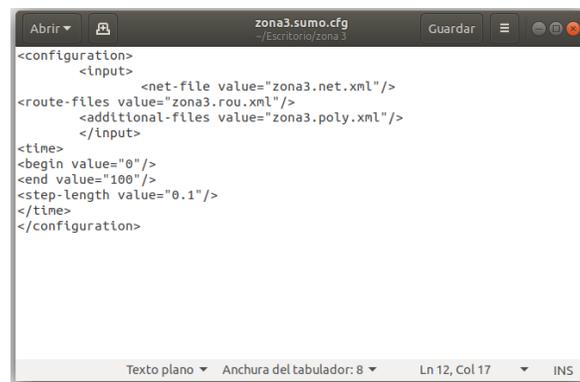
Ejecución del programa *randomTrips.py* para la simulación de la zona 3



```
lewys@lewys-300E4C-300E5C-300E7C: ~/Escritorio/zona 3
Archivo Editar Ver Buscar Terminal Ayuda
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 3$ python /home/lewys/sumo-0.32.0/tools/randomTrips.py -n zona3.net.xml -r zona3.rou.xml -e 100 -l
calling /home/lewys/sumo-0.32.0/bin/duarouter -n zona3.net.xml -r trips.trips.xml -o zona3.rou.xml --ignore-errors --begin 0 --end 100.0 --no-step-log --no-warnings
Success.
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 3$
```

Figura 41: Ejecución del programa *randomTrips.py* para la simulación de la zona 4
Fuente: Autor. Generado por medio de la consola de Ubuntu 18.04.2.

Creación del archivo de configuración *zona3.sumo.cfg* e importación de los archivos *zona3.net.xml* y *zona3.poly.xml*.



```
zona3.sumo.cfg
~/Escritorio/zona 3
Guardar
<configuration>
  <input>
    <net-file value="zona3.net.xml"/>
  </input>
  <route-files value="zona3.rou.xml"/>
  <additional-files value="zona3.poly.xml"/>
</input>
<time>
  <begin value="0"/>
  <end value="100"/>
  <step-length value="0.1"/>
</time>
</configuration>
Texto plano Anchura del tabulador: 8 Ln 12, Col 17 INS
```

Figura 42: Creación del fichero *zona3.sumo.cfg*
Fuente: Autor. Generado por medio de gedit de Ubuntu 18.04.2.

Ejecución del programa *sumo-gui* con parámetro *zona3.sumo.cfg* para observar el mapa convertido de la zona 3.

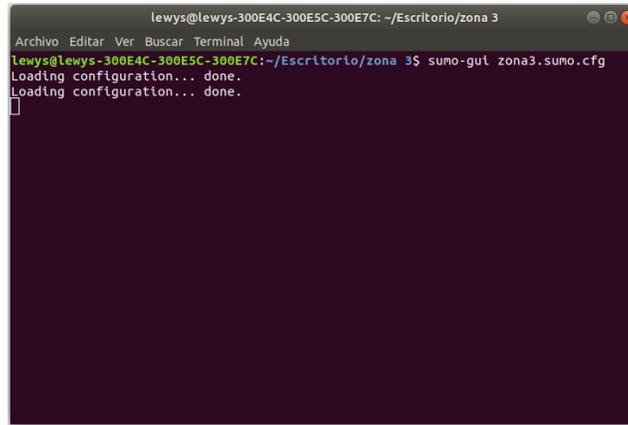


Figura 43: Ejecución del programa *sumo-gui* para la simulación de la zona 3
Fuente: Autor. Generado por medio de la consola de Ubuntu 18.04.2.

El mapa en ejecución de la zona 3 se puede observar en la Figura 44.

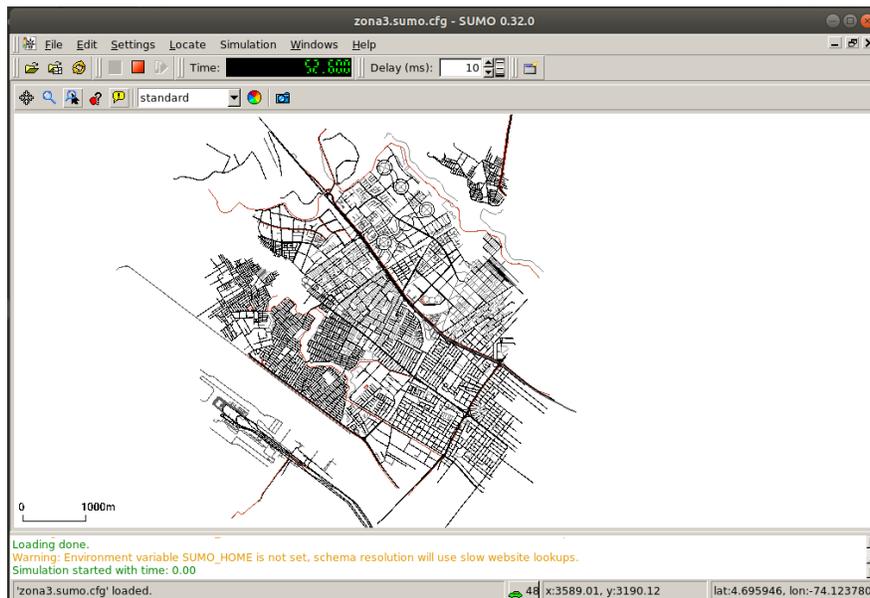
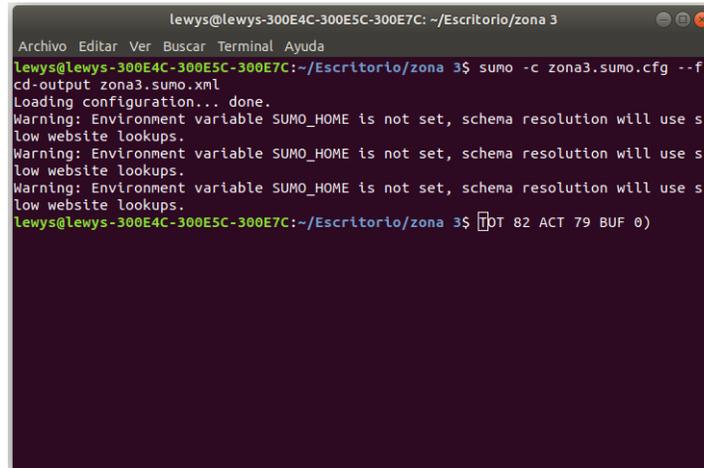


Figura 44: Mapa en ejecución de la zona 3
Fuente: Autor. Generado por medio de la herramienta *sumo-0.32.0*.

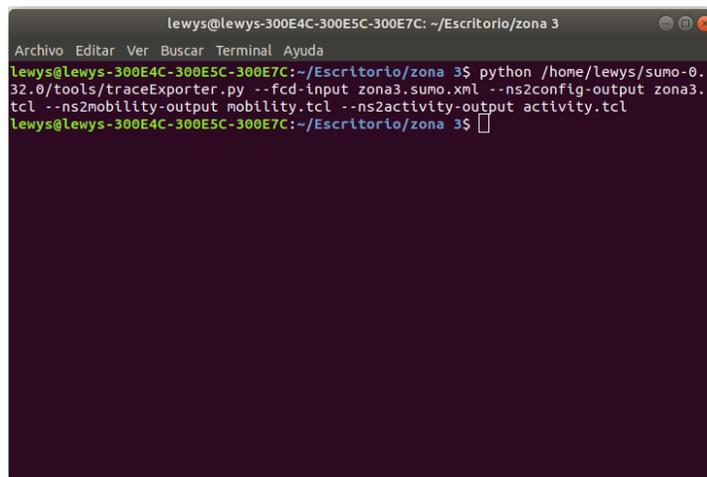
Creación del fichero *zona3.sumo.xml* con el programa *sumo -c*.



```
lewys@lewys-300E4C-300E5C-300E7C: ~/Escritorio/zona 3
Archivo Editar Ver Buscar Terminal Ayuda
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 3$ sumo -c zona3.sumo.cfg --f
cd-output zona3.sumo.xml
Loading configuration... done.
Warning: Environment variable SUMO_HOME is not set, schema resolution will use s
low website lookups.
Warning: Environment variable SUMO_HOME is not set, schema resolution will use s
low website lookups.
Warning: Environment variable SUMO_HOME is not set, schema resolution will use s
low website lookups.
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 3$
```

Figura 45: Ejecución del programa *sumo -c* para la simulación de la zona 3
Fuente: Autor. Generado por medio de la consola de Ubuntu 18.04.2.

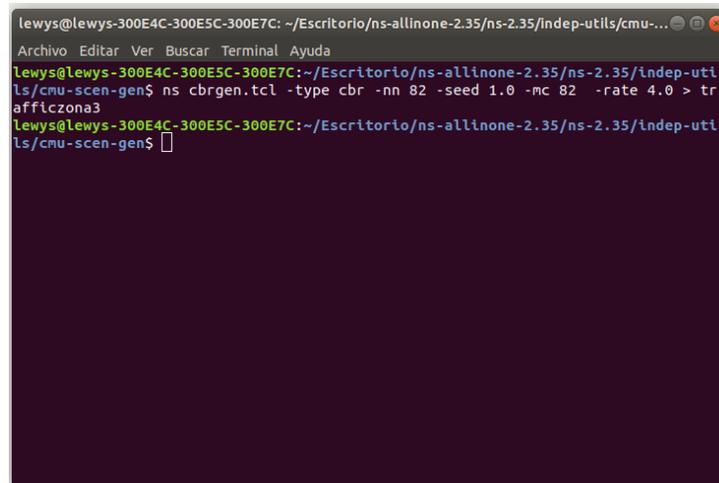
Creación del archivo básico de configuración de la red, del archivo de movilidad de la red y del archivo de actividad de la red para la simulación de la zona 3.



```
lewys@lewys-300E4C-300E5C-300E7C: ~/Escritorio/zona 3
Archivo Editar Ver Buscar Terminal Ayuda
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 3$ python /home/lewys/sumo-0.
32.0/tools/traceExporter.py --fcd-input zona3.sumo.xml --ns2config-output zona3.
tcl --ns2mobility-output mobility.tcl --ns2activity-output activity.tcl
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 3$
```

Figura 46: Ejecución del programa *traceExporter.py* para la simulación de la zona 3
Fuente: Autor. Generado por medio de la consola de Ubuntu 18.04.2.

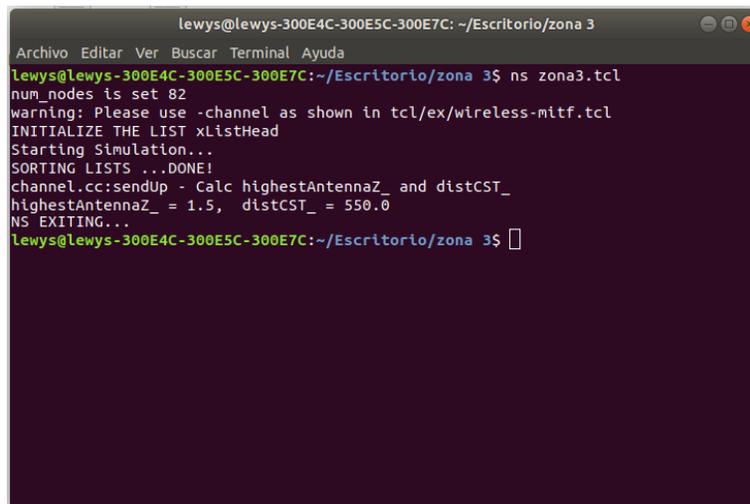
Ejecución del programa *cbrgen.tcl* para la generación de tráfico de la zona 3.



```
lewys@lewys-300E4C-300E5C-300E7C: ~/Escritorio/ns-allinone-2.35/ns-2.35/indep-utils/cmu-...
Archivo Editar Ver Buscar Terminal Ayuda
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/ns-allinone-2.35/ns-2.35/indep-uti
ls/cnu-scen-gen$ ns cbrgen.tcl -type cbr -nn 82 -seed 1.0 -mc 82 -rate 4.0 > tr
afflczona3
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/ns-allinone-2.35/ns-2.35/indep-uti
ls/cnu-scen-gen$
```

Figura 47: Ejecución del programa *cbrgen.tcl* para la generación de tráfico de la zona 3
Fuente: Autor. Generado por medio de la consola de Ubuntu 18.04.2.

Ejecución de la simulación para la zona 3 creada el fichero *zona3.tcl*.

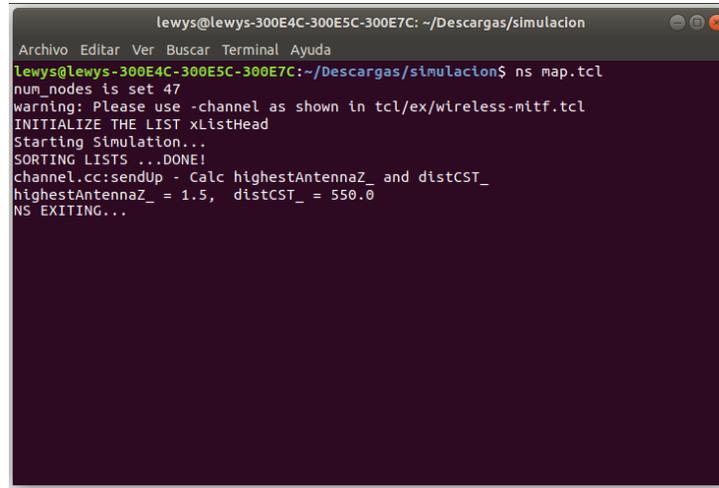


```
lewys@lewys-300E4C-300E5C-300E7C: ~/Escritorio/zona 3
Archivo Editar Ver Buscar Terminal Ayuda
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 3$ ns zona3.tcl
num_nodes is set 82
warning: Please use -channel as shown in tcl/ex/wireless-mitf.tcl
INITIALIZE THE LIST xListHead
Starting Simulation...
SORTING LISTS ...DONE!
channel.cc:sendUp - Calc highestAntennaZ_ and distCST_
highestAntennaZ_ = 1.5, distCST_ = 550.0
NS EXITING...
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 3$
```

Figura 48: Ejecución de la simulación para la zona 3 de una red Vanet
Fuente: Autor. Generado por medio de la consola de Ubuntu 18.04.2.

Simulación zona 3 de red Vanet interconectada con IoT

Ejecución de la simulación de una red Vanet con IoT para la zona 3.



```
lewys@lewys-300E4C-300E5C-300E7C: ~/Descargas/simulacion
Archivo Editar Ver Buscar Terminal Ayuda
lewys@lewys-300E4C-300E5C-300E7C:~/Descargas/simulacion$ ns map.tcl
num_nodes is set 47
warning: Please use -channel as shown in tcl/ex/wireless-mitf.tcl
INITIALIZE THE LIST xListHead
Starting Simulation...
SORTING LISTS ...DONE!
channel.cc:sendUp - Calc highestAntennaZ_ and distCST_
highestAntennaZ_ = 1.5, distCST_ = 550.0
NS EXITING...
```

Figura 49: Ejecución de la simulación para la zona 3 de una red Vanet interconectada con IoT
Fuente: Autor. Generado por medio de la consola de Ubuntu 18.04.2.

Simulación zona 4 de red Vanet tradicional

Obtener el mapa en la plataforma *openstreetmap.org* para la simulación de la zona 4.

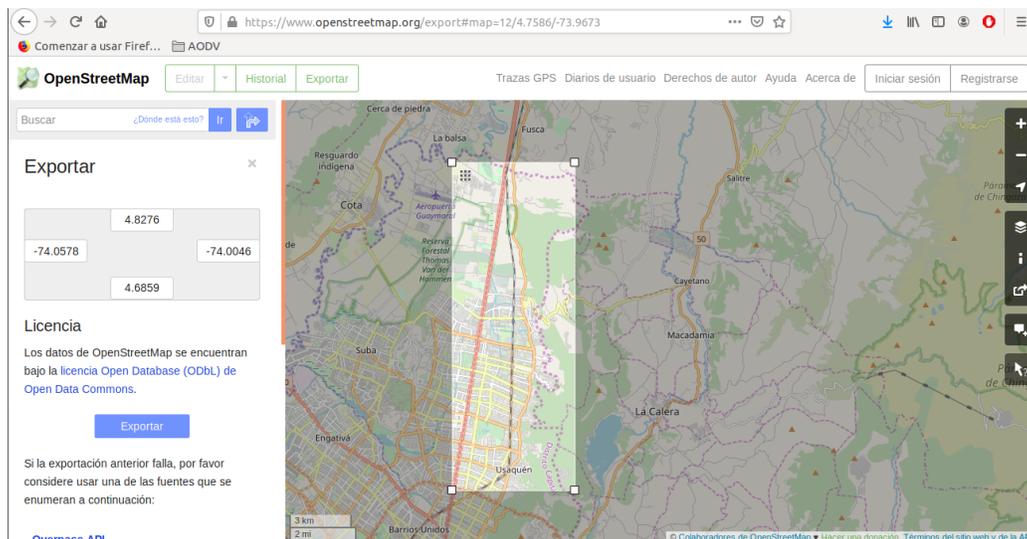
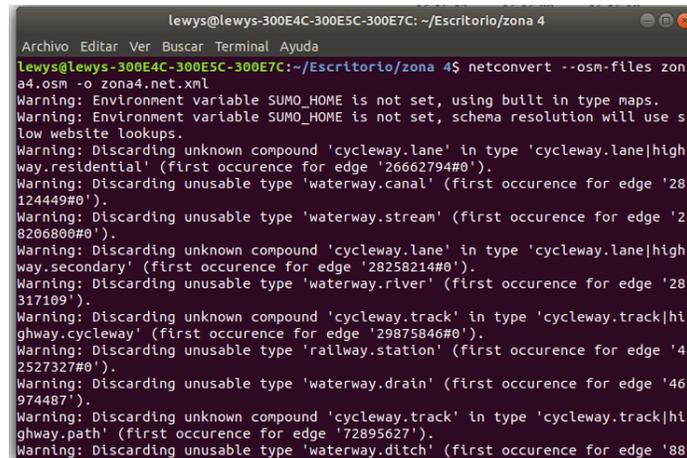


Figura 50: Obtención del mapa para la zona 4
Fuente: Autor. Generada por medio de la plataforma *openstreetmap.org*

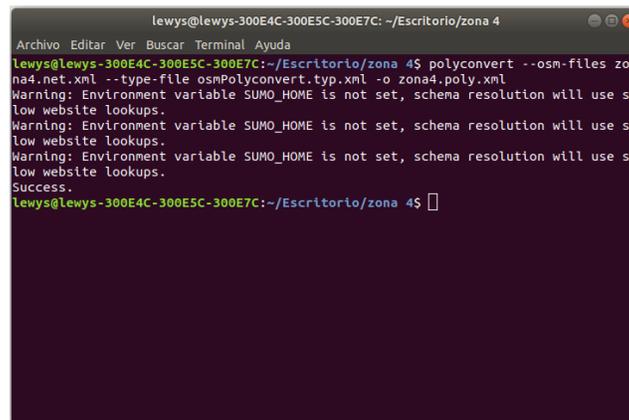
Conversión de carreteras digitales del mapa de la zona 4 a redes de carreteras por medio del programa *netconvert* del simulador Sumo.



```
lewys@lewys-300E4C-300E5C-300E7C: ~/Escritorio/zona 4
Archivo Editar Ver Buscar Terminal Ayuda
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 4$ netconvert --osm-files zona4.osm -o zona4.net.xml
Warning: Environment variable SUMO_HOME is not set, using built in type maps.
Warning: Environment variable SUMO_HOME is not set, schema resolution will use slow website lookups.
Warning: Discarding unknown compound 'cycleway.lane' in type 'cycleway.lane|highway.residential' (first occurrence for edge '26662794#0').
Warning: Discarding unusable type 'waterway.canal' (first occurrence for edge '28124449#0').
Warning: Discarding unusable type 'waterway.stream' (first occurrence for edge '28206800#0').
Warning: Discarding unknown compound 'cycleway.lane' in type 'cycleway.lane|highway.secondary' (first occurrence for edge '28258214#0').
Warning: Discarding unusable type 'waterway.river' (first occurrence for edge '28317109').
Warning: Discarding unknown compound 'cycleway.track' in type 'cycleway.track|highway.cycleway' (first occurrence for edge '29875846#0').
Warning: Discarding unusable type 'railway.station' (first occurrence for edge '42527327#0').
Warning: Discarding unusable type 'waterway.drain' (first occurrence for edge '46974487').
Warning: Discarding unknown compound 'cycleway.track' in type 'cycleway.track|highway.path' (first occurrence for edge '72895627').
Warning: Discarding unusable type 'waterway.ditch' (first occurrence for edge '88
```

Figura 51: Conversión del mapa de la zona 4 a redes de carreteras
Fuente: Autor. Generado por medio de la consola de Ubuntu 18.04.2.

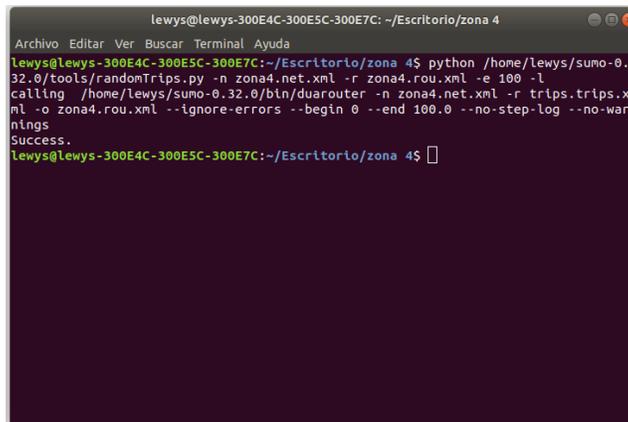
Importación de polígonos adicionales del mapa de la zona 4 por medio del programa *polyconvert* y creación del fichero *zona4.poly.xml*.



```
lewys@lewys-300E4C-300E5C-300E7C: ~/Escritorio/zona 4
Archivo Editar Ver Buscar Terminal Ayuda
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 4$ polyconvert --osm-files zona4.net.xml --type-file osmPolyconvert.typ.xml -o zona4.poly.xml
Warning: Environment variable SUMO_HOME is not set, schema resolution will use slow website lookups.
Warning: Environment variable SUMO_HOME is not set, schema resolution will use slow website lookups.
Warning: Environment variable SUMO_HOME is not set, schema resolution will use slow website lookups.
Success.
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 4$
```

Figura 52: Importación de polígonos adicionales del mapa de la zona 4
Fuente: Autor. Generado por medio de la consola de Ubuntu 18.04.2.

Ejecución del programa *randomTrips.py* para la simulación de la zona 4.



```
lewys@lewys-300E4C-300E5C-300E7C: ~/Escritorio/zona 4
Archivo Editar Ver Buscar Terminal Ayuda
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 4$ python /home/lewys/sumo-0.32.0/tools/randomTrips.py -n zona4.net.xml -r zona4.rou.xml -e 100 -l calling /home/lewys/sumo-0.32.0/bin/duarouter -n zona4.net.xml -r trips.trips.xml -o zona4.rou.xml --ignore-errors --begin 0 --end 100.0 --no-step-log --no-warnings
Success.
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 4$
```

Figura 53: Ejecución del programa *randomTrips.py* para la simulación de la zona 4
Fuente: Autor. Generado por medio de la consola de Ubuntu 18.04.2.

Creación del archivo de configuración *zona4.sumo.cfg* e importación de los archivos *zona4.net.xml* y *zona4.poly.xml*.



```
zona4.sumo.cfg
~/Escritorio/zona 4
Guardar
<<configuration>
  <input>
    <net-file value="zona4.net.xml"/>
    <route-files value="zona4.rou.xml"/>
    <additional-files value="zona4.poly.xml"/>
  </input>
  <time>
    <begin value="0"/>
    <end value="100"/>
    <step-length value="0.1"/>
  </time>
</configuration>
Texto plano Anchura del tabulador: 8 Ln 12, Col 17 INS
```

Figura 54: Creación del fichero *zona4.sumo.cfg*
Fuente: Autor. Generado por medio de gedit de Ubuntu 18.04.2.

Ejecución del programa *sumo-gui* con parámetro *zona4.sumo.cfg* para observar el mapa convertido de la zona 4.

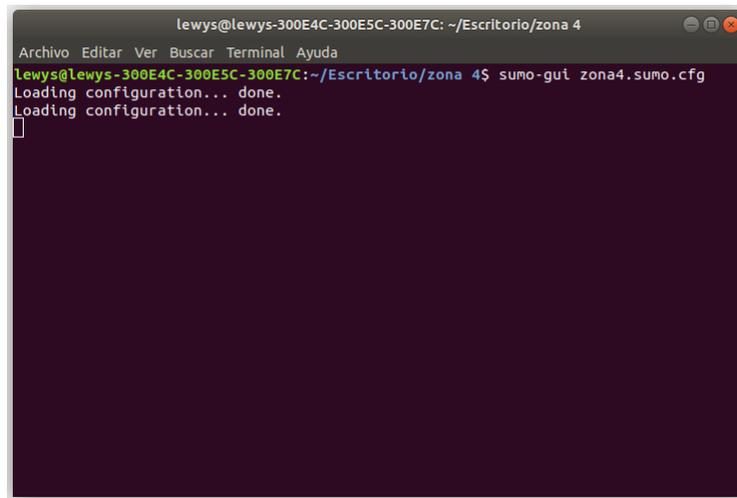


Figura 55: Ejecución del programa sumo-gui para la simulación de la zona 4
Fuente: Autor. Generado por medio de la consola de Ubuntu 18.04.2.

El mapa en ejecución de la zona 4 se puede observar en la Figura 56.

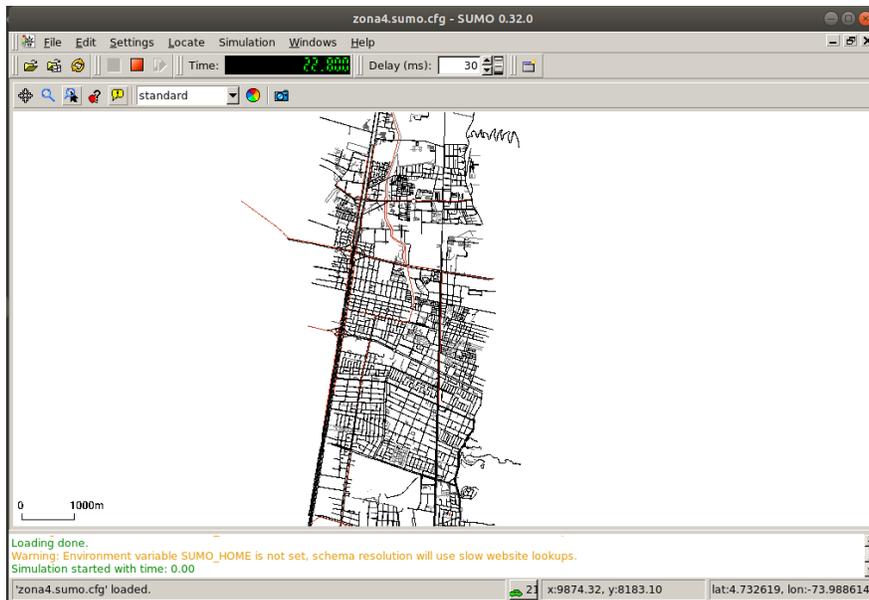
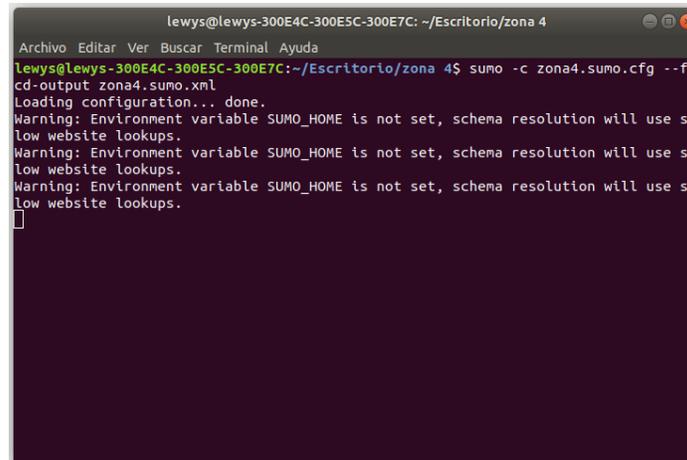


Figura 56: Mapa en ejecución de la zona 4
Fuente: Autor. Generado por medio de la herramienta sumo-0.32.0.

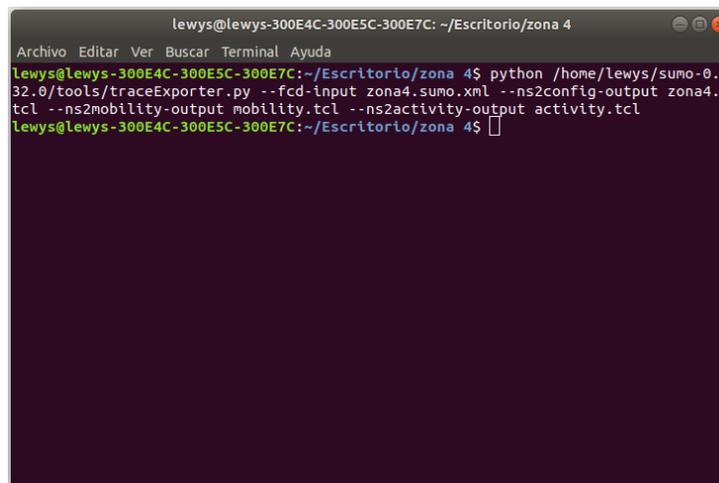
Creación del fichero *zona4.sumo.xml* con el programa `sumo -c`.



```
lewys@lewys-300E4C-300E5C-300E7C: ~/Escritorio/zona 4
Archivo Editar Ver Buscar Terminal Ayuda
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 4$ sumo -c zona4.sumo.cfg --f
cd-output zona4.sumo.xml
Loading configuration... done.
Warning: Environment variable SUMO_HOME is not set, schema resolution will use s
low website lookups.
Warning: Environment variable SUMO_HOME is not set, schema resolution will use s
low website lookups.
Warning: Environment variable SUMO_HOME is not set, schema resolution will use s
low website lookups.
```

Figura 57: Ejecución del programa `sumo -c` para la simulación de la zona 4
Fuente: Autor. Generado por medio de la consola de Ubuntu 18.04.2.

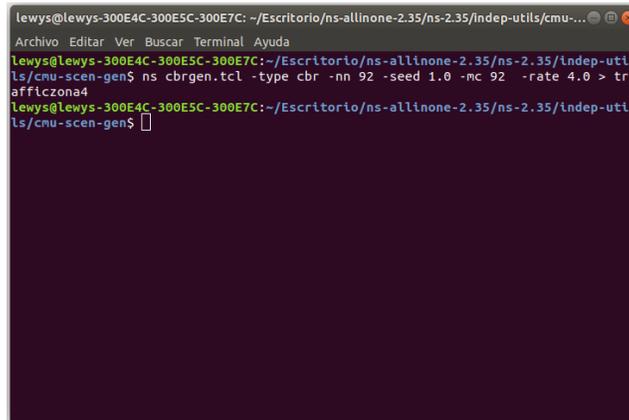
Creación del archivo básico de configuración de la red, del archivo de movilidad de la red y del archivo de actividad de la red para la simulación de la zona 4.



```
lewys@lewys-300E4C-300E5C-300E7C: ~/Escritorio/zona 4
Archivo Editar Ver Buscar Terminal Ayuda
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 4$ python /home/lewys/sumo-0.
32.0/tools/traceExporter.py --fcd-input zona4.sumo.xml --ns2config-output zona4.
tcl --ns2mobility-output mobility.tcl --ns2activity-output activity.tcl
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 4$
```

Figura 58: Ejecución del programa `traceExporter.py` para la simulación de la zona 4
Fuente: Autor. Generado por medio de la consola de Ubuntu 18.04.2.

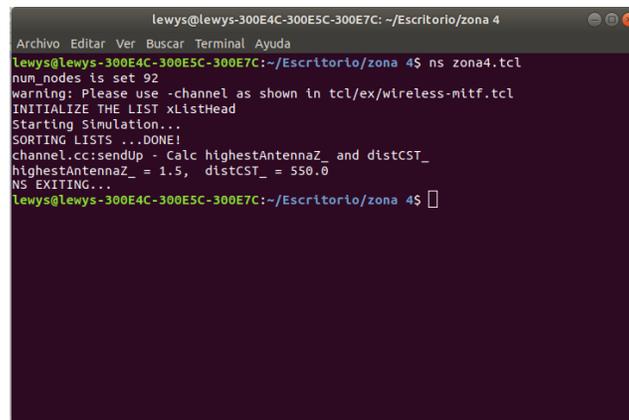
Ejecución del programa *cbrgen.tcl* para la generación de tráfico de la zona 4.



```
lewys@lewys-300E4C-300E5C-300E7C: ~/Escritorio/ns-allinone-2.35/ns-2.35/indep-utils/cmu-...
Archivo Editar Ver Buscar Terminal Ayuda
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/ns-allinone-2.35/ns-2.35/indep-uti
ls/cmu-scen-gen$ ns cbrgen.tcl -type cbr -nn 92 -seed 1.0 -mc 92 -rate 4.0 > tr
afficzona4
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/ns-allinone-2.35/ns-2.35/indep-uti
ls/cmu-scen-gen$
```

Figura 59: Ejecución del programa *cbrgen.tcl* para la generación de tráfico de la zona 4
Fuente: Autor. Generado por medio de la consola de Ubuntu 18.04.2.

Ejecución de la simulación para la zona 4 creada el fichero *zona4.tcl*.



```
lewys@lewys-300E4C-300E5C-300E7C: ~/Escritorio/zona 4
Archivo Editar Ver Buscar Terminal Ayuda
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 4$ ns zona4.tcl
num_nodes is set 92
warning: Please use -channel as shown in tcl/ex/wireless-mitf.tcl
INITIALIZE THE LIST xListHead
Starting Simulation...
SORTING LISTS ...DONE!
channel.cc:sendUp - Calc highestAntennaZ_ and distCST_
highestAntennaZ_ = 1.5, distCST_ = 550.0
NS EXITING...
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 4$
```

Figura 60: Ejecución de la simulación para la zona 4 de una red Vanet
Fuente: Autor. Generado por medio de la consola de Ubuntu 18.04.2.

Simulación zona 4 de red Vanet interconectada con IoT

Ejecución de la simulación de una red Vanet con IoT para la zona 4.

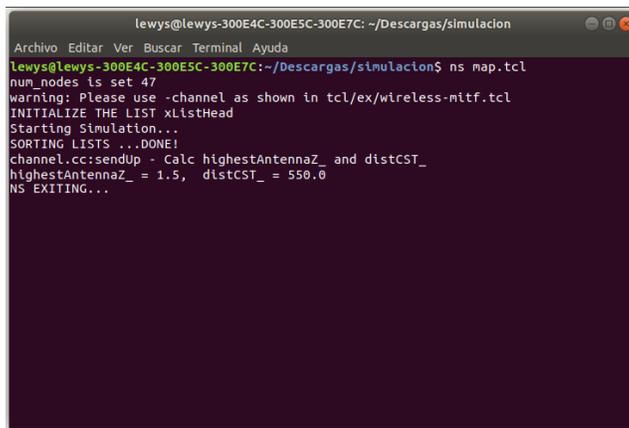


Figura 61: Ejecución de la simulación para la zona41 de una red Vanet interconectada con IoT
Fuente: Autor. Generado por medio de la consola de Ubuntu 18.04.2.

Simulación zona 5 de red Vanet tradicional

Obtener el mapa en la plataforma *openstreetmap.org* para la simulación de la zona 5.

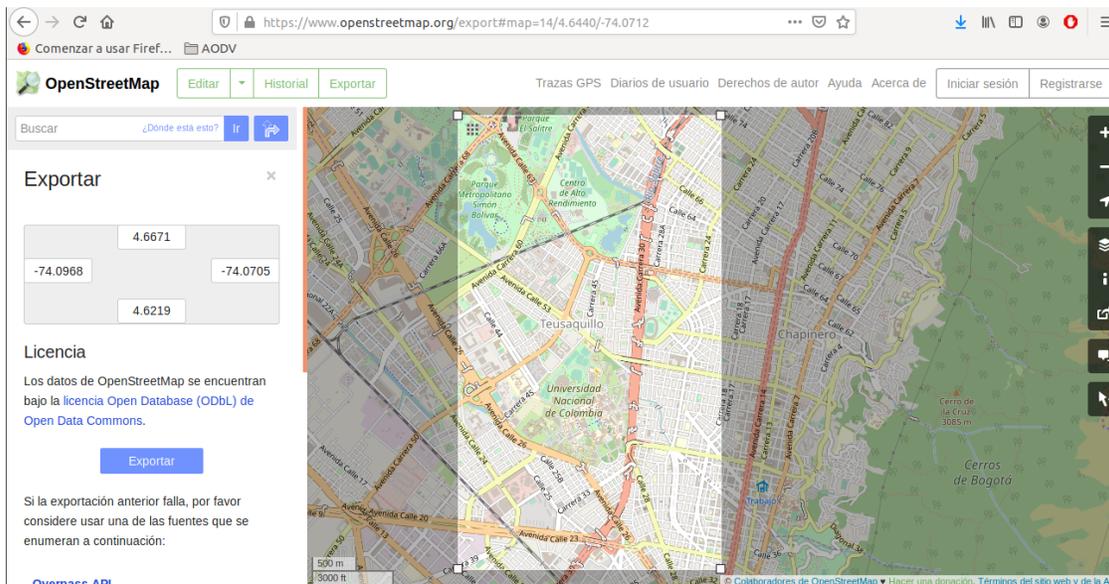
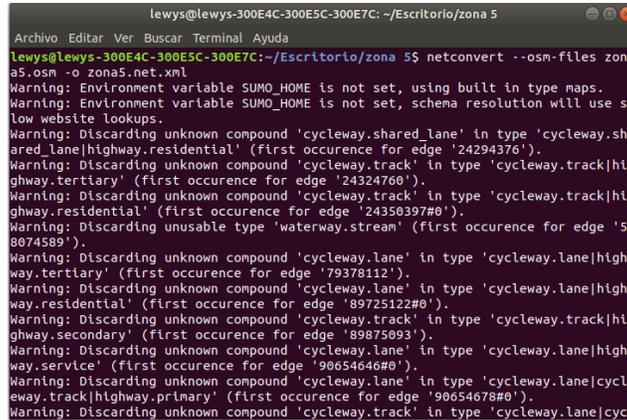


Figura 62: Obtención del mapa para la zona 5
Fuente: Autor. Generada por medio de la plataforma *openstreetmap.org*

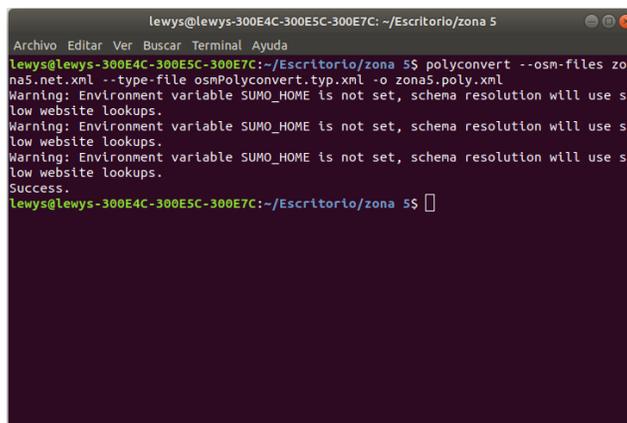
Conversión de carreteras digitales del mapa de la zona 5 a redes de carreteras por medio del programa *netconvert* del simulador Sumo.



```
lewys@lewys-300E4C-300E5C-300E7C: ~/Escritorio/zona 5
Archivo Editar Ver Buscar Terminal Ayuda
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 5$ netconvert --osm-files zona5.osm -o zona5.net.xml
Warning: Environment variable SUMO_HOME is not set, using built in type maps.
Warning: Environment variable SUMO_HOME is not set, schema resolution will use slow website lookups.
Warning: Discarding unknown compound 'cycleway.shared_lane' in type 'cycleway.shared_lane|highway.residential' (first occurrence for edge '24294376').
Warning: Discarding unknown compound 'cycleway.track' in type 'cycleway.track|highway.tertiary' (first occurrence for edge '24324760').
Warning: Discarding unknown compound 'cycleway.track' in type 'cycleway.track|highway.residential' (first occurrence for edge '24350397#0').
Warning: Discarding unusable type 'waterway.stream' (first occurrence for edge '58074589').
Warning: Discarding unknown compound 'cycleway.lane' in type 'cycleway.lane|highway.tertiary' (first occurrence for edge '79378112').
Warning: Discarding unknown compound 'cycleway.lane' in type 'cycleway.lane|highway.residential' (first occurrence for edge '89725122#0').
Warning: Discarding unknown compound 'cycleway.track' in type 'cycleway.track|highway.secondary' (first occurrence for edge '89875093').
Warning: Discarding unknown compound 'cycleway.lane' in type 'cycleway.lane|highway.service' (first occurrence for edge '90654646#0').
Warning: Discarding unknown compound 'cycleway.lane' in type 'cycleway.lane|cycleway.track|highway.primary' (first occurrence for edge '90654678#0').
Warning: Discarding unknown compound 'cycleway.track' in type 'cycleway.lane|cyc
```

Figura 63: Conversión del mapa de la zona 5 a redes de carreteras
Fuente: Autor. Generado por medio de la consola de Ubuntu 18.04.2.

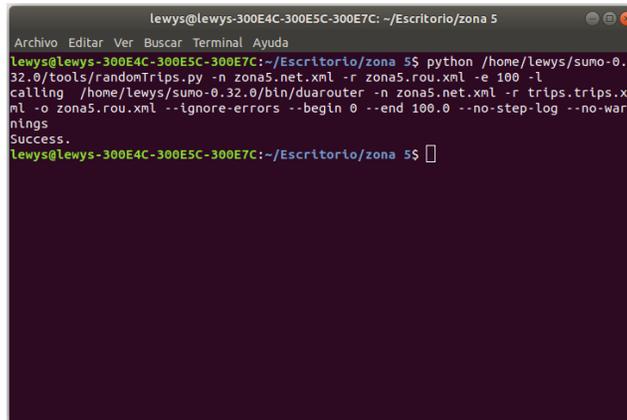
Importación de polígonos adicionales del mapa de la zona 5 por medio del programa *polyconvert* y creación del fichero *zona5.poly.xml*.



```
lewys@lewys-300E4C-300E5C-300E7C: ~/Escritorio/zona 5
Archivo Editar Ver Buscar Terminal Ayuda
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 5$ polyconvert --osm-files zona5.net.xml --type-file osmPolyconvert.typ.xml -o zona5.poly.xml
Warning: Environment variable SUMO_HOME is not set, schema resolution will use slow website lookups.
Warning: Environment variable SUMO_HOME is not set, schema resolution will use slow website lookups.
Warning: Environment variable SUMO_HOME is not set, schema resolution will use slow website lookups.
Warning: Environment variable SUMO_HOME is not set, schema resolution will use slow website lookups.
Success.
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 5$
```

Figura 64: Importación de polígonos adicionales del mapa de la zona 5
Fuente: Autor. Generado por medio de la consola de Ubuntu 18.04.2.

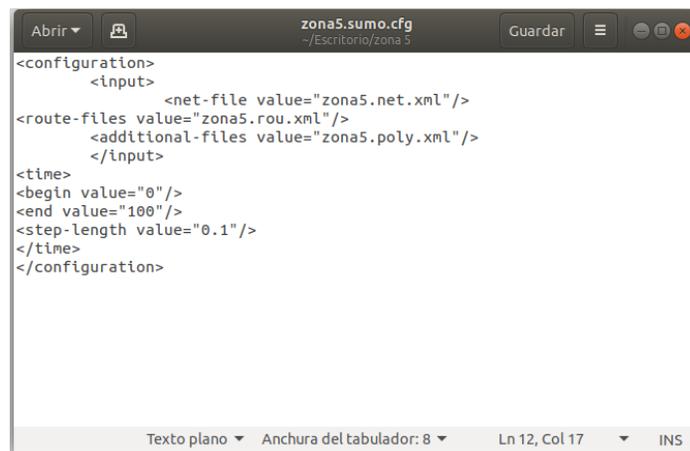
Ejecución del programa *randomTrips.py* para la simulación de la zona 5



```
lewys@lewys-300E4C-300E5C-300E7C: ~/Escritorio/zona 5
Archivo Editar Ver Buscar Terminal Ayuda
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 5$ python /home/lewys/sumo-0.
32.0/tools/randomTrips.py -n zona5.net.xml -r zona5.rou.xml -e 100 -l
calling /home/lewys/sumo-0.32.0/bin/duarouter -n zona5.net.xml -r trips.trips.x
ml -o zona5.rou.xml --ignore-errors --begin 0 --end 100.0 --no-step-log --no-war
nings
Success.
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 5$
```

Figura 65: Ejecución del programa *randomTrips.py* para la simulación de la zona 5
Fuente: Autor. Generado por medio de la consola de Ubuntu 18.04.2.

Creación del archivo de configuración *zona5.sumo.cfg* e importación de los archivos *zona5.net.xml* y *zona5.poly.xml*.



```
zona5.sumo.cfg
~/Escritorio/zona 5
Guardar
<configuration>
  <input>
    <net-file value="zona5.net.xml"/>
    <route-files value="zona5.rou.xml"/>
    <additional-files value="zona5.poly.xml"/>
  </input>
  <time>
    <begin value="0"/>
    <end value="100"/>
    <step-length value="0.1"/>
  </time>
</configuration>
Texto plano Anchura del tabulador: 8 Ln 12, Col 17 INS
```

Figura 66: Creación del fichero *zona5.sumo.cfg*
Fuente: Autor. Generado por medio de gedit de Ubuntu 18.04.2.

Ejecución del programa *sumo-gui* con parámetro *zona5.sumo.cfg* para observar el mapa convertido de la zona 5.

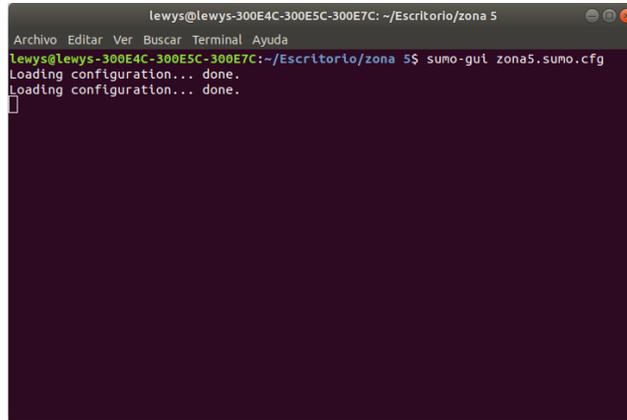


Figura 67: Ejecución del programa *sumo-gui* para la simulación de la zona 5
Fuente: Autor. Generado por medio de la consola de Ubuntu 18.04.2.

El mapa en ejecución de la zona 5 se puede observar en la Figura 68.

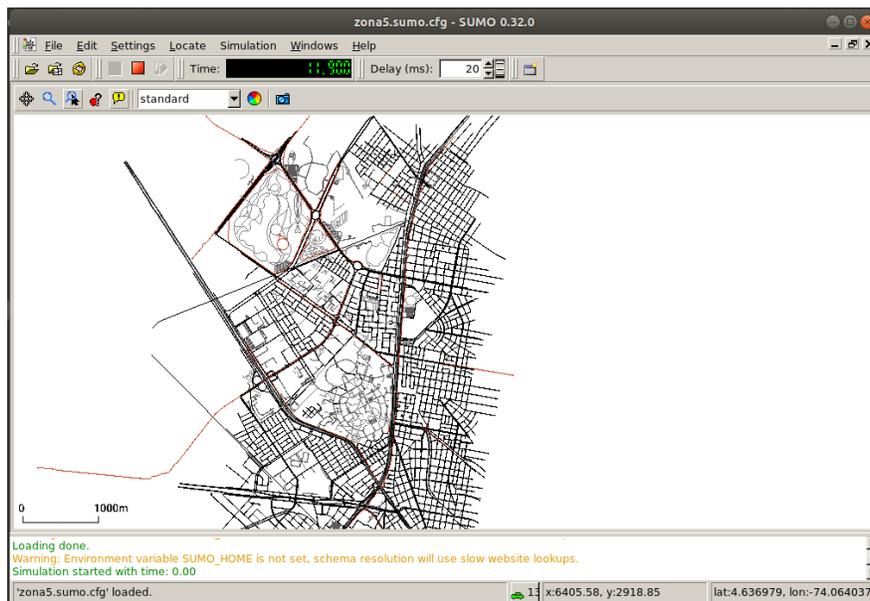
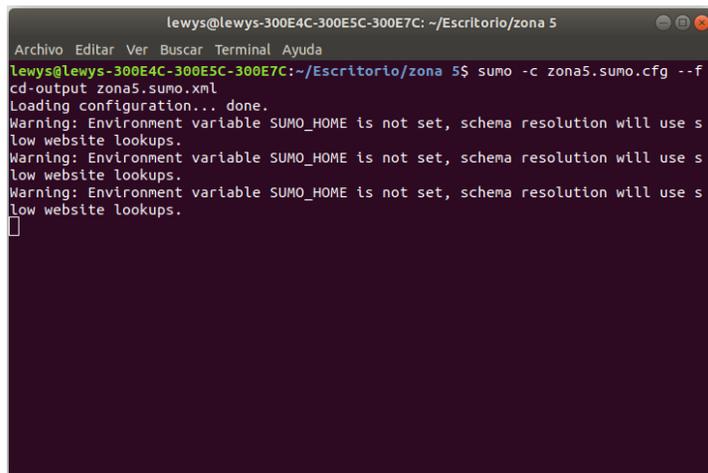


Figura 68: Mapa en ejecución de la zona 5
Fuente: Autor. Generado por medio de la herramienta *sumo-0.32.0*.

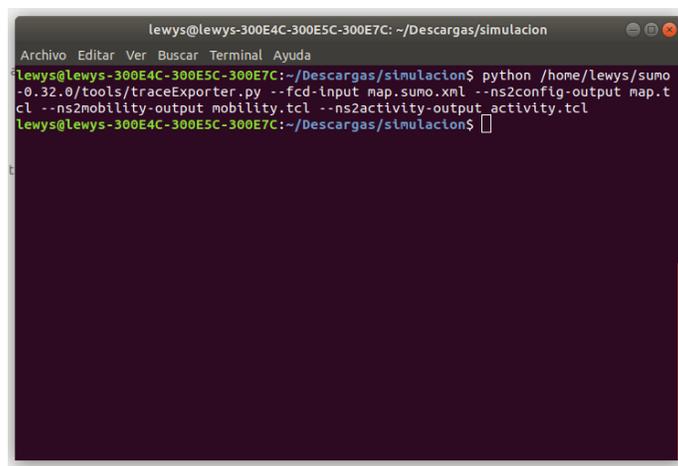
Creación del fichero *zona5.sumo.xml* con el programa `sumo -c`.



```
lewys@lewys-300E4C-300E5C-300E7C: ~/Escritorio/zona 5
Archivo Editar Ver Buscar Terminal Ayuda
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 5$ sumo -c zona5.sumo.cfg --f
cd-output zona5.sumo.xml
Loading configuration... done.
Warning: Environment variable SUMO_HOME is not set, schema resolution will use s
low website lookups.
Warning: Environment variable SUMO_HOME is not set, schema resolution will use s
low website lookups.
Warning: Environment variable SUMO_HOME is not set, schema resolution will use s
low website lookups.
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/zona 5$
```

Figura 69: Ejecución del programa `sumo -c` para la simulación de la zona 5
Fuente: Autor. Generado por medio de la consola de Ubuntu 18.04.2.

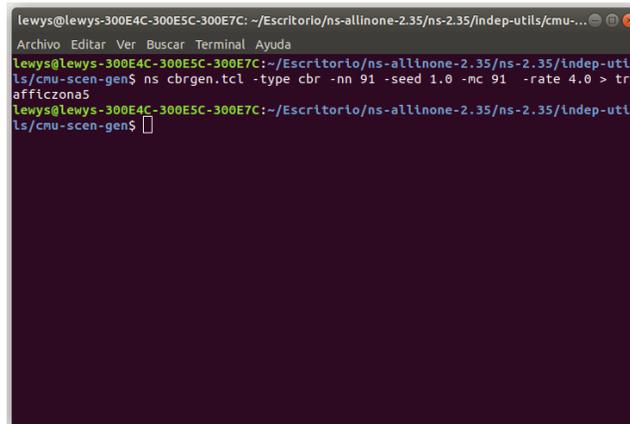
Creación del archivo básico de configuración de la red, del archivo de movilidad de la red y del archivo de actividad de la red para la simulación de la zona 5.



```
lewys@lewys-300E4C-300E5C-300E7C: ~/Descargas/simulacion
Archivo Editar Ver Buscar Terminal Ayuda
lewys@lewys-300E4C-300E5C-300E7C:~/Descargas/simulacion$ python /home/lewys/sumo
-0.32.0/tools/traceExporter.py --fcd-input map.sumo.xml --ns2config-output map.t
cl --ns2mobility-output mobility.tcl --ns2activity-output activity.tcl
lewys@lewys-300E4C-300E5C-300E7C:~/Descargas/simulacion$
```

Figura 70: Ejecución del programa `traceExporter.py` para la simulación de la zona 5
Fuente: Autor. Generado por medio de la consola de Ubuntu 18.04.2.

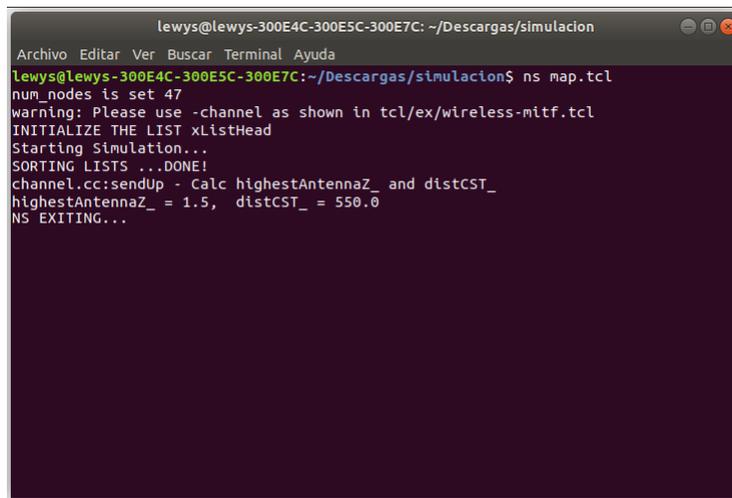
Ejecución del programa *cbrgen.tcl* para la generación de tráfico de la zona 5.



```
lewys@lewys-300E4C-300E5C-300E7C: ~/Escritorio/ns-allinone-2.35/ns-2.35/indep-utils/cmu-...
Archivo Editar Ver Buscar Terminal Ayuda
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/ns-allinone-2.35/ns-2.35/indep-uti
ls/cmu-scen-gen$ ns cbrgen.tcl -type cbr -nn 91 -seed 1.0 -mc 91 -rate 4.0 > tr
afficzona5
lewys@lewys-300E4C-300E5C-300E7C:~/Escritorio/ns-allinone-2.35/ns-2.35/indep-uti
ls/cmu-scen-gen$
```

Figura 71: Ejecución del programa *cbrgen.tcl* para la generación de tráfico de la zona 5
Fuente: Autor. Generado por medio de la consola de Ubuntu 18.04.2.

Ejecución de la simulación para la zona 5 creada el fichero *zona5.tcl*.

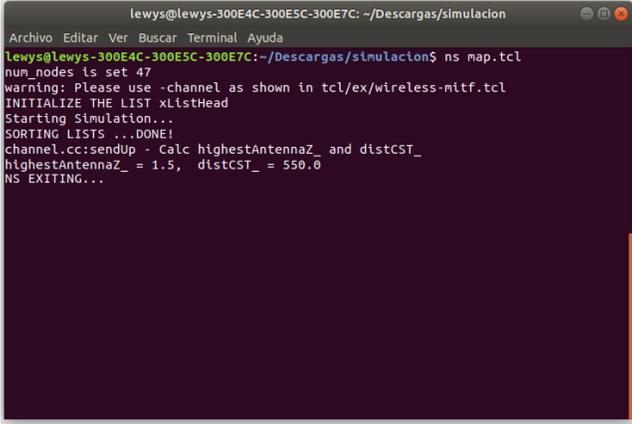


```
lewys@lewys-300E4C-300E5C-300E7C: ~/Descargas/simulacion
Archivo Editar Ver Buscar Terminal Ayuda
lewys@lewys-300E4C-300E5C-300E7C:~/Descargas/simulacion$ ns map.tcl
num_nodes is set 47
warning: Please use -channel as shown in tcl/ex/wireless-mitf.tcl
INITIALIZE THE LIST xListHead
Starting Simulation...
SORTING LISTS ...DONE!
channel.cc:sendup - Calc highestAntennaZ_ and distCST_
highestAntennaZ_ = 1.5, distCST_ = 550.0
NS EXITING...
```

Figura 72: Ejecución de la simulación para la zona 5 de una red Vanet
Fuente: Autor. Generado por medio de la consola de Ubuntu 18.04.2.

Simulación zona 5 de red Vanet interconectada con IoT

Ejecución de la simulación de una red Vanet con IoT para la zona 1.



```
lewys@lewys-300E4C-300E5C-300E7C: ~/Descargas/simulacion
Archivo Editar Ver Buscar Terminal Ayuda
lewys@lewys-300E4C-300E5C-300E7C:~/Descargas/simulacion$ ns map.tcl
num_nodes is set 47
warning: Please use -channel as shown in tcl/ex/wireless-nitf.tcl
INITIALIZE THE LIST xListHead
Starting Simulation...
SORTING LISTS ...DONE!
channel.cc:sendUp - Calc highestAntennaZ_ and distCST_
highestAntennaZ_ = 1.5, distCST_ = 550.0
NS EXITING...
```

Figura 73: Ejecución de la simulación para la zona 5 de una red Vanet interconectada con IoT
Fuente: Autor. Generado por medio de la consola de Ubuntu 18.04.2.

5. DISCUSIÓN DE RESULTADOS OBTENIDOS

En este capítulo se contrastó los resultados obtenidos en los diseños simulados vs resultados encontrados en los autores de referencia. Para el desarrollo de esta evaluación se utilizaron las siguientes métricas:

- Packet Delivery Ratio (PDF)
- Throughput
- End-End delay
- Packet Loss
- Packet Loss Ratio

Packet Delivery Ratio (PDF)

La PDF se evalúa como la proporción entre la cantidad de paquetes recibidos en el destino y la cantidad de paquetes transmitidos por la fuente. En otras palabras, la relación de entrega de paquetes es la relación entre el número de paquetes recibidos en el destino y el número de paquetes enviados desde la fuente. El rendimiento es mejor cuando la proporción de entrega de paquetes es alta. Para el cálculo de la PDF se utiliza la Ecuación 1.

$$PDF = \frac{\sum \text{Number of packect received}}{\sum \text{Number of packect send}} \quad (1)$$

En esta sección se observan los resultados matemáticos y gráficos de *PDF* para cada las simulaciones realizadas con una red Vanet Tradicional y una red Vanet interconectada con IoT. En la Tabla 13 se encuentran los resultados para una red Vanet Tradicional.

	Zona 1	Zona 2	Zona 3	Zona 4	Zona 5
Packet Delivery Ratio	1,288273458	1,467937463	1,471064111	1,307515887	1,963858411

Tabla 13: Pacject Delivery Ratio de las simulaciones realizadas con una red Vanet Tradicional

En la Tabla 14 se encuentran los resultados para una red Vanet interconecta con IoT.

	Zona 1	Zona 2	Zona 3	Zona 4	Zona 5
Packet Delivery Ratio	1,650071153	6,208576016	2,489575571	2,926701827	6,367711508

Tabla 14: Packet Delivery Ratio de las simulaciones realizadas con una red Vanet interconecta con IoT

Gráficamente los resultados de *Packet Delivery Ratio* por zonas se pueden observar en las Figuras: Figura 74, Figura 75, Figura 76, Figura 77 y Figura 78 donde se contrasta el resultado de la zona con red Vanet tradicional versus la zona con la red Vanet interconectada con IoT.

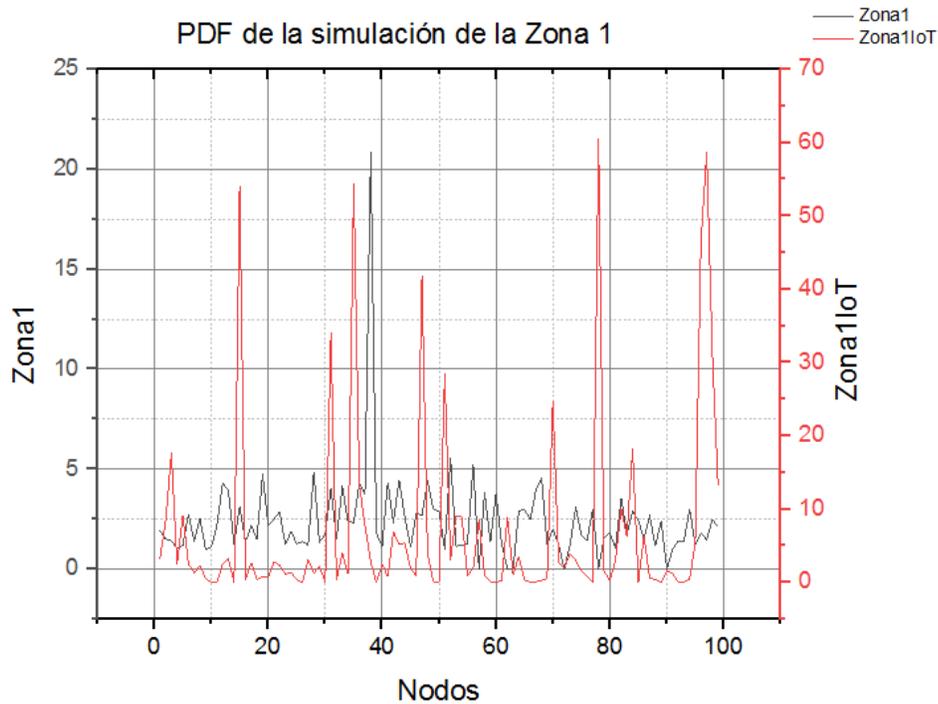


Figura 74: PDF de la simulación para la Zona 1
Fuente: Autor. Generado por medio del programa Origin 2020

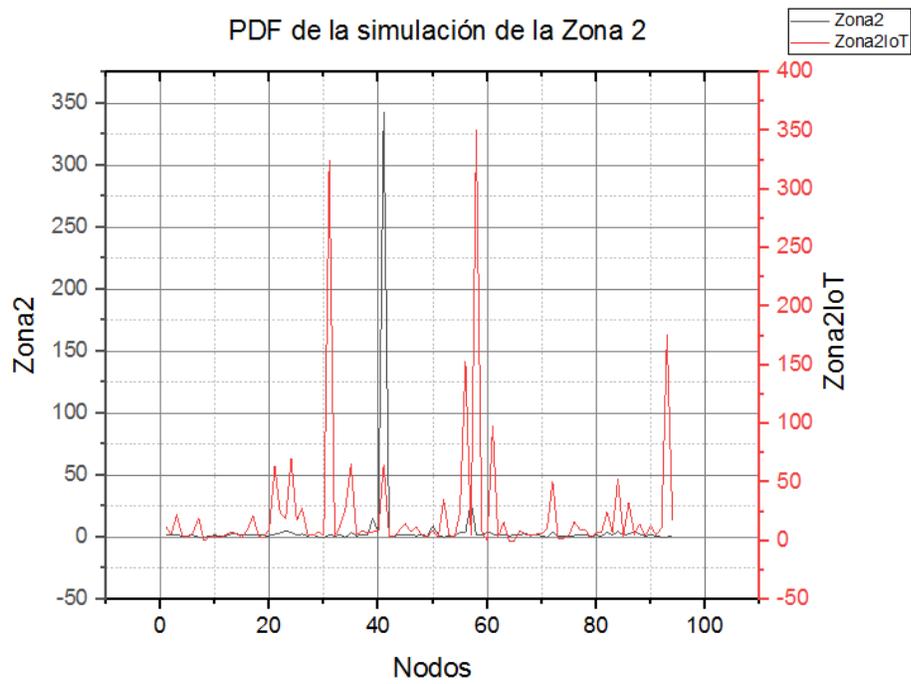


Figura 75: PDF de la simulación para la Zona 2
Fuente: Autor. Generado por medio del programa Origin 2020

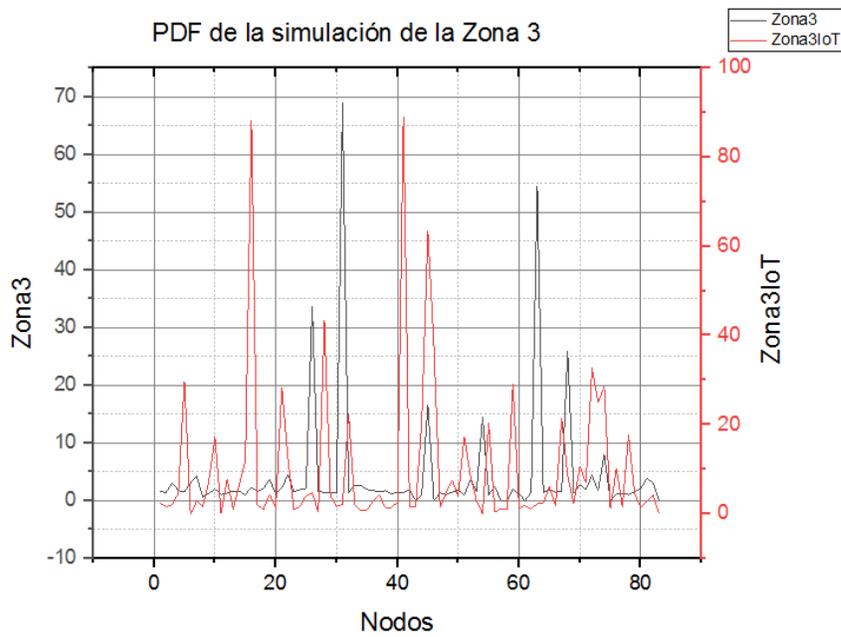


Figura 76: PDF de la simulación para la Zona 3
Fuente: Autor. Generado por medio del programa Origin 2020

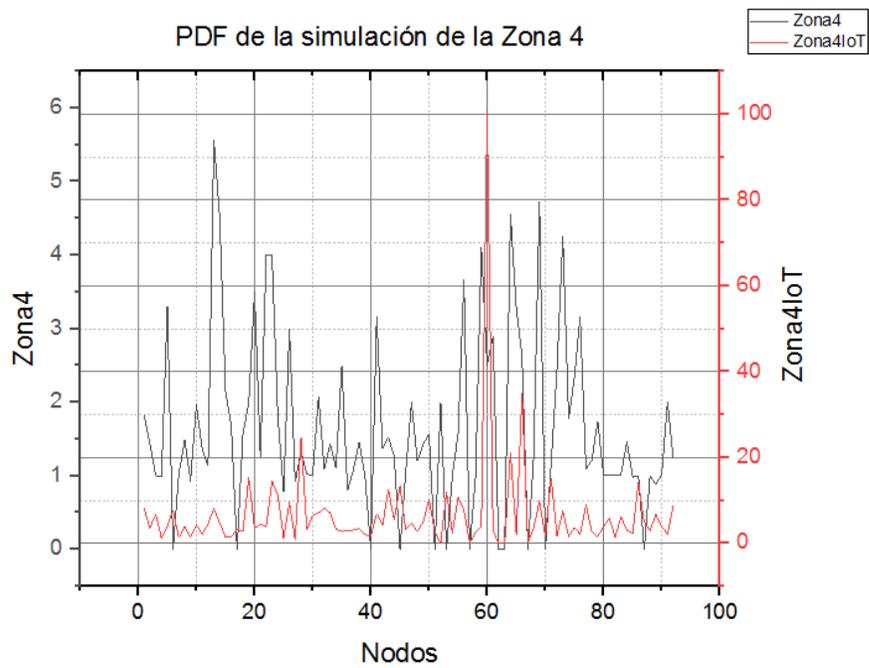


Figura 77: PDF de la simulación para la Zona 4
Fuente: Autor. Generado por medio del programa Origin 2020

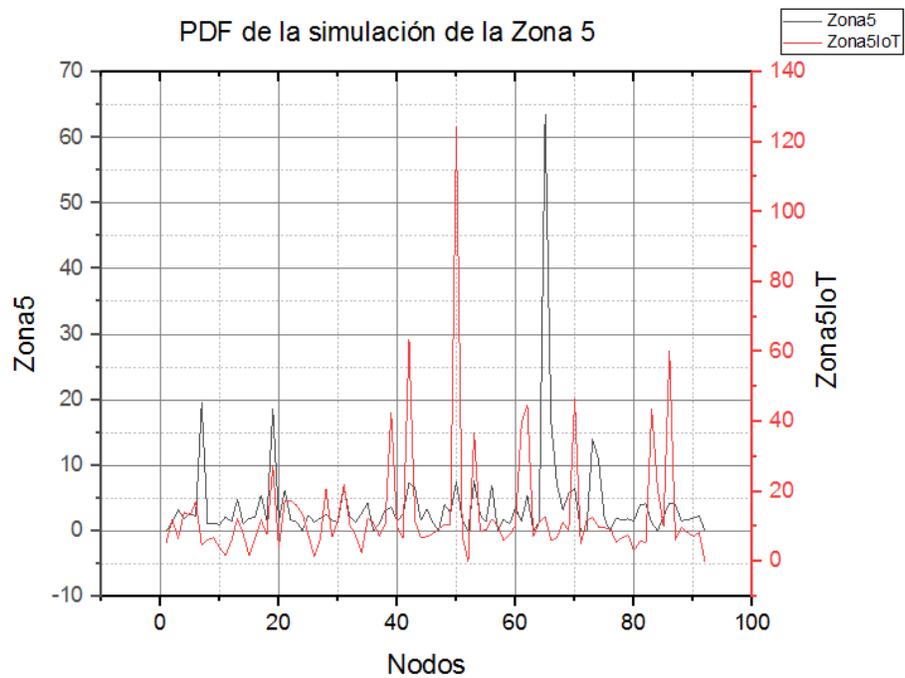


Figura 78: PDF de la simulación para la Zona 5
Fuente: Autor. Generado por medio del programa Origin 2020

De estos resultados se observa que hay mejor rendimiento en las simulaciones de red Vanet interconectada con IoT, ya que en todas las zonas se obtuvo un valor superior al presentado en la red Vanet tradicional.

Throughput

El rendimiento por nodo es el número de bits que puede un nodo transmitir a su destino. También, el conjunto del rendimiento por nodo sobre cada uno de los de nodos un sistema se conoce como el rendimiento del sistema. Para el cálculo del rendimiento se utiliza la Ecuación 2.

$$\text{Throughput} = \frac{\text{File Size}}{\text{Transmission Time (bps)}} \quad (2)$$

Gráficamente los resultados *Throughput de generación de bits* por zonas se pueden observar en las Figuras: Figura 79, Figura 80, Figura 81, Figura 82 y Figura 83 donde se contrasta el resultado de la zona con red Vanet tradicional versus la zona con la red Vanet interconectada con IoT.

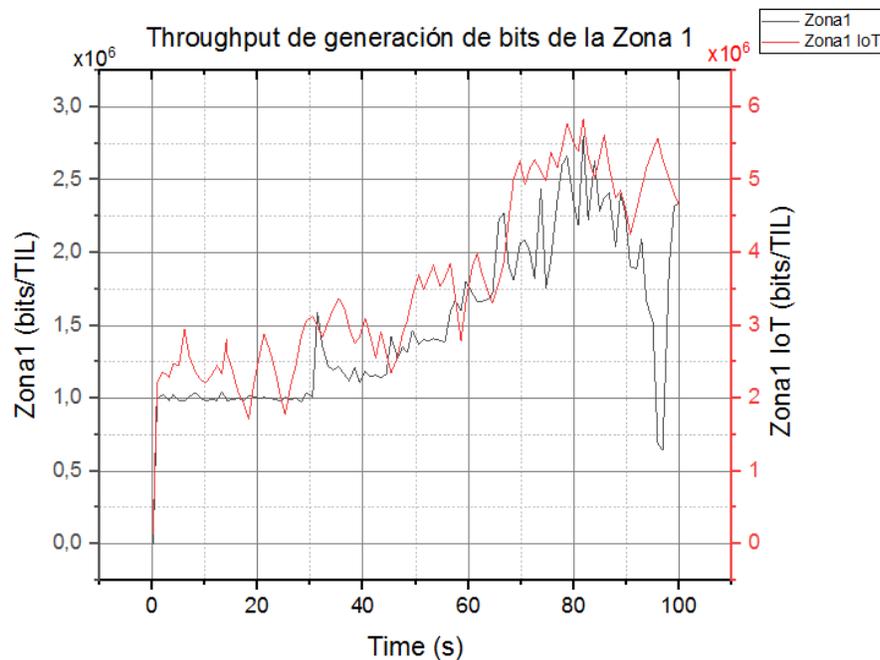


Figura 79: Throughput de generación de bits para la Zona 1
Fuente: Autor. Generado por medio del programa Origin 2020

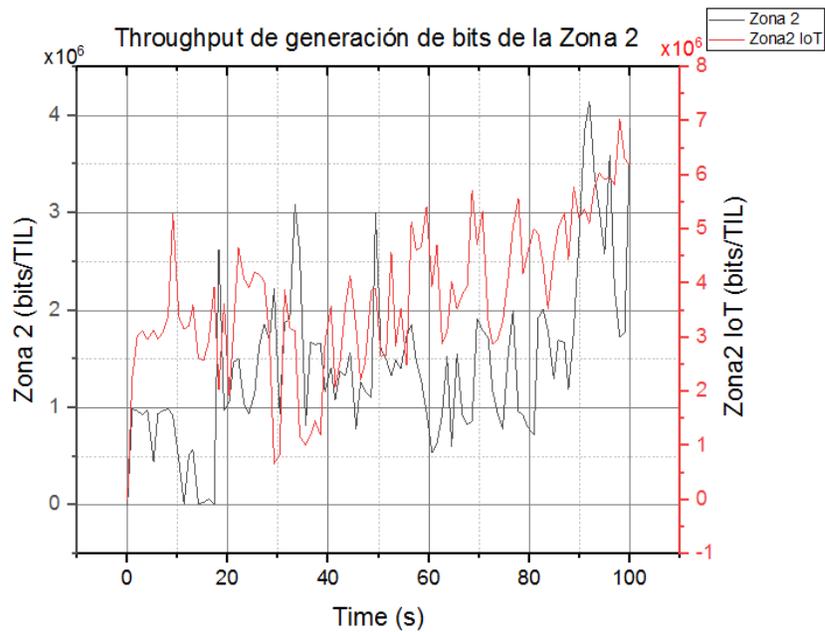


Figura 80: Throughput de generación de bits para la Zona 2
Fuente: Autor. Generado por medio del programa Origin 2020

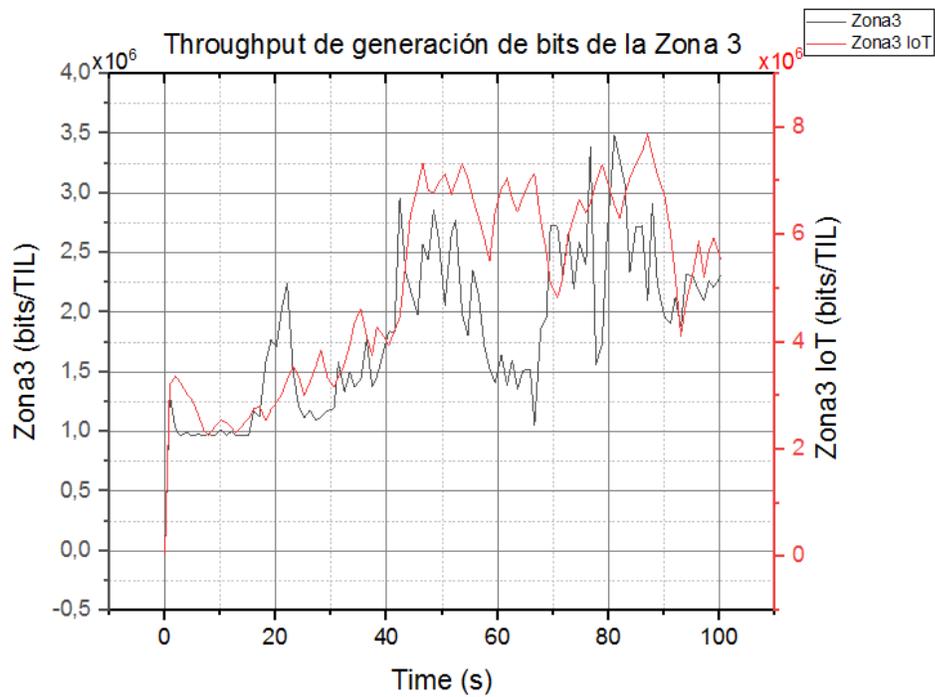


Figura 81: Throughput de generación de bits para la Zona 3
Fuente: Autor. Generado por medio del programa Origin 2020

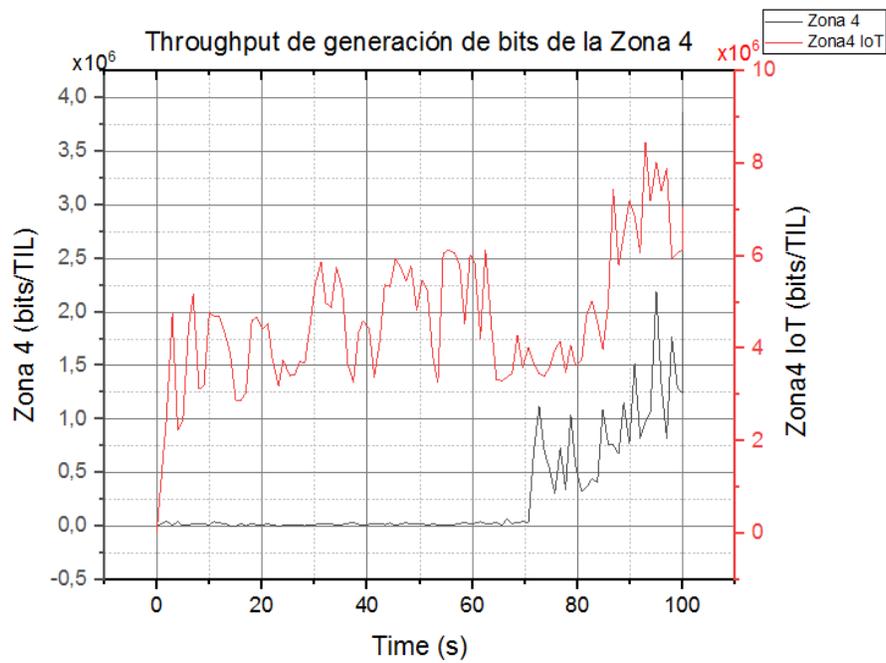


Figura 82: Throughput de generación de bits para la Zona 4
Fuente: Autor. Generado por medio del programa Origin 2020

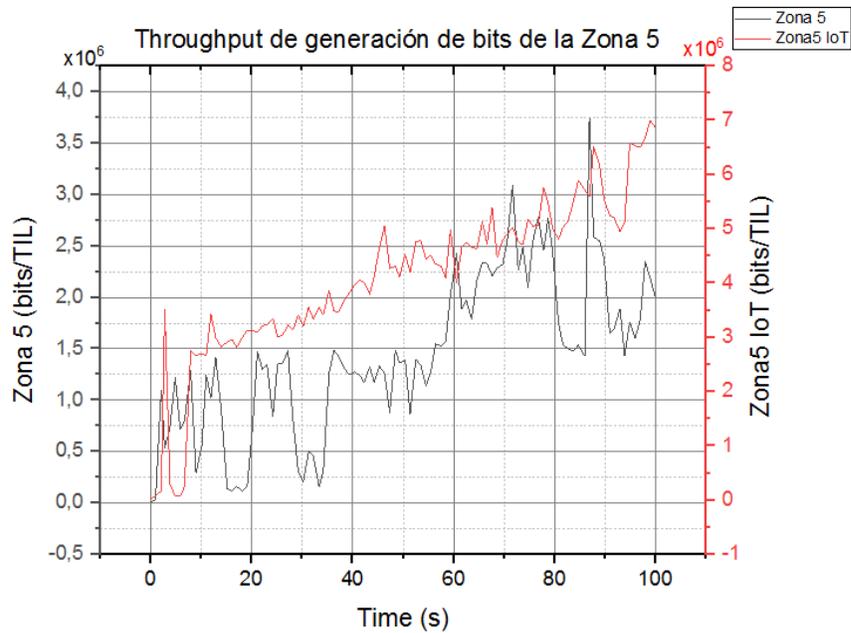


Figura 83: Throughput de generación de bits para la Zona 5
Fuente: Autor. Generado por medio del programa Origin 2020

Dentro de las simulaciones realizadas, el *throughput* se ve puede observar notablemente que las simulaciones realizadas de redes Vanet interconectadas con IoT presentan menor *overhead*.

End-End delay

El retraso de extremo a extremo es el tiempo ordinario que tarda un grupo de datos para ponerse en contacto en el objetivo. De la misma manera se une la demora provocada por el proceso de exposición del curso y la línea en la transmisión del paquete de datos. Conocido también como los paquetes de datos que pasaron de manera viable a los objetivos que se verificaron. La ecuación (3) se usa para determinarlo.

$$End - End\ delay = \frac{\sum(arrive\ time - send\ time)}{\sum Number\ of\ connections} \quad (3)$$

En esta sección se observan los resultados matemáticos y gráficos de *End-End delay* para las simulaciones realizadas con una red Vanet Tradicional y una red Vanet interconectada con IoT. En la Tabla 15 se encuentran los resultados para una red Vanet Tradicional.

	Zona 1	Zona 2	Zona 3	Zona 4	Zona 5
End-End delay	0.262004927	0.9034417387	0.4134866792	1.350530672	0.3531549645

Tabla 15: Promedio End-End delay de las simulaciones realizadas con una red Vanet tradicional

En la Tabla 16 se encuentran los resultados para una red Vanet interconecta con IoT.

	Zona 1	Zona 2	Zona 3	Zona 4	Zona 5
End-End delay	0.1247327895	0.1848232696	0.3891343781	0.3419632678	0.2294264466

Tabla 16: Promedio End-End delay de las simulaciones realizadas con una red Vanet interconecta con IoT

De estos resultados se observa que hay menores retardos en las simulaciones de red Vanet interconectada con IoT, ya que en todas las zonas se obtuvo un valor inferior al presentado en la red Vanet tradicional.

Packet Loss

La pérdida de paquetes es la relación entre la cantidad de paquetes que nunca llegaron al destino y la cantidad de paquetes originados por la fuente Para el cálculo de la pérdida de paquetes se utiliza la Ecuación 4.

$$PL = \frac{\text{Number of packets sent} - \text{Number of packets received}}{\text{Number of packets send}} \quad (4)$$

En esta sección se observan los resultados matemáticos y gráficos de *Packet Loss* para cada las simulaciones realizadas con una red Vanet Tradicional y una red Vanet interconectada con IoT. En la Tabla 17 se encuentran los resultados para una red Vanet Tradicional.

	Zona 1	Zona 2	Zona 3	Zona 4	Zona 5
Packet Loss	0,393965528	0,838932471	0,598325107	0,658318456	0,84295771

Tabla 17: Packet Loss de las simulaciones realizadas con una red Vanet tradicional

En la Tabla 18 se encuentran los resultados para una red Vanet interconecta con IoT.

	Zona 1	Zona 2	Zona 3	Zona 4	Zona 5
Packet Loss	0,223767288	0,31877207	0,320219974	0,235190937	0,490798321

Tabla 18: Packet Loss de las simulaciones realizadas con una red Vanet interconecta con IoT

Gráficamente los resultados *Packet Loss* por zonas se pueden observar en las Figuras: Figura 84, Figura 85, Figura 86, Figura 87 y Figura 88 donde se contrasta el resultado de la zona con red Vanet tradicional versus la zona con la red Vanet interconectada con IoT.

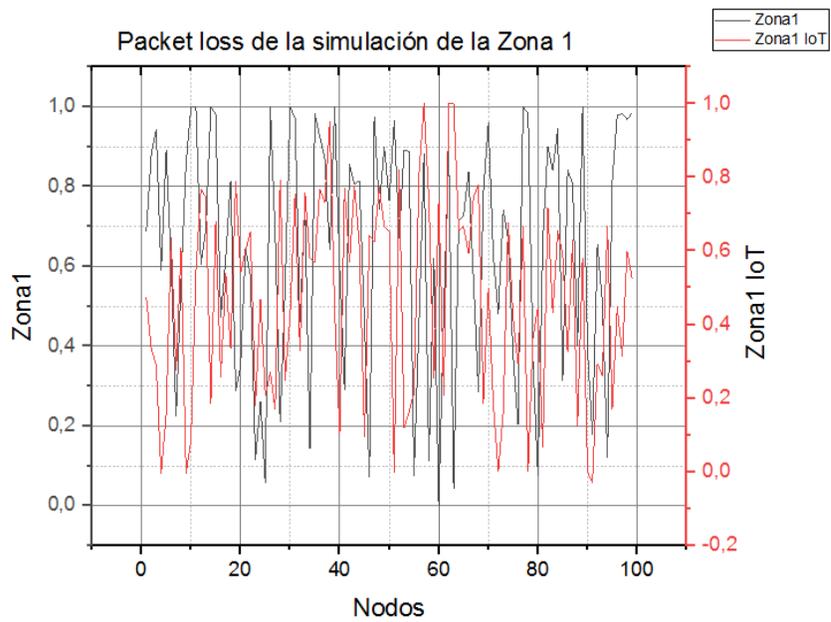


Figura 84: Packet Loss de la simulación para la Zona 1
Fuente: Autor. Generado por medio del programa Origin 2020

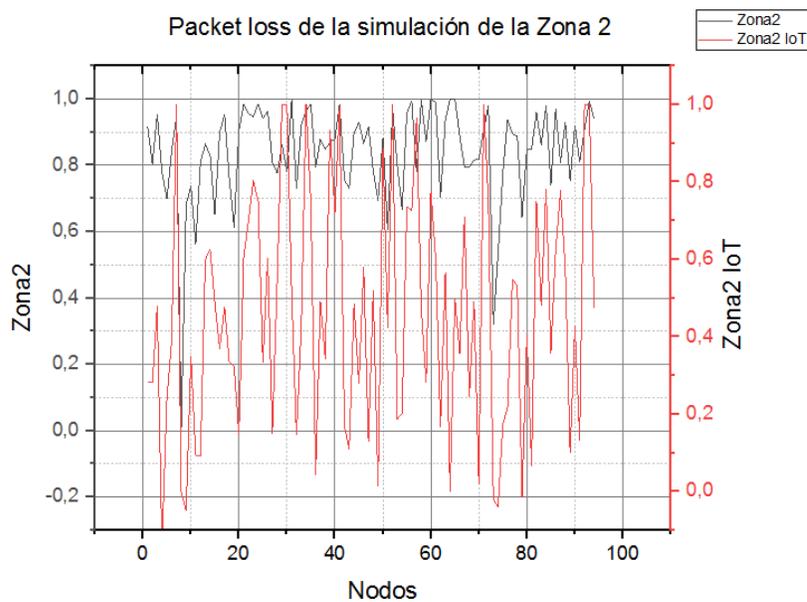


Figura 85: Packet Loss de la simulación para la Zona 2
Fuente: Autor. Generado por medio del programa Origin 2020

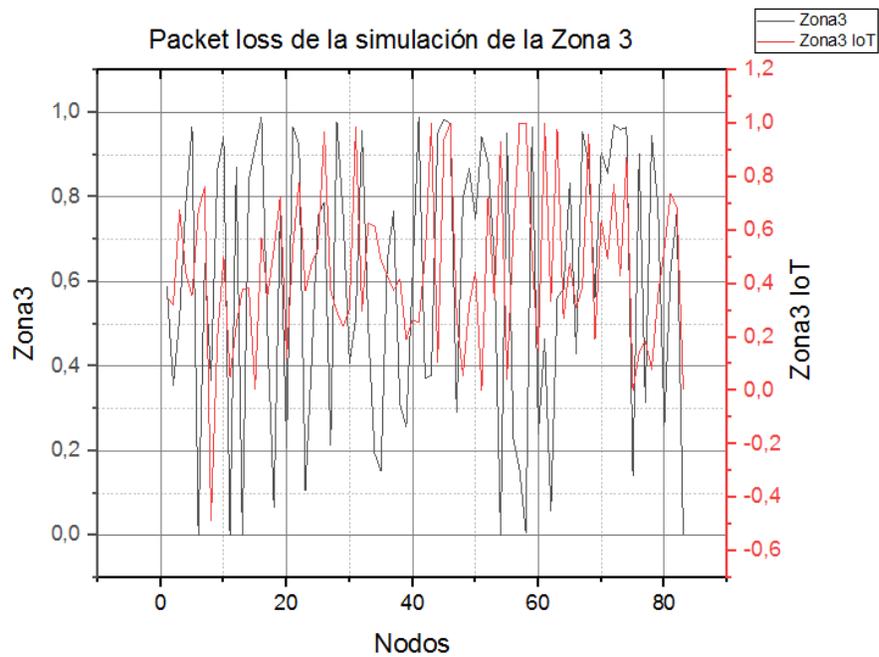


Figura 86: Packet Loss de la simulación para la Zona 3
Fuente: Autor. Generado por medio del programa Origin 2020

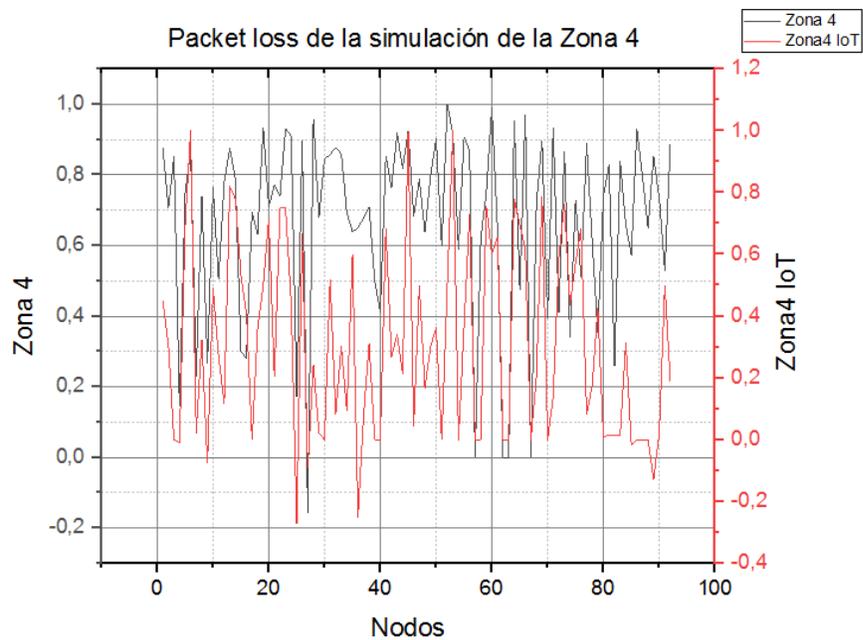


Figura 87: Packet Loss de la simulación para la Zona 4
Fuente: Autor. Generado por medio del programa Origin 2020

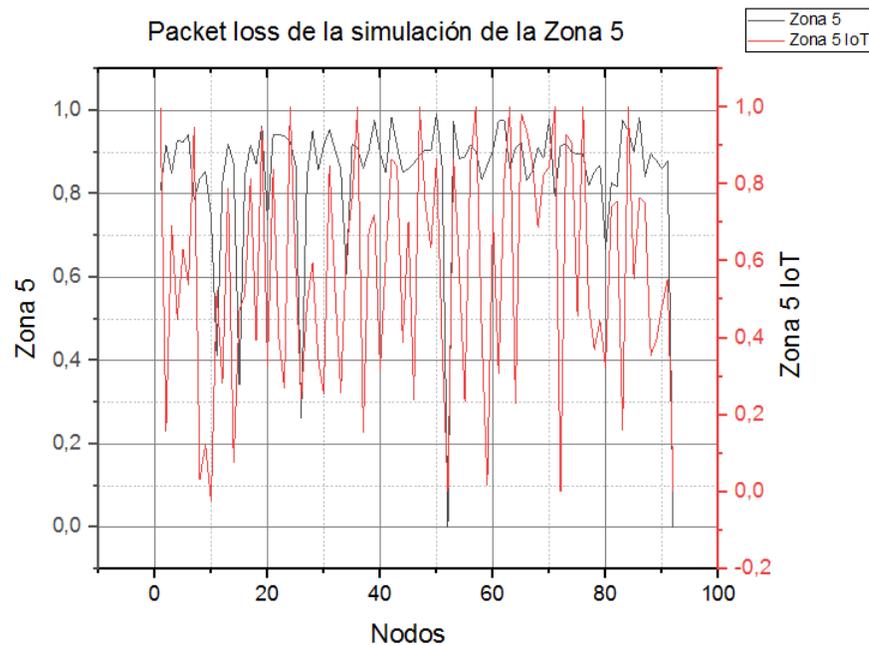


Figura 88: Packet Loss de la simulación para la Zona 5
Fuente: Autor. Generado por medio del programa Origin 2020

De estos resultados se observa que hay menor pérdida de paquetes en las simulaciones de red Vanet interconectada con IoT, ya que en todas las zonas se obtuvo un valor inferior al presentado en la red Vanet tradicional.

Packet Loss Ratio

La tasa de pérdida de paquetes es la relación entre el número de paquetes que nunca llegaron al destino y el número de paquetes originados por la fuente. Para el cálculo de la pérdida de paquetes se utiliza la Ecuación 5.

$$PLR = \frac{\text{Number of packets sent} - \text{Number of packets received}}{\text{Number of packets send}} \times 100 \quad (5)$$

En esta sección se observan los resultados matemáticos y gráficos de *Packet Loss Ratio* cada las simulaciones realizadas con una red Vanet Tradicional y una red Vanet interconectada con IoT. En la Tabla 19 se encuentran los resultados para una red Vanet Tradicional.

	Zona 1	Zona 2	Zona 3	Zona 4	Zona 5
Packet Loss Ratio	39,3965528	83,89324706	59,83251074	65,83184556	84,29577096

Tabla 19: Packet Loss Ratio de las simulaciones realizadas con una red Vanet tradicional

En la Tabla 20 se encuentran los resultados para una red Vanet interconecta con IoT.

	Zona 1	Zona 2	Zona 3	Zona 4	Zona 5
Packet Loss Ratio	22,3767288	31,87720697	32,02199736	23,51909373	49,07983211

Tabla 20: Packet Loss Ratio de las simulaciones realizadas con una red Vanet interconecta con IoT

Gráficamente los resultados *Packet Loss Ratio* por zonas se pueden observar en las Figuras: Figura 89, Figura 90, Figura 91, Figura 92 y Figura 93 donde se contrasta el resultado de la zona con red Vanet tradicional versus la zona con la red Vanet interconectada con IoT.

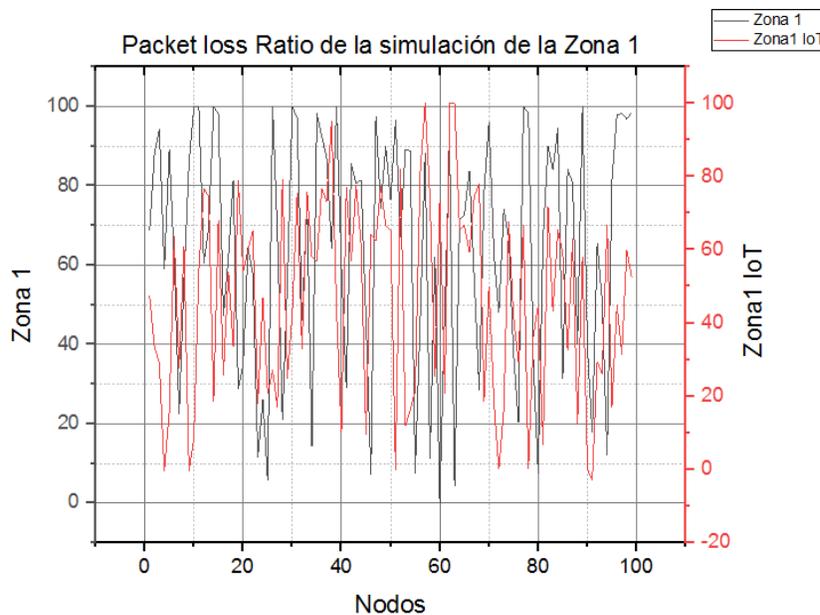


Figura 89: Packet Loss Ratio de la simulación para la Zona 1
Fuente: Autor. Generado por medio del programa Origin 2020

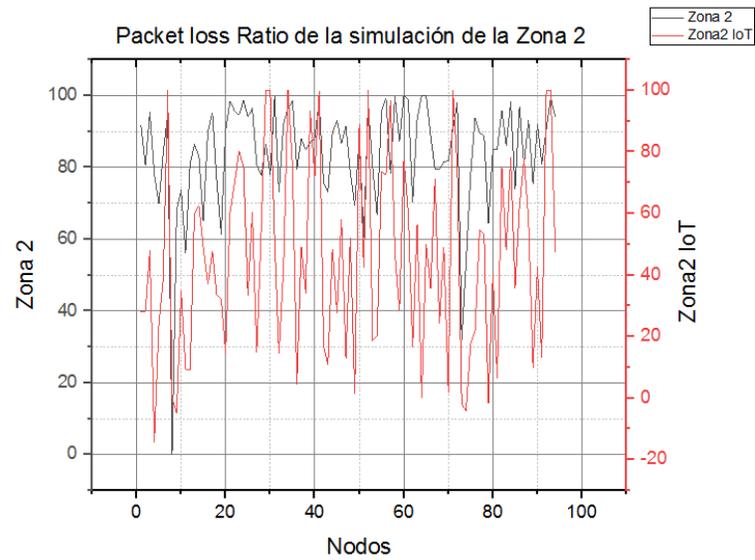


Figura 90: Packet Loss Ratio de la simulación para la Zona 2
Fuente: Autor. Generado por medio del programa Origin 2020

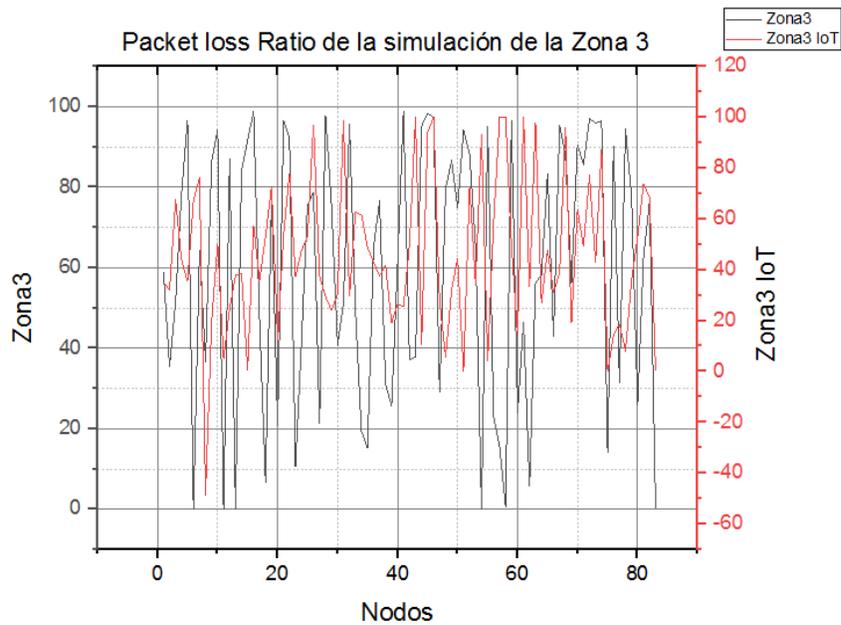


Figura 91: Packet Loss Ratio de la simulación para la Zona 3
Fuente: Autor. Generado por medio del programa Origin 2020

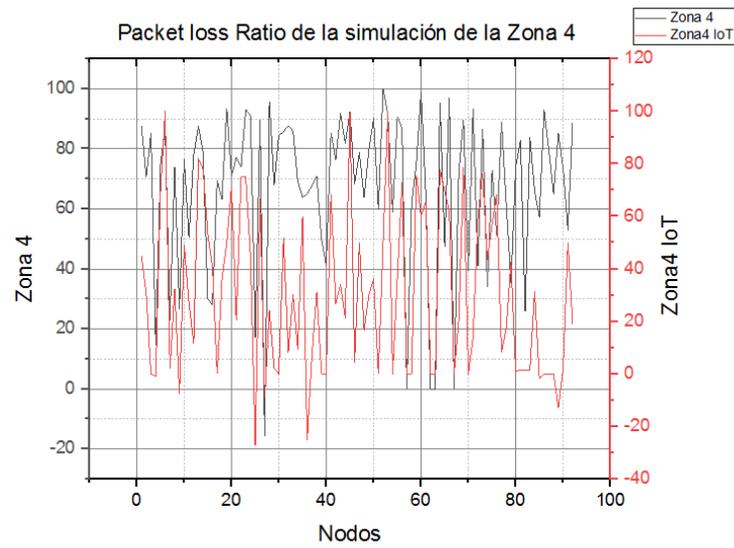


Figura 92: Packet Loss Ratio de la simulación para la Zona 4
Fuente: Autor. Generado por medio del programa Origin 2020

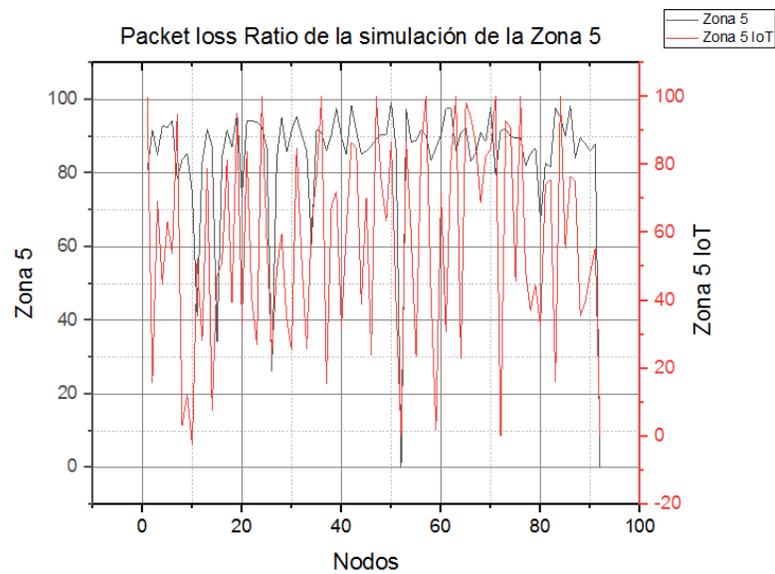


Figura 93: Packet Loss Ratio de la simulación para la Zona 5
Fuente: Autor. Generado por medio del programa Origin 2020

De estos resultados se observa que hay menor tasa de pérdida de paquetes en las simulaciones de red Vanet interconectada con IoT, ya que en todas las zonas se obtuvo un valor inferior al presentado en la red Vanet tradicional.

5.1 CONTRASTE CON TRABAJOS RELACIONADOS

Teniendo en cuenta los resultados expuestos previamente se procede a compararlos con trabajos anteriormente realizados por otros autores, con el fin de determinar la eficiencia obtenida del modelo implementado en este trabajo de investigación:

En (Tian et al., 2016), los investigadores propusieron un mecanismo de reenvío de datos basado en redes sociales (SDFM) para la comunicación V2V en Vanet. El SDFM aprende las características sociales de los vehículos de manera distribuida, y luego transfiere los mensajes en un patrón de "Guardar-Llevar-Enviar". Los resultados experimentales de este trabajo arrojan que, para los vehículos cívicos en movimiento, el mecanismo SDFM ha mejorado significativamente la relación de entrega de mensajes y el rendimiento promedio de retraso de entrega en comparación con un protocolo de enrutamiento basado en redes sociales diseñado para el reenvío de datos en redes de conmutación de paquetes.

En la Figura 94, se puede observar que el protocolo SDFM supera a Bubble Rap en todos los casos. Especialmente cuando aumenta la escala de la red, la brecha entre las dos curvas se vuelve aún mayor.

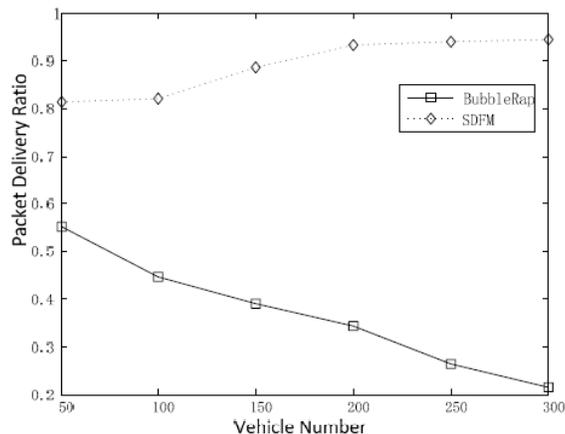


Figura 94: The packet delivery ratio versus el número de vehículos en la red
Fuente: (Tian et al., 2016)

Y al hacer la comparación con el modelo propuesto en el presente trabajo de investigación, se puede observar que el modelo propuesto de una red Vanet interconectada con IoT supera en todas las zonas a los resultados del trabajo anterior.

En la literatura científica también se puede encontrar una serie de trabajo donde se ahonda en la necesidad de mejorar las redes Vanet con el uso otras tecnologías, a continuación, una selección de cinco trabajos donde se realizan mejoras a redes Vanet:

En (Bui & Jung, 2018) afirman que, con el crecimiento y el desarrollo de Internet de las cosas, las tecnologías de vehículos conectados se convierten en una nueva tendencia en tecnología y que la sociedad que espera que redefina los sistemas de transporte. Y que por eso es válido estudiar cómo los vehículos conectados impactan en la planificación del transporte convirtiéndose en un tema de suma importancia. En su trabajo los investigadores consideraron un escenario en el que los vehículos conectados pasan la intersección sin el sistema de semáforos. Ellos propusieron un nuevo marco de vehículos conectados en el que se pueden comunicar y colaborar entre sí para mejorar el flujo de tráfico en la intersección sin ningún controlador a través de enlaces inalámbricos.

En (Hussain et al., 2019) afirman que las soluciones de seguridad eficientes, viables, robustas y adaptativas allanarán el camino para las aplicaciones de redes de vehículos comerciales. Y que la academia ha producido resultados prometedores de investigación tanto a través de publicaciones como de la industria a través de experimentos prácticos. Pero que la demanda de nuevas y emocionantes aplicaciones en tiempo real a través de redes vehiculares y la integración con otras tecnologías habilitadoras como la computación en la nube e IoT necesitarán un paradigma de comunicación que cumpla con los requisitos de las nuevas aplicaciones. En su trabajo los investigadores estudiaron la seguridad de Vanet centrándose específicamente en los requisitos, las soluciones y los estándares actuales, y señalaron las deficiencias existentes en las soluciones de seguridad de Vanet. Una de las conclusiones de su trabajo fue que fusionar Vanet con otras tecnologías habilitadoras como IoT, computación en la nube y redes sociales es esencial para admitir los nuevos servicios en Vanet.

Los investigadores en (Kombate & Wanglina, 2017) confirmaron que el reciente desarrollo en el gran dominio de Internet de las Cosas está impulsando un cambio convencional en

las Redes Adhoc de Vehículos, ahora conocidas como Internet de Vehículos (IOV). El mercado de vehículos conectados será extremadamente grande en términos de número de puntos finales conectados como también el intercambio de datos. Con el tiempo, cualquier vehículo tendrá el potencial de conectarse a cualquier cosa en cualquier momento, desde personas hasta cosas físicas, procesos, contenidos, teniendo acceso a información pertinente y oportuna y a productos de todo tipo de maneras completamente flexibles, confiables y seguras. Ellos es su investigación han descrito los beneficios de las comunicaciones 5G dentro del nuevo concepto de Internet de vehículos en términos de baja latencia, ancho de banda extremadamente alto y confiabilidad.

En (Sahbi et al., 2019) afirman que Internet de vehículos (IoV) es la evolución de la red de vehículos convencionales (Vanet) y que esta es una integración de tres redes: una red entre vehículos, una red dentro del vehículo e Internet móvil vehicular, en la cual el vehículo se considera como un objeto inteligente equipado con una potente plataforma de sensores múltiples, tecnologías de conectividad y comunicación, lo que le permite para comunicarse con el mundo. Y que para esta integración o paso existen una serie de retos tales como la comunicación cooperativa entre vehículos y otros dispositivos en términos de: capacidad de almacenamiento y computación, energía del vehículo y control y gestión de la red. Los investigadores propusieron un modelo de red para IoV basado en el software Defined Network y Cloud Computing.

En un quinto y último trabajo, los investigadores (Sharma & Kaushik, 2019) sostienen que Internet of Vehicles (IoV) es un concepto emergente en los sistemas de transporte inteligente (ITS) para mejorar las capacidades existentes de Vanet al integrarse con Internet of Things (IoT). También afirma que IoV dominará los sistemas de transporte debido a numerosos rasgos especiales como estructuras topológicas dinámicas, gran escala de red, conexión a internet confiable, compatibilidad con dispositivos personales y alta capacidad de procesamiento, etc. Internet of Vehicles (IoV) se ha convertido en la tecnología clave que permite realizar futuros escenarios de conducción autónoma. Y que ha ganado un gran interés comercial y valor de investigación debido al rápido desarrollo de tecnologías de computación y comunicación como *Edge Computing*, *Grid Computing*, *Procesamiento paralelo*, *Big Data Analysis*, *Web Semantics* e *Artificial Intelligence*.

Como se puede observar en los trabajos anteriormente expuestos, todas las investigaciones conllevan a que hay que realizar un salto en las redes Vanet actuales, evolucionando a *Internet of Vehicles* o a redes Vanet de próxima generación. Las anteriores investigaciones aportan diferentes enfoques, pero validan la importancia de la investigación realizada a la hora de interconectar redes Vanet con IoT.

6. CONCLUSIONES, TRABAJOS FUTUROS Y RECOMENDACIONES

6.1 CONCLUSIONES

Del anterior trabajo se puede concluir que el desarrollo de la investigación aplicada fue interesante y desafiante. En primer lugar, la investigación en este campo es una tendencia a nivel mundial especialmente en Estados Unidos, Canadá, Europa y Asia. Esta tendencia es muy promisoría ya que diferentes compañías fabricantes de automóviles, operadores y fabricantes de dispositivos de telecomunicaciones y el sector académico han puesto los ojos en este desafiante tema, así como también entidades gubernamentales se encuentran a las expectativas de la evolución de esta tecnología. Las redes Vanet interconectadas con IoT, no se limita a hacer la vida de las personas más cómoda, sino que también preservar el medio ambiente, salvar vidas y hacer un uso más eficiente de los recursos naturales, estas tan solo algunas bondades que ofrecen las redes Vanet. Desafortunadamente en el panorama Latinoamericano la región no se nota el interés, esto se ve reflejado en la no existencia de grupos de investigación dedicados a las redes Vanet y mucho menos en los avances actuales y por ende son pocos aportes, la literatura en español y portugués es mínima, la realización de eventos para presentar los adelantos es inexistente como si sucede en otras regiones como América del Norte, Europa y Asia donde cada año realizan eventos puntualmente dedicados a éste tema, como también la existencia de revistas especializadas.

La complejidad y diversidad de dispositivos y requerimientos que se deben atender en el paradigma de Internet de las cosas dan como resultado arquitecturas que no son fáciles de concebir y manejar. En este escenario, es incuestionable el papel de las arquitecturas de referencia como mecanismo para guiar la construcción de los sistemas IoT y minimizar la falta de estandarización existente al desarrollar estas soluciones. Internet de las cosas es un paradigma que adolece de un uso inconsistente y de una unificación y acuerdo de sus principales elementos. Por lo tanto, el modelo de arquitectura propuesto es importante porque ofrece una comprensión común del dominio IoT y un fundamento sólido sobre el cual se construyó el modelo de arquitectura.

El modelo de arquitectura propuesto es una base para comparar y evaluar soluciones Vanet. Facilita el proceso de selección y diseño de soluciones Vanet. El diseño en capas de la arquitectura brinda la posibilidad de identificar claramente los componentes que debe contener la red y como cada uno de estos debe interactuar con los demás elementos.

Es claro que esta es apenas la primera versión de un modelo como este debido a que Vanet en su propia arquitectura no es compatible con nuevas tecnologías, pero se hace necesario darles una evolución o un salto a las redes Vanet integrándola con nuevas tecnologías.

Otro factor que hace necesaria la generación de este tipo de modelos aparte de la generación de Vanet es la evolución de la internet, a medida que pasan los años surgen nuevos servicios, nuevas transmisiones y nuevas integraciones con otras tecnologías las cuales aumentan el número de nodos, esto hace que constantemente se vea la necesidad de creaciones de nuevos y más efectivos modelos.

En el trabajo de investigación se hizo uso de las métricas: Packet Delivery Ratio (PDF), Throughput, End-End delay, Packet Loss y Packet Loss Ratio para hacer la evaluación y comparación del modelo planteado de una red Vanet Interconectada con IoT versus una red Vanet tradicional. Con estas métricas se pudo observar que con respecto al Packet Delivery Ratio la zona donde la diferencia fue menor fue la Zona 1 con un valor 0,361797695 y la zona donde se presentó una mayor diferencia fue la Zona 2 con un valor 4,740638553. En ambas Zonas el Packet Delivery Ratio fue mayor para las redes Vanet interconectadas con IoT.

Con el retraso End-End delay, la Zona 5 fue la zona que presentó una menor diferencia (0,123728518) y la zona 4 fue la que presentó una mayor diferencia. En ambas zonas, la red Vanet interconectada con IoT presento menor retraso respecto a la red Vanet tradicional. Y con la métrica Packet Loss Ratio, la zona que presento una menor perdida de paquetes fue la zona 1 con un valor de 17,0198% y la zona que presento una mayor variación fue la zona 2 con un valor de 52,016%. En ambas zonas la perdida de paquetes fue menor en la red Vanet interconecta con IoT. Por eso se puede afirmar que el desarrollo de integración de protocolos IoT con redes Vanet permitió un mejor rendimiento en la transmisión de mensajes desde y hacia una red de sensores y actuadores, con limitaciones evidentes en cuanto al consumo, velocidad de transmisión y procesamiento.

La conclusión en términos generales de todo el proyecto corresponde a lo satisfactorio de haber desarrollado el presente trabajo ya que fue muy enriquecedor además el tema es de actualidad y con mucho futuro, también permitió conocer gente de otros países que comparten esta línea de conocimiento a quienes han hecho de una u otra forma sus aportes.

6.2 TRABAJOS FUTUROS

- Evaluar el modelo de arquitectura propuesto dentro de una situación de un proyecto concreto.
- Mejoras del modelo para que pueda trabajar con diferentes protocolos IoT de acuerdo a las necesidades de los distintos escenarios.
- Exploración de nuevas tecnologías tales Redes 5G, SDN, Cloud Computing, Inteligencia artificial y Machine Learning.

6.3 APORTES DE LA INVESTIGACIÓN

- Se realizó una investigación del estado del arte el estado actual de las redes Vanet y los factores que influyen para un salto de generación o evolución de las redes Vanet.
- Un modelo de arquitectura para la interconexión de redes Vanet con IoT que muestra mejorar respecto a las redes Vanet tradicionales.

6.4 RECOMENDACIONES

- Para simulaciones de redes *ad hoc* como es el caso de las redes Vanet se recomienda el uso del simulador NS2, ya que este simulador permite la configuración de una gran cantidad de parámetros tales como la topología de la red, la pila de protocolos y parámetros específicos de cada protocolo, para evaluar el impacto de diferentes tipos de tráfico de red.

- Para el análisis de las trazas de resultados arrojados por el simulador NS2, se recomiendan el uso de los lenguajes de programación Perl y AWK porque estos dos lenguajes extraen la información de archivos de texto útiles, dependiendo el fin, a partir del contenido de los ficheros.
- El software de gráficos para trazar de archivos de seguimiento de NS2: *Tracegraph*, es de gran ayuda para analizar los resultados. Este software es recomendable usarlo en sistemas operativos GNU/Linux con sistema de gestión de paquetes dpkg, ya que, a la hora de configurar el software, se requieren librerías antiguas y que son fáciles de encontrar en sitios web de soporte para el sistema operativo Debian. En este trabajo se configuró en el sistema operativo Ubuntu pero es compatible con Debian ya que es un derivado de esto.
- La distribución del sistema operativo GNU/Linux Ubuntu, es un sistema operativo donde la mayoría de los elementos de una computadora como procesador, memoria RAM, discos duros, tarjeta gráfica y demás elementos se aprovechan mejor debido a que el consumo de recursos es menor en este sistema operativo y la velocidad de procesamiento aumenta.

7. REFERENCIAS

- Al-Sultan, S., Al-Doori, M. M., Al-Bayatti, A. H., & Zedan, H. (2014). A comprehensive survey on vehicular Ad Hoc network. *Journal of Network and Computer Applications*, 37, 380–392. <https://doi.org/10.1016/j.jnca.2013.02.036>
- Alcaldía Mayor de Bogota. (2018). *PROGRAMA DE GESTIÓN DE LA VELOCIDAD Documento base*.
- Atzori, L., Iera, A., & Morabito, G. (2010). The Internet of Things: A survey. *Computer Networks: The International Journal of Computer and Telecommunications Networking*, 54(15), 2787–2805. <https://doi.org/10.1016/j.comnet.2010.05.010>
- BonnMotion. (2016). *BonnMotion - A mobility scenario generation and analysis tool*. <https://sys.cs.uos.de/bonnmotion/>
- Bui, K. H. N., & Jung, J. J. (2018). Internet of agents framework for connected vehicles: A case study on distributed traffic control system. *Journal of Parallel and Distributed Computing*, 116, 89–95. <https://doi.org/10.1016/j.jpdc.2017.10.019>
- Chang, C. Y., Yen, H. C., & Deng, D. J. (2016). V2V QoS Guaranteed Channel Access in IEEE 802.11p VANETs. *IEEE Transactions on Dependable and Secure Computing*, 13(1), 5–17. <https://doi.org/10.1109/TDSC.2015.2399912>
- Choffnes, D. R., & Bustamante, F. E. (2006). An integrated mobility and traffic model for vehicular wireless networks. *VANET - Proceedings of the Second ACM International Workshop on Vehicular Ad Hoc Networks, 2006*, 69–78. <https://doi.org/10.1145/1080754.1080765>
- Cisco IBSG. (2011). *The Internet of Things How the Next Evolution of the Internet Is Changing Everything*.
- Contreras, J., Zeadally, S., & Guerrero-Ibanez, J. A. (2017). Internet of Vehicles: Architecture, Protocols, and Security. *IEEE Internet of Things Journal*, 4662(c), 1–1.

<https://doi.org/10.1109/JIOT.2017.2690902>

Dandala, T. T., Krishnamurthy, V., & Alwan, R. (2017). Internet of Vehicles (IoV) for traffic management. *2017 International Conference on Computer, Communication and Signal Processing (ICCCSP)*, 1–4. <https://doi.org/10.1109/ICCCSP.2017.7944096>

Electronics Notes. (2017). *IEEE 802.11ah | Sub GHz Wi-Fi | Electronics Notes*. <https://www.electronics-notes.com/articles/connectivity/wifi-ieee-802-11/802-11ah-sub-ghz-wifi.php>

Gaitán Victoria, C., Diana Sánchez Guerrero, P., Diego Luis Buelvas Ramírez, P., Nelson Arturo Chaparro, P., Vanessa Cediél Sánchez, P., & Álvaro Montenegro Rodríguez, P. (2018). *Análisis demográfico y proyecciones poblacionales de Bogotá*.

Golestan, K., Soua, R., Karray, F., & Kamel, M. S. (2016). Situation awareness within the context of connected cars: A comprehensive review and recent trends. *Information Fusion*, 29, 68–83. <https://doi.org/10.1016/j.inffus.2015.08.001>

Guevara Riveros, A. Y. (2018). *Diseño de una Arquitectura para el monitoreo de los signos vitales en entornos restringidos*. <http://repository.udistrital.edu.co/handle/11349/14494>

Hafizah, A., Aman, M., Hashim, A.-H. A., Abdullah, A., Adibah, H., Ramli, M., & Islam, S. (2016). Network Simulators Parametric Comparison for Network Mobility Management. *International Journal of Future Generation Communication and Networking*, 9(10), 17–28. <https://doi.org/10.14257/ijfgcn.2016.9.10.02>

Huang, R., Wu, J., Long, C., Zhu, Y., Li, B., & Lin, Y.-B. (2017). SPRCA: Distributed Multisource Information Propagation in Multichannel VANETs. *IEEE Transactions on Vehicular Technology*, 66(12), 11306–11316. <https://doi.org/10.1109/TVT.2017.2726159>

Hussain, R., Hussain, F., & Zeadally, S. (2019). Integration of VANET and 5G Security: A review of design and implementation issues. *Future Generation Computer Systems*, 101, 843–864. <https://doi.org/10.1016/j.future.2019.07.006>

- IEEE. (2016). *802.11ah-2016 - 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)*. https://standards.ieee.org/standard/802_11ah-2016.html
- IETF. (2018). *RFC 3561 - Ad hoc On-Demand Distance Vector (AODV) Routing*. <https://tools.ietf.org/html/rfc3561>
- INRIX. (2014). *The future economic and environmental costs of gridlock in 2030*. [https://ibttta.org/sites/default/files/documents/MAF/Costs-of-Congestion-INRIX-Cebr-Report %283%29.pdf](https://ibttta.org/sites/default/files/documents/MAF/Costs-of-Congestion-INRIX-Cebr-Report%283%29.pdf)
- jist. (2017). *JiST - Java in Simulation Time / SWANS - Scalable Wireless Ad hoc Network Simulator*. <http://jist.ece.cornell.edu/index.html>
- Kombate, D., & Wanglina. (2017). The Internet of Vehicles Based on 5G Communications. *Proceedings - 2016 IEEE International Conference on Internet of Things; IEEE Green Computing and Communications; IEEE Cyber, Physical, and Social Computing; IEEE Smart Data, IThings-GreenCom-CPSCoM-Smart Data 2016*, 445–448. <https://doi.org/10.1109/iThings-GreenCom-CPSCoM-SmartData.2016.105>
- López, R., & Guijarro, R. (2018). *Análisis del protocolo IEEE 802.11.p en sistemas de comunicación vehiculares V2X*.
- Luo, G., Yuan, Q., Zhou, H., Cheng, N., Liu, Z., Yang, F., & Shen, X. S. (2018). Cooperative vehicular content distribution in edge computing assisted 5G-VANET. *China Communications*, 15(7), 1–17. <https://doi.org/10.1109/CC.2018.8424578>
- Mai, A., Schlesinger, D., & Ibsg, C. (2011). *Connected Vehicles and Government A Catalyst To Unlock the Societal Benefits of Transportation*. https://www.cisco.com/c/dam/en_us/about/ac79/docs/mfg/Connected-Vehicles_Government.pdf
- Mcewen, A., & Cassimally, H. (2014). *Designing the Internet of Things*. Wiley.

- Montes, L., & Clemente, P. (2016). *Trabajadores conectados al Internet de las Cosas y autobuses que hablan*. El Mundo. <https://www.elmundo.es/economia/2016/10/31/5817127cca474149528b456f.html>
- Nitti, M., Girau, R., Floris, A., & Atzori, L. (2014). On adding the social dimension to the Internet of Vehicles: Friendship and middleware. *2014 IEEE International Black Sea Conference on Communications and Networking (BlackSeaCom)*, 134–138. <https://doi.org/10.1109/BlackSeaCom.2014.6849025>
- Nsnam. (2017). *The Network Simulator - ns-2*. <https://www.isi.edu/nsnam/ns/>
- Nsnam. (2018). *ns-3 | a discrete-event network simulator for internet systems*. <https://www.nsnam.org/overview/what-is-ns-3/>
- omnetpp. (2018). *OMNeT++ Discrete Event Simulator*. <https://omnetpp.org/>
- Organisation Internationale des Constructeurs d'Automobiles (OICA). (2017). *Number of vehicles in use worldwide 2015 | Statistic*. <https://www.statista.com/statistics/281134/number-of-vehicles-in-use-worldwide/>
- Postscapes. (2018). *IoT Overview Handbook*. <https://www.postscapes.com/iot/#definition>
- Qi, W., Song, Q., Wang, X., Guo, L., & Ning, Z. (2018). SDN-Enabled Social-Aware Clustering in 5G-VANET Systems. *IEEE Access*, 6, 28213–28224. <https://doi.org/10.1109/ACCESS.2018.2837870>
- Sahbi, R., Ghanemi, S., & Djouani, R. (2019). A Network Model for Internet of vehicles based on SDN and Cloud Computing. *Proceedings - 2018 International Conference on Wireless Networks and Mobile Communications, WINCOM 2018*, 1–4. <https://doi.org/10.1109/WINCOM.2018.8629610>
- Sanchez, J. F., & Cobo, L. A. (2014). Theoretical model of congestion control in VANET networks. *2014 IEEE Colombian Conference on Communications and Computing, COLCOM 2014 - Conference Proceedings*.

<https://doi.org/10.1109/ColComCon.2014.6860400>

Sánchez Sánchez, J. A. (2017). *Redes vehiculares aplicadas a la movilidad inteligente y sostenibilidad ambiental en entornos de ciudades inteligentes* [Universidad de Oviedo].

<https://dialnet.unirioja.es/servlet/tesis?codigo=207357>

scalable-networks. (2017). *Constructive Simulation | SCALABLE Networks*.

<https://www.scalable-networks.com/simulation-platform-architecture>

Scrum.org. (2018). *¿Qué es Scrum?* <https://www.scrum.org/resources/blog/que-es-scrum>

Secretaría Distrital de Ambiente. (2018). *Información detallada del indicador – Observatorio Ambiental de Bogota*. <http://oab.ambientebogota.gov.co/indicadores/?id=272>

Seliem, H., Shahidi, R., Ahmed, M. H., & Shehata, M. S. (2018). Drone-based highway-VANET and das service. *IEEE Access*, 6, 20125–20137.

<https://doi.org/10.1109/ACCESS.2018.2824839>

Sharma, S., & Kaushik, B. (2019). A survey on internet of vehicles: Applications, security issues & solutions. *Vehicular Communications*, 20, 100182.

<https://doi.org/10.1016/j.vehcom.2019.100182>

Strategy & Formely Booz & Company. (n.d.). *Racing ahead – The Connected C@r 2014 study*. Retrieved August 11, 2017, from

<https://www.strategyand.pwc.com/media/file/Racing-ahead.pdf>

SUMO – *Simulation of Urban MObility*. (2017). DLR - Institute of Transportation Systems.

https://www.dlr.de/ts/en/desktopdefault.aspx/tabid-9883/16931_read-41000/

Surati, S., Jinwala, D. C., & Garg, S. (2017). A survey of simulators for P2P overlay networks with a case study of the P2P tree overlay using an event-driven simulator. *Engineering Science and Technology, an International Journal*, 20(2), 705–720.

<https://doi.org/10.1016/j.jestch.2016.12.010>

- Tan, H., & Chung, I. (2020). Secure Authentication and Key Management With Blockchain in VANETs. *IEEE Access*, 8, 2482–2498. <https://doi.org/10.1109/ACCESS.2019.2962387>
- Tian, R., Jiao, Z., Bian, G., Huang, Z., & Hou, Y. (2016). A Social-based Data Forwarding Mechanism for V2V communication in VANETs. *Proceedings of the 2015 10th International Conference on Communications and Networking in China, CHINACOM 2015*, 595–599. <https://doi.org/10.1109/CHINACOM.2015.7498007>
- U.S. Census Bureau. (2012). *La Oficina del Censo de los EE. UU. proyecta que dentro de medio siglo la población del país crecerá más lentamente, será de mayor edad y más diversa.*
- VanetMobiSim. (2018). *VanetMobiSim - Newcom | Institut Eurecom | Politecnico di Torino.* <http://vanet.eurecom.fr/>
- Voelcker, J. (2011). *It's Official: We Now Have One Billion Vehicles On The Planet.* http://www.greencarreports.com/news/1065070_its-official-we-now-have-one-billion-vehicles-on-the-planet
- Weingärtner, E., Vom Lehn, H., & Wehrle, K. (2009). A performance comparison of recent network simulators. *IEEE International Conference on Communications.* <https://doi.org/10.1109/ICC.2009.5198657>
- Wu, J., Fang, M., Li, H., & Li, X. (2020). RSU-Assisted Traffic-Aware Routing Based on Reinforcement Learning for Urban VANETs. *IEEE Access*, 1–1. <https://doi.org/10.1109/ACCESS.2020.2963850>
- Xia, Y., Qin, X., Liu, B., & Zhang, P. (2018). A greedy traffic light and queue aware routing protocol for urban VANETs. *China Communications*, 15(7), 77–87. <https://doi.org/10.1109/CC.2018.8424605>
- Xie, L., Ding, Y., Yang, H., & Wang, X. (2019). Blockchain-Based Secure and Trustworthy Internet of Things in SDN-Enabled 5G-VANETs. *IEEE Access*, 7, 56656–56666. <https://doi.org/10.1109/ACCESS.2019.2913682>

Xie, Y., Ho, I. W. H., & Magsino, E. R. (2018). The Modeling and Cross-Layer Optimization of 802.11p VANET Unicast. *IEEE Access*, 6, 171–186. <https://doi.org/10.1109/ACCESS.2017.2761788>

Yang, F., Wang, S., Li, J., Liu, Z., & Sun, Q. (2014). An overview of Internet of Vehicles. *China Communications*, 11(10), 1–15. <https://doi.org/10.1109/CC.2014.6969789>

ANEXOS

ANEXO 1: MODELO DE SIMULACIÓN PARA RED VANET TRADICIONAL

```
set val(chan) Channel/WirelessChannel ;# channel type
set val(prop) Propagation/TwoRayGround ;# radio-propagation model
set val(netif1) Phy/WirelessPhy ;# network interface type
set val(netif2) Phy/WirelessPhy ;# network interface type
set val(mac) Mac/802_11 ;# MAC type
set val(ifq) Queue/DropTail/PriQueue ;# interface queue type
set val(ll) LL ;# link layer type
set val(ant) Antenna/OmniAntenna ;# antenna model
set val(ifqlen) 50 ;# max packet in ifq
set val(nn) 99 ;# number of mobilenodes
set val(rp) AODV ;# routing protocol
set opt(x) 6803 ;# X dimension of topography
set opt(y) 12783 ;# Y dimension of topography
set val(stop) 100.0 ;# time of simulation end
```

```
set ns_ [new Simulator]
```

```
set topo [new Topography]
$topo load_flatgrid $opt(x) $opt(y)
create-god $val(nn)
```

```
set tracefile [open zona1.tr w]
$ns_ trace-all $tracefile
```

```
set dist(20m) 4.80696e-07
set dist(26m) 2.84435e-07
set dist(27m) 2.63756e-07
set dist(28m) 2.45253e-07
set dist(25m) 3.07645e-07
set dist(30m) 2.13643e-07
set dist(35m) 1.56962e-07
set dist(50m) 7.69113e-08
set dist(75m) 3.41828e-08
set dist(60m) 5.34106e-08
set dist(70m) 3.92405e-08
set dist(250m) 3.65262e-10
set dist(500m) 2.28289e-11
```

```
Antenna/OmniAntenna set X_ 0
Antenna/OmniAntenna set Y_ 0
Antenna/OmniAntenna set Z_ 1.5
```

```
Antenna/OmniAntenna set Gt_ 1.0
Antenna/OmniAntenna set Gr_ 1.0
```

```
$val(netif1) set CPTthresh_ 10.0
$val(netif1) set CSTthresh_ $dist(250m)
$val(netif1) set RXThresh_ $dist(250m)
$val(netif1) set Rb_ 2*1e6
$val(netif1) set Pt_ 0.2818
$val(netif1) set freq_ 914e+6
$val(netif1) set L_ 1.0
```

```
$val(netif2) set CPTthresh_ 10.0
$val(netif2) set CSTthresh_ $dist(500m)
$val(netif2) set RXThresh_ $dist(500m)
$val(netif2) set Rb_ 2*1e6
$val(netif2) set Pt_ 0.2818
$val(netif2) set freq_ 914e+6
$val(netif2) set L_ 1.0
```

```
set namfile [open zona1.nam w]
$ns_ namtrace-all $namfile
$ns_ namtrace-all-wireless $namfile $opt(x) $opt(y)
```

```
set chan_1_ [new $val(chan)];
$ns_ node-config -adhocRouting $val(rp) \
    -llType $val(ll) \
    -macType $val(mac) \
    -ifqType $val(ifq) \
    -ifqLen $val(ifqlen) \
    -antType $val(ant) \
    -propType $val(prop) \
    -phyType $val(netif1) \
    -channel $chan_1_ \
    -topoInstance $topo \
    -agentTrace ON \
    -routerTrace ON \
    -macTrace ON \
#    -macTrace ON \
    -movementTrace ON
```

```
for {set i 0} {$i < 1} {incr i} {
    set node_($i) [$ns_ node]
    $node_($i) color red
    $node_($i) random-motion 0 ;# disable random motion
    $ns_ initial_node_pos $node_($i) 20
}
```

```
$ns_ node-config -adhocRouting $val(rp) \
    -llType $val(ll) \
```

```
-macType    $val(mac) \  
-ifqType    $val(ifq) \  
-ifqLen     $val(ifqlen) \  
-antType    $val(ant) \  
-propType   $val(prop) \  
-phyType    $val(netif2) \  
-channel    $chan_1_ \  
-topoInstance $topo \  
-agentTrace ON \  
-routerTrace ON \  
-macTrace   ON \  
#           -macTrace   ON \  
-movementTrace ON  
  
for {set i 1} {$i < $val(nn)} {incr i} {  
    set node_($i) [$ns_ node]  
    $node_($i) random-motion 0           ;# disable random motion  
    $ns_ initial_node_pos $node_($i) 20  
}  
  
source mobility.tcl  
source trafficzona1  
  
set tcp0 [new Agent/TCP]  
$ns_ attach-agent $node_(3) $tcp0  
set sink2 [new Agent/TCPSink]  
$ns_ attach-agent $node_(8) $sink2  
$ns_ connect $tcp0 $sink2  
$tcp0 set packetSize_ 1500  
  
set tcp1 [new Agent/TCP]  
$ns_ attach-agent $node_(14) $tcp1  
set sink3 [new Agent/TCPSink]  
$ns_ attach-agent $node_(10) $sink3  
$ns_ connect $tcp1 $sink3  
$tcp1 set packetSize_ 1500  
  
set ftp0 [new Application/FTP]  
$ftp0 attach-agent $tcp0  
$ns_ at 1.0 "$ftp0 start"  
$ns_ at 95.0 "$ftp0 stop"  
  
set ftp1 [new Application/FTP]  
$ftp1 attach-agent $tcp1  
$ns_ at 2.0 "$ftp1 start"  
$ns_ at 100.0 "$ftp1 stop"  
  
proc finish {} {  
    global ns_ tracefile namfile
```

```
$ns_ flush-trace
close $tracefile
close $namfile
#exec nam AODV.nam &
exit 0
}

for {set i 0} {$i < $val(nn)} {incr i} {
    $ns_ at $val(stop) "\$node_($i) reset"
}

$ns_ at $val(stop) "$ns_ nam-end-wireless $val(stop)"
$ns_ at $val(stop) "finish"
$ns_ at $val(stop) "puts \"done\" ; $ns_ halt"
$ns_ run
```

ANEXO 2: MODELO DE SIMULACIÓN PARA RED VANET INTERCONECTADA CON IOT

```
Phy/WirelessPhyExt set CStresh_          3.9810717055349694e-13;
Phy/WirelessPhyExt set Pt_              0.1 ;
Phy/WirelessPhyExt set freq_            5.9e+9
Phy/WirelessPhyExt set noise_floor_     1.26e-13 ;
Phy/WirelessPhyExt set L_               1.0 ;
Phy/WirelessPhyExt set PowerMonitorThresh_ 3.981071705534985e-18 ;
Phy/WirelessPhyExt set HeaderDuration_  0.000040 ;
Phy/WirelessPhyExt set BasicModulationScheme_ 0
Phy/WirelessPhyExt set PreambleCaptureSwitch_ 1
Phy/WirelessPhyExt set DataCaptureSwitch_ 1
Phy/WirelessPhyExt set SINR_PreambleCapture_ 3.1623; ;
Phy/WirelessPhyExt set SINR_DataCapture_ 10.0; ;
Phy/WirelessPhyExt set trace_dist_      1e6 ;
Phy/WirelessPhyExt set PHY_DBG_         0
```

```
Mac/802_11Ext set CWMin_                15 ;
Mac/802_11Ext set CWMax_                1023 ;
Mac/802_11Ext set SlotTime_             0.000052 ;
Mac/802_11Ext set SIFS_                 0.0000160 ;
Mac/802_11Ext set ShortRetryLimit_      7
Mac/802_11Ext set LongRetryLimit_       4
Mac/802_11Ext set HeaderDuration_       0.000040
Mac/802_11Ext set SymbolDuration_       0.000008
Mac/802_11Ext set BasicModulationScheme_ 0
Mac/802_11Ext set use_802_11a_flag_     true
Mac/802_11Ext set RTSThreshold_         2346
Mac/802_11Ext set MAC_DBG_              0
```

```
set val(chan) Channel/WirelessChannel ;# channel type
set val(prop) Propagation/TwoRayGround ;# radio-propagation model
set val(netif1) Phy/WirelessPhyExt ;# network interface type
set val(netif2) Phy/WirelessPhyExt ;# network interface type
set val(mac) Mac/802_11Ext ;# MAC type
set val(ifq) Queue/DropTail/PriQueue ;# interface queue type
set val(ll) LL ;# link layer type
set val(ant) Antenna/OmniAntenna ;# antenna model
set val(ifqlen) 50 ;# max packet in ifq
set val(nn) 99 ;# number of mobilenodes
set val(rp) AODV ;# routing protocol
set opt(x) 6803 ;# X dimension of topography
set opt(y) 12783 ;# Y dimension of topography
set val(stop) 100.0 ;# time of simulation end
```

```
set ns_ [new Simulator]

set topo [new Topography]
$topo load_flatgrid $opt(x) $opt(y)
create-god $val(nn)

set tracefile [open zona1iot.tr w]
$ns_ trace-all $tracefile

set dist(20m) 4.80696e-07
set dist(26m) 2.84435e-07
set dist(27m) 2.63756e-07
set dist(28m) 2.45253e-07
set dist(25m) 3.07645e-07
set dist(30m) 2.13643e-07
set dist(35m) 1.56962e-07
set dist(50m) 7.69113e-08
set dist(75m) 3.41828e-08
set dist(60m) 5.34106e-08
set dist(70m) 3.92405e-08
set dist(250m) 3.65262e-10
set dist(500m) 2.28289e-11

Antenna/OmniAntenna set X_ 0
Antenna/OmniAntenna set Y_ 0
Antenna/OmniAntenna set Z_ 1.5
Antenna/OmniAntenna set Gt_ 1.0
Antenna/OmniAntenna set Gr_ 1.0

set namfile [open zona1iot.nam w]
$ns_ namtrace-all $namfile
$ns_ namtrace-all-wireless $namfile $opt(x) $opt(y)

set chan_1_ [new $val(chan)];
$ns_ node-config -adhocRouting $val(rp) \
    -llType $val(ll) \
    -macType $val(mac) \
    -ifqType $val(ifq) \
    -ifqLen $val(ifqlen) \
    -antType $val(ant) \
    -propType $val(prop) \
    -phyType $val(netif1) \
    -channel $chan_1_ \
    -topoInstance $topo \
    -agentTrace ON \
    -routerTrace ON \
    -macTrace ON \
#    -macTrace ON \
    -movementTrace ON
```

```

for {set i 0} {$i < 1} {incr i} {
    set node_($i) [$ns_ node]
    $node_($i) color red
    $node_($i) random-motion 0 ;# disable random motion
    $ns_ initial_node_pos $node_($i) 20
}

$ns_ node-config -adhocRouting $val(rp) \
    -llType $val(ll) \
    -macType $val(mac) \
    -ifqType $val(ifq) \
    -ifqLen $val(ifqlen) \
    -antType $val(ant) \
    -propType $val(prop) \
    -phyType $val(netif2) \
    -channel $chan_1_ \
    -topoInstance $topo \
    -agentTrace ON \
    -routerTrace ON \
    -macTrace ON \
#    -macTrace ON \
    -movementTrace ON
    for {set i 1} {$i < $val(nn)} {incr i} {
        set node_($i) [$ns_ node]
        $node_($i) random-motion 0 ;# disable random motion
        $ns_ initial_node_pos $node_($i) 20
    }
}

source mobility.tcl
source trafficzona1

set tcp0 [new Agent/TCP]
$ns_ attach-agent $node_(3) $tcp0
set sink2 [new Agent/TCPSink]
$ns_ attach-agent $node_(8) $sink2
$ns_ connect $tcp0 $sink2
$tcp0 set packetSize_ 1500

set tcp1 [new Agent/TCP]
$ns_ attach-agent $node_(14) $tcp1
set sink3 [new Agent/TCPSink]
$ns_ attach-agent $node_(10) $sink3
$ns_ connect $tcp1 $sink3
$tcp1 set packetSize_ 1500

set ftp0 [new Application/FTP]
$ftp0 attach-agent $tcp0
$ns_ at 1.0 "$ftp0 start"
$ns_ at 95.0 "$ftp0 stop"

```

```
set ftp1 [new Application/FTP]
$ftp1 attach-agent $tcp1
$ns_ at 2.0 "$ftp1 start"
$ns_ at 100.0 "$ftp1 stop"

proc finish {} {
    global ns_ tracefile namfile
    $ns_ flush-trace
    close $tracefile
    close $namfile
    #exec nam AODV.nam &
    exit 0
}

for {set i 0} {$i < $val(nn)} {incr i} {
    $ns_ at $val(stop) "\$node_($i) reset"
}

$ns_ at $val(stop) "$ns_ nam-end-wireless $val(stop)"
$ns_ at $val(stop) "finish"
$ns_ at $val(stop) "puts \"done\\\" ; $ns_ halt"
$ns_ run
```

ANEXO 3: RESULTADOS DE LA SEGUNDA SERIE DE SIMULACIONES

En esta segunda serie de simulaciones se han variado los parámetros de nodos móviles a 150 y el tiempo de simulación a 150 segundos.

Packet Delivery Ratio (PDF)

En la Tabla 21 se muestran los resultados matemáticos de *PDF* para las segundas simulaciones realizadas con una red Vanet Tradicional.

	Zona 1	Zona 2	Zona 3	Zona 4	Zona 5
Packet Delivery Ratio	2,2495156	6,9657142	5,4882972	1,651229467	4,2690768

Tabla 21: Packet Delivery Ratio de las segundas simulaciones realizadas con una red Vanet Tradicional

En la Tabla 22 se encuentran los resultados para una red Vanet interconecta con IoT.

	Zona 1	Zona 2	Zona 3	Zona 4	Zona 5
Packet Delivery Ratio	8,362098558	24,7684544	11,76782727	5,969402867	13,6065134

Tabla 22: Packet Delivery Ratio de las simulaciones realizadas con una red Vanet interconecta con IoT

Gráficamente los resultados de *Packet Delivery Ratio* por zonas se pueden observar en las Figuras: Figura 95, Figura 96, Figura 97, Figura 98 y Figura 99. Donde se contrasta el resultado de la zona con red Vanet tradicional versus la zona con la red Vanet interconectada con IoT.

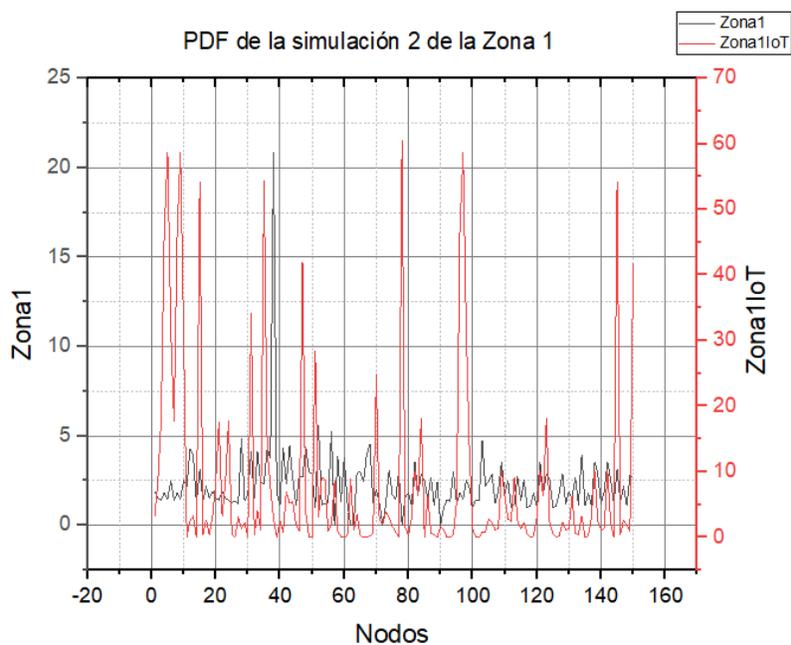


Figura 95: PDF de la simulación 2 para la Zona 1
Fuente: Autor. Generado por medio del programa Origin 2020

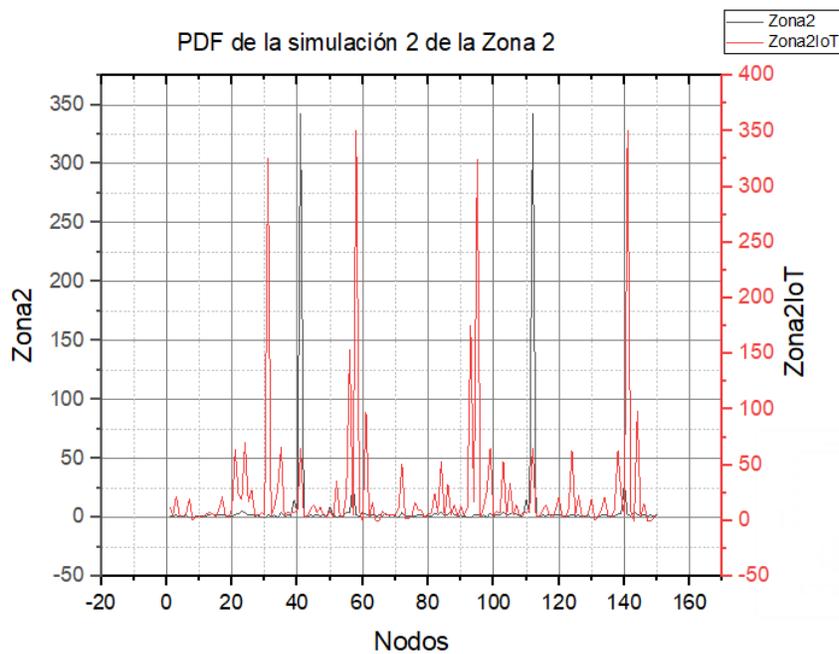


Figura 96: PDF de la simulación 2 para la Zona 2
Fuente: Autor. Generado por medio del programa Origin 2020

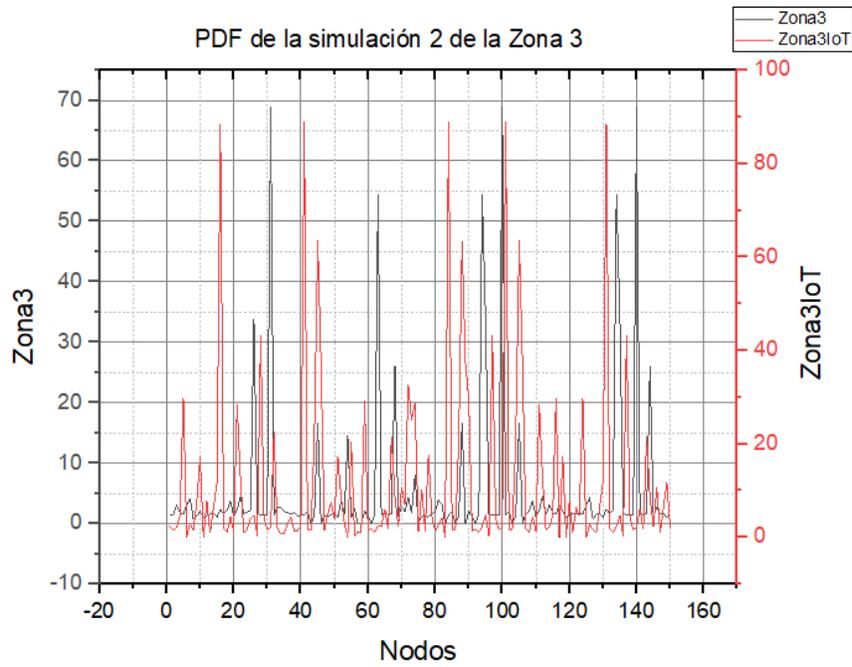


Figura 97: PDF de la simulación 2 para la Zona 3
Fuente: Autor. Generado por medio del programa Origin 2020

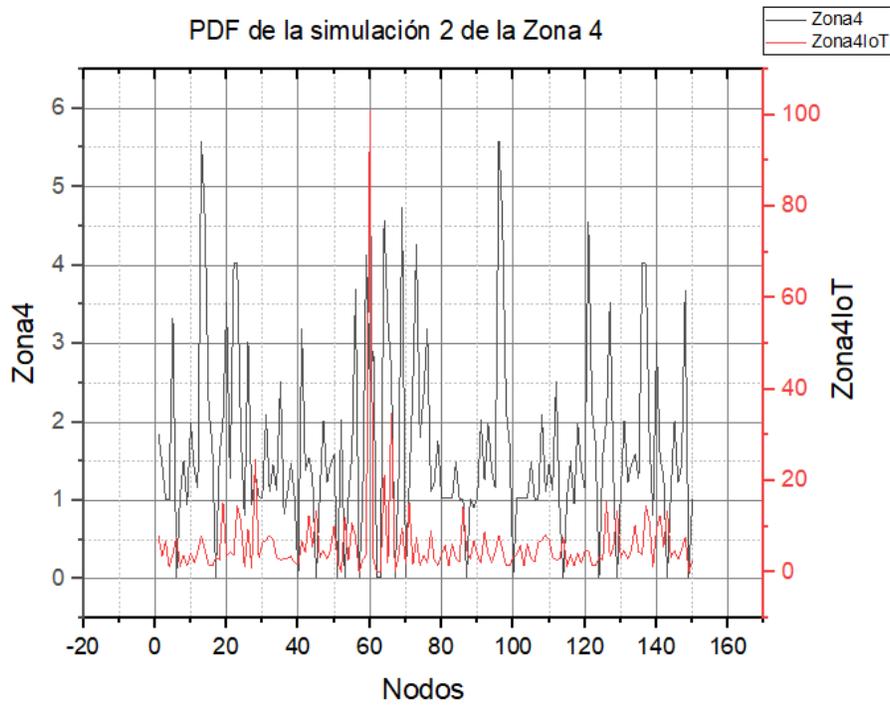


Figura 98: PDF de la simulación 2 para la Zona 4
Fuente: Autor. Generado por medio del programa Origin 2020

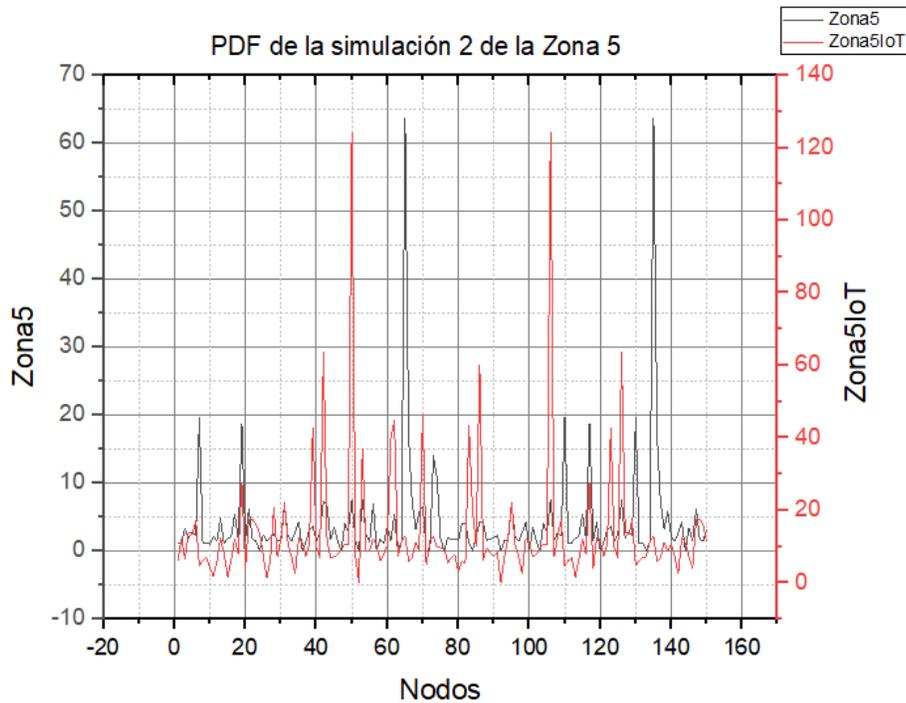


Figura 99: PDF de la simulación 2 para la Zona 5
Fuente: Autor. Generado por medio del programa Origin 2020

De estos resultados se observa que nuevamente hay mejor rendimiento en las simulaciones de red Vanet interconectada con IoT,

Throughput

Gráficamente los resultados *Throughput de generación de bits* por zonas se pueden observar en las Figuras: Figura 100, Figura 101, Figura 102, Figura 103 y Figura 104 donde se contrasta el resultado de la zona con red Vanet tradicional versus la zona con la red Vanet interconectada con IoT.

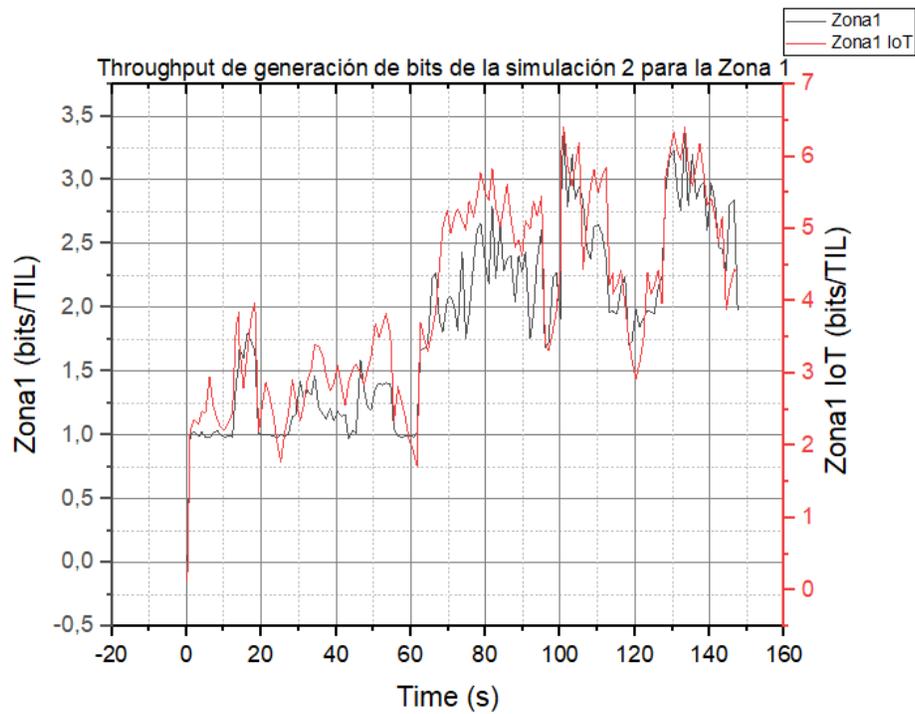


Figura 100: Throughput de generación de bits de la simulación 2 para la Zona 1
Fuente: Autor. Generado por medio del programa Origin 2020

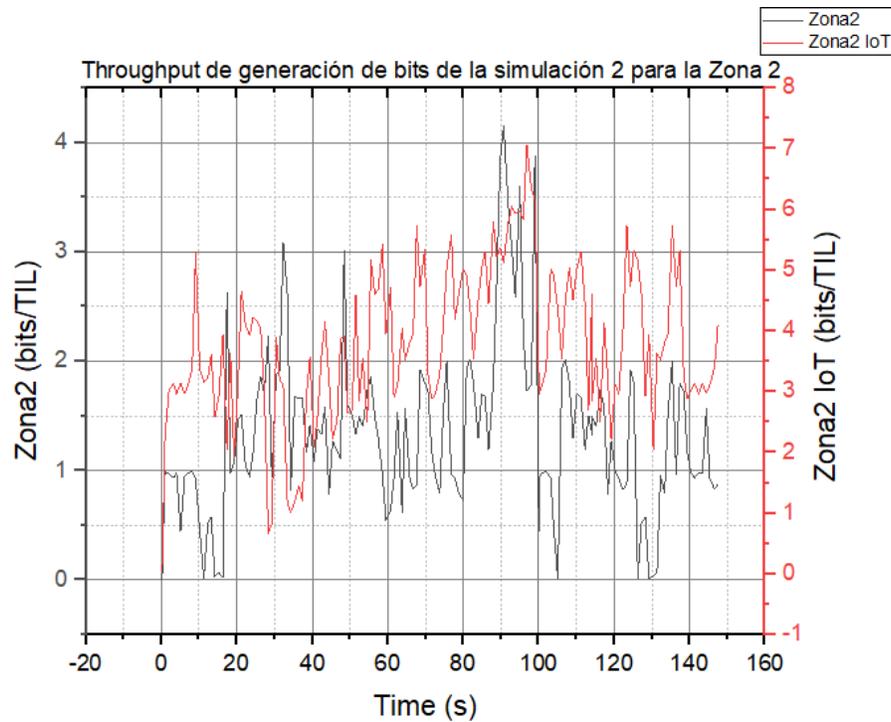


Figura 101: Throughput de generación de bits de la simulación 2 para la Zona 2
Fuente: Autor. Generado por medio del programa Origin 2020

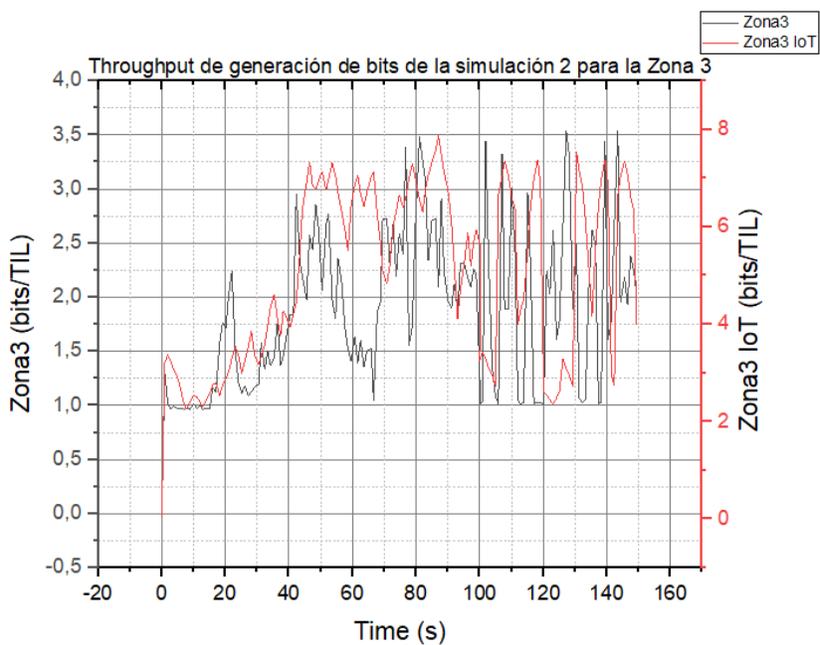


Figura 102: Throughput de generación de bits de la simulación 2 para la Zona 3
Fuente: Autor. Generado por medio del programa Origin 2020

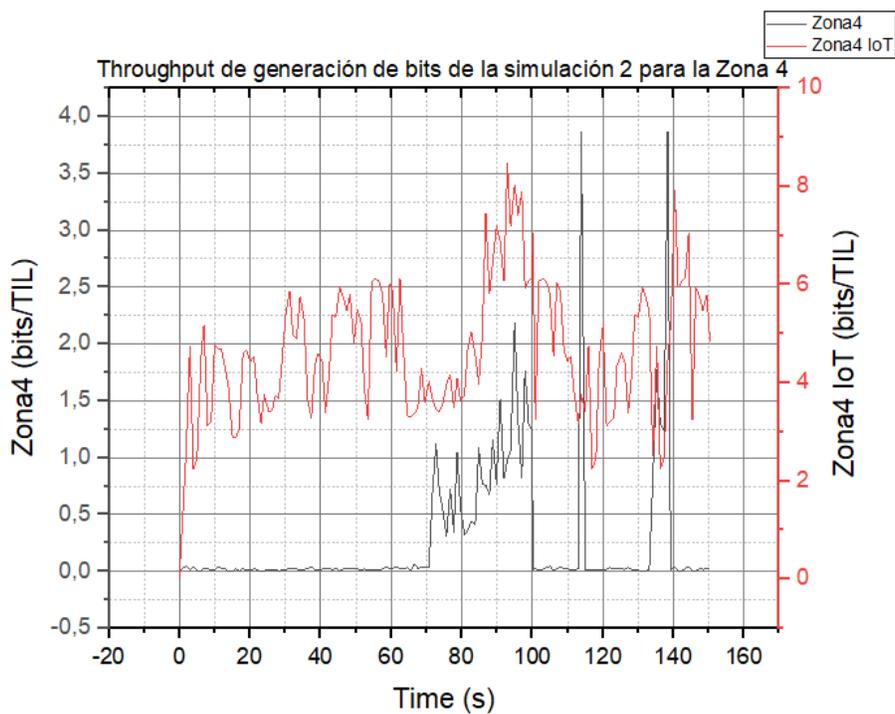


Figura 103: Throughput de generación de bits de la simulación 2 para la Zona 4
Fuente: Autor. Generado por medio del programa Origin 2020

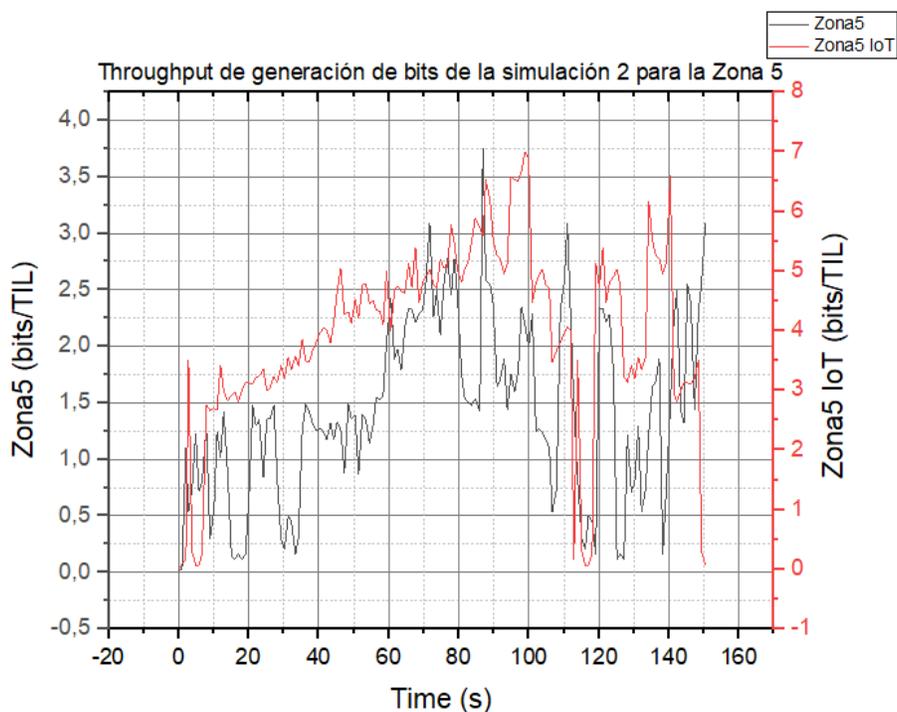


Figura 104: Throughput de generación de bits de la simulación 2 para la Zona 5
Fuente: Autor. Generado por medio del programa Origin 2020

Dentro de las simulaciones realizadas, el *throughput* se evidencia notablemente que las simulaciones realizadas de redes Vanet interconectadas con IoT presentan menor *overhead*.

End-End delay

En la Tabla 23 se observan los resultados matemáticos de *End-End delay* para las segundas simulaciones realizadas con una red Vanet Tradicional.

	Zona 1	Zona 2	Zona 3	Zona 4	Zona 5
End-End delay	0,60076747	1,5661676	1,75553764	2,70326453	0,981278217

Tabla 23: Promedio End-End delay de las segundas simulaciones realizadas con una red Vanet tradicional

En la Tabla 24 se encuentran los resultados para una red Vanet interconecta con IoT.

	Zona 1	Zona 2	Zona 3	Zona 4	Zona 5
End-End delay	0,27913912	0,79968762	0,88017211	1,32797704	0,515587987

Tabla 24: Promedio End-End delay de las segundas simulaciones realizadas con una red Vanet interconecta con IoT

De estos resultados nuevamente se puede observar que hay menores retardos en las simulaciones de red Vanet interconectada con IoT, ya que en todas las zonas se obtuvo un valor inferior al presentado en la red Vanet tradicional.

Packet Loss

En la Tabla 25 se muestran los resultados matemáticos de *Packet Loss* para las segundas simulaciones realizadas con una red Vanet Tradicional.

	Zona 1	Zona 2	Zona 3	Zona 4	Zona 5
Packet Loss	0,48037185	0,64706745	0,43443575	0,5236741	0,63381711

Tabla 25: Packet Loss de las segundas simulaciones realizadas con una red Vanet tradicional

En la Tabla 26 se encuentran los resultados para una red Vanet interconecta con IoT.

	Zona 1	Zona 2	Zona 3	Zona 4	Zona 5
Packet Loss	0,38194865	0,34106605	0,32852655	0,4808331	0,42408922

Tabla 26: Packet Loss de las segundas simulaciones realizadas con una red Vanet interconecta con IoT

Gráficamente los resultados *Packet Loss* por zonas se pueden observar en las Figuras: Figura 105, Figura 106, Figura 107, Figura 108 y Figura 109 donde se contrasta el resultado de la zona con red Vanet tradicional versus la zona con la red Vanet interconectada con IoT.

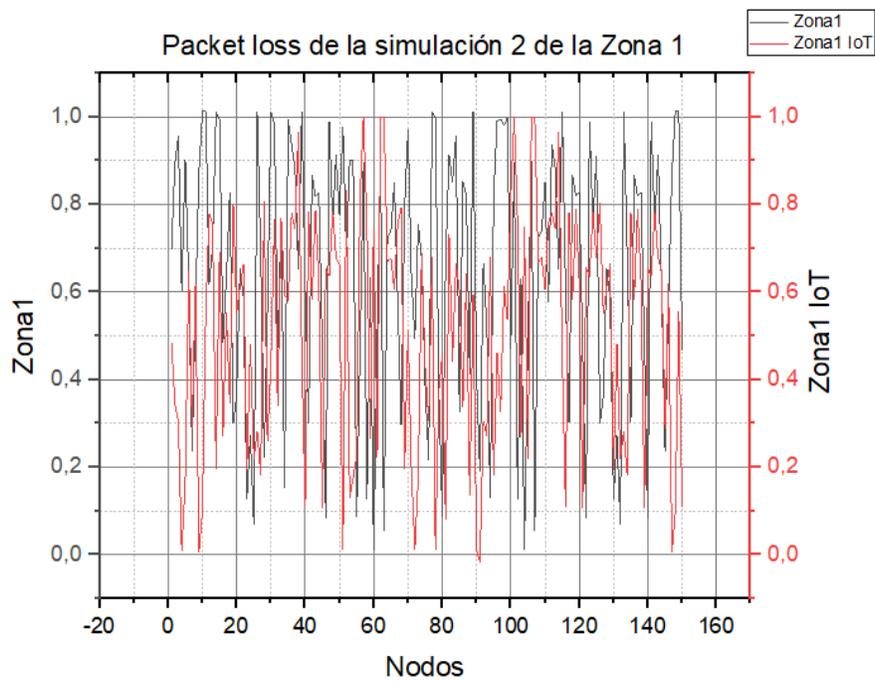


Figura 105: Packet Loss de la segunda simulación para la Zona 1
Fuente: Autor. Generado por medio del programa Origin 2020

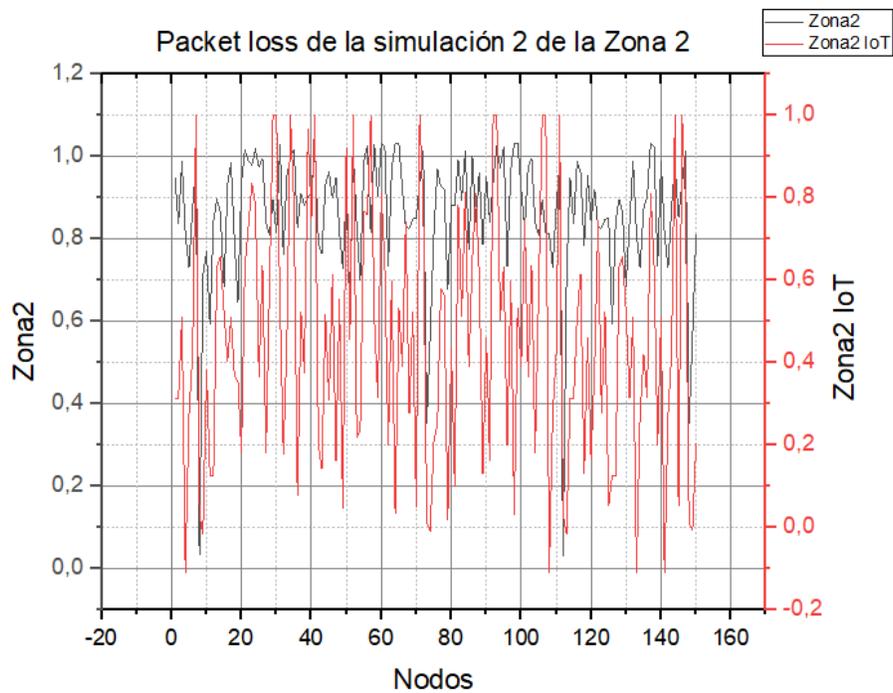


Figura 106: Packet Loss de la segunda simulación para la Zona 2
Fuente: Autor. Generado por medio del programa Origin 2020

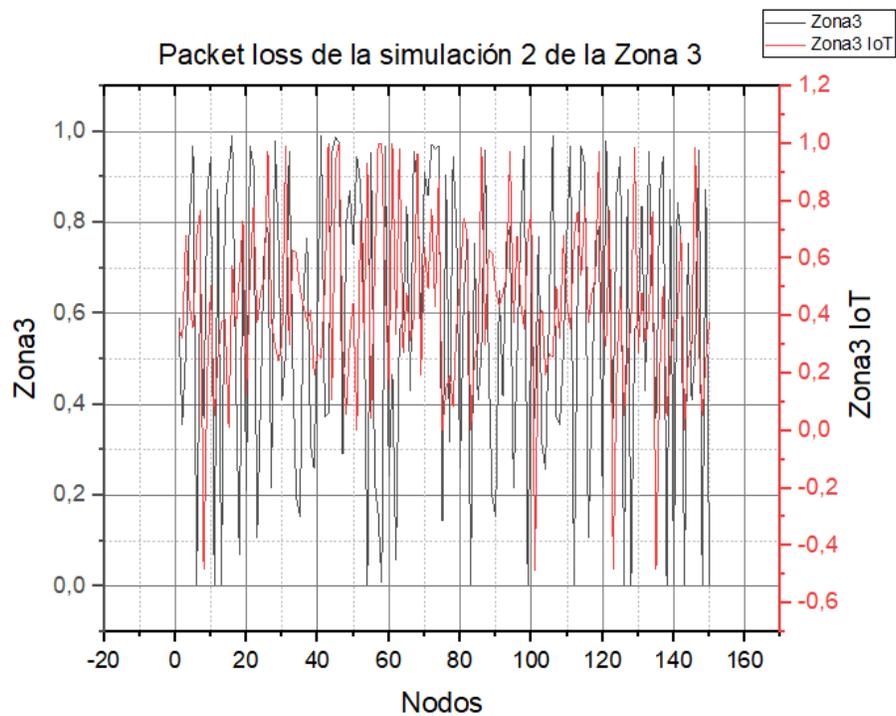


Figura 107: Packet Loss de la segunda simulación para la Zona 3
Fuente: Autor. Generado por medio del programa Origin 2020

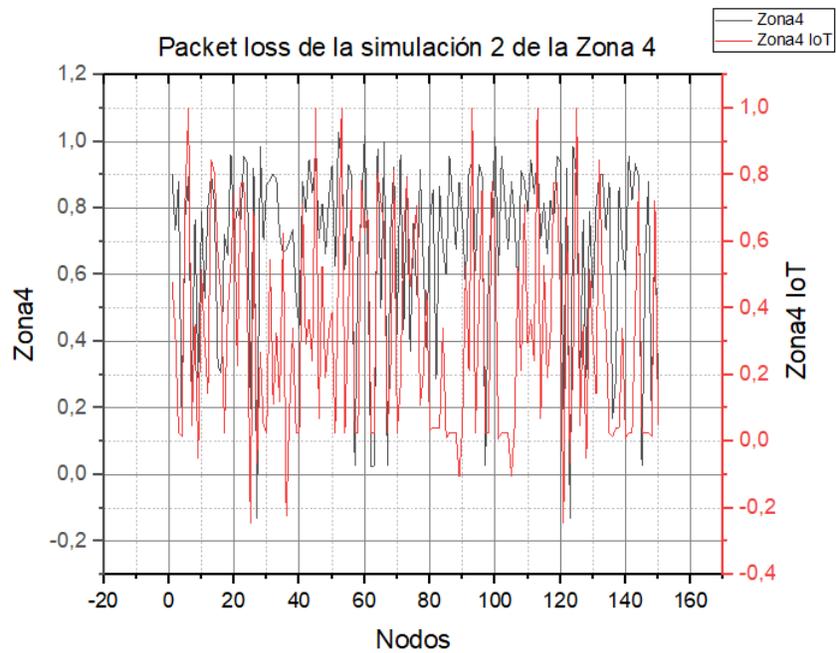


Figura 108: Packet Loss de la segunda simulación para la Zona 4
Fuente: Autor. Generado por medio del programa Origin 2020

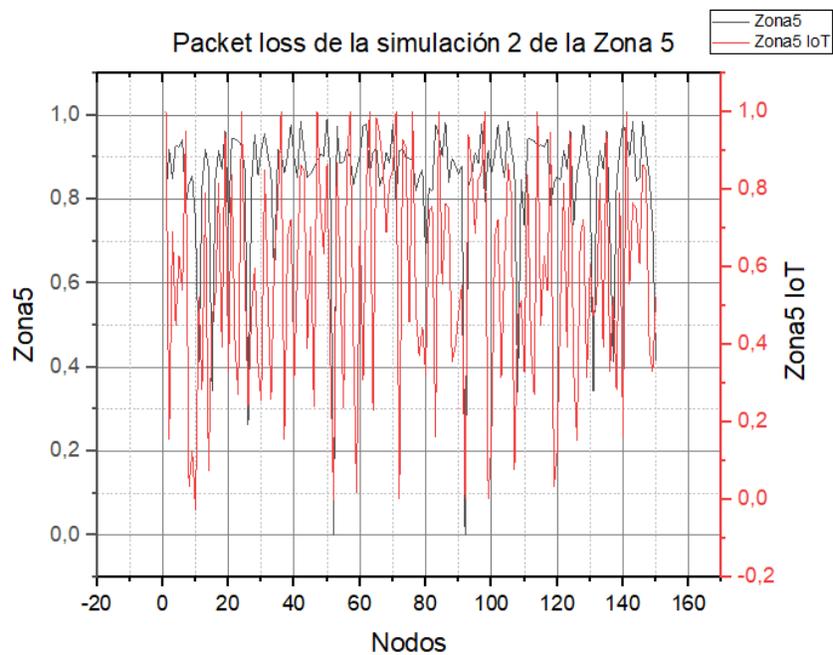


Figura 109: Packet Loss de la segunda simulación para la Zona 5
Fuente: Autor. Generado por medio del programa Origin 2020

De estos resultados se observa nuevamente que hay menor pérdida de paquetes en las simulaciones de red Vanet interconectada con IoT.

Packet Loss Ratio

En la Tabla 27 se muestran los resultados matemáticos de *Packet Loss* para las segundas simulaciones realizadas con una red Vanet Tradicional.

	Zona 1	Zona 2	Zona 3	Zona 4	Zona 5
Packet Loss Ratio	48,27052	64,706745	43,14834	52,36741	63,381711

Tabla 27: Packet Loss Ratio de las segundas simulaciones realizadas con una red Vanet tradicional

En la Tabla 28 se encuentran los resultados para una red Vanet interconecta con IoT.

	Zona 1	Zona 2	Zona 3	Zona 4	Zona 5
Packet Loss Ratio	38,2482	34,106605	32,675915	48,08331	42,408922

Tabla 28: Packet Loss Ratio de las segundas simulaciones realizadas con una red Vanet interconecta con IoT

Gráficamente los resultados *Packet Loss Ratio* por zonas se pueden observar en las Figuras: Figura 110, Figura 111, Figura 112, Figura 113 y Figura 114 donde se contrasta el resultado de la zona con red Vanet tradicional versus la zona con la red Vanet interconectada con IoT.

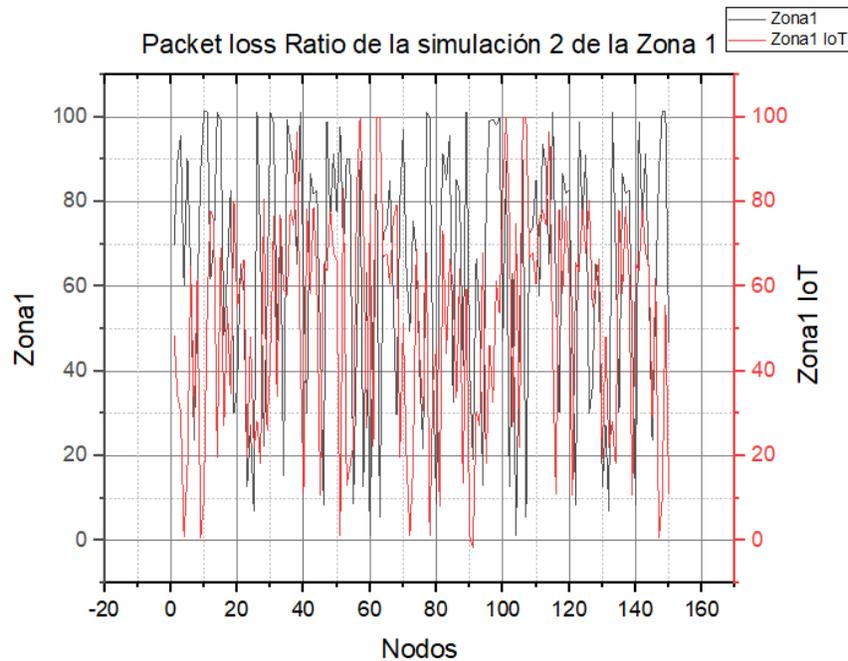


Figura 110: Packet Loss Ratio de la segunda simulación para la Zona 1
Fuente: Autor. Generado por medio del programa Origin 2020

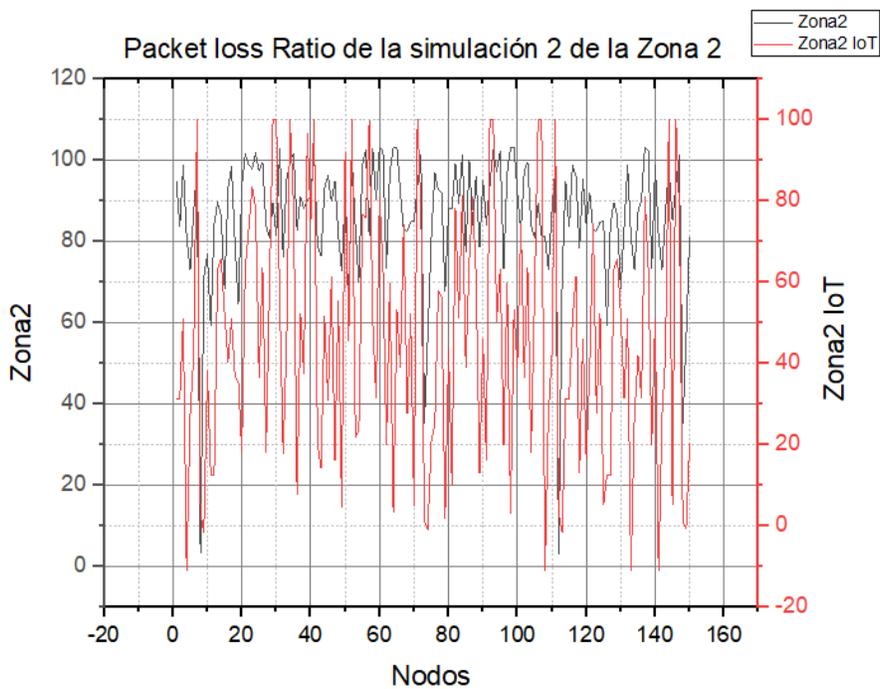


Figura 111: Packet Loss Ratio de la segunda simulación para la Zona 2
Fuente: Autor. Generado por medio del programa Origin 2020

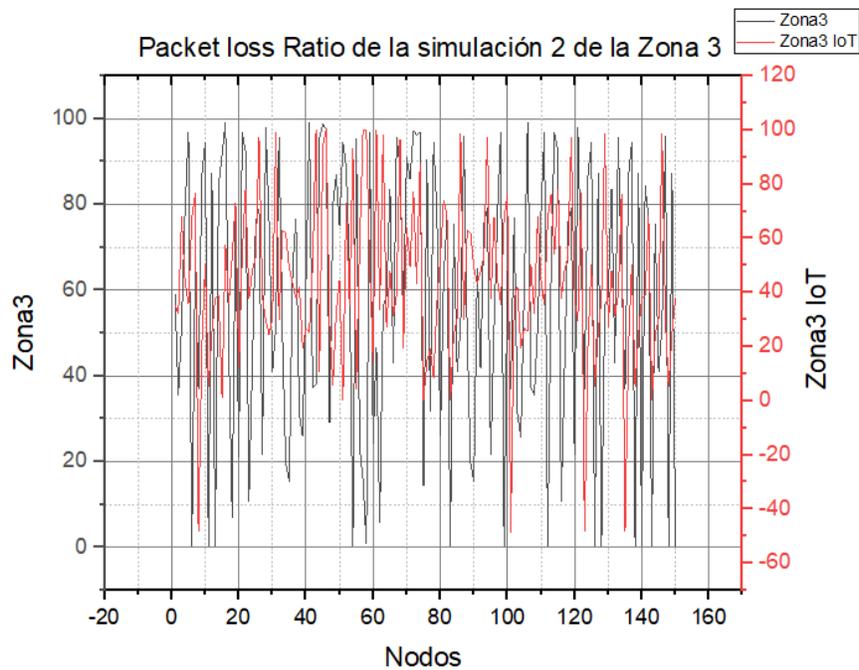


Figura 112: Packet Loss Ratio de la segunda simulación para la Zona 3
Fuente: Autor. Generado por medio del programa Origin 2020

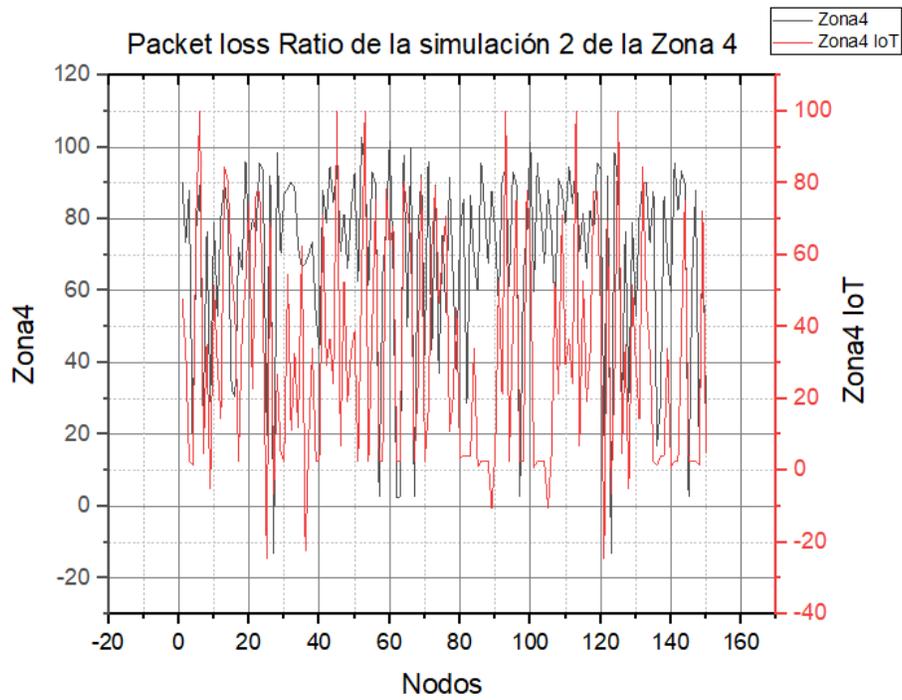


Figura 113: Packet Loss Ratio de la segunda simulación para la Zona 4
Fuente: Autor. Generado por medio del programa Origin 2020

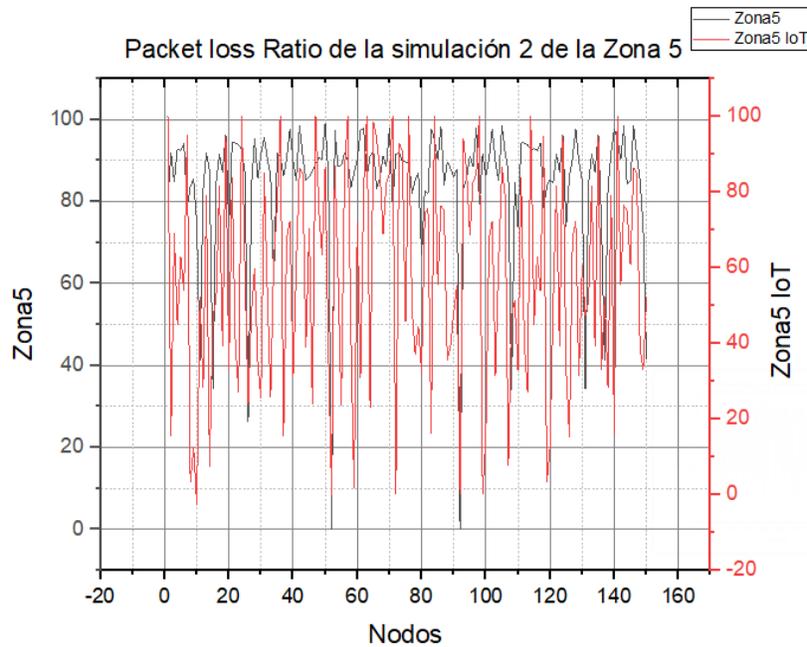


Figura 114: Packet Loss Ratio de la segunda simulación para la Zona 5
Fuente: Autor. Generado por medio del programa Origin 2020

De estos resultados se evidencia nuevamente que hay menor tasa de pérdida de paquetes en las simulaciones de red Vanet interconectada con IoT, ya que en todas las zonas se obtuvo un valor inferior al presentado en la red Vanet tradicional.