

Escuela Superior Politécnica del Litoral
Facultad de Ingeniería en Electricidad y Computación

Evaluación de recursos radio en una comunicación vehículo a
infraestructura usando la red móvil 5G

INGE-2861

Proyecto Integrador

Previo la obtención del Título de:

Ingeniería en Telecomunicaciones

Presentado por:

Nicolás Antonio Segovia Estrella

Darwin Anthony Alvarado Medina

Guayaquil – Ecuador

Año: 2025

Dedicatoria

A mis padres, quienes con su amor y esfuerzo me han guiado en cada paso; a mis hermanos, por ser mi motivación constante; a mis amigos del vóley, que siempre estuvieron apoyándome dentro y fuera de la cancha; y, de manera muy especial, a Ana Paula, quien creyó en mí incluso cuando yo mismo dudaba, y me impulsó a seguir adelante.

Darwin Anthony Alvarado Medina

Dedicatoria

A mis incansables padres, Roberto y Anita quienes estuvieron detrás de mí en cada momento, dedicándose a inculcar sus enseñanzas y valores, evidencia de su responsabilidad y amor que tendré conmigo toda mi vida. También a mis hermanos Roberto, Ana Cristina, Valeria y Lourdes, y todos los miembros de mi gran familia, quienes me dan alegría y confianza todos los días.

Nicolás Antonio Segovia Estrella

Agradecimientos

A mis padres, por su apoyo incondicional, su paciencia y por ser el pilar fundamental que me ha permitido alcanzar esta meta; al Ing. Chancay, por su constante disposición para guiarnos y brindarnos soluciones en cada etapa del proyecto; y al Ing. Novillo, por compartir sus valiosos puntos de vista que me ayudaron a analizar e interpretar los resultados con mayor claridad y profundidad.

Darwin Anthony Alvarado Medina

Agradecimientos

A Dios y a mis padres, quienes van siempre al frente de mí en cada paso de mi vida.

A todos mis amigos y compañeros que estuvieron conmigo todos estos años, a aquellos quienes me extendieron la mano y a quienes tomaron de la mía.

Al Dr. Francisco Novillo e Ing. Eduardo Chancay por su guía en el desarrollo y elaboración de este proyecto. También a la Dra. Patricia Chávez por su paciencia y dedicación a nuestro desarrollo profesional en esta importante etapa. A todos mis profesores quienes fueron parte de mi crecimiento académico, por sus clases y su tiempo.

Nicolás Antonio Segovia Estrella

Declaración Expresa

Nosotros, Nicolás Antonio Segovia Estrella y Darwin Anthony Alvarado Medina acordamos y reconocemos que:

La titularidad de los derechos patrimoniales de autor (derechos de autor) del proyecto de graduación corresponderá al autor o autores, sin perjuicio de lo cual la ESPOL recibe en este acto una licencia gratuita de plazo indefinido para el uso no comercial y comercial de la obra con facultad de sublicenciar, incluyendo la autorización para su divulgación, así como para la creación y uso de obras derivadas. En el caso de usos comerciales se respetará el porcentaje de participación en beneficios que corresponda a favor del autor o autores.

La titularidad total y exclusiva sobre los derechos patrimoniales de patente de invención, modelo de utilidad, diseño industrial, secreto industrial, software o información no divulgada que corresponda o pueda corresponder respecto de cualquier investigación, desarrollo tecnológico o invención realizada por mí/nosotros durante el desarrollo del proyecto de graduación, pertenecerán de forma total, exclusiva e indivisible a la ESPOL, sin perjuicio del porcentaje que me/nos corresponda de los beneficios económicos que la ESPOL reciba por la explotación de mi/nuestra innovación, de ser el caso.

En los casos donde la Oficina de Transferencia de Resultados de Investigación (OTRI) de la ESPOL comunique los autores que existe una innovación potencialmente patentable sobre los resultados del proyecto de graduación, no se realizará publicación o divulgación alguna, sin la autorización expresa y previa de la ESPOL.

Guayaquil, 7 de Septiembre del 2025.

Nicolás Antonio Segovia Estrella

Darwin Anthony Alvarado Medina

Evaluadores

Patricia Ximena Chávez Burbano

Profesor de Materia

Francisco Vicente Novillo

Parales

Tutor de proyecto

Resumen

El trabajo se orientó al estudio de las comunicaciones vehiculares en el Malecón 2000 de Guayaquil, analizando el comportamiento de la transición de redes 4G hacia 5G y el aporte de las capacidades de MEC. El objetivo planteado consistió en determinar en qué medida la incorporación de nodos MEC permite reducir la latencia y mejorar la confiabilidad en entornos de alta movilidad, partiendo de la hipótesis de que su distribución estratégica favorece la eficiencia de transmisión en aplicaciones V2X. Para el desarrollo se recurrió al simulador OMNeT++ y al entorno Simu5G, mediante el diseño de tres escenarios: una red 4G como referencia, una red 5G con un MEC por operador y una red 5G con múltiples MEC distribuidos. Se efectuaron 100 simulaciones en cada escenario con 30 usuarios móviles, evaluando indicadores como la latencia y el throughput. Los resultados evidenciaron que el escenario 5G con varios MEC superó ampliamente a la red 4G y al 5G con un único MEC, mostrando menores retardos y un desempeño más estable. Se concluye que la implementación de MEC en 5G constituye una alternativa eficaz para optimizar la conectividad vehicular en contextos urbanos.

Palabras clave: Simulación, 5G, MEC, Comunicaciones vehiculares

Abstract

This study focused on vehicular communications in Guayaquil's Malecón 2000, examining the transition from 4G to 5G networks and the contribution of Multi-access Edge Computing (MEC). The main objective was to determine how the deployment of MEC nodes could reduce latency and improve reliability in high-mobility environments, based on the hypothesis that their distributed configuration enhances data transmission efficiency for V2X applications. The project was developed using the OMNeT++ simulator with the Simu5G framework, through three designed scenarios: a 4G reference network, a 5G network with one MEC per operator, and a 5G network with several distributed MECs. For each case, 100 simulations with 30 mobile users were conducted, analyzing key performance indicators such as latency and throughput. The findings revealed that the 5G scenario with multiple MECs clearly outperformed both 4G and 5G with a single MEC, achieving lower delays and more consistent performance. It is concluded that the integration of MEC into 5G networks is an effective solution to enhance vehicular connectivity in urban areas.

Keywords: Simulation, 5G, MEC, Vehicular communications

Índice general

Resumen	1
<i>Abstract</i>	2
Índice general	3
Abreviaturas	5
Simbología	8
Índice de figuras	9
Índice de tablas	11
Capítulo 1	1
1. Introducción	2
1.1 Descripción del problema	3
1.2 Justificación del Problema	4
1.3 Objetivos	5
1.3.1 Objetivo general	6
1.3.2 Objetivos específicos	6
1.4 Marco teórico	6
1.4.1 Arquitectura de la red 5G según 3GPP	7
1.4.2 Comunicaciones Vehículo-a-Infraestructura (V2I) y ETSI ITS-G5	8
1.4.3 Bandas de frecuencia y asignación de recursos en comunicaciones V2X	9
1.4.4 Elementos y herramienta a usar en las simulaciones	11
1.4.5 Contexto ecuatoriano y necesidad del estudio	12
Capítulo 2	13
2. Metodología	14
2.5 Cumplimiento de normativas y principios técnicos	17
2.7 Consideraciones éticas y legales	18
2.8 Identificación y análisis causal	18

2.9 Escenarios finales para las simulaciones	19
Capítulo 3.....	24
3. Resultados y análisis	25
 3.1 Resultados del escenario 4G	25
 3.1.1 Análisis de métricas.....	26
 3.2 Resultados del escenario 5G	29
 3.2.1 Análisis de métricas	30
 3.2.1.1 Single MEC.....	30
 3.2.1.2 MultiMec	34
 3.3 Análisis de las métricas con gráficas de función de distribución acumulada (CDF)	37
 3.3.1 Latencia Uplink.....	38
 3.3.2 Latencia Downlink	39
 3.3.3 Throughput Uplink	40
 3.3.4 Throughput Downlink.....	41
Capítulo 4.....	43
4. Conclusiones y Recomendaciones	44
 4.1 Conclusiones	44
 4.2 Recomendaciones.....	45
Referencias.....	47
Anexos	49

Abreviaturas

3GPP	3rd Generation Partnership Project (Proyecto de Asociación de Tercera Generación)
5GC	5G Core (Núcleo de la red 5G)
AMF	Access and Mobility Management Function (Función de gestión de acceso y movilidad)
BLER	Block Error Rate (Tasa de error por bloque)
CDF	Cumulative Distribution Function (Función de distribución acumulada)
CNT	Corporación Nacional de Telecomunicaciones
CQI	Channel Quality Indicator (Indicador de calidad del canal)
DSRC	Dedicated Short-Range Communications (Comunicaciones de corto alcance dedicadas)
eMBB	Enhanced Mobile Broadband (Banda ancha móvil mejorada)
eNodeB	Evolved Node B (Nodo B evolucionado)
ETSI	European Telecommunications Standards Institute (Instituto Europeo de Normas de Telecomunicaciones)
FCC	Federal Communications Commission (Comisión Federal de Comunicaciones, EE. UU.)
FR1	Frequency Range 1 (Rango de frecuencias 1 en 5G)
GICOM	Grupo de Investigación en Comunicaciones Móviles
gNodeB	Next Generation Node B (Nodo B de próxima generación)

ITU-R	International Telecommunication Union - Radiocommunication Sector (Unión Internacional de Telecomunicaciones – Sector de Radiocomunicaciones)
ITS-G5	Intelligent Transportation Systems G5 (Sistema de transporte inteligente G5)
MEC	Multi-Access Edge Computing (Computación perimetral de acceso múltiple)
mMTC	Massive Machine-Type Communications (Comunicaciones masivas de tipo máquina)
NFV	Network Functions Virtualization (Virtualización de funciones de red)
ODS	Objetivos de Desarrollo Sostenible
OMNeT++	Objective Modular Network Testbed in C++ (Simulador de redes)
PGW	Packet Data Network Gateway (Puerta de enlace a la red de paquetes de datos)
RAN	Radio Access Network (Red de acceso por radio)
RB	Resource Block (Bloque de recursos)
SACK	Selective Acknowledgement (Reconocimiento selectivo en TCP)
SCTP	Stream Control Transmission Protocol (Protocolo de transmisión de flujo de mensajes)
SIMU5G	(Framework de simulación 5G en OMNeT++)
SMF	Session Management Function (Función de gestión de sesiones)
SUPERTEL	Superintendencia de Telecomunicaciones (Ecuador)
UDP	User Datagram Protocol (Protocolo de datagramas de usuario)
UE	User Equipment (Equipo de usuario)

UPF	User Plane Function (Función del plano de usuario)
URLLC	Ultra-Reliable Low Latency Communications (Comunicaciones ultra confiables y de baja latencia)
V2I	Vehicle-to-Infrastructure (Vehículo a infraestructura)
V2X	Vehicle-to-Everything (Vehículo a todo)

Simbología

Bps Bits por segundo

dBm Decibelio-milivatio

ms Milisegundo

m Metro

s Segundo

m/s Metros por segundo

Índice de figuras

Ilustración 1.4.3.1. Diagrama de la Metodología del Proyecto	10
Ilustración 2.1. Diagrama de la Metodología del Proyecto	17
Ilustración 2.2. Diagrama de la Solución Seleccionada	19
Ilustración 2.3. Escenario 1 – Red 4G (referencia comparativa)	20
Ilustración 2.4. Escenario 2– Red 5G con un solo MEC	22
Ilustración 2.5. Escenario 3 – Red 5G con varios MEC	25
Ilustración 3.1. Gráfica promedio de la latencia UL del recorrido de los UE	26
Ilustración 3.2. Gráfica general de la latencia DL del recorrido de los UE.....	27
Ilustración 3.3. Gráfica general del throughput DL del recorrido de los UE.....	28
Ilustración 3.4. Gráfica general del throughput UL del recorrido de los UE.....	29
Ilustración 3.5. Gráfica general del Delay UL del recorrido de los UE.....	30
Ilustración 3.6. Gráfica general del Delay DL del recorrido de los UE.....	31
Ilustración 3.7. Gráfica general del Throughput UL del recorrido de los UE	32
Ilustración 3.8. Gráfica general del Throughput UL del recorrido de los UE	33
Ilustración 3.9. Gráfica general del Delay UL del recorrido de los UE.....	34
Ilustración 3.10. Gráfica general del Delay DL del recorrido de los UE.....	35
Ilustración 3.11. Gráfica general del Throughput UL del recorrido de los UE	36
Ilustración 3.12. Gráfica general del Throughput UL del recorrido de los UE	37
Ilustración 3.13. Gráfica CDF del Delay UL de los 3 escenarios simulados.....	38
Ilustración 3.14. Gráfica CDF del Delay DL de los 3 escenarios simulados.....	39

Ilustración 3.15. Gráfica CDF del Throughput UL de los 3 escenarios simulados 40

Ilustración 3.16. Gráfica CDF del Throughput DL de los 3 escenarios simulados 41

Índice de tablas

Tabla 2.1. Tabla de configuración de los parámetros en el escenario 4G.....	21
Tabla 2.2. Tabla de configuración de los parámetros en el escenario 5G SingleMEC.....	23
Tabla 2.3. Tabla de configuración de los parámetros en el escenario 5G MultiMEC	26

Capítulo 1

1. Introducción

Las tecnologías de comunicación inalámbrica han evolucionado considerablemente desde la introducción de la primera generación de redes móviles. Cada nueva generación ha transformado la forma en que las personas y los dispositivos interactúan con el mundo digital. En este marco de innovación, la tecnología 5G se posiciona como un catalizador esencial para el desarrollo de ciudades inteligentes, la automatización industrial y, especialmente, la movilidad vehicular avanzada. La arquitectura definida por el Proyecto de Asociación de Tercera Generación (3rd Generation Partnership Project - 3GPP) establece una red flexible y escalable, diseñada para cubrir tres casos de uso principales: eMBB (banda ancha móvil mejorada), URLLC (comunicaciones ultra fiables y de baja latencia) y mMTC (comunicaciones masivas tipo máquina) (3GPP, 2022).

En el dominio del transporte, la integración de 5G con los sistemas vehiculares ha abierto el camino a nuevas formas de comunicación que priorizan la seguridad y la eficiencia. Las comunicaciones vehículo-a-infraestructura (V2I), como parte del ecosistema V2X, permiten que los vehículos interactúen en tiempo real con elementos del entorno urbano. Esto se logra mediante el uso de tecnologías como ITS-G5, promovida por el Instituto Europeo de Normas de Telecomunicaciones (European Telecommunications Standards Institute - ETSI), que define una arquitectura de red vehicular basada en mensajes de corto alcance (DSRC) diseñados para advertencias de colisión, condiciones de tráfico o gestión de semáforos (ETSI, 2021).

Sin embargo, Ecuador se encuentra rezagado en la implementación de estas tecnologías. A diferencia de países como Chile, Brasil o Colombia, donde ya se han desplegado pilotos de 5G, en Ecuador aún no existe una red comercial activa ni estudios aplicados que evalúen su comportamiento en condiciones urbanas reales. La topografía diversa, la densidad poblacional

heterogénea y las limitaciones en infraestructura suponen desafíos técnicos considerables para la implementación efectiva de esta tecnología.

Este proyecto se propone cerrar esa brecha mediante el diseño y simulación de un entorno urbano representativo, con el objetivo de analizar la viabilidad técnica del despliegue de redes 5G para aplicaciones V2I en el país. El estudio se centrará en el centro de Guayaquil, una de las zonas con mayor densidad vehicular, utilizando simuladores reconocidos como OMNeT++ y Simu5G. La información obtenida permitirá evaluar la calidad de conexión, los tiempos de latencia y otros indicadores clave para determinar si el entorno urbano ecuatoriano está preparado para la transición hacia tecnologías de movilidad inteligente.

1.1 Descripción del problema

En Ecuador, la implementación de la tecnología 5G todavía no se ha materializado, lo que contrasta con el avance de otros países de América Latina que ya han adoptado esta tecnología. Este retraso se atribuye a la falta de estudios detallados que evalúen la funcionalidad de los diseños de red propuestos en el contexto ecuatoriano. En particular, se requiere validar cómo estas redes operan bajo condiciones locales específicas, como la topografía compleja del país, las diferencias en densidad poblacional y la variabilidad en las velocidades de los usuarios en movimiento.

El problema es actual e importante, ya que la ausencia de redes 5G afecta directamente tanto a los ciudadanos como a las operadoras de telecomunicaciones. Para los usuarios finales, la implementación de 5G representa mejoras en velocidad de conexión, acceso a servicios innovadores y una mayor capacidad de integración con tecnologías emergentes. Por otro lado, para las empresas de telecomunicaciones, como Telconet Cia. Ltda. (una organización reconocida en el mercado nacional), la falta de una red 5G limita su competitividad y capacidad de ofrecer servicios de última generación.

Para abordar este problema, es esencial identificar los requisitos, restricciones y variables de interés asociados al diseño y despliegue de redes 5G en el país. Esto incluye la evaluación de parámetros técnicos como la capacidad de transmisión, la latencia, la cobertura en zonas urbanas y rurales, y el impacto de las condiciones de movilidad en el rendimiento de la red. Además, es necesario considerar factores económicos, regulatorios y sociales que influyen en la implementación de esta tecnología.

La propuesta de solución implica diseñar y validar un escenario funcional mediante simulaciones avanzadas y pruebas controladas que reflejen las condiciones reales de uso en Ecuador. Estas simulaciones permitirán medir, analizar y ajustar los parámetros clave de la red, asegurando su viabilidad técnica y económica. Los resultados de este estudio no solo facilitarán la toma de decisiones informadas para iniciar la implementación de la red 5G, sino que también proporcionarán una base sólida para el desarrollo tecnológico del país, promoviendo la innovación y el crecimiento socioeconómico.

1.2 Justificación del Problema

La transformación digital de las ciudades no puede ser concebida sin una infraestructura de telecomunicaciones sólida y moderna. En este sentido, el despliegue de redes 5G constituye uno de los pasos más relevantes hacia el establecimiento de entornos urbanos inteligentes y sostenibles. Más allá de proporcionar velocidades de conexión más altas, el verdadero potencial del 5G radica en su capacidad para soportar aplicaciones de misión crítica, como los sistemas de transporte autónomo y las comunicaciones vehiculares V2I. Estas aplicaciones requieren niveles muy bajos de latencia, alta confiabilidad y la capacidad de conectar a miles de dispositivos en tiempo real, condiciones que no pueden ser satisfechas de manera óptima por redes 4G (3GPP, 2019).

De acuerdo con ETSI (2021), las comunicaciones vehículo-a-infraestructura (V2I), cuando se integran con redes 5G mediante arquitecturas como ITS-G5, permiten una

transmisión de datos crítica entre vehículos y semáforos, señales inteligentes y centros de control urbano. Esta conectividad resulta fundamental para reducir el número de accidentes, mejorar la respuesta ante emergencias y aumentar la eficiencia energética del transporte. Por ejemplo, estudios en Alemania y Corea del Sur han demostrado que el uso de ITS-G5 sobre 5G reduce los tiempos de reacción en un 40 % y mejora el rendimiento del tráfico urbano hasta en un 30 % (Müller et al., 2020; Lee & Park, 2021).

En el caso ecuatoriano, la ausencia de estudios de validación en contextos reales limita tanto la inversión de las operadoras de telecomunicaciones como la planificación estratégica de las autoridades de tránsito y seguridad. Actualmente, la mayoría de decisiones en esta materia se basan en estándares internacionales o pruebas en laboratorios aislados, sin una contextualización adecuada al entorno nacional. Es por ello que este proyecto propone una metodología basada en simulaciones técnicas contextualizadas en Guayaquil, ciudad que representa uno de los mayores retos de movilidad del país.

La importancia de este estudio radica en su potencial para generar datos empíricos que respalden decisiones de infraestructura y política pública. Además, los resultados obtenidos podrían ser utilizados por operadores como Telconet o CNT para diseñar sus despliegues de red, garantizando un servicio eficiente y seguro para aplicaciones vehiculares. Finalmente, la integración de 5G y V2I contribuirá al cumplimiento de objetivos de desarrollo sostenible (ODS), especialmente el ODS 9: Industria, innovación e infraestructura, promoviendo un ecosistema digital más resiliente y competitivo en Ecuador.

1.3 Objetivos

El presente proyecto se centra en abordar las necesidades tecnológicas y sociales derivadas de la transición hacia redes móviles 5G, con un enfoque particular en las comunicaciones vehículo-a-infraestructura (V2I). La investigación busca responder a

problemáticas específicas del contexto ecuatoriano, garantizando que las soluciones planteadas sean funcionales y efectivas en escenarios locales.

Para ello, los objetivos se han diseñado con el propósito de guiar cada etapa del proyecto, desde el diseño del escenario de simulación hasta el análisis detallado de los datos obtenidos, asegurando un enfoque integral que permita alcanzar los resultados esperados.

1.3.1 Objetivo general

- Diseñar y comparar tres escenarios de simulación urbano con alta densidad de vehículos autónomos que usan la red 5G y 4G para establecer sus comunicaciones.

1.3.2 Objetivos específicos

- Crear tres escenarios de simulación funcionales con varios UE para comparar el comportamiento tanto de la red 4G y 5G.
- Configurar el sistema para integrar capacidades MEC que permitan la comunicación vehicular eficiente, sentando las bases para aplicaciones avanzadas como vehículos autónomos.
- Obtener data de los UE con información acerca del desempeño de la conexión que tuvieron con la red, considerando parámetros clave como latencia y throughput.

1.4 Marco teórico

Las redes móviles han pasado por un proceso de evolución continua desde 1G hasta la actual quinta generación (5G). A diferencia de 4G, que se centraba en el acceso a banda ancha móvil, 5G introduce mejoras significativas en velocidad de transmisión, baja latencia, alta confiabilidad y capacidad de conectar millones de dispositivos simultáneamente (3GPP, 2022). Estas características permiten habilitar casos de uso avanzados como vehículos autónomos, cirugías remotas y ciudades inteligentes.

La arquitectura de 5G definida por el Proyecto de Asociación de Tercera Generación (3rd Generation Partnership Project - 3GPP) se basa en la separación de la red de acceso (RAN) y el núcleo 5G Core (5GC). En el dominio de acceso, las estaciones base se conocen como gNodeB (Next Generation Node B - Nodo B de próxima generación), mientras que en el núcleo se definen funciones como el AMF (Access and Mobility Management Function - Función de gestión de acceso y movilidad), el SMF (Session Management Function - Función de gestión de sesiones) y el UPF (User Plane Function - Función del plano de usuario), todas virtualizadas y escalables (3GPP, 2019).

1.4.1 Arquitectura de la red 5G según 3GPP

La red 5G opera sobre un modelo de comunicación definido por tres pilares técnicos. El primero es eMBB (Enhanced Mobile Broadband - Banda ancha móvil mejorada), que proporciona mayor capacidad de transmisión y velocidad, ideal para aplicaciones de alto consumo de datos como video en 4K o realidad aumentada. El segundo es URLLC (Ultra-Reliable Low Latency Communications - Comunicaciones ultra confiables de baja latencia), diseñado para aplicaciones críticas que requieren latencia extremadamente baja y alta confiabilidad, como la conducción autónoma o la automatización industrial. Finalmente, mMTC (Massive Machine-Type Communications - Comunicaciones masivas de tipo máquina) permite la conexión de una gran cantidad de dispositivos simultáneamente, como sensores IoT desplegados en ciudades inteligentes o sectores agrícolas (3GPP, 2022).

La arquitectura definida por el 3GPP también incluye la función de red virtualizada (NFV) y el uso de *network slicing*, lo cual permite a los operadores segmentar la red en “rebanadas” adaptadas a diferentes necesidades de usuarios y servicios. Esto resulta fundamental en entornos urbanos complejos, donde coexistirán múltiples servicios como vehículos autónomos, transmisión de video, servicios de emergencia y más (5G Americas, 2022).

Una de las innovaciones más relevantes de 5G es la incorporación de Computación perimetral de acceso múltiple (Multi-Access Edge Computing - MEC). Esta tecnología permite procesar datos directamente en el borde de la red, sin necesidad de enviarlos al núcleo o a servidores remotos en la nube, lo que reduce drásticamente la latencia. El MEC Host aloja aplicaciones sensibles al tiempo, y su gestión está a cargo del MEC Orchestrator, que asigna recursos y balancea la carga entre nodos.

En aplicaciones vehiculares, este esquema es clave porque soporta comunicaciones de misión crítica como V2I (Vehicle-to-Infrastructure – Vehículo a infraestructura), donde los vehículos interactúan en tiempo real con semáforos inteligentes, señales de tránsito y centros de control urbano (ETSI, 2021).

1.4.2 Comunicaciones Vehículo-a-Infraestructura (V2I) y ETSI ITS-G5

Las comunicaciones V2I (Vehicle-to-Infrastructure – Vehículo a infraestructura) constituyen una subcategoría dentro del ecosistema V2X (Vehicle-to-Everything – Vehículo a todo). Estas comunicaciones permiten el intercambio de información entre los vehículos y elementos de la infraestructura urbana, como semáforos, centros de control, estaciones base y señales de tráfico. El estándar ETSI ITS-G5, promovido por el Instituto Europeo de Normas de Telecomunicaciones (European Telecommunications Standards Institute - ETSI), define un conjunto de protocolos y servicios para permitir este tipo de interacción segura y eficiente, garantizando la interoperabilidad entre dispositivos y proveedores (ETSI, 2021).

La tecnología ETSI ITS-G5 está diseñada para operar en bandas dedicadas al tráfico vehicular, utilizando mecanismos de comunicación directos (DSRC) que permiten a los vehículos enviar y recibir información crítica con tiempos de respuesta mínimos. Esta arquitectura se complementa con redes móviles como 5G para extender el alcance y aumentar la capacidad de procesamiento mediante MEC. La combinación de 5G con ITS-G5 se perfila

como una solución robusta para el despliegue de sistemas inteligentes de transporte, especialmente en zonas urbanas de alta densidad vehicular.

Investigaciones realizadas en Alemania y Corea del Sur han demostrado que la integración de ITS-G5 con redes 5G mejora la seguridad vial y optimiza la gestión del tráfico, reduciendo los tiempos de respuesta ante emergencias y disminuyendo la congestión vehicular (Müller et al., 2020; Lee & Park, 2021). Las aplicaciones incluyen sistemas de advertencia de colisión, gestión dinámica de semáforos y soporte para vehículos autónomos.

1.4.3 Bandas de frecuencia y asignación de recursos en comunicaciones V2X

En el contexto de las comunicaciones V2X (Vehicle-to-Everything), la asignación de frecuencias constituye un factor esencial para garantizar baja latencia y alta confiabilidad en los enlaces. A nivel internacional, se han destinado diferentes bandas para este tipo de aplicaciones, destacando la banda de 5.9 GHz, identificada como espectro para ITS (Intelligent Transportation Systems – Sistemas de Transporte Inteligente) (ITU-R, 2017), y la banda de 4.9 GHz, utilizada de manera complementaria en ciertos despliegues de 5G NR (FCC, 2020); ambas se encuentran dentro del FR1 (Frequency Range 1 – Rango 1 de Frecuencia), definido entre 410 MHz y 7.125 GHz (3GPP, 2018). En la simulación realizada se emplearon dos portadoras en agregación: CC0 en 5.9 GHz con 50 bloques de recursos (Resource Blocks - RBs) asignados al Operador A y CC1 en 4.9 GHz con 50 RBs asignados al Operador B. La asignación de RBs se basó en un ancho de banda de 10 MHz bajo una numerología $\mu = 0$, equivalente a un espaciamiento de 15 kHz entre subportadoras dentro de cada RB (3GPP, 2018; 3GPP, 2020), lo que permitió modelar de manera precisa la capacidad de transmisión y la eficiencia espectral. Asimismo, se fijó la potencia de transmisión en 23 dBm para los UEs y en 46 dBm para las estaciones base (gNBs), definiendo además una tasa de error de bloque (Block Error Rate - BLER) objetivo de 0.01 con retroalimentación del Indicador de calidad del canal (Channel Quality Indicator - CQI) cada 40 ms. Tanto el

handover dinámico como la interconexión X2 entre UEs y estaciones base fueron habilitados, lo que permitió representar de manera realista escenarios de movilidad vehicular. En conjunto, estas configuraciones aseguraron una caracterización sólida del entorno radioeléctrico, constituyendo la base para comparar el desempeño entre redes 4G LTE y 5G NR con capacidades MEC.

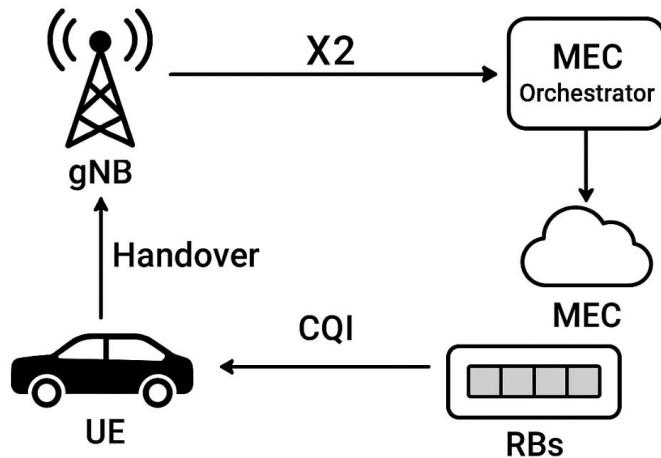


Ilustración 1.4.3.1. Diagrama de la Metodología del Proyecto

El gráfico muestra cómo funcionan los elementos principales de una red 5G aplicada a la comunicación vehicular. El vehículo (UE) se conecta a una antena (gNB) y le envía información sobre la calidad de la señal, conocida como CQI. Gracias a este dato, la red puede ajustar automáticamente los recursos y la modulación según las condiciones del canal. Cuando el vehículo se mueve, la conexión pasa de una antena a otra mediante el proceso de handover, que se coordina a través del enlace X2 entre estaciones base. Dentro de cada antena, los RBs son las unidades básicas de espectro que permiten asignar capacidad de transmisión a los usuarios. Al mismo tiempo, el MEC acerca el poder de cómputo al borde de la red, mientras que el MEC Orchestrator organiza y distribuye esas tareas para reducir la latencia y mejorar la confiabilidad del sistema. En conjunto, todos estos componentes trabajan de forma integrada para lograr una comunicación vehicular más rápida, estable y eficiente.

1.4.4 Elementos y herramienta a usar en las simulaciones

En el escenario inicial basado en la red 4G, se emplean componentes que forman parte de la arquitectura tradicional LTE. El servidor cumple la función de proveer servicios y aplicaciones a los usuarios finales, desde navegación web hasta plataformas de comunicación. El *router* actúa como un dispositivo intermediario, encargado de interconectar la red central con *Internet* y otros sistemas externos, garantizando el enrutamiento adecuado de los datos. La red de paquetes de datos (Packet Data Network Gateway - PGW) desempeña un papel esencial dentro del núcleo EPC (Evolved Packet Core - Núcleo de paquetes evolucionado), ya que se encarga de asignar direcciones IP a los dispositivos de usuario y de gestionar el tráfico hacia las redes externas. Finalmente, los eNodeB, que corresponden a las estaciones base LTE, permiten la conexión inalámbrica entre los usuarios (UE) y la infraestructura de la red, facilitando la transmisión y recepción de datos en la interfaz aérea. Esta configuración refleja cómo operaban las redes 4G, donde el procesamiento estaba centralizado en el núcleo, generando limitaciones de latencia para aplicaciones que requieren tiempos de respuesta mínimos.

En el segundo escenario, correspondiente a la red 5G con capacidades de MEC los elementos se diversifican para ofrecer un mejor desempeño en aplicaciones críticas. Los gNB (Next Generation Node B – Nodo B de siguiente generación) representan las estaciones base 5G que conectan a los dispositivos de usuario con la red, proporcionando mayor capacidad de transmisión y una latencia reducida respecto a LTE. A estos gNB se conectan los *User Plane Functions* (UPF), que constituyen el plano de usuario en el núcleo 5G y se encargan de enrutar y procesar los datos. Una de las innovaciones centrales es el despliegue de MEC *Host*, servidores locales que procesan información directamente en el borde de la red sin necesidad de enviarla a centros de datos lejanos. Esto reduce significativamente la latencia y habilita la ejecución de aplicaciones en tiempo real. Por encima de estos, los MEC *Orchestrators* cumplen

la tarea de gestionar y coordinar los recursos entre los distintos servidores MEC, asegurando un balance adecuado de carga y disponibilidad de servicios. En conjunto, estos elementos permiten a la red 5G no solo ofrecer un mayor ancho de banda, sino también responder a los requerimientos de comunicaciones vehiculares eficientes y confiables.

Finalmente, el tercer escenario plantea una ampliación de la red 5G con MEC, en la que se incluyen cuatro MEC *Host* por cada segmento de la red. Esta expansión tiene como objetivo incrementar la capacidad de procesamiento local y mejorar la resiliencia frente a una mayor demanda de servicios. Al contar con más servidores en el borde, la red puede distribuir mejor las cargas de trabajo, ofreciendo un soporte más robusto a un mayor número de aplicaciones simultáneamente. Este diseño es particularmente relevante en entornos urbanos densos, como el Malecón 2000 en Guayaquil, donde coexisten un alto flujo vehicular y peatonal con múltiples servicios que requieren conectividad en tiempo real. La ampliación de recursos en el borde demuestra cómo el escalamiento de infraestructura MEC puede optimizar las comunicaciones V2I, garantizando estabilidad y confiabilidad incluso en escenarios de alta congestión.

Cabe destacar que para la elaboración de estos tres escenarios se empleará el simulador OMNeT++, una herramienta de simulación de redes ampliamente utilizada en el ámbito académico y de investigación. OMNeT++ permitirá modelar tanto la arquitectura de la red como el comportamiento del tráfico vehicular y de los dispositivos conectados, brindando resultados precisos en cuanto a métricas como latencia y throughput. Gracias a su flexibilidad, este simulador se convierte en la base metodológica del proyecto, garantizando que los escenarios diseñados reflejen de forma realista el funcionamiento esperado de la red 5G en el Malecón 2000.

1.4.5 Contexto ecuatoriano y necesidad del estudio

En el caso de Ecuador, los estudios y despliegues tecnológicos se han concentrado históricamente en redes 3G y 4G. La falta de investigación aplicada a 5G, especialmente en el

ámbito de movilidad urbana, ha limitado la capacidad del país para adoptar tecnologías de transporte inteligente. Según la Superintendencia de Telecomunicaciones (SUPERTEL, 2022), existe un alto interés institucional por acelerar esta transición, pero aún persisten brechas técnicas, económicas y regulatorias que deben ser abordadas con estudios como el presente.

Este proyecto contribuye a reducir estas brechas mediante la creación de escenarios de simulación contextualizados, capaces de evaluar cómo responderían las redes 5G ante las condiciones de tráfico urbano de ciudades como Guayaquil. Además, los resultados generados permitirán a entidades como Telconet, CNT o GICOM tomar decisiones estratégicas fundamentadas en evidencia técnica y estadística.

Capítulo 2

2. Metodología

La metodología empleada en este proyecto se estructuró con base en la arquitectura definida por 3GPP para redes 5G y las especificaciones ITS-G5 para comunicaciones V2I. Se plantearon tres alternativas para el desarrollo del entorno de simulación: una simulación genérica sin contexto local, una segunda alternativa con integración de movilidad real de Guayaquil y una tercera con soporte de MEC, optimizando los tiempos de respuesta del sistema.

Tras comparar su viabilidad, se seleccionó la tercera opción por su mayor precisión para replicar escenarios críticos. Se implementó una arquitectura compuesta por dispositivos UE, estaciones gNodeB y nodos MEC, siguiendo la separación funcional y virtualización de funciones de red propuestas por 3GPP. Para las comunicaciones vehiculares, se consideró la pila de protocolos definida por ETSI ITS-G5, simulando mensajes de advertencia (como CAM o DENM) entre vehículos y la infraestructura.

El escenario seleccionado fue la calle Simón Bolívar, la cual es la avenida que se encuentra el Malecón 2000, por su densidad vehicular y facilidad de obtención de datos. Las herramientas OMNeT++ y Simu5G permitieron configurar la movilidad y analizar métricas como latencia y throughput. Los datos simulados fueron recolectados y analizados para determinar el grado de preparación del país en relación con la infraestructura tecnológica necesaria para el despliegue de redes 5G vehiculares. Se consideraron también los principios éticos y legales, asegurando el uso de datos no personales y respetando la normativa de telecomunicaciones vigente.

2.1 Formulación de alternativas de solución

En la fase inicial, se plantearon tres alternativas principales para abordar la problemática identificada. La primera consistió en el uso de escenarios simulados basados únicamente en datos genéricos de movilidad y tráfico urbano, utilizando parámetros estándar definidos por la

literatura internacional y sin considerar las particularidades locales. Esta alternativa presentaba la ventaja de ser rápida y económica, pero su precisión era limitada debido a la falta de personalización para las condiciones ecuatorianas.

La segunda alternativa integró el modelado de condiciones locales específicas, incluyendo densidad vehicular, topografía urbana y patrones de tráfico típicos de ciudades como Guayaquil. Este enfoque permitía un análisis más representativo, aunque requería mayor tiempo y recursos para la recolección y procesamiento de datos.

La tercera alternativa incorporó simulaciones avanzadas que incluyeron soporte de tecnologías MEC, un enfoque innovador que permite procesar datos más cerca del usuario final. Esta solución no solo optimizaba la latencia y la capacidad de transmisión, sino que también ofrecía un marco más robusto para evaluar el rendimiento en condiciones complejas, como el tráfico vehicular denso y los patrones de movilidad variable. Este último enfoque, aunque más demandante en términos de recursos y complejidad técnica, fue seleccionado por su capacidad para ofrecer una representación precisa y práctica de las condiciones reales de comunicación en entornos urbanos ecuatorianos.

Descripción y selección de la mejor alternativa

La selección de la tercera alternativa se basó en un análisis comparativo de las ventajas y limitaciones de cada enfoque, utilizando criterios como precisión, viabilidad técnica y relevancia para los objetivos del proyecto. Además de su capacidad para representar de manera más fiel las condiciones locales, la integración de tecnologías MEC en la simulación permitió abordar desafíos clave relacionados con la baja latencia, la confiabilidad y la gestión eficiente de los recursos de red, elementos críticos para las comunicaciones V2I.

2.3 Diseño conceptual

El diseño conceptual del proyecto incluyó la definición de parámetros técnicos y operativos clave para las simulaciones. Entre ellos se encontraban la densidad vehicular, los patrones de movimiento, la topología de red y las métricas de rendimiento como latencia y throughput. El escenario urbano seleccionado fue el centro de la ciudad de Guayaquil, una de las zonas más transitadas y representativas del país, donde la diversidad de condiciones de tráfico y la infraestructura urbana compleja ofrecían un entorno ideal para evaluar la efectividad del diseño de red propuesto.

La selección del Malecón 2000 se justificó por la mayor disponibilidad de datos relacionados con el tráfico y la movilidad en esta área, así como por su relevancia estratégica para futuras implementaciones de redes 5G en el país. Este escenario también permitió simular condiciones de alta densidad vehicular y flujo de tráfico dinámico, lo que representaba un desafío significativo para la evaluación de las capacidades de la red 5G.

2.4 Metodología de diseño y análisis final

Se adoptó una metodología iterativa que incluyó la creación y validación de escenarios mediante herramientas de simulación como OMNet++ y Simu5G. En la primera iteración, se configuraron los elementos básicos de la red, incluyendo la conexión de unidades de usuario (UE) con nodos gNodeB. Este diseño inicial permitió evaluar parámetros fundamentales, como el rango de cobertura y el nivel básico de latencia.

Posteriormente, se integraron capacidades MEC para optimizar la latencia y mejorar el manejo de datos en tiempo real. Este paso fue crucial para simular interacciones avanzadas entre vehículos y elementos de la infraestructura urbana, como semáforos y señales de tráfico. En cada etapa, se realizaron pruebas controladas para ajustar los parámetros técnicos y garantizar que el diseño respondiera de manera efectiva a los objetivos específicos del proyecto.

Una vez recopilados los datos de las simulaciones, se llevó a cabo una etapa final de análisis. Durante esta fase, los resultados se evaluaron para determinar si Ecuador está preparado para implementar redes 5G o si, por el contrario, se requieren mejoras adicionales en áreas como infraestructura, calidad del servicio y capacidad de gestión por parte de las operadoras de telecomunicaciones. Este análisis incluyó la identificación de posibles limitaciones en la infraestructura existente y la propuesta de recomendaciones para superar estos desafíos.

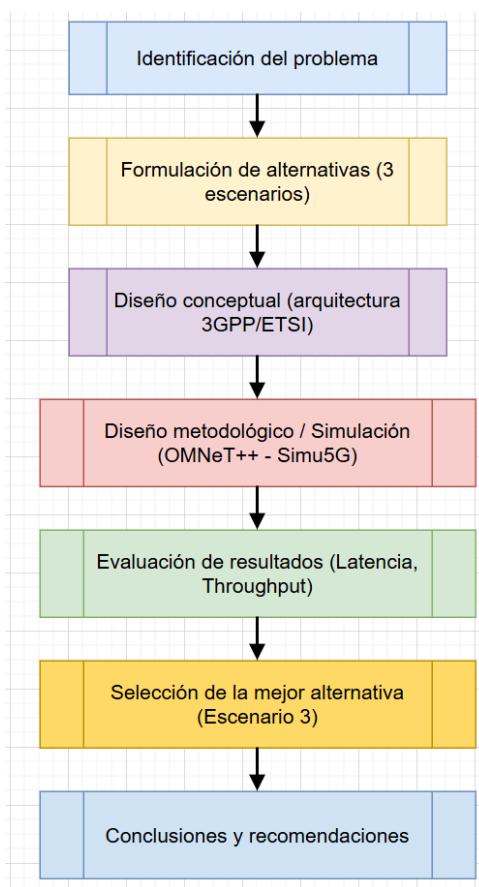


Ilustración 2.1. Diagrama de la Metodología del Proyecto

2.5 Cumplimiento de normativas y principios técnicos

El proyecto se desarrolló en alineación con las normativas internacionales establecidas por organismos como 3GPP (Third Generation Partnership Project) y ETSI (European Telecommunications Standards Institute), que proporcionan estándares técnicos para el despliegue de redes 5G. Estos estándares cubrieron aspectos como la gestión de espectro, el

diseño de antenas y la optimización de recursos radio. La integración de estas normativas garantizó que el diseño propuesto fuera técnicamente sólido y aplicable en un contexto real.

2.6 Especificaciones técnicas del diseño final

El diseño final incluyó una infraestructura de red simulada que integró nodos gNodeB, dispositivos UE y capacidades MEC. Los parámetros técnicos establecidos en las simulaciones incluyeron una latencia promedio menor a 5 ms y un throughput superior a 100 Mbps por dispositivo. Además, las simulaciones se basaron en un modelo realista de tráfico urbano, caracterizado por patrones dinámicos y alta densidad vehicular.

2.7 Consideraciones éticas y legales

El desarrollo del proyecto tomó en cuenta importantes consideraciones éticas y legales. Todos los datos de tráfico y movilidad utilizados en las simulaciones provinieron de fuentes públicas y no involucraron información personal o sensible. Se garantizó el cumplimiento de las normativas nacionales e internacionales relacionadas con la privacidad, la seguridad de los datos y la implementación de redes 5G. Estas medidas aseguraron que el proyecto se desarrollara de manera ética y responsable.

2.8 Identificación y análisis causal

La estrategia de identificación se centró en analizar las relaciones causales entre las configuraciones de la red simulada y el desempeño observado en términos de métricas clave como latencia y throughput. Se utilizaron técnicas estadísticas avanzadas para validar las correlaciones y determinar la viabilidad del despliegue de la red 5G bajo las condiciones específicas del escenario estudiado. Este enfoque permitió identificar patrones de comportamiento en la red y proporcionó información valiosa para futuras implementaciones en Ecuador.

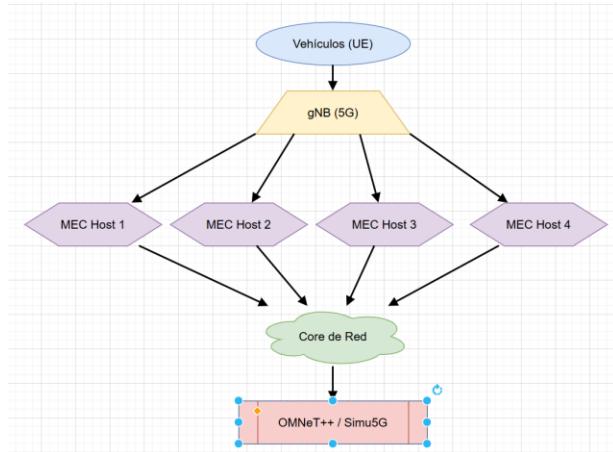


Ilustración 2.2. Diagrama de la Solución Seleccionada

2.9 Escenarios finales para las simulaciones

En las simulaciones se probaron distintos protocolos de transporte para entender cómo afectan el desempeño de las comunicaciones vehiculares, especialmente en métricas como la latencia, el throughput y la tasa de entrega de paquetes. Se utilizó TCP Reno con SACK como referencia en los tres escenarios (4G, 5G con un MEC y 5G con múltiples MEC), ya que es un estándar ampliamente conocido y permite ver cómo la congestión y la recuperación de pérdidas impactan en la red.

UDP, por su parte, se incluyó en todos los escenarios para modelar aplicaciones críticas sensibles al retardo, como los mensajes de seguridad vehicular, mostrando el límite inferior de la latencia al no contar con control de congestión ni retransmisión.

En los escenarios de 5G con MEC, se probó SCTP por su capacidad de manejar múltiples flujos y mantener la sesión activa durante la movilidad, lo que es clave en los handovers. Además, QUIC se añadió como protocolo emergente que combina la velocidad de UDP con la confiabilidad y seguridad de TCP, optimizando la transmisión de datos en entornos de alta movi

Al integrar todos estos protocolos, se pudo comparar soluciones tradicionales con alternativas más modernas, ofreciendo un análisis más completo del comportamiento de la red.

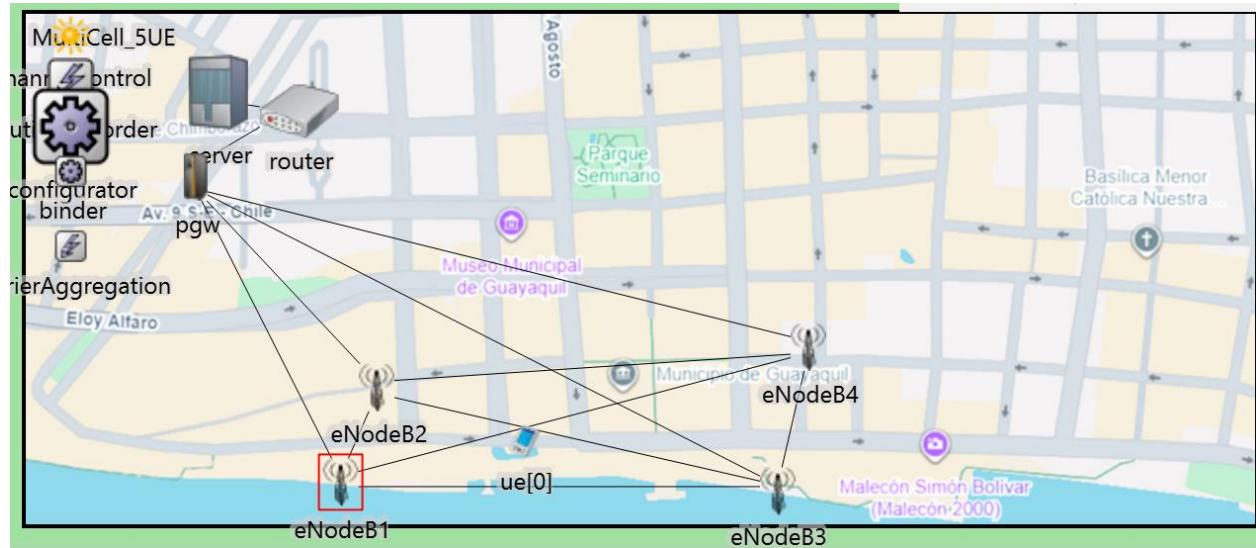


Ilustración 2.3. Escenario 1 – Red 4G (referencia comparativa)

Entidad	Parámetro	Valor
General	Tiempo de simulación	40 s
	Grabación	Escalares y vectores habilitados; result-recording-modes = all
VRU	Nodos eNodeB	eNodeB1–eNodeB4 con enlaces X2 entre pares
	Coordenadas eNodeB1	(350 m, 520 m)
	Coordenadas eNodeB2	(390 m, 415 m)
	Coordenadas eNodeB3	(836 m, 530 m)
	Coordenadas eNodeB4	(870 m, 370 m)
RAN/PHY	Interferencia DL/UL	Activada (true)
	Potencia Tx UE	23 dBm
	Potencia Tx eNodeB	46 dBm
	Target BLER	0.01
	Período de retroalimentación	40 TTI
	Binder (anillos)	10
Asociación/HO	Asociación dinámica	true
	Handover	true (habilitado)
Vehicles	Número de UEs	1
	Movilidad	LinearMobility, updateInterval = 0.2 s
	Velocidad UE	8.5–15.28 m/s (\approx 30.6–55.0 km/h)
	Rumbo inicial	0°
	Posición inicial	X \approx 540–570 m; Y \approx 474–478 m
UE Applications	Apps por UE	2 (UDP + TCP)
	UDP UE \rightarrow Server	600 B cada 50 ms, puerto 10000
	TCP UE \rightarrow Server	req 100 B, resp 1 KiB, think Exp(0.1s), idle Exp(1 s), puerto 20000
Configuración IP	Archivo de configuración	demo.xml

Tabla 2.1. Tabla de configuración de los parámetros en el escenario 4G

Este escenario se planteó como una referencia para contrastar el desempeño de las redes 5G. Se definieron tres casos de simulación: (A) cinco UEs conectados a un solo eNodeB bajo cobertura centralizada; (B) distribución de UEs entre varios eNodeB con handover habilitado; y (C) congestión inducida en el PGW y enlaces X2 activos entre eNodeB.

El área de simulación se mantuvo en 1360 metros por 555 metros, con cinco usuarios móviles desplazándose linealmente en el eje X entre las coordenadas 290 y 350 metros, mientras que en el eje Y permanecieron fijos en 480 metros. La velocidad osciló entre 8.5 y 15.28 m/s, con un tiempo total de simulación de 40 segundos. Los eNodeB se ubicaron en posiciones estratégicas: (350,520), (390,415), (836,530) y (870,370).

En cuanto a la parametrización radio, los UEs transmitieron a 23 dBm y los eNodeB a 46 dBm. Se fijó un BLER objetivo de 0.01 con retroalimentación CQI cada 40 ms. El modelo incluyó interferencia tanto en uplink como en downlink, y se configuraron 10 anillos en el binder de LTE. La asociación dinámica y el handover estuvieron habilitados, con enlaces X2 configurados bidireccionalmente entre pares de estaciones base.

Las técnicas de mejora se enfocaron en habilitar el handover inter-eNodeB mediante los enlaces X2, implementar priorización de tráfico en el PGW para reducir efectos de congestión y aplicar balanceo de carga entre estaciones base en situaciones de alta demanda.

En la comparación de protocolos se incluyeron UDP (*User Datagram Protocol*), como ejemplo de tráfico en tiempo real y baja complejidad; TCP (*Transmission Control Protocol*) Reno con SACK (*Selective Acknowledgement*), para tráfico confiable, aunque con mayores retardos; y SCTP (*Stream Control Transmission Protocol*), como protocolo adicional robusto y orientado a la transmisión de mensajes en entornos multi-enlace LTE.

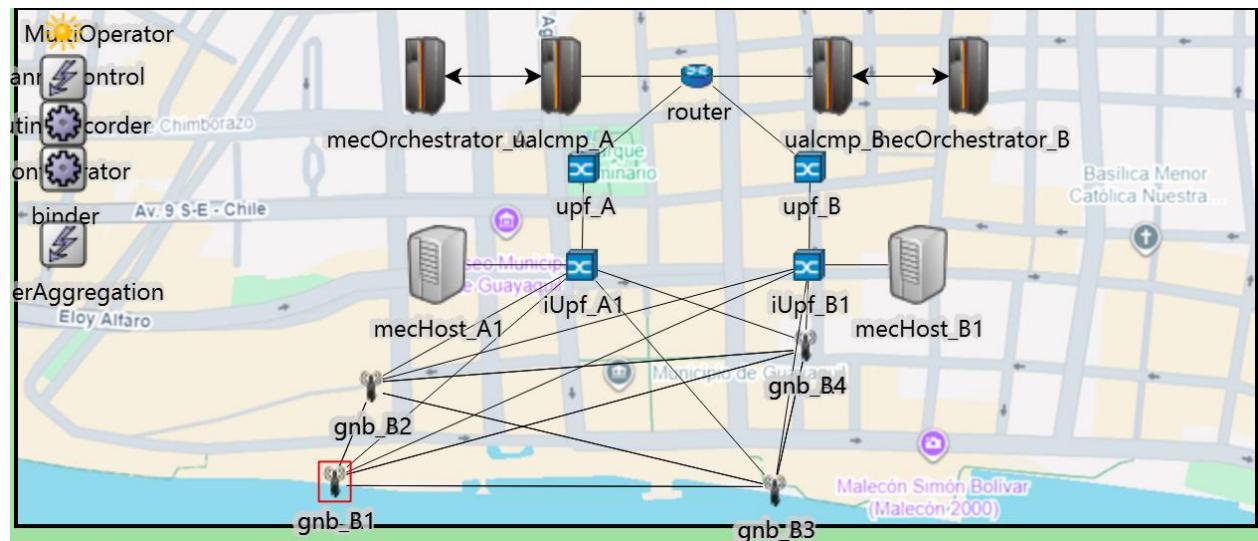


Ilustración 2.4. Escenario 2–Red 5G con un solo MEC

Entidad	Parámetro	Valor
General	Tiempo de simulación	40 s
	Red	simu5g.simulations.NR.mec.multiOperator.MultiOperator
gNodes	Nodos gNB	gnb_A1..A4 y gnb_B1..B4 (co-localizados)
	Coordenadas gNB1	(350 m, 520 m)
	Coordenadas gNB2	(390 m, 415 m)
	Coordenadas gNB3	(836 m, 530 m)
	Coordenadas gNB4	(870 m, 370 m)
VRU	Interferencia DL/UL	Activada (true)
	Potencia Tx UE	23 dBm
	Potencia Tx gNB	46 dBm
	Target BLER	0.01
	Periodo de feedback	40 TTI
Vehicles	Número de UEs por operador	30
	Velocidad	LinearMobility, speed = 8.5–15.28 m/s (\approx 30.6–55 km/h)
	Asociación inicial	dynamicCellAssociation = true, handover = true
MEC	Configuración	MEC único: mecHost_A* maneja todos los gNBs A y B
	Servicios MEC	LocationService en puerto 10020, serviceRegistry en puerto 10021
	Orquestador	mecOrchestrator_A
Aplicaciones en UE	Número de apps	2 por UE
	App 0 (control)	DeviceApp, puerto 4500, conectado a UALCMP_A
	App 1 (tráfico)	UEWarningAlertApp, 6000 B, inicio 1s, fin 40s, posición (600,600)

Tabla 2.2. Tabla de configuración de los parámetros en el escenario 5G SingleMEC

En este escenario se modeló la operación de dos operadores, cada uno con un único MEC Host encargado de procesar el tráfico vehicular de sus respectivos usuarios. Se contemplaron tres casos de análisis: (A) usuarios conectados únicamente al Operador A y gestionados por su MEC Host A1; (B) usuarios conectados únicamente al Operador B y gestionados por el MEC Host B1; y (C) un escenario mixto donde coexistían UEs de ambos operadores con movilidad entre celdas y handover activo.

La parametrización del escenario se estableció sobre un área de simulación de 1360 metros por 555 metros, con un total de 30 usuarios (15 para cada operador). Se empleó un modelo de movilidad lineal (LinearMobility), en el cual las posiciones iniciales se distribuyeron entre los rangos $93\text{--}108 \times 3$ metros en el eje X y $237\text{--}239 \times 2$ metros en el eje

Y. La velocidad de los vehículos varió de forma uniforme entre 8.5 y 15.28 m/s, y la simulación se ejecutó por un tiempo total de 40 segundos.

En la parametrización radio, se utilizaron dos portadoras en agregación: la primera (CC0) en frecuencia de 5.9 GHz con 50 RBs asignados al Operador A, y la segunda (CC1) en frecuencia de 4.9 GHz con 50 RBs asignados al Operador B. La potencia de transmisión se fijó en 23 dBm para los UEs y en 46 dBm para los gNBs. Además, se definió un BLER objetivo de 0.01 y la retroalimentación CQI cada 40 ms. Tanto la asociación dinámica como el *handover* fueron habilitados, junto con la interconexión X2 entre estaciones base.

Las técnicas aplicadas para mejorar el desempeño se basaron en el uso de un MEC distribuido con un único *host* por operador, lo que permitió disminuir la latencia respecto a la arquitectura centralizada de LTE. Cada MEC *Host* dispuso de un MEC *Orchestrator* que gestionó los recursos y habilitó servicios como LocationService, clave para aplicaciones vehiculares. Asimismo, se desplegó la aplicación *WarningAlertApp* en los MEC, permitiendo comunicaciones críticas en tiempo real.

Para la comparación de protocolos se incluyeron tres alternativas: TCP Reno con SACK, empleado como referencia de tráfico confiable; UDP, representativo de aplicaciones críticas con baja latencia; y SCTP, considerado por su robustez y orientación a mensajes en entornos multi-enlace.

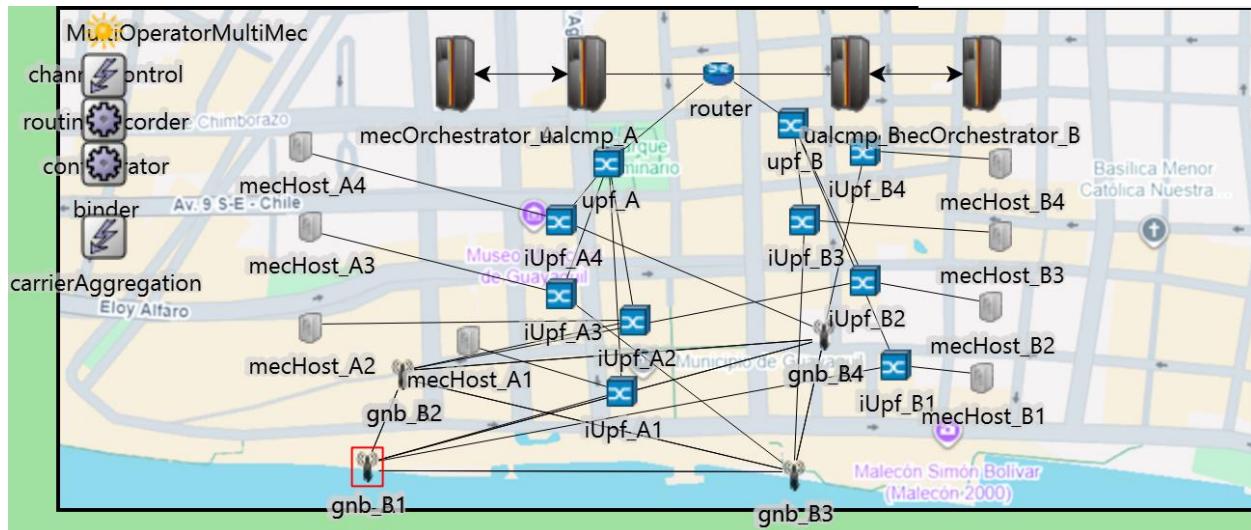


Ilustración 2.5. Escenario 3 – Red 5G con varios MEC

Entity / Módulo	Parameter	Value
General	Simulation time	40 s
	Network	simu5g.simulations.NR.mec.multiOperator.multiMec.MultiOperatorMultiMec
	Scalars/Vectors recording	Enabled (all, counters, RLC)
	IPv4 Config file	demo.xml
Scenario & Mobility	Floorplan (X,Y)	0–1360 m ; 250–555 m
	UE Mobility model	LinearMobility
	UE Initial X	intuniform(93,118)*3 m
	UE Initial Y	intuniform(237,239)* 2 m
	UE Speed	uniform(12.5, 15.28) m/s
	UE Heading	0°
VRU	UL/DL Interference	true / true
	UE Tx power	23 dBm
	gNB Tx power	46 dBm
	Link Adaptation target BLER	0.01
	CQI feedback period	40 TTI (~40 ms)
	Carrier A (CC0)	5.9 GHz / 50 RBs / numerology u
	Carrier B (CC1)	4.9 GHz / 50 RBs / numerology u
	Dynamic association / HO	true / true
gNodes	gNB_A1 / gNB_B1	(350 m, 520 m)
	gNB_A2 / gNB_B2	(390 m, 415 m)
	gNB_A3 / gNB_B3	(836 m, 530 m)
	gNB_A4 / gNB_B4	(870 m, 370 m)
	X2 peering	Full mesh entre A1–A4 y B1–B4
	Num. X2 apps por gNB	3
	X2 server ports	5000 + index
	numUe_A / numUe_B	15 / 15 (editable)
Vehicles	UE→gNB_A (start)	nrMacCellId=1; nrMasterId=1

	UE→gNB_B (start)	nrMacCellId=4; nrMasterId=4
MEC	IP forwarding	true
	Capacity (maxApps/RAM/Disk/CP U)	100 / 32 GB / 100 TB / 400000
	Gateways	MEC_A→upf_A, MEC_B→upf_B
	Num. services	1 (LocationService)
	Service ports	Service=10020 ; Registry=10021
	Service timing	Request=20μs ; Subscription=11μs
	RNG mapping	Registry→rng-0; Service→rng-1
Orchestration	enableMecOperatorB	true
	MECHost A1–A4	gNB_A1–A4(uno a uno)
	MECHost B1–B4	gNB_B1–B4(uno a uno)
	Orchestrator lists	A:[A1..A4], B:[B1..B4]
	UALCMP Gateways	A→upf_A ; B→upf_B
UE Applications	Num. UE apps	2
	App[0]	DeviceApp (port 4500) + UALCMP (ualcmp_A/B, port 1000)
	App[0] MEC Package	ApplicationDescriptors/WarningAlertApp.json
	App[1]	UEWarningAlertApp (self deviceAppAddress, port 4500)

Tabla 2.3. Tabla de configuración de los parámetros en el escenario 5G MultiMEC

Este escenario representó la configuración más avanzada del estudio, al incluir múltiples MEC Hosts por operador para evaluar balanceo de carga y resiliencia de red. Se establecieron tres casos de análisis: (A) ambos operadores disponiendo de MEC dedicados en paralelo; (B) todos los usuarios de ambos operadores compartiendo un único MEC; y (C) variaciones de tráfico en aplicaciones de seguridad vehicular y de comunicaciones UDP.

El área de simulación fue de 1360 metros por 555 metros. Se modelaron tres UEs conectados al Operador A y dos al Operador B, con posiciones iniciales entre $(93\text{--}118) \times 3$ metros y $(237\text{--}239) \times 2$ metros. La velocidad de los vehículos osciló entre 12.5 y 15.28 m/s bajo un modelo de movilidad lineal. El tiempo de simulación se mantuvo en 40 segundos.

En la parametrización radio, se consideraron dos portadoras: CC0 en frecuencia 5.9 GHz con 50 RBs y CC1 en frecuencia 4.9 GHz con 50 RBs. La potencia de transmisión de los UEs se fijó en 23 dBm y la de los gNodeB en 46 dBm. Se estableció un BLER objetivo de 0.01 con retroalimentación CQI cada 40 ms, habilitando asociación dinámica y handover.

Las técnicas aplicadas incluyeron la integración de múltiples MEC Hosts (cuatro por operador), gestionados a través de un MEC Orchestrator con listas de recursos para balanceo de carga. Se habilitaron además servicios MEC de localización (LocationService), críticos en aplicaciones vehiculares.

En cuanto a los protocolos comparados, se incluyó TCP Reno con SACK como referencia de tráfico confiable, UDP como protocolo ligero de baja latencia para mensajes de seguridad, y QUIC como protocolo moderno orientado a mejorar la latencia y la seguridad en redes 5G.

Capítulo 3

3. Resultados y análisis

La presente sección expone los principales resultados obtenidos a partir de la simulación de los escenarios diseñados en OMNeT++ con el *framework* INET y Simu5G, integrando el uso de MEC para la evaluación de aplicaciones vehiculares. El análisis se centró en tres entornos comparativos: una red 4G como punto de referencia, una red 5G con MEC centralizado y una red 5G con MEC distribuido.

Para cada escenario se realizaron 100 simulaciones, considerando la participación de 30 UEs en cada ejecución. Estos datos permitieron obtener distribuciones estadísticas que sirvieron de base para la elaboración de las funciones de distribución acumulada (CDF). Dichas gráficas facilitaron la comparación directa de las métricas críticas del sistema: la latencia y el throughput.

El objetivo de este capítulo es presentar de manera clara y estructurada los hallazgos más relevantes, empleando gráficos y tablas que permiten visualizar las diferencias entre escenarios y protocolos evaluados. Los resultados se presentan como datos agregados, evitando la exposición de información sensible y privilegiando la interpretación comparativa entre tecnologías y configuraciones.

El análisis buscó no solo cuantificar los beneficios del 5G y del MEC en entornos vehiculares, sino también identificar los desafíos técnicos que persisten, tales como el *handover* dinámico y el balanceo de carga. Con ello, se estableció un marco de referencia que contribuye a proponer soluciones para mejorar la confiabilidad y eficiencia de las comunicaciones vehiculares en Ecuador.

3.1 Resultados del escenario 4G

Los resultados obtenidos en el escenario de red 4G constituyeron la línea base de comparación para este estudio. Este entorno permitió evaluar el desempeño de las

comunicaciones vehiculares bajo la tecnología actualmente disponible en el Malecón 2000, donde los usuarios experimentan, en promedio, los niveles de latencia y throughput que se reflejan en las simulaciones realizadas.

El análisis de este escenario fue fundamental, ya que proporcionó una referencia directa de las condiciones reales que enfrentan los sistemas de comunicación en la zona. A partir de estas métricas, se estableció el contraste necesario frente a los escenarios de red 5G, tanto con MEC centralizado como distribuido, permitiendo identificar de manera más clara las ventajas y mejoras potenciales que ofrecería la nueva generación de redes móviles en un entorno urbano de alta densidad como el de Guayaquil.

3.1.1 Análisis de métricas

- **Latencia:**

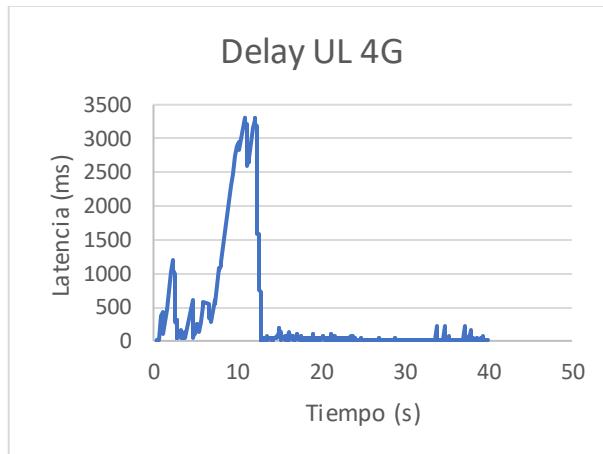


Ilustración 3.1. Gráfica promedio de la latencia UL del recorrido de los UE

La ilustración 3.1 se puede observar como al inicio de la simulación la red experimentó una congestión, esto es debido a que al principio todos los dispositivos se encuentran juntos todos los UE, donde se ve que tienen un retardo hasta de 3000 ms, por ende es donde el pico de latencia es el mas alto. Una vez pasados los 15 segundos la gráfica empieza a estar mas estable, ya que los dispositivos se van separando y el tráfico se vuelve a normalizar, por

consiguiente se ve que el retardo pasa a ser en promedio unos 50 ms, que es lo que los usuarios experimentan en la mayoría de la simulación.

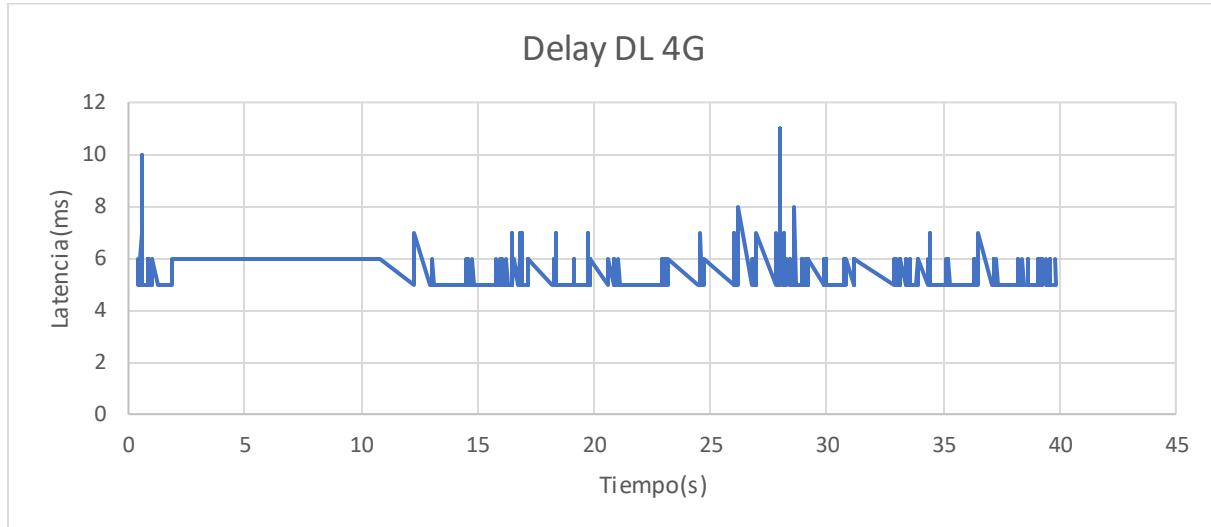


Ilustración 3.2. Gráfica general de la latencia DL del recorrido de los UE

Al contrario de la ilustración 3.2, en esta vemos que el enlace descendente es más estable y mantiene sus valores fluctuando con en promedio unos 6 ms, valores relativamente bajos en comparación con el uplink. Sin embargo, cabe recalcar que hay unos pequeños picos momentáneos entre los 25 a 30 segundos de simulación, esto puede por algunas fluctuaciones en el tráfico en ese momento en la red durante el handover, pero el valor promedio aún muestra una eficiencia al momento de que los dispositivos reciban datos durante el recorrido.

- **Throughput:**

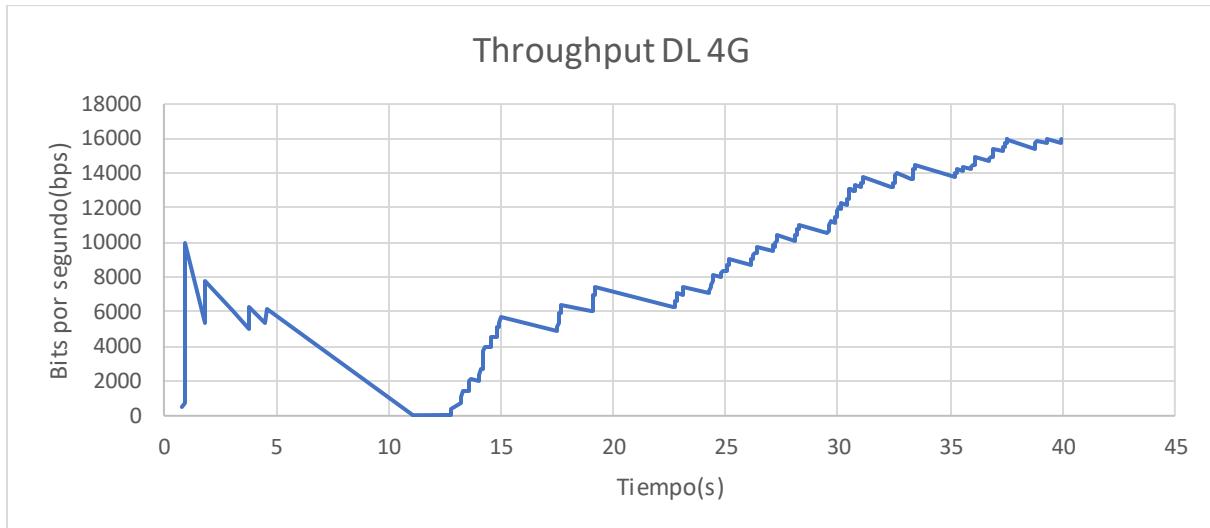


Ilustración 3.3. Gráfica general del throughput DL del recorrido de los UE

Para la ilustración 3.3, se obtiene el throughput descendente de los dispositivos durante la simulación, esto se refiere a los datos exitosos transmitidos en lo que dura la simulación. En los 12 primeros segundos de trayecto, vemos como hay una caída abrupta en el throughput, nuevamente, esto es debido a la congestión en el tráfico y un posible problema de asignación de recursos en la red. Pasado el segundo 15, podemos observar como la red se recupera y empieza a manejar una carga más grande de datos hasta los 16000 bps, así que se puede decir que la red estuvo en crecimiento y optimizada durante la mayor parte de la simulación.

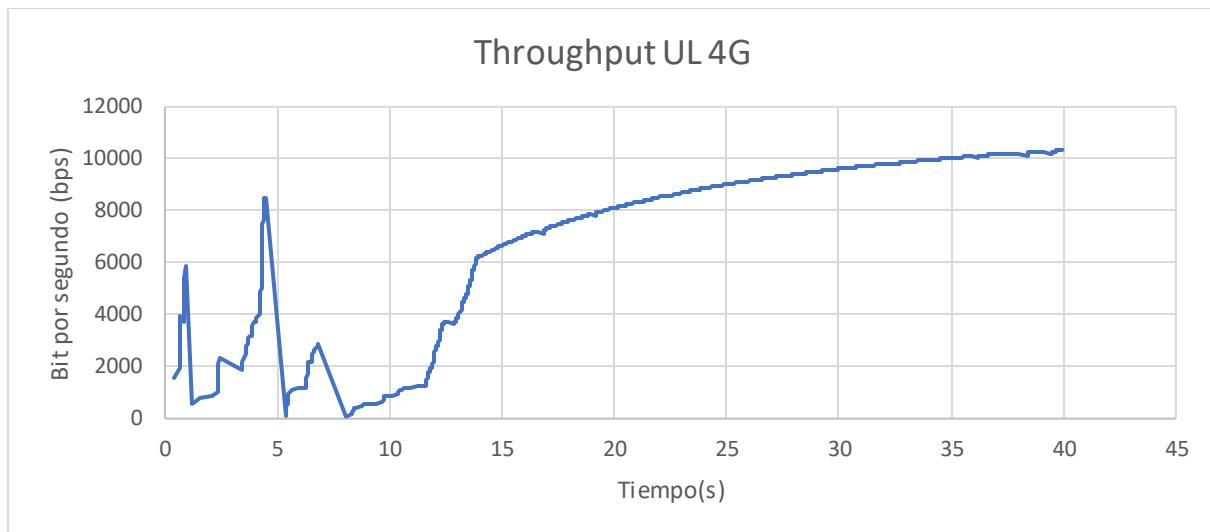


Ilustración 3.4. Gráfica general del throughput UL del recorrido de los UE

Comparando con la ilustración 3.4, tenemos que el enlace ascendente durante los primeros 12 segundos presenta un rendimiento un tanto volátil, ya que presenta muchas subidas caídas bruscas, que son generadas por el tráfico que hubo en la red, el cual afectó tanto al enlace descendente como ascendente. Pero luego del segundo 12, se observa como empieza a tener una recuperación progresiva, alcanzando unos valores hasta de 10000 bps, ya una vez que la red se estabilizó. Así que podemos decir que, a pesar de tener un inicio con congestión, la red puede ser eficiente cuando se logra controlar.

3.2 Resultados del escenario 5G

El escenario de red 5G representó un entorno ficticio y prospectivo, diseñado para simular el comportamiento que podría experimentarse en el Malecón 2000 en caso de iniciarse la implementación de esta nueva generación tecnológica. A diferencia del escenario 4G, que refleja la situación actual, el 5G se modeló considerando sus capacidades avanzadas de transmisión, baja latencia y soporte para aplicaciones vehiculares críticas.

Este escenario permitió realizar una comparación directa con los resultados de 4G, identificando las mejoras potenciales en latencia y throughput que se obtendrían al migrar hacia la nueva red. De esta manera, los resultados del 5G no solo funcionaron como un contraste con

las condiciones reales, sino también como una herramienta para estudiar el comportamiento proyectado de la red vehicular en el proceso de implementación de la tecnología 5G en Ecuador.

3.2.1 Análisis de métricas

3.2.1.1 Single MEC

- **Latencia:**

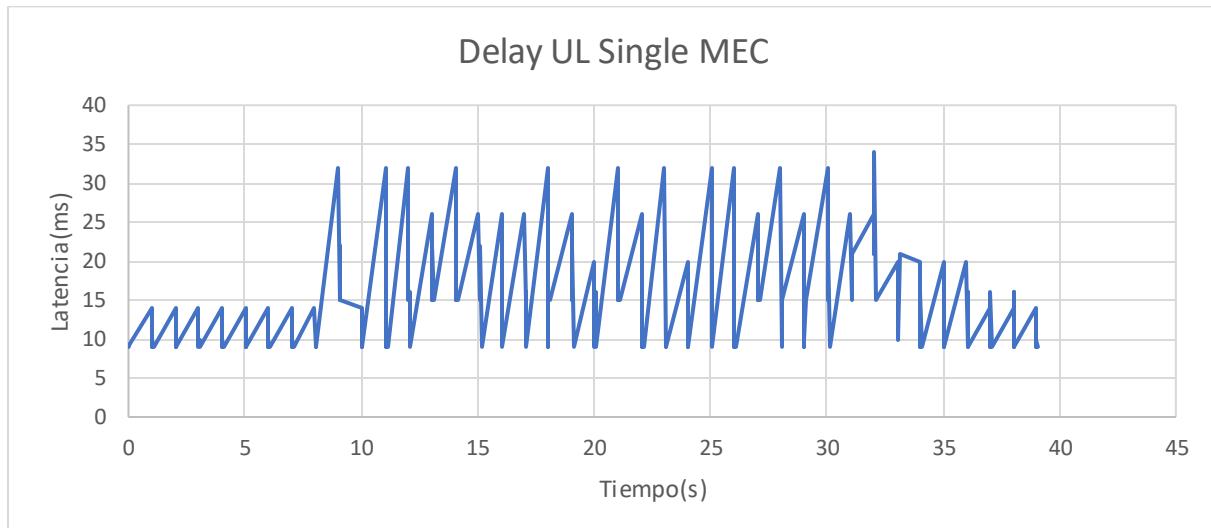


Ilustración 3.5. Gráfica general del Delay UL del recorrido de los UE

Comparando con la ilustración 3.1, tenemos un delay que se mantiene consistentemente bajo en comparación al enlace UL de 4G, ya que observamos que va desde 9 hasta 35 ms, manteniéndose estable durante toda la simulación, esto se debe a la incorporación del MEC, ya que este procesa los datos cerca del usuario, es decir al borde de la red, eliminando así el *delay* que se genera por la distancia hasta el servidor centralizado como en 4G y esto es importante para la comunicación vehicular, ya que esta exige latencias para una mejor experiencia para los usuarios.

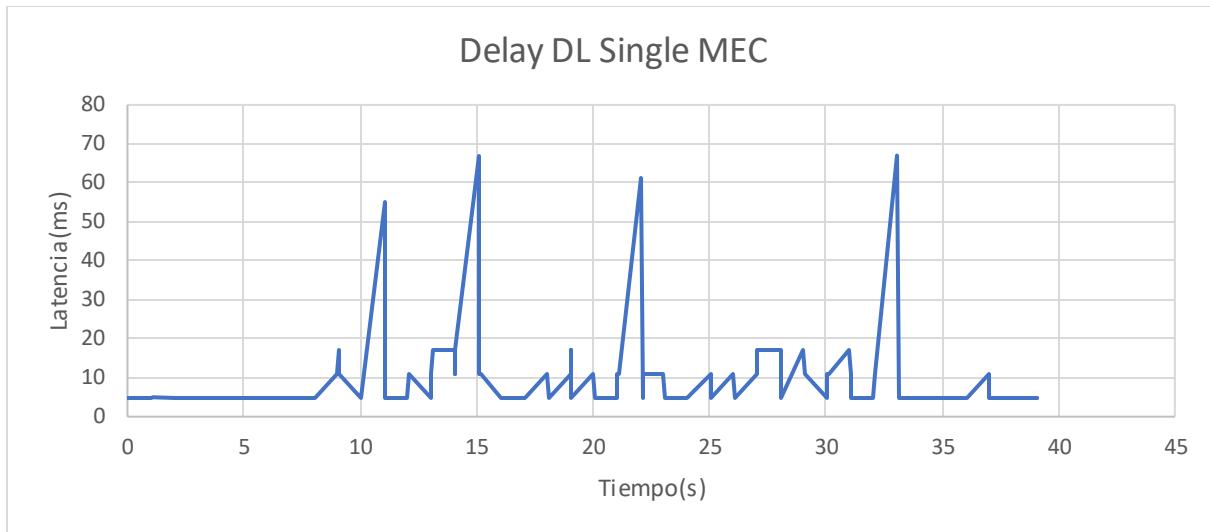


Ilustración 3.6. Gráfica general del Delay DL del recorrido de los UE

A diferencia de la ilustración 3.5, se tiene un comportamiento un poco diferente, ya que en diferentes partes de la simulación se encuentran unos picos muy pronunciados que alcanzan los 0.065 ms, esto sucede debido a una saturación breve al momento de procesar múltiples solicitudes a la vez, por eso es necesaria la incorporación del escenario 3, que se maneja con Multi MEC, el cual mas adelante analizaremos sus datos, pero a pesar de estos picos el retardo promedio sigue siendo bajos entre 0.01 y 0.02 ms. En comparación con los valores de 4G, se ve una clara ventaja la incorporación de la tecnología 5G para estas aplicaciones que requieren una baja latencia.

- **Throughput:**

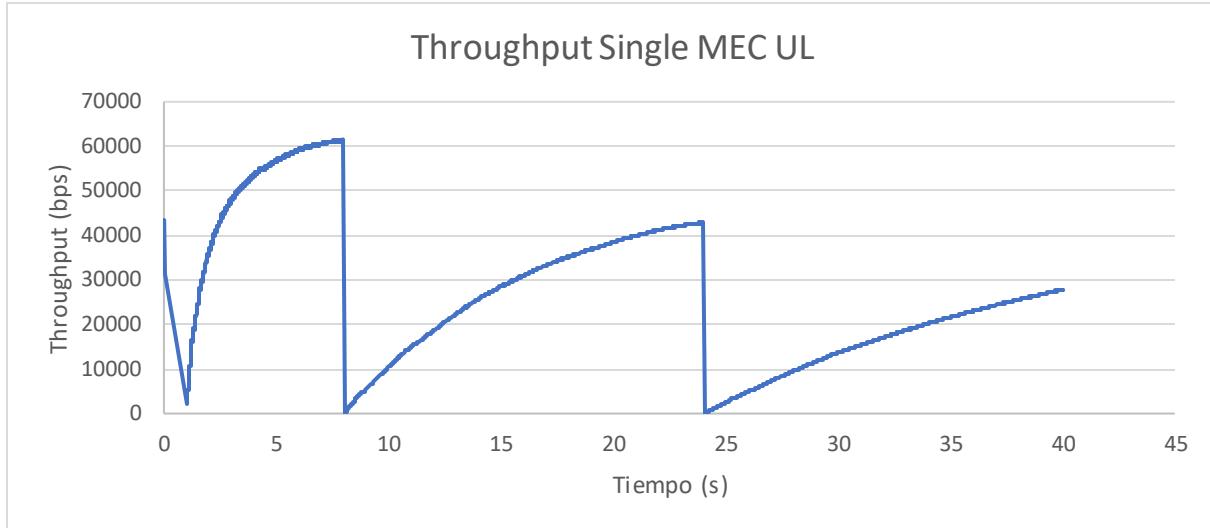


Ilustración 3.7. Gráfica general del Throughput UL del recorrido de los UE

Al momento de analizar la ilustración 3.7, podemos observar que en los 8 primeros segundos de simulación, llega hasta un poco mas de 60000 bps, al momento de hacer el cambio de celda, estos bajan y luego rondan hasta los 40000 bps, y en último trayecto vuelven a bajar y luego se mantienen en subida hasta los 30000 bps. Con estos valores obtenido se puede decir que la red puede ofrecer alto rendimiento pero en cierto momentos porque luego sufre un congestión en el tráfico, esto se debe a la alta demanda de usuarios que se encuentra, lo cual analizaremos luego si se soluciona con varios de estos Mec para reducir la carga al tener uno solo.

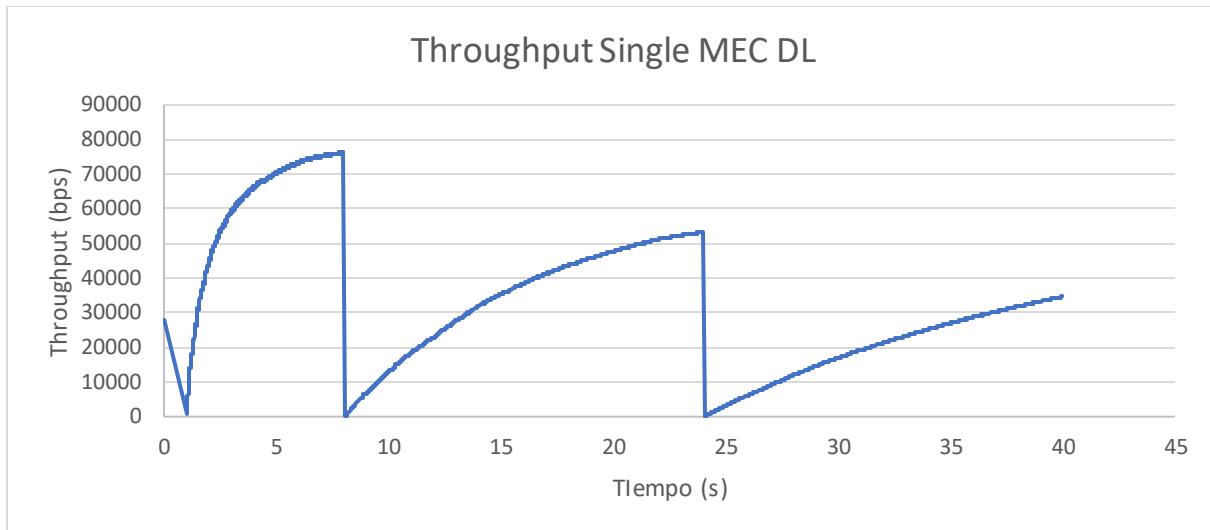


Ilustración 3.8. Gráfica general del Throughput UL del recorrido de los UE

En esta ilustración perteneciente al de descargar, se tiene que ahora en los primeros segundos el throughput alcanza un pico de 78000 bps, al cambiar de celda vemos que bajan drásticamente, pero se empieza a recuperar y subir hasta unos 52000 bps, al realizar un último cambio de celda vemos que vuelven a bajar los bps, pero empieza a subir hasta llegar a un valor de 35000 bps. Comparando con los valores de 4G, tenemos muchos mejores resultados de rendimiento, aunque suelen caer mucho al momento del cambio de celda, esto se debe a la saturación que recibe un MEC, al ser solo uno, no abarcaría la cantidad de tráfico que se genera. Ya luego se analizará el comportamiento con múltiples MEC y se espera que esto mejore.

3.2.1.2 MultiMec

- **Latencia:**

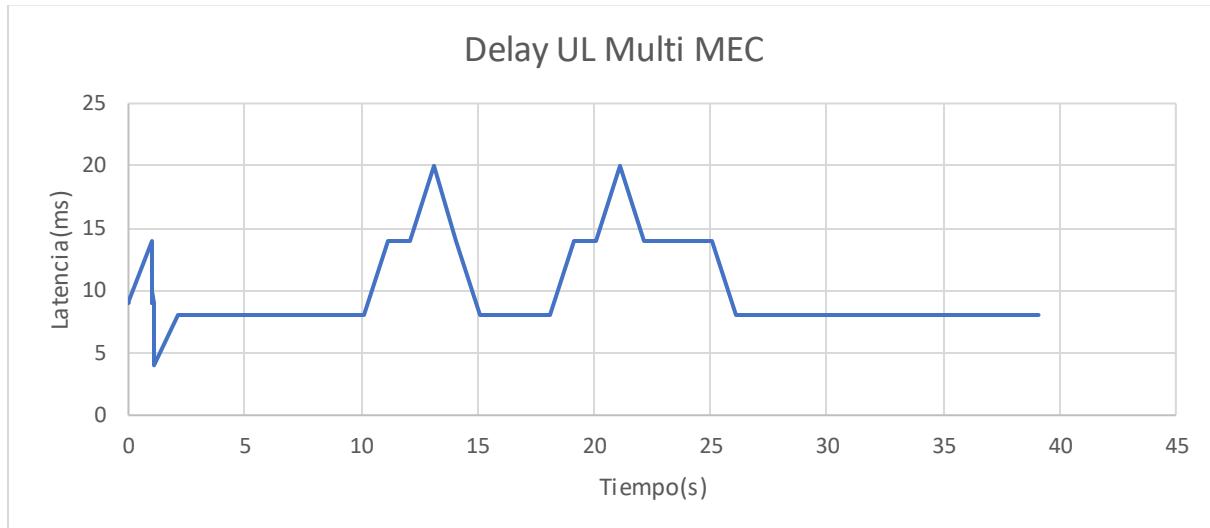


Ilustración 3.9. Gráfica general del Delay UL del recorrido de los UE

En la ilustración 3.9 podemos notar un retardo muy bajo, que durante toda la simulación se mantiene estable entre 5 y 20 ms, valores beneficiosos para la aplicación de comunicaciones móviles, ya que esta exige una buena latencia. Comparando con el escenario Single MEC, se puede ver una mejoría en lo que se refiere al retardo, en especial cuando hay gran cantidad de usuarios, así que podemos ver su verdadero desempeño en rendimiento y capacidad para escenarios de alta demanda de tráfico.

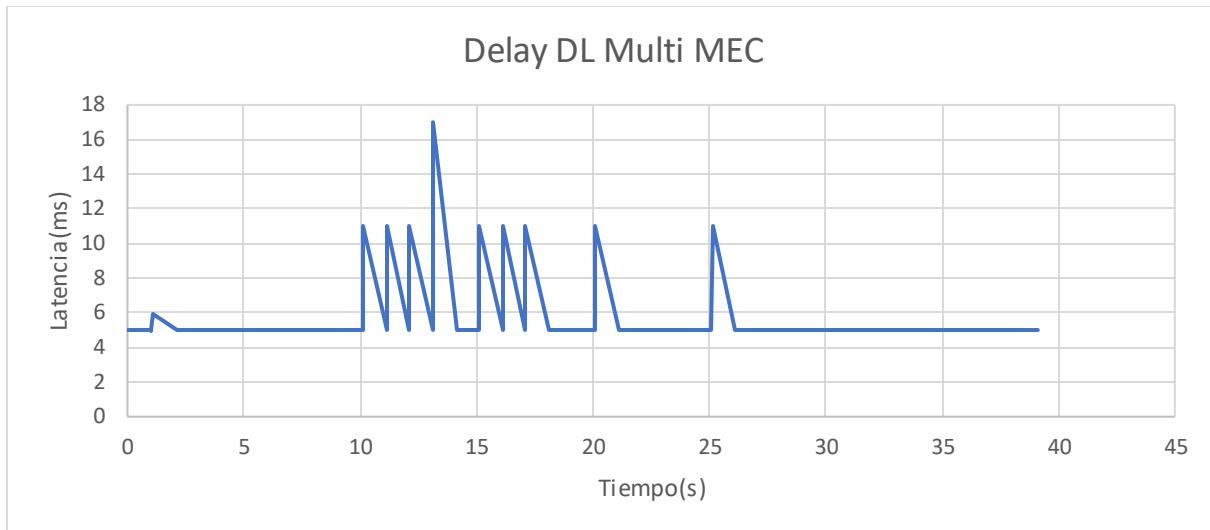


Ilustración 3.10. Gráfica general del Delay DL del recorrido de los UE

En esta ilustración se muestra que la latencia de mantiene en un rango estable de 5ms alcanzando picos de 11ms con el mayor alcanzando 17 ms, pero a comparación el escenario 4G, este se mantiene muy por debajo de esos valores. Ahora a comparación con el escenario Single MEC, esta gráfica es ligeramente más estable, ya que funciona de manera eficiente y se encuentra distribuyendo la carga y evitando la saturación temporal que se presentaba en el otro escenario 5G. Aun así, en ambos escenarios, la latencia se mantuvo baja, por ende, la red prioriza la entrega de los paquetes de datos para mantener la comunicación entre vehículos en tiempo real.

- **Throughput:**

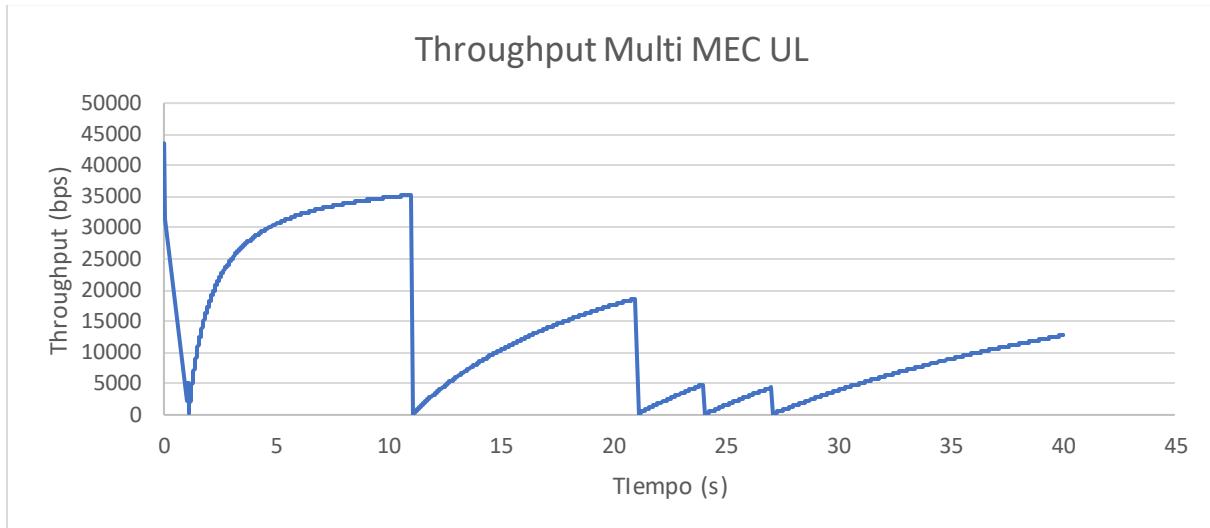


Ilustración 3.11. Gráfica general del Throughput UL del recorrido de los UE

Comparando con la ilustración anterior, tenemos valores mucho menores de bps, esto se debe a que, en este tipo de redes el usuario se concentra más en la navegación por internet, archivos multimedia o descargar archivos, por ende, se necesita más bps en ese sentido. Ahora analizando la gráfica, podemos ver como en 10 primeros segundos los bps llegando hasta un pico de 35000 bps, luego al cambiar de celda, empieza a bajar hasta llegar cerca de los 20000 bps y por último se estabiliza y se mantiene entre los 5000 bps. Esto puede que no sea lo suficientemente alto para la arquitectura, pero cumple con su función, debido a que se necesita que siempre esté en buen rendimiento la red y se mantenga siempre con una latencia baja.

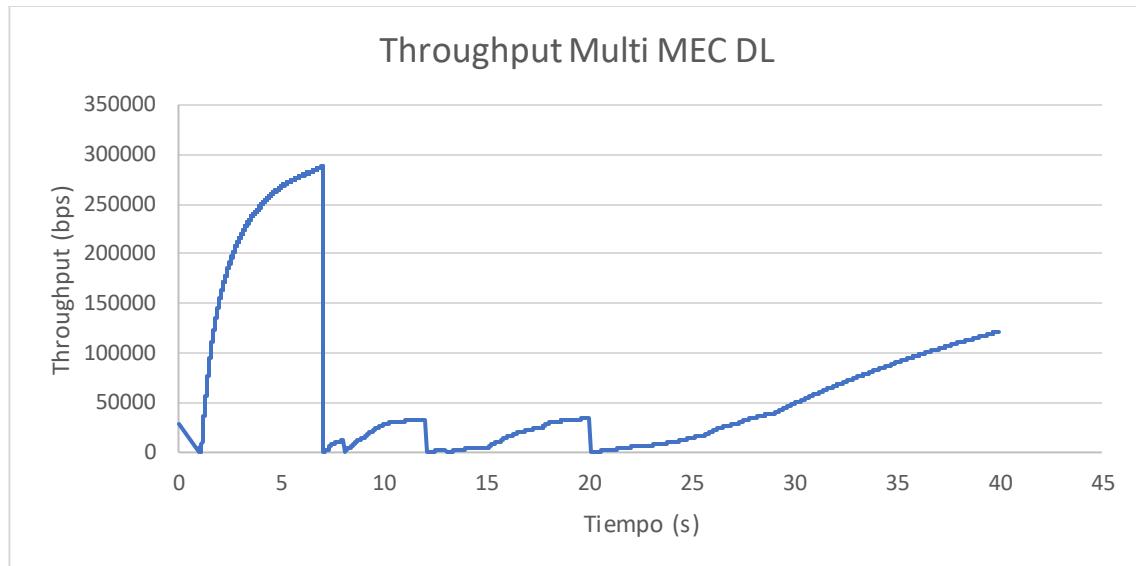


Ilustración 3.12. Gráfica general del Throughput UL del recorrido de los UE

En la ilustración 3.12, se puede notar que durante los primeros 5 segundos de simulación, llega hasta una capacidad de 290000 bps, luego cuando ocurre el cambio de celda, vemos que se estabiliza y va rondando de los de 30000 a 40000 bps, lo cual significa que la red si está entregando los datos a los vehículos de manera eficiente. Las fluctuaciones presentes son normales, ya que los vehículos están en movimiento, y a pesar de esto, mantiene sus valores de bps altos, lo que demuestra que este escenario Multi MEC funciona y es capaz de balancear la carga y evitar saturaciones que se podría dar con un solo MEC.

3.3 Análisis de las métricas con gráficas de función de distribución acumulada (CDF)

Las gráficas de función de la CDF se emplearon con el fin de representar de manera comparativa el comportamiento de las métricas clave: latencia y throughput. Este tipo de análisis permitió visualizar cómo se distribuyen los valores obtenidos a lo largo de las 100 simulaciones por escenario, facilitando la identificación de tendencias, valores límite y diferencias significativas entre las configuraciones evaluadas (4G, 5G con MEC único y 5G con MEC distribuido).

3.3.1 Latencia Uplink

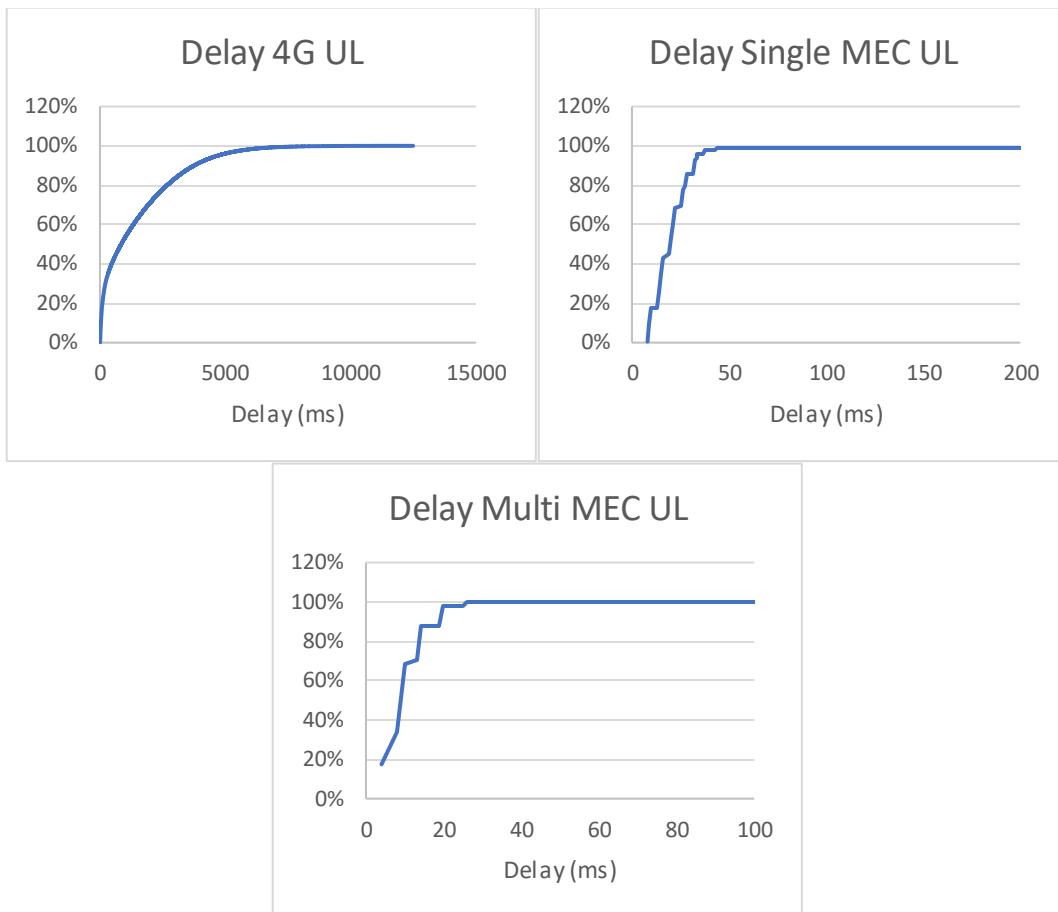


Ilustración 3.13. Gráfica CDF del Delay UL de los 3 escenarios simulados

Para comparar las 3 gráficas de esta ilustración, tenemos que analizar por partes, empezando por el escenario 4G, donde el primer cuartil alcanza los 160 ms, el segundo cuartil los 1000 ms y el tercer cuartil 2000 ms, siendo los picos más altos valores de hasta 10000 ms. Luego tenemos el escenario Single MEC, con una curva mas pronunciada y en contraste a la red 4G, el primer cuartil entra hasta los 14 ms, el segundo cuartil muestra el valor de la mediana de 20ms y el tercer cuartil hasta los 25 ms, con algunas muestras alcanzando hasta 200 ms. Estos resultados quedan como evidencia que la implementación de la arquitectura 5G Single MEC, reduce drásticamente la latencia en el enlace ascendente, por lo que una muy buena opción al implementarla para la comunicación vehicular. Por último tenemos el escenario Multi MEC, cuya curva pronunciada alcanza los 8 ms en el primer cuartil, 9 ms en el segundo cuartil,

13 ms en el tercer cuartil, con los picos máximos hasta los 100 ms. Por ende, tenemos que el escenario que posee mejor latencia para ser usado en la comunicación vehicular, es la del escenario Multi MEC, ya que logra mantener los valores más bajos y estables durante la simulación, así que es lo más conveniente usar en los entornos urbanos de alta densidad.

3.3.2 Latencia Downlink

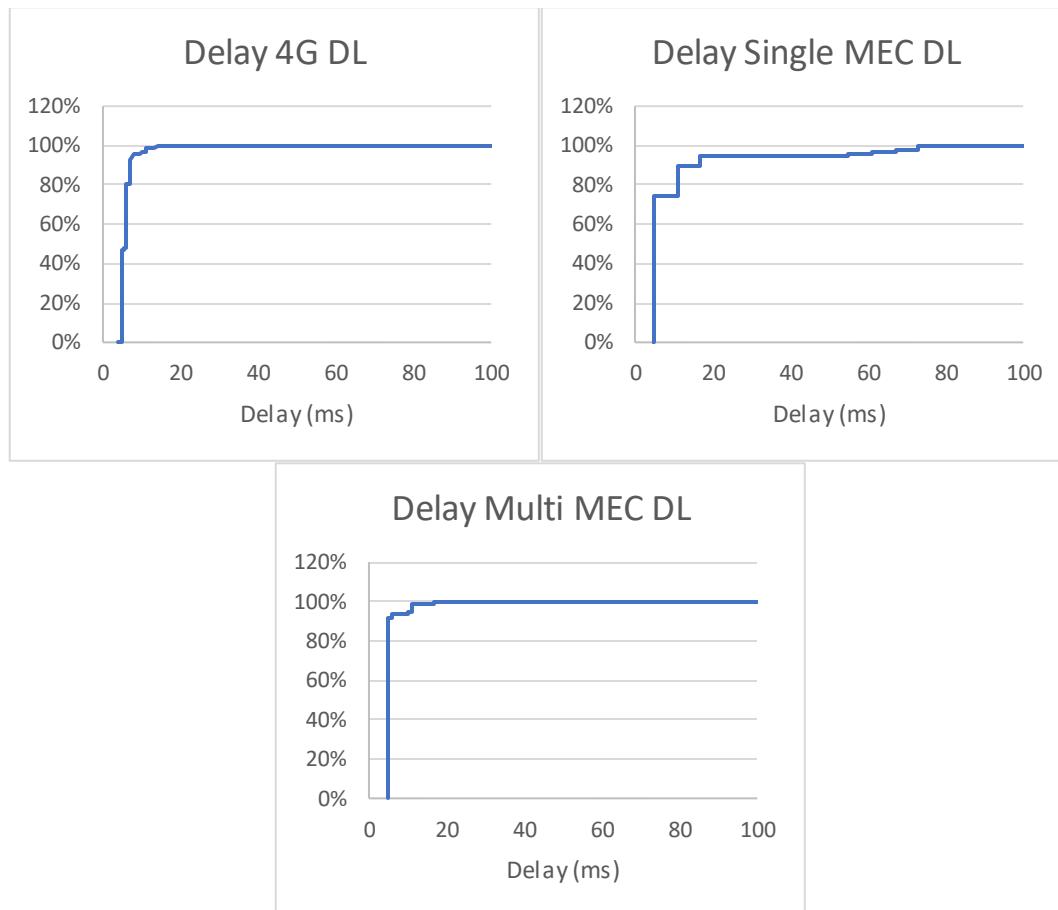


Ilustración 3.14. Gráfica CDF del Delay DL de los 3 escenarios simulados

Para esta ilustración, tenemos que la gráfica de 4G que la curva sube de manera muy abrupta en comparación al uplink, teniendo valores de 5 ms en el primer cuartil, 5.82 ms en el segundo cuartil y 6 ms en el tercer cuartil, con los picos superando los 100 ms. A pesar de esto el escenario ofrece una latencia relativamente buena para aplicaciones de consumo de datos como streaming, no es suficiente para su uso en la comunicación vehicular, ya que hay una pequeña porción de datos que experimenta una latencia de enlace descendente más alta. En el

escenario 5G Single MEC, es muy parecido al Multi MEC, ambos tienen valores estables en los 3 primeros cuartiles de 5 ms, siendo la principal diferencia que en Single MEC el último cuartil presenta latencia más alta mientras que en Multi MEC mantiene un valor menor o igual a 5 ms el 90% de los datos. Terminando con el análisis, hay una clara superioridad por parte de la arquitectura Multi MEC 5G, ya que es la que mejor cumple con las exigencias de latencia ultrabaja dado un procesamiento de datos con una latencia mínima y estable, y así poder aplicarlo en los servicios de comunicación vehicular por su estructura robusta y confiable.

3.3.3 Throughput Uplink

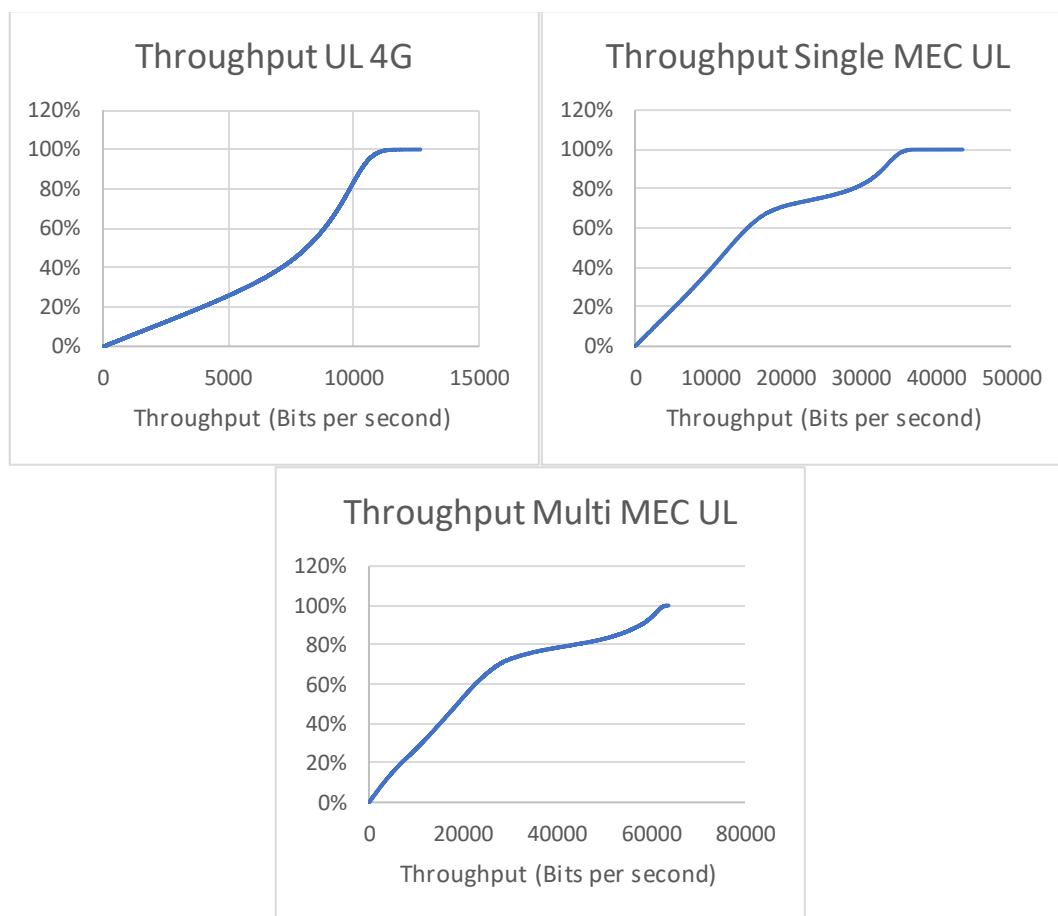


Ilustración 3.15. Gráfica CDF del Throughput UL de los 3 escenarios simulados

Para esta ilustración, el throughput del enlace uplink tiene valores 5000 bps para el primer cuartil, 8000 bps para el segundo cuartil, 9500 bps para el tercer cuartil, y llega hasta 12000 bps para el cuarto cuartil. Por otro lado tenemos el escenario Single MEC, esta gráfica

de throughput tiene valores 6800 bps para el primer cuartil, 12000 bps para el segundo cuartil, 26000 bps para el tercer cuartil, y llega hasta 43000 bps para el cuarto cuartil, esto denota una mejoría con respecto a la arquitectura 4G pero debido al tráfico de usuarios manejado por solo MEC, sigue siendo inferior al último escenario de Multi MEC, donde tiene valores 10000 bps para el primer cuartil, el segundo cuartil muestra una mediana alrededor de los 18000 bps, el tercer cuartil 34000 bps, y llega hasta 63000 bps para el cuarto cuartil, es mejor a la de Single MEC en términos de valores, lo cual eso es lo que se necesita para las comunicaciones móviles, una red robusta y capaz de soportar la saturación de tantos usuarios, así que esto concuerda con la investigación ya que nos brinda más confiabilidad que los demás escenarios.

3.3.4 Throughput Downlink

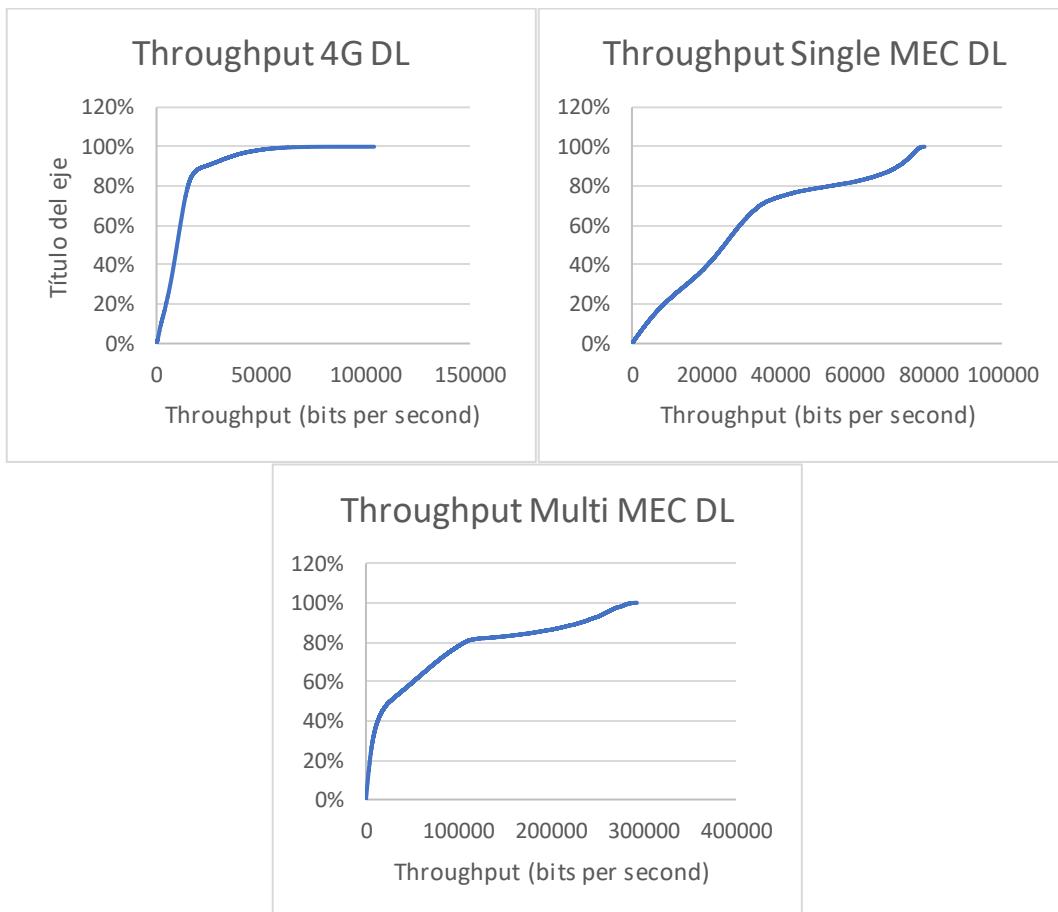


Ilustración 3.16. Gráfica CDF del ThroughputDL de los 3 escenarios simulados

Al contrario de la anterior comparación, donde el primer cuartil entran valores hasta 5600 bps, una mediana de 8000 bps y un throughput de 9500 bps en el tercer cuartil. Los resultados evidencian valores relativamente bajos para su uso en las comunicaciones móviles, ya que esta exige una alta demanda en la capacidad de descarga del usuario y el 90% de los usuarios experimentaron un throughput menor a 24000 bps. Ahora analizando la gráfica del escenario Single MEC se observa en el enlace downlink un throughput de 11400 bps para el primer cuartil, 25200 bps para el segundo cuartil, 43500 bps para el tercer cuartil, y llega hasta 80000 bps para el cuarto cuartil, eso demuestra la superioridad de esta arquitectura ante la 4G. Por último, tenemos la configuración Multi MEC, tiene valores 5700 bps para el primer cuartil, 25200 bps para el segundo cuartil, 96000 bps para el tercer cuartil, y llega hasta 280000 bps para el cuarto cuartil, evidenciando valores de throughput mayores que el resto de los escenarios, debido al que tener varios MEC, se balancea la carga entre estos y aligera el tráfico que poseería un solo servidor permitiendo que los usuarios en una celda no se vean afectados por la densidad de usuarios de otra celda.

Capítulo 4

4. Conclusiones y Recomendaciones

4.1 Conclusiones

- El uso de la herramienta OMNeT++ fue fundamental para recreación del entorno de simulación discreta de las distintas redes, donde cada evento de la red fue registrado según la configuración del script del entorno 4G y 5G para realizar el análisis. Asimismo, los frameworks de INET y Simu5G proporcionaron los módulos esenciales para la elaboración de una red que incorpore de la manera más cercana las características de una red. En conjunto, los frameworks en el simulador permitieron crear y simular escenarios de una red 4G, 5G con un MEC por operadora y 5G con un MEC por gNodeB 5G en una zona urbana de la ciudad de Guayaquil con 30 usuarios en una celda, y posterior a la obtención de los datos se realizó la evaluación comparativa entre ellos.
- Se realizó el diseño y evaluación de dos arquitecturas distintas que incorporan MEC hosts dentro del entorno 5G. Los resultados mostraron que la inclusión de MEC fortalece la eficiencia de la comunicación en comparación a una red LTE, al reducir la latencia y tener mayores valores de throughput, factores esenciales en aplicaciones V2X. En el escenario con un MEC centralizado por operadora, donde los cuatro gNodeBs se conectan a una misma UPF y posteriormente a un único host MEC, se observó una mayor simplicidad en la gestión de recursos, aunque con posibles limitaciones en la distribución de carga cuando aumenta la densidad vehicular en varias celdas. Por otro lado, el escenario con MEC distribuido por gNodeB, en el cual cada estación base cuenta con su propio UPF y host MEC, ofreció un mejor desempeño en términos de escalabilidad y balance de tráfico, garantizando tiempos de respuesta más estables incluso en condiciones de alta demanda.

- El análisis de los datos de los UE evidenció que la red 4G presenta limitaciones críticas, especialmente en la latencia de uplink, con valores hasta 10 segundos, lo que la hace inadecuada para aplicaciones vehiculares en tiempo real. En contraste, la red 5G con un MEC mostró mejoras significativas: en el escenario con un MEC único se redujo la latencia a decenas de milisegundos y se duplicó el throughput respecto a 4G, mientras que en el escenario Multi-MEC se alcanzaron los mejores resultados, con latencias inferiores a 13 ms en uplink, 90% de valores de downlink en 5 ms y un throughput que superó ampliamente al de 4G y al de 5G con MEC centralizado. Estos hallazgos confirman que el enfoque 5G Multi-MEC es el más adecuado para soportar comunicaciones vehiculares eficientes y aplicaciones avanzadas como los vehículos autónomos.

4.2 Recomendaciones

- Profundizar el análisis del desempeño de la red 5G mediante el registro de la tasa de pérdida de paquetes en los enlaces. Este parámetro permitiría observar adicionalmente como valores de latencia y throughput no ideales pueden afectar a la conectividad efectiva de los dispositivos a la red. OMNeT++ permite realizar mediciones de paquetes de entrada y salida; configuraciones de las estadísticas medidas durante la simulación que realicen un seguimiento del paquete enviado a su destino sería la clave para obtener una tasa sobre tiempo de la pérdida de paquetes que posee la red.
- Integrar simuladores de movimiento vehicular como SUMO (*Simulation of Urban Mobility*) resultaría en un movimiento de vehículos más impredecible y aleatorios. Estos usuarios presentarían variación de valores como distancia y velocidad más fluctuantes, cuyos resultados reflejarían a valores más apegados a la realidad del movimiento de los vehículos.

- Añadir protocolos de comunicación que realicen conexiones entre vehículos (V2V), y poder tener una comparación del desempeño no solamente de la conexión al 5GC sino también con otros automóviles. Realizando esto se puede garantizar una red de comunicación vehicular que abarca los varios sistemas de V2X y un análisis más completo.

Referencias

- 3GPP. (2019). System architecture for the 5G System; Stage 2 (Release 16). Technical Specification Group Services and System Aspects. Recuperado de <https://www.3gpp.org>
- 3GPP. (2020). TS 38.211: NR; Physical channels and modulation (Release 16). 3rd Generation Partnership Project.
- 3GPP. (2022). Technical Specification Group Services and System Aspects; System architecture for the 5G System (Release 17). <https://www.3gpp.org>
- 5G Americas. (2022). The future of mobile connectivity: Insights on 5G and its deployment. Recuperado de <https://www.5gamericas.org>
- Chen, X., Liu, Y., & Wang, J. (2021). Simulation models for 5G vehicular networks. International Journal of Network Design, 45(2), 233–248.
- ETSI. (2021). Intelligent Transport Systems (ITS); Access layer specification for Intelligent Transport Systems operating in the 5 GHz frequency band (ITS-G5). Recuperado de <https://www.etsi.org>
- FCC. (2020). *Use of the 5.850–5.925 GHz Band*. Federal Communications Commission Report and Order.
- INET Framework. (s. f.). INET Framework. Recuperado de <https://inet.omnetpp.org>
- ITU-R. (2017). Report ITU-R M.2370-0: Frequency bands for global or regional harmonization of Intelligent Transport Systems. International Telecommunication Union.

- Lee, H., & Park, S. (2021). The impact of 5G on intelligent transportation systems. *Telecommunications Review*, 33(2), 145–157.
- Müller, A., Köhler, T., & Schmitt, M. (2020). Evaluating V2I in 5G urban scenarios. *European Transport Research Review*, 12(4), 200–215.
- Nardini, G., Sabella, D., Stea, G., Thakkar, P., & Virdis, A. (2020). Simu5G – An OMNeT++ Library for End-to-End Performance Evaluation of 5G Networks. *IEEE Access*, 8, 181176–181191. <https://doi.org/10.1109/ACCESS.2020.3028550>
- OMNeT++ Team. (s. f.). OMNeT++ Discrete Event Simulator. Recuperado de <https://omnetpp.org>
- SUPERTEL. (2022). Estado del desarrollo de 5G en Ecuador. Recuperado de <https://www.telecomunicaciones.gob.ec>
- United Nations. (2023). Sustainable Development Goals: Industry, innovation and infrastructure. Recuperado de <https://www.un.org/sustainabledevelopment>
- Varga, A., & Hornig, R. (2020). An overview of the OMNeT++ simulation environment. *Simulation News Europe*, 18(3), 34–42.

Anexos

Anexo 1. Archivo omnetpp.ini para simulación de una red 4G

```

[General]
repeat = 100
network = simu5g.simulations.LTE.networks.MultiCell
sim-time-limit = 40s
output-scalar-file = ${resultdir}/${configname}/${iterationvars}-${repetition}.sca
output-vector-file = ${resultdir}/${configname}/${iterationvars}-${repetition}.vec
seed-set = ${repetition}
num-rngs = 3
**.sctp.**.scalar-recording = true # remove annoying statistics from SCTP
**.sctp.**.vector-recording = true # remove annoying statistics from SCTP
**.vector-recording = true
*.ue_*[*].*.recordStatistics = true
*.gnb_*.*.recordStatistics = true
*.*.statistic-recording = true
# intenta activar modos de grabación típicos INET
*.ue_*[*].app[*].packetDelay.result-recording-modes = all
*.ue_*[*].app[*].endToEndDelay.result-recording-modes = all
*.ue_*[*].app[*].packetReceived.result-recording-modes = all
*.ue_*[*].app[*].packetSent.result-recording-modes = all
*.*.throughput.result-recording-modes = all
**.endToEndDelay.result-recording-modes = vector
**.throughput.result-recording-modes = vector
# Contadores como escalares (para pérdida de paquetes)
**.ue_*[*].app[*].packetSent.result-recording-modes = count
**.ue_*[*].app[*].packetReceived.result-recording-modes = count

# Configurador IP
*.configurator.config = xmldoc("./demo.xml")

**.mobility.constraintAreaMaxX = 1360m
**.mobility.constraintAreaMaxY = 555m
**.mobility.constraintAreaMinX = 0m
**.mobility.constraintAreaMinY = 250m
**.mobility.constraintAreaMinZ = 0m
**.mobility.constraintAreaMaxZ = 0m
**.mobility.initFromDisplayString = false

[Config LTE_Connection]

# Radio LTE
**.downlink_interference = true
**.uplink_interference = true
**.ueTxPower = 23dBm
**.eNodeBTxPower = 46dBm
**.targetBler = 0.01
**.fbPeriod = 40tti

# Binder (requerido por LTE)
*.binder.numRings = 10

# Asociación/HO
**.dynamicCellAssociation = true
**.enableHandover = true
*.ue[*].macCellId = 0
*.ue[*].masterId = 0

# === Posiciones de eNodeBs ===
*.eNodeB1.mobility.initialX = 350m
*.eNodeB1.mobility.initialY = 520m
*.eNodeB2.mobility.initialX = 390m
*.eNodeB2.mobility.initialY = 415m
*.eNodeB3.mobility.initialX = 836m
*.eNodeB3.mobility.initialY = 530m
*.eNodeB4.mobility.initialX = 870m
*.eNodeB4.mobility.initialY = 370m
# ===== X2 entre eNodeB1 y eNodeB2 =====
*.eNodeB1.numX2Apps = 2

```

```

*.eNodeB2.numX2Apps = 2
*.eNodeB3.numX2Apps = 2
*.eNodeB4.numX2Apps = 2

# Un x2App por par, con puertos deterministas
*.eNodeB*.x2App[*].server.localPort = 5000 + ancestorIndex(1)

# Enlaces cliente (dirección = nombreDelNodo%interfazX2)
*.eNodeB1.x2App[0].client.connectAddress = "eNodeB2%x2ppp0"
*.eNodeB1.x2App[1].client.connectAddress = "eNodeB3%x2ppp0"
*.eNodeB2.x2App[0].client.connectAddress = "eNodeB1%x2ppp0"
*.eNodeB2.x2App[1].client.connectAddress = "eNodeB4%x2ppp0"
*.eNodeB3.x2App[0].client.connectAddress = "eNodeB1%x2ppp0"
*.eNodeB3.x2App[1].client.connectAddress = "eNodeB4%x2ppp0"
*.eNodeB4.x2App[0].client.connectAddress = "eNodeB2%x2ppp0"
*.eNodeB4.x2App[1].client.connectAddress = "eNodeB3%x2ppp0"

# === Movilidad SOLO en UEs ===
#*.ue_A[0].mobility.typename = "VehicleMobility"
*.numUe = 8 + ${repetition}*4

*.ue*.mobility.typename = "LinearMobility"
*.ue*[*].mobility.initialX = intuniform(180,195) * 3m
*.ue*[*].mobility.initialY = intuniform(237,239) * 2m
*.ue*[*].mobility.speed = uniform(12.5mps, 15.28mps)
#*.ue*[*].mobility.initFromDisplayString = false
#*.ue*[*].mobility.updateInterval = 0.2s      # refresco

*.ue[*].mobility.initialMovementHeading = 0deg

# --- TRÁFICO TCP SOLO EN UEs Y SERVER ---
# UEs -> server
*.ue*[*].numApps = 2
*.server.numApps = 2
*.ue*[*].app[0].typename = "UdpBasicApp"
*.ue*[*].app[0].packetName = "UDP_PACKET"
*.ue*[*].app[0].destAddresses = "server"
*.ue*[*].app[0].destPort = 10000
*.ue*[*].app[0].messageLength = 3000B
*.ue*[*].app[0].sendInterval = 50ms
*.ue*[*].app[0].startTime = 1s

*.server.app[0].typename = "UdpSink"
*.server.app[0].localPort = 10000
# UEs -> server por TCP
*.ue*[*].app[1].typename = "TcpBasicClientApp"
*.ue*[*].app[1].packetName = "TCP_PACKET"
*.ue*[*].app[1].connectAddress = "server"
*.ue*[*].app[1].connectPort = 20000
*.ue*[*].app[1].startTime = 1s
*.ue*[*].app[1].numRequestsPerSession = 3
*.ue*[*].app[1].requestLength = 100B
*.ue*[*].app[1].replyLength = 1KiB
*.ue*[*].app[1].thinkTime = 0.1s
*.ue*[*].app[1].idleInterval = 1s

# Server TCP receptor
*.server.app[1].typename = "TcpGenericServerApp"
*.server.app[1].localPort = 20000

```

Anexo 2. Archivo MultiCell.ned para simulación de una red 4G

```

package simu5g.simulations.LTE.networks;

import inet.networklayer.configurator.ipv4.Ipv4NetworkConfigurator;
import inet.networklayer.ipv4.RoutingTableRecorder;
import inet.node.ethernet.Eth10G;
import inet.node.inet.Router;
import inet.node.inet.StandardHost;
import simu5g.common.binder.Binder;

```

```

import simu5g.common.carrierAggregation.CarrierAggregation;
import simu5g.nodes.ExtCell;
import simu5g.nodes.PgwStandard;
import simu5g.nodes.Ue;
import simu5g.nodes.eNodeB;
import simu5g.world.radio.LteChannelControl;
import inet.mobility.single.VehicleMobility;
import inet.applications.udpapp.UdpEchoApp;

network MultiCell
{
    parameters:
        int numExtCells = default(0);
        int numUe= default(0);
    @display("i=block/network2;bgb=1368,565;bgi=background/malecon");
    submodules:
        channelControl: LteChannelControl {
            @display("p=50,25;is=s");
        }
        routingRecorder: RoutingTableRecorder {
            @display("p=50,75;is=s");
        }
        configurator: Ipv4NetworkConfigurator {
            @display("p=50,125");
        }
        binder: Binder {
            @display("p=50,175;is=vs");
        }
        carrierAggregation: CarrierAggregation {
            @display("p=50,258;is=vs");
        }
        server: StandardHost {
            @display("p=214,87;is=n;i=device/server");
        }
        pgw: PgwStandard {
            @display("p=188.8,181.1;is=s");
        }
        router: Router {
            @display("p=304,108;i=device/smallrouter");
        }

        // eNodeBs
        eNodeB1: eNodeB {
            @display("p=350,520;is=s");
        }
        eNodeB2: eNodeB {
            @display("p=390,415;is=vs");
        }
        eNodeB3: eNodeB {
            @display("p=836,530;is=s");
        }
        eNodeB4: eNodeB {
            @display("p=870,370;is=s");
        }

        // UEs (puedes ajustar cantidad)
        ue[numUe]: Ue {
            @display("p=284,490;is=vs");
        }

        extCell[numExtCells]: ExtCell {
            @display("p=100,480;is=vs");
        }
    connections:
        server.pppg++ <--> Eth10G <--> router.pppg++;
        router.pppg++ <--> Eth10G <--> pgw.filterGate;

        // Conexiones PGW ↔ eNodeBs
        pgw.pppg++ <--> Eth10G <--> eNodeB1.ppp;
        pgw.pppg++ <--> Eth10G <--> eNodeB2.ppp;
        pgw.pppg++ <--> Eth10G <--> eNodeB3.ppp;
        pgw.pppg++ <--> Eth10G <--> eNodeB4.ppp;

        // Conexiones X2 (completa según lo que quieras probar)
}

```

```

eNodeB1.x2++ <--> Eth10G <--> eNodeB2.x2++;
eNodeB1.x2++ <--> Eth10G <--> eNodeB3.x2++;
eNodeB1.x2++ <--> Eth10G <--> eNodeB4.x2++;
eNodeB2.x2++ <--> Eth10G <--> eNodeB3.x2++;
eNodeB2.x2++ <--> Eth10G <--> eNodeB4.x2++;
eNodeB3.x2++ <--> Eth10G <--> eNodeB4.x2++;

}

```

Anexo 3. Archivo demo.xml para simulación de una red 4G

```

<config>
  <interface hosts="*" address="10.x.x.x" netmask="255.x.x.x"/>
    <wireless hosts="eNodeB*" ue "*" interfaces="cellular"/>
</config>

```

Anexo 4. Archivo omnetpp.ini para simulación de una red 5G Single MEC

```

[General]
repeat = 100
**.routingRecorder.enabled = true

##### Statistics #####
output-scalar-file = ${resultdir}/${configname}/${iterationvars}-${repetition}.sca
output-vector-file = ${resultdir}/${configname}/${iterationvars}-${repetition}.vec
seed-set = ${repetition}
num-rngs = 3
**.sctp.**.scalar-recording = true # remove annoying statistics from SCTP
**.sctp.**.vector-recording = true # remove annoying statistics from SCTP
**.vector-recording = true
*.ue_*[*].**.recordStatistics = true
*.gnb_*.*.recordStatistics = true
*.*.statistic-recording = true
# intenta activar modos de grabación típicos INET
*.ue_*[*].app[*].packetDelay.result-recording-modes = all
*.ue_*[*].app[*].endToEndDelay.result-recording-modes = all
*.ue_*[*].app[*].packetReceived.result-recording-modes = all
*.ue_*[*].app[*].packetSent.result-recording-modes = all
*.*.throughput.result-recording-modes = all
**.endToEndDelay.result-recording-modes = vector
**.throughput.result-recording-modes = vector
# Contadores como escalares (para pérdida de paquetes)
**.ue_*[*].app[*].packetSent.result-recording-modes = count
**.ue_*[*].app[*].packetReceived.result-recording-modes = count
# Guardar estadísticas de movilidad (incluye velocidad)

##### SCTP configuration #####
**.sctp.nagleEnabled = false      # if true, transmission of small packets will be delayed on the X2
**.sctp.enableHeartbeats = false

##### General PHY parameters #####
**.downlink_interference = true
**.uplink_interference = true
**.ueTxPower = 23dBm
**.eNodeBTxPower = 46dBm
**.targetBler = 0.01
**.blerShift = 5
**.fbPeriod = 40tti # reports CQI every 40ms

##### IPv4 configurator config #####
*.configurator.config = xmldoc("./demo.xml")

#
# ----- #
# Config MultiOperator
#
# General configuration for the multicell, multi-operator system

```

```

# Also see 'demo.xml' for the correct configuration of network interfaces
#
[Config MultiOperator]
description = "Base configuration, do not run this config!"
sim-time-limit=40s
network = simu5g.simulations.NR.mec.multiOperator.singleMec.MultiOperatorSingleMec

##### Floorplan parameters #####
**.mobility.constraintAreaMaxX = 1360m
**.mobility.constraintAreaMaxY = 555m
**.mobility.constraintAreaMinX = 0m
**.mobility.constraintAreaMinY = 250m
**.mobility.constraintAreaMinZ = 0m
**.mobility.constraintAreaMaxZ = 0m
**.mobility.initFromDisplayString = false

##### CA configuration #####
# we configure two carrier components: operator A uses CC 0, while operator B uses CC 1
*.carrierAggregation.numComponentCarriers = 2
*.carrierAggregation.componentCarrier[0].numerologyIndex = ${u=0}
*.carrierAggregation.componentCarrier[0].carrierFrequency = 5.9GHz
*.carrierAggregation.componentCarrier[0].numBands = ${numRbs=50}
*.carrierAggregation.componentCarrier[1].numerologyIndex = ${u}
*.carrierAggregation.componentCarrier[1].carrierFrequency = 4.9GHz
*.carrierAggregation.componentCarrier[1].numBands = ${numRbs}

*.gnb*.cellularNic.numCarriers = 1
*.ue_*[*].cellularNic.numCarriers = 1

*.gnb_A*.cellularNic.channelModel[0].componentCarrierModule = "carrierAggregation.componentCarrier[0]"
*.gnb_B*.cellularNic.channelModel[0].componentCarrierModule = "carrierAggregation.componentCarrier[1]"
*.ue_A[*].cellularNic.nrChannelModel[0].componentCarrierModule =
"carrierAggregation.componentCarrier[0]"
*.ue_B[*].cellularNic.nrChannelModel[0].componentCarrierModule =
"carrierAggregation.componentCarrier[1]"

##### BS position #####
# operator A's and B's gNBs are co-located
*.gnb_*1.mobility.initialX = 350m
*.gnb_*1.mobility.initialY = 520m
*.gnb_*2.mobility.initialX = 390m
*.gnb_*2.mobility.initialY = 415m
*.gnb_*3.mobility.initialX = 836m
*.gnb_*3.mobility.initialY = 530m
*.gnb_*4.mobility.initialX = 870m
*.gnb_*4.mobility.initialY = 370m

##### BS gateway #####
*.gnb_A*.gateway = "upf_A"
*.gnb_B*.gateway = "upf_B"

##### X2 configuration #####
#*.gnb_*2.numX2Apps = 2 # one x2App per peering node
*.gnb*.numX2Apps = 3 # one x2App per peering node
*.gnb*.x2App[*].server.localPort = 5000 + ancestorIndex(1) # Server ports (x2App[0]=5000, x2App[1]=5001, ...)

#### Operator A gNodeB####
*.gnb_A1.x2App[0].client.connectAddress = "gnb_A2%x2ppp0"
*.gnb_A1.x2App[1].client.connectAddress = "gnb_A3%x2ppp0"
*.gnb_A1.x2App[2].client.connectAddress = "gnb_A4%x2ppp0"

*.gnb_A2.x2App[0].client.connectAddress = "gnb_A1%x2ppp0"
*.gnb_A2.x2App[1].client.connectAddress = "gnb_A3%x2ppp0"
*.gnb_A2.x2App[2].client.connectAddress = "gnb_A4%x2ppp0"

*.gnb_A3.x2App[0].client.connectAddress = "gnb_A1%x2ppp0"
*.gnb_A3.x2App[1].client.connectAddress = "gnb_A2%x2ppp0"
*.gnb_A3.x2App[2].client.connectAddress = "gnb_A4%x2ppp0"

*.gnb_A4.x2App[0].client.connectAddress = "gnb_A1%x2ppp0"
*.gnb_A4.x2App[1].client.connectAddress = "gnb_A2%x2ppp0"
*.gnb_A4.x2App[2].client.connectAddress = "gnb_A3%x2ppp0"

```

```

#####Operator B gNodeB#####
*.gnb_B1.x2App[0].client.connectAddress = "gnb_B2%x2ppp0"
*.gnb_B1.x2App[1].client.connectAddress = "gnb_B3%x2ppp0"
*.gnb_B1.x2App[2].client.connectAddress = "gnb_B4%x2ppp0"

*.gnb_B2.x2App[0].client.connectAddress = "gnb_B1%x2ppp0"
*.gnb_B2.x2App[1].client.connectAddress = "gnb_B3%x2ppp0"
*.gnb_B2.x2App[2].client.connectAddress = "gnb_B4%x2ppp0"

*.gnb_B3.x2App[0].client.connectAddress = "gnb_B1%x2ppp0"
*.gnb_B3.x2App[1].client.connectAddress = "gnb_B2%x2ppp0"
*.gnb_B3.x2App[2].client.connectAddress = "gnb_B4%x2ppp0"

*.gnb_B4.x2App[0].client.connectAddress = "gnb_B1%x2ppp0"
*.gnb_B4.x2App[1].client.connectAddress = "gnb_B2%x2ppp0"
*.gnb_B4.x2App[2].client.connectAddress = "gnb_B3%x2ppp0"

##### UE configuration #####
*.numUe_A = 3
*.numUe_B = 2

# UEs associates to the best BS at the beginning of the simulation
*.ue*[*].macCellId = 0
*.ue*[*].masterId = 0
*.ue_A[*].nrMacCellId = 1
*.ue_A[*].nrMasterId = 1
*.ue_B[*].nrMacCellId = 2
*.ue_B[*].nrMasterId = 2
**.dynamicCellAssociation = true
**.enableHandover = true

##### UEs position #####
*.ue*[*].mobility.typename = "LinearMobility"
*.ue*[*].mobility.initialX = intuniform(93,108) * 3m
*.ue*[*].mobility.initialY = intuniform(237,239) * 2m
*.ue*[*].mobility.speed = uniform(12.5mps, 15.28mps)
**.ue*[*].mobility.record = true
*.ue*[*].mobility.initialMovementHeading = 0deg

#####
##### MEC Configuration #####
**.hasRNISupport = true

# tcp settings
**.tcp.typename = "Tcp"
**.tcp.advertisedWindow = 65535      # in bytes, corresponds with the maximal receiver buffer
capacity (Note: normally, NIC queues should be at least this size)
**.tcp.tcpAlgorithmClass = "TcpReno"    # TcpReno/TcpTahoe/TcpNewReno/TcpNoCongestionControl/DumbTcp
**.tcp.sackSupport = true              # Selective Acknowledgment (RFC 2018, 2883, 3517) support
(header option) (SACK will be enabled for a connection if both endpoints support it)

# MEC Hosts
**.mecHost*.virtualisationInfrastructure.ipv4.forwarding = true
*.mecHost*.maxMECAApps = 100          # max ME Apps to instantiate
*.mecHost*.maxRam = 32GB             # max KBytes of Ram
*.mecHost*.maxDisk = 100TB            # max KBytes of Disk Space
*.mecHost*.maxCpuSpeed = 400000        # max CPU
*.mecHost_A*.gateway = "upf_A"
*.mecHost_B*.gateway = "upf_B"

# MEC Services
# MEC host 1 services configurations
*.mecHost*.mecPlatform.numMecServices = 1
*.mecHost*.mecPlatform.mecService[0].typename = "LocationService"
#
*.mecHost_A1.mecPlatform.mecService[0].localAddress = "mecHost_A1.virtualisationInfrastructure"
*.mecHost_B1.mecPlatform.mecService[0].localAddress = "mecHost_B1.virtualisationInfrastructure"
*.mecHost*.mecPlatform.mecService[0].localPort = 10020
#

```

```

#
*.mecHost_A1.mecPlatform.serviceRegistry.localAddress = "mecHost_A1.virtualisationInfrastructure"
*.mecHost_B1.mecPlatform.serviceRegistry.localAddress = "mecHost_B1.virtualisationInfrastructure"
*.mecHost*.mecPlatform.serviceRegistry.localPort = 10021
#


# random number generators configuration
*.mecHost*.mecPlatform.serviceRegistry.rng-0 = 0 # request service time
*.mecHost*.mecPlatform.serviceRegistry.rng-1 = 1 # subscription service time
*.mecHost*.mecPlatform.mecService[0].rng-0 = 0 # request service time
*.mecHost*.mecPlatform.mecService[0].rng-1 = 1 # subscription service time
*.mecHost*.mecPlatform.mecService[0].requestServiceTime = 20us
*.mecHost*.mecPlatform.mecService[0].subscriptionServiceTime = 11us

# MEC Orchestrator & UALCMP configuration
*.mecOrchestrator*.mecApplicationPackageList = ["WarningAlertApp"] # List of MEC app descriptors to
be onboarded at
*.mecHost_A*.mecPlatformManager.mecOrchestrator = "mecOrchestrator_A" # the MECPM needs to know the
MEC orchestrator
*.mecHost_B*.mecPlatformManager.mecOrchestrator = "mecOrchestrator_B" # the MECPM needs to know the
MEC orchestrator
*.ualcmp_A.mecOrchestratorHostname = "mecOrchestrator_A"
*.ualcmp_B.mecOrchestratorHostname = "mecOrchestrator_B"
*.ualcmp_A.gateway = "upf_A"
*.ualcmp_B.gateway = "upf_B"

[Config MultiOperator_OneMecPerOperator]
extends=MultiOperator
description = "Cada operador A y B tiene un solo MEC Host (mecHost_A1 y mecHost_B1) y todos sus gNBs
se conectan a él"
**.ue*[*].app[1].txPacket:vector = false
**.ue*[*].app[1].rxPacket:vector = false
*.enableMecOperatorB = true

# MEC host A maneja todos Los gNBs de operador A
*.mecHost_A1.bsList = ["gnb_A1", "gnb_A2", "gnb_A3", "gnb_A4"]

# MEC host B maneja todos Los gNBs de operador B
*.mecHost_B1.bsList = ["gnb_B1", "gnb_B2", "gnb_B3", "gnb_B4"]

-----UEWarningAlertApp-----
*.ue*[*].numApps = 2
*.ue*[*].app[0].typename = "DeviceApp"
*.ue*[*].app[0].localPort = 4500
*.ue_A[*].app[0].UALCMPAddress = "ualcmp_A"
*.ue_B[*].app[0].UALCMPAddress = "ualcmp_B"
*.ue*[*].app[0].UALCMPPort = 1000
*.ue*[*].app[0].appPackageSource = "ApplicationDescriptors/WarningAlertApp.json"

*.ue*[*].app[1].typename = "UEWarningAlertApp"
*.ue_A[*].app[1].deviceAppAddress = "ue_A["+string(ancestorIndex(1))+"]"
*.ue_B[*].app[1].deviceAppAddress = "ue_B["+string(ancestorIndex(1))+"]"
*.ue*[*].app[1].deviceAppPort = 4500
*.ue*[*].app[1].logger = true
*.ue*[*].app[1].startTime = 1s
*.ue*[*].app[1].period = 0.001s
*.ue*[*].app[1].stopTime = 41s
*.ue*[*].app[1].positionX = 600m
*.ue*[*].app[1].positionY = 600m
*.ue*[*].app[1].radius = 1000m

-----#

```

Anexo 5. Archivo MultiOperatorSingleMec.ned para simulación de una red 5G Single MEC

```

//  

//  

//  

Simu5G

```

```

// Authors: Giovanni Nardini, Giovanni Stea, Antonio Virdis (University of Pisa)
//
// This file is part of a software released under the license included in file
// "license.pdf". Please read LICENSE and README files before using it.
// The above files and the present reference are part of the software itself,
// and cannot be removed from it.
//

package simu5g.simulations.NR.mec.multiOperator.singleMec;

import inet.networklayer.configurator.ipv4.Ipv4NetworkConfigurator;
import inet.networklayer.ipv4.RoutingTableRecorder;
import inet.node.ethernet.Eth10G;
import inet.node.inet.Router;
import simu5g.common.binder.Binder;
import simu5g.common.carrierAggregation.CarrierAggregation;
import simu5g.nodes.NR.NRUe;
import simu5g.nodes.NR.gNodeB;
import simu5g.nodes.Upf;
import simu5g.nodes.backgroundCell.BackgroundCell;
import simu5g.nodes.mec.MECHost;
import simu5g.nodes.mec.MECOrchestrator.MecOrchestrator;
import simu5g.nodes.mec.UALCMP.UALCMP;
import simu5g.world.radio.LteChannelControl;

network MultiOperatorSingleMec
{
    parameters:
        int numUe_A = default(1);
        int numUe_B = default(1);
        int numBgCells = default(0);
        bool enableMecOperatorB = default(true);
        @display("i=block/network2;bgb=1358,565;bg=i=background/malecon");

    submodules:
        channelControl: LteChannelControl {
            @display("p=50,25;is=s");
        }
        routingRecorder: RoutingTableRecorder {
            @display("p=50,75;is=s");
        }
        configurator: Ipv4NetworkConfigurator {
            @display("p=50,125;is=s");
        }
        binder: Binder {
            @display("p=50,175;is=s");
        }
        carrierAggregation: CarrierAggregation {
            @display("p=50,258;is=s");
        }
        router: Router {
            @display("p=750,73;is=vs");
        }
        upf_A: Upf {
            @display("p=625,173;is=vs");
        }
        upf_B: Upf {
            @display("p=875,173;is=vs");
        }

        //iUpf_AX
        iUpf_A1: Upf {
            @display("p=623,281;is=vs");
        }
        //iUpf_BX
        iUpf_B1: Upf {
            @display("p=874,281;is=vs");
        }

        gnb_A1: gNodeB {
            @display("p=274,454;is=vs");
        }
        gnb_A2: gNodeB {

```

```

        @display("p=800,452;is=vs");
    }
    gnb_A3: gNodeB {
        @display("p=1245,452;is=vs");
    }
    gnb_A4: gNodeB {
        @display("p=1245,452;is=vs");
    }
    gnb_B1: gNodeB {
        @display("p=330,514;is=vs");
    }
    gnb_B2: gNodeB {
        @display("p=848,513;is=vs");
    }
    gnb_B3: gNodeB {
        @display("p=1304,513;is=vs");
    }
    gnb_B4: gNodeB {
        @display("p=1304,513;is=vs");
    }
    bgCell[numBgCells]: BackgroundCell {
        @display("p=138,630;is=vl");
    }
    ue_A[numUe_A]: NRUE {
        @display("p=717,691;is=vs");
    }
    ue_B[numUe_B]: NRUE {
        @display("p=937,691;is=vs");
    }

    //# MEC modules
    meCHost_A1: MECHost {
        @display("p=463,281");
    }

    meCHost_B1: MECHost if enableMecOperatorB {
        @display("p=994,281");
    }

    meCOrchestrator_A: MecOrchestrator {
        @display("p=450,73");
    }
    meCOrchestrator_B: MecOrchestrator if enableMecOperatorB {
        @display("p=1050,73");
    }
    ualcmp_A: UALCMP {
        @display("p=600,73");
    }
    ualcmp_B: UALCMP if enableMecOperatorB {
        @display("p=900,73");
    }
}

```

connections:

```

    //# 5G Core Network connections
    //Operator A
    upf_A.pppg++ <--> Eth10G <--> iUpf_A1.pppg++;
    iUpf_A1.pppg++ <--> Eth10G <--> meCHost_A1.ppp++;

    iUpf_A1.pppg++ <--> Eth10G <--> gnb_A1.ppp;
    iUpf_A1.pppg++ <--> Eth10G <--> gnb_A2.ppp;
    iUpf_A1.pppg++ <--> Eth10G <--> gnb_A3.ppp;
    iUpf_A1.pppg++ <--> Eth10G <--> gnb_A4.ppp;

    //Operator B
    upf_B.pppg++ <--> Eth10G <--> iUpf_B1.pppg++;
    iUpf_B1.pppg++ <--> Eth10G <--> meCHost_B1.ppp++ if enableMecOperatorB;

```

```

iUpf_B1.pppg++ <--> Eth10G <--> gnb_B1.ppp;
iUpf_B1.pppg++ <--> Eth10G <--> gnb_B2.ppp;
iUpf_B1.pppg++ <--> Eth10G <--> gnb_B3.ppp;
iUpf_B1.pppg++ <--> Eth10G <--> gnb_B4.ppp;

//# MEC-related connections
router.pppg++ <--> Eth10G <--> upf_A.filterGate;
ualcmp_A.pppg++ <--> Eth10G <--> router.pppg++;
ualcmp_A.toMecOrchestrator --> mecOrchestrator_A.fromUALCMP;
ualcmp_A.fromMecOrchestrator <--> mecOrchestrator_A.toUALCMP;

router.pppg++ <--> Eth10G <--> upf_B.filterGate;
ualcmp_B.pppg++ <--> Eth10G <--> router.pppg++ if enableMecOperatorB;
ualcmp_B.toMecOrchestrator --> mecOrchestrator_B.fromUALCMP if enableMecOperatorB;
ualcmp_B.fromMecOrchestrator <--> mecOrchestrator_B.toUALCMP if enableMecOperatorB;

//# X2 connections
// Operator A gNodeBs interconnections
gnb_A1.x2++ <--> Eth10G <--> gnb_A2.x2++;
gnb_A1.x2++ <--> Eth10G <--> gnb_A3.x2++;
gnb_A1.x2++ <--> Eth10G <--> gnb_A4.x2++;
gnb_A2.x2++ <--> Eth10G <--> gnb_A3.x2++;
gnb_A2.x2++ <--> Eth10G <--> gnb_A4.x2++;
gnb_A3.x2++ <--> Eth10G <--> gnb_A4.x2++;

// Operator B gNodeBs interconnections
gnb_B1.x2++ <--> Eth10G <--> gnb_B2.x2++;
gnb_B1.x2++ <--> Eth10G <--> gnb_B3.x2++;
gnb_B1.x2++ <--> Eth10G <--> gnb_B4.x2++;
gnb_B2.x2++ <--> Eth10G <--> gnb_B3.x2++;
gnb_B2.x2++ <--> Eth10G <--> gnb_B4.x2++;
gnb_B3.x2++ <--> Eth10G <--> gnb_B4.x2++;

}


```

Anexo 6. Archivo demo.xml para simulación de una red 5G Single MEC

```

<config>
  <!-- Assign addresses 10.x.x.x to operator A's nodes -->
  <interface hosts="gnb_A*" names="cellular" address="10.x.x.x" netmask="255.0.0.0"/>
  <interface hosts="ue_A*" names="cellular" address="10.x.x.x" netmask="255.0.0.0"/>

  <!-- Assign addresses 11.x.x.x to operator B's nodes -->
  <interface hosts="gnb_B*" names="cellular" address="11.x.x.x" netmask="255.0.0.0"/>
  <interface hosts="ue_B*" names="cellular" address="11.x.x.x" netmask="255.0.0.0"/>

  <!-- 'wireless' tags specify which hosts are within the same wireless network (i.e. which nodes
have
      a direct link between them). This configuration prevents the CellularNetworkConfigurator; to
setup
      routes from gnb_A(B)* and ue_B(A)*.
      If not specified, the CellularNetworkConfigurator assumes that all cellular interfaces are at
1-hop
      distance and routing tables are built accordingly -->
  <wireless hosts="gnb_A* ue_A*" interfaces="cellular"/>
  <wireless hosts="gnb_B* ue_B*" interfaces="cellular"/>

  <!-- Assign 192.168.x.x addresses to all other interfaces -->
  <interface hosts="**" address="192.168.x.x" netmask="255.255.255.0"/>
</config>

```

Anexo 7. Archivo WarningAlertApp.json para simulación de una red 5G Single MEC

```

{
  "appDid" : "WAMECAPP",
  "appName" : "MECWarningAlertApp",

```

```

"appProvider" : "simu5g.apps.mec.WarningAlert.MECWarningAlertApp",
"appInfoName" : "appInfoName_",
"appDescription" : "appDescription_",
"virtualComputeDescriptor":{

    "virtualDisk": 10,
    "virtualCpu" : 1500,
    "virtualMemory":10
},
"appServiceRequired": [
{
    "ServiceDependency" :{
        "serName" : "LocationService",
        "version" : "v2",
        "serCategory": "Location"
    }
}
],
"omnetppServiceRequired": "MEWarningAlertService"
}

```

Anexo 8. Archivo omnetpp.ini para simulación de una red 5G Multi MEC

```

[General]
repeat = 100
**.routingRecorder.enabled = true

##### Statistics #####
output-scalar-file = ${resultdir}/${configname}/${iterationvars}-${repetition}.sca
output-vector-file = ${resultdir}/${configname}/${iterationvars}-${repetition}.vec
seed-set = ${repetition}
num-rngs = 3
**.sctp.**.scalar-recording = true # remove annoying statistics from SCTP
**.sctp.**.vector-recording = true # remove annoying statistics from SCTP
**.vector-recording = true
**.ue*[*].**.recordStatistics = true
**.gnb_*.*.recordStatistics = true
**.throughput.result-recording-modes = all
**.endToEndDelay.result-recording-modes = vector
# Contadores como escalares (para pérdida de paquetes)
**.ue_*[*].app[*].packetSent.result-recording-modes = count
**.ue_*[*].app[*].packetReceived.result-recording-modes = count
**.nrRlc.*.recordScalar = true
**.nrRlc.*.recordVector = true
**.ue_*[*].**.recordStatistics = true
**.gnb_*.*.recordStatistics = true
*.*.**.statistic-recording = true
*.ue_*[*].app[*].packetDelay.result-recording-modes = all
*.ue_*[*].app[*].endToEndDelay.result-recording-modes = all
*.ue_*[*].app[*].packetReceived.result-recording-modes = all
*.ue_*[*].app[*].packetSent.result-recording-modes = all

**.ue_*[*].rrc.servingCellRsrpVector.record = true
**.ue_*[*].rrc.servingCellRsrpVector.recordStatistics = true

# Enable recording of neighbor cell RSRP at each UE

```

```

**.ue*[*].rrc.neighborCellRsrpVector.record = true
**.ue*[*].rrc.neighborCellRsrpVector.recordStatistics = true

# Optional: handover events
**.ue*[*].rrc.handoverEventVector.record = true
**.ue*[*].rrc.handoverEventVector.recordStatistics = true
# Optionally, record which UE triggered handover
**.gnb*[*].rrc.recordHandoverEvent = true

##### SCTP configuration #####
**.sctp.nagleEnabled = false          # if true, transmission of small packets will be delayed on the
X2
**.sctp.enableHeartbeats = false

##### General PHY parameters #####
**.downlink_interference = true
**.uplink_interference = true
**.ueTxPower = 23dBm
**.eNodeBTxPower = 46dBm
**.targetBler = 0.01
**.blerShift = 5
**.fbPeriod = 40tti    # reports CQI every 40ms

##### IPv4 configurator config #####
*.configurator.config = xmldoc("./demo.xml")

# -----
# Config MultiOperator
#
# General configuration for the multicell, multi-operator system
# Also see 'demo.xml' for the correct configuration of network interfaces
#
[Config MultiOperator]
description = "Base configuration, do not run this config!"
sim-time-limit=40s
network = simu5g.simulations.NR.mec.multiOperator.multiMec.MultiOperatorMultiMec

##### Floorplan parameters #####
**.mobility.constraintAreaMaxX = 1360m
**.mobility.constraintAreaMaxY = 555m
**.mobility.constraintAreaMinX = 0m
**.mobility.constraintAreaMinY = 250m
**.mobility.constraintAreaMinZ = 0m
**.mobility.constraintAreaMaxZ = 0m
**.mobility.initFromDisplayString = false

##### CA configuration #####
# we configure two carrier components: operator A uses CC 0, while operator B uses CC 1
*.carrierAggregation.numComponentCarriers = 2
*.carrierAggregation.componentCarrier[0].numerologyIndex = ${u=0}
*.carrierAggregation.componentCarrier[0].carrierFrequency = 5.9GHz
*.carrierAggregation.componentCarrier[0].numBands = ${numRbs=50}
*.carrierAggregation.componentCarrier[1].numerologyIndex = ${u}
*.carrierAggregation.componentCarrier[1].carrierFrequency = 4.9GHz
*.carrierAggregation.componentCarrier[1].numBands = ${numRbs}

*.gnb*.cellularNic.numCarriers = 1
*.ue_*[*].cellularNic.numCarriers = 1

*.gnb_A*.cellularNic.channelModel[0].componentCarrierModule = "carrierAggregation.componentCarrier[0]"
*.gnb_B*.cellularNic.channelModel[0].componentCarrierModule = "carrierAggregation.componentCarrier[1]"
*.ue_A[*].cellularNic.nrChannelModel[0].componentCarrierModule =
"carrierAggregation.componentCarrier[0]"
*.ue_B[*].cellularNic.nrChannelModel[0].componentCarrierModule =
"carrierAggregation.componentCarrier[1]"

##### BS position #####
# operator A's and B's gNBs are co-located
*.gnb_*1.mobility.initialX = 350m
*.gnb_*1.mobility.initialY = 520m

```

```

*.gnb_*2.mobility.initialX = 390m
*.gnb_*2.mobility.initialY = 415m
*.gnb_*3.mobility.initialX = 836m
*.gnb_*3.mobility.initialY = 530m
*.gnb_*4.mobility.initialX = 870m
*.gnb_*4.mobility.initialY = 370m

##### BS gateway #####
*.gnb_A*.gateway = "upf_A"
*.gnb_B*.gateway = "upf_B"

##### X2 configuration #####
#*.gnb_*2.numX2Apps = 2 # one x2App per peering node
*.gnb*.numX2Apps = 3 # one x2App per peering node
*.gnb*.x2App[*].server.localPort = 5000 + ancestorIndex(1) # Server ports (x2App[0]=5000, x2App[1]=5001, ...)

#### Operator A gNodeB####
*.gnb_A1.x2App[0].client.connectAddress = "gnb_A2%x2ppp0"
*.gnb_A1.x2App[1].client.connectAddress = "gnb_A3%x2ppp0"
*.gnb_A1.x2App[2].client.connectAddress = "gnb_A4%x2ppp0"

*.gnb_A2.x2App[0].client.connectAddress = "gnb_A1%x2ppp0"
*.gnb_A2.x2App[1].client.connectAddress = "gnb_A3%x2ppp0"
*.gnb_A2.x2App[2].client.connectAddress = "gnb_A4%x2ppp0"

*.gnb_A3.x2App[0].client.connectAddress = "gnb_A1%x2ppp0"
*.gnb_A3.x2App[1].client.connectAddress = "gnb_A2%x2ppp0"
*.gnb_A3.x2App[2].client.connectAddress = "gnb_A4%x2ppp0"

*.gnb_A4.x2App[0].client.connectAddress = "gnb_A1%x2ppp0"
*.gnb_A4.x2App[1].client.connectAddress = "gnb_A2%x2ppp0"
*.gnb_A4.x2App[2].client.connectAddress = "gnb_A3%x2ppp0"

#### Operator B qNodeB####
*.gnb_B1.x2App[0].client.connectAddress = "gnb_B2%x2ppp0"
*.gnb_B1.x2App[1].client.connectAddress = "gnb_B3%x2ppp0"
*.gnb_B1.x2App[2].client.connectAddress = "gnb_B4%x2ppp0"

*.gnb_B2.x2App[0].client.connectAddress = "gnb_B1%x2ppp0"
*.gnb_B2.x2App[1].client.connectAddress = "gnb_B3%x2ppp0"
*.gnb_B2.x2App[2].client.connectAddress = "gnb_B4%x2ppp0"

*.gnb_B3.x2App[0].client.connectAddress = "gnb_B1%x2ppp0"
*.gnb_B3.x2App[1].client.connectAddress = "gnb_B2%x2ppp0"
*.gnb_B3.x2App[2].client.connectAddress = "gnb_B4%x2ppp0"

*.gnb_B4.x2App[0].client.connectAddress = "gnb_B1%x2ppp0"
*.gnb_B4.x2App[1].client.connectAddress = "gnb_B2%x2ppp0"
*.gnb_B4.x2App[2].client.connectAddress = "gnb_B3%x2ppp0"

##### UE configuration #####
*.numUe_A = 15
*.numUe_B = 15

# UEs associates to the best BS at the beginning of the simulation
*.ue*[*].macCellId = 0
*.ue*[*].masterId = 0
*.ue_A[*].nrMacCellId = 1
*.ue_A[*].nrMasterId = 1
*.ue_B[*].nrMacCellId = 4
*.ue_B[*].nrMasterId = 4
**.dynamicCellAssociation = true
**.enableHandover = true
**.logHandover = true

##### UEs position #####
*.ue*.mobility.typename = "LinearMobility"
*.ue*[*].mobility.initialX = intuniform(93,118) * 3m
*.ue*[*].mobility.initialY = intuniform(237,239) * 2m
*.ue*[*].mobility.speed = uniform(12.5mps, 15.28mps)
*.ue*[*].mobility.speed.record = true
*.ue*[*].mobility.initialMovementHeading = 0deg

```

```

##### MEC Configuration #####
**.hasRNISupport = true

# tcp settings
**.tcp.typename = "Tcp"
**.tcp.advertisedWindow = 65535          # in bytes, corresponds with the maximal receiver buffer
capacity (Note: normally, NIC queues should be at least this size)
**.tcp.tcpAlgorithmClass = "TcpReno"      # TcpReno/TcpTahoe/TcpNewReno/TcpNoCongestionControl/DumbTcp
**.tcp.sackSupport = true                 # Selective Acknowledgment (RFC 2018, 2883, 3517) support
(header option) (SACK will be enabled for a connection if both endpoints support it)

# MEC Hosts
**.mecHost*.virtualisationInfrastructure.ipv4.forwarding = true
*.mecHost*.maxMECAapps = 100             # max ME Apps to instantiate
*.mecHost*.maxRam = 32GB                  # max KBytes of Ram
*.mecHost*.maxDisk = 100TB                # max KBytes of Disk Space
*.mecHost*.maxCpuSpeed = 400000           # max CPU
*.mecHost_A*.gateway = "upf_A"
*.mecHost_B*.gateway = "upf_B"

# MEC Services
# MEC host 1 services configurations
*.mecHost*.mecPlatform.numMecServices = 1
*.mecHost*.mecPlatform.mecService[0].typename = "LocationService"
#
*.mecHost_A1.mecPlatform.mecService[0].localAddress = "mecHost_A1.virtualisationInfrastructure"
*.mecHost_A2.mecPlatform.mecService[0].localAddress = "mecHost_A2.virtualisationInfrastructure"
*.mecHost_A3.mecPlatform.mecService[0].localAddress = "mecHost_A3.virtualisationInfrastructure"
*.mecHost_A4.mecPlatform.mecService[0].localAddress = "mecHost_A4.virtualisationInfrastructure"

*.mecHost_B1.mecPlatform.mecService[0].localAddress = "mecHost_B1.virtualisationInfrastructure"
*.mecHost_B2.mecPlatform.mecService[0].localAddress = "mecHost_B2.virtualisationInfrastructure"
*.mecHost_B3.mecPlatform.mecService[0].localAddress = "mecHost_B3.virtualisationInfrastructure"
*.mecHost_B4.mecPlatform.mecService[0].localAddress = "mecHost_B4.virtualisationInfrastructure"

*.mecHost*.mecPlatform.mecService[0].localPort = 10020

*.mecHost_A1.mecPlatform.serviceRegistry.localAddress = "mecHost_A1.virtualisationInfrastructure"
*.mecHost_A2.mecPlatform.serviceRegistry.localAddress = "mecHost_A2.virtualisationInfrastructure"
*.mecHost_A3.mecPlatform.serviceRegistry.localAddress = "mecHost_A3.virtualisationInfrastructure"
*.mecHost_A4.mecPlatform.serviceRegistry.localAddress = "mecHost_A4.virtualisationInfrastructure"

*.mecHost_B1.mecPlatform.serviceRegistry.localAddress = "mecHost_B1.virtualisationInfrastructure"
*.mecHost_B2.mecPlatform.serviceRegistry.localAddress = "mecHost_B2.virtualisationInfrastructure"
*.mecHost_B3.mecPlatform.serviceRegistry.localAddress = "mecHost_B3.virtualisationInfrastructure"
*.mecHost_B4.mecPlatform.serviceRegistry.localAddress = "mecHost_B4.virtualisationInfrastructure"

*.mecHost*.mecPlatform.serviceRegistry.localPort = 10021

# random number generators configuration
*.mecHost*.mecPlatform.serviceRegistry.rng-0 = 0 # request service time
*.mecHost*.mecPlatform.serviceRegistry.rng-1 = 1 # subscription service time
*.mecHost*.mecPlatform.mecService[0].rng-0 = 0 # request service time
*.mecHost*.mecPlatform.mecService[0].rng-1 = 1 # subscription service time
*.mecHost*.mecPlatform.mecService[0].requestServiceTime = 20us
*.mecHost*.mecPlatform.mecService[0].subscriptionServiceTime = 11us

# MEC Orchestrator & UALCMP configuration
*.mecOrchestrator_A.mecHostList = ["mecHost_A1", "mecHost_A2", "mecHost_A3", "mecHost_A4"]
# MEC hosts associated to the MEC system
*.mecOrchestrator_B.mecHostList = ["mecHost_B1", "mecHost_B2", "mecHost_B3", "mecHost_B4"]
# MEC hosts associated to the MEC system
*.mecOrchestrator*.mecApplicationPackageList = ["WarningAlertApp"]    # List of MEC app descriptors to
be onboarded at
*.mecHost_A*.mecPlatformManager.mecOrchestrator = "mecOrchestrator_A" # the MECPM needs to know the
MEC orchestrator
*.mecHost_B*.mecPlatformManager.mecOrchestrator = "mecOrchestrator_B" # the MECPM needs to know the
MEC orchestrator
*.ualcmp_A.mecOrchestratorHostname = "mecOrchestrator_A"
*.ualcmp_B.mecOrchestratorHostname = "mecOrchestrator_B"
*.ualcmp_A.gateway = "upf_A"
*.ualcmp_B.gateway = "upf_B"

```

```

# -----
# 
# MultiOperator_MultiMec
#
# In this configuration, UEs connected to operator A's network communicate with operator A's MEC Host,
# whereas UEs connected to operator B's network communicate with operator B's MEC host
#
[Config MultiOperator_MultiMec]
extends=MultiOperator
description = "UEs connected to operator A's network communicate with operator A's MEC Host, UEs
connected to B with B's MEC host"

*.enableMecOperatorB = true

# MEC host A handles operator A's gNBs
# MEC host B handles operator B's gNBs
*.mecHost_A1.bsList = ["gnb_A1"]
*.mecHost_B1.bsList = ["gnb_B1"]
*.mecHost_A2.bsList = ["gnb_A2"]
*.mecHost_B2.bsList = ["gnb_B2"]
*.mecHost_A3.bsList = ["gnb_A3"]
*.mecHost_B3.bsList = ["gnb_B3"]
*.mecHost_A4.bsList = ["gnb_A4"]
*.mecHost_B4.bsList = ["gnb_B4"]

-----UEWarningAlertApp-----
*.ue*[*].numApps = 2
*.ue*[*].app[0].typename = "DeviceApp"
*.ue*[*].app[0].localPort = 4500
*.ue_A[*].app[0].UALCMPAddress = "ualcmp_A"
*.ue_B[*].app[0].UALCMPAddress = "ualcmp_B"
*.ue*[*].app[0].UALCMPPort = 1000
*.ue*[*].app[0].appPackageSource = "ApplicationDescriptors/WarningAlertApp.json"

*.ue*[*].app[1].typename = "UEWarningAlertApp"
*.ue_A[*].app[1].deviceAppAddress = "ue_A["+string(ancestorIndex(1))+"]"
*.ue_B[*].app[1].deviceAppAddress = "ue_B["+string(ancestorIndex(1))+"]"
*.ue*[*].app[1].deviceAppPort = 4500
*.ue*[*].app[1].logger = true
*.ue*[*].app[1].startTime = 1s
*.ue*[*].app[1].period = 0.01s
*.ue*[*].app[1].stopTime = 41s
*.ue*[*].app[1].positionX = 600m
*.ue*[*].app[1].positionY = 600m
*.ue*[*].app[1].radius = 1000m

```

Anexo 9. Archivo MultiOperatorMultiMec.ned para simulación de una red 5G Multi MEC

```

// 
// Simu5G
//
// Authors: Giovanni Nardini, Giovanni Stea, Antonio Virdis (University of Pisa)
//
// This file is part of a software released under the license included in file
// "license.pdf". Please read LICENSE and README files before using it.
// The above files and the present reference are part of the software itself,
// and cannot be removed from it.
//

package simu5g.simulations.NR.mec.multiOperator.multiMec;

import inet.networklayer.configurator.ipv4.Ipv4NetworkConfigurator;
import inet.networklayer.ipv4.RoutingTableRecorder;
import inet.node.ethernet.Eth10G;
import inet.node.inet.Router;
import simu5g.common.binder.Binder;
import simu5g.common.carrierAggregation.CarrierAggregation;
import simu5g.nodes.NR.NRUe;
import simu5g.nodes.NR.gNodeB;
import simu5g.nodes.Upf;
import simu5g.nodes.backgroundCell.BackgroundCell;

```

```

import simu5g.nodes.mec.MECHost;
import simu5g.nodes.mec.MECOrchestrator.MecOrchestrator;
import simu5g.nodes.mec.UALCMP.UALCMP;
import simu5g.world.radio.LteChannelControl;
import inet.applications.tcpapp.TcpBasicClientApp;
import inet.applications.udpapp.UdpBasicApp;

network MultiOperatorMultiMec
{
    parameters:
        int numUe_A = default(1);
        int numUe_B = default(1);
        int numBgCells = default(0);
        bool enableMecOperatorB = default(true);
        @display("i=block/network2;bgb=1358,565;bgc=background/malecon");
    submodules:
        channelControl: LteChannelControl {
            @display("p=50,25;is=s");
        }
        routingRecorder: RoutingTableRecorder {
            @display("p=50,75;is=s");
        }
        configurator: Ipv4NetworkConfigurator {
            @display("p=50,125;is=s");
        }
        binder: Binder {
            @display("p=50,175;is=s");
        }
        carrierAggregation: CarrierAggregation {
            @display("p=50,258;is=s");
        }
        router: Router {
            @display("p=750,73;is=vs");
        }
        upf_A: Upf {
            @display("p=625,173;is=vs");
        }
        upf_B: Upf {
            @display("p=875,173;is=vs");
        }

        //iUpf_AX
        iUpf_A1: Upf {
            @display("p=443,305;is=vs");
        }
        iUpf_A2: Upf {
            @display("p=444,255;is=vs");
        }
        iUpf_A3: Upf {
            @display("p=443,205;is=vs");
        }
        iUpf_A4: Upf {
            @display("p=446,155;is=vs");
        }

        //iUpf_BX
        iUpf_B1: Upf {
            @display("p=1210,305;is=vs");
        }
        iUpf_B2: Upf {
            @display("p=1210,255;is=vs");
        }
        iUpf_B3: Upf {
            @display("p=1210,205;is=vs");
        }
        iUpf_B4: Upf {
            @display("p=1210,155;is=vs");
        }

        gnb_A1: gNodeB {
            @display("p=292,452;is=vs");
        }
        gnb_A2: gNodeB {
    
```

```

        @display("p=800,452;is=vs");
    }
    gnb_A3: gNodeB {
        @display("p=1245,452;is=vs");
    }
    gnb_A4: gNodeB {
        @display("p=1245,452;is=vs");
    }
    gnb_B1: gNodeB {
        @display("p=330,514;is=vs");
    }
    gnb_B2: gNodeB {
        @display("p=848,513;is=vs");
    }
    gnb_B3: gNodeB {
        @display("p=1304,513;is=vs");
    }
    gnb_B4: gNodeB {
        @display("p=1304,513;is=vs");
    }
    bgCell[numBgCells]: BackgroundCell {
        @display("p=100,468;is=vl");
    }
    ue_A[numUe_A]: NRUE {
        @display("p=249,477;is=vs");
    }
    ue_B[numUe_B]: NRUE {
        @display("p=249,426;is=vs");
    }

    //# MEC modules
    meCHost_A1: MECHost {
        @display("p=272,306;is=vs");
    }
    meCHost_A2: MECHost {
        @display("p=271,257;is=vs");
    }
    meCHost_A3: MECHost {
        @display("p=271,206;is=vs");
    }
    meCHost_A4: MECHost {
        @display("p=271,157;is=vs");
    }

    meCHost_B1: MECHost if enableMecOperatorB {
        @display("p=1310,305;is=vs");
    }

    meCHost_B2: MECHost if enableMecOperatorB {
        @display("p=1310,255;is=vs");
    }

    meCHost_B3: MECHost if enableMecOperatorB {
        @display("p=1310,205;is=vs");
    }
    meCHost_B4: MECHost if enableMecOperatorB {
        @display("p=1310,155;is=vs");
    }

    meCOrchestrator_A: MecOrchestrator {
        @display("p=450,73");
    }
    meCOrchestrator_B: MecOrchestrator if enableMecOperatorB {
        @display("p=1050,73");
    }
    ualcmp_A: UALCMP {
        @display("p=600,73");
    }
    ualcmp_B: UALCMP if enableMecOperatorB {
        @display("p=900,73");
    }
}

```

connections:

```

## 5G Core Network connections
//Operator A
upf_A.pppg++ <--> Eth10G <--> iUpf_A1.pppg++;
upf_A.pppg++ <--> Eth10G <--> iUpf_A2.pppg++;
upf_A.pppg++ <--> Eth10G <--> iUpf_A3.pppg++;
upf_A.pppg++ <--> Eth10G <--> iUpf_A4.pppg++;

iUpf_A1.pppg++ <--> Eth10G <--> mecHost_A1.ppp++;
iUpf_A2.pppg++ <--> Eth10G <--> mecHost_A2.ppp++;
iUpf_A3.pppg++ <--> Eth10G <--> mecHost_A3.ppp++;
iUpf_A4.pppg++ <--> Eth10G <--> mecHost_A4.ppp++;

iUpf_A1.pppg++ <--> Eth10G <--> gnb_A1.ppp;
iUpf_A2.pppg++ <--> Eth10G <--> gnb_A2.ppp;
iUpf_A3.pppg++ <--> Eth10G <--> gnb_A3.ppp;
iUpf_A4.pppg++ <--> Eth10G <--> gnb_A4.ppp;

//Operator B
upf_B.pppg++ <--> Eth10G <--> iUpf_B1.pppg++;
upf_B.pppg++ <--> Eth10G <--> iUpf_B2.pppg++;
upf_B.pppg++ <--> Eth10G <--> iUpf_B3.pppg++;
upf_B.pppg++ <--> Eth10G <--> iUpf_B4.pppg++;

iUpf_B1.pppg++ <--> Eth10G <--> mecHost_B1.ppp++ if enableMecOperatorB;
iUpf_B2.pppg++ <--> Eth10G <--> mecHost_B2.ppp++ if enableMecOperatorB;
iUpf_B3.pppg++ <--> Eth10G <--> mecHost_B3.ppp++ if enableMecOperatorB;
iUpf_B4.pppg++ <--> Eth10G <--> mecHost_B4.ppp++ if enableMecOperatorB;

iUpf_B1.pppg++ <--> Eth10G <--> gnb_B1.ppp;
iUpf_B2.pppg++ <--> Eth10G <--> gnb_B2.ppp;
iUpf_B3.pppg++ <--> Eth10G <--> gnb_B3.ppp;
iUpf_B4.pppg++ <--> Eth10G <--> gnb_B4.ppp;

## MEC-related connections
router.pppg++ <--> Eth10G <--> upf_A.filterGate;
ualcmp_A.ppp++ <--> Eth10G <--> router.pppg++;
ualcmp_A.toMecOrchestrator --> mecOrchestrator_A.fromUALCMP;
ualcmp_A.fromMecOrchestrator <--> mecOrchestrator_A.toUALCMP;

router.pppg++ <--> Eth10G <--> upf_B.filterGate;
ualcmp_B.ppp++ <--> Eth10G <--> router.pppg++ if enableMecOperatorB;
ualcmp_B.toMecOrchestrator --> mecOrchestrator_B.fromUALCMP if enableMecOperatorB;
ualcmp_B.fromMecOrchestrator <--> mecOrchestrator_B.toUALCMP if enableMecOperatorB;

## X2 connections
// Operator A gNodeBs interconnections
gnb_A1.x2++ <--> Eth10G <--> gnb_A2.x2++;
gnb_A1.x2++ <--> Eth10G <--> gnb_A3.x2++;
gnb_A1.x2++ <--> Eth10G <--> gnb_A4.x2++;
gnb_A2.x2++ <--> Eth10G <--> gnb_A3.x2++;
gnb_A2.x2++ <--> Eth10G <--> gnb_A4.x2++;
gnb_A3.x2++ <--> Eth10G <--> gnb_A4.x2++;

// Operator B gNodeBs interconnections
gnb_B1.x2++ <--> Eth10G <--> gnb_B2.x2++;
gnb_B1.x2++ <--> Eth10G <--> gnb_B3.x2++;
gnb_B1.x2++ <--> Eth10G <--> gnb_B4.x2++;
gnb_B2.x2++ <--> Eth10G <--> gnb_B3.x2++;
gnb_B2.x2++ <--> Eth10G <--> gnb_B4.x2++;
gnb_B3.x2++ <--> Eth10G <--> gnb_B4.x2++;

}

}

```

Anexo 10. Archivo demo.xml para simulación de una red 5G Multi MEC

```

<config>
  <!-- Assign addresses 10.x.x.x to operator A's nodes -->
  <interface hosts="gnb_A*" names="cellular" address="10.x.x.x" netmask="255.0.0.0"/>
  <interface hosts="ue_A*" names="cellular" address="10.x.x.x" netmask="255.0.0.0"/>

  <!-- Assign addresses 11.x.x.x to operator B's nodes -->
  <interface hosts="gnb_B*" names="cellular" address="11.x.x.x" netmask="255.0.0.0"/>
  <interface hosts="ue_B*" names="cellular" address="11.x.x.x" netmask="255.0.0.0"/>

```

```

<!-- 'wireless' tags specify which hosts are within the same wireless network (i.e. which nodes
have
      a direct link between them). This configuration prevents the CellularNetworkConfigurator; to
setup
      routes from gnb_A(B)* and ue_B(A)*.
      If not specified, the CellularNetworkConfigurator assumes that all cellular interfaces are at
1-hop
      distance and routing tables are built accordingly -->
<wireless hosts="gnb_A* ue_A*" interfaces="cellular"/>
<wireless hosts="gnb_B* ue_B*" interfaces="cellular"/>

<!-- Assign 192.168.x.x addresses to all other interfaces -->
<interface hosts="**" address="192.168.x.x" netmask="255.255.255.0"/>
</config>

```

Anexo 11. Archivo WarningAlertApp.json para simulación de una red 5G Multi MEC

```

{
  "appDid" : "WAMECAPP",
  "appName" : "MECWarningAlertApp",
  "appProvider" : "simu5g.apps.mec.WarningAlert.MECWarningAlertApp",
  "appInfoName" : "appInfoName_",
  "appDescription" : "appDescription_",
  "virtualComputeDescriptor": {
    "virtualDisk": 10,
    "virtualCpu" : 1500,
    "virtualMemory":10
  },
  "appServiceRequired": [
    {
      "ServiceDependency" :{
        "serName" : "LocationService",
        "version" : "v2",
        "serCategory": "Location"
      }
    }
  ],
  "omnetppServiceRequired": "MEWarningAlertService"
}

```