



UNIVERSIDAD TÉCNICA DE AMBATO

FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E INDUSTRIAL

CARRERA DE INGENIERÍA EN ELECTRÓNICA Y COMUNICACIONES

TEMA:

“ANÁLISIS DEL COMPORTAMIENTO DE LA COMUNICACIÓN INALÁMBRICA EN EL SISTEMA INTEGRADO AMBULATORIO DE PRESIÓN ARTERIAL.”

Trabajo de Graduación. Modalidad: Proyecto de Investigación, presentado previo la obtención del título de Ingeniero en Electrónica y Comunicaciones.

SUBLINEA DE INVESTIGACIÓN: Comunicaciones Inalámbricas
AUTOR: Michael Enrique Intriago Cobeña
TUTOR: Ing. Santiago Manzano, Mg.

**Ambato – Ecuador
Enero 2018**

APROBACIÓN DEL TUTOR

En mi calidad de tutor del trabajo de investigación con el tema: "ANÁLISIS DEL COMPORTAMIENTO DE LA COMUNICACIÓN INALÁMBRICA EN EL SISTEMA INTEGRADO AMBULATORIO DE PRESIÓN ARTERIAL", del Sr. Michael Enrique Intriago Cobeña, estudiante de la Carrera de Ingeniería en Electrónica y Comunicaciones, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial de la Universidad Técnica de Ambato, considero que el informe final reúne los requisitos suficientes para que continúe con los trámites y consiguiente aprobación de conformidad con el numeral 7.2 de los Lineamientos Generales para la aplicación de Instructivos de las Modalidades de Titulación de las Facultades de la Universidad Técnica de Ambato.

Ambato, 22 de enero de 2018

EL TUTOR



Ing. Victor Santiago Manzano Villafuerte, Mg.

AUTORÍA DEL TRABAJO

El presente Proyecto de Investigación titulado: "ANÁLISIS DEL COMPORTAMIENTO DE LA COMUNICACIÓN INALÁMBRICA EN EL SISTEMA INTEGRADO AMBULATORIO DE PRESIÓN ARTERIAL", es absolutamente original, auténtico y personal, en tal virtud, el contenido, efectos legales y académicos que se desprenden del mismo son de exclusiva responsabilidad del autor.

Ambato, 22 de enero de 2018



Michael Enrique Intriago Cobeña

C.C.: 1804219796

DERECHOS DE AUTOR

Autorizo a la Universidad Técnica de Ambato, para que haga uso de este Trabajo de Titulación como un documento disponible para la lectura, consulta y procesos de investigación.

Cedo los derechos de mi Trabajo de Titulación, con fines de difusión pública, además autorizo su reproducción dentro de las regulaciones de la Universidad.

Ambato, 22 de enero de 2018



Michael Enrique Intriago Cobeña

C.C.: 1804219796

APROBACIÓN DE LA COMISIÓN CALIFICADORA

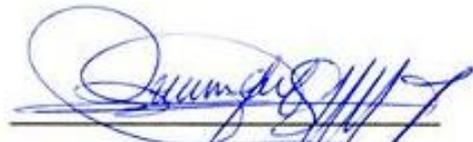
La Comisión Calificadora del presente trabajo conformada por los señores docentes Ing. Juan Pablo Pallo Noroña, Mg. e Ing. Marco Antonio Jurado Lozada, Mg., revisó y aprobó el Informe Final del Proyecto de Investigación titulado “ANÁLISIS DEL COMPORTAMIENTO DE LA COMUNICACIÓN INALÁMBRICA EN EL SISTEMA INTEGRADO AMBULATORIO DE PRESIÓN ARTERIAL”, presentado por el señor Michael Enrique Intriago Cobeña de acuerdo al numeral 9.1 de los Lineamientos Generales para la aplicación de Instructivos de las Modalidades de Titulación de las Facultades de la Universidad Técnica de Ambato.



Ing. Elsa Pilar Urrutia, Mg.
PRESIDENTA DEL TRIBUNAL



Ing. Marco Jurado, Mg.
DOCENTE CALIFICADOR



Ing. Juan Pablo Pallo, Mg.
DOCENTE CALIFICADOR

DEDICATORIA

A mi madre, por ser mi motivación desde el día en que nací, por su lucha incansable y su entrega infinita para poder brindar un futuro mejor para sus hijos. Por apoyarme y darme siempre ese empujón para lograr mis metas.

A mi familia por su apoyo incondicional en cada aspecto de mi vida diaria, por guiarme por el camino correcto y estar presentes en los momentos más difíciles.

Michael Intriago

AGRADECIMIENTO

A Dios por su amor infinito, por darme la fortaleza y la sabiduría necesaria en mi vida y jamás dejarme solo.

Gracias infinitas a mi madre Adaleinda y a mi segunda madre Rocío por su dedicación, su paciencia y su amor, que han forjado en mí un ser humano de bien.

A Mateo y José por ser mis pequeños hermanos y mis amigos.

Gracias a mi tutor Ing. Santiago Manzano por compartir su conocimiento y don de persona durante mi vida académica en la universidad y especialmente en este trabajo.

A mis amigos incondicionales, con quienes compartimos las mejores anécdotas y buenos momentos.

Michael Intriago

ÍNDICE

| | |
|--|-------|
| UNIVERSIDAD TÉCNICA DE AMBATO | I |
| APROBACION DEL AUTOR | II |
| AUTORÍA DEL TRABAJO | III |
| DERECHOS DE AUTOR | IV |
| APROBACIÓN DE LA COMISIÓN CALIFICADORA | V |
| DEDICATORIA | VI |
| AGRADECIMIENTO | VII |
| ÍNDICE | VIII |
| ÍNDICE DE TABLAS | XI |
| ÍNDICE DE FIGURAS..... | XIII |
| RESUMEN EJECUTIVO | XVII |
| ABSTRACT..... | XVIII |
| INTRODUCCIÓN | XIX |
| CAPÍTULO I..... | 1 |
| 1.1 Tema..... | 1 |
| 1.2 Planteamiento del problema | 1 |
| 1.3 Delimitación | 2 |
| 1.4 Justificación..... | 3 |
| 1.5 Objetivos | 4 |
| 1.5.1 Objetivo General: | 4 |
| 1.5.2 Objetivos Específicos:..... | 4 |
| CAPÍTULO II | 5 |

| | | |
|--------------------|--|----|
| 2.1 | Antecedentes Investigativos | 5 |
| 2.2 | Fundamentación teórica | 7 |
| | ■ | 7 |
| | ■ | 7 |
| 2.2.1 | Redes inalámbricas..... | 7 |
| 2.2.2 | Tipos de redes inalámbricas | 7 |
| 2.2.3 | Generalidades del estándar IEEE 802.15 | 8 |
| 2.2.4 | Generalidades del estándar IEEE 802.15.1 | 11 |
| 2.2.5 | Bluetooth 4.0 | 12 |
| 2.2.6 | Generalidades del estándar IEEE 802.11 | 21 |
| 2.2.7 | Generalidades de redes LTE | 27 |
| 2.2.8 | Parámetros para el análisis de comunicaciones inalámbricas | 41 |
| 2.2.9 | Aplicaciones médicas en redes WBAN y WPAN..... | 50 |
| 2.2.10 | Telemedicina | 51 |
| 2.2.11 | Monitor Médico | 51 |
| 2.2.12 | Presión Arterial | 52 |
| 2.2.13 | Frecuencia cardiaca..... | 53 |
| 2.2.14 | Simulación de redes | 54 |
| 2.2.15 | OMNeT++..... | 55 |
| 2.2.16 | INET Framework | 57 |
| 2.2.17 | SIMUlte INET | 59 |
| 2.3 | Propuesta de solución | 60 |
| CAPÍTULO III | | 61 |
| 3.1 | Modalidad de la investigación..... | 61 |
| 3.2 | Recolección de información | 62 |
| 3.3 | Procesamiento y análisis de datos | 62 |
| 3.4 | Desarrollo del proyecto | 62 |

| | |
|---|-----|
| CAPÍTULO IV | 64 |
| 4.1 Introducción..... | 64 |
| 4.2 Análisis del funcionamiento del prototipo del Sistema Integrado Ambulatorio de Presión Arterial..... | 65 |
| 4.3 Tecnologías de comunicación del Sistema Integrado Ambulatorio de Presión Arterial | 72 |
| 4.4 Pruebas en pacientes y resultados | 75 |
| 4.5 Análisis de plataformas de simulación | 78 |
| 4.6 Configuraciones de OMNeT++ y del Framework INET | 80 |
| 4.7 Simulación del Sistema Integrado Ambulatorio de Presión Arterial en OMNeT++..... | 84 |
| 4.7.1 Escenario 1: BLE y Wi-Fi (Simulación A)..... | 90 |
| 4.7.2 Escenario 1: BLE y Wi-Fi (Simulación B) | 104 |
| 4.7.3 Escenario 2: BLE y LTE (Simulación C) | 109 |
| 4.8 Análisis de resultados | 125 |
| 4.8.1 Análisis Escenario 1: BLE y Wi-Fi (Simulación A)..... | 130 |
| 4.8.2 Análisis Escenario 1: BLE y Wi-Fi (Simulación B)..... | 136 |
| 4.8.3 Análisis Escenario 2: BLE y LTE (Simulación C) | 141 |
| CAPÍTULO V | 149 |
| 5.1 Conclusiones | 149 |
| 5.2 Recomendaciones | 150 |
| BIBLIOGRAFÍA | 152 |
| ANEXOS | 161 |

ÍNDICE DE TABLAS

| | |
|--|-----|
| Tabla 1. Comparación de tecnologías del grupo de trabajo IEEE 802.15 [8, 9, 10].... | 9 |
| Tabla 2. Modelo ISO-OSI vs. Modelo IEEE 802.15 [11]..... | 10 |
| Tabla 3. Enmiendas del estándar 802.11 [26]..... | 21 |
| Tabla 4. Estándares en comunicaciones móviles [32]. | 27 |
| Tabla 5. Asignación de bandas LTE en Ecuador [35]. | 31 |
| Tabla 6. Bloques de recursos, subportadoras y ancho de banda en LTE [33]. | 35 |
| Tabla 7. Procedimientos de Handover en LTE [40]. | 36 |
| Tabla 8. Esquemas de modulación y codificación MCS [43]..... | 39 |
| Tabla 9. Principales características de SIMULte [65]..... | 60 |
| Tabla 10. Descripción de los componentes utilizados en el dispositivo electrónico. | 65 |
| Tabla 11. Formato del dato a enviar..... | 70 |
| Tabla 12. Detalles de parámetros de consumo energético de BLE [50, 68]..... | 71 |
| Tabla 13. Detalles de parámetros de consumo energético del DPA [69]..... | 72 |
| Tabla 14. Tecnologías utilizadas en la simulación [8, 9, 10, 70, 71, 72, 73]..... | 74 |
| Tabla 15. Resultados de las pruebas en pacientes..... | 76 |
| Tabla 16. Mediciones de presión arterial y frecuencia cardiaca el voltios. | 77 |
| Tabla 17. Comparación de resultados entre dispositivo validado y dispositivo en desarrollo..... | 77 |
| Tabla 18. Mediciones de retardos de extremo a extremo enlace BLE..... | 78 |
| Tabla 19. Comparación de plataformas de simulación de redes [62]. | 79 |
| Tabla 20. Requerimientos de la simulación vs Prestaciones de OMNeT++..... | 79 |
| Tabla 21. Resultados de retos de conexión a internet. | 96 |
| Tabla 22. Parámetros por defecto Wi-Fi en INET. | 103 |
| Tabla 23. Resultados de retos de conexión a internet con red de datos móviles.. | 117 |
| Tabla 24. Parámetros del canal inalámbrico de la Simulación C archivo config_channel.xml..... | 121 |
| Tabla 25. Parámetros del canal inalámbrico de retorno de la Simulación C archivo config_channel.xml..... | 123 |
| Tabla 26. Pérdida de paquetes Escenario 1 - Simulación A. | 133 |

| | |
|--|-----|
| Tabla 27. Pérdida de paquetes Escenario 1 - Simulación B..... | 137 |
| Tabla 28. Relación de entrega Escenario 2 - Simulación B..... | 137 |
| Tabla 29. Comparación de Casos de envío de datos Escenario 1 - Simulación B... | 138 |
| Tabla 30. Detalle de pérdida de paquetes en el Escenario 2 - Simulación C..... | 142 |
| Tabla 31. Pérdida de paquetes Escenario 2 - Simulación C..... | 143 |
| Tabla 32. Relación de entrega Escenario 2 - Simulación C..... | 143 |

ÍNDICE DE FIGURAS

| | |
|---|----|
| Fig. 1. Tipos de redes inalámbricas [7]. | 8 |
| Fig. 2. Grupo de trabajo IEEE 802.15 WPAN [3]. | 9 |
| Fig. 3. Stack o Pila de protocolos Bluetooth [11]. | 11 |
| Fig. 4. Pila de Bluetooth 4.0 [13]. | 12 |
| Fig. 5. Arquitectura Piconet y Scatternet de Bluetooth [14]. | 12 |
| Fig. 6. Canales utilizados en BLE [13]. | 13 |
| Fig. 7. Esquema de canal de un solo canal BLE de 2 MHz de ancho [15]. | 13 |
| Fig. 8. Desviación de frecuencia en GFSK para 1 y 0 [17]. | 14 |
| Fig. 9. Pulsos digitales suavizados en GFSK [17]. | 15 |
| Fig. 10. Acceso Múltiple por División de Tiempo, TDMA [19]. | 16 |
| Fig. 11. Formato general de paquete básico BLE [21]. | 17 |
| Fig. 12. Formato general de paquete mejorado en BLE [21]. | 18 |
| Fig. 13. (a) Formato de paquete BLE. (b) Ejemplo de paquete BLE [15]. | 18 |
| Fig. 14. Formato del paquete L2CAP BLE [21]. | 19 |
| Fig. 15. Característica de una conexión BLE en modo de publicación [24]. | 21 |
| Fig. 16. Característica de una conexión BLE en Tx y Rx [25]. | 21 |
| Fig. 17 Modo infraestructura de redes 802.11 [27]. | 22 |
| Fig. 18 Modo Ad-hoc de redes 802.11 [27]. | 22 |
| Fig. 19. Diagrama de modulación QPSK [29]. | 23 |
| Fig. 20. Estructura de cabecera UDP [31]. | 26 |
| Fig. 21. Estructura de cabecera TCP [31]. | 27 |
| Fig. 22. Arquitectura de red LTE [33]. | 28 |
| Fig. 23. Diagrama Lógico de EPC [34]. | 30 |
| Fig. 24. Anchos de banda para portadoras en LTE [34]. | 30 |
| Fig. 25. Mapa de cobertura de señal 4G LTE de la ciudad de Ambato. | 31 |
| Fig. 26. Señal OFDM representada en tiempo y frecuencia [36]. | 32 |
| Fig. 27. Comparación entre OFDMA y SC-FDMA [36]. | 33 |
| Fig. 28. Modelo de generador y receptor de señal OFDMA y SC-FDMA [36]. | 33 |
| Fig. 29. Estructura Slot, elementos de Recursos y Bloques de Recursos [33]. | 34 |
| Fig. 30. Programación de usuarios o User schedulling [33]. | 34 |

| | |
|---|----|
| Fig. 31. Niveles de RSRP, RSRQ y SINR necesarios en LTE [41]. | 38 |
| Fig. 32. Ilustración del CQI en sub bandas y la mejor banda M [33]. | 38 |
| Fig. 33. Símbolos y representación de forma de onda [51]. | 45 |
| Fig. 34. Efecto Delay Spread [53]. | 46 |
| Fig. 35. Efecto Multitrayectoria [53]. | 46 |
| Fig. 36. Señal afectada por Rayleigh Fading [54]. | 47 |
| Fig. 37. Ilustración de la presión arterial sistólica y diastólica [60]. | 52 |
| Fig. 38. Gráfica de frecuencia cardiaca [61]. | 54 |
| Fig. 39. Módulos simples y compuestos [64]. | 56 |
| Fig. 40. Módulos principales del framework INET. | 57 |
| Fig. 41. Módulos secundarios del framework INET. | 58 |
| Fig. 42. Dispositivo electrónico para adquisición de presión arterial y ECG. | 65 |
| Fig. 43. Brazaletes Welch Allyn para adultos [66]. | 67 |
| Fig. 44. Conexiones exteriores del prototipo hacia el brazalete. | 67 |
| Fig. 45. Sistema de adquisición de señales de electrocardiograma. | 68 |
| Fig. 46. Diagrama esquemático del prototipo. | 68 |
| Fig. 47. Diagrama de flujo del funcionamiento del prototipo. | 69 |
| Fig. 48. Batería Ultrafire 18650 2000 mAh 3,7 V [67]. | 70 |
| Fig. 49. Módulo BLE HM-10 [68]. | 71 |
| Fig. 50. Medición de consumo energético con Charger Doctor. | 72 |
| Fig. 51. Escenario 1: BLE y Wi-Fi. | 73 |
| Fig. 52. Escenario 2: BLE y red celular LTE 4G. | 73 |
| Fig. 53. Pruebas de dispositivo electrónico de adquisición de datos. | 75 |
| Fig. 54. Importación de proyecto en OMNeT++. | 81 |
| Fig. 55. Importación de INET framework en OMNeT++. | 82 |
| Fig. 56. Construcción del protocolo 802.15.1 para BLE en el framework INET. | 82 |
| Fig. 57. Inclusión de LTE en el código fuente del framework INET. | 83 |
| Fig. 58. Descripción de simulaciones realizadas. | 84 |
| Fig. 59. Menú desplegable para creación de archivos. “.ned”, “.ini”, “.xml”, etc. | 85 |
| Fig. 60. Organización de los archivos de la simulación. | 85 |
| Fig. 61. Generación de archivo “.ned”. | 88 |
| Fig. 62. Pestaña Design del archivo “.ned”. | 88 |
| Fig. 63. Pestaña Source del archivo “.ned”. | 89 |

| | |
|--|-----|
| Fig. 64. Creación de nueva red en el archivo “.ned”..... | 89 |
| Fig. 65. Descripción de red de la Simulación A del escenario 1. | 90 |
| Fig. 66. Prueba 1 de latencia de conexión a internet [77]...... | 95 |
| Fig. 67. Prueba 2 de latencia de conexión a internet [78]...... | 95 |
| Fig. 68. Prueba 3 de latencia de conexión a internet..... | 96 |
| Fig. 69. Detalles de parámetros de canales de fibra óptica y su conexión a internet. | 96 |
| Fig. 70. Detalles de parámetros para las tarjetas inalámbricas. | 97 |
| Fig. 71. Trayectoria del nodo según TractorMobility. | 100 |
| Fig. 72. Inclusión de objetos físicos en la simulación..... | 101 |
| Fig. 73. Descripción de red de la Simulación B del escenario 1..... | 105 |
| Fig. 74. Ingreso del número de clientes para la Simulación B del Escenario 1. | 106 |
| Fig. 75. Ambiente de la Simulación B al ingresar la cantidad de clientes. | 107 |
| Fig. 76. Descripción de red de la Simulación C del escenario 2..... | 110 |
| Fig. 77. Prueba 1 de latencia de conexión a internet..... | 116 |
| Fig. 78. Prueba 1 de latencia de conexión a internet..... | 116 |
| Fig. 79. Estructura de la NIC de LTE en OMNeT++ [81]...... | 117 |
| Fig. 80. Módulo compuesto LteNicBase en OMNeT++..... | 118 |
| Fig. 81. Ejemplo de Trayectoria del nodo según GaussMarkovMobility. | 119 |
| Fig. 82. Ejecución de las simulación desde archivo omnetpp.ini. | 125 |
| Fig. 83. Ventana de configuración del ambiente de simulación. | 126 |
| Fig. 84. Selección de configuración desde ambiente de simulación..... | 126 |
| Fig. 85. Ambiente de ejecución de simulaciones. | 127 |
| Fig. 86. Contenido del archivo “.anf” pestaña Inputs. | 128 |
| Fig. 87. Selección de resultados en pestaña Browse Data del archivo “.anf”..... | 129 |
| Fig. 88. Ejemplo de gráfica generada desde el archivo “.anf”..... | 129 |
| Fig. 89. Ambiente de simulación Escenario 1 – Simulación A. | 130 |
| Fig. 90. Consumo energético en modo SLEEP..... | 131 |
| Fig. 91. Consumo energético del dispositivo al transmitir..... | 131 |
| Fig. 92. Pérdida de paquetes de información Escenario 1 - Simulación A..... | 134 |
| Fig. 93. Retardo extremo a extremo Escenario 1 - Simulación A. | 135 |
| Fig. 94. Resultados de retardo extremo a extremo Escenario 1 - Simulación A..... | 135 |
| Fig. 95. Ambiente de simulación Escenario 1 – Simulación B..... | 136 |
| Fig. 96. Porcentaje de pérdidas de información Escenario 1 - Simulación B..... | 138 |

| | |
|--|-----|
| Fig. 97. Retardo extremo a extremo 10 usuarios Escenario 1 - Simulación B. | 139 |
| Fig. 98. Resultados de retardo extremo a extremo Escenario 1 - Simulación B..... | 139 |
| Fig. 99. Throughput de red Escenario 1 – Simulación B. | 140 |
| Fig. 100. Ambiente de simulación Escenario 2 – Simulación C..... | 141 |
| Fig. 101. Porcentaje de pérdidas de información Escenario 1 - Simulación C..... | 144 |
| Fig. 102. Retardo extremo a extremo 150 usuarios Escenario 2 - Simulación C. ... | 144 |
| Fig. 103. Resultados retardo extremo a extremo Escenario 2 - Simulación C..... | 145 |
| Fig. 104. Throughput de red Escenario 2 – Simulación C. | 145 |
| Fig. 105. CQI enlace de bajada de 150 usuarios Escenario 2 – Simulación C. | 146 |
| Fig. 106. Asignación de técnica de modulación según nivel de señal en LTE [82]. | 147 |
| Fig. 107. CQI enlace de bajada de 150 usuarios Escenario 2 – Simulación C. | 147 |

RESUMEN EJECUTIVO

El presente proyecto de investigación analiza la eficiencia del Sistema Integrado de Monitoreo Ambulatorio de Presión Arterial desarrollado por los investigadores de la Universidad Técnica de Ambato como herramienta para el diagnóstico de hipertensión. Determinando la estructura, arquitectura y tecnologías de comunicación entre pacientes y el servidor médico, evaluando dos posibles escenarios en OMNeT++ 5.0: comunicación en redes de área local inalámbrica y las redes de datos móviles. Los resultados obtenidos del estudio en el primer escenario, determinan una confiabilidad del 96,63% para comunicación y entrega de información entre los usuarios y el servidor médico, en el segundo escenario trabajando en la red de datos móviles LTE se revela un porcentaje mínimo de entrega de información del 94,71% al día, en un caso de estudio de 150 usuarios.

ABSTRACT

The present investigation project analyses the efficiency of the Integrated System of Ambulatory Blood Pressure Monitoring developed by the researchers of the Technical University of Ambato as a tool for the diagnosis of hypertension. Determining the structure, architecture and communication technologies between patients and the medical server, evaluating two possible scenarios in OMNeT++ 5.0: communication in wireless local area networks and mobile data networks. The results obtained from the study in the first scenario, determine a reliability of 96,63% for communication and delivery of information between users and the medical server, in the second scenario working in the LTE mobile data network reveals a minimum percentage of information delivery of 94,71% per day, in a case study of 150 users.

INTRODUCCIÓN

Para implementar sistemas electrónicos y de comunicaciones es necesario realizar un exhaustivo estudio de su comportamiento por medio de simulaciones, pruebas y análisis que determinen la mejor alternativa de métodos de acceso, protocolos y tecnologías de comunicación para que el sistema cumpla con las normas y necesidades deseadas por el usuario y el desarrollador.

Los sistemas electrónicos en su mayoría diseñados e implementados por estudiantes e investigadores, incluyen prototipos relacionados a mantener un control y monitoreo de la salud de pacientes que remotamente y con la ayuda de métodos de comunicación permitan conocer las variables físicas y estado del paciente en tiempo real, así se pueden ejecutar acciones correctivas o preventivas en busca de evitar complicaciones médicas o muertes prematuras como es el caso de pacientes con problemas de presión arterial.

El Capítulo I del presente proyecto describe las problemáticas que se presentan al momento de realizar un estudio de factibilidad para la producción en masa de un prototipo o sistema electrónico, enfocándose en el desarrollo del Sistema Integrado Ambulatorio de Presión Arterial que en la actualidad está siendo desarrollado por los investigadores de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial.

En el Capítulo II se hace mención a trabajos investigativos relacionados a estudios aplicados a sistemas de control de la presión arterial y otras variables físicas relacionadas con problemas de salud. Este capítulo incluye la fundamentación teórica del proyecto de investigación y concluye con la propuesta de solución al problema detallado en el Capítulo I.

A continuación del marco teórico el Capítulo III detalla la metodología utilizada en el desarrollo de este proyecto de investigación.

En el Capítulo IV se describe la realización del análisis de la comunicación inalámbrica del Sistema Integrado Ambulatorio de Presión Arterial y cada uno de los pasos que llevaron al cumplimiento de los objetivos planteados.

Finalmente en el Capítulo V se redactan las conclusiones y recomendaciones obtenidas del presente proyecto de investigación.

CAPÍTULO I

El problema

1.1 Tema

“ANÁLISIS DEL COMPORTAMIENTO DE LA COMUNICACIÓN INALÁMBRICA EN EL SISTEMA INTEGRADO AMBULATORIO DE PRESIÓN ARTERIAL”

1.2 Planteamiento del problema

Según la información publicada por el INEC en el año 2013, la hipertensión es definida en Ecuador como la segunda causa de muerte en el sexo femenino y la tercera en el sexo masculino, mientras que a nivel mundial las complicaciones derivadas de esta patología crónica son la causa de 9,4 millones de defunciones cada año [1]. Con este antecedente, investigadores desarrollan prototipos con el fin de mantener un control instantáneo del paciente por parte del personal médico de manera remota y con tecnología de punta, haciendo necesario que la información de las variables físicas del paciente como frecuencia cardiaca y presión arterial sean manejadas de manera correcta, instantánea y confiable. Prototipos de esta índole utilizan tecnologías con servidores virtuales médicos (SVM) implantados sobre redes WBAN y WPAN de gran auge en el área médica gracias a su portabilidad.

Diseñar e implantar estos sistemas en todos los pacientes que son parte de la demanda, genera altos costos de producción con resultados finales desconocidos, si se los pone en funcionamiento sin un análisis riguroso previo; por lo tanto, se requiere simular por software una gran cantidad de prototipos, paso primordial para saber su comportamiento y justificar su fabricación a gran escala. Con las simulaciones se requiere analizar y procesar información para evaluar parámetros como latencia, retardo, pérdida de paquetes y consumo de energía. Además, las problemáticas de capa física, capa de red, canal inalámbrico, empaquetado de información, procesamiento de señales e interferencias necesitan ser simuladas para saber cuáles son las características que deben tener estos dispositivos para funcionar correctamente. Así se pueden evitar datos erróneos o tardíos, pérdidas de información o mal funcionamiento del sistema que ocasionarían emergencias médicas fatales cuando los dispositivos estén funcionando en la vida real [2].

Para realizar simulaciones de redes WPAN con acceso remoto, la falta de información es evidente y su desarrollo en plataformas de software libre es escasa, incluso, para realizar este tipo de estudios existen varias empresas que prestan sus servicios cobrando altos costos ya que manejan hardware y software de licencia. Esta realidad lleva a los desarrolladores e investigadores a implementar prototipos que no cumplen el objetivo real para el que fueron diseñados, el cual es brindar un servicio de alta calidad a un gran número de personas en el área médica.

1.3 Delimitación

De contenidos:

Área académica: Comunicaciones.

Línea de investigación: Tecnologías de la Información y la Comunicación.

Delimitación espacial: El presente proyecto de investigación se desarrolló en la Universidad Técnica de Ambato, Facultad de Ingeniería en Sistemas, Electrónica e Industrial.

Delimitación temporal: La presente investigación se ha desarrollado en el período académico Marzo 2017 – Septiembre 2017 de acuerdo a lo establecido en el Reglamento de Graduación para obtener el Título Terminal de Tercer Nivel de la Universidad Técnica de Ambato.

1.4 Justificación

El desarrollo del proyecto se enfoca en el análisis del comportamiento de la comunicación inalámbrica en el Sistema integrado ambulatorio de presión arterial, que está siendo desarrollado en la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, por medio de simulaciones con el fin de sustentar la implementación de este sistema de manera masiva, y su uso en cientos de personas del cantón Ambato de la provincia de Tungurahua.

Al sustentar la implementación de este dispositivo y su futura implantación en los pacientes con problemas de salud vinculados con la presión arterial, el paciente contará con la facilidad de realizar el control de su salud cuando se encuentre realizando cualquier actividad o descansando; además no tendrá la necesidad de realizarse un chequeo médico rutinario o de emergencia en una clínica u hospital. El doctor o el personal médico a cargo del paciente pueden aprovechar la tecnología que está al alcance de sus manos para gestionar de manera eficaz y rápida el diagnóstico del paciente, tomando decisiones inmediatas en caso de presentarse alguna emergencia.

Las simulaciones manejan datos de presión arterial y electrocardiograma en tiempo real, siendo enviados vía inalámbrica de manera continua, utilizando varias tecnologías de comunicación como Bluetooth, Wi-Fi o Red Móvil Celular, hasta llegar al dispositivo móvil o de escritorio del doctor de cabecera, las simulaciones permiten analizar y evaluar parámetros como latencia, retardo, pérdida de paquetes y consumo de energía, que en muchos casos generan colapsos de red, congestión de redes de área local, saturación de canales de comunicación inalámbrico, pérdida de información, datos tardíos y erróneos, etc. Con este análisis se eligieron las mejores alternativas de métodos de acceso, protocolos y tecnologías de comunicación para que el sistema funcione de manera fiable y así se garantiza que el dispositivo que se está

desarrollando cumpla con la normas y necesidades deseadas por el desarrollador, el doctor y principalmente el paciente.

El prototipo del Sistema Integrado Ambulatorio de Presión Arterial actualmente no está siendo manufacturado para ser comercializado, ya que no existe un fundamento el cual refleje que el dispositivo funcionará correctamente al estar conectado y enviando datos de cientos de personas en tiempo real. Con las herramientas de software libre y sin costo, principalmente con estas simulaciones, es posible simular ambientes de esta índole y determinar la factibilidad de la manufactura a gran escala del prototipo.

1.5 Objetivos

1.5.1 Objetivo General:

Simular el sistema de comunicación inalámbrico integrado ambulatorio de presión arterial para determinar la factibilidad de su producción masiva.

1.5.2 Objetivos Específicos:

- Analizar los estándares de comunicaciones inalámbricas y móviles para redes WPAN basados en la nube.
- Determinar la estructura, arquitectura, tecnología y los posibles métodos de comunicación entre el paciente y el servidor médico en base a los requerimientos del sistema.
- Analizar las plataformas de simulación que permiten desarrollar redes WPAN basados en la nube.
- Desarrollar la simulación del sistema de comunicación inalámbrico integrado ambulatorio de presión arterial en la plataforma seleccionada.

CAPÍTULO II

Marco teórico

2.1 Antecedentes Investigativos

Al desarrollar una investigación de tipo bibliográfica se hallaron los siguientes resultados:

Johanna Estefanía Averos Vargas en su proyecto de titulación denominado “ESTUDIO DEL ESTÁNDAR IEEE 802.15.6 Y SIMULACIÓN DE LOS PARÁMETROS DE TRANSMISIÓN EN UNA RED DE ÁREA CORPORAL EN LA BANDA DE FRECUENCIA DE 2.4 GHz”, analiza el estándar IEEE 802.15.6 en la banda de 2.4 GHz y simula los parámetros de transmisión para una red de área corporal para un máximo de seis nodos sobre el cuerpo humano, utilizando Castalia que es un framework de OMNeT++, con la simulación desarrollada se obtuvieron resultados como latencia, pérdida de paquetes y consumo de energía que se presentan al manejar este estándar [3].

En relación a artículos científicos publicados en revistas internacionales se hallaron los siguientes resultados:

Jamil Y. Khan, Mehmet R. Yuce y Farbood Karami, en su artículo científico “PERFORMANCE EVALUATION OF A WIRELESS BODY AREA SENSOR

NETWORK FOR REMOTE PATIENT MONITORING”, parte de las publicaciones del “30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society” desarrollado en Australia, analizan el rendimiento basado en el estándar IEEE802.15.4 / Zigbee MAC WBAN operando en diferentes entornos de monitorización de pacientes utilizando un modelo de simulación basado OPNET. Sus resultados muestran que es posible monitorear pacientes desde lugares remotos usando WBAN sin retardos considerables, utilizando paquetes cortos con la red corporal que no está directamente conectada a la red del hospital, sino utilizando un nodo de servicios con lo cual se reduce colapsos de red y retardos de transmisión [2].

Diana Patricia Tobón Vallejo y Natalia Gaviria Gómez, desarrollaron el “ANÁLISIS DE MÉTRICAS DE CALIDAD DE SERVICIO PARA LA CONFIGURACIÓN DEL PROTOCOLO CSMA/CA EN REDES DE SENSORES INALÁMBRICAS DE ÁREA CORPORAL”, publicado en la Revista Científica Ingeniería y Desarrollo, Vol 30, No 1 en el año 2012, en el cual se realiza un análisis de varianza ANOVA para encontrar una configuración adecuada del protocolo CSMA, utilizando el simulador OMNeT++ y luego en una plataforma real de red de sensores. De esta forma, se concluyó que las tres variables (tráfico, el macMinBE y el macSuperFrameOrder) elegidas influyen en el desempeño de la red, y se encontraron los valores óptimos para los parámetros seleccionados [4].

Deena M. Barakah y Muhammad Ammad-Uddin, en su artículo científico “A SURVEY OF CHALLENGES AND APPLICATIONS OF WIRELESS BODY AREA NETWORK (WBAN) AND ROLE OF A VIRTUAL DOCTOR SERVER IN EXISTING ARCHITECTURE”, publicado en: “Third International Conference on Intelligent Systems Modelling and Simulation” del 2012, proponen un estudio exhaustivo de la tecnología inalámbrica de red de área del cuerpo (WBAN), así como la introducción de un servidor médico virtual (VDS) en la arquitectura WBAN existente, con lo que se desplegó un análisis completo de los roles que cumplen este tipo de redes en la mejora de la calidad de vida de las personas, incluyendo aplicaciones diarias y retos actuales de ingeniería, que se deben tomarse en cuenta para

el desarrollo de sistemas inteligentes capaces de simular escenarios similares en el área médica [5].

En los registros de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial de la Universidad Técnica de Ambato no se encontró un trabajo de investigación relacionado con el tema desarrollado.

2.2 Fundamentación teórica

2.2.1 Redes inalámbricas

Red inalámbrica es un término que define la conexión de nodos por medios no guiados a través de ondas electromagnéticas, sin utilizar cables. La transmisión y recepción se realiza por medio de antenas. El uso de este tipo de redes es la única solución para zonas a las que no llega el cableado, como es el caso de zonas rurales diseminadas aunque presenta desventajas como la susceptibilidad a cambios atmosféricos y la falta de seguridad en el intercambio de información [6].

2.2.2 Tipos de redes inalámbricas

Los tipos de redes inalámbricas se distinguen según su alcance y tecnología [6]:

- **PAN o WPAN (*Wireless Personal Area Network*):** Red Inalámbrica de Área Personal: es una red creada para la conexión de dispositivos inalámbricos dentro de una zona de área personal limitada a pocos metros. Bluetooth es la tecnología más utilizada en la actualidad, pero existen otras tecnologías como los infrarrojos, RFID, TAG, UWB, ZigBee, etc., que cumplen el estándar IEEE 802.15. Pueden ser de bajo alcance (LR-WPAN), medio alcance (MR-WPAN) y alto alcance (HG-WPAN).
- **WLAN (*Wireless Local Area Network*):** Red Inalámbrica de Área Local: es el tipo de red inalámbrica más desplegada en redes domésticas y edificios que permite múltiples conexiones dependiendo de la versión del estándar 802.11

que se utilice (802.11a, 802.11b, 802.11g y la 802.11n). También es conocido como Wi-Fi.

- **WMAN (Wireless Metropolitan Area Network):** Red Inalámbrica de Área Metropolitana: conocidas como Inalámbricas de Banda Ancha, son redes implementadas en áreas geográficas medianas como ciudades o municipios pequeños, urbanizaciones, barrios, etc. Por lo general se basan en el protocolo IEEE 802.16.
- **WWAN (Wireless Wide Area Network):** Red Inalámbrica de Área Mundial: redes globales satelitales y móviles, tales como: vSAT, 2G, 3G y 4G.

Estos tipos de redes se diferencian por sus áreas de cobertura, las tecnologías y estándares que manejan, tal como se muestra en la Fig.1:

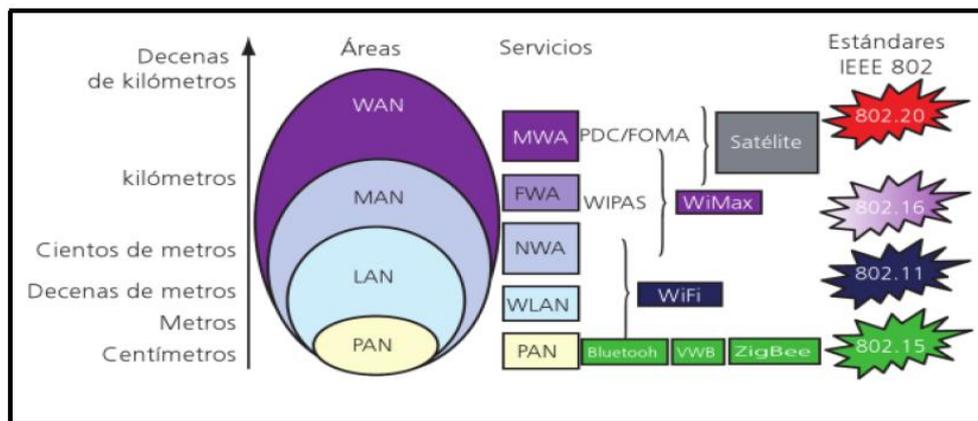


Fig. 1. Tipos de redes inalámbricas [7].

2.2.3 Generalidades del estándar IEEE 802.15

El estándar IEEE 802.15 es la base de las Redes de Área Corporal Inalámbrica (Wireless Personal Area Network), que son redes utilizadas para el compartimiento de información a cortas distancias de manera privada e íntima de un grupo de participante, permite soluciones a redes pequeñas, eficiencia energética y bajos costos [3]. En la Fig.2 se indican los estándares que se deducen del mismo.

Las tecnologías como bluetooth, infrarrojos, RFID, TAG, UWB, ZigBee, etc., han sido desarrolladas para cubrir múltiples aplicaciones en el campo industrial, investigativo,

médico y científico. Cada tecnología se diferencia por los métodos de acceso al medio, topologías, capacidades de transferencia de datos, características inalámbricas, etc.

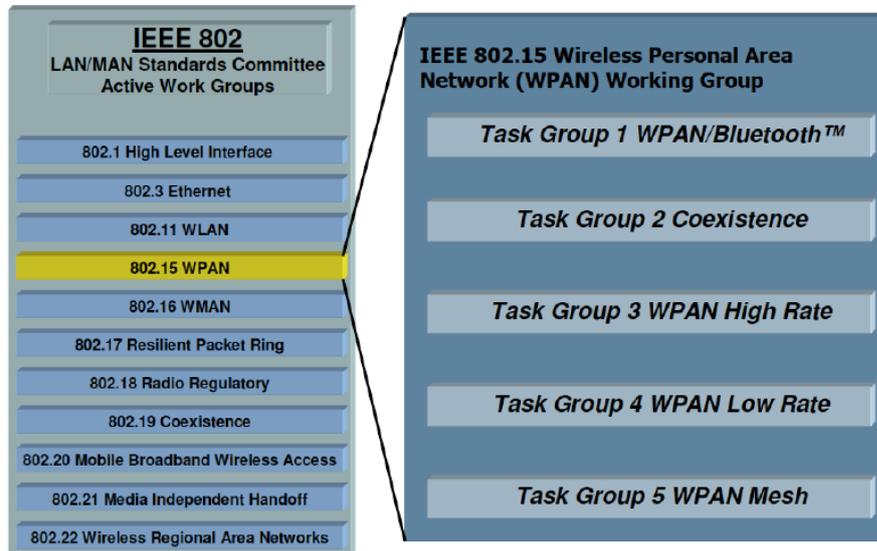


Fig. 2. Grupo de trabajo IEEE 802.15 WPAN [3].

En la Tabla 1 se detalla Bluetooth en su versión clásica, Bluetooth 4.0 y ZigBee, tecnologías pertenecientes al estándar IEEE 802.15 que tienen la mayor acogida en el mercado debido al desarrollo investigativo y tecnológico disponible.

Tabla 1. Comparación de tecnologías del grupo de trabajo IEEE 802.15 [8, 9, 10].

| Parámetro\Tecnología | Bluetooth Clásico | Bluetooth 4.0 | ZigBee |
|---------------------------------------|----------------------|----------------------|----------------------------|
| Estándar | 802.15.1 | 802.15.1 | 802.15.4 |
| Frecuencia | 2,4 GHz | 2,4 GHz | 2,4 y 5 GHz |
| Tasa Máxima de Bits | 1 - 3 Mbps | 721 kbps – 1 Mbps | 0,25 Mbps |
| Tipo de datos | Audio, imagen, datos | Audio, imagen, datos | Pequeños paquetes de datos |
| Rango Máximo Cobertura | 100 m | 50 m | 100 m |
| Vida de la Batería | Días | 1 semana | 100 días |
| Tamaño de red | 7 nodos | 8 nodos | 64.000+ nodos |
| Potencia típica de transmisión | 100 mW | 1 - 100 mW | 1 - 10 mW |
| Ancho de canal | 1 MHz | 2 MHz | 2 MHz |
| Modulación | GFSK | GFSK | BPSK |

| | | | |
|--|---|---|---------------------------------------|
| Sensibilidad del receptor | -90 dBm | -87 a -93 dBm | -85 dBm |
| Número de canales | 79 canales | 40 canales | 16 canales |
| Difusión del espectro | FHSS | FHSS | DSSS/CSS |
| Método de acceso al medio (MAC) | TDMA | TDMA | CSMA/CA, TDMA |
| Identificación MAC | 48 bits | 8 y 32 bits | 16 y 64 bits |
| Control de errores FEC/CRC | 8 bits con ACK opcional | CRC de 24 bits y ACK | 16 bits con ACK opcional |
| QoS | Si | Si | Si |
| Latencia | < 100 ms | < 3 ms | < 5 ms |
| Topología de red | Estrella | Estrella, Scatternet | Estrella, Cluster, Malla |
| Tiempo de unión a red | < 10 s | < 10 s | < 1 s |
| Consumo de energía | Medio | Bajo | Muy bajo |
| Complejidad del protocolo | Sencillo | Complejo | Sencillo |
| Seguridad | Autenticación, Encriptación | Autenticación, Encriptación | Autenticación, Encriptación |
| Aplicación | WPAN | MR-WPAN | LR-WPAN |
| Fortalezas | Bajo Costo y múltiples perfiles de aplicación | Bajo Costo y múltiples perfiles de aplicación | Fiabilidad, bajo consumo y bajo costo |

El modelo IEEE 802.15 está basado en el modelo ISO-OSI y existe una relación por capas o niveles de comunicación. El proyecto IEEE divide la capa de enlace de datos (DLL) en dos subcapas, la capa MAC de control de acceso al medio y la capa LLC de control lógico de enlace, esta última común en los estándares 802.3, 802.11 y en la familia del 802.15. La MAC varía dependiendo del hardware y su implementación física, en la Tabla 2 se muestra una comparativa de los modelos mencionados [11]:

Tabla 2. Modelo ISO-OSI vs. Modelo IEEE 802.15 [11].

| | Modelo ISO-OSI | Modelo IEEE 802.15 |
|----------|-------------------------------|----------------------------------|
| 7 | Capa de Aplicación | Capas Superiores |
| 6 | Capa de Presentación | |
| 5 | Capa de Sesión | |
| 4 | Capa de transporte | |
| 3 | Capa de red | |
| 2 | Capa de enlace de datos (DLL) | Control de enlace lógico (LLC) |
| | | Control de acceso a medios (MAC) |
| 1 | Capa física | Capa física (PHY) |

2.2.4 Generalidades del estándar IEEE 802.15.1

El estándar IEEE 802.15.1 ratificado en el 2005 por la IEEE, es una formalización de la tecnología inalámbrica Bluetooth que permite comunicaciones a cortas distancias entre dispositivos eliminando el uso de cableado y haciendo que los dispositivos sean portables en su totalidad. Una de las ventajas es la robustez, bajo consumo energético, bajo costo de operación, conexión más rápida, entre otras que dependen de la funcionalidad que se le desee dar a la tecnología.

El término sistema central dentro de bluetooth involucra la combinación del transceptor de radio frecuencia (RF), la banda base (BB), y la pila del protocolo como se muestra en la Fig. 3, este sistema permite que la conexión de dispositivos comparta datos de diferente clase [12].

En bluetooth se utilizan protocolos propios desarrollados para esta tecnología y se los combina con protocolos adoptados de otras especificaciones, lo que permite aprovechar un conjunto muy amplio de ventajas. La pila de protocolos divide en cuatro capas lógicas [11]:

- Núcleo de Bluetooth: Radio, Banda Base, LMP, L2CAP, SDP.
- Sustitución de cables: RFCOMM.
- Protocolos adoptados: PPP, UDP, TCP, IP, OBEX, WAP, IRMC, WAE.
- Control de telefonía: TCS-binary, AT-Commands

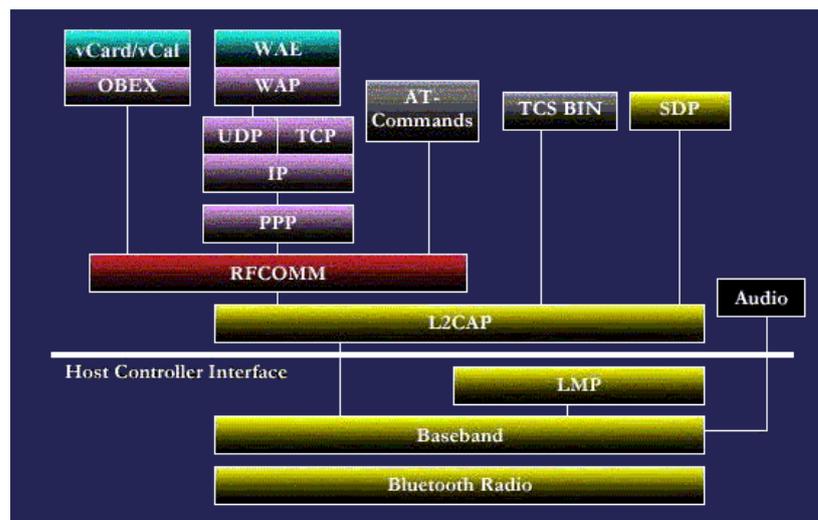


Fig. 3. Stack o Pila de protocolos Bluetooth [11].

2.2.5 Bluetooth 4.0

Bluetooth 4.0 también llamado Bluetooth Low Energy (BLE) o Smart Bluetooth basado en el protocolo IEEE 802.15.1 de Bluetooth, incluyendo una nueva pila y perfil de arquitectura diseñado para hacerlo barato y de fácil implementación. La pila de bluetooth se muestra en la Fig. 4:

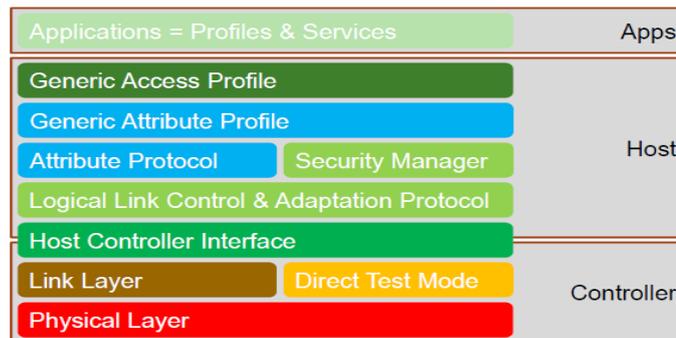


Fig. 4. Pila de Bluetooth 4.0 [13].

Arquitectura

La arquitectura de Bluetooth genera la formación de *piconets* y *scatternets*. En las *piconets* existe un dispositivo maestro y hasta 8 esclavos, donde el maestro determina el canal y la fase. En las *scatternets* un dispositivo de una *piconet* puede existir como maestro o esclavo en otra *piconet*, esta arquitectura permite a muchos dispositivos compartir la misma área, aprovechando el ancho de banda y haciéndolo más eficiente. La Fig. 5 presenta una conexión convencional entre dispositivos bluetooth incluyendo uno trabajando en los dos modos y en diferentes *piconets*. [14].

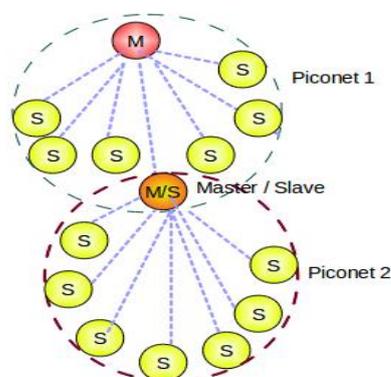


Fig. 5. Arquitectura Piconet y Scatternet de Bluetooth [14].

Rango de Frecuencias

El rango de frecuencias para BLE se ilustra en la Fig. 6, BLE trabaja en la banda de frecuencias ISM a diferencia del Bluetooth clásico de 79 canales, este trabaja con 40 canales espaciados por 2 MHz, 37 (0-36) canales para transmisión de datos y 3 (37,38 y 39) canales de publicación de dispositivos [13].

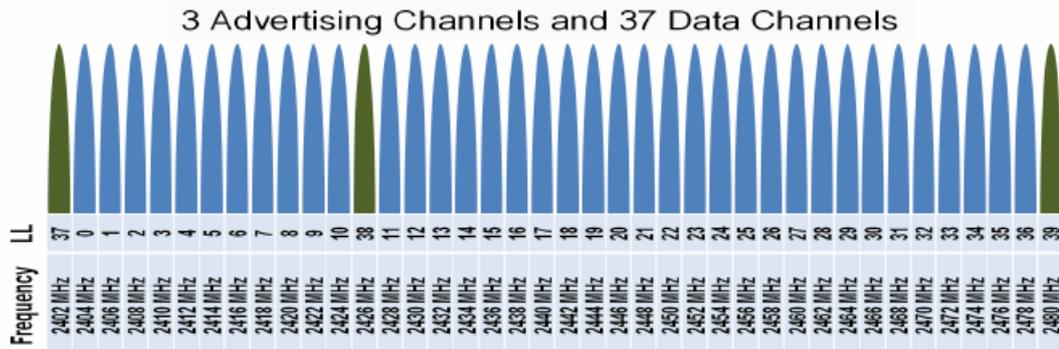


Fig. 6. Canales utilizados en BLE [13].

Cada canal tiene un número que corresponde a su frecuencia central F_c , mientras cada canal tiene una frecuencia de desviación mayor a 185 KHz sobre la F_c y corresponde a un bit “1” y de 185 kHz por debajo de la misma correspondiendo a un bit “0” como se indica en la Fig. 7. El espectro de las transmisiones BLE está compuesto por una máscara Gausiana para caber dentro del ancho de banda de 2 MHz [15].

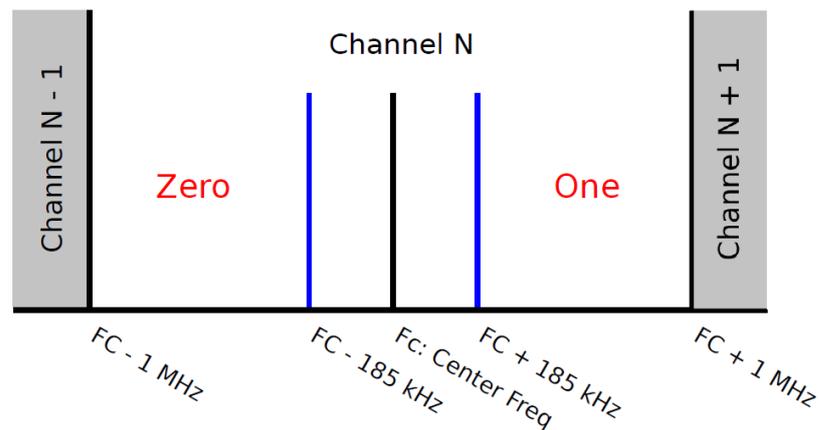


Fig. 7. Esquema de canal de un solo canal BLE de 2 MHz de ancho [15].

Un dispositivo BLE puede trabajar en dos modos dependiendo de su funcionalidad: publicación y escaneo. En modo de publicación, se lo denomina dispositivo publicador, se envían periódicamente paquete PDUs llamados ADV_IND por medio los tres canales de publicación (index = 37, 38, 39) para detectar los dispositivos cercanos para una posible conexión. Los retardos para el envío de estos paquetes son aleatorios y en caso de enviarse paquetes PDU al mismo tiempo por diferente canal existen colisiones.

Por otro lado, en modo de escaneo, llamado escáner, se escanea periódicamente los canales de publicación y se escucha la información de publicación de los demás dispositivos. Al recibir un paquete de publicación el escáner envía un paquete de regreso como respuesta [16].

Modulación GFSK

La modulación GFSK (*Gaussian frequency shift keying*) codifica datos como una serie de cambios en la frecuencia portadora de la señal a transmitir. La forma más básica de la implementación de GFSK es 2-GFSK; esta usa dos frecuencias dependiendo del dato a ser transmitido, sea 1 o 0. Como se ilustra en la Fig. 8, para transmitir un 1 la frecuencia portadora es aumentada a cierta desviación, mientras que los 0 ceros son codificados disminuyendo las frecuencia por la misma desviación, en la vida real esta desviación de frecuencia es mínima [17].

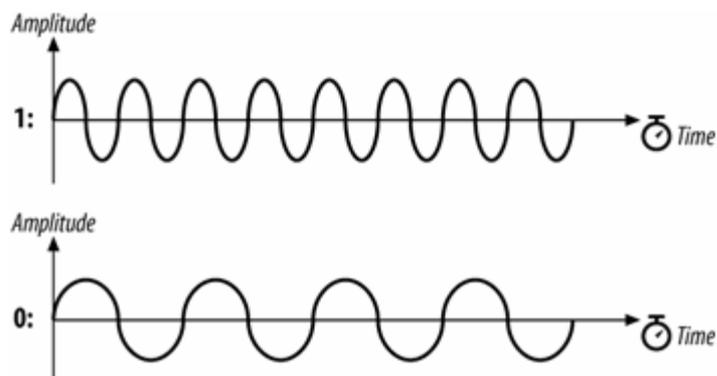


Fig. 8. Desviación de frecuencia en GFSK para 1 y 0 [17].

La Fig. 9 ilustra esta técnica de modulación, que realiza transiciones suaves entre cada símbolo digital, evitando los cambios abruptos de frecuencia que se producen en otras modulaciones como FSK que presenta transiciones de frecuencia de hasta 180°. GFSK utiliza un banco de filtros Gaussianos donde los pulsos digitales son suavizados antes de ingresar a un modulador, estos pulsos digitales son recortados para tener una forma similar a senoidal, pero sin perder su periodo y frecuencia. Ese filtro permite reducir la potencia de la banda lateral, la interferencia con canales adyacentes pero la desventaja es el aumento de interferencia intersimbólica [18].

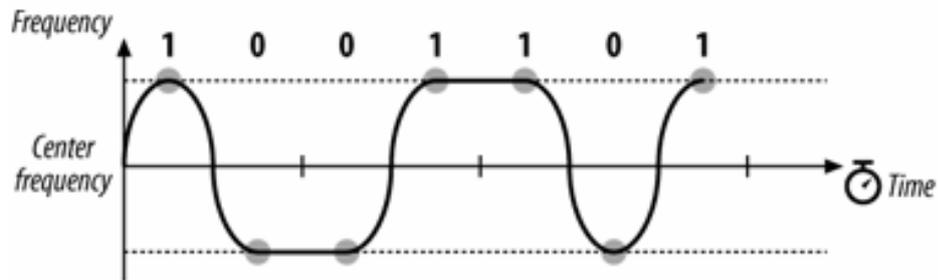


Fig. 9. Pulsos digitales suavizados en GFSK [17].

Las frecuencias en megahercios están definidas de acuerdo a la Ecuación 1 para cada canal:

$$f_c = 2400 + n \quad n = 0,1,2,3, \dots,39 \quad (1)$$

Por ejemplo, el primer canal usa una frecuencia portadora de 2402 MHz y el segundo canal una frecuencia portadora de 2403 MHz [19].

Método de acceso al medio TDMA

El acceso múltiple por división de tiempo TDMA permite que los nodos o estaciones compartan un ancho de banda de un canal en el tiempo. Cada estación está ubicada en una ranura de tiempo en la cual envía datos como se observa en la Fig. 10:

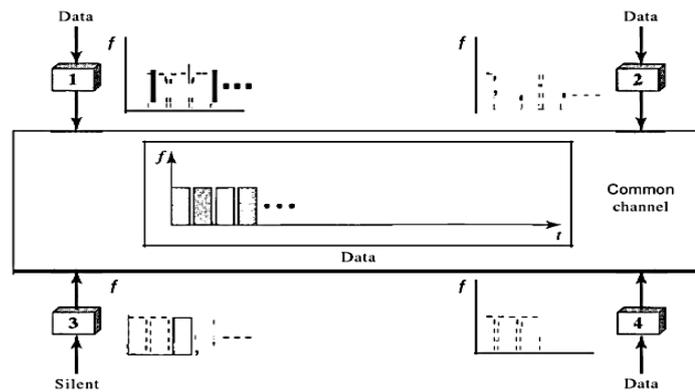


Fig. 10. Acceso Múltiple por División de Tiempo, TDMA [19].

Lograr una sincronización entre las estaciones es un problema en TDMA, ya que cada una necesita saber el inicio y la localización de su ranura o espacio; esto es difícil porque los retardos de propagación generados en cada estación están esparcidos en un área considerable. Para compensar este problema se utilizan tiempos de guarda y bits de sincronización al inicio de cada ranura.

TDMA es un método de acceso en la capa de enlace donde cada estación le comunica a su capa física usar cierto espacio de tiempo para comunicar. En Bluetooth se utiliza una forma de TDMA llamada TDD-TDMA que es una forma de comunicación half-duplex donde los implicados en una comunicación envían y reciben datos, pero no al mismo tiempo e inclusive utilizando diferentes frecuencias debido al método FHSS [19].

BMAC

Berkeley Media Access Control, es un protocolo desarrollado para comunicaciones con método de acceso al medio TDMA, utilizado en redes WPAN, WSN y WBAN para comunicaciones de baja y mediana tasa de transmisión de datos. Este protocolo MAC permite el bajo consumo energético ya que maneja los recursos de radio de forma que en caso de ser necesario permite que los nodos cambien de estado entre encendido y apagado repetidamente sin perder paquetes, lo cual se traduce en una reducción de consumo energético notable al no tener que estar encendido todo el tiempo. Cuando un nodo tiene un paquete que enviar, espera un determinado tiempo llamado “tiempo

de inactividad” antes de revisar el canal de transmisión. Si el canal está libre, el nodo transmite; en otro caso empieza un segundo “tiempo de inactividad” hasta que el canal esté libre [20].

FHSS

FHSS (*frequency-hopping spread spectrum*) es utilizado en bluetooth como un método de capa física que permite evitar interferencias de otras redes y dispositivos inalámbricos. En bluetooth el cambio de frecuencias se realiza 1600 veces por segundo cambiando la misma cantidad de veces la frecuencia de modulación. Dando como resultado que una frecuencia se utilice por un lapso de $625\mu\text{s}$ ($1/1600\text{ s}$) [19].

Enlaces físicos

Existen dos tipos de enlaces físicos básicos para BLE [19]:

- **SCO (*Synchronous Connection Oriented Link*):** El SCO es un enlace físico síncrono orientado a la conexión generado en intervalos regulares para reservar ranuras de tiempo específicas, por lo general son dos ranuras, una para cada dirección.
- **ACL (*Asynchronous Connectionless Link*):** EL ACL un enlace asíncrono sin conexión es usado cuando la integridad de los datos es mucho más importante que la latencia, permite que datos erróneos o dañados sean retransmitidos.

Estructura del paquete BLE

La estructura del paquete consiste en tres entidades, como se ilustra en la Fig. 11:

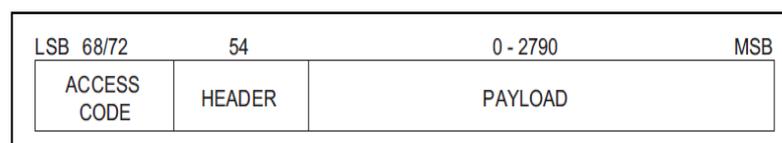


Fig. 11. Formato general de paquete básico BLE [21].

El código de acceso es de 72 o 68 bits y la cabecera de 54 bits. La carga útil entre 0 y 2790 bits y cada paquete puede estar construido por:

- Solamente el código de acceso
- Código de acceso y cabecera
- Código de acceso, cabecera y carga útil

Por otro lado existen mejoras en el formato del paquete con tiempos de guarda luego de la cabecera y además bits de sincronización como se muestra en la Fig.12 [21]:

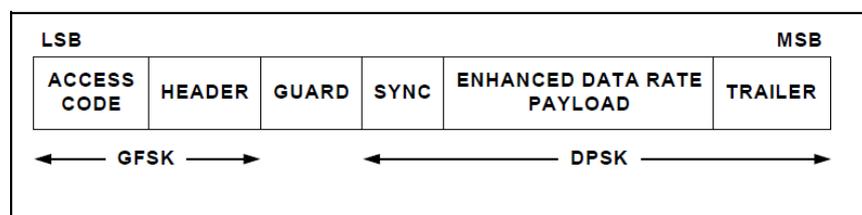


Fig. 12. Formato general de paquete mejorado en BLE [21].

Estructura del paquete de publicación

La trama del paquete de publicación de BLE consta de cuatro partes, la Fig. 13 indica su contenido: el preámbulo, dirección de acceso, PDU y el CRC. En el PDU se incluye información de nombre del dispositivo, en la dirección de acceso se especifica que un paquete de publicación. Al final de la trama se cuenta con CRC con un dato de entrada que es el PDU [15].

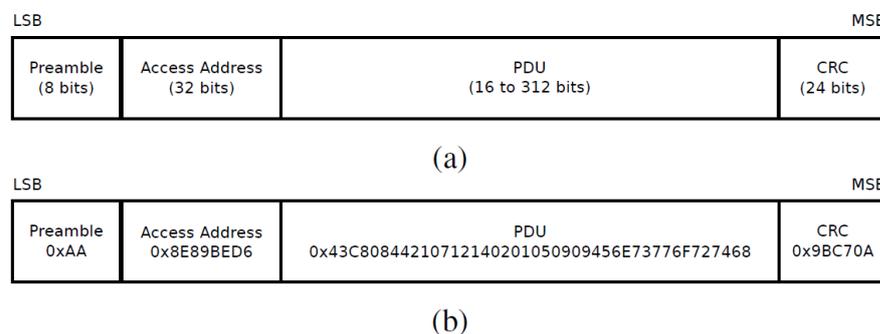


Fig. 13. (a) Formato de paquete BLE. (b) Ejemplo de paquete BLE [15].

L2CAP y LMP

L2CAP (*Logical Link Control and Adaptation Protocol*) es el equivalente a la subcapa LLC en redes LAN y es usado para el intercambio de datos en un enlace ACL. El formato del paquete está estructurado como expone la Fig. 14:

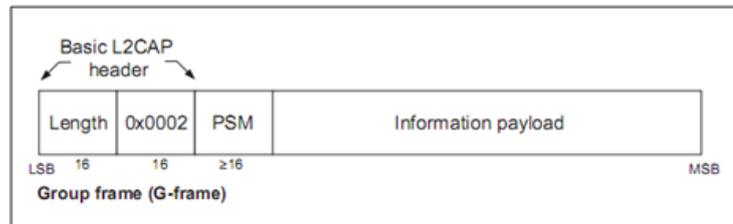


Fig. 14. Formato del paquete L2CAP BLE [21].

El campo de longitud de 16 bits define el tamaño del dato proveniente de capas superiores en bytes, este dato puede tener un tamaño superior a 65.535 bytes. Seguido a la longitud se especifica con 16 bits el canal virtual creado en este nivel.

En el transmisor esta capa toma los datos de capas superiores y los entrama para entregarlos a la capa de banda base. En el receptor se toma la trama de la banda base, se extrae el dato o carga útil y es entregado a la capa de protocolo apropiado.

El tamaño máximo de la carga útil en la banda base es de 2790 bits o 348 bytes, incluyendo bytes de definición de paquete y longitud. Sin embargo, una aplicación de capas superiores puede enviar datos de tamaño superior a 65.535 bytes. La capa L2CAP permite segmentar este dato en paquetes, añadiendo información extra como su ubicación en el paquete original, así la banda base transmitirá el dato n veces dependiendo del tamaño del mensaje original [19].

LMP (*Link Manager Protocol*) es un protocolo utilizado para establecer la conexión y control del enlace en Bluetooth y encargado de descubrir los dispositivos cercanos. Junto a L2CAP forman la MAC de bluetooth, utilizada además por el método de acceso TDMA [22].

Consumo energético

El consumo energético es un papel esencial del BLE ya que utilizar la mínima cantidad de energía es uno de los objetivos primordiales junto a la búsqueda de eficiencia de transmisión. El consumo energético depende del tiempo y la cantidad de energía que ocupan los dispositivos BLE en cada una de las etapas o estados de funcionamiento utilizados para transmitir información. Estos estados son los siguientes [23]:

- **Discover:** En estado *Discover* el dispositivo está listo para recibir paquetes de publicación de dispositivos remotos. Esta fase termina cuando el primer mensaje de publicación es detectado.
- **Connect:** En modo *Connect* el dispositivo espera por otro paquete de publicación, y envía una respuesta de conexión inmediatamente después de recibirlo. Cuando esa respuesta es recibida la fase de conexión termina, en caso de 6 intervalos de intento de conexión fallidos, la conexión es considerada como perdida. En ese caso un nuevo requerimiento de conexión es enviado.
- **Update:** En estado *Update* los parámetros de conexión son actualizados para el uso de canales.
- **Transfer:** En etapa *Transfer* una solicitud de lectura es enviada desde el dispositivo central hasta un dispositivo periférico. Luego de recibir una respuesta del dispositivo periférico esta fase finaliza.
- **Idle:** En las transiciones *Idle* la conexión se mantiene activa por 500 ms sin transmitir ningún dato.
- **Disconnect:** Al ejecutarse el modo *Disconnect* una solicitud de desconexión es enviada al dispositivo periférico y la fase termina cuando la conexión es conocida por el periférico.
- **Sleep:** En modo *Sleep* ambos dispositivos son puestos en modo dormido para ejecutar la calibración de corriente.
- **Advertising:** En fase *Advertising* el dispositivo central está continuamente escaneando, y el dispositivo periférico está continuamente publicando.

La Fig. 15 y la Fig. 16 indican el consumo de corriente que se genera en un dispositivo BLE mientras transmite, recibe y procesa información:

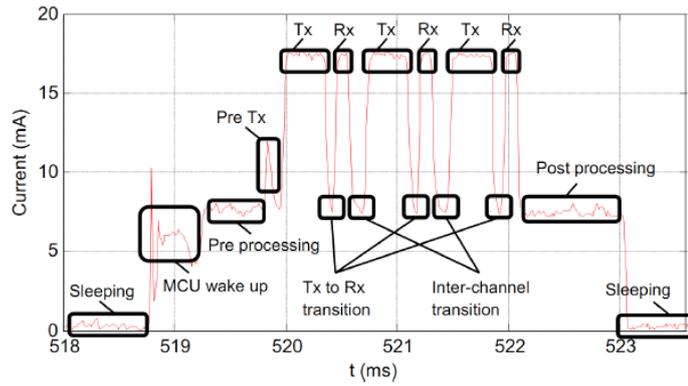


Fig. 15. Característica de una conexión BLE en modo de publicación [24].

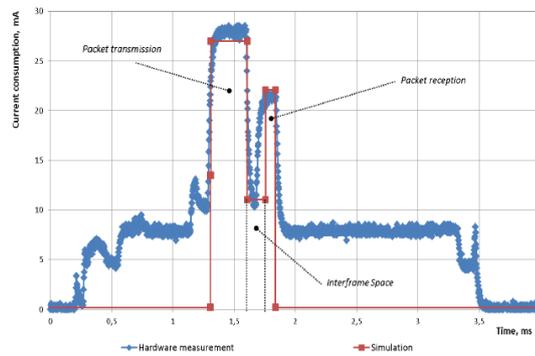


Fig. 16. Característica de una conexión BLE en Tx y Rx [25].

2.2.6 Generalidades del estándar IEEE 802.11

El estándar desarrollado para redes inalámbricas de área local es el IEEE 802.11, a través de los años se han desarrollado nuevas enmiendas de este protocolo con mejoras o extensiones en base a los anteriores. Utiliza la banda ISM no licenciada para realizar las transmisiones de RF [26]. La Tabla 3 muestra algunas de las características de las enmiendas del estándar 802.11:

Tabla 3. Enmiendas del estándar 802.11 [26].

| Estándar | Frecuencia | Técnica de modulación | Tasa de transmisión |
|----------|-----------------|-----------------------|---------------------|
| 802.11a | 5 GHz | OFDM | 54 Mbps |
| 802.11b | 2.4 GHz | DSSS, CCK | 11 Mbps |
| 802.11g | 2.4 GHz | OFDM, CCK, DSSS | 54 Mbps |
| 802.11n | 2.4 y 5 GHz | OFDM | 300 Mbps |
| 802.11ac | 5 GHz | OFDM | 1.3 Gbps |
| 802.11ad | 2.4, 5 y 60 GHz | OFDM | 7 Gbps |

Modos de funcionamiento

El estándar 802.11 define dos modos operativos [27]:

- **Modo de infraestructura:** El modo infraestructura presenta un escenario como el de la Fig. 17 en el que los clientes se conectan inalámbricamente a un punto de acceso.
- **Modo Ad-hoc:** En modo Ad-Hoc como el de la Fig. 18 los clientes se conectan entre sí sin ningún punto de acceso, es decir se puede considerar a cada uno de los involucrados como un AP.

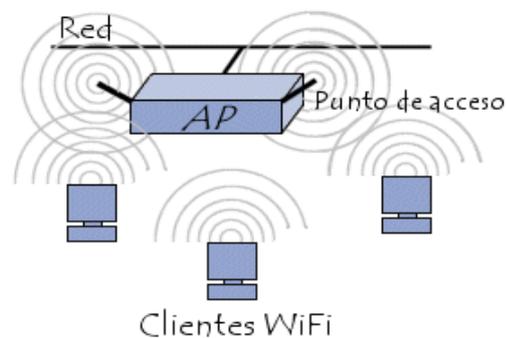


Fig. 17 Modo infraestructura de redes 802.11 [27].

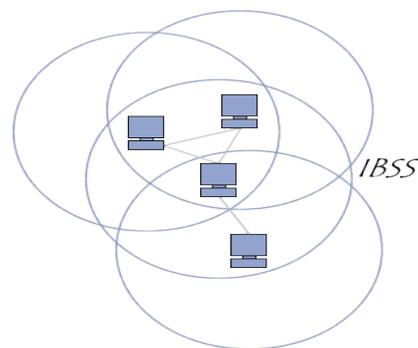


Fig. 18 Modo Ad-hoc de redes 802.11 [27].

Rango de frecuencias

El rango de frecuencias utilizadas en el estándar IEEE 802.11 se encuentra en la banda ISM (industrial, científica y médica) definida por la UIT se trabaja en un rango de 2,4 GHz hasta los 2,4835 GHz. Estándares como el “a y n” utilizan la banda de los 5GHz UNII (*Unlicensed-National Information Infrastructure*) en el rango de 5.15-5.35 GHz

y 5.725-5.825 GHz en EE. UU., mientras en otros países este rango varía. Esta última banda tiene mayores ventajas ya que la banda de 2,4 GHz está saturada por redes y dispositivos que utilizan la misma banda de frecuencia, aunque sin embargo son más susceptibles a fenómenos físicos como la lluvia [26].

Modulaciones

Las modulaciones más utilizadas en estándar IEEE 802.11 son: QPSK, BPSK, QAM, 16QAM, CCK, DQPSK. En la actualidad cada una de las modulaciones básicas digitales mencionadas se las trata con técnicas de transmisión de datos más avanzadas como OFDM. Los equipos inalámbricos utilizan varios tipos de modulación según la enmienda del estándar principal 802.11, es posible configurar un equipo para que trabaje en un solo estándar por ejemplo el “802.11b” o también podrían configurarse con modos mixtos. Múltiples equipos cuentan con la opción “*mixed-bgn*” capaz de utilizar los estándares b, g y n aleatoriamente. [28].

Modulación QPSK

La modulación QPSK (*Quadrature phase shift keying*) también llamada codificación por cambio de fase en cuadratura, donde la información de la señal que se transmite está en la fase, con un desplazamiento de fase de cuatro símbolos con 90° de diferencia y cada símbolo aporta 2 bits. El ancho de banda para esta modulación es la mitad de la velocidad de transmisión original ya que su procesamiento divide la señal en dos partes para la generación de los dos bits necesarios para asignar valores al desfase [29]. Una señal modulada en fase QPSK tiene la forma de la Fig. 19:

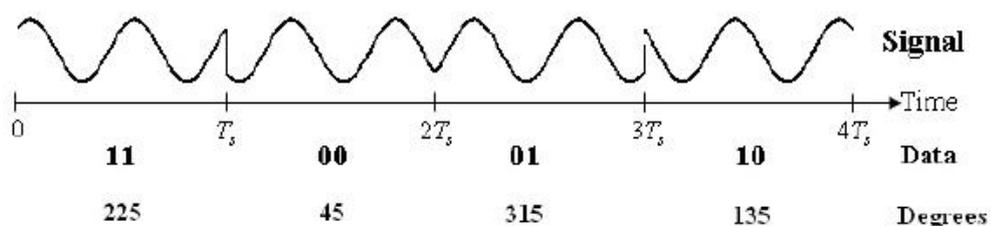


Fig. 19. Diagrama de modulación QPSK [29].

OFDM

OFDM (*Orthogonal Frequency-Division Multiplexing*) es una idea de multiplexación de división de frecuencia (FDM) donde múltiples señales se transmiten al mismo tiempo, pero en diferentes frecuencias. Un solo transmisor puede utilizar docenas o millares de frecuencias ortogonales, es decir una relación de fase específica entre sí para minimizar interferencias entre ellas. Una señal OFDM es la suma de un número de subportadoras ortogonales donde cada subportadora es modulada usando QAM, PSK u otro tipo de modulación digital [26].

FHSS

FHSS (*Frequency Hopping Spread Spectrum*) es una técnica de modulación que se basa en cambiar constantemente de frecuencia saltando de una a otra de manera aleatoria, estas frecuencias dentro la banda ISM van desde los 2,4 GHz hasta los 2,4835 GHz, todo este ancho de banda dividido en 79 canales de 1 MHz. Los cambios de frecuencia se realizan cada 300 a 400 ms, aunque este tipo de modulación no es muy utilizado en la actualidad [26].

DSSS

DSSS (*Direct Sequence Spread Spectrum*) es una técnica de modulación la cual por cada bit que representa a un dato genera una secuencia pseudoaleatoria que debe ser transmitida, según el estándar 802.11 un "1" se representa con 11 bits por ejemplo 10110111000 y un "0" como su complemento 01001000111. Cada bit se codifica en una secuencia de impulsos más cortos que ocupan el mismo intervalo de tiempo del bit original sin utilizar diferentes frecuencias. Este tipo de modulación es el más utilizado en redes inalámbricas actuales [26].

MAC 802.11

La MAC en la capa de enlace de datos del protocolo 802.11 consta de dos partes: control de acceso al medio (MAC) y control lógico del enlace (LLC). La subcapa LLC

de 802.11 es idéntica a la de 802.2 permitiendo una compatibilidad con cualquier otra red 802, mientras que la MAC debido a su naturaleza inalámbrica tiene algunos cambios sustanciales.

Esta MAC sustituye al estándar 802.3 (CSMA-Ethernet) de redes cableadas, obviamente con funcionalidades específicas para radio como fragmentación, control de errores, retransmisiones de tramas y acuses de recibido [26].

Método de acceso al medio CSMA/CA

CSMA/CA (*Carrier Sense Multiple Access with Collision Avoidance*) es un método de acceso al medio que intenta evitar colisiones utilizando paquetes de acuse de recibido ACK, donde una estación o nodo que recibió la información envía este paquete para confirmar que la información llegó correctamente.

En CSMA/CA la estación que desea transmitir, sensa el aire y si no se detecta actividad o dicho de otra manera el canal inalámbrico está libre, la estación espera un tiempo seleccionado aleatoriamente y entonces transmite si el medio continua libre.

Si este paquete se recibe satisfactoriamente se devuelve un ACK al transmisor y el proceso de recepción culmina. Si el ACK no es detectado por la estación transmisora se asume que hubo colisión y que el paquete se perdió, por lo tanto, vuelve a ser retransmitido luego de esperar un tiempo aleatorio [30].

Protocolo UDP

UDP (*User Datagram Protocol*) es un protocolo a nivel de transporte que permite enviar datos en la red sin que se haya establecido previamente una conexión, ya que el datagrama contiene la suficiente información de destino y fuente en su cabecera. Este protocolo debido a su estructura no permite conocer si el paquete llegó correctamente al destino, tampoco tiene un control de flujo por lo que algunos paquetes tienden a adelantarse a otros. Además, proporciona una sencilla interfaz entre capa de red y la capa de aplicación y su estructura se indica en la Fig. 20 [31]:

| + | Bits 0 - 15 | 16 - 31 |
|----|----------------------|----------------------|
| 0 | Puerto origen | Puerto destino |
| 32 | Longitud del Mensaje | Suma de verificación |
| 64 | Datos | |

Fig. 20. Estructura de cabecera UDP [31].

Protocolo TCP

TCP (*Transmission Control Protocol*) es uno de los protocolos más usados en una red de datos para conexiones entre dispositivos garantizando que la información sea entregada sin errores y en el mismo orden en el que se transmiten. Sus principales características son [31]:

- Orientado a la conexión, donde los sistemas de los extremos se sincronizan para manejar el flujo de información.
- Operación full-dúplex.
- Revisión de errores con una técnica llamada Checksum usada para verificar que los datos no sean erróneos.
- Acuses de recibo.
- Controla el flujo si el transmisor desborda el buffer del receptor por transmitir demasiado rápido, el receptor descarta paquetes, lo cual alerta al transmisor de reducir la tasa de transferencia o dejar de transmitir.
- El receptor puede solicitar retransmitir un paquete, si el paquete no es notificado como recibido (ACK) el trasmisor retransmite el mensaje.

Su estructura se indica en la Fig. 21:

| Bits | 0-3 | 4-7 | 8-15 | 16-31 |
|------|---------------------------------|-----------|-------|-----------------|
| 0 | Puerto Origen | | | Puerto Destino |
| 32 | Número de Secuencia | | | |
| 64 | Número de Acuse de Recibo (ACK) | | | |
| 96 | Longitud Cabecera TCP | Reservado | Flags | Ventana |
| 128 | Suma de Verificación (Checksum) | | | Puntero Urgente |
| 160 | Opciones + Relleno (Opcional) | | | |
| 224 | DATOS | | | |

Fig. 21. Estructura de cabecera TCP [31].

2.2.7 Generalidades de redes LTE

LTE es un estándar desarrollado por el grupo *3rd Generation Partnership Project (3GPP)* para comunicaciones móviles de alta velocidad, buscando mejorar los picos máximos de descarga de datos, tener un ancho de banda escalable, mejorar la eficiencia espectral, utilizar IP en toda la red, soportar multitud de tipos diferentes de usuarios y reducir la brecha de intercambio de datos con redes de datos fijas.

Estándares en comunicaciones móviles

Los estándares y tecnologías de comunicaciones móviles en la actualidad convergen y trabajan en conjunto para dar cobertura a la mayor cantidad de usuarios en una determinada zona geográfica, por lo que es común que un dispositivo conectado a una red LTE, pierda cobertura de la misma y se conecte a una red HSPA. La Tabla 4 muestra una comparación entre diversas tecnologías de comunicaciones móviles:

Tabla 4. Estándares en comunicaciones móviles [32].

| Generación | Estándar | Nombre | Velocidad máxima de descarga | Velocidad máxima de subida |
|------------|----------|---|------------------------------|----------------------------|
| 2G | GSM | Global System for Mobile Communications | 14.4 Kbits/s | 14.4 Kbits/s |
| 2G | GPRS | General Packet Radio Service | 53.6 Kbits/s | 26.8 Kbits/s |
| 2.5 G | EDGE | Enhanced Data rates for GSM Evolution | 217.6 Kbits/s | 108.8 Kbits/s |

| | | | | |
|---------------|--------------|---|------------------------------|--------------|
| 3G | UMTS | Universal Mobile Telecommunications System | 384 Kbits/s | 128 Kbits/s |
| 3.5 G | HSPA | High-Speed Packet Access | 7.2 Mbits/s | 3.6 Mbits/s |
| 3.75 G | HSPA+ | Evolved High-Speed Packet Access - Release 6 | 14.4 Mbits/s | 5.76 Mbits/s |
| 3.75 G | HSPA+ | Evolved High-Speed Packet Access - Release 7 | 21.1 Mbits/s or 28.0 Mbits/s | 11.5 Mbits/s |
| 3.75 G | HSPA+ | Evolved High-Speed Packet Access - Release 8 | 42.2 Mbits/s | 11.5 Mbits/s |
| 3.75 G | HSPA+ | Evolved High-Speed Packet Access - Release 9 | 84.4 Mbits/s | 11.5 Mbits/s |
| 3.75 G | HSPA+ | Evolved High-Speed Packet Access - Release 10 | 168.8 Mbits/s | 23.0 Mbits/s |
| 4 G | LTE | Long Term Evolution | 100 Mbits/s | 50 Mbits/s |
| 4 G | LTE-A | Long Term Evolution - Advanced | 1 Gbits/s | 500 Mbits/s |

Arquitectura

La arquitectura en LTE con su modelo SAE (*System Architecture Evolution*) de red simplificada utiliza un número pequeño de elementos para el despliegue de la red, abaratando costos de inversión para operadores de LTE y mejorando además el desempeño y latencia de la red. Esta arquitectura simplificada se puede observar en la Fig. 22 [33]:

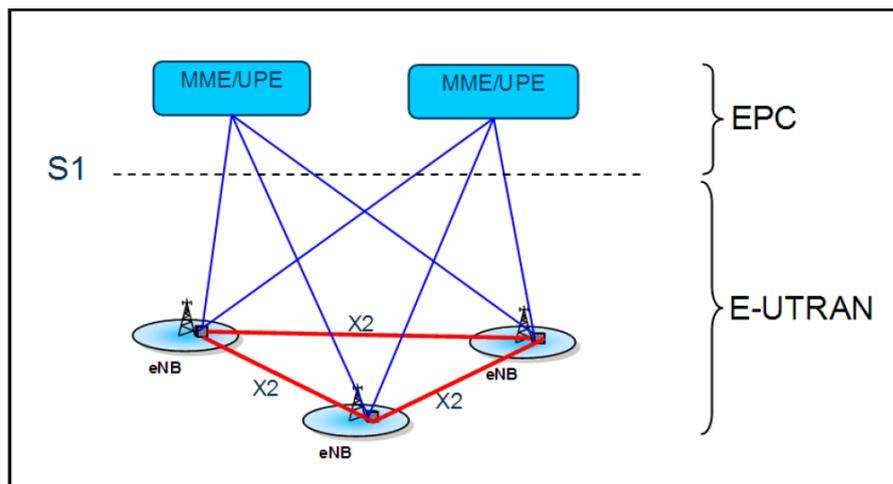


Fig. 22. Arquitectura de red LTE [33].

Entre los elementos se destacan [33]:

- **eNB (Evolved NodeB):** Los eNBs son el hardware al que se conectan los equipos terminales (UE) a través de la interfaz S1 a la EPC (*Evolved Packet Core*) que maneja el tráfico de datos en LTE). Sus funciones son: manejo de recursos de radio, control de admisión de radio, control en conexiones móviles y planificación dinámica de recursos.
- **MME (Mobility Management Entity):** El MME es una entidad perteneciente al plano de control dentro de la EPS (*Evolved Packet System*) desempeña la gestión de la movilidad de usuario entre eNBs y distribuye mensajes paging hacia los eNBs para localizar dispositivos. Cumple con funciones como señalización, autenticación, gestión de portadoras, etc. Muchas otras funcionalidades pueden ser configuradas en el MME según especificaciones TS 23.xxx.
- **UPE (User plane entity):** Los UPE son dispositivos que realizan la compresión de cabeceras IP y cifrado de flujo de datos de usuario, termina los paquetes de plano de usuario debido a razones de paging y generan el switcheo el plano de usuario debido a soporte de movilidad de UE.
- **SWG (Serving Gateway):** El SWG es un elemento del núcleo que encamina los datos del usuario y tiene la capacidad de proveer manejo de QoS usados por otras redes.
- **PGW (Packet Data Network Gateway):** La PGW es la interfaz que interconecta las comunicaciones de datos con las redes externas. Provee filtrado de datos vía inspección profunda de paquetes, con el objetivo de manejar la gestión de movilidad y handovers hacia redes que no pertenecen a la 3GPP como WiMAX, CDMA, etc.

El funcionamiento de la EPC se despliega según la Fig. 23:

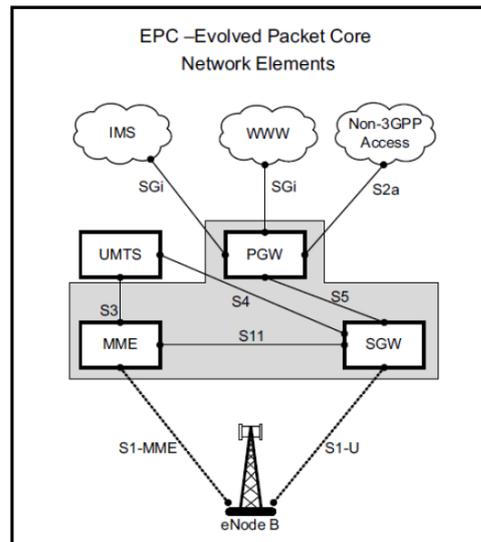


Fig. 23. Diagrama Lógico de EPC [34].

Rango de frecuencias

El rango de frecuencias en una red LTE es desplegada usando varios canales de RF con diferentes anchos de banda, estos anchos de banda son indicados en la Fig. 24:

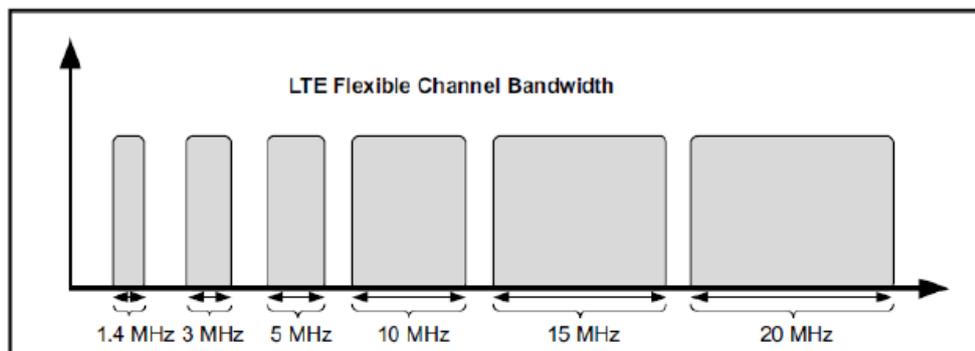


Fig. 24. Anchos de banda para portadoras en LTE [34].

Por lo tanto, una operación de LTE tiene un ancho de banda de 5, 10, 15 o 20 MHz; anchos de banda menores de 5 MHz también son compatibles, es decir, 1,4 y 3 MHz para el modo FDD (Frequency Division Multiplex). Esto asegura que LTE funciona en la mayoría de las asignaciones de espectro, es decir, en un espectro apareado para FDD (bandas de enlaces ascendente y descendente separadas) o ser implementado en

un espectro no apareado con canales de 1,4 y 3 MHz utilizando TDD (Time Division Multiplexing, empleando la misma frecuencia de enlaces ascendente y descendente separadas en el tiempo).

Esto permite un alto grado de flexibilidad, y re-uso de las bandas existentes utilizadas para los sistemas 2G y 3G, e incluso permite coexistir con estas tecnologías en los espectros adyacentes [34]. La Tabla 5 muestra los cuatro proveedores de telefonía móvil 4G que existen en Ecuador, y sus bandas de operación son las siguientes:

Tabla 5. Asignación de bandas LTE en Ecuador [35].

| Operadora | Banda LTE | Enlace de Subida (MHz) | Enlace de bajada (MHz) |
|-----------------|-----------|------------------------|------------------------|
| CLARO | Banda 4 | 1710 - 1755 | 2110 - 2155 |
| CNT | Banda 4 | 1710 - 1755 | 2110 - 2155 |
| | Banda 28 | 703 - 748 | 758 - 803 |
| MOVISTAR | Banda 2 | 1850 - 1910 | 1930 - 1990 |
| TUENTI | Banda 2 | 1850 - 1910 | 1930 - 1990 |

Para el enlace de bajada la tecnología utilizada es OFDMA (*Orthogonal Frequency Division Multiple Access*) y para el enlace de subida SC-FDMA (*Single Carrier Frequency Division Multiple Access*). En la Fig. 25 se muestra un mapa de cobertura de señal LTE 4G de Ambato, obtenido desde el software *Open Signal* instalado en un celular Android:

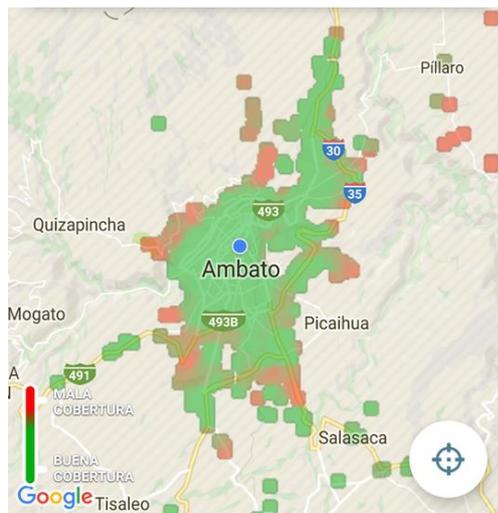


Fig. 25. Mapa de cobertura de señal 4G LTE de la ciudad de Ambato.

Fuente: Investigador.

OFDMA

OFDMA (*Orthogonal Frequency-Division Multiple Access*) es una variante de OFDM modificada para telefonía celular, se hace uso de un gran número de subportadoras muy cercanas entre sí que se transmiten en paralelo. Cada subportadora es modulada con un método convencional (QPSK, 16QAM, etc.) a una velocidad de símbolos baja, pero combinando todas las subportadoras permite velocidades de datos similares a los esquemas de modulación de portadora única.

Se utiliza la transformada rápida de Fourier (FFT) para pasar del dominio del tiempo al de frecuencia, en el mismo que existen varias subportadoras, moduladas cada una de forma independiente, mientras tanto en el dominio del tiempo existen intervalos de guarda para evitar interferencias entre símbolos como se ilustra en la Fig. 26 [36].

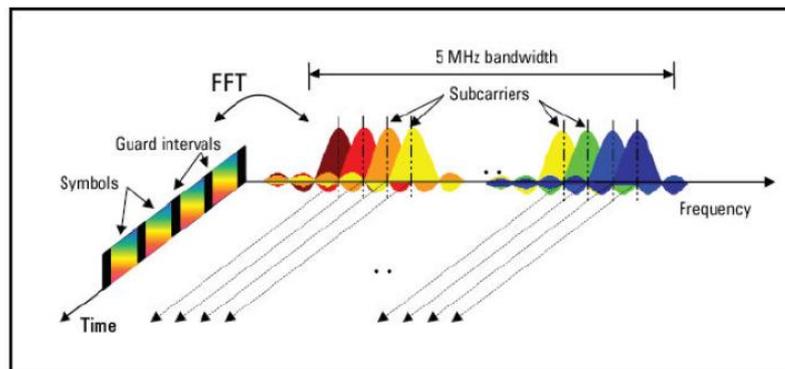


Fig. 26. Señal OFDM representada en tiempo y frecuencia [36].

SC-FDMA

El SC-FDMA (*Single Carrier Frequency Division Multiple Access*) es un método de acceso múltiple por división de frecuencia de portadora única manejado para reducir la alta relación de pico a promedio o *peak to average* (PAR) ya que combina técnicas similares a las de GSM y CDMA con resistencia a multicamino y la asignación flexible de frecuencias de OFDMA.

En la Fig. 27 se indica como los símbolos en el dominio del tiempo se convierten al dominio de la frecuencia usando una transformada discreta de Fourier (DFT); a continuación, en el dominio de la frecuencia que se asignan a la ubicación deseada en el ancho de banda total del canal antes de ser convertida de nuevo

al dominio del tiempo usando una FFT, por último, se inserta un Cyclic Prefix (CP).

La Fig. 28 expone la generación de señal SC-FDMA, el proceso sigue los mismos pasos que para OFDMA. Realización de una transformación inversa DFT que convierte la señal desde el dominio de la frecuencia al dominio del tiempo y el hecho de insertar el CP proporciona la robustez fundamental de OFDMA contra trayectos múltiples [36].

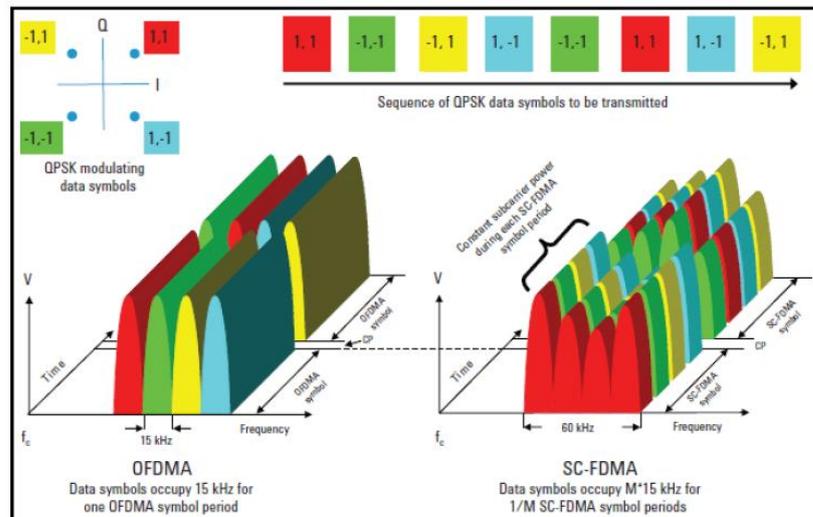


Fig. 27. Comparación entre OFDMA y SC-FDMA [36].

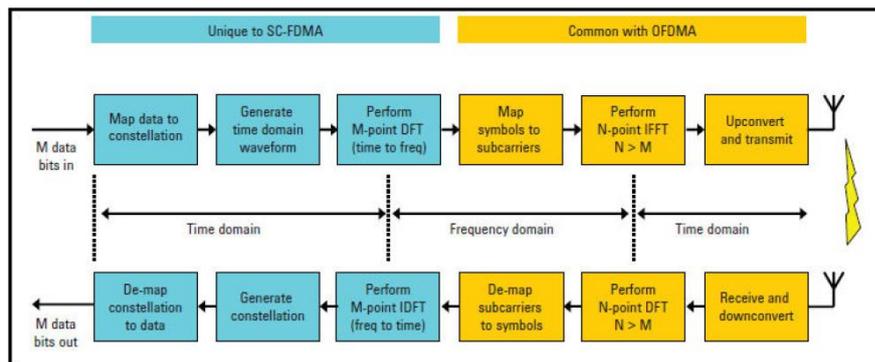


Fig. 28. Modelo de generador y receptor de señal OFDMA y SC-FDMA [36].

Los elementos de recursos y los bloques de recursos (*Resource Block - RB*) de 0,5 ms forman parte de la estructura de SLOT LTE, donde la unidad más pequeña de frecuencia/tiempo para el enlace descendente es el elemento de recurso. Un Slot se

compone de 7 elementos de recursos o símbolos, y cada uno de ellos corresponde a una subportadora OFDM de 15 KHz durante un intervalo de símbolo OFDM.

En la Fig. 29 se observa que un grupo de 12 subportadoras equivale a 180 KHz, que es un bloque de recursos. Los datos correspondientes a cada usuario se le asigna en términos bloque de recurso, en un mínimo de dos bloques de recursos consecutivos denominados bloques de planificación (*scheduling blocks*) [33].

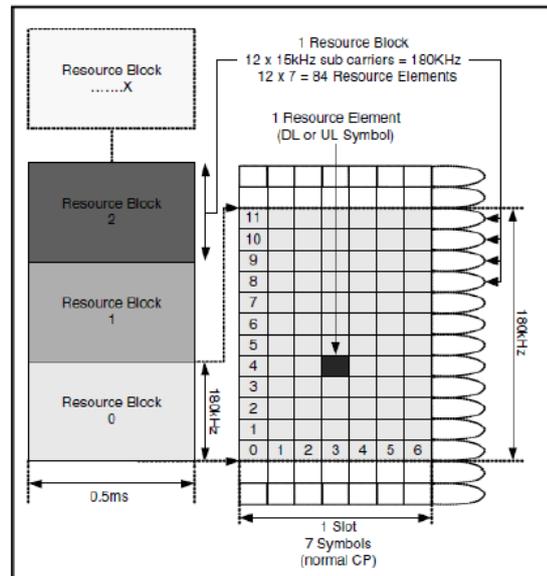


Fig. 29. Estructura Slot, elementos de Recursos y Bloques de Recursos [33].

La información que corresponde a cada usuario se estructura según la Fig. 30:

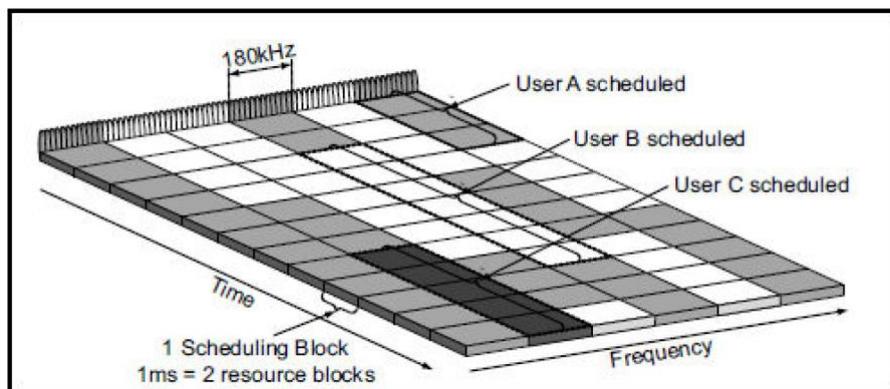


Fig. 30. Programación de usuarios o User scheduling [33].

Según el ancho de banda seleccionado sea para el enlace de subida o de bajada en número de bloque y subportadoras se determinan según la Tabla 6:

Tabla 6. Bloques de recursos, subportadoras y ancho de banda en LTE [33].

| Ancho de banda | Bloques de recursos | Subportadoras (downlink) | Subportadoras (uplink) |
|----------------|---------------------|--------------------------|------------------------|
| 1.4 MHz | 6 | 73 | 72 |
| 3 MHz | 15 | 181 | 180 |
| 5 MHz | 25 | 301 | 300 |
| 10 MHz | 50 | 601 | 600 |
| 15 MHz | 75 | 901 | 900 |
| 20 MHz | 100 | 1201 | 1200 |

Tipo de asignación de recursos

El Tipo de asignación de recursos o *Resource allocation type*, caracteriza la forma en la que el planificador o *scheduler* asigna los bloques de recurso para cada transmisión. La red asigna un conjunto de RB contiguos. Pero este RB contiguo es de concepto "virtual", no el concepto "físico". Aunque la capa MAC asigna los múltiples RB contiguos, es posible que no se alineen contiguamente cuando se transmiten en la capa PHY. Esto significa que debería haber una regla / algoritmo para convertir esta asignación RB lógica (virtual) a la asignación física de RB. Hay dos tipos de conversión, uno es "localizado" y el otro es "distribuido". Cuando se selecciona 'localizada', tanto la asignación de virtual como la asignación física asignan RB de forma contigua. Cuando se selecciona 'distribuido', la asignación RB virtual es contigua, pero la asignación física no es contigua (se distribuyen sobre rangos de frecuencia más amplios) [37].

Tipos de celdas

El tipo de celda se lo determina según el tamaño del área de cobertura a la cual se brinda cobertura de red celular móvil [38]:

- **Megaceldas:** Las megaceldas son de amplia cobertura mayor a 35 km con tráfico ocasional.

- **Macroceldas:** Las Macroceldas son celdas para áreas de población dispersa, ambientes urbanos poco densos, su área de cobertura va desde 2 a 20 km.
- **Microceldas:** Las Microceldas abarcan un área intensa en tráfico y un ambiente densamente poblado, cubre áreas entre 200 m a 2 km.
- **Picoceldas:** Las Picoceldas se utilizan en ambientes urbanos intensos, cubren calles o edificios.

Interfaces X2

Las Interfaces X2 conectan las estaciones de radio de LTE (*eNodeBs*) entre sí, por las cuales se envían mensajes de señalización para gestionar de manera eficiente los recursos de radio de la red, evitando y reduciendo el índice de interferencias para el mejor manejo del tráfico de datos de los usuarios cuando se realiza en proceso de *handover* [39].

Handover

El Handover es una función principal de los *eNodeBs* en LTE y están destinados a reducir el tiempo de interrupción en comparación con el proceso de transferencia de conmutación de circuitos de redes 2G. Los *eNodeBs* de LTE deben ejecutar transferencias tanto dentro de la E-UTRAN (inter/intra eNB) como entre las tecnologías de acceso por radio (RAT) heredada como UMTS. En la Tabla 7 se detallan los procedimientos de Handover [40]:

Tabla 7. Procedimientos de Handover en LTE [40].

| Procedimiento de traspaso | Descripción |
|---|--|
| Inter eNB Handover (UE saliente) | Traspaso dentro de una E-UTRAN. Procedimiento para cuando un UE abandona una celda gestionada por el eNB y entra en una celda gestionada por un segundo eNB. |
| Inter eNB Handover (UE entrante) | Traspaso dentro de una E-UTRAN. Procedimiento para cuando un UE está entrando en una celda gestionada por el eNB desde una célula gestionada por un segundo eNB. |
| Inter RAT Handover (UE saliente) | Entrega de una E-UTRAN a otra RAT. Procedimiento para cuando un UE abandona la célula E-UTRAN. |

| | |
|---|--|
| Inter RAT Handover (UE entrante) | Entrega de una E-UTRAN a otra RAT. El procedimiento para cuando un UE está entrando en la célula E-UTRAN. |
| Intra eNB Handover (UE saliente) | Traspaso de un sector a otro gestionado por el mismo eNB. Procedimiento para cuando un UE abandona el sector. |
| Intra eNB Handover (UE entrante) | Traspaso de un sector a otro gestionado por el mismo eNB. El procedimiento para cuando un UE está entrando en el nuevo sector. |

Received signal code power (RSRP)

EL RSRP es usado para la medición del enlace de bajada mediante el protocolo *Radio Resource Control (RRC)*, sus valores van desde -140 dBm hasta -44 dBm con resolución de 1 dB. Su principal objetivo es determinar la mejor interfaz de radio enlace descendente y seleccionarla como celda servidora de un equipo de usuario. Por otro lado, la *Reference Signal Received Quality (RSRQ)* también se usa con el mismo propósito, sin embargo no es un valor absoluto de medición de potencia recibida como el *RSRP* sino que se considera como la relación señal a ruido.

El rango de RSRQ va desde -19.5 dB a -3 dB con una resolución de 0.5 dB. Si un *Ue* cambia su ubicación o si las condiciones de radio cambian y el RSRP permanece estable o incluso es mejor que antes y el RSRQ está disminuyendo, esto indica un aumento de interferencia. Si ambos parámetros disminuyen al mismo tiempo, claramente es un indicio de falta de cobertura de la red. En general valores de RSRQ superiores a -9 dB garantizan mejor experiencia para el usuario, entre -9 y -12 dB es visto como neutro con una pequeña degradación del QoS, para valores de -13 dB e inferiores la calidad de la comunicación y el rendimiento es muy bajo [33].

Las condiciones del canal de RF dependen de los parámetros antes mencionados, y el índice de calidad sigue la tendencia de la Fig. 31:

| | | RSRP (dBm) | RSRQ (dB) | SINR (dB) |
|---------------|-----------|-------------|------------|-----------|
| RF Conditions | Excellent | ≥ -80 | ≥ -10 | ≥ 20 |
| | Good | -80 to -90 | -10 to -15 | 13 to 20 |
| | Mid Cell | -90 to -100 | -15 to -20 | 0 to 13 |
| | Cell Edge | ≤ -100 | ≤ -20 | ≤ 0 |

Fig. 31. Niveles de RSRP, RSRQ y SINR necesarios en LTE [41].

Channel Quality Indicator (CQI)

El CQI es una información que el *UE* envía al *eNodeB* la cual reporta que tan buena es la calidad del canal de enlace descendente que recibe, donde el programador puede reaccionar a esa información de retroalimentación asignando nuevos recursos de radio, como por ejemplo cambiando el esquema de modificación y modulación (MCS) o transmitir los siguientes bloques de datos en diferentes subportadoras.

Los informes CQI indican la calidad de recepción de cada subportadora con respecto al promedio de banda ancha, catalogándolas como: peor, igual, mejor y mucho mejor. El *UE* debe escoger un esquema MCS a partir de la información de medición anterior, que está indexado del 0 al 15 para adaptarse al objetivo de un *Block Error Rate (BLER)* que debe ser menor al 10%. Para un mayor CQI se obtiene una mejor modulación y una mayor tasa de código como se indica en la Fig. 32 [33].

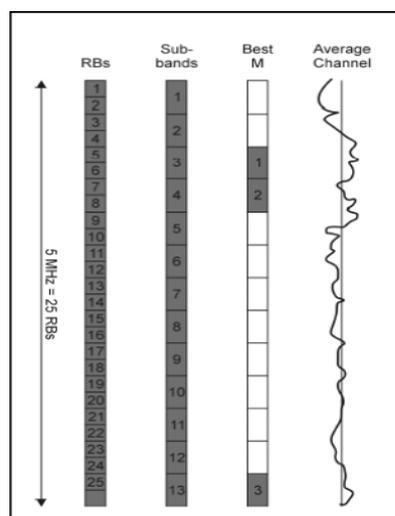


Fig. 32. Ilustración del CQI en sub bandas y la mejor banda M [33].

LTE MAC

La capa MAC en LTE existe tanto en los *Ue* como en los *eNodeBs*, y es parte del proceso de control de la interfaz de aire, los principales servicios y funciones de la subcapa MAC son [42]:

- Mapeo entre canales lógicos y canales de transporte.
- Multiplexación / demultiplexación de MAC.
- Programación de informes de información.
- Corrección de errores a través de HARQ.
- Manejo de prioridad entre canales lógicos de un *Ue*.
- Manejo de prioridad entre *Ues* mediante programación dinámica.

Esquema de modulación y codificación (MCS)

Modulation and codification scheme (MCS), es una combinación entre modulación y tasa de codificación que se utiliza en diversos tipos de comunicaciones inalámbricas como Wi-Fi o LTE. Estos sistemas de radio se adaptan a las características de señal y de ruido del canal de RF, para ello cambian la velocidad de modulación, es decir, los índices mostrados en la Tabla 8 para así brindar un óptimo balance entre SNR, velocidad de datos y tasa de errores [43].

Tabla 8. Esquemas de modulación y codificación MCS [43].

| Índice MCS | Modulación y Codificación | Índice MCS | Modulación y Codificación |
|------------|---------------------------|------------|---------------------------|
| 0 | BPSK 1/2 | 15 | 256-QAM 5/6 |
| 1 | QPSK 1/2 | 16 | BPSK 1/2 |
| 2 | QPSK 3/4 | 17 | QPSK 1/2 |
| 3 | 16-QAM 1/2 | 18 | QPSK 3/4 |
| 4 | 16-QAM 3/4 | 19 | 16-QAM 1/2 |
| 5 | 64-QAM 2/3 | 20 | 16-QAM 3/4 |
| 6 | 64-QAM 3/4 | 21 | 64-QAM 2/3 |
| 7 | 64-QAM 5/6 | 22 | 64-QAM 3/4 |
| 7 | 256-QAM 3/4 | 23 | 64-QAM 5/6 |
| 7 | 256-QAM 5/6 | 23 | 256-QAM 3/4 |
| 8 | BPSK 1/2 | 23 | 256-QAM 5/6 |
| 9 | QPSK 1/2 | 24 | BPSK 1/2 |
| 10 | QPSK 3/4 | 25 | QPSK 1/2 |
| 11 | 16-QAM 1/2 | 26 | QPSK 3/4 |

| | | | |
|----|-------------|----|------------|
| 12 | 16-QAM 3/4 | 27 | 16-QAM 1/2 |
| 13 | 64-QAM 2/3 | 28 | 16-QAM 3/4 |
| 14 | 64-QAM 3/4 | 29 | 64-QAM 2/3 |
| 15 | 64-QAM 5/6 | 30 | 64-QAM 3/4 |
| 15 | 256-QAM 3/4 | 31 | 64-QAM 5/6 |

HARQ y AMC

En las redes de telefonía móvil y redes de datos existen entornos cada vez más ruidosos y a la vez mayores requerimientos de los usuarios por lo que los operadores de red deben soportar las grandes demandas del mercado mientras se utilizan recursos mínimos para ofrecer un servicio de buena calidad. El estándar LTE aprovecha la modulación y codificación adaptativa (AMC) y procesos de solicitud de repetición automática híbrida con el fin de minimizar el tiempo de respuesta y maximizar el rendimiento de datos del sistema.

El HARQ utiliza paquetes de datos ACK para mensajes recibidos y NACK cuando no se recibe el dato y se necesita retransmitir. En cualquier caso (ACK o NACK), la entidad transmisora está obligada a programar y procesar la próxima transmisión dentro de un período de tiempo específico. Para métodos TDD se configuran el número de procesos HARQ y para métodos FDD exactamente existen 8 procesos HARQ para el enlace ascendente y en el descendente un máximo de 8 procesos.

En cambio, la AMC intenta hacer coincidir las transmisiones de un proceso HARQ con las condiciones de canal. Bajo condiciones de señal fuertes, se emplea menos redundancia y / o un formato de modulación de orden superior en la transmisión inicial, lo que permite una tasa de datos de usuario más alta para un ancho de banda dado. En condiciones de señal débil, se utilizan más bits de redundancia y / o se utiliza un formato de modulación de orden inferior para mejorar la probabilidad de recepción [44].

Rank Indicator (RI)

El RI es una información de control que un *Ue* reporta al *eNodeB* en *Physical Uplink Control Channel (PUCCH)* o *Physical Uplink Shared Channel (PUSCH)* basado en la programación de enlace ascendente. En un caso de tener dos antenas de transmisión el número de bits informados en RI es de uno, un bit de 0 indica RANK1 y un bit de 1 indica RANK2 [45].

- **RANK1:** El RANK1 indica que el *Ue* está viendo un buen SNIR solamente en una de sus antenas de recepción, solicitando así que al *eNodeB* que reduzca el modo de transmisión a una sola antena o transmita diversidad.
- **RANK2:** El RANK2 es un indicador en el *Ue* que significa un buen SNIR en ambos puertos de antena y *eNodeB* puede programar MIMO.

2.2.8 Parámetros para el análisis de comunicaciones inalámbricas

Los parámetros relacionados a la propagación de señales en un medio inalámbrico reflejan que tan fiable es la comunicación de extremo a extremo, la calidad del canal, nivel de interferencias, ganancias y pérdidas de propagación en el sistema, incluyendo otros indicadores de la calidad de la comunicación y a continuación se detallan.

Eb/No y Bit Error Rate (BER)

BER significa la tasa de error de bits y es la relación entre el número de bits erróneos recibidos en el receptor y el número total de bits transmitidos desde el transmisor. Se utiliza la Ecuación 2:

$$BER = \frac{\text{bits erróneos recibidos}}{\text{número total de bits enviados}} \quad (2)$$

Con una buena señal de fuente y sin interferencias este valor se vuelve insignificante. Se vuelve significativo cuando se debe tener una relación de señal a ruido (SNR) en presencia de una transmisión imperfecta desde los equipos y el medio de propagación [46]. El BER además puede ser interpretado como una probabilidad de error (P_e), Ecuación 3:

$$Pe = \frac{1}{2}(1 - \text{erf}) \sqrt{E_b/N_o} \quad (3)$$

Donde “*erf*” representa la función de error que cambia según el método de modulación digital, E_b (joules) se refiere a la energía de un bit y determina al dividir la potencia de la onda portadora entre la tasa de bits y N_o (joules*segundo*hercio) que es la densidad de potencia en el espectro del ruido [47].

El E_b/N_o se encuentra según la Ecuación 4 y es especialmente útil para encontrar el BER de distintos esquemas de modulación digital:

$$\frac{E_b}{N_o} = \frac{\text{relación de señal ruido (SNR)}}{\text{eficiencia espectral (E)}} \quad (4)$$

La eficiencia espectral está definida por la Ecuación 5:

$$\rho \left(\frac{\text{bps}}{\text{Hz}} \right) = \frac{\text{tasa de bits (Rb)}}{\text{ancho de banda (B)}} \quad (5)$$

Block Error Rate (BLER)

El BLER es la proporción de bloques erróneos recibidos del total de bloques de datos enviados. Por ejemplo, en un esquema de codificación 400 bits de información son enviados en un bloque. BLER solamente muestra el resultado de OK o no OK por cada bloque; en cambio el *Data Block Error Rate (DBLER)* mide esta relación con la relación matemática de la Ecuación 6 [46]:

$$DBLER = \frac{\text{bloques erróneos recibidos}}{\text{número total de bloques enviados}} \quad (6)$$

Relación señal a ruido

La relación señal a ruido abreviada SNR o S/N es una medida usada en ingeniería que compara el nivel de una señal deseada con el nivel de ruido de fondo, la Ecuación 7 indica esta relación:

$$SNR = \frac{\text{potencia de la señal (S)}}{\text{potencia del ruido (N)}} \quad (7)$$

En la Ecuación 8 se detalla que el SNIR análogamente a la SNR se define como la potencia de señal (S) dividida para la suma de las interferencias de otras señales (I) y la potencia de ruido de fondo (N) [48]:

$$SNIR = \frac{S}{I + N} \quad (8)$$

Ruido

Ruido se conoce a todo componente de tensión o intensidad de corriente que causa superposición a una señal de información que se trasmite. El ruido puede ser generado por diferentes fuentes como el ruido causado por la agitación térmica de electrones del mismo sistema, acoplamiento magnético o el generado por dispositivos electrónicos externos o por fenómenos generales como tormenta o neblina [49].

Factor y figura de ruido

El factor de ruido (F) es una magnitud que mide la contribución a los niveles de ruido que realiza un amplificador o sistema de medida. Es adimensional y se define con la Ecuación 9:

$$F = \frac{\frac{S_i}{N_i}}{\frac{S_o}{N_o}} \quad (9)$$

Que son las relaciones de señal a ruido de entrada y salida sea del sistema o un amplificador. Frecuentemente, este factor de ruido se lo expresa en dB, a esta forma se la denomina figura de ruido y se lo halla según lo dice la Ecuación 10 [49]:

$$NF = 10 \log(F) \quad (10)$$

Sensibilidad y detección de energía

La sensibilidad establece un valor de potencia mínimo para que la información de una señal de radio sea receptada y por lo general está expresada en dBm, en el caso de que la señal recibida sea mayor al umbral de sensibilidad el dispositivo de radio recibe información y cambia de estado pasivo a ocupado. Cuando ninguna señal es detectada, esto significa que el RSSI es menor al nivel de detección de energía [50].

El RSSI (*Received Signal Strength Indicator*) es un término en radio frecuencia que indica el nivel de potencia que está recibiendo un dispositivo por parte de una infraestructura de radio en un determinado lugar y tiempo, sea de una red Wi-Fi, 3G, 4G, etc. [41].

Símbolos

El símbolo es un conjunto de k bits considerados como unidad o carácter m_i , de un conjunto finito de símbolos o alfabeto. El tamaño del alfabeto está dado por la Ecuación 11:

$$M = 2^k \quad (11)$$

Donde k es el número de bits dentro del símbolo, cada símbolo m_i ($i=1, \dots, M$) es representado por su correspondiente forma de onda $s_1(t), s_2(t), \dots, s_M(t)$. En la Fig. 33 se muestran varios ejemplos y su representación gráfica: un símbolo binario tiene $k=1$ y $M=2$ y un cuaternario tiene $k=2$ y $M=4$ [51].

- 1 Binary symbol ($k = 1, M = 2$)
- 10 Quaternary symbol ($k = 2, M = 4$)
- 011 8-ary symbol ($k = 3, M = 8$)

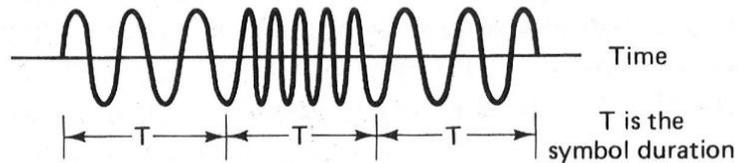


Fig. 33. Símbolos y representación de forma de onda [51].

En todo canal con un ancho de banda específico existen efectos que causan dispersión de los impulsos cuadrados sean bits o símbolos. Cuando los símbolos son consecutivos esa dispersión genera que el parte de la energía de símbolo se solape con los símbolos vecinos causando la llamada Interferencia Intersímbolo (ISI).

Interferencia inter símbolos (ISI)

La ISI degrada la capacidad del receptor para diferenciar un símbolo real a partir de la energía que se ha solapado entre símbolos adyacentes. Incluso sin la existencia de ruido de existir este tipo de interferencia lleva a errores de detección. La ISI se controla manipulando las características de filtrado del canal y de cualquier procesado en el transmisor o en el receptor, de manera que no degrade la proporción de bits de error (BER) del enlace [52].

Delay Spread

El retardo de esparcimiento o *Delay Spread* es uno de los efectos que causan la ISI, donde un símbolo que es enviado por el transmisor es recibido varias veces por el receptor, asemejándose a un eco, esto se muestra en la Fig. 34:

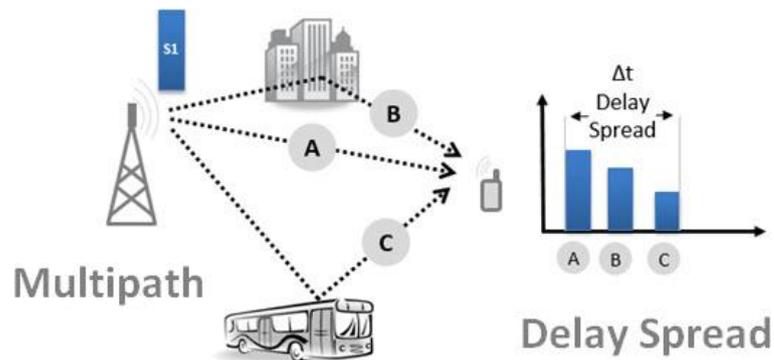


Fig. 34. Efecto Delay Spread [53].

Un mensaje que sale desde el transmisor toma múltiples trayectorias mientras se propaga y llega a su destino, por el efecto de estas múltiples trayectorias llega al receptor en diferentes instantes de tiempo como replicas. El tiempo total transcurrido depende del ambiente, por ejemplo, en un ambiente urbano donde el reflejo es alto por los edificios y automóviles este retraso tiene un valor típico de 5-10 microsegundos [53].

Multitrayectoria

La multitrayectoria como se ilustra en la Fig. 35 se produce cuando la señal se propaga por múltiples caminos hasta un receptor y las señales llegan en diferentes instantes al receptor:

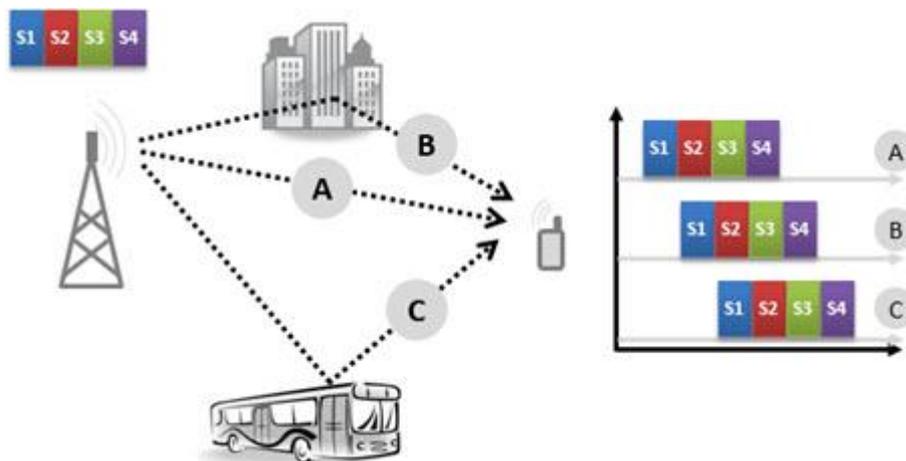


Fig. 35. Efecto Multitrayectoria [53].

En receptor las señales y sus componentes se suman, como se muestra en la figura los símbolos S3, S2 y S1 llegan al mismo instante y se suman, el resultado práctico es que se tiene una sobre posición o *Overlap* de símbolos, causando la Interferencia intersimbólica [53].

Desvanecimiento Rayleigh

El desvanecimiento Rayleigh es el causado por la propagación multitrayectoria en los sistemas de comunicación inalámbrica está basado comúnmente por una distribución con el mismo nombre. Una muestra de señal con desvanecimiento Rayleigh presenta caída de amplitud de la señal en función del tiempo cuando la antena receptora se mueve con una velocidad constante. Las caídas de potencia ocurren ocasionalmente y de forma aleatoria, con tendencia a aparecer cada media longitud de onda de movimiento como en la Fig. 36.

Existe un modelo de desvanecimiento basado en el Rayleigh, se llama el modelo Jakes, que también añade un efecto Doppler al proceso de comunicación, esto quiere decir que la señal sufre ciertas modificaciones en frecuencia debido al movimiento de la fuentes de información [54].

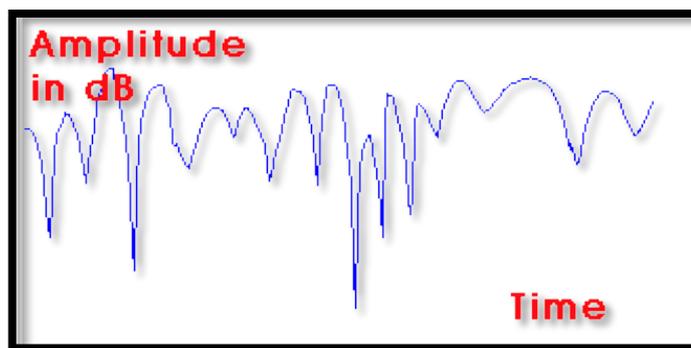


Fig. 36. Señal afectada por Rayleigh Fading [54]

Modelos de propagación

Los modelos de propagación permiten realizar cálculos de propagación en los sistemas inalámbricos tomando en cuenta las condiciones del medio que pueden variar como la neblina, lluvia, edificios, etc. Los modelos de propagación se utilizan para predecir las

pérdidas a través del tramo entre la estación transmisora y la receptora, existen varios factores a tener en cuenta como son: el perfil del terreno de la zona de cobertura a modelar, la presencia de obstáculos como edificios y árboles [39].

Modelo de Propagación en el espacio libre

El modelo de propagación en el espacio libre considera visibilidad de las estaciones, de manera ideal cuando no existen obstáculos entre el emisor y el receptor. El más utilizado en este tipo de modelo es el de Friss, la Ecuación 12 indica cómo se obtiene la potencia recibida.

$$Pr = \frac{Pt Gt Gr \lambda^2}{L (4\pi d)^2} \quad (12)$$

Donde Pr (potencia recibida a una distancia d), Pt (potencia transmitida), Gt (ganancia de antena transmisora), Gr (ganancia de antena receptora), λ (longitud de onda de la señal portadora), d (distancia entre antena transmisora y receptora) y L (factor de pérdidas por filtros, líneas de transmisión, acopes, amplificadores. Debe ser menor a 1) [39].

Con la Ecuación 13 se encuentran las pérdidas por espacio libre:

$$L(\text{dB}) = 92,45 + 20 \log f(\text{GHz}) + 20 \log d(\text{Km}) \quad (13)$$

QoS

QoS o calidad de servicio es un conjunto de tecnologías que le proporcionan la capacidad para administrar el tráfico de red de manera rentable y mejorar las experiencias de usuario en entornos empresariales, oficinas pequeñas e incluso entornos de red domésticos. Las redes deben proveer servicios seguros, medibles y garantizados, para lograr una calidad de servicio QoS se manejan los retardos, variación de llegada de datos, ancho de banda y parámetros de paquetes perdidos. QoS es un conjunto de técnicas para manejar los recursos de las redes [55].

Los protocolos en comunicaciones inalámbricas y móviles permiten a las estaciones definir la calidad de servicio basado en cuatro características [19]:

- **Confiabilidad:** la falta de confiabilidad significa pérdida de paquetes o su reconocimiento, lo cual implica retransmisión.
- **Retardo (Delay):** El retardo es el tiempo que tarda en realizarse una correcta comunicación entre fuente y destino.
- **Jitter:** El Jitter es la variación de retardo entre paquetes que se envían y se reciben. Por ejemplo, si paquetes se envían en los tiempos 0, 1, 2 y llegan al receptor en los tiempos 20, 21, 22 el Jitter será de 20 unidades de tiempo.
- **Ancho de banda:** El Ancho de banda utilizado por diferentes aplicaciones depende de cuantos bits se envían por segundo.

Throughput de la red

El Throughput indica la cantidad de datos transmitidos en bytes o bits, en un determinado periodo de tiempo expresado en segundos. Este indicador permite conocer el nivel de rendimiento de la red, es decir la capacidad que tiene la red para manejar la información que circula en ella. El throughput se calcula en un nodo o para toda la red la Ecuación 14 es utilizada para aquello [56]:

$$\text{Throughput de la red} = \frac{\text{Bits totales recibidos}}{\text{Tiempo de transmisión (seg)}} \quad (14)$$

Retardo de extremo a extremo

Retardo de extremo a extremo (*End to end Delay*), es el tiempo que transcurre en llegar la información desde el origen al destino. En este parámetro solo se consideran los paquetes que han llegado satisfactoriamente al receptor. La Ecuación 15 permite calcular el tiempo promedio de retardo de extremo a extremo [56]:

$$\text{Promedio de retardo} = \frac{\text{Suma de los retardos de todos los paquetes}}{\text{Total de paquetes recibidos}} \quad (15)$$

Relación de entrega

Relación de entrega (*Delivery Ratio*), es el porcentaje de paquetes recibidos con éxito y permite tener una visión del congestionamiento presente en la red, es un indicador

muy importante al momento de analizar el desempeño de los paquetes. Para encontrar este parámetro se utiliza la Ecuación 16 [56]:

$$Delivery\ Ratio = \frac{Número\ de\ paquetes\ recibidos}{número\ de\ paquetes\ transmitidos} * 100 \quad (16)$$

2.2.9 Aplicaciones médicas en redes WBAN y WPAN

Las aplicaciones médicas de las WBAN se clasifican dentro de tres categorías [3]:

- **WBAN Wearable o Portátil:** Las WBAN Portables son una clase de aplicaciones donde los biosensores tienen la característica principal de portabilidad, además se catalogan en dos clases: Asistencia por Discapacidad y Gestión del Performance humano.
- **WBAN Implantable:** las WBAN implantables son aplicaciones en las que se relaciona a los nodos implantados dentro del cuerpo humano, ya sea debajo de la piel o en la corriente sanguínea.
- **Control remoto de dispositivos médicos:** La conectividad ubicua a Internet permite la conexión en red de los dispositivos y servicios de atención a domicilio. Con ello se pretende prolongar el autocuidado de los pacientes, minimizando la dependencia del cuidado intensivo personal, lo que aumenta la calidad de vida y la disminución de los costos relacionados a tratamientos medicinales, fomentando una nueva generación de sistemas de TI con características tales como: el comportamiento de anticipación, sensibilidad, facilidad de uso y flexibilidad.

La clase y número de sensores a emplearse en redes WBAN y WPAN dependen básicamente del escenario de aplicación [3]:

- **Sensor de presión arterial:** El sensor de presión arterial trabaja apoyado sobre una arteria de la muñeca que detecta la forma de la onda de presión a medida que la sangre fluye a través de ella. Nuevos dispositivos que tienen el tamaño y la apariencia de un reloj de pulsera miden la presión de forma continua, proporcionando una medida mucho más precisa.
- **Electrocardiógrafo (ECG):** El electrocardiógrafo es un sensor que registra gráficamente la actividad eléctrica del corazón. Empleado para ayudar a

diagnosticar una enfermedad cardíaca; a fin de obtener una señal los electrodos se colocan en varios sitios específicos sobre la piel (p. ej., los brazos y el pecho), y las diferencias de potencial entre estos electrodos son las que se miden.

2.2.10 Telemedicina

La telemedicina se define como “El suministro de servicios de atención sanitaria, en los que la distancia constituye un factor crítico, por profesionales que apelan a las tecnologías de la información y de la comunicación con objeto de intercambiar datos para hacer diagnósticos, preconizar tratamientos y prevenir enfermedades y heridas, así como para la formación permanente de los profesionales de atención de salud y en actividades de investigación y evaluación, con el fin de mejorar la salud de las personas y de las comunidades en que viven” [57].

2.2.11 Monitor Médico

Un monitor médico es un equipo de uso hospitalario destinado a la adquisición de información de parámetros biológicos de un paciente, de carácter gráfico, como electrocardiograma ECG, curva de saturación parcial de oxígeno SpO₂, presión arterial invasiva, curva respiratoria y de carácter numérico como la presión arterial, saturación parcial de oxígeno, frecuencia cardíaca y temperatura. El monitor médico dispone de una serie de sensores que conectados al paciente recogen la información del mismo y mediante un sistema de adquisición de datos dicha información es digitalizada, ofreciendo la posibilidad de presentarla en la propia pantalla del monitor y de transmitirla a otros sistemas externos, opción muy útil cuando se supervisa un conjunto de enfermos desde un único puesto.

La velocidad neta del canal de comunicaciones necesaria para transmitir las señales fundamentales del monitor médico (7 derivaciones del ECG, presión arterial invasiva, respiración y SpO₂, numéricas y alarmas; un total de 10 gráficas) corresponde a 22

kbits por segundo (para una conexión tipo Ethernet). La solución de usar sistemas de redes inalámbricas de tipo comercial, como Bluetooth, Zigbee o IEEE 802.11 para la transmisión de esta información médica desde los diferentes monitores médicos hasta la unidad central de control de pacientes, es válida para los traslados dentro del complejo hospitalario, pero totalmente inviable para los traslados interhospitalarios [58].

2.2.12 Presión Arterial

La presión arterial (PA) es el volumen de sangre generado por el bombeo del corazón hacia todo el cuerpo, actúa sobre la sangre dentro de las arterias, éstas son vasos sanguíneos por los que pasa la sangre desde el corazón hasta todas las partes vivas del organismo, variando en el transcurso del día debido al ritmo circadiano, el cual opera mediante un reloj biológico sincronizado, controlando funciones en el organismo como el sueño y el comportamiento. La falta de monitoreo de la PA es una de las principales causas del aumento de la Hipertensión Arterial (HTA) que representa por sí misma una enfermedad, como también un factor de riesgo importante para otras enfermedades, como las cerebrovasculares, cardíacas, renales, entre otras. Para contextualizarlo mejor, si la presión arterial fuera cero la sangre se mantendría atrapada en las arterias, y el organismo no obtendría los recursos necesarios para su supervivencia, concluyendo con la muerte. La Fig. 37 ilustra los niveles de presión arterial [59]:

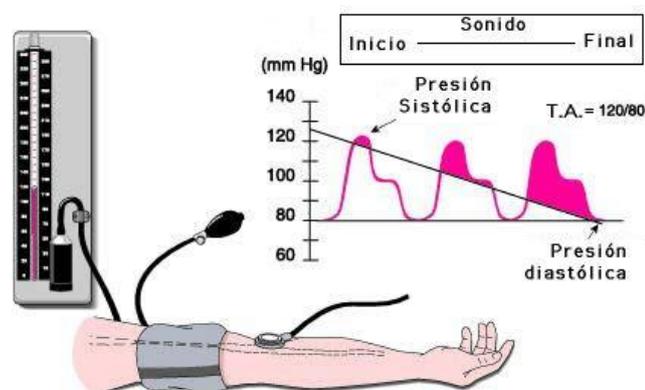


Fig. 37. Ilustración de la presión arterial sistólica y diastólica [60].

Hipertensión e Hipotensión

La Hipertensión Arterial no es más que un aumento fuera de los rangos normales tolerantes por el ser humano, que viene definida por la presión arterial sistólica a partir de 140mmHg o una presión arterial diastólica a partir de 90mmHg. Es necesario tener en cuenta el riesgo que tienen los pacientes hipertensos en relación con las enfermedades cardiovasculares, el cual se determina no sólo por el nivel de su presión arterial sino además por la presencia o ausencia de daño en órganos o de otros factores de riesgo como el tabaquismo y diabetes, entre otros. Por su parte la hipotensión arterial (presión baja) es el estado de la presión arterial sistólica y diastólica por debajo de lo normal. Sin embargo, hay que considerar que lo que, para una persona es una presión sanguínea baja, para otra puede ser normal. Cuando la presión sanguínea está demasiado baja, hay un flujo inadecuado de sangre al corazón, al cerebro y a los demás órganos vitales, lo que ocasiona diversas patologías debido a que, junto con la sangre no llegan suficiente oxígeno y nutrientes a las células [59].

2.2.13 Frecuencia cardiaca

Existen dos tipos de frecuencia cardiaca que se obtienen por métodos o herramientas de adquisición de signos vitales [61]:

- ***Frecuencia cardiaca periférica:*** La frecuencia cardiaca periférica es determinada después de tomar lecturas en localidades del cuerpo humano alejadas del corazón.
- ***Frecuencia cardiaca central:*** La frecuencia cardiaca central se la obtiene tomando medidas del pulso en el tórax a la altura del corazón.

Electrocardiograma

El Electrocardiograma (ECG) es uno de los métodos más utilizados y más fiables para obtener la frecuencia cardiaca e información derivada del corazón. En un electrocardiograma se obtiene los pulsos eléctricos como el de la Fig. 38 que son generados por medio de electrodos de plata con gel conductor situados en zonas de

interés que a su vez se conectan a un electrocardiógrafo; herramienta diseñada para procesar las señales eléctricas y representarlas de manera gráfica [61].

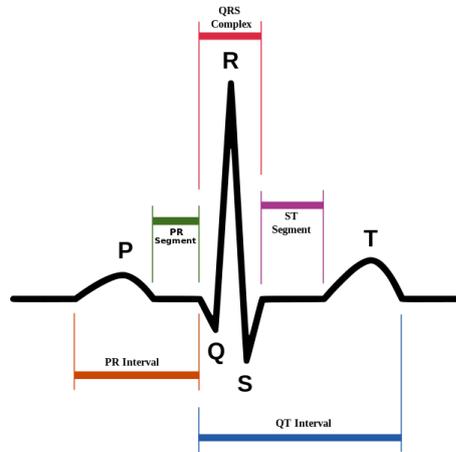


Fig. 38. Gráfica de frecuencia cardiaca [61].

2.2.14 Simulación de redes

Las simulaciones de red han sido utilizadas desde hace mucho tiempo para estudiar modelos e ideas, además de contribuir con el desarrollo de prototipos y sistemas de una manera más ágil sin la necesidad implementar físicamente estos sistemas. Las simulaciones tratan de recrear ambientes reales usando modelos teóricos, ecuaciones y expresiones matemáticas. Los simuladores de red deben permitir la parametrización de diferentes escenarios con variables como el medio, tecnologías, protocolos y movilidad. Existen tres tipos de simuladores [62]:

- **Simuladores de tiempo real:** Los simuladores de tiempo real son buenos para casos cuando el tiempo real es relevante, es decir para simulaciones que son sensibles al tiempo.
- **Simulación continua:** La simulación continua es un sistema computacional de un sistema físico que continuamente sigue la respuesta de los dispositivos en base a un conjunto de ecuaciones, son mayormente usados en física para fenómenos de robótica, trayectoria de cohetes, etc.
- **Simulación de eventos discretos:** Las simulaciones de eventos discretos son configuradas para representar una secuencia de eventos en el tiempo, donde no se debe esperar el intervalo de tiempo hasta que un evento suceda.

Existen múltiples simuladores de redes que han sido desarrollados para fines específicos en el área de las redes y las comunicaciones, según su tipo algunos son de naturaleza totalmente discreta como NS2, otros permiten simulaciones de evento discretos y de tiempo real como NS3 y OMNeT++ y por su puesto existen aquellos simuladores como WDSi y Mac802.11 [62].

2.2.15 OMNeT++

OMNeT++ es una plataforma de simulación está basada en Eclipse y muchas de sus funcionalidades. OMNeT++ incluye la capacidad de introducir y configurar modelos (archivos “.ned” e “.ini”), realizando varias simulaciones y ejecutando el análisis de sus resultados, mientras que Eclipse da el soporte del código fuente C++, integración y otras características opcionales de código abierto y comercial [63].

OMNeT++ es un simulador de eventos discretos orientado a objetos y usado para múltiples soluciones en redes:

- Modelado de redes alámbricas e inalámbricas.
- Modelado de protocolos.
- Modelado de multiprocesadores y sistemas de hardware distribuido.
- Validación de arquitectura de hardware.
- Evaluación del funcionamiento de sistemas complejos de software.
- En general modela y simula cualquier sistema donde una aproximación de eventos discretos es posible, y puede ser convenientemente trazados en entidades de comunicación con intercambio de mensajes.

Las entidades o modelos son módulos simples, y juntos forman módulos compuestos, estos modelos son reusables y son combinados de diferentes maneras como bloques de LEGO. Como se ilustra en la Fig. 39, los módulos están conectados vía puertas y conexiones para transportar e intercambiar mensajes con diferentes estructuras, estos módulos están programados en C++ y hacen uso de librerías de simulación [64].

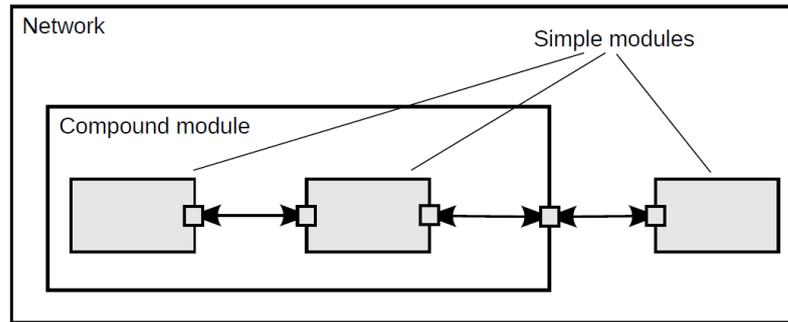


Fig. 39. Módulos simples y compuestos [64].

Parametrización

La parametrización permite que cada módulo tenga diferentes variables y estas son asignados tanto en los archivos “.ned” o en los archivos de configuración de simulación *omnetpp.ini*. Los parámetros son una palabra, números, valores booleanos o contener una estructura de archivo “.xml” [64].

El lenguaje NED

NED significa *Network Description* o Descripción de Red, que permite al usuario declarar, conectar y ensamblar módulos simples para generar módulos compuestos o redes. El IDE de OMNeT++ permite realizar conexiones entre módulos desde una interfaz de gráficos [64].

El archivo INI

Los archivos INI parametrizan y configuran los modelos de simulación para su ejecución. Son archivos de texto con extensión “.ini” que se modifican desde cualquier editor de texto, en OMNeT++ el editor de archivos INI es muy útil porque tiene conocimiento del modelo de simulación e indica si existen errores de sintaxis o incoherencias en la parametrización [63].

2.2.16 INET Framework

INET es un framework de OMNeT++ que contiene implementaciones de protocolos IPv4, IPv6, TCP, SCTP, UDP y varios modelos de aplicación. También incluye un modelo MPLS con RSVP-TE y señalización LDP. Los modelos de capa de enlace son PPP, Ethernet y 802.11. El enrutamiento estático se configura al usar un auto configurador de red, o utilizando diferentes implementaciones de protocolos de enrutamiento. Además soporta comunicaciones inalámbricas y móviles.

Inet como Framework de simulación basado en OMNeT++

OMNeT++ 5 contiene entre sus ejemplos todo el código de INET, y funciona correctamente sobre esta versión. Inet utiliza el concepto de traspaso de mensajes entre módulos, hosts, routers, teléfonos móviles, switches y otros dispositivos de red están representados por módulos compuestos (compound modules) de OMNeT++. Dichos módulos están ensamblados por módulos simples (simple modules) que representan aplicaciones, protocolos y unidades funcionales. Una red también en un módulo compuesto que contiene hosts, routers, etc. Las interfaces externas de los módulos están descritas en los archivos “.ned”, los cuales describen los parámetros y puertas (gates) como puertos y conectores de módulos, y también submódulos y conexiones de módulos compuestos.

Los módulos están organizados de manera jerárquica como indica la Fig. 40, formando paquetes en forma de árbol, y organizados de acuerdo a las capas del modelo OSI, todos estos paquetes corresponden al directorio *src/inet* del paquete Inet como está definido en el archivo *src/inet/package.ned*.

| | |
|----------|---------------------|
| src/inet | inet.applications |
| | inet.transportlayer |
| | inet.networklayer |
| | inet.linklayer |
| | inet.physicallayer |

Fig. 40. Módulos principales del framework INET.

Fuente: Investigador.

La Fig. 41 muestra otros paquetes que se incluyen en INET:

| | |
|----------|------------------|
| src/inet | inet.common |
| | inet.environment |
| | inet.mobility |
| | inet.node |
| | inet.power |
| | inet.routing |
| | inet.visualizer |

Fig. 41. Módulos secundarios del framework INET.

Fuente: Investigador.

Los módulos simples están implementados en clases C++ con el mismo nombre, y en el mismo directorio del archivo “.ned”. El paquete `inet.node` contiene dispositivos de red prediseñados como `hosts`, `router`, `Switch`, etc. Pero no son universales y pueden ser modificados para escenarios particulares de simulación. Las interfaces de red alámbricas e inalámbricas son módulos compuestos de `Mac` y posiblemente otros módulos simples.

No todos los módulos implementan protocolos, `IPv4RoutingTable` contiene datos, `IPv4NetworkConfigurator` auto configura una red, `ConstSpeedMobility` mueve nodos móviles y `RadioMedium` que desarrolla limpiezas asociadas con señales de radio. Las cabeceras y formatos de los paquetes están descritos en los archivos `msg`, que son traducidos a clases C++ por OMNeT++.

Elaborar y ejecutar simulaciones

Para elaborar una simulación en un archivo “.ned” se deben especificar los componentes de la red como son routers, hosts y otros dispositivos de red. Los módulos de la red tienen muchos parámetros sin asignar y que necesitan ser asignados antes de ejecutar la simulación. El nombre de la red y los valores de los parámetros se especifican en el archivo `omnetpp.ini`. Este archivo contiene asignaciones de parámetros como líneas `key=values`, cada línea es un patrón de comodín. El simulador

coincide estos patrones de comodín con la ruta completa del árbol de módulos en la que se encuentra el parámetro.

Ejemplo: *Test.host[2].tcp.nagleEnabled*

El valor de la línea con la que coincide será asignado, en caso de no coincidir se usará un valor por defecto. Hay dos tipos de comodines:

- **El asterisco simple ***: El asterisco simple coincide con máximo un componente en la especificación de la ruta. No coincide puntos y corchetes. (. , [,]).
- **Doble asterisco ****: El doble asterisco enmarca a múltiples componentes. Los patrones son capaces de expresar rangos de índices (***host[1..3,5,8].tcp.nagleEnabled*) y rangos de números embebidos en nombres (***switch{2..3}.relayUnitType*).

En el archivo *omnetpp.ini* se realizan varias configuraciones, bajo las secciones [*Config <name>*], y la configuración deseada se selecciona cuando se ejecuta la simulación. Es posible utilizar extensiones de otros archivos, es decir utilizar las configuraciones de otros archivos con *extends=<name>*. Las configuraciones de la sección [*General*] se aplican a todas las demás configuraciones. En ocasiones es necesario utilizar parámetros con múltiples valores con la sintaxis *\${10, 50, 100..500 step 100, 1000}* [50].

2.2.17 SIMULte INET

SIMULte es una herramienta innovadora que permite analizar y evaluar a un nivel complejo el funcionamiento de una red LTE y LTE avanzada (3GPP versión 8 y más allá). Está escrito en C++ y es completamente personalizable con una sencilla interfaz donde se permite desarrollar nuevos módulos, nuevos algoritmos y protocolos. Además, es un proyecto de código abierto basado en OMNeT++ e INET.

SIMULte permite a los investigadores simular y comparar sus configuraciones en un marco fácil de usar. Usando la capacidad de manejar módulos y submódulos de OMNeT+, se extienden sus funcionalidades según la necesidad del programador y se logra integrar módulos de INET fácilmente inclusive VEINS para redes LTE vehiculares [65]. Varias de sus características se muestran en la Tabla 9:

Tabla 9. Principales características de SIMULte [65].

| Aplicaciones: | RLC: | MAC: |
|---|--|--|
| VoIP GSM AMR, Video Streaming H.264, Real-time gaming, FTP, etc. | UM y segmentación AM y retransmisiones re ensambladas (AM only). | Buffering, concatenación PDU, recepción de CQI, selección de formato de transporte y asignación de recursos, codificación diseñada para facilitar el análisis entre capas. |
| PHY: | Terminal de usuario: | E-NodeB: |
| Transmite diversidad usando curvas SINR, cálculos del canal de retorno y modelo de canal muy relista. | Movilidad, interferencia, todos los tipos de tráfico, aplicaciones: VoIP, juegos, VoD, web, etc., handover, comunicaciones D2D, etc. | Macro, micro, pico eNodeBs, Inter-eNB a través de interfaces X2, soporte para handover, soporte para CoMP, algoritmos de planificación: Max C/I, Round Robin, etc. |

2.3 Propuesta de solución

La simulación del sistema de comunicación inalámbrico integrado ambulatorio de presión arterial, será una herramienta técnica para verificar la factibilidad de la implementación de dicho sistema de manera masiva, con lo cual el prototipo en un futuro podrá ser puesto en el mercado e implementado en cientos de personas del cantón Ambato de la provincia de Tungurahua, logrando una reducción del porcentaje de víctimas fatales debido a problemas relacionados con la presión arterial.

CAPÍTULO III

Metodología

3.1 Modalidad de la investigación

La investigación es de modalidad aplicada, donde se ha puesto en práctica los conocimientos adquiridos en el transcurso de los nueve semestres de la Carrera de Ingeniería en Electrónica y Comunicaciones de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, aplicando técnicas de diseño prototipos electrónicos, redes de comunicaciones y desarrollo de software.

La información utilizada en este proyecto fue obtenida con la investigación bibliográfica-documental de libros, revistas, artículos y base de datos científica de la Universidad Técnica de Ambato referentes a las tecnologías de comunicación inalámbrica y simulación en plataformas de software libre. Dicha información proporcionó el conocimiento suficiente como teorías, resultados, experimentaciones, instrumentos y técnicas usadas para realizar el proceso de simulación.

Además, con la investigación de campo se obtuvo la información verídica del comportamiento del prototipo implementado, los resultados e inconvenientes obtenidos se han simulado para analizar los diferentes escenarios planteados.

3.2 Recolección de información

Se realizó la recolección de información referente a resultados de pruebas aplicadas en pacientes, con el dispositivo ya conectado al paciente y este a su vez enviando datos al dispositivo móvil del usuario.

3.3 Procesamiento y análisis de datos

Una vez obtenida la información referente a los estudios del prototipo de manera individual se procesó los datos para simularlos con la plataforma de software libre, dicha simulación genera los mismos resultados del prototipo pero de manera masiva, es decir con decenas o cientos de dispositivos, ya con la simulación en correcto funcionamiento se realizó de análisis de resultados con gráficas de cada canal de comunicación utilizado.

3.4 Desarrollo del proyecto

Para llevar a cabo el desarrollo del proyecto se aplicó la siguiente metodología:

- Análisis de los estándares de comunicaciones inalámbricas y móviles que se utilizarán en la simulación.
- Determinación de la estructura, arquitectura, tecnología y los posibles métodos de comunicación entre el paciente y el servidor médico en base a los requerimientos del sistema.
- Recolección de información del funcionamiento del prototipo.
- Procesamiento y análisis de la información recolectada.
- Estudio de campo y pruebas en pacientes.
- Recolección de datos, fallos, tiempos de transmisión, interferencias.
- Establecimiento de los parámetros tecnológicos necesarios para la elección de la plataforma de software libre a ser utilizada.
- Desarrollo de la simulación del sistema de comunicación inalámbrico integrado ambulatorio de presión arterial.

- Pruebas del funcionamiento de la simulación con gran cantidad de dispositivos.
- Análisis y tabulación de resultados de las simulaciones realizadas.
- Elaboración del informe final.

CAPÍTULO IV

Desarrollo del proyecto

4.1 Introducción

El control médico de forma remota y con tecnología de punta se ha convertido en tema de investigación con enfoque productivo, por lo que se requiere de un análisis riguroso para justificar la elaboración y la utilidad de un sistema médico nuevo.

Mediante simulaciones se analiza y procesa información para evaluar parámetros como latencia, retardo, pérdida de paquetes y consumo de energía; además las problemáticas de capa física, capa de red, canal inalámbrico, empaquetado de información, procesamiento de señales e interferencias necesitan ser simuladas para saber cuáles son las características que debe tener un sistema médico para funcionar correctamente.

El presente proyecto analiza la eficiencia del Sistema Integrado Ambulatorio de Presión Arterial en un entorno real, desarrollado por los investigadores de la FISEI. Detallando el análisis del dispositivo electrónico, métodos de comunicación, arquitectura, tecnologías utilizadas y sus características de funcionamiento. A partir de esa información se ha desarrollado simulaciones para analizar el comportamiento del dispositivo y del sistema completo con gran cantidad de dispositivos.

4.2 Análisis del funcionamiento del prototipo del Sistema Integrado Ambulatorio de Presión Arterial

El dispositivo de adquisición de datos del sistema procesa dos tipos de variables provenientes de dos sensores: el primero es un sensor de presión arterial y el segundo es un sensor de electrocardiograma; para cumplir esto, el dispositivo está dispuesto según la Fig. 42:

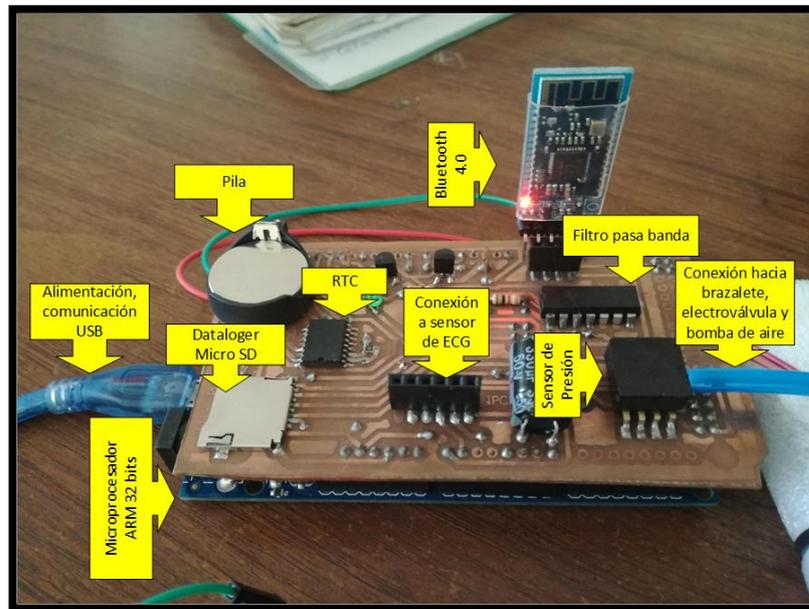


Fig. 42. Dispositivo electrónico para adquisición de presión arterial y ECG.

Fuente: Investigador.

En la Tabla 10 se detallan los componentes utilizados en el prototipo electrónico, con sus marcas y características:

Tabla 10. Descripción de los componentes utilizados en el dispositivo electrónico.

| COMPONENTE | DESCRIPCIÓN |
|-------------------------|-------------------------------|
| Brazalete | Welch Allyn |
| Electroválvula | Solenoides 3 V 75 mA |
| Electrobomba | Microbomba de vacío 3V 160 mA |
| Sensor Presión Arterial | MPX V5050CP |
| Bluetooth 4.0 | BLE HM-10 |

| | |
|--------------------------------|-----------------------|
| Sensor y acondicionador ECG | ADB 232 Heart Monitor |
| RTC | DS323/SN |
| Dataloger | Micro SD 8 GB |
| Pila | CR2032 3 V |
| Microprocesador | Arduino Due 32 bits |

Fuente: Investigador.

El dispositivo se encuentra en fase de pruebas, por lo cual se ha desarrollado sobre una plataforma Arduino Due. Está planificado reducir el tamaño del prototipo electrónico para contar con un sistema embebido con el mínimo tamaño y peso, de esta forma el dispositivo ambulatorio brindará portabilidad al usuario.

El microprocesador activa la adquisición de datos, el encendido del brazalete inflable Welch Allyn similar al de la Fig. 43 para tomar la presión arterial, convierte los datos análogos a digitales de presión arterial y electrocardiograma, y también se encarga de que toda esa información sea verídica y fiable.

La inyección de aire hacia el brazalete está controlado por el microprocesador que acciona la electroválvula y la electrobomba. La medición de la presión arterial se realiza cada 30 minutos, en ese instante el brazalete se llena de aire proveniente de una tubería plástica, presionando el brazo del paciente por un tiempo razonable para detectar la presión arterial sin distorsión. Posterior a esto el aire escapa por una tubería diferente hacia el sensor de presión arterial, este sensor detecta el cambio de presión de aire que sale de la tubería de retorno, esas conexiones se denotan en la Fig. 44; ese cambio de presión se traduce a una señal analógica variable en el tiempo la cual debe ser manejada por el microprocesador y transformada a dato digital. Además, es necesario que la señal pase por un filtro pasabanda para poder ser interpretada por el procesador.



Fig. 43. Brazaete Welch Allyn para adultos [66].

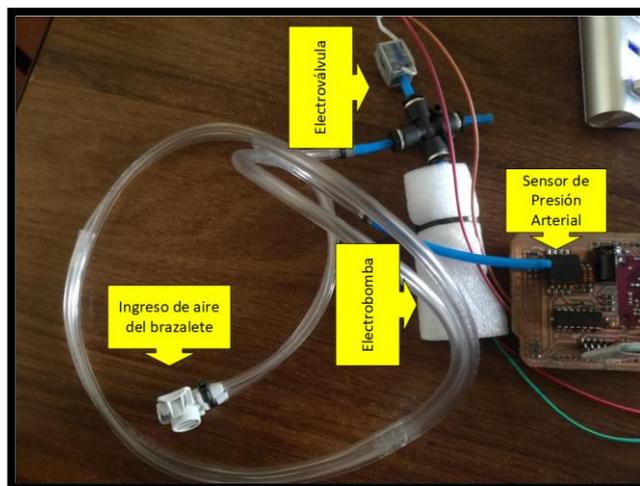


Fig. 44. Conexiones exteriores del prototipo hacia el brazaete.

Fuente: Investigador.

Como se ilustra en la Fig. 45, la toma de datos de electrocardiograma se realiza por medio de tres electrodos ubicados en diferentes partes del cuerpo del paciente. Los electrodos detectan pulsos eléctricos que serán transmitidos de manera alámbrica hacia el acondicionador de señal de ECG (electrocardiograma) y posteriormente al microprocesador. Este proceso es poco complejo en relación al de adquisición de presión arterial. Estos electrodos realizan una función semejante al de un electrocardiógrafo.

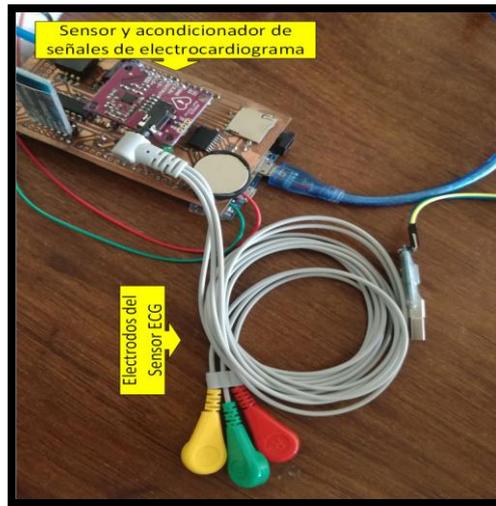


Fig. 45. Sistema de adquisición de señales de electrocardiograma.
Fuente: Investigador.

El diagrama de conexiones o esquemático del prototipo electrónico de adquisición de datos de presión arterial y electrocardiograma se describe en la Fig. 46:

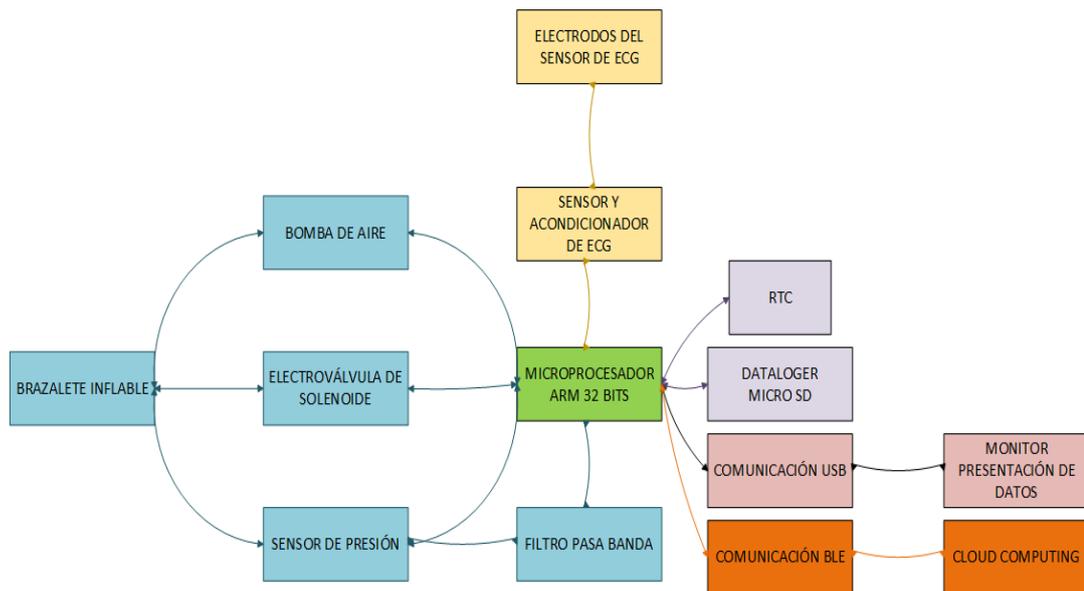


Fig. 46. Diagrama esquemático del prototipo.
Fuente: Investigadores desarrolladores del sistema.

El funcionamiento del dispositivo sigue un flujo o algoritmo que permitirá la adquisición, procesamiento, el almacenamiento y la transmisión de datos. En la adquisición y conversión ADC se incluyen las variables relacionadas tanto a la presión arterial como a las señales de electrocardiograma. Una vez realizada la adquisición de datos, la información es almacenada en el Dataloger que es una memoria SD instalada

en el sistema electrónico embebido y transmitida por el puerto de comunicaciones UART hacia módulo BLE HM-10. Este a su vez hace el trabajo de interfaz inalámbrica bluetooth para comunicar el dispositivo móvil del paciente con el dispositivo electrónico de adquisición de datos.

El siguiente esquema muestra de manera detallada el algoritmo utilizado partiendo desde el encendido del dispositivo hasta que los datos de presión arterial media, presión sistólica, presión diastólica y frecuencia cardiaca son almacenados en la SD y transmitidos por los puertos seriales hacia un PC (monitor) y hacia el módulo BLE:

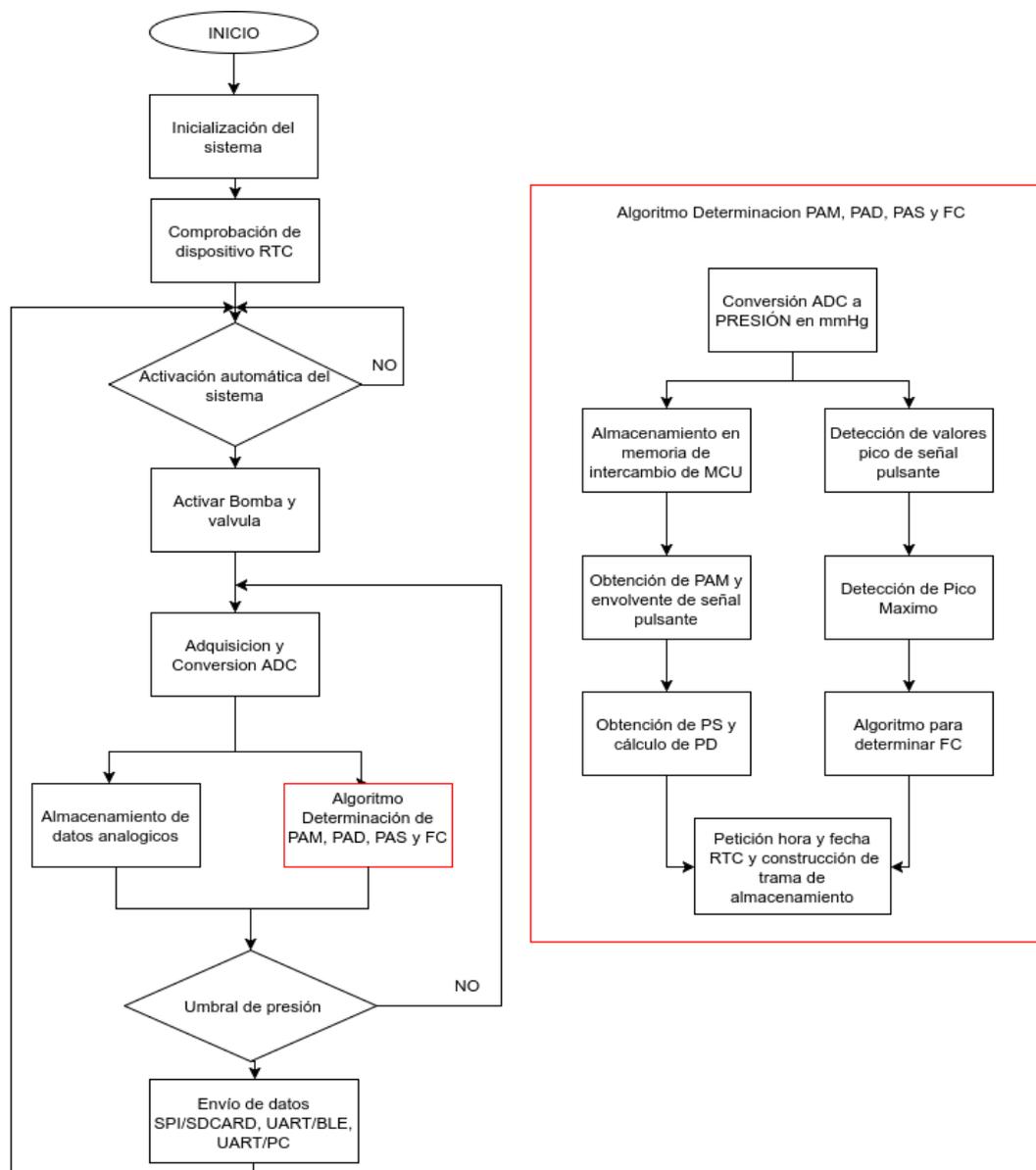


Fig. 47. Diagrama de flujo del funcionamiento del prototipo.

Fuente: Investigadores desarrolladores del sistema.

Es importante saber el tamaño del dato que se va a enviar, en este caso el dato generado por la aplicación contiene información de la presión arterial y de las señales del electrocardiograma. La Tabla 11 muestra como resultados de la información recolectada, que cada dato tiene el siguiente contenido: 2017/6/15 13:50:04 2,0878 1,0879 120 80 95 60. Con el código ASCII esta cadena de valores a ser enviados se traducen a una cantidad aproximada de 125 Bytes.

Tabla 11. Formato del dato a enviar.

| Ejemplo | | | | | |
|--------------------------------|----------------------|-------------------|--------------------|---------------|---------|
| Año | Mes | Día | Hora | Minuto | Segundo |
| 2017 | 6 | 15 | 13 | 50 | 4 |
| Valor presión arterial voltios | Valor ECG en voltios | Presión sistólica | Presión diastólica | Presión media | ECG |
| 2,0878 | 1,0879 | 120 | 80 | 95 | 60 |

Fuente: Investigador.

El dispositivo electrónico de adquisición de datos utiliza dos baterías Ultrafire 18650 recargables de 2000 mAh y 3,7 V como las de la Fig. 48, en conjunto se cuenta con 4000 mAh y 7.4 V, con este nivel de voltaje y energía se asegura un correcto funcionamiento del equipo.



Fig. 48. Batería Ultrafire 18650 2000 mAh 3,7 V [67].

El consumo de radio se produce únicamente cuando se transmite información hacia el MÓVIL, es decir cada 30 minutos durante el día. En la Fig. 49 se muestra el dispositivo BLE utilizado en el DPA, el HM-10.



Fig. 49. Módulo BLE HM-10 [68].

Este módulo tiene dos modos de funcionamiento, el primero es el modo SLEEP en este modo se consume entre 0.4 y 1.5 mA, el otro modo de funcionamiento es el ACTIVE que consume 8.5 mA al momento de transmitir o recibir información, este es el consumo que se genera al alimentar el módulo HM-10 con 3.3 VDC. OMNeT++ en sus diversos modelos de energía toma en cuenta otros modos de funcionamiento con sus respectivos valores de consumo energético. En la Tabla 12 se muestran los parámetros utilizados en la simulación los cuales son característicos de BLE:

Tabla 12. Detalles de parámetros de consumo energético de BLE [50, 68].

| Estado | Descripción | Consumo (mA) | Consumo (mW) |
|--------------|---|--------------|----------------|
| OFF | Dispositivo y HM-10 apagado | 0 mA | 0 mW |
| SLEEP | Dormido o estado pasivo en el que no se transmite ni recibe información | 0.4 - 1.5 mA | 1.32 - 4.95 mW |
| SWITCHING | Cambio de estado entre SLEEP y ACTIVE | 0.4 - 1.5 mA | 1.32 - 4.95 mW |
| IDLE TX | Estado en el que no hay transmisión en progreso y se mantiene la conexión | 0.4 - 1.5 mA | 1.32 - 4.95 mW |
| IDLE RX | Estado en el que no hay recepción en progreso y se mantiene la conexión | 0.4 - 1.5 mA | 1.32 - 4.95 mW |
| BUSY RX | Estado en el que la señal recibida no puede ser interpretada | 8.5 mA | 28.05 mW |
| TRANSMITTING | Transmisión en progreso | 8.5 mA | 28.05 mW |
| RECEIVING | Recepción en progreso | 8.5 mA | 28.05 mW |

Además, se incluye el consumo del funcionamiento en general, cuando este está apagado, encendido o adquiriendo la información de los sensores y transmitiendo la información. Recordar que el consumo energético se eleva considerablemente por el uso de la electroválvula y la electrobomba. Con la ayuda de un medidor de corriente y

voltaje conectado a un puerto USB del dispositivo se obtuvo su consumo en general como se indica en la Fig. 50.



Fig. 50. Medición de consumo energético con Charger Doctor.

Fuente: Investigador.

Dependiendo del estado de funcionamiento del dispositivo se consumen diferentes cantidades de corriente como se muestra en la Tabla 13:

Tabla 13. Detalles de parámetros de consumo energético del DPA [69].

| Estado | Descripción | Consumo (mA) | Consumo (mW) |
|---------------|--|--------------|--------------|
| APAGADO | Dispositivo apagado | 0 mA | 0 mW |
| REPOSO/SLEEP | Dispositivo encendido en reposo | 75 mA | 555 mW |
| ENCENDIDO | Dispositivo encendido en estado pasivo adquiriendo datos de ECG y sin encender válvulas | 190 mA | 1406 mW |
| TRANSMITIENDO | Dispositivo encendido en estado activo adquiriendo datos y transmitiendo, válvulas y sensores funcionando. | 520 mA | 3848 mW |

4.3 Tecnologías de comunicación del Sistema Integrado Ambulatorio de Presión Arterial

Una vez realizada la conexión entre el dispositivo de adquisición de datos y el Smartphone o Tablet del usuario, se envía la información de presión arterial y frecuencia cardiaca. En el dispositivo móvil del usuario por medio de una aplicación inteligente estos datos son mostrados asemejándose a un monitor médico, donde el usuario observa el estado de su presión arterial y la frecuencia cardiaca.

La información mostrada en el dispositivo móvil debe ser conocida por el doctor de cabecera que remotamente controla la actividad del paciente, por lo tanto es necesario un tipo de tecnología que permita a estos datos ser transmitidos desde cualquier lugar y en el momento planificado.

En este contexto se presentan dos escenarios en los cuales el usuario hace uso de diferentes tecnologías para acceder a internet y enviar a través del dispositivo móvil del usuario (Smartphone o Tablet) sus datos y encaminarlos hacia la red del servidor médico.

- **Escenario 1:** En el escenario de la Fig. 51 el usuario tiene acceso a una conexión inalámbrica Wi-Fi, sea doméstica o parte de una red inalámbrica de acceso público como la que existen en parques y calles de la ciudad.

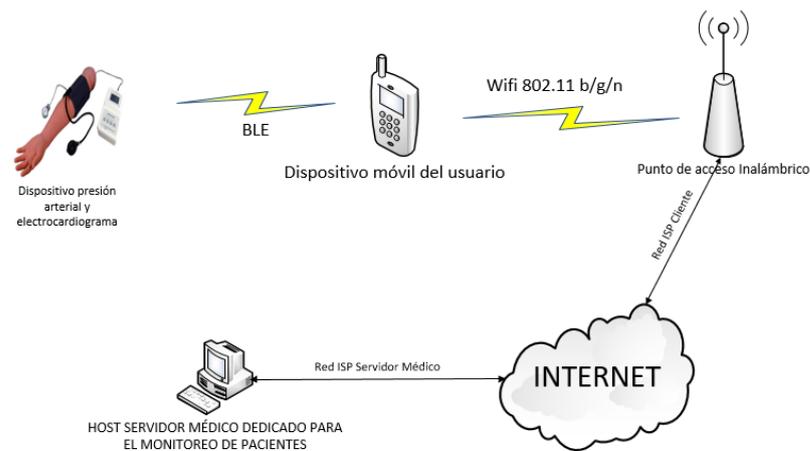


Fig. 51. Escenario 1: BLE y Wi-Fi.

Fuente: Investigador.

- **Escenario 2:** En el escenario de la Fig. 52 el usuario tiene una conexión celular con acceso a internet vía datos móviles 4G.

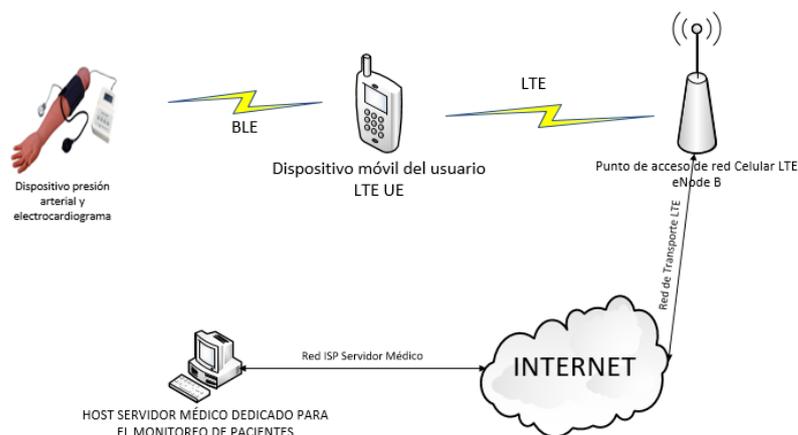


Fig. 52. Escenario 2: BLE y red celular LTE 4G.

Fuente: Investigador.

Cada tecnología utilizada en los dos escenarios esta descrita en la Tabla 14, estas tres tecnologías serán utilizadas en los escenarios de simulación para el análisis de resultados:

Tabla 14. Tecnologías utilizadas en la simulación [8, 9, 10, 70, 71, 72, 73].

| Parámetro \ Tecnología | Bluetooth 4.0 | Wi-Fi | LTE 4G |
|---------------------------------|-----------------------------|--|---|
| Estándar | 802.15.1 | 802.11 a/b/g/n | 3GPP |
| Frecuencia | 2,4 GHz | 2,4 y 5 GHz | Banda 2 * (1850 - 1910 MHz) Subida (1930 - 1990 MHz) Bajada |
| Tasa Máxima de Bits | 721 Kbps – 1 Mbps | 11 Mbps - 300 Mbps | 1 Gbps |
| Tipo de datos | Audio, imagen, datos | Audio, imagen, datos, video, streaming | Audio, imagen, datos, video, streaming |
| Rango Máximo Cobertura | 50 m | 100 m | 5 Km |
| Vida de la Batería | 1 semana | - | - |
| Tamaño de red | 8 nodos | 15 usuarios simultáneos | 200 usuarios por celda |
| Potencia típica de transmisión | 1 - 100 mW | < 20 dBm | < 47 dBm |
| Ancho de canal | 2 MHz | 20, 40 MHz | 1.4/3/5/10/15/20 MHz |
| Modulación | GFSK | OFDM (QPSK, QAM y otros) | OFDMA (QPSK, QAM y otros) |
| Sensibilidad del receptor | -87 a -93 dBm | -70 a -87 dBm | -87 dBm |
| Número de canales | 40 canales | 13 canales | 12 subportadoras de 15 KHz |
| Difusión del espectro | FHSS | DSSS | TDD, FDD |
| Método de acceso al medio (MAC) | TDMA | CSMA/CA | OFDMA |
| QoS | Si | Si | Si |
| Latencia | < 3 ms | < 50 ms | < 250 ms |
| Topología de red | Estrella, Scatternet | Ad-Hoc, Malla, Infraestructura | Estrella |
| Consumo de energía | Bajo | Alto | Bajo |
| Seguridad | Autenticación, Encriptación | Autenticación, Encriptación | Autenticación, Encriptación |
| Aplicación | MR-WPAN | WLAN | Redes móviles 4G |

Los parámetros de las tecnologías Bluetooth 4.0, Wi-Fi y LTE 4G mostrados en la Tabla 14 son los que se incluyeron en los archivos de simulación de cada uno de los escenarios planteados.

4.4 Pruebas en pacientes y resultados

El prototipo ha sido evaluado en estudiantes de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, los resultados han sido comparados con mediciones realizadas con un dispositivo validado para la medición de presión arterial Well Allyn. En la Fig. 53 se visualiza como se coloca el dispositivo para realizar la adquisición de datos:

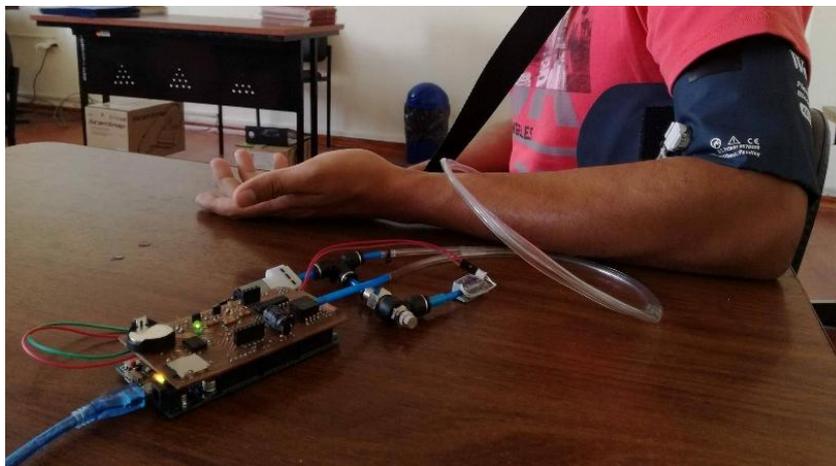


Fig. 53. Pruebas de dispositivo electrónico de adquisición de datos.

Fuente: Investigador.

Para las mediciones de presión arterial y frecuencia cardiaca, se tomó en cuenta el peso, la estatura y si los pacientes ingieren alcohol, fuman tabaco o hacen uso de drogas y estupefacientes. Obviamente el consumo de estos productos genera una alteración de la presión arterial, haciéndola irregular. Por otro lado la actividad física es muy importante para que la presión arterial se regule, entonces es importante saber si el paciente lleva una vida sedentaria o no. Estos datos llevan a un estudio médico más profundo, pero se utilizan como estadística en este proyecto como parte del estudio de campo.

La Tabla 15 muestra los resultados de las pruebas realizadas a 7 pacientes. A cada paciente se le realizaron cuatro mediciones: presión sistólica (PS), presión diastólica (PD), presión arterial media (PAM) y frecuencia cardiaca (FC).

Tabla 15. Resultados de las pruebas en pacientes.

| N.º | PACIENTE | EDAD | GENERO | FUMA | ALCOHOL | DROGAS | SEDENTARIO | PESO (Kg) | TALLA (m) | DATO | TIPO |
|-----|--------------------|------|-----------|------|---------|--------|------------|-----------|-----------|--------|------|
| 1 | ANDREA ALARCÓN | 23 | FEMENINO | No | Si | No | Si | 50 | 1,5 | 0 | PS |
| | | | | | | | | | | 120,06 | PD |
| | | | | | | | | | | 80,04 | PAM |
| | | | | | | | | | | 56 | FC |
| 2 | GISSELA BONILLA | 22 | FEMENINO | No | No | No | Si | 50 | 1,5 | 0 | PS |
| | | | | | | | | | | 102,77 | PD |
| | | | | | | | | | | 68,51 | PAM |
| | | | | | | | | | | 0 | FC |
| 3 | VALERY PICO | 22 | FEMENINO | Si | Si | No | Si | 55 | 1,65 | 116,53 | PS |
| | | | | | | | | | | 94,2 | PD |
| | | | | | | | | | | 101,65 | PAM |
| | | | | | | | | | | 110 | FC |
| 4 | SUSANA GALLEGOS | 25 | FEMENINO | No | No | No | Si | 50 | 1,53 | 102,88 | PS |
| | | | | | | | | | | 63,59 | PD |
| | | | | | | | | | | 76,69 | PAM |
| | | | | | | | | | | 34 | FC |
| 5 | FERNANDO MAYANQUER | 24 | MASCULINO | Si | Si | No | Si | 60 | 1,85 | 0 | PS |
| | | | | | | | | | | 128,46 | PD |
| | | | | | | | | | | 85,64 | PAM |
| | | | | | | | | | | 44 | FC |
| 6 | DIEGO BENITEZ | 25 | MASCULINO | Si | Si | No | No | 60 | 1,69 | 117,65 | PS |
| | | | | | | | | | | 36,56 | PD |
| | | | | | | | | | | 63,59 | PAM |
| | | | | | | | | | | 56 | FC |
| 7 | BRYAN ARAUJO | 24 | MASCULINO | No | No | No | No | 68 | 1,62 | 103,33 | PS |
| | | | | | | | | | | 60,34 | PD |
| | | | | | | | | | | 74,67 | PAM |
| | | | | | | | | | | 0 | FC |

Fuente: Investigador.

Los datos de presión arterial se muestran en mmHg y los de frecuencia cardiaca en pulsaciones por minuto, todos estos valores fueron tomados de los dos sensores con los que consta el dispositivo electrónico e inicialmente son tomados en voltios, en la Tabla 16 se muestran algunos de los valores:

Tabla 16. Mediciones de presión arterial y frecuencia cardiaca el voltios.

| SEÑAL DE PRESIÓN ARTERIAL EN VOLTS | SEÑAL DE FRECUENCIA CARDIACA EN VOLTS |
|---|--|
| 0,278 | 2,1691 |
| 0,3049 | 2,1866 |
| 0,4016 | 2,1812 |
| 0,2861 | 2,1503 |
| 0,2874 | 2,1772 |
| 0,2955 | 2,1678 |
| 0,3103 | 2,1637 |
| 0,2847 | 2,1342 |
| 0,2901 | 2,1731 |
| 0,2915 | 2,1651 |
| 0,3076 | 2,1919 |
| 0,3358 | 2,1302 |

Fuente: Investigador.

Estos niveles bajos de voltaje atraviesan el algoritmo de conversión análogo a digital para ser almacenados en la memoria SD instalada y ser enviados a los módulos de comunicación.

La Tabla 17 indica la diferencia entre resultados tomados por un dispositivo validado y el dispositivo analizado en este proyecto:

Tabla 17. Comparación de resultados entre dispositivo validado y dispositivo en desarrollo.

| Tipo | Valor tomado por dispositivo validado | Valor tomado por prototipo en desarrollo | Error | Porcentaje |
|-------------|--|---|--------------|-------------------|
| PAM | 80 | 84,52 | 4,52 | 5,7% |
| PAD | 64 | 68,99 | 4,99 | 7,8% |
| PAS | 111 | 107,07 | -3,93 | -3,5% |
| FC | 44 | 48 | 4 | 9,1% |

Fuente: Investigador.

En este ejemplo los resultados reflejan que el error de medición de presión arterial va entre 3,93 y 4,99 mmHg, según estudios un valor erróneo de más de 10 mmHg es inaceptable. En pruebas realizadas se denota que el valor de diferencia entre las

mediciones de un dispositivo ya validado y el que se está desarrollando no pasa de los 10 mmHg lo cual clínicamente es aceptable [74].

En el caso de la frecuencia cardíaca un límite de pulsos erróneos en la medición es de hasta 8 pulsos por minuto, en las mediciones realizadas la mayoría de mediciones fueron exactas, en el ejemplo mostrado existe un error de 4 PPM que es un error aceptable para determinar el estado del paciente [75].

El sistema tiene la capacidad de la adquisición de datos por medio del dispositivo electrónico, mismo que los envía a una aplicación por medio de bluetooth y visualizar los datos en el móvil. En este contexto, se han tomado mediciones de los tiempos de conexión entre ambos dispositivos. Los resultados mostrados en la Tabla 18 indican que el tiempo que se tarda un mensaje en llegar de extremo a extremo en la conexión BLE es:

Tabla 18. Mediciones de retardos de extremo a extremo enlace BLE.

| | |
|------------------------------|------|
| Retardo mínimo (ms) | 51 |
| Retardo máximo (ms) | 1035 |
| Retardo promedio (ms) | 544 |

Fuente: Investigador.

Hasta el momento el sistema web no está desarrollado, imposibilitando la toma de datos reales como: retardos de conexión, rendimiento de red y pérdidas de información, desde que el paquete parte del dispositivo móvil (Smartphone o Tablet) hacia el servidor médico.

4.5 Análisis de plataformas de simulación

Existen varias plataformas que permiten simular redes con diferentes protocolos manejados para redes cableadas e inalámbricas, entre las plataformas con mejores características se encuentran las detalladas en la Tabla 19:

Tabla 19. Comparación de plataformas de simulación de redes [62].

| Plataforma | OMNeT++ | WDSim | mac802.11 | NS-2 | NS-3 | Máquinas virtuales |
|---------------------------|------------|-------------|-----------|-------------|-------------|--------------------|
| Tipo de fuente | Abierto | Abierto | Abierto | Abierto | Abierto | Abierto/Cerrado |
| Lenguaje de programación | C++, NED | Java | C | C++, TCL | C++, Python | - |
| Estado de desarrollo | Activo | - | Activo | Detenido | Activo | Activo |
| Calidad de resultados | Alto | - | Alto | Bajo | Alto | Alto |
| Uso de CPU | Alto | Alto | Bajo | Alto | Alto | Alto |
| Uso de memoria | Bajo | Alto | Bajo | Alto | Bajo | Alto |
| Soporte de movilidad | Si | Parcial | No | Si | Si | Parcial |
| Protocolos disponibles | Muchos | Pocos | Pocos | Muchos | Muchos | Pocos |
| Comunidad | Muy activa | Poco activa | Activa | Poco activa | Muy activa | Muy activa |
| Cantidad de publicaciones | Muchas | Pocas | Pocas | Muchas | Muchas | Pocas |
| Modelos de energía | Si | No | No | Si | Si | No |
| Ejecución paralela | Si | No | No | Si | Si | No |

La tabla claramente indica que existen dos plataformas OMNeT++ y NS-3 con las mejores características. OMNeT++ ha sido elegido en este trabajo ya que se ajusta a los requerimientos planteados en la Tabla 20 y puede ser instalado en Windows o Linux, además su interfaz presta muchas facilidades para comodidad del usuario.

Tabla 20. Requerimientos de la simulación vs Prestaciones de OMNeT++.

| Requerimiento | OMNeT++ |
|--|---------|
| Disponibilidad protocolo BLE | No |
| Disponibilidad protocolos Wi-Fi | Si |
| Disponibilidad redes móviles | Si |
| Disponibilidad protocolos LTE | Si |
| Parametrización | Si |
| Protocolos editables | Si |
| Gráficas de resultados | Si |
| Interfaz gráfica amigable para usuario | Si |
| Capacidad de simular cientos de nodos | Si |
| Análisis de parámetros de radio | Si |

Fuente: Investigador.

OMNeT++ y el framework INET soportan protocolos como el 802.11, 3GPP (LTE con SimuLTE) y 802.15.4, el único requerimiento que no cumple OMNeT++ es la disponibilidad del protocolo 802.15.1 de bluetooth que se utiliza para BLE.

El protocolo 802.15.4 en INET está disponible para simular redes de sensores inalámbricos con UWB (*Ultra wide band*) y Zigbee, en base a este protocolo disponible en la plataforma se desarrolló el protocolo 802.15.1 para BLE y así incluirlo en la simulación.

La simulación del sistema de comunicación inalámbrico integrado ambulatorio de presión arterial se ha desarrollado en OMNET++, y es una herramienta técnica para verificar la factibilidad de la implementación de dicho sistema de manera masiva.

4.6 Configuraciones de OMNeT++ y del Framework INET

Instalación de OMNeT++ 5

OMNeT++ puede ser descargado como un archivo .zip desde <http://omnet.org>. El paquete incluye los archivos fuente de OMNeT++, el compilador C++ y un entorno de compilación de línea de comandos.

Se debe copiar y extraer el archivo en un directorio que no contenga espacios en su nombre, por ejemplo, no se lo debe instalar en “Archivos de programa”, una recomendación es ubicarlo sobre la raíz principal del disco C.

Una vez descomprimido el archivo, iniciar el archivo *mingwenv.cmd*, a continuación, se abrirá una ventana de comandos donde se deben ingresar los siguientes comandos:

```
$ ./configure
```

```
$ make
```

Para verificar la instalación es posible ejecutar un ejemplo como *aloha* dentro de la carpeta *samples*.

```
$ cd samples/aloha
```

```
$ ./aloha
```

Si no existen problemas, por defecto se ejecutará una interfaz de simulación con el ejemplo *aloha*. Para ejecutar OMNeT++ se escribe `$ omnetpp` en la terminal. Para obtener un acceso directo a la aplicación y no tener que usar la línea de comandos del archivo *mingwenv.cmd*, dirigirse al directorio *omnetpp-5.0\ide* y dar clic derecho en el archivo *omnetpp.exe* y enviar a escritorio (crear acceso directo) [76].

Instalación del framework INET

El framework está disponible como archivo comprimido en la página web <http://inet.omnetpp.org>. Luego de extraerlo se debe colocar en el directorio de trabajo a ser utilizado (no necesariamente dentro de la carpeta de código fuente de OMNeT++). A continuación, en el Guide de OMNeT++ según la Fig. 54 se importa el proyecto usando: *File / Import / General / Existing projects into Workspace*.

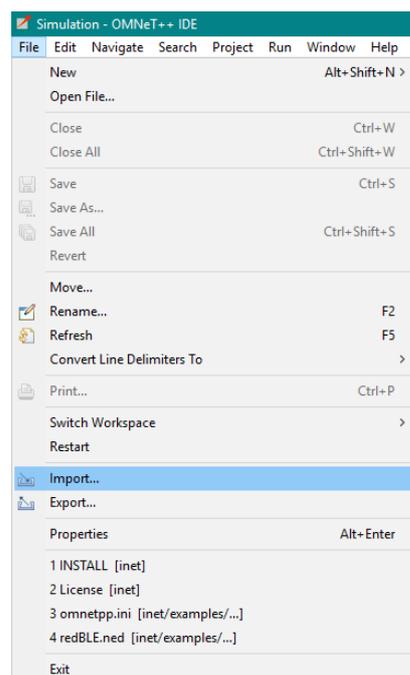


Fig. 54. Importación de proyecto en OMNeT++.

Fuente: Investigador.

Si el archivo no se ha copiado aun en la carpeta de trabajo, se debe marcar la opción “*Copy projects into workspace*” como se indica en la Fig. 55, en caso de que el archivo ya se encuentra en la carpeta de trabajo no es necesario marcar esta opción.

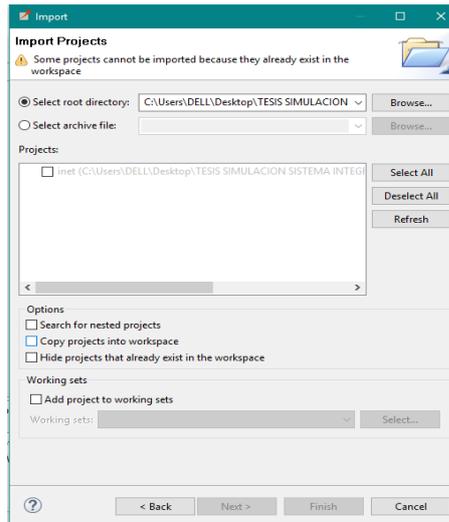


Fig. 55. Importación de INET framework en OMNeT++.

Fuente: Investigador.

Inclusión de protocolo 802.15.1 para BLE

El protocolo 802.15.1 no está disponible en INET, pero este framework cuenta con el protocolo 802.15.4 desarrollado para trabajar con redes de sensores con tecnología Zigbee y UWB. Para desarrollar BLE se modificó los parámetros del protocolo 802.15.4 en capa de enlace (MAC) y capa física (radio). Estos cambios se realizaron en el directorio fuente de INET: `inet/src/inet/physicallayer/ieee802151` e `inet/src/inet/linklayer/ieee802151` como se muestra en la Fig. 56:

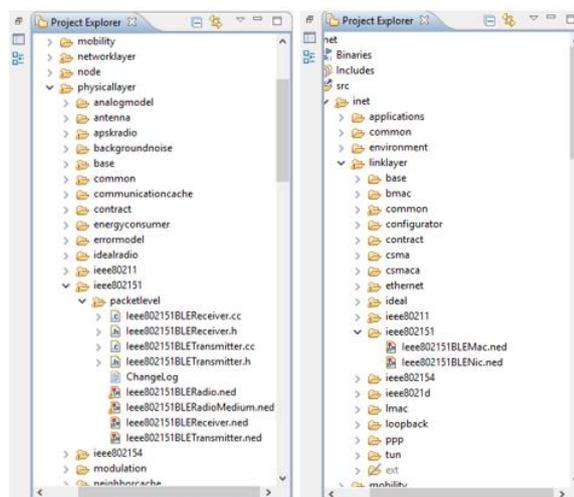


Fig. 56. Construcción del protocolo 802.15.1 para BLE en el framework INET.

Fuente: Investigador.

Inclusión de LTE

El framework INET es la base del protocolo desarrollado por los desarrolladores de LTE en OMNeT++. LTE ha sido implementado en un framework paralelo a INET y se denomina SIMULte, el cual manipula una gran cantidad de módulos simples y compuestos de INET para diseñar redes de telefonía móvil.

De manera tradicional SIMULte se incluye en el mismo directorio de trabajo de INET, teniendo que compilarlo para empezar a trabajar y simular redes LTE. Por motivos de falla en la compilación del código fuente, se realizó la inclusión de las librerías de LTE en la carpeta del código fuente de INET.

SIMULte es descargado desde su sitio oficial *simulte.com*, una vez obtenido el archivo es necesario descomprimir y copiar las carpetas del código fuente en la dirección de INET que se muestra en la Fig. 57: *inet/src/lte*.

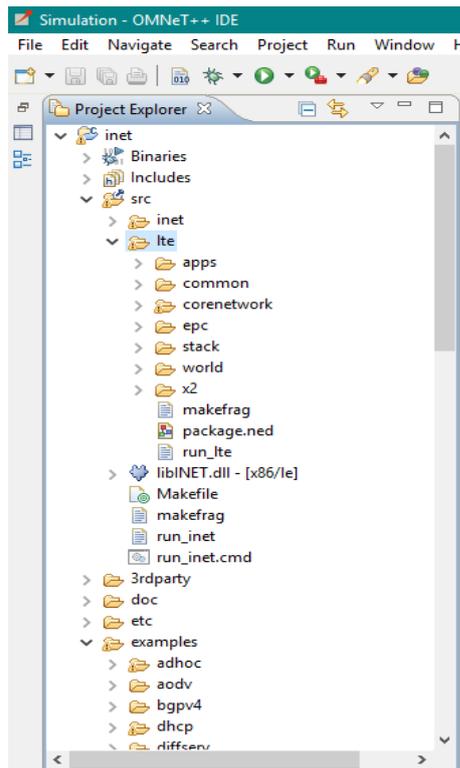


Fig. 57. Inclusión de LTE en el código fuente del framework INET.

Fuente: Investigador.

4.7 Simulación del Sistema Integrado Ambulatorio de Presión Arterial en OMNeT++

Para realizar un estudio completo de los parámetros y resultados arrojados por la simulación, cada una de las tecnologías fue analizada por separado y en otros casos de forma conjunta. Las simulaciones muestran diferentes topologías que estudian BLE, LTE y Wi-Fi para sistemas de un solo usuario y de múltiples usuarios. Estas topologías se establecieron según los dos escenarios planteados, esto es desglosado en la Fig. 58:

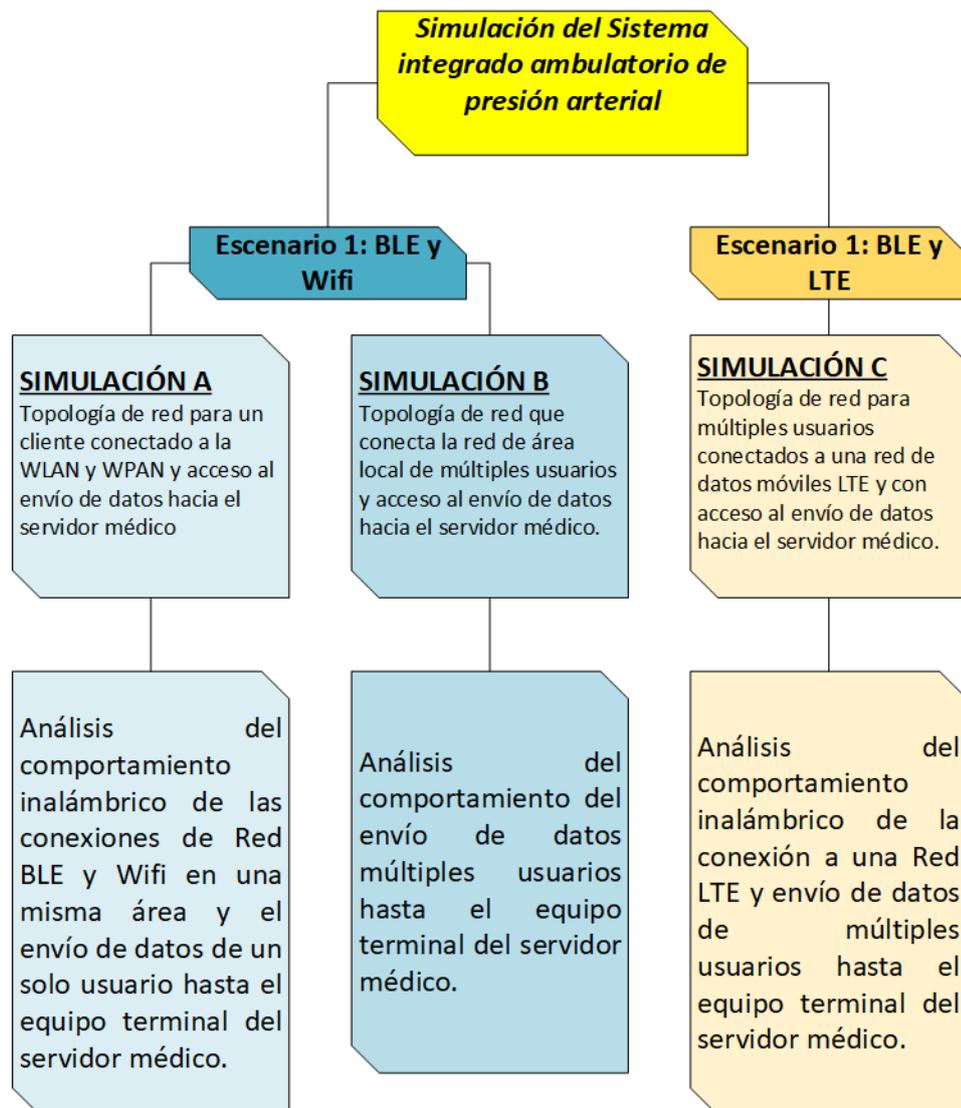


Fig. 58. Descripción de simulaciones realizadas.

Fuente: Investigador.

Los archivos: “.ned”, “.ini”, “.xml” y la carpeta de resultados están ubicados dentro del directorio: *inet/examples/sistema_pa_ecg/*, dentro de las carpetas escenario1 y

escenario2. La Fig. 59 indica la interfaz de OMNeT++ donde al dar clic derecho sobre las carpetas, se crean múltiples tipos de archivos y carpetas.

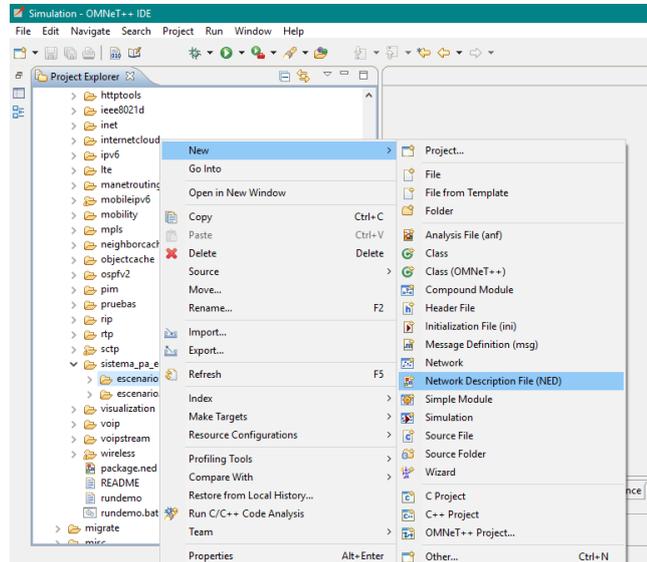


Fig. 59. Menú desplegable para creación de archivos. “.ned”, “.ini”, “.xml”, etc.

Fuente: Investigador.

En cada escenario se han incluido los archivos de descripción de red “.ned”, de inicialización de simulación y “.xml” indicados en la Fig. 60:

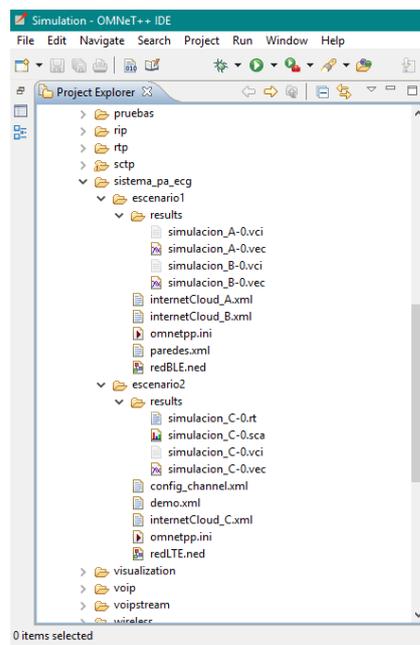


Fig. 60. Organización de los archivos de la simulación.

Fuente: Investigador.

Como muestra la Fig. 60, el escenario1 está compuesto por los siguientes archivos:

- ***Carpeta “results”***: es una carpeta que se genera automáticamente luego de ejecutar la simulación. Dentro de ella se guardan archivos que contienen información de los resultados obtenidos de la simulación unos se guardan como escalares (simulación_A-0.sca, simulación_B-0.sca) y otros como vectores (simulación_A-0.vec, simulación_B-0.vec).
- ***interneCloud_A.xml***: en este archivo se especifica cómo se realiza el direccionamiento en la nube para interconectar la red del usuario y la red del servidor médico de la simulación A, se lo llama desde el archivo *omnetpp.ini*.
- ***interneCloud_B.xml***: en este archivo se especifica cómo se realiza el direccionamiento en la nube para interconectar la red de múltiples usuarios y la red del servidor médico de la simulación B, se lo llama desde el archivo *omnetpp.ini*.
- ***omnetpp.ini***: es el archivo de inicialización donde se incluyen las características y parámetros del funcionamiento de la simulación, como las configuraciones de radio, capa física, movilidad, ambiente físico, modelo de propagación. Para el escenario1 este archivo tiene dos configuraciones diferentes: [*Config simulacion_A*] desarrollada para la configuración de la Simulación A y [*Config simulacion_B*] para la Simulación B.
- ***paredes.xml***: en este archivo se detalla las configuraciones de ambiente físico y permite incluir obstáculos como paredes, vidrios, madera, etc., se lo llama desde el archivo *omnetpp.ini*.
- ***redBLE.ned***: es el archivo de descripción de red, en el cual se detalla la topología de red y se incluyen las librerías y todos los módulos de configuración de red, capa física, visualizadores, etc. Este archivo define dos topologías: *network simulacion_A* y *network simulacion_B*, pertenecientes a lo descrito en la Fig. 55 del escenario1. Los parámetros de estas topologías se los detalla en su respectiva configuración del archivo *omnetpp.ini*.

Para el escenario2 los archivos son los siguientes:

- ***Carpeta “results”***: es una carpeta que se genera automáticamente luego de correr la simulación. Dentro de ella se guardan archivos que contienen información de los resultados obtenidos de la simulación unos se guardan como

escalares (simulación_C-0.vci, simulación_C-0.sca, simulación_C-0.rt) y otros como vectores (simulación_C-0.vec).

- ***config_channel.xml***: en este archivo se especifican las configuraciones de canal inalámbrico en la red LTE, se lo llama desde el archivo *omnetpp.ini*.
- ***demo.xml***: archivo de configuración para enrutamiento dinámico utilizado por el módulo *configurator* incluido en el archivo *redLTE.ned*, su uso es opcional.
- ***interneCloud_C.xml***: en este archivo se especifica cómo se realiza el direccionamiento en la nube para interconectar la red LTE y la red del servidor médico de la simulación C, se lo llama desde el archivo *omnetpp.ini*.
- ***omnetpp.ini***: es el archivo de inicialización donde se incluyen las características y parámetros del funcionamiento de la simulación, como las configuraciones de radio, capa física, movilidad, ambiente físico, modelo de propagación. Para el escenario2 este archivo tiene dos configuraciones diferentes: *[General]* desarrollada para configuraciones específicas y que no se deben cambiar para un correcto funcionamiento de la red LTE y *[Config simulacion_C]* en la cual se detallan las configuraciones específicas del sistema integrado ambulatorio de presión arterial.
- ***redLTE.ned***: es el archivo de descripción de red, en el cual se detalla la topología de red y se incluyen las librerías y todos los módulos de configuración de red, capa física, visualizadores, etc. Este archivo define una topología: *network simulacion_C*, según lo descrito en la Fig. 55 del escenario2. Los parámetros de esta topología se los detalla en su respectiva configuración del archivo *omnetpp.ini*.

La manera más práctica de diseñar una topología de red descrita en el archivo “.ned” es utilizando las facilidades gráficas del entorno de OMNeT++. Para generar un archivo “.ned” se da clic izquierdo sobre la carpeta de trabajo a continuación: *New / Network Description File (NED)*. Se mostrará el menú ilustrado en la Fig. 61 donde se incluye el nombre del archivo, luego se selecciona *Empty NED file (archivo NED vacío)* y finalizar.

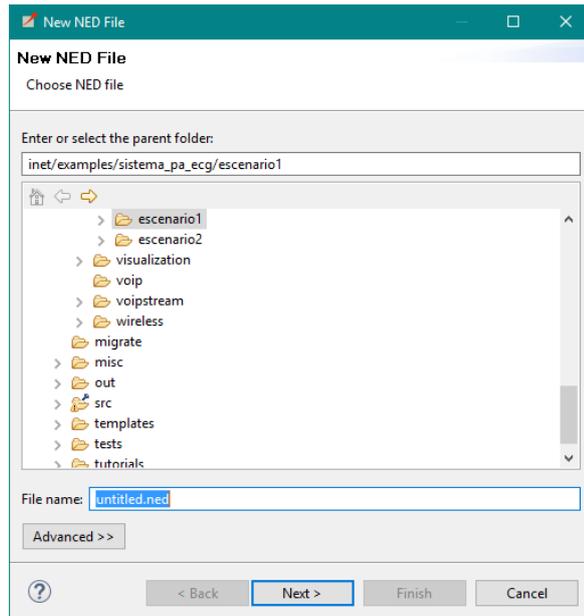


Fig. 61. Generación de archivo “.ned”.

Fuente: Investigador.

En la parte inferior al abrir el archivo “.ned”, aparecen dos pestañas *Design* y *Source*. La pestaña *Design* ilustrada en la Fig. 62 permite al usuario añadir módulos y submódulos a la topología desde una paleta de componentes que se seleccionan y se añaden de manera gráfica al área de trabajo, la pestaña *Source* mostrada en la Fig. 63 permite añadir esos componentes digitando el código completamente.

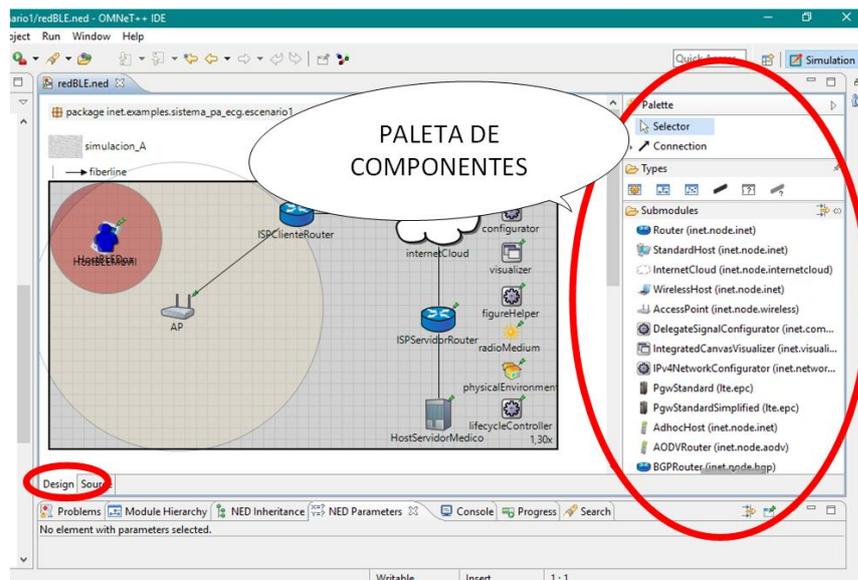


Fig. 62. Pestaña Design del archivo “.ned”.

Fuente: Investigador.

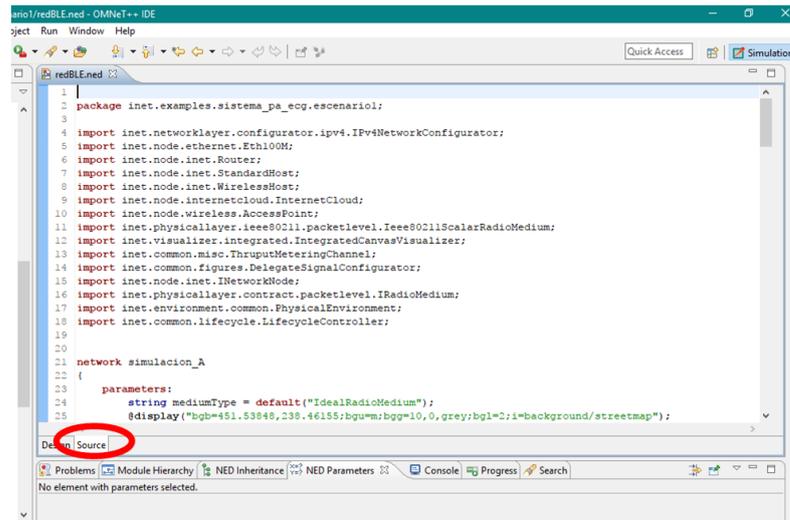


Fig. 63. Pestaña Source del archivo “.ned”.

Fuente: Investigador.

Al abrir el archivo el primer paso es establecer un área de trabajo donde se colocarán los dispositivos que intervienen en la topología, es decir una red (caja) que contenga todos esos componentes. Esto se logra al dar clic en el ícono *Network* de la paleta de diseño mostrado en la Fig. 64 y colocarlo en el área de trabajo. Una vez colocada la red, añadir los módulos pertenecientes a la topología desde la paleta, como se indica en la Fig. 62.

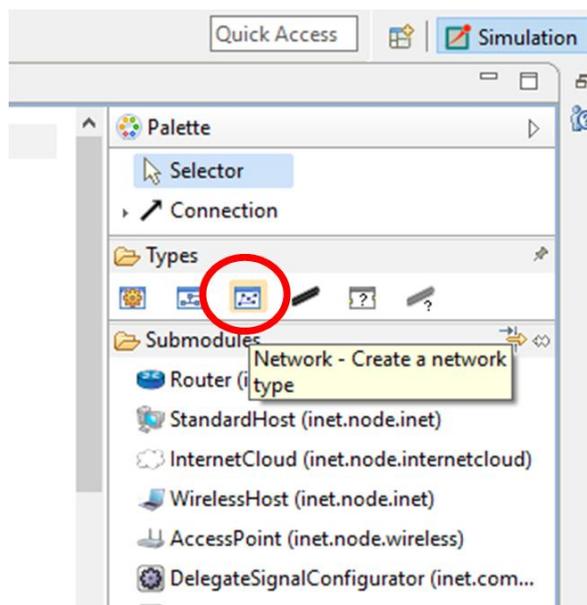


Fig. 64. Creación de nueva red en el archivo “.ned”.

Fuente: Investigador.

En el caso de un archivo “.ini” se da clic izquierdo sobre la carpeta de trabajo a continuación: *New / Initialization file (ini)*. Luego se muestra el menú donde se incluye el nombre del archivo y se selecciona *Empty Ini file (archivo INI vacío)* y finalizar. En el caso de los archivos “.xml”: *New / File*, en el menú siguiente se coloca el nombre del archivo con extensión “.xml”, por ejemplo *config_channel.xml*.

4.7.1 Escenario 1: BLE y Wi-Fi (Simulación A)

La Fig. 65 ilustra una red en la cual un usuario usa una red WPAN para intercambiar datos entre el dispositivo de adquisición de signos vitales y el Smartphone, que a su vez se conecta la red de área local Wi-Fi para enviar los datos de presión arterial y electrocardiograma.

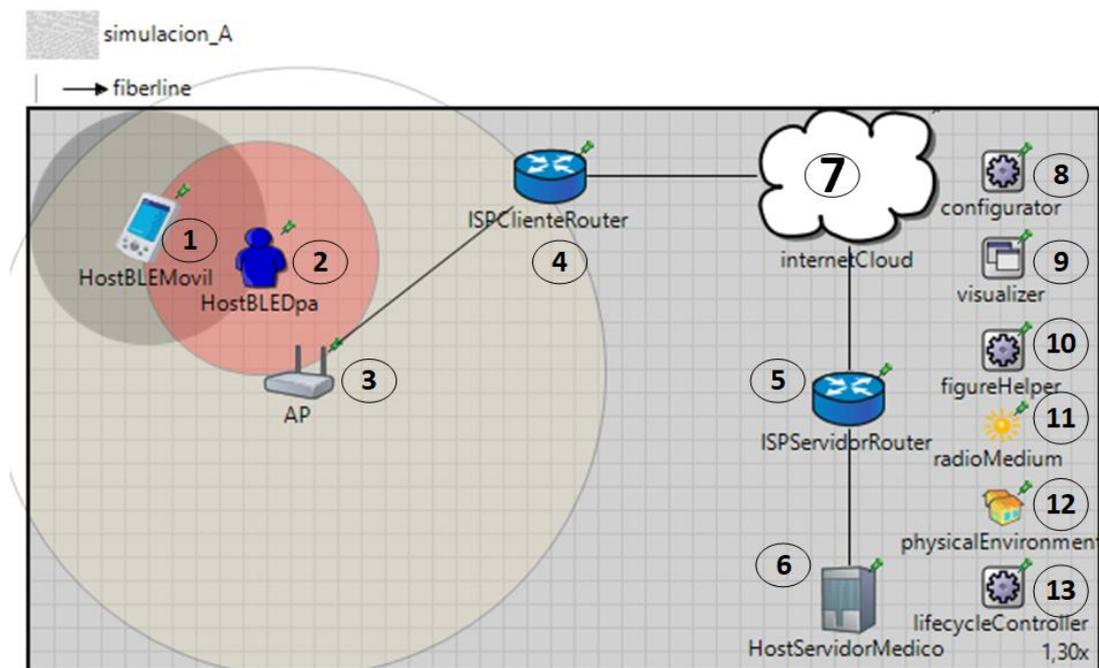


Fig. 65. Descripción de red de la Simulación A del escenario 1.

Fuente: Investigador.

El dispositivo electrónico para la adquisición de datos (HostBLEDpa 2) emplea el protocolo 802.15.1 Bluetooth Low Energy con el dispositivo móvil inteligente del usuario (HostBLEMovil 1), sea este un Smartphone o una Tablet. A su vez el móvil se enlaza con Wi-Fi 802.11 a la red inalámbrica domestica a través de un punto de acceso o un router inalámbrico (ISPClienteRouter 4) con acceso a internet (internetCloud 7),

en la simulación se utiliza un canal de fibra óptica para simular el acceso. Este canal también interconecta a la nube con el router del servidor médico (ISPServidorRouter 5), finalmente llegando con conexión Ethernet hacia el host dedicado al control de los pacientes (HostServidorMedico 6), mismo que deberá contar con una interfaz y base de datos capaz de mostrar y almacenar la situación actual de las variables fisiológicas del usuario.

Con esta topología se analiza todos los parámetros concernientes al estándar 802.15.1 BLE y al 802.11 Wi-Fi, entre los cuales están: latencia, pérdida de paquetes, relación de entrega, consumo energético, problemáticas de canal inalámbrico, etc.

En el archivo “.ned” *redBLE.ned* la red *simulación_A*, especifica 7 submódulos principales:

1. **HostBLEMovil:** dispositivo móvil del usuario sea un Smartphone o Tablet.
2. **HostBLEDpa:** dispositivo electrónico de adquisición de datos.
3. **AP:** punto de acceso inalámbrico Wi-Fi.
4. **ISPClienteRouter:** puerta de acceso del usuario hacia una red de área extensa con conexión a la nube, es decir es el simula la función del ISP del cliente.
5. **ISPServidorRouter:** puerta de acceso del servidor médico hacia una red de área extensa con conexión a internet, es decir es el simula la función del ISP del servidor.
6. **HostServidorMedico:** ordenador del médico de cabecera que utilizado para monitorear la salud del usuario.
7. **internetCloud:** submódulo que simula una conexión a internet con un ancho de banda y latencia determinada, permite la conexión de la red del ISP del usuario y la del servidor médico.

El código que se genera al colocar los módulos desde la interfaz de diseño es el siguiente:

```
HostBLEMovil: WirelessHost {
    parameters:
        @display ("p=50,50;i=device/pocketpc;r=50,black");
}
HostServidorMedico: StandardHost {
    parameters:
        @display ("p=346.15387,208.46155;i=device/server");
}
ISPClienteRouter: Router {
```

```

        parameters:
            @display("p=220.00002,28.46154");
    }
    AP: AccessPoint {
        parameters:
            @display("p=114.24671,111.97088;r=130,wheat");
    }
    HostBLEDpa: WirelessHost {
        parameters:
            @display("p=50,50;i=abstract/person;r=50,red");
    }
    ISPServidorRouter: Router {
        parameters:
            @display("p=346.15387,123.07693");
    }

    internetCloud: InternetCloud {
        @display("p=346.15387,27.692308;is=v1");
    }
}

```

En cada módulo se destacan el punto de ubicación (x, y) “p”, “r” para visualizar un radio de cobertura y el tipo de ícono “i”.

Los submódulos de configuración utilizados son los siguientes:

8. **Configurator:** configura las direcciones IP de manera dinámica o estática, tomando en cuenta subredes y optimiza las tablas de ruteo generadas en cada nodo de la red.
9. **Visualizer:** permite generar un solo módulo que contiene las diferentes visualizaciones o animaciones de la simulación.
10. **FigureHelper:** encuentra y administra todas las propiedades de los denominados *@delegatesignal*. Que almacenan datos específicos dentro de la simulación, por ejemplo, para ser visualizados en la ventana de la propia simulación.
11. **RadioMedium:** este submódulo describe el medio físico compartido en donde la comunicación se da lugar. Mantiene la comunicación de radio, agrega interferencias, ruidos externos e internos. El módulo procesa cuando, donde y como las transmisiones y ruidos llegan a los receptores.
12. **PhysicalEnvironment:** las transmisiones de radio, rango de propagación dependen de los obstáculos físicos, tipo de suelo y muchos agentes de medio y este módulo permite generar esas variables.

13. LifeCycleController: maneja los modos de operación de los nodos como: encendido, apagado, reinicio, etc. Además, su uso es importante para los diferentes submódulos que necesitan información del estado actual del dispositivo.

La conexión alámbrica entre el AP y el servidor del cliente se la estableció como conexión Ethernet, de la misma manera la conexión entre el router del servidor y el host del servidor médico. Además, tiene un canal de fibra óptica para simular la conexión del servidor de internet de cada uno de los involucrados hacia internet. Todas estas conexiones y canales extras de comunicación se introducen desde la pestaña *Source* del archivo “.ned” con las siguientes sintaxis:

types:

```
channel fiberline extends ThruputMeteringChannel
{
    delay = 1us;
    datarate = 100Mbps;
    thruputDisplayFormat = "u";
}
```

connections:

```
HostServidorMedico.ethg++ <--> Eth100M <-->ISPServidorRouter.ethg++;
ISPServidorRouter.pppg++ <--> fiberline <--> internetCloud.pppg++;
ISPClienteRouter.pppg++ <--> fiberline <--> internetCloud.pppg++;
AP.ethg++ <--> Eth100M <--> ISPClienteRouter.ethg++;
```

Para facilitar la comprensión de aquí en adelante se identifica al dispositivo electrónico de adquisición de datos como DPA y al Smartphone o Tablet del usuario como MÓVIL.

El archivo de configuraciones de simulación “.ini”, permite múltiples configuraciones para las diferentes redes establecidas, para la red *simulacion_A* la configuración se denomina [*Config simulacion_A*], los parámetros más importantes de esta configuración se describen a continuación:

Configuraciones de capa de aplicación

Tanto el MÓVIL como el DPA tienen una aplicación UDP que generará un dato a ser transmitido por un puerto determinado, para este ejemplo el puerto seleccionado es el 1000. De la misma manera para recibir un dato se especifica el puerto de destino. La

capa de aplicación genera una trama que contiene la dirección de destino del paquete generado, el DPA envía datos hacia el MÓVIL y este los envía al servidor médico.

En pruebas a usuarios, el dispositivo se activa cada 30 minutos, en ese instante toma la muestra de las señales corporales y las transmite por bluetooth y el MOVIL lo transmite vía internet hacia el servidor médico, ese proceso se repite durante 24 horas. Esto quiere decir que se envía la información 48 veces con un tamaño de 125 Bytes, entre los diferentes dispositivos; por motivos de agilizar la simulación de eventos discretos la simulación genera el envío de un paquete cada segundo durante 48 segundos, generando así la cantidad deseada de paquetes. Los parámetros de inicialización son los siguientes:

```
** .addDefaultRoutes = false
** .Host*.numUdpApps = 1
** .Host*.udpApp[0].localPort = 1000
** .Host*.udpApp[0].typename = "UDPBasicApp"
** .HostBLEdpa.udpApp[0].destAddresses = "HostBLEMovil" #
** .HostBLEMovil.udpApp[0].destAddresses = "HostServidorMedico"
** .Host*.udpApp[0].destPort = 1000
** .Host*.udpApp[0].messageLength = 125B
** .Host*.udpApp[0].sendInterval = 1s
** .Host*.udpApp[0].stopTime = 48.8s
```

Configuraciones de direccionamiento en la nube

Se simula la conexión a internet de los usuarios, con el submódulo *internetCloud* que esta descrito en un archivo “.xml” en el que se configuran parámetros como retardos, velocidad de transmisión, fuente y destino de información que llega a la nube. Para hacer referencia y llamar al archivo para ser parte de la parametrización se utiliza el siguiente código:

```
** .internetCloud.networkLayer.delayer.config = xmldoc
("internetCloud_A.xml")
```

El archivo *internetCloud_A.xml*, contiene la descripción de la fuente del tráfico de datos y el destino, en esta simulación la fuente es el router del ISP del usuario y el destino el router del ISP del servidor médico.

Para establecer los retardos que generará la conexión a internet se realizaron pruebas para medir la latencia de conexión por parte del proveedor de internet fijo TVCABLE, como resultado se detectaron los siguientes valores:

- Prueba 1 y 2: En la Fig. 66 y 67 se muestran resultados de retardos que se producen en una conexión típica a internet. Las mediciones se realizaron desde las páginas: Velocímetro Web [77] y Grupo TV-cable [78].

TEST DE LATENCIA

| RESULTADOS | Madrid (España) | Amsterdam (Holanda) | Virginia (USA) |
|-----------------|-----------------|---------------------|----------------|
| Retardo máximo: | 138.0 ms | 124.5 ms | 112.0 ms |
| Retardo mínimo: | 124.0 ms | 109.0 ms | 61.0 ms |
| Retardo medio: | 128.55 ms | 115.45 ms | 80.9 ms |
| Pérdidas: | 0.0 % | 0.0 % | 0.0 % |

Fig. 66. Prueba 1 de latencia de conexión a internet [77].

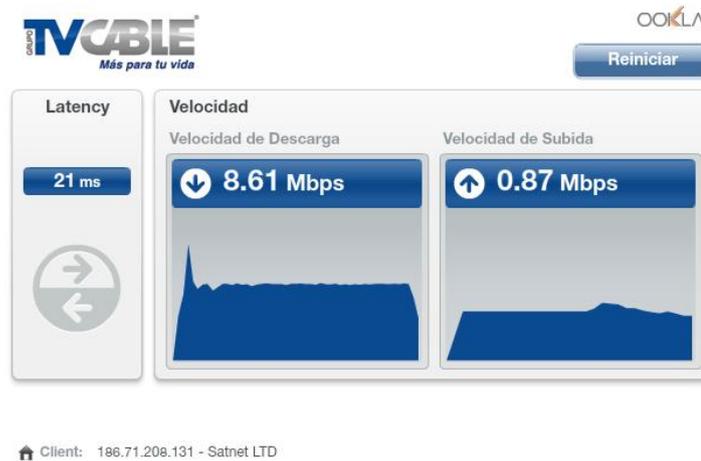


Fig. 67. Prueba 2 de latencia de conexión a internet [78].

- Prueba 3: La Fig. 68 muestra otra prueba desde el CMD Windows con el comando PING apuntando a la dirección 8.8.8.8:

```

Respuesta desde 74.125.141.106: bytes=32 tiempo=85ms TTL=44
Respuesta desde 74.125.141.106: bytes=32 tiempo=87ms TTL=44
Respuesta desde 74.125.141.106: bytes=32 tiempo=85ms TTL=44
Respuesta desde 74.125.141.106: bytes=32 tiempo=84ms TTL=44

Estadísticas de ping para 74.125.141.106:
  Paquetes: enviados = 23, recibidos = 23, perdidos = 0
  (0% perdidos),
  Tiempos aproximados de ida y vuelta en milisegundos:
    Mínimo = 82ms, Máximo = 94ms, Media = 85ms
Control-C
^C
C:\Users\DELL>

```

Fig. 68. Prueba 3 de latencia de conexión a internet.

Fuente: Investigador.

La Tabla 21 detalla en resumen los resultados de las pruebas realizadas:

Tabla 21. Resultados de retados de conexión a internet.

| Prueba | Retardo Mínimo | Retardo Máximo | Retardo Promedio |
|--------|----------------|----------------|------------------|
| 1 | 61 ms | 138 ms | 108,3 ms |
| 2 | 21 ms | - | 21 ms |
| 3 | 82 ms | 94 ms | 85 ms |

Fuente: Investigador.

Los resultados muestran un rango de retardos entre 21 ms y 108,3 ms. Para esta simulación se manejará un retardo (delay) entre 20 ms y 200 ms, donde la librería de simulación permite generar tiempos randómicos con distribución normal entre el intervalo dicho. El ancho de banda a simular en los canales de la Fig. 69 está entre los 10 Mbps y los 100 Mbps, que es el rango de velocidades de trasmisión por fibra óptica que en la actualidad ofrecen empresas servidoras de servicios de internet como CNT y TV-CABLE [79].

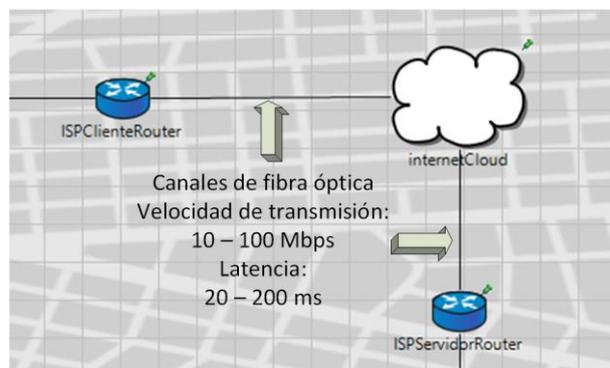


Fig. 69. Detalles de parámetros de canales de fibra óptica y su conexión a internet.

Fuente: Investigador.

Una vez obtenidos esos datos en el archivo *internetCloud_A.xml* se especifican estos datos:

```
<traffic src="ISPClienteRouter" dest="ISPServidorRouter"  
delay="20ms+truncnormal(200ms,60ms)"  
datarate="uniform(10Mbps,100Mbps)" drop="uniform(0,1) &lt; 0.01" />
```

Configuraciones de capa de enlace

En la Fig. 70 se ilustran las interfaces inalámbricas: en el MÓVIL son 2, una para la conexión BLE (wlan[0]) y la otra WLAN (wlan[1]), ambas se especifican como *WirelessNic*. En el DPA solamente se utiliza una interfaz para BLE (wlan[0]), las características de la MAC están definidas por *BMacLayer*, que se encuentra incluida en la pila de protocolos de INET para comunicaciones que usan un método de acceso TDMA (método de acceso característico de BLE y del Bluetooth clásico). Además el ancho de banda de BLE es de 721Kbps o 0,7Mbps, mientras en las definiciones de la interfaz WLAN del MÓVIL definida como *Ieee80211Nic* con método de acceso CSMA/CA se utiliza un ancho de banda de 11Mbps como en el IEEE802.11b.

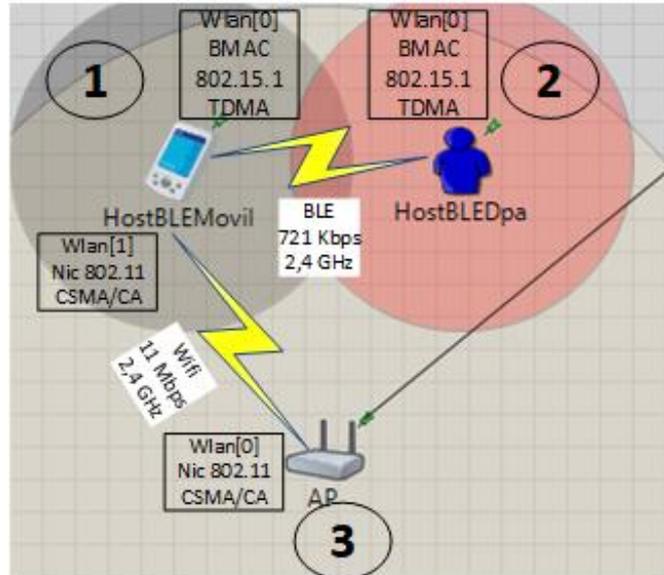


Fig. 70. Detalles de parámetros para las tarjetas inalámbricas.

Fuente: Investigador.

Para estas configuraciones el archivo *omnetpp.ini* contiene las siguientes configuraciones:

```

*.HostBLEMovil.numRadios = 2
*.HostBLEDpa.wlan[0].typename = "WirelessNic"
*.HostBLEDpa.wlan[0].macType = "BMacLayer"
*.HostBLEMovil.wlan[0].typename = "WirelessNic"
*.HostBLEMovil.wlan[0].macType = "BMacLayer"
*.HostBLEMovil.wlan[1].typename = "Ieee80211Nic"
*.HostBLEDpa.**.bitrate = 0.7Mbps
*.HostBLEMovil.wlan[0]**.bitrate = 0.7Mbps
*.HostBLEMovil.wlan[1]**.bitrate = 11Mbps

```

Mientras tanto la MAC y sus parámetros de dirección MAC, tamaño de cabecera, intervalo de chequeo, etc. BLE hace uso de ACKs o acuses de recibido, que permiten comprobar si la comunicación se realizó correctamente.

```

*.HostBLE*.wlan[0].mac.address = "auto"
*.HostBLE*.wlan[0].mac.headerLength = 10B
*.HostBLE*.wlan[0].mac.mtu = 0B
*.HostBLE*.wlan[0].mac.useMACAcks = true
*.HostBLE*.wlan[0].mac.slotDuration = 0.1s
*.HostBLE*.wlan[0].mac.checkInterval = 0.01s
*.HostBLE*.wlan[0].mac.queueLength = 20
*.HostBLE*.wlan[0].mac.animation = false
*.HostBLE*.wlan[0].mac.macMaxFrameRetries = 3

```

Los parámetros de la capa de enlace para la conexión Wi-Fi en INET están configurados por defecto, no hay necesidad de parametrizar los mismos.

Configuraciones de consumo energético BLE

La disponibilidad energética permanente para la adquisición de datos y el envío de información por parte del DPA es de vital importancia para tener un sistema fiable para el usuario. Saber cuánto consume el dispositivo al estar encendido y cuanto recurso energético se gasta al transmitir la información dando una idea clara de cuánto durará el dispositivo encendido.

OMNeT++ maneja unidades como el joule (J) y el miliwatt (mW) para analizar el consumo energético, por lo tanto, los 2000 mAh y 3,7 V que brinda la batería, fueron transformados a joules según la Ecuación 17:

$$Energía(J) = 3600 * Voltaje(V) * Corriente nominal(mAh) \quad (17)$$

$$E(J) = 3600 * 3,7 V * 2000 mAh = 26640 Joules$$

Cada batería aporta con 26640 Joules, en total con las dos baterías se tiene 53280 Joules, la parametrización es la siguiente:

```
*.HostBLE*.energyStorageType = "StateBasedEnergyConsumer "  
*.HostBLE*.energyStorage.nominalCapacity = 53280J  
*.HostBLE*.energyStorage.nodeShutdownCapacity = 12960J  
*.HostBLE*.energyStorage.nodeStartCapacity = this.nominalCapacity
```

El modelo energético *StateBasedEnergyConsumer* se utiliza en OMNeT++ para analizar el consumo energético de radio, por lo cual el consumo en general requiere un análisis a parte según su uso diario. En el siguiente extracto de código se ha configurado el consumo de radio:

```
*.HostBLE*.wlan[0].radio.energyConsumerType =  
"StateBasedEnergyConsumer"  
*.HostBLE*.wlan[0].radio.energyConsumer.offPowerConsumption = 0mW  
*.HostBLE*.wlan[0].radio.energyConsumer.sleepPowerConsumption =  
1.32mW  
*.HostBLE*.wlan[0].radio.energyConsumer.switchingPowerConsumption =  
28.05mW  
*.HostBLE*.wlan[0].radio.energyConsumer.receiverIdlePowerConsumption  
= 28.05mW  
*.HostBLE*.wlan[0].radio.energyConsumer.receiverBusyPowerConsumption  
= 28.05mW  
*.HostBLE*.wlan[0].radio.energyConsumer.receiverReceivingPowerConsumption  
= 28.05mW  
*.HostBLE*.wlan[0].radio.energyConsumer.transmitterIdlePowerConsumption  
= 28.05mW  
*.HostBLE*.wlan[0].radio.energyConsumer.transmitterTransmittingPower  
Consumption = 28.05mW  
*.HostBLE*.energyStorageType = "IdealEnergyStorage"
```

Configuraciones de movilidad

La simulación utiliza modelos realísticos de movimiento de los nodos, esto influye en nivel de señal recibida y las interferencias de radio. Existen varios modelos para movilidad entre los cuales existen movimientos individuales de nodos o en conjunto, también existen modelos determinísticos y modelos de movimiento randómicos.

El nodo del usuario conectado a una red inalámbrica de área local puede moverse en toda un área específica sin perder conexión, para Wi-Fi su rango es de aproximadamente 100 metros [72], en el caso de una conexión BLE el rango de cobertura de la señal es de aproximadamente entre 10 a 50 metros [10].

La simulación A se desarrolla en un área de 200 metros de ancho por 200 metros de largo donde se incluye un tipo de trayectoria establecida por el modelo *TractorMobility* que semeja al movimiento de un usuario dentro de su hogar u oficina como se muestra en la Fig. 71:

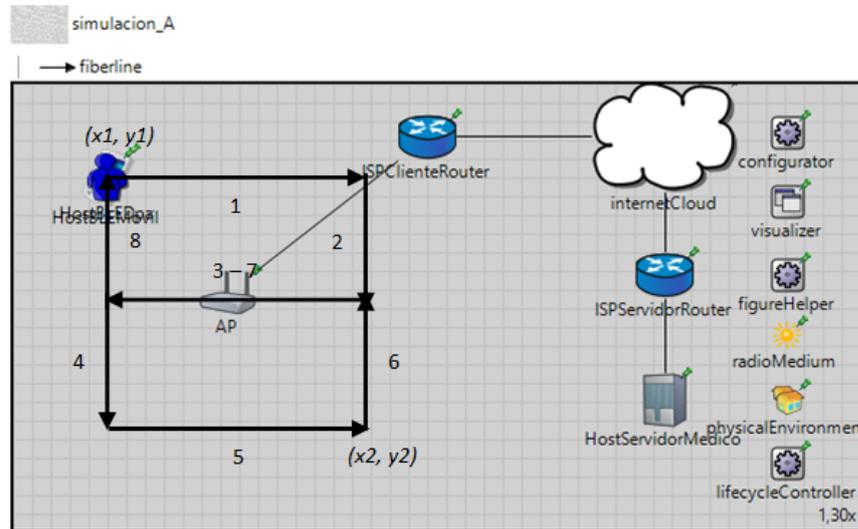


Fig. 71. Trayectoria del nodo según *TractorMobility*.

Fuente: Investigador.

Para este tipo de movilidad se traza una trayectoria con cierto número de filas, la Fig. 71, ilustra el movimiento del nodo con dos columnas, siguiendo la secuencia 1, 2, 3, 4, 5, 6, 7, 8. El punto de partida del nodo (x_1, y_1) , el punto extremo (x_2, y_2) y la velocidad están parametrizados de la siguiente forma:

```

**.HostBLE*.mobilityType = "TractorMobility"
**.HostBLE*.mobility.x1 = 50m
**.HostBLE*.mobility.y1 = 50m
**.HostBLE*.mobility.x2 = 190m
**.HostBLE*.mobility.y2 = 190m
**.HostBLE*.mobility.rowCount = 2
**.HostBLE*.mobility.speed = 10mps

```

Configuraciones de obstáculos físicos

El principal objetivo del submódulo *physicalEnvironment* es describir edificaciones, muros, vegetación, terreno, clima y otros agentes físicos que tendrán efectos en la propagación de la señal, el movimiento, la duración de la batería, etc. Estas configuraciones están descritas en un archivo de extensión “.xml”, en este documento

se especifica la posición de los objetos, sus propiedades físicas y su forma. En la simulación se añaden muros de concreto para analizar el comportamiento inalámbrico de los nodos en el archivo *paredes.xml* estos objetos se definen de la siguiente manera:

```
<object position="min 20 220 0" orientation="0 0 0" shape="cuboid
200 0.3 4" material="concrete" fill-color="203 65 84"
opacity="0.4"/>
```

En el archivo *omnetpp.ini* se especifica el archivo de configuraciones de obstáculos físicos y el modelo de pérdidas por obstáculos:

```
*.physicalEnvironment.config = xmldoc ("paredes.xml")
*.radioMedium.obstacleLossType = "DielectricObstacleLoss"
*.physicalEnvironment.groundType = "FlatGround"
*.physicalEnvironment.ground.elevation = 0m
```

El resultado de la inclusión de obstáculos físicos se muestra en la Fig. 72:

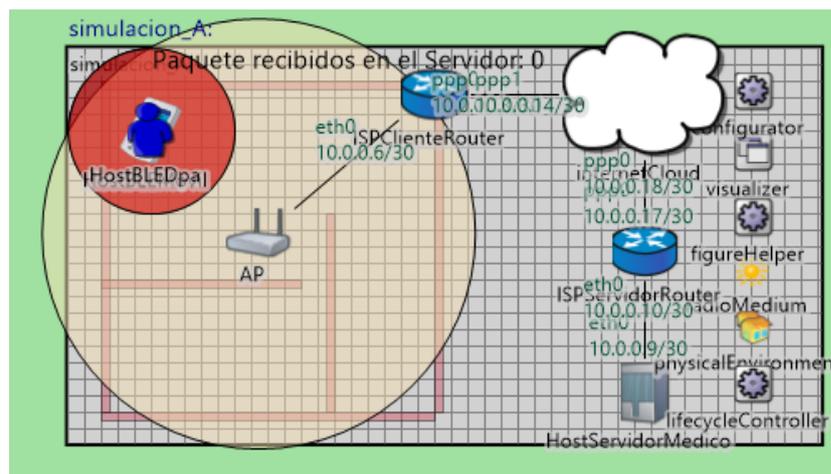


Fig. 72. Inclusión de objetos físicos en la simulación.

Fuente: Investigador.

El módulo *DielectricObstacleLoss* determina la pérdida de potencia en base a las propiedades dieléctricas de los objetos y la reflexión a lo largo del trayecto de propagación considerando la forma, la posición, la orientación y el material de los objetos físicos. Además, se han añadido parámetros del tipo de suelo, *FlatGround* con una elevación de 0 metros, recordar que todas esas variables físicas tienen efecto en el comportamiento de la comunicación.

Configuraciones de radio

El submódulo *radioMedium* permite especificar qué tipo de configuraciones de radio se van a utilizar en la simulación, este módulo realiza un seguimiento de cuales de los nodos inmersos en el enlace de comunicación están dentro de un rango de interferencia generado por otros nodos.

En el desarrollo del protocolo BLE se creó el archivo *Ieee802151BLERadioMedium.ned* compatible con el submódulo *radioMedium* para asemejar un enlace de Bluetooth 4.0, dentro de los parámetros característicos de esta tecnología la simulación maneja los siguientes:

```
*.mediumType = "Ieee802151BLERadioMedium"
*.radioMedium.backgroundNoise.power = -90dBm
*.radioMedium.mediumLimitCache.carrierFrequency = 2.45GHz
*.HostBLE*.wlan[0].radioType = "Ieee802151BLERadio"
*.HostBLE*.wlan[0].radio.carrierFrequency = 2.45GHz
*.HostBLE*.wlan[0].radio.bandwidth = 2MHz
*.HostBLE*.wlan[0].radio.transmitter.power = 1mW
*.HostBLE*.wlan[0].radio.transmitter.preambleDuration = 10us
*.HostBLE*.wlan[0].radio.receiver.sensitivity = -87dBm
*.HostBLE*.wlan[0].radio.receiver.energyDetection = -87dBm
*.HostBLE*.wlan[0].radio.receiver.snirThreshold = -8dB
*.HostBLE*.wlan[0].radio.antennaType = "IsotropicAntenna"
*.HostBLE*.wlan[0].radio.antenna.numAntennas = 1
*.HostBLE*.wlan[0].radio.*.headerBitLength = 54b
```

A la modulación GFSK de BLE en el lado del receptor se ha añadido un modelo de error al recibir un paquete:

```
*.HostBLE*.wlan[0].radio.receiver.errorModelType = "ErrorModelBase"
*.HostBLE*.wlan[0].radio.transmitter.modulation = "GFSK"
*.HostBLE*.wlan[0].radio.receiver.modulation = "GFSK"
```

INET FRAMEWORK es una herramienta de OMNeT++ que tiene un desarrollo robusto del protocolo IEEE 802.11.x por lo cual en la simulación no es necesario configurar y parametrizar este protocolo, basta con declarar las interfaces de los dispositivos e INET se encarga de la parametrización por defecto. Las configuraciones para el enlace Wi-Fi en modo de operación “b” son las siguientes:

```
*.HostBLEMovil.wlan[1].radio.receiver.modulation = "QPSK"
*.HostBLEMovil.wlan[1].radio.transmitter.modulation = "QPSK"
*.AP.wlan[0].radio.receiver.modulation = "QPSK"
*.AP.wlan[0].radio.transmitter.modulation = "QPSK"
*.AP.wlan[0].opMode = "b"
```

De ser necesario se realizan modificaciones del protocolo WLAN desde el archivo *omnetpp.ini*, como el modo de operación del protocolo y el tipo de modulación base. Los parámetros que se utilizan tanto en el MÓVIL y en el AP por defecto son los mostrados en la Tabla 22:

Tabla 22. Parámetros por defecto Wi-Fi en INET.

| Parámetros Wi-Fi | AP y MÓVIL |
|--------------------------------|------------|
| Modo de operación | "b" |
| Velocidad de transmisión | 11 Mbps |
| Duración mínima de canal | 150 ms |
| Duración máxima de canal | 300 ms |
| Tiempo máximo de autenticación | 5 s |
| Intervalo de Beacon | 100 ms |
| Tipo de antena | Isotrópica |
| Frecuencia portadora | 2.412 GHz |
| Ancho de banda | 2 MHz |
| Número de antenas | 1 |
| Duración de preámbulo | 0 s |
| Longitud de cabecera | 192 bits |
| Potencia de transmisión | 20 mW |
| Nivel de detección de energía | -85 dBm |
| Sensibilidad | -85 dBm |
| Umbral de SNIR | 4 dB |

Fuente: Investigador.

Configuraciones de propagación

De manera similar al mundo real un modelo de propagación de señales debe ser utilizado para conocer las variables físicas de una transmisión de radio. En este caso se utiliza un modelo de propagación de interiores que incluye desvanecimientos de señal denominado *RayleighFading*, que en su modelo matemático maneja la longitud de onda, la distancia, un parámetro denominado alpha y la pérdida propia del sistema. También es necesario establecer un tipo de propagación sea de velocidad o tiempo constante y un tipo de ruido de fondo:

```
*.radioMedium.pathLossType = "RayleighFading"
**.propagationType = "ConstantSpeedPropagation"
**.analogModelType = "ScalarAnalogModel "
**.backgroundNoiseType = "IsotropicScalarBackgroundNoise "
```

ConstantSpeedPropagation es un modelo que calcula el tiempo de propagación para que sea proporcional a la distancia recorrida, la proporción es determinada por el parámetro de velocidad constante de propagación.

ScalarAnalogModel es un modelo análogo que calcula la potencia de la señal la cual no puede cambiar sobre el tiempo y la frecuencia a excepción de los límites de la señal.

IsotropicScalarBackgroundNoise describe un ruido de fondo que no cambia en el espacio, tiempo y frecuencia, produce señales de ruido escalar y también puede ser usado en cálculos escalares.

Configuraciones de visualizaciones

El submódulo *visualizer* maneja las características de la simulación para visualizar de forma más clara de lo que sucede en cada evento, este submódulo interviene por ejemplo en la visualización de la trayectoria de los nodos y al mostrar el tipo de paquetes que se está transmitiendo en una comunicación inalámbrica o alámbrica:

```
*.visualizer.sceneVisualizer.descriptionFigure = "title"  
*.visualizer.physicalLinkVisualizer.packetNameFilter =  
"UDPBasicApp*"  
*.visualizer.mobilityVisualizer.displayMovementTrail = true
```

4.7.2 Escenario 1: BLE y Wi-Fi (Simulación B)

Una vez analizado el comportamiento inalámbrico que se da en la red de área personal y en la red de área local, se ha diseñado una red para analizar el tráfico y el comportamiento de capa de aplicación de la red cuando se conectan una gran cantidad de dispositivos para enviar la información de los signos vitales al servidor médico. La topología que se describe en el archivo “.ned” es la mostrada en la Fig. 73.

Un usuario envía 48 paquetes UDP diariamente al servidor médico, estos datos parten desde puntos diferentes y son enviados a la nube para ser direccionados a una base de datos a la cual únicamente tiene acceso el personal médico que realizará el control de salud.

En esta topología no se procesan datos relacionados a la comunicación inalámbrica, únicamente se procesa el funcionamiento del sistema en relación a capa de red y capa de aplicación, comprobando posibles colapsos en la red, pérdida de información, pérdida de paquetes, etc.

Toda la red de área local y área personal que se manejó en la simulación anterior estará representada por un único nodo (*redHostCliente* 1) que representa a un usuario con conexión a internet enviando los 48 datos hacia internet para luego ser recibidos por el host médico (*HostServidorMédico* 3).

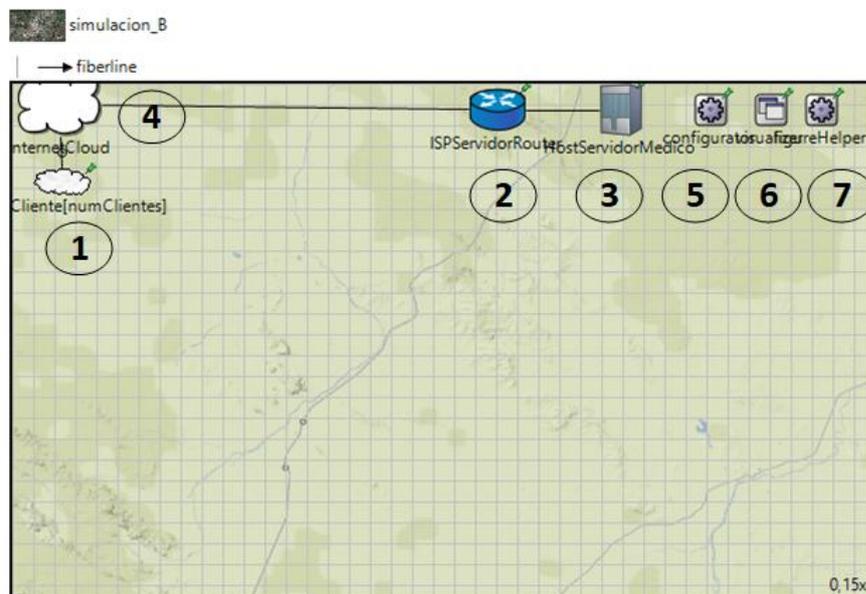


Fig. 73. Descripción de red de la Simulación B del escenario 1.

Fuente: Investigador.

En el archivo “.ned” la red *simulación_B*, especifica cuatro submódulos principales:

1. ***redHostCliente***: representa la red de área local del cliente que tiene salida hacia internet.
2. ***ISPServidorRouter***: puerta de acceso del servidor médico hacia una red de área extensa con conexión a internet, simula la función del ISP del servidor.
3. ***HostServidorMedico***: ordenador del médico de cabecera que controla la salud del usuario.

4. *internetCloud*: submódulo que simula una conexión a internet con un ancho de banda y latencia determinada, permite la conexión de N cantidad de redes de usuario hacia el host del servidor médico.

El código generado es el siguiente:

```
HostServidorMedico: StandardHost {
    parameters:
        @display ("p=2809.3555,128.67278;i=device/server");
}
redHostCliente[numClientes]: StandardHost {
    parameters:
        @display ("p=228.735,452.4975,matrix,15,200,200;i=old/cloud");
}
ISPServidorRouter: Router {
    parameters:
        @display ("p=2237.4766,128.67278");
}
internetCloud: InternetCloud {
    @display ("p=224.50874,100.86625;is=1");
}
```

Una de las fortalezas de esta simulación es la de poder ingresar N cantidad de nodos que envía datos al servidor médico, este parámetro (numClientes) se declara en el archivo “.ned” pero se lo ingresa al ejecutar la simulación desde el archivo *omnetpp.ini*. Apenas ejecutada la simulación aparecerá una ventana tipo diálogo como la de la Fig. 74 donde se ingresan los parámetros aún no designados. En el archivo “.ned” se lo define así:

```
network simulacion_B
{
    parameters:
        int numClientes;
    ...
}
```

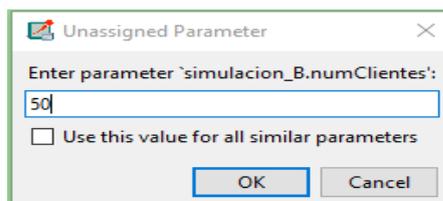


Fig. 74. Ingreso del número de clientes para la Simulación B del Escenario 1.

Fuente: Investigador.

La Fig. 75 ilustra el resultado de ingresar el número de usuarios, en este ejemplo 50 usuarios, el ambiente de simulación muestra lo siguiente:

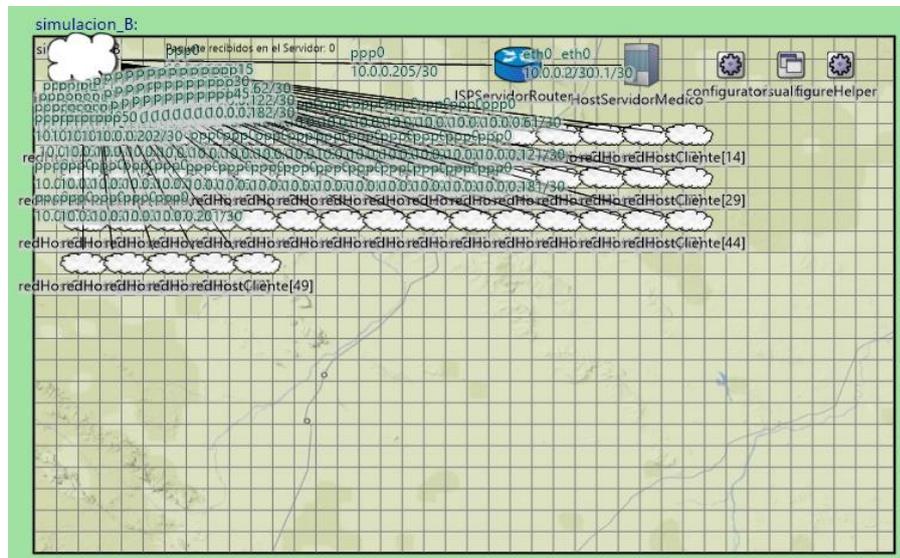


Fig. 75. Ambiente de la Simulación B al ingresar la cantidad de clientes.

Fuente: Investigador.

Los submódulos de configuración utilizados son los siguientes:

5. **Configurator:** configura las direcciones IP de manera dinámica o estática, tomando en cuenta subredes y optimiza las tablas de ruteo generadas en cada nodo de la red.
6. **Visualizer:** permite generar un solo módulo que contiene las diferentes visualizaciones o animaciones de la simulación.
7. **FigureHelper:** encuentra y administra todas las propiedades de los denominados *@delegatesignal*. Que almacenan datos específicos dentro de la simulación, por ejemplo, para ser visualizados en la ventana de la propia simulación.

El canal de fibra óptica se mantiene en la red *simulación_B*, para simular la conexión de la red de cada usuario hacia la nube y también desde la nube al router del servidor médico (ISP Servidor Router 2).

types:

channel fiberline **extends** ThruputMeteringChannel

```

    {
        delay = 1us;
        datarate = 100Mbps;
        thruputDisplayFormat = "u";
    }

connections:
HostServidorMedico.ethg++ <--> Eth100M <-->ISPServidorRouter.ethg++;
ISPServidorRouter.pppg++ <--> fiberline <--> internetCloud.pppg++;

for i=0..numClientes-1
{
redHostCliente[i].pppg++ <--> fiberline <--> internetCloud.pppg++;
}

```

El ciclo “for” permite que según la cantidad de usuarios ingresados se creen el mismo número de conexiones e interfaces para comunicar a los clientes con la nube.

Las configuraciones de simulación en el archivo “.ini” para la red *simulacion_B* la configuración se denomina [*Config simulacion_B*], los parámetros más importantes de esta configuración se describen a continuación:

Configuraciones de capa de aplicación

Los clientes *redHostCliente[*]* tienen una aplicación UDP que generará un dato a ser transmitido por un puerto determinado estas configuraciones son similares a la simulación anterior. La capa de aplicación de la red de cada cliente genera una trama que contiene la dirección de destino del paquete generado (HostServidorMedico 3).

El tamaño del dato que se va a enviar es el mismo determinado en la simulación anterior, es decir 125 Bytes con el formato: 2017/6/15 13:50:04 2,0878 1,0879 120 80 95 60.

Cada nodo envía la información 48 veces, por motivos de agilizar la simulación de eventos discretos la simulación genera el envío de un paquete cada segundo durante 48 segundos generando así la cantidad deseada de paquetes.

```

**.addDefaultRoutes = false
**.Host*.numUdpApps = 1
**.Host*.udpApp[0].localPort = 1000
**.Host*.udpApp[0].typename = "UDPBasicApp"

```

```

**.redHostCliente[*].udpApp[0].destAddresses = "HostServidorMedico"
**.Host*.udpApp[0].destPort = 1000
**.Host*.udpApp[0].messageLength = 125B
**.Host*.udpApp[0].sendInterval = 1s
**.Host*.udpApp[0].stopTime = 48.99s

```

Configuraciones de direccionamiento en la nube

La diferencia con relación a la *simulacion_A* es en el submódulo *internetCloud* la fuente de destino de la conexión son los nodos *redHostCliente[*]*.

```

**.internetCloud.networkLayer.delayer.config = xmldoc
("internetCloud_B.xml")

```

Los demás parámetros en el archivo *internetCloud_B.xml* son los mismos:

```

<traffic src="redHostCliente[*]" dest="ISPServidorRouter"
delay="20ms+truncnormal(200ms,60ms)"
datarate="uniform(100kbps,512Mbps)" drop="0 &lt; 0" />

```

Configuraciones de visualizaciones

Como en la simulación anterior los parámetros de visualización son los siguientes:

```

*.visualizer.sceneVisualizer.descriptionFigure = "title"
*.visualizer.physicalLinkVisualizer.packetNameFilter =
"UDPBasicApp*"
*.visualizer.mobilityVisualizer.displayMovementTrail = true

```

4.7.3 Escenario 2: BLE y LTE (Simulación C)

Topología en la cual un usuario se conecta a una red de telefonía móvil 4G LTE para enviar los datos de presión arterial y electrocardiograma, esta definición de red se utiliza para realizar el análisis del comportamiento de la comunicación inalámbrica de múltiples dispositivos celulares que tengan disponibilidad de datos móviles para enviar la información de sus variables físicas a internet y posteriormente al servidor médico, esta topología se detalla en la Fig. 76:

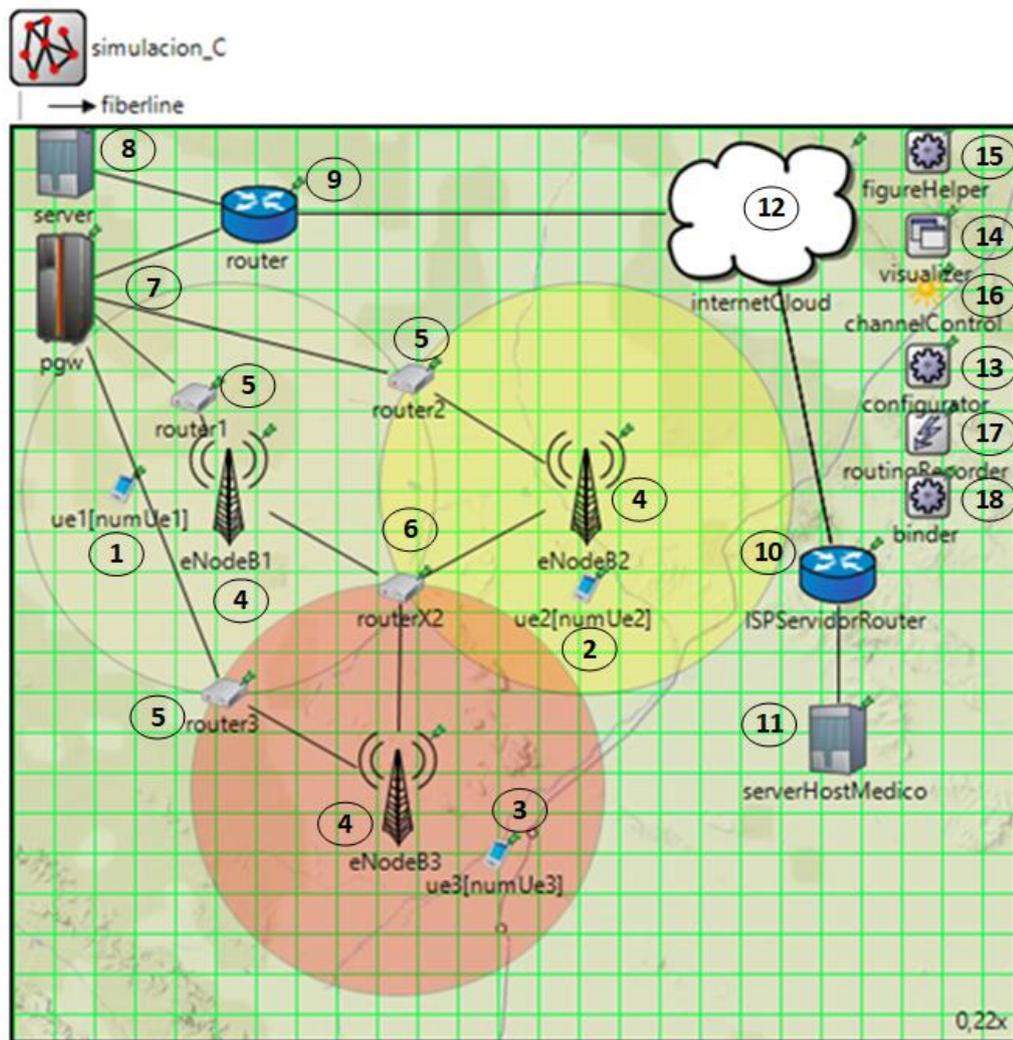


Fig. 76. Descripción de red de la Simulación C del escenario 2.

Fuente: Investigador.

En esta simulación no se implementa el protocolo de WPAN BLE ya que la plataforma LTE de OMNeT++ permite únicamente la inclusión de parámetros de radio relacionados a redes celulares 4G. El nodo del usuario es decir un celular o una Tablet con conexión a una red de telefonía móvil en LTE denominados Ue ($ue^{*[*]}$) se conectan a los *Evolved Node B* ($eNodeB^{*}$) por medio de una interfaz de radio especificada por el protocolo 3GPP.

Los *Evolved Node B* ($eNodeB^{*}$) son las estaciones de radio que dan la cobertura de telefonía celular en un área específica, en esta simulación se han colocado tres estaciones de radio que están interconectadas con una topología de estrella, debido a esto, es necesario que cada estación de radio en su infraestructura cuente con un enrutador de paquetes ($router^{*}$) que dirija el tráfico de paquetes hacia el *PDN Gateway*

o *PGW* (pgw 7) que cumple un papel crucial en el núcleo de la red 4G, ya que es encargado de comunicar la red celular con cualquier red de tráfico de datos como internet [80].

Si no existe conexión directa entre un *eNodeB* y la red núcleo de LTE se utilizan las interfaces X2 que interconectan los *eNodeBs* comúnmente por medio de un router (routerX2 6) que encamina los paquetes hacia el PGW (pgw 7).

El PGW (pgw 7) se conecta a una red de datos con un router de salida (router 9), el cual se conecta a un servidor de telefonía LTE (server 8) simulando la estación desde la cual se administra y gestiona la red. Este router de la red de núcleo tiene acceso a internet (internetCloud 12), en la simulación se utiliza un canal de fibra óptica para simular ese acceso, canal que de igual forma interconecta internet y el router del servidor médico (ISPServidorRouter 10), finalmente llegando con conexión Ethernet hacia el host dedicado al control de los pacientes (serverHostMedico 11), mismo que deberá contar con una interfaz y base de datos capaz de mostrar y almacenar la situación actual de las variables fisiológicas del usuario.

De tal manera esta topología analiza todos los parámetros concernientes al estándar 3GPP, entre los cuales están: latencia, pérdida de paquetes, efectos de movilidad de los nodos, calidad y problemáticas de canal inalámbrico, etc.

En el archivo “.ned” *redLTE.ned* la red *simulacion_C*, especifica 7 submódulos principales:

1. ***ue1[numUe1]***: dispositivo móvil del usuario sea un Smartphone o Tablet, perteneciente a la celda del *eNodeB1*, el número de dispositivos viene dado por el parámetro *numUe1* que es ingresado al momento de correr la simulación.
2. ***ue2[numUe2]***: dispositivo móvil del usuario sea un Smartphone o Tablet, perteneciente a la celda del *eNodeB2*, el número de dispositivos viene dado por el parámetro *numUe2* que es ingresado al momento de correr la simulación.
3. ***ue3[numUe3]***: dispositivo móvil del usuario sea un Smartphone o Tablet, perteneciente a la celda del *eNodeB3*, el número de dispositivos viene dado por el parámetro *numUe3* que es ingresado al momento de correr la simulación.
4. ***eNodeB1, eNodeB2, eNodeB3***: es el componente de la estación base de la red LTE que provee una cobertura para los usuarios de banda ancha móvil.

5. **router1, router2, router3:** enrutadores que enlazan los *eNodeBs* con el *Gateway PGW* (pgw).
6. **routerX2:** interconecta las interfaces de comunicación X2 entre *eNodeBs*.
7. **pgw:** *PDN Gateway* es la parte del núcleo de LTE encargado de comunicar la red móvil con la red de acceso a internet.
8. **server:** representa al servidor utilizado en la red LTE para el control y gestión del tráfico de información.
9. **router:** puerta de acceso de la red LTE hacia una red de área extensa con conexión a la nube.
10. **ISPServidorRouter:** puerta de acceso del servidor médico hacia una red de área extensa con conexión a internet, simula la función del ISP del servidor.
11. **serverHostMedico:** ordenador del médico de cabecera que monitorea la salud del usuario.
12. **internetCloud:** submódulo que simula una conexión a internet con un ancho de banda y latencia determinada, permite la conexión de la red del ISP del usuario y la del servidor médico.

Luego de incluir los componentes de topología desde la ventana de diseño, se generó el siguiente código:

```

server: StandardHost {
    @display ("p=125.199974,87.639984;is=n;i=device/server");
}
pgw: PgwStandardSimplified {
    nodeType = "PGW";
    @display ("p=121.02664,392.29324;is=1");
}
router1: Router {
    @display ("p=438.19992,655.2132;i=device/smallrouter;is=s");
}
router2: Router {
    @display ("p=959.86646,605.13324;i=device/smallrouter;is=s");
}
router3: Router {
    @display ("p=509.14658,1364.6797;i=device/smallrouter;is=s");
}
routerX2: Router {
    @display ("p=934.8265,1110.1064;i=device/smallrouter;is=s");
}

```

```

}
eNodeB1: eNodeB {
    @display("p=517.4932,872.2265;is=l;r=500,wheat");
}
eNodeB2: eNodeB {
    @display("p=1377.1997,872.2265;is=l;r=500,yellow");
}
eNodeB3: eNodeB {
    @display("p=926.4798,1594.213;is=l;r=500,red");
}
ue1[numUe1]: Ue {
    @display("p=267.0933,872.2265;is=vs");
}
ue2[numUe2]: Ue {
    @display("p=1377.1997,1110.1064;is=vs");
}
ue3[numUe3]: Ue {
    @display("p=1164.3597,1752.7997;is=vs");
}

internetCloud: InternetCloud {
    @display("p=1802.8796,204.49329;is=v1");
}

ISPServidorRouter: Router {
    parameters:
        @display("p=1982.3329,1076.7197");
}

serverHostMedico: StandardHost {
    parameters:
        @display("p=1982.3329,1473.1864;i=device/server");
}

router: Router {
    @display("p=592.6132,208.66663");
}

```

Los submódulos de configuración utilizados son los siguientes:

- 13. Configurator:** configura las direcciones IP de manera dinámica o estática, tomando en cuenta subredes y optimiza las tablas de ruteo generadas en cada nodo de la red.
- 14. Visualizer:** permite generar un solo módulo que contiene las diferentes visualizaciones o animaciones de la simulación.
- 15. FigureHelper:** encuentra y administra todas las propiedades de los denominados *@delegatesignal*. Que almacenan datos específicos dentro de la simulación, por ejemplo, para ser visualizados en la ventana de la propia simulación.

16. channelControl: mantiene la información de la ubicación y el movimiento de los nodos, y determina cuales nodos están en un rango de comunicación o interferencia. Esa información es utilizada por las interfaces de radio de los nodos al momento de transmitir.

17. routingRecorder: guarda los cambios en las tablas de enrutamiento y la tabla de interfaces de todos los hosts y routers.

18. binder: almacena una tabla que contiene por cada dirección IP el ID correspondiente a cada nodo. Es usado por el nodo que transmite para encontrar el ID del nodo de destino.

Al igual que en las simulaciones anteriores se ha añadido un canal de fibra óptica para simular la conexión de las distintas redes a internet de cada uno de los involucrados hacia internet. Además, se utilizan diferentes canales Ethernet para interconectar los diferentes dispositivos, el más usado es el canal Eth10G.

types:

```
channel fiberline extends ThruptMeteringChannel
{
    delay = 1us;
    datarate = 100Mbps;
    thruptDisplayFormat = "u";
}
```

connections:

```
router.pppg++ <--> Eth10G <--> pgw.filterGate;
server.pppg++ <--> Eth10G <--> router.pppg++;
pgw.pppg++ <--> Eth10G <--> router1.pppg++;
pgw.pppg++ <--> Eth10G <--> router2.pppg++;
pgw.pppg++ <--> Eth10G <--> router3.pppg++;
router1.pppg++ <--> Eth10G <--> eNodeB1.ppp;
router2.pppg++ <--> Eth10G <--> eNodeB2.ppp;
router3.pppg++ <--> Eth10G <--> eNodeB3.ppp;
ISPServidorRouter.pppg++ <--> fiberline <--> internetCloud.pppg++;
serverHostMedico.ethg++ <--> Eth100M <--> ISPServidorRouter.ethg++;
eNodeB1.x2++ <--> Eth10G <--> routerX2.pppg++;
eNodeB2.x2++ <--> Eth10G <--> routerX2.pppg++;
eNodeB3.x2++ <--> Eth10G <--> routerX2.pppg++;
router.pppg++ <--> fiberline <--> internetCloud.pppg++;
```

La red se ha definido dentro del archivo “.ned” como *simulacion_C* y en el archivo de configuraciones de simulación “.ini” la configuración se denomina “[Config simulacion_C]”, los parámetros más importantes de esta configuración se describen a continuación:

Configuraciones de capa de aplicación

Cada *Ue* que está conectado a la red LTE a los diferentes *eNodeBs* tiene una aplicación UDP que generará un dato a ser transmitido por un puerto determinado, para este ejemplo el puerto seleccionado es el 1000. La capa de aplicación genera una trama de 125 Bytes que contiene los datos de presión arterial y electrocardiograma como se especificó en las simulaciones anteriores. Es necesario especificar la dirección de destino del paquete generado, en este caso el destino de los paquetes es el Servidor médico (*serverHostMedico*).

El dispositivo se activa cada 30 minutos, en ese instante toma la muestra de las señales corporales y las transmite por bluetooth al MÓVIL (*ue*[*]*), el cual lo trasmite utilizando la infraestructura 4G y accede a internet para direccionar los datos hacia el servidor médico, ese proceso se repite durante 24 horas. Esto quiere decir que se envía la información 48 veces por día, por motivos de agilizar la simulación de eventos discretos la simulación genera el envío de un paquete cada segundo durante 48 segundos generando así la cantidad deseada de paquetes.

```
** .addDefaultRoutes = false
** .ue*[*].numUdpApps = 1
** .server*.numUdpApps = 1
** .ue*[*].udpApp[0].typename = "UDPBasicApp"
** .ue*[*].udpApp[*].localPort = 1000
** .server*.udpApp[0].typename = "UDPBasicApp"
** .server*.udpApp[*].localPort = 1000
** .ue*[*].udpApp[0].destAddresses = "serverHostMedico"
** .server*.udpApp[0].destPort = 1000
** .server*.udpApp[0].messageLength = 125B
** .ue*[*].udpApp[0].destPort = 1000
** .ue*[*].udpApp[0].messageLength = 125B
** .ue*[*].udpApp[0].sendInterval = 1s
** .ue*[*].udpApp[0].stopTime = 48.9s
```

Configuraciones de direccionamiento en la nube

La red LTE por medio del Gateway PDN (*pgw*) y el router (*router*) permiten la salida de la información hacia internet que es simulado en el módulo *internetCloud* (*internetCluod*) que esta descrito en un archivo “.xml” en el que se configuran

parámetros como retardos, velocidad de transmisión, fuente y destino de información que llega a la nube.

```
**internetCloud.networkLayer.delayer.config = xmldoc  
("internetCloud_C.xml")
```

El archivo *internetCloud_C.xml*, contiene la descripción de la fuente del tráfico de datos y el destino, en esta simulación la fuente es el router conectado a la red LTE (router) y el destino el router del ISP del servidor médico (ISPServidorRouter).

Para establecer los retardos que generará la conexión a internet se realizaron pruebas para medir la latencia de conexión de la red móvil del proveedor Movistar, como resultado se detectaron los siguientes resultados:

- PRUEBA 1 y 2: Con la ayuda de un programa terminal instalado en un Smartphone, las pruebas se realizaron haciendo ping a la dirección 8.8.8.8 y www.google.com. En las Fig. 77 y Fig. 78 se han analizado zonas con nivel de señal medio y bajo, los resultados de latencia muestran los siguientes resultados:

```
64 bytes from mia07s27-in-f4.1e100.net (216.58.219.164): icmp  
p_seq=14 ttl=50 time=97.8 ms  
64 bytes from mia07s27-in-f4.1e100.net (216.58.219.164): icmp  
p_seq=15 ttl=50 time=86.5 ms  
^C  
--- www.google.com ping statistics ---  
15 packets transmitted, 15 received, 0% packet loss, time 14  
022ms  
rtt min/avg/max/mdev = 80.384/89.198/103.203/6.934 ms  
u0_a146@HwVNS-H:/ $
```

Fig. 77. Prueba 1 de latencia de conexión a internet.

Fuente: Investigador.

```
64 bytes from 8.8.8.8: icmp_seq=27 ttl=53 time=92.0 ms  
64 bytes from 8.8.8.8: icmp_seq=28 ttl=53 time=81.1 ms  
64 bytes from 8.8.8.8: icmp_seq=29 ttl=53 time=80.5 ms  
64 bytes from 8.8.8.8: icmp_seq=30 ttl=53 time=91.6 ms  
^C  
--- 8.8.8.8 ping statistics ---  
30 packets transmitted, 30 received, 0% packet loss, time 29  
044ms  
rtt min/avg/max/mdev = 74.645/83.967/115.270/10.211 ms  
u0_a146@HwVNS-H:/ $
```

Fig. 78. Prueba 1 de latencia de conexión a internet.

Fuente: Investigador.

La Tabla 22 muestra en resumen esos resultados:

Tabla 23. Resultados de retados de conexión a internet con red de datos móviles.

| Prueba | Retardo Mínimo | Retardo Máximo | Retardo Promedio |
|--------|----------------|----------------|------------------|
| 1 | 80,38 ms | 103,2 ms | 89,2 ms |
| 2 | 74,64 ms | 115,27 ms | 83,97 ms |

Fuente: Investigador.

Los resultados muestran un rango de retardos entre 74,64 ms y 115,27 ms. Para esta simulación se manejará un retardo (delay) entre 20 ms y 200 ms. Los parámetros son los mismos que en la simulación A y B, pero detallando la fuente de información y el destino de la misma de la siguiente manera:

```
<traffic src="router" dest="ISPServidorRouter"
delay="20ms+truncnormal(200ms,60ms)"
datarate="uniform(10Mbps,100Mbps)" drop="uniform(0,1) &lt; 0.01" />
```

Configuraciones de capa de enlace

Tanto los *eNodeBs* como los dispositivos móviles *Ue* cuentan con una interfaz inalámbrica (nic) específica para conexiones LTE. Como indica la Fig. 79, las tarjetas inalámbricas LTE en OMNeT++, utilizan el método de acceso OFDMA y están dispuestas de la siguiente manera:

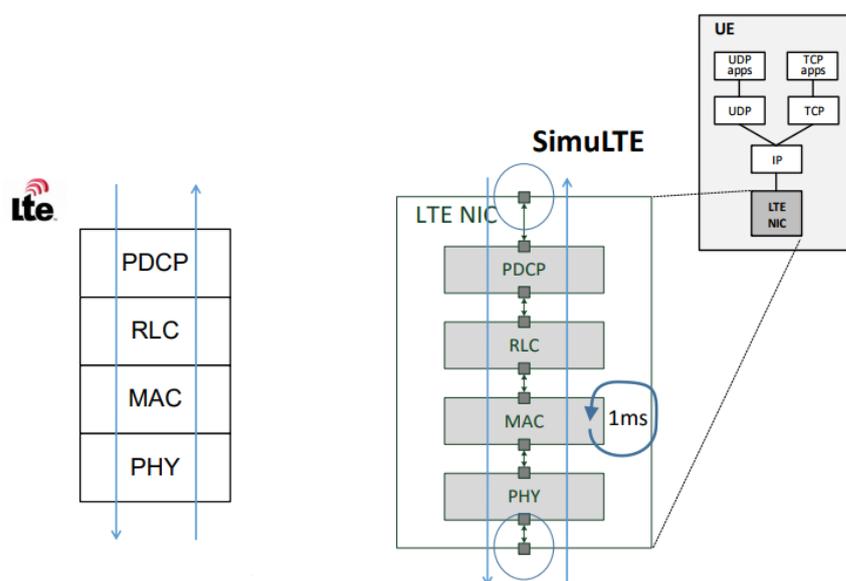


Fig. 79. Estructura de la NIC de LTE en OMNeT++ [81].

En OMNeT++ se componen de los submódulos mostrados en la Fig. 80:

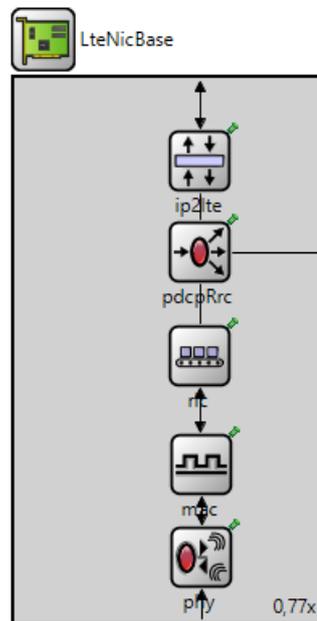


Fig. 80. Módulo compuesto LteNicBase en OMNeT++.

Fuente: Investigador.

Los parámetros Mac incluyen la cantidad de bytes gestión de paquetes para la detección de dispositivos móviles por parte de los *eNodeBs*, tamaño de buffers para datos en cola y el tipo de algoritmo de planificación de recursos del enlace. El algoritmo MAXCI permite conexiones UL, DL y D2D.

```

**.mac.queueSize = 1MiB
**.mac.maxBytesPerTti = 1KiB
**.mac.schedulingDisciplineDl = "MAXCI"
**.mac.schedulingDisciplineUl = "MAXCI"

```

Configuraciones de movilidad y asociación

Este escenario permite que todos los nodos estén en constante movimiento, la velocidad que se ha parametrizado semeja al traslado de los usuarios en un automóvil con tráfico moderado, y es de aproximadamente 36 Km/h o 10 mps. Existen varios modelos para movilidad entre los cuales existen movimientos individuales de nodos o en conjunto, también existen modelos determinísticos y modelos de movimiento randómicos.

Cada nodo de usuario o Ue pertenece a una determinada celda es decir se enlaza a un determinado $eNodeB$, para esta simulación se ha dividido el área en tres celdas 4G en la que se ingresa la cantidad de usuarios que se encuentran en la misma. La asociación permite seleccionar a que $eNodeB$ pertenece cada grupo de dispositivos móviles, detallando el ID de la estación 4G. Sus configuraciones son las siguientes:

```

** .ue1[*].masterId = 1
** .ue1[*].macCellId = 1
** .ue2[*].masterId = 2
** .ue2[*].macCellId = 2
** .ue3[*].masterId = 3
** .ue3[*].macCellId = 3

```

La simulación C se desarrolla en un área de 1 Km de radio de cada celda donde se incluye un tipo de trayectoria establecida por el modelo *GaussMarkovMobility* que semeja al movimiento aleatorio de un usuario con una velocidad dada, esto se observa en la Fig. 81.

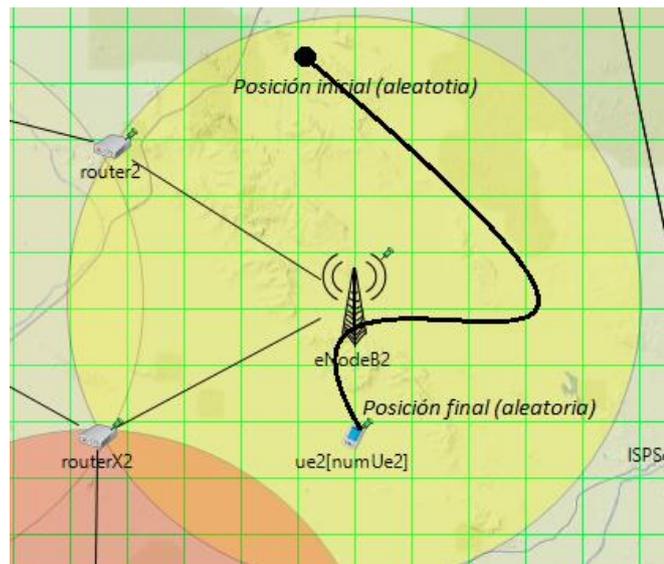


Fig. 81. Ejemplo de Trayectoria del nodo según *GaussMarkovMobility*.

Fuente: Investigador.

Este modelo de movilidad contiene parámetros de ajuste, que controlan y generan de forma randómica el movimiento del nodo. La magnitud y dirección de la velocidad ingresada en el lapso de tiempo n serán s_n y d_n . La siguiente dirección y velocidad son calculadas con la Ecuación 18 y Ecuación 19, así:

$$s_{n+1} = \alpha s_n + (1 - \alpha) \bar{s} + \sqrt{(1 - \alpha^2)} s_{x_n} \quad (18)$$

$$d_{n+1} = \alpha s_n + (1 - \alpha)\bar{d} + \sqrt{(1 - \alpha^2)}d_{x_n} \quad (19)$$

Donde \bar{s} y \bar{d} son constantes que representan el valor medio de la velocidad y dirección, s_{x_n} y d_{x_n} son variables randómicas con distribución Gaussiana [50]. Los puntos entre los cuales se ubicarán los nodos de usuario según la ubicación del *eNodeB* al que pertenecen, son los siguientes:

```
*.ue1[*].mobility.constraintAreaMinX = 300m
*.ue1[*].mobility.constraintAreaMinY = 600m
*.ue1[*].mobility.constraintAreaMaxX = 800m
*.ue1[*].mobility.constraintAreaMaxY = 1200m
*.ue2[*].mobility.constraintAreaMinX = 1200m
*.ue2[*].mobility.constraintAreaMinY = 600m
*.ue2[*].mobility.constraintAreaMaxX = 1700m
*.ue2[*].mobility.constraintAreaMaxY = 1200m
*.ue3[*].mobility.constraintAreaMinX = 700m
*.ue3[*].mobility.constraintAreaMinY = 1400m
*.ue3[*].mobility.constraintAreaMaxX = 1200m
*.ue3[*].mobility.constraintAreaMaxY = 2000m
```

Para este tipo de movilidad se ha definido un intervalo de actualización de posición de un segundo, la velocidad de 10 mps, los parámetros de varianza para generar las variables randómicas Gaussianas y el margen al cual el nodo móvil se repele de los bordes especificados:

```
**.updateInterval = 1s
**.ue*[*].mobilityType = "GaussMarkovMobility"
**.ue*[*].mobility.alpha = 0.9
**.ue*[*].mobility.speed = 10mps
**.ue*[*].mobility.variance = 40
**.ue*[*].mobility.margin = 30m
```

Es posible añadir métodos de *Handover* que permiten a los nodos móviles pasar de una celda a otra sin perder comunicación, tornando el siguiente parámetro a *true*.

```
**.enableHandover = false
```

Configuraciones de radio

El submódulo *channelControl* permite especificar qué tipo de configuraciones de radio se van a utilizar en la simulación, y la mantiene al tanto de la ubicación y el movimiento de los nodos, para de tal manera determinar que nodos están en un área de cobertura y que nodos están en un rango de interferencia.

Los parámetros principales de este submódulo son: la potencia máxima de transmisión de la red, umbral de atenuación de la señal, coeficiente de pérdida por trayectoria, frecuencia de portadora base para todos los canales.

```
** .channelControl.pMax = 10W
** .channelControl.alpha = 1.0
** .channelControl.carrierFrequency = 1980e+6Hz
** .channelControl.sat = -110dBm
```

Las potencias de transmisión de los *Ue* (26 dBm), *eNodeBs* (46 dBm) y de micro celdas (20 dBm) se parametrizan de la siguiente manera:

```
** .ueTxPower = 26
** .microTxPower = 20
** .*TxPower = 46
```

En la simulación se habilitan retardos de transmisión en el canal inalámbrico:

```
** .nic.phy.usePropagationDelay = true
```

Para redes LTE en OMNeT++ el canal inalámbrico se maneja de manera muy detallada desde los archivos con extensión “.xml”. Para la simulación C el modelo de canal se ha detallado en el archivo *config_channel.xml* y contiene los parámetros detallados en la Tabla 24:

Tabla 24. Parámetros del canal inalámbrico de la Simulación C archivo *config_channel.xml*.

| Parámetros LTE | Valor | Detalle |
|--|-----------------|---|
| Shadowing | true | Efecto de que la potencia de la señal recibida fluctúa debido a que los objetos obstruyen la trayectoria de propagación entre el transmisor y el receptor |
| Escenario (scenario) | URBAN_MACROCELL | Tipo de escenario de la red LTE |
| Altura eNodeB (nodeb-height) | 25 m | Altura de la estación de radio de la red LTE |
| Altura de las construcciones (building-height) | 20 m | Altura de las edificaciones del escenario |
| Frecuencia portadora (carrierFrequency) | 1,98 Ghz | Frecuencia portadora base para todos los canales |
| BLER (targetBler) | 0.001 | Relación entre el número de bloques erróneos recibidos y el número total de bloques enviados |
| Reducción de HARQ (harqReduction) | 0.2 | Índice de reducción del proceso de solicitud de repetición automática híbrida (HARQ) |
| Indicador de clasificación (lambdaMinTh) | 0.02 | Indica el número de diferentes flujos de datos que se transmitirán simultáneamente en los mismos recursos de tiempo y frecuencia |

| | | |
|---|------------|--|
| Indicador de clasificación (lambdaMaxTh) | 0.2 | Indica el número de diferentes flujos de datos que se transmitirán simultáneamente en los mismos recursos de tiempo y frecuencia |
| Indicador de clasificación (lambdaRatioTh) | 20 | Indica el número de diferentes flujos de datos que se transmitirán simultáneamente en los mismos recursos de tiempo y frecuencia |
| Ganancia del Ue (antennaGainUe) | 0 dB | Ganancia de la antena de los dispositivos móviles |
| Ganancia de las antenas del eNodeB (antennGainEnB) | 18 dB | Ganancia de las antenas de la radio base de la red LTE |
| Ganancia de las antenas del eNodeB microcelda (antennGainMicro) | 5 dB | Ganancia de las antenas de la radio base de la red LTE en una microcelda |
| Ruido térmico para ancho de banda de 10MHz (thermalNoise) | -104.5 dBm | Tipo de ruido que aparece entre el emisor y receptor que no se puede eliminar |
| Figura de ruido en el Ue (ue-noise-figure) | 7 dB | Cociente entre el S/N de entrada y el S/N de salida en unidades logarítmicas |
| Figura de ruido en los eNodeBs (bs-noise-figure) | 5 dB | Cociente entre el S/N de entrada y el S/N de salida en unidades logarítmicas |
| Pérdida por cable (cable-loss) | 2 dB | Pérdida de potencia por las conexiones cableadas |
| Línea de vista (dynamic-los) | false | Si se habilita cambia dinámicamente entre el cálculo de pérdida por trayectoria LOS/NLOS |
| Línea de vista (fixed-los) | false | Si dynamic-los es falso y este parámetro es verdadero calcula como LOS caso contrario NLOS |
| Desvanecimiento (fading) | True | Habilita los efectos del desvanecimiento de la señal |
| Desvanecimiento (fading-type) | RAYGHLEY | Modelo estadístico para el efecto de un entorno de propagación en una señal de radio |
| Interferencias (extCell-interference) | true | Habilita interferencias generadas por celdas externas al sistema |
| Interferencias (multiCell-interference) | false | Deshabilita la interferencia de otras celdas del mismo sistema |

Fuente: Investigador.

Todos los parámetros descritos se basan en el protocolo 3GPP de LTE y en especificaciones y directrices del modelo de canal realista de UIT-R M.2135-1, dedicado a la evaluación de tecnologías de interfaz radioeléctrica para IMT-Advanced. Estas configuraciones se especifican siguiendo la estructura nombre de parámetro, tipo de dato y valor:

```

<!-- Enable/disable shadowing -->
<parameter name="shadowing" type="bool" value="true"/>
<!-- Pathloss scenario from ITU -->
<parameter name="scenario" type="string" value="URBAN_MICROCELL"/>
<!-- eNodeB height -->
<parameter name="nodeb-height" type="double" value="25"/>

```

Configuraciones de canal inalámbrico de retorno

En el archivo *config_channel.xml* también se especifican características detalladas en la Tabla 25 con respecto al canal de retroalimentación o de retorno, esos parámetros controlan el sensado periódico y no periódico de los canales y las transmisiones de retroalimentación.

Tabla 25. Parámetros del canal inalámbrico de retorno de la Simulación C archivo *config_channel.xml*.

| Parámetros LTE | Valor | Detalle |
|--|-------|--|
| BLER (targetBler) | 0.001 | Relación entre el número de bloques erróneos recibidos y el número total de bloques enviados |
| Indicador de clasificación (lambdaMinTh) | 0.02 | Indica el número de diferentes flujos de datos que se transmitirán simultáneamente en los mismos recursos de tiempo y frecuencia |
| Indicador de clasificación (lambdaMaxTh) | 0.2 | Indica el número de diferentes flujos de datos que se transmitirán simultáneamente en los mismos recursos de tiempo y frecuencia |
| Indicador de clasificación (lambdaRatioTh) | 20 | Indica el número de diferentes flujos de datos que se transmitirán simultáneamente en los mismos recursos de tiempo y frecuencia |

Fuente: Investigador.

En la sección final del archivo *config_channel.xml* se detallan estos parámetros:

```

<!-- Target bler used to compute feedback -->
<parameter name="targetBler" type="double" value="0.001"/>
<!-- Rank indicator tracefile -->
<parameter name="lambdaMinTh" type="double" value="0.02"/>
<parameter name="lambdaMaxTh" type="double" value="0.2"/>
<parameter name="lambdaRatioTh" type="double" value="20"/>

```

Configuraciones de modelo de propagación

Los modelos de propagación disponibles en OMNeT++ para redes LTE son: *FreeSpaceModel*, *TwoRayGroundModel*, *RiceModel*, *RayleighModel*,

NakagamiModel y *LogNormalShadowingModel*. La simulación de la red LTE utiliza un modelo de propagación *FreeSpaceModel*.

```
*.channelControl.propagationModel = "FreeSpaceModel"
```

Configuraciones de los eNodeBs

Los *eNodeBs* cumplen el papel crucial de dar la zona de cobertura de la red 4G, administrar los recursos de radio, manejar de manera adecuada el tráfico de red, sincronización, mediciones sobre el canal inalámbrico, etc.

Para que todas estas funcionen correctamente, uno de los módulos principales es el *deployer* encargado de desplegar y manejar todas las características antes mencionadas utilizando toda la información que se genera en la celda. Este módulo obtiene constantemente la posición de cada nodo enlazado a su respectivo *eNodeB* y envía un mensaje broadcast para realizar ese proceso, sus configuraciones según las especificaciones del protocolo son las siguientes:

```
**deployer.positionUpdateInterval = 0.001s  
**deployer.broadcastMessageInterval = 1s
```

También se especifican el número de bloques de recursos de radio (*numRb**), el número de subportadoras (*rby**), número de símbolos OFDM (*rbx**), el número de símbolos de señalización (*signal**) y el número de bandas, todos estos para el canal de subida UL y el de bajada DL.

```
**deployer.numRbDl = 6  
**deployer.numRbUl = 6  
**deployer.rbyDl = 12  
**deployer.rbyUl = 12  
**deployer.rbxDl = 7  
**deployer.rbxUl = 7  
**deployer.signalDl = 1  
**deployer.signalUl = 1  
**deployer.numBands = 1
```

El tipo de asignación de recursos tiene el tipo de conversión *localized*, mientras el tipo de modulación y codificación adaptiva es automática.

```
**rbAllocationType = "localized"  
**mac.amcMode = "AUTO"
```

En los *eNodeBs* el canal de retroalimentación se utilizan en todas las bandas en caso de utilizarse varias al mismo tiempo, además los *eNodeBs* tienen la capacidad de realizar tres retransmisiones para solicitudes automáticas de repetición híbrida HARQ.

4.8 Análisis de resultados

Una vez realizadas las configuraciones de los archivos de inicialización *omnetpp.ini*, para ejecutar una simulación se da clic derecho sobre el archivo mencionado, luego *Run As | 1 OMNeT++ Simulation*, como se ilustra en la Fig. 82:

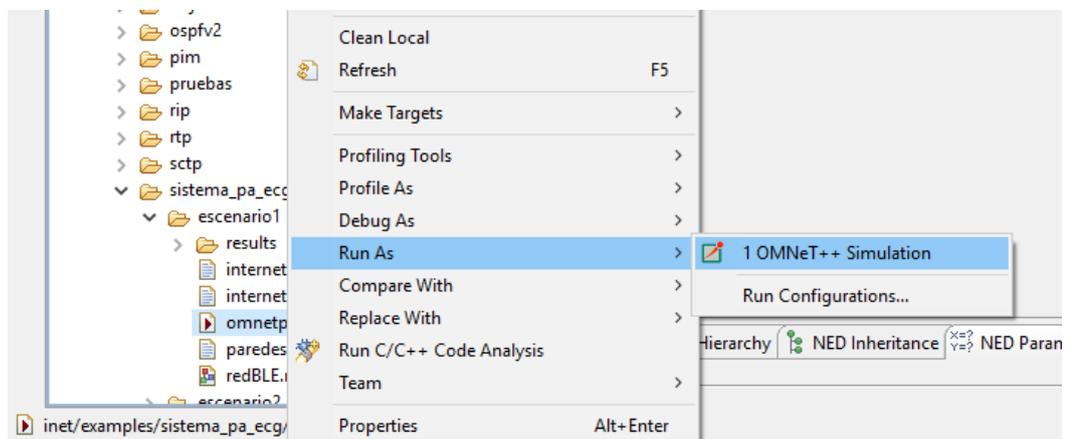


Fig. 82. Ejecución de la simulación desde archivo *omnetpp.ini*.

Fuente: Investigador.

Existen diferentes entornos gráficos de simulación, entre ellos: *Tkenv*, *Qtenv* y *Cmdenv*. Sin duda el mejor entorno gráfico es el *Qtenv* ya que permite la ejecución de simulaciones de una manera más rápida. *Qtenv* soporta ejecución de simulación interactiva, animación, inspección, rastreo y depuración. Además del desarrollo y verificación de modelos, *Qtenv* también es útil para fines educativos, ya que permite al usuario obtener un cuadro detallado del estado y la historia de la simulación en cualquier punto de su ejecución [63]. Para configurar este ambiente de simulación en el menú de la Fig. 82 se selecciona la opción *Run As | Run Configurations...* aparecerá una ventana como se indica en la Fig. 83, donde se selecciona el ambiente *Qtenv*.

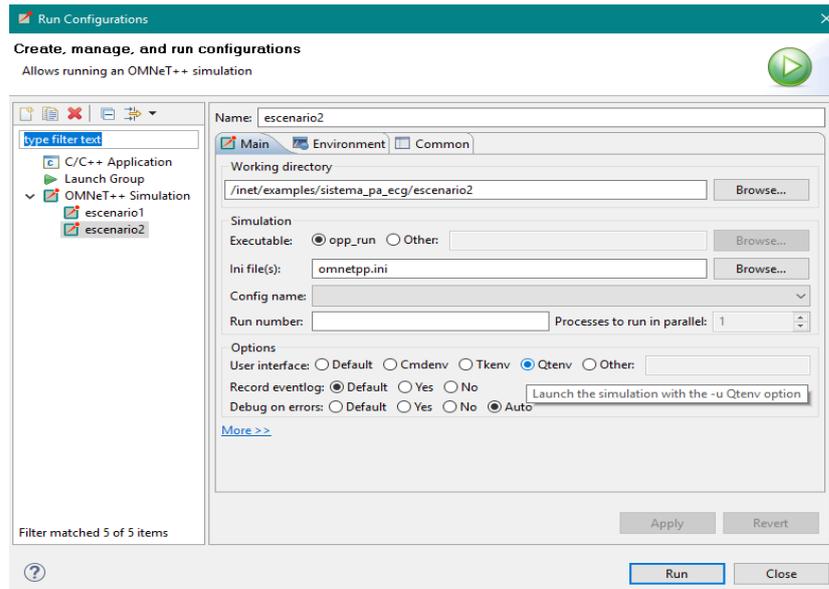


Fig. 83. Ventana de configuración del ambiente de simulación.

Fuente: Investigador.

Al dar clic en *Run*, se abre el ambiente de simulación donde se ejecutarán cada uno de los eventos discretos según las configuraciones realizadas. Según la simulación ejecutada se abrirá un cuadro de diálogo como el de la Fig. 84 donde se seleccionará la configuración del archivo *omnetpp.ini* que se desea simular.

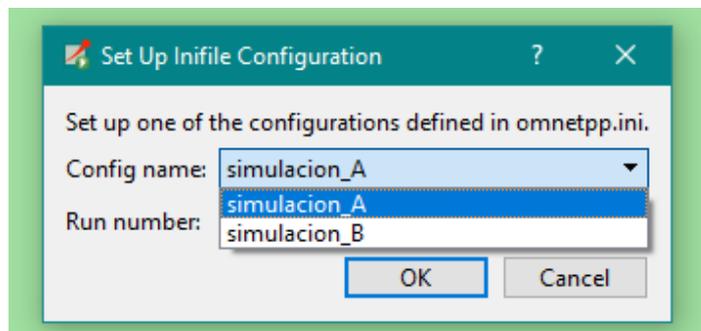


Fig. 84. Selección de configuración desde ambiente de simulación.

Fuente: Investigador.

En la Fig. 85 se muestran los principales componentes del ambiente de simulación *Qtenv*:

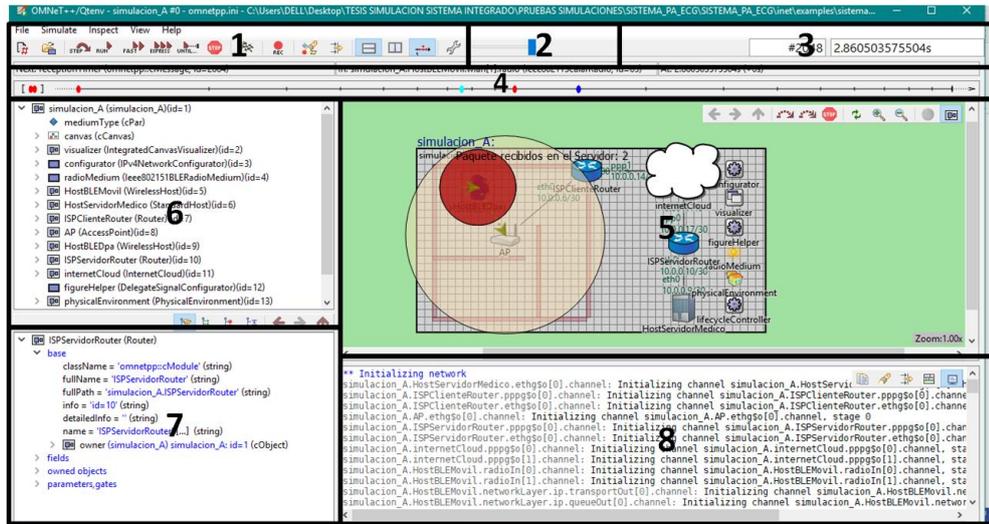


Fig. 85. Ambiente de ejecución de simulaciones.

Fuente: Investigador.

Cada componente de detalla a continuación:

1. **Barra de herramientas:** contiene las principales funciones de ejecución como iniciar, parar, paso a paso, etc.
2. **Velocidad de animación:** es una barra en forma de *slider* que permite aumentar la velocidad de los eventos discretos.
3. **Número de evento y tiempo de simulación:** muestran el número de evento discreto de la simulación y el tiempo en el que se ejecutan.
4. **Línea de tiempo:** en esta línea se presentan una serie de eventos futuros en una escala logarítmica de tiempo.
5. **Pantalla de red:** muestra la topología de red descrita en el archivo “.ned” y donde se dan lugar las animaciones.
6. **Navegador de objetos:** muestra de manera jerárquica los módulos y submódulos de cada componente de la red.
7. **Inspector de objetos:** muestra las propiedades y contenidos de los objetos.
8. **Visor de registro:** muestra los paquetes o mensajes enviados entre módulos, o registro de mensajes de salida durante la simulación

En la barra de herramientas los botones más utilizados son:

1. **Run:** corre la simulación mostrando cada uno de los eventos discretos y sus animaciones.

2. **Fast:** ejecuta 10 veces más rápido la simulación, pero la animación de mensajes se deshabilita.
3. **Express:** la simulación se ejecuta de manera aún más rápida que en el modo *Fast*, el problema está en que los trazos y animaciones no se ejecutan.
4. **Stop:** para o pausa la simulación.

El ambiente va mostrando cada uno de los pasos y eventos discretos que se dan en la simulación, se observa el intercambio de paquetes, mensajes de acuse de recibido, mensajes broadcast, trazos de movimiento, estado de componentes de la red, etc. En el Visor de registros, se detallan claramente las acciones, módulos y submódulos que intervienen en cada evento.

Al terminar la simulación los resultados se generan archivos en formato de vectores y escalares en la carpeta *Results* del directorio de trabajo del archivo de inicialización ejecutado. Estos archivos son de extensión “.vec” y “.sca”, al dar doble clic sobre cualquiera de ellos se generará un archivo “.anf” o archivo de análisis, que contiene la información de escalares y vectores que se han generado tras la simulación. La pestaña *Inputs* muestra los archivos de datos que pertenecen al archivo de análisis como se ilustra en la Fig. 86:

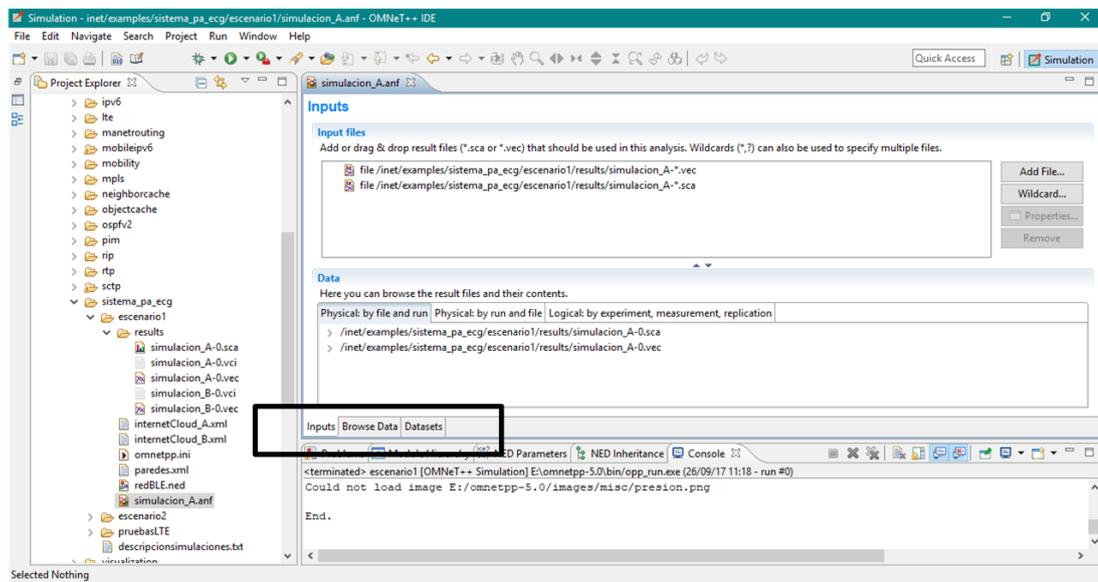


Fig. 86. Contenido del archivo “.anf” pestaña *Inputs*.

Fuente: Investigador.

En la pestaña *Browse Data* se pueden seleccionar uno a varios módulos o submódulos, con sus resultados escalares y vectoriales, para poder graficarlos. En la Fig. 87 se indica un ejemplo en el cual se selecciona el vector de resultados del submódulo *energyStorage* del módulo *HostBLEDpa* de la Simulación A:

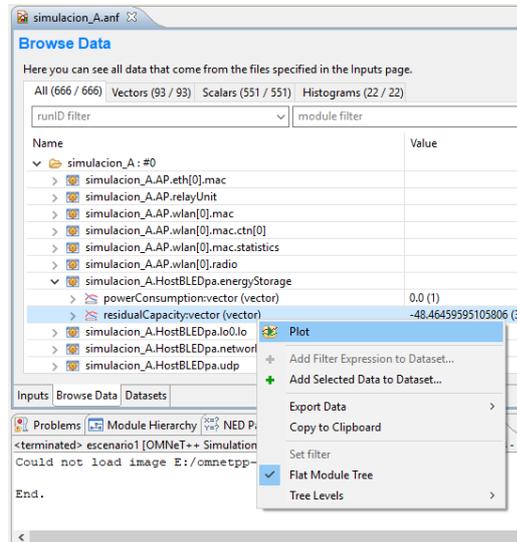


Fig. 87. Selección de resultados en pestaña *Browse Data* del archivo “.anf”.

Fuente: Investigador.

Al seleccionar *Plot* se muestra un gráfico como el de la Fig. 88, esta gráfica se abre en otra pestaña del archivo “.anf”. Es posible que en cada gráfica se muestren varios resultados, de diferentes módulos y submódulos para realizar comparaciones y análisis. Además la búsqueda de valores la realiza por medio de filtrado de nombres de módulos y componentes.

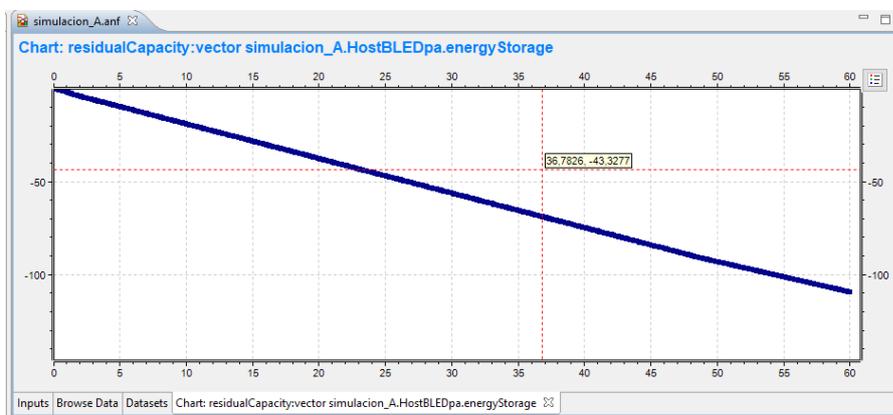


Fig. 88. Ejemplo de gráfica generada desde el archivo “.anf”.

Fuente: Investigador

4.8.1 Análisis Escenario 1: BLE y Wi-Fi (Simulación A)

A continuación se analiza la topología donde un cliente está conectado a la su red de área personal WPAN y a su red de área local WLAN, teniendo acceso al envío de datos hacia el servidor médico. En esta topología de red ilustrada en la Fig. 89 se analizan parámetros como consumo energético de DPA, pérdida de paquetes, retardo de extremo a extremo y relación de entrega.

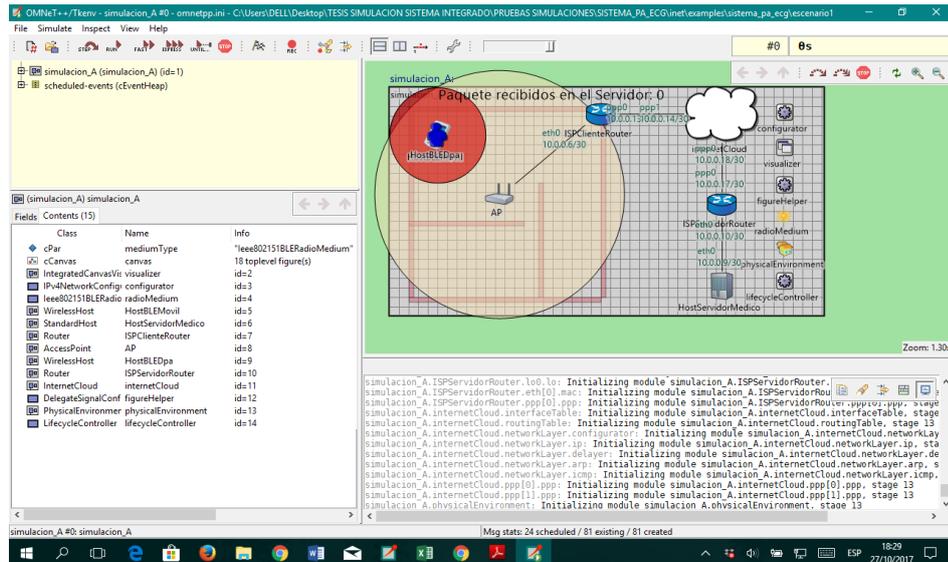


Fig. 89. Ambiente de simulación Escenario 1 – Simulación A.

Fuente: Investigador.

Consumo energético en el DPA (dispositivo electrónico de adquisición de datos)

El dispositivo electrónico de adquisición de datos utiliza dos baterías Ultrafire 18650 recargables de 2000 mAh y 3,7 V, en conjunto se cuenta con 4000 mAh y 7,4 V, equivalente a 53280 Joules.

Para analizar el consumo energético se deben tomar en cuenta el consumo de radio y el consumo general del dispositivo. En la simulación es posible hacer las mediciones de consumo de radio. En modo SLEEP, es decir cuando el dispositivo no transmite información se consumen 0,2249112323 Joules por minuto. La Fig. 90 muestra esos resultados:

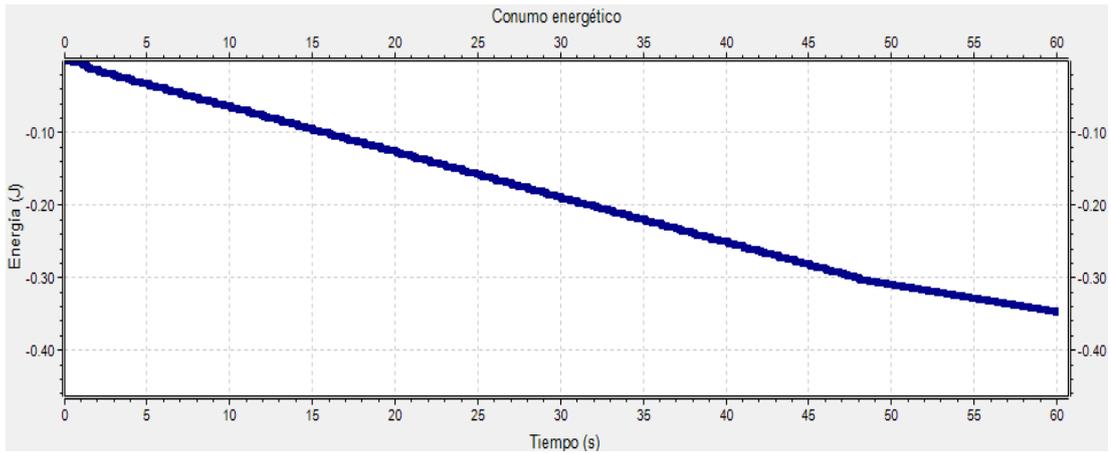


Fig. 90. Consumo energético en modo SLEEP.

Fuente: Investigador.

Esto se traduce a un consumo diario de:

$$0,2249112323 \frac{\text{Joules}}{\text{min}} * \frac{60 \text{ min}}{1 \text{ hora}} * \frac{24 \text{ horas}}{1 \text{ día}} = 323,8721 \frac{\text{Joules}}{\text{día}}$$

La Fig. 91 indica el consumo energético de una sola transmisión desde el DPA hacia el MÓVIL es decir en modo ACTIVO.

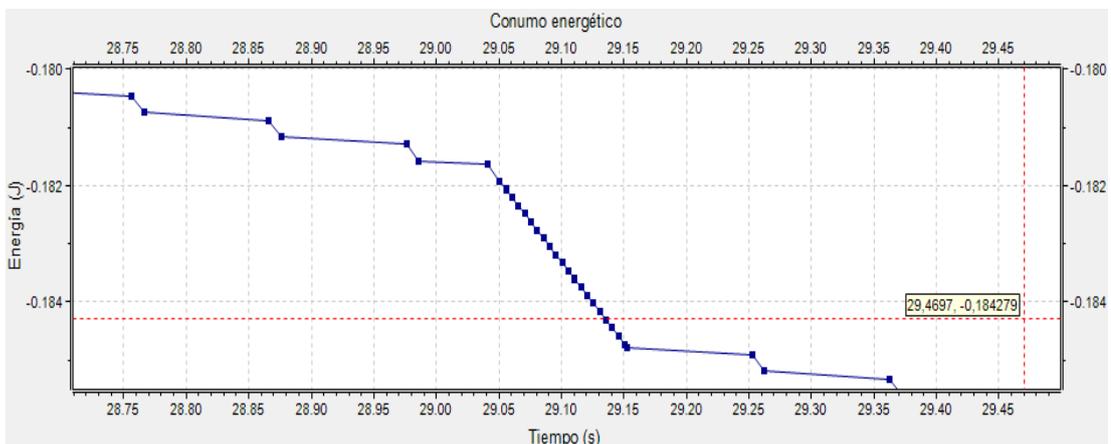


Fig. 91. Consumo energético del dispositivo al transmitir.

Fuente: Investigador.

Este lapso de transmisión dura 0,111951 segundos consumiendo 0,003140237 Joules por transmisión, por las 48 transmisiones diarias se consumirían 0,1507314 Joules por día. En total el consumo de radio se determina con la Ecuación 20:

$$\text{Consumo de radio} = \text{MODO SLEEP} + \text{MODO TRANSMITIENDO} \quad (20)$$

$$\begin{aligned} \text{Consumo de radio} &= 323,8721 \frac{\text{Joules}}{\text{día}} + 0,1507314 \frac{\text{Joules}}{\text{día}} \\ &= 323,9628 \frac{\text{Joules}}{\text{día}} \end{aligned}$$

Según la Tabla 20, en modo REPOSO se consumen 555 mW y al momento de activar los sensores, válvulas y cuando está TRANSMITIENDO se consume 3848 mW. El modo TRANSMITIENDO se ejecuta por aproximadamente 30 segundos cada media hora, en total al día 1440 segundos al día (24 minutos al día o 0,4 Horas al día) durante las 23,6 horas restantes se activa el modo REPOSO. Con la Ecuación 21 se expresan estos valores en Joules:

$$\text{Energía}(J) = \text{Potencia}(W) * \text{tiempo}(h) * 3600 \quad (21)$$

$$\text{MODO REPOSO}(23,6 \text{ horas}) = 0,555W * 23,6 h * 3600 = 47152,8 \text{ Joules}$$

$$\begin{aligned} \text{MODO TRANSMITIENDO}(0,4 \text{ horas}) &= 3,848W * 0,4 h * 3600 \\ &= 5541,12 \text{ Joules} \end{aligned}$$

$$\text{Consumo funcionamiento general} = 47152,8 \text{ Joules} + 5541,12 \text{ Joules}$$

$$= 52693,92 \frac{\text{Joules}}{\text{día}}$$

En consecuencia el dispositivo en consumo general y en consumo de radio gasta (Ecuación 22):

$$\text{Consumo total dispositivo} = \text{Consumo de radio} + \text{Consumo funcionamiento general} \quad (22)$$

$$\begin{aligned} \text{Consumo total dispositivo} &= 323,9628 \frac{\text{Joules}}{\text{día}} + 52693,92 \frac{\text{Joules}}{\text{día}} \\ &= 53017,88 \frac{\text{Joules}}{\text{día}} \end{aligned}$$

Se consumen 53017,88 Joules con el dispositivo encendido todo el día, con las baterías que proporcionan 53280 Joules se tendría una duración del dispositivo encendido de:

$$\frac{53280 \text{ Joules}}{53017,88 \frac{\text{Joules}}{\text{día}}} = 1,0049 \text{ días}$$

Pero se debe tomar en cuenta que las baterías de litio no deben descargarse totalmente, es aconsejable que su nivel de batería no llegue a valores inferiores de 0,9 V o el 24,32%, que según la capacidad de las baterías utilizadas en el dispositivo sería 12960 Joules por ambas baterías. En total es posible consumir 40320 Joules de los 53280 Joules disponibles, por lo tanto la duración de las baterías sería:

$$\frac{40320 \text{ Joules}}{53017,88 \frac{\text{Joules}}{\text{día}}} = 0,76 \text{ días}$$

El dispositivo logra permanecer encendido durante 0,76 días ininterrumpidamente (equivalente 18 horas y 15 minutos).

Pérdida de paquetes

La Tabla 26 indica la cantidad de paquetes de información que cada componente de red ha recibido y ha transmitido:

Tabla 26. Pérdida de paquetes Escenario 1 - Simulación A.

| Conexión | Número de Paquetes Transmitidos | Número de Paquetes Recibidos | Número de paquetes Perdidos |
|---|--|-------------------------------------|------------------------------------|
| HostBLEDpa – HostBLEMovil | 48 | 47 | 1 |
| HostBLEMovil – AP | 47 | 47 | 0 |
| AP - ISPClienteRouter | 47 | 47 | 0 |
| ISPClienteRouter – internetCloud | 47 | 47 | 0 |
| internetCloud – ISPServidorRouter | 47 | 46 | 1 |
| ISPServidorRouter - HostServidorMedico | 46 | 46 | 0 |

Fuente: Investigador.

Los resultados de la Tabla 26 muestran que existe la pérdida de un paquete de información en el trayecto de intercambio de información entre el DPA y el MÓVIL, por lo general las pérdidas de información en la red WPAN entre estos dos dispositivos se dará únicamente si se pierde el rango de cobertura de la señal Bluetooth o por obstáculos físicos. El uso del protocolo ACK entre el DPA y el MÓVIL asegura que la mayoría de los paquetes sean recibidos y que no tengan errores, por lo cual es

necesario ser configurado en ambos extremos. Otro paquete se pierde al atravesar el módulo que simula la conexión hacia internet (*internetCloud*).

En una de las simulaciones ejecutadas se pudo observar los dos paquetes que no llegaron al servidor médico, como se observa en la Fig. 92:

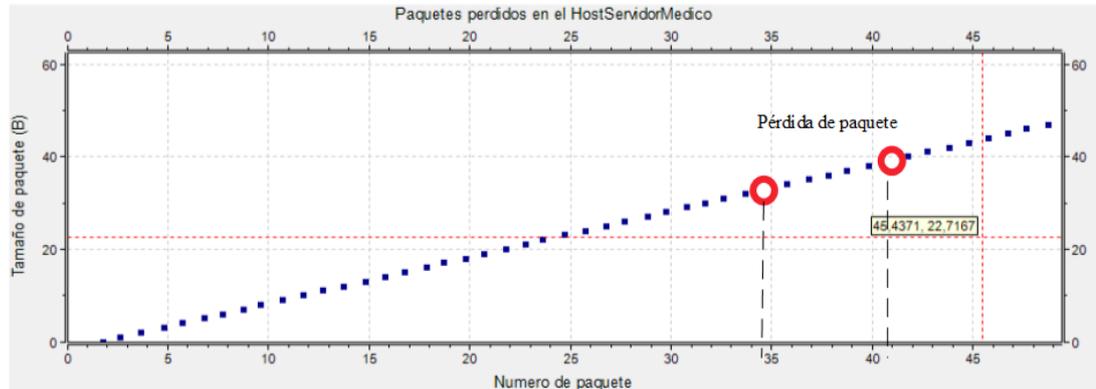


Fig. 92. Pérdida de paquetes de información Escenario 1 - Simulación A.

Fuente: Investigador.

Relación de entrega

Este valor indica el porcentaje de paquetes que llegan correctamente al destinatario:

$$\begin{aligned}
 \text{Relación de entrega} &= \frac{\text{Paquetes recibidos exitosamente}}{\text{Paquetes enviados}} * 100\% \\
 &= \frac{46}{48} * 100\% = 95,83\%
 \end{aligned}$$

El 95,83% de los paquetes son entregados en la simulación A del escenario 1, porcentaje idóneo que indica que las configuraciones programadas en la red aseguran que la información llegue correctamente al doctor que realizará el control médico.

Retardo de extremo a extremo

En la Fig. 93 se ilustran los tiempos de envío de información desde el DPA (azul) y el tiempo en el que llegan los paquetes al servidor (rojo):

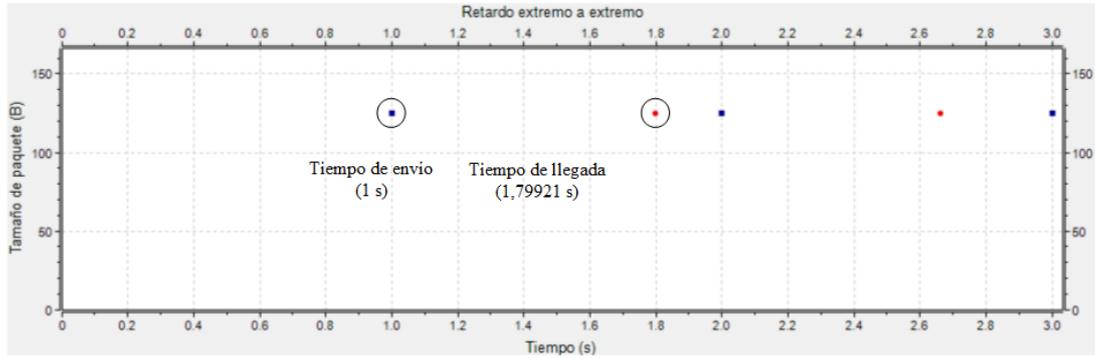


Fig. 93. Retardo extremo a extremo Escenario 1 - Simulación A.

Fuente: Investigador.

La conexión Bluetooth tiene un retardo promedio de 544 ms. La Fig. 94 presenta un resumen de los datos obtenidos de los retardos generados desde que la información parte del DPA y llega al servidor médico:



Fig. 94. Resultados de retardo extremo a extremo Escenario 1 - Simulación A.

Fuente: Investigador.

Un retardo promedio de 750 ms es aceptable para la aplicación, la información llegará casi de manera inmediata al doctor de cabecera y el control de la salud del usuario se realizará de manera oportuna.

Tabla 27. Pérdida de paquetes Escenario 1 - Simulación B.

| # Usuarios | Número de Paquetes Transmitidos | Número de Paquetes Recibidos | Número de paquetes Perdidos Simulación B | Número de paquetes Perdidos conexión BLE | Total de paquetes perdidos |
|------------|---------------------------------|------------------------------|--|--|----------------------------|
| 10 | 480 | 466 | 4 | 10 | 14 |
| 50 | 2400 | 2323 | 27 | 50 | 77 |
| 100 | 4800 | 4638 | 62 | 100 | 162 |
| 150 | 7200 | 6961 | 89 | 150 | 239 |
| 200 | 9600 | 9299 | 101 | 200 | 301 |

Fuente: Investigador.

Relación de entrega

Este valor indica el porcentaje de paquetes que llegan correctamente al destinatario, la Tabla 28 muestra el porcentaje o relación de entrega de paquetes según la cantidad de usuarios:

Tabla 28. Relación de entrega Escenario 2 - Simulación B.

| # Usuarios | Relación de Entrega | Porcentaje paquetes perdidos | Paquetes perdidos por usuario al día (de 48 enviados) |
|------------|---------------------|------------------------------|---|
| 10 | 97,08% | 2,92% | 1,4 |
| 50 | 96,79% | 3,21% | 1,54 |
| 100 | 96,63% | 3,37% | 1,62 |
| 150 | 96,68% | 3,32% | 1,59 |
| 200 | 96,86% | 3,14% | 1,51 |

Fuente: Investigador.

Los resultados obtenidos muestran un mínimo de pérdidas de paquetes de 2,92% y un máximo de pérdidas de 3,37%, incluso no se pierden los dos paquetes esperados según la simulación A, por lo tanto, se comprueba que no siempre se pierde un paquete diario por usuario en el módulo *internetCloud*. La pérdida de paquetes no es proporcional al número de clientes en la red y se mantiene estable tal y como se puede observar en la Fig. 96, aun cuando todos los nodos ejecutan un envío de información al mismo tiempo.

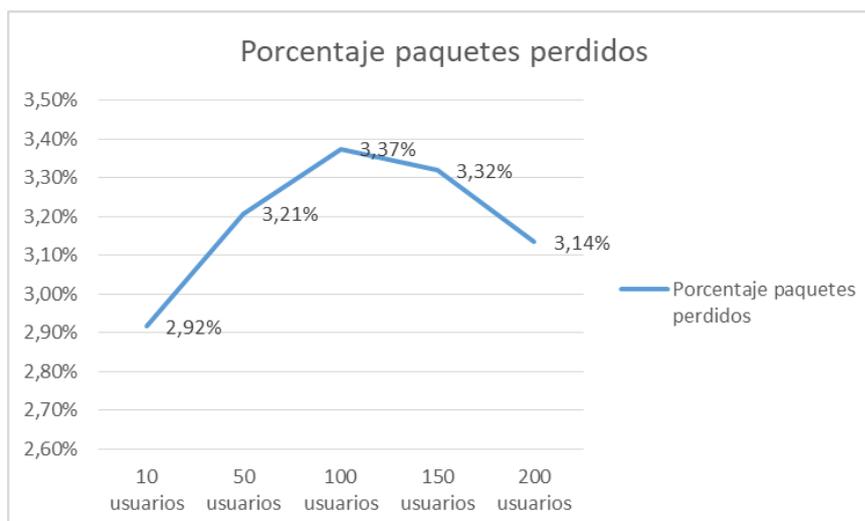


Fig. 96. Porcentaje de pérdidas de información Escenario 1 - Simulación B.

Fuente: Investigador.

En los resultados anteriores se analizó las pérdidas cuando todos los usuarios envían información al mismo tiempo; si existen 150 usuarios, los 150 paquetes de información se envían al mismo instante (CASO A). A continuación, se realiza una comparación de los resultados del CASO A con otro caso en el cual los usuarios envía aleatoriamente sus 48 paquetes de información diarios hacia el servidor médico (CASO B). La Tabla 29 detalla una comparación entre el CASO A y CASO B:

Tabla 29. Comparación de Casos de envío de datos Escenario 1 - Simulación B.

| # Usuarios | Número de Paquetes Recibidos en el Servidor (CASO A) | Relación de Entrega (CASO A) | Número de Paquetes Recibidos en el Servidor (CASO B) | Relación de Entrega (CASO B) | Balance (B-A) |
|------------|--|------------------------------|--|------------------------------|---------------|
| 10 | 466 | 97,08% | 467 | 97,29% | Aumenta |
| 50 | 2323 | 96,79% | 2323 | 96,79% | Se mantiene |
| 100 | 4638 | 96,63% | 4643 | 96,73% | Aumenta |
| 150 | 6961 | 96,68% | 6971 | 96,82% | Aumenta |
| 200 | 9299 | 96,86% | 9286 | 96,73% | Disminuye |

Fuente: Investigador.

En varios casos el porcentaje de paquetes entregados correctamente al servidor médico aumenta cuando se transmite la información en diferentes momentos. En otros casos esta relación de entrega disminuye, lo que indica que es irrelevante que los paquetes se generen al mismo tiempo o en distintos tiempos.

Retardo de extremo a extremo

La simulación B arroja resultados de tiempo de conexión desde el router de la red de área local de cada uno de los usuarios hasta llegar al host del servidor médico, pero para tener un panorama claro del retardo total se le añadió el retardo promedio de conexión inalámbrica de los dispositivos del usuario que es de 544 ms. En la Fig. 97 se exponen los tiempos de envío y llegada con 10 usuarios enviando información al servidor:

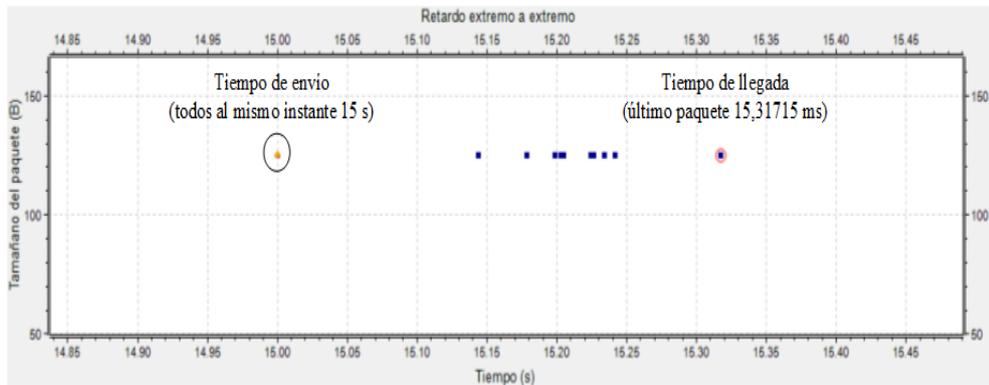


Fig. 97. Retardo extremo a extremo 10 usuarios Escenario 1 - Simulación B.

Fuente: Investigador.

El ejemplo se observa que el paquete enviado a los 15 segundos se recibe en el host del servidor médico a los 15,31715 segundos, para los demás envíos se producen tiempos similares, si el número de usuarios aumenta el retardo de extremo a extremo no aumenta considerablemente. La Fig. 98 presenta un resumen de los datos obtenidos de los retardos generados incluyendo el retardo de la conexión bluetooth:



Fig. 98. Resultados de retardo extremo a extremo Escenario 1 - Simulación B.

Fuente: Investigador.

Para 150 usuarios el retardo promedio es de 908,24 ms cuando el tráfico de usuarios es alto. Al hacer una comparación entre la tasa de retardo al estar conectados 10 usuarios y 200 usuarios se deduce que el retardo no sobrepasa los 64 ms. Lo cual indica que no habrá un incremento considerable de retardos si se aumentan usuarios al sistema.

Throughput de red

El rendimiento de red se ha medido desde el lado del servidor médico, donde se debe contar con un ancho de banda adecuado para procesar y administrar adecuadamente toda la información enviada por todos los usuarios del sistema. Se debe tomar en cuenta que este rendimiento toma únicamente el tráfico de datos del sistema de presión arterial, la simulación no considera otro tipo de tráfico como descarga de archivos o solicitudes de navegación. Según la cantidad de usuarios el rendimiento de red varía ya que es proporcional al número de paquetes exitosos recibidos en el destino por unidad de tiempo. En la Fig. 99 se muestran los rendimientos obtenidos:

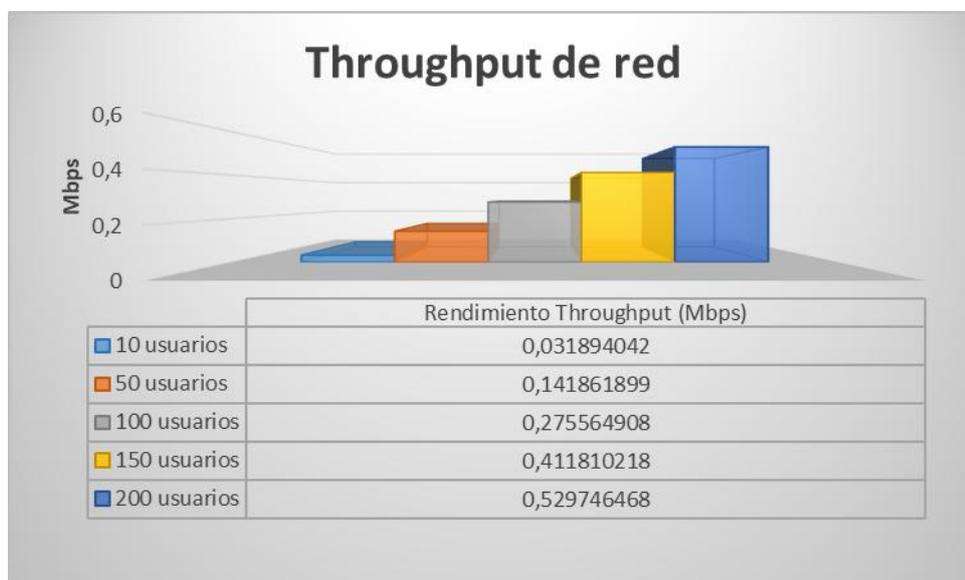


Fig. 99. Throughput de red Escenario 1 – Simulación B.

Fuente: Investigador.

Según aumentan el número de clientes el rendimiento de la red aumenta, el usuario deberá tener un ancho de banda mínimo dedicado para el sistema de presión arterial equivalente a 0,41181 Mbps para 150 usuarios.

4.8.3 Análisis Escenario 2: BLE y LTE (Simulación C)

Este análisis muestra el comportamiento de la comunicación de múltiples usuarios conectados a una red de datos móviles LTE y con acceso al envío de datos hacia el servidor médico. Esa topología mostrada en la Fig. 100 permite obtener resultados como pérdida de paquetes, relación de entrega, retardo de extremo a extremo, rendimiento de la red, calidad del canal, etc.

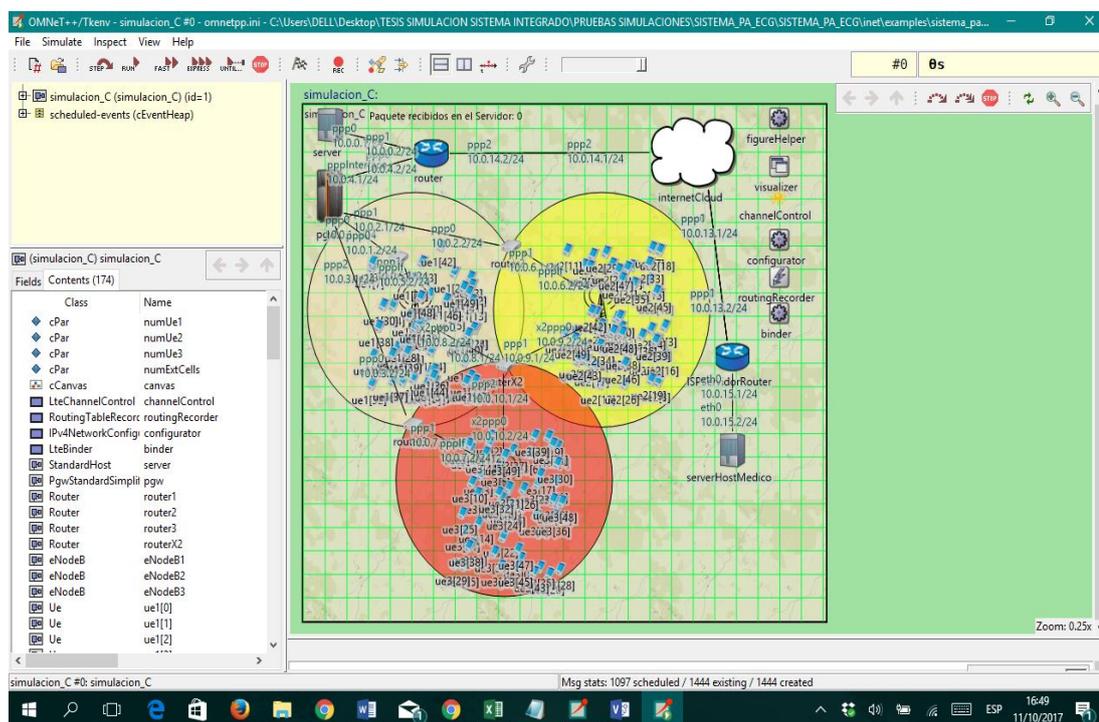


Fig. 100. Ambiente de simulación Escenario 2 – Simulación C.

Fuente: Investigador.

Pérdida de paquetes

La simulación C muestra un escenario en el cual los usuarios se reparten equitativamente en tres celdas. La Tabla 30 detalla la transferencia de paquetes entre los diferentes componentes de la red, el intercambio de información sigue el orden descendente de la tabla:

Tabla 30. Detalle de pérdida de paquetes en el Escenario 2 - Simulación C.

| # Usuarios | Conexión | Número de Paquetes Transmitidos | Número de Paquetes Recibidos | Número de paquetes Perdidos | Total Perdidos |
|------------|--------------------------------------|---------------------------------|------------------------------|-----------------------------|----------------|
| 10 | uEs - eNodeBs | 480 | 479 | 1 | 6 |
| | eNodeBs - internetCloud | 479 | 479 | 0 | |
| | internetCloud - ISPServidorRouter | 479 | 474 | 5 | |
| | ISPServidorRouter - serverHostMedico | 474 | 474 | 0 | |
| 50 | uEs - eNodeBs | 2400 | 2389 | 11 | 77 |
| | eNodeBs - internetCloud | 2389 | 2389 | 0 | |
| | internetCloud - ISPServidorRouter | 2389 | 2373 | 16 | |
| | ISPServidorRouter - serverHostMedico | 2373 | 2323 | 50 | |
| 100 | uEs - eNodeBs | 4800 | 4799 | 1 | 81 |
| | eNodeBs - internetCloud | 4799 | 4799 | 0 | |
| | internetCloud - ISPServidorRouter | 4799 | 4755 | 44 | |
| | ISPServidorRouter - serverHostMedico | 4755 | 4719 | 36 | |
| 150 | uEs - eNodeBs | 7200 | 7199 | 1 | 172 |
| | eNodeBs - internetCloud | 7199 | 7199 | 0 | |
| | internetCloud - ISPServidorRouter | 7199 | 7128 | 71 | |
| | ISPServidorRouter - serverHostMedico | 7128 | 7028 | 100 | |
| 200 | uEs - eNodeBs | 9600 | 9599 | 1 | 324 |
| | eNodeBs - internetCloud | 9599 | 9599 | 0 | |
| | internetCloud - ISPServidorRouter | 9599 | 9513 | 86 | |
| | ISPServidorRouter - serverHostMedico | 9513 | 9276 | 237 | |

Fuente: Investigador.

En este escenario la pérdida de paquetes se produce en el trayecto de los *Ues* hacia los *eNodeBs*, debido a la calidad de la señal o CQI es muy bajo en un determinado tiempo para un nodo de usuario o por el efecto de movilidad de los nodos. También se pierden paquetes en el enrutamiento de la nube debido a la probabilidad de pérdida de paquetes se le añadió al módulo.

El último trayecto en el que la información se pierde es entre el router del servidor médico y el host del servidor médico, la mayor cantidad de paquetes se pierden en este segmento de red. Los paquetes UDP llegan correctamente hasta la capa de transporte del host del servidor médico, pero se desechan varios paquetes cuando el proceso ejecutado en esta capa no logra determinar el tipo de aplicación al que pertenece el dato. Este problema se presenta debido a errores en bytes que posiblemente se generaron en la red de acceso de la red LTE. En total los paquetes perdidos incluyendo los de la conexión BLE son los que se muestran en la Tabla 31:

Tabla 31. Pérdida de paquetes Escenario 2 - Simulación C.

| # Usuarios | Número de Paquetes Transmitidos | Número de Paquetes Recibidos | Número de paquetes Perdidos Simulación C | Número de paquetes Perdidos conexión BLE | Total de paquetes perdidos |
|------------|---------------------------------|------------------------------|--|--|----------------------------|
| 10 | 480 | 464 | 6 | 10 | 16 |
| 50 | 2400 | 2273 | 77 | 50 | 127 |
| 100 | 4800 | 4619 | 81 | 100 | 181 |
| 150 | 7200 | 6878 | 172 | 150 | 322 |
| 200 | 9600 | 9076 | 324 | 200 | 524 |

Fuente: Investigador.

Relación de entrega

Tabla 32 muestra el porcentaje o relación de entrega de paquetes según la cantidad de usuarios:

Tabla 32. Relación de entrega Escenario 2 - Simulación C.

| # Usuarios | Relación de Entrega | Porcentaje paquetes perdidos | Paquetes perdidos por usuario al día (de 48 enviados) |
|------------|---------------------|------------------------------|---|
| 10 | 96,67% | 3,33% | 1,6 |
| 50 | 94,71% | 5,29% | 2,54 |
| 100 | 96,23% | 3,77% | 1,81 |
| 150 | 95,53% | 4,47% | 2,15 |
| 200 | 94,54% | 5,46% | 2,62 |

Fuente: Investigador.

Los resultados obtenidos muestran un mínimo de pérdidas de paquetes de 3,33% y un máximo de pérdidas de 5,29%. La pérdida de paquetes no es proporcional al número de clientes en la red y varía según las condiciones en las que se realiza la comunicación

entre el nodo de usuario y la red LTE aun cuando todos los nodos ejecutan un envío de información al mismo tiempo, esto se puede observar en la Fig. 101.

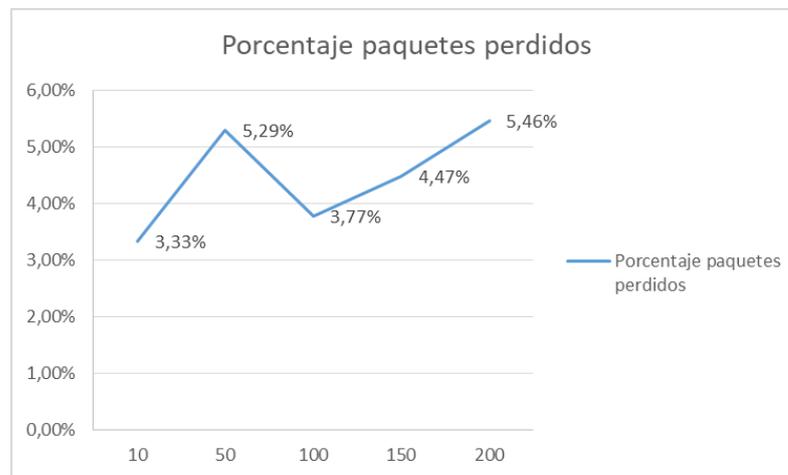


Fig. 101. Porcentaje de pérdidas de información Escenario 1 - Simulación C.

Fuente: Investigador.

Retardo de extremo a extremo

En la Fig. 102 se ilustra un ejemplo del envío de información desde el móvil del usuario (rojo) y el tiempo en el que llegan los paquetes al servidor (azul):

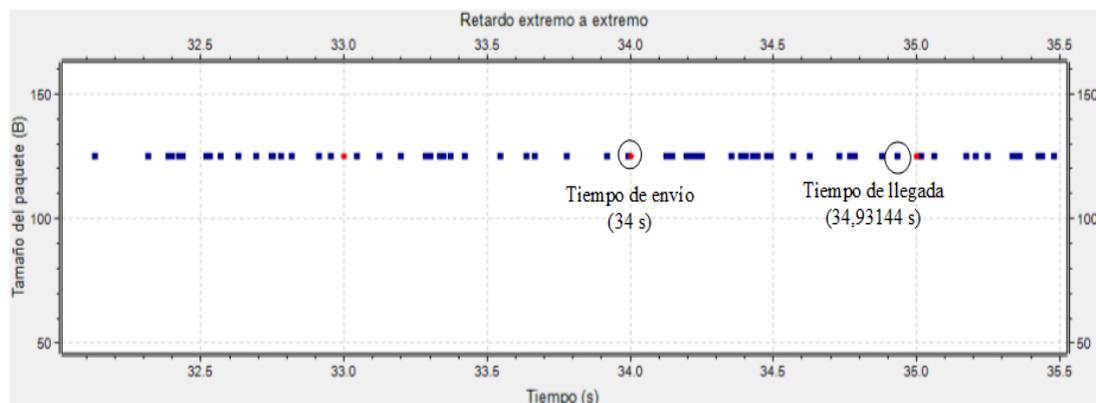


Fig. 102. Retardo extremo a extremo 150 usuarios Escenario 2 - Simulación C.

Fuente: Investigador.

En el ejemplo se observa que el paquete enviado a los 34 segundos se recibe en el host del servidor médico a los 34,93144 segundos. La Fig. 103 presenta un resumen de los datos obtenidos de los retardos generados incluyendo el retardo de la WPAN entre el MÓVIL y el DPA:



Fig. 103. Resultados retardo extremo a extremo Escenario 2 - Simulación C.

Fuente: Investigador.

Para 150 usuarios el retardo promedio es de 1,35 segundos cuando el tráfico de usuarios es alto, hasta llegar al servidor médico. En relación a simulaciones anteriores los retardos aumentan por efectos relacionados a la mayor complejidad de transporte de información en una red LTE.

Throughput de red

Según la cantidad de usuarios el rendimiento de red varía ya que es proporcional al número de paquetes exitosos recibidos en el destino por unidad de tiempo. En la Fig. 104 se muestran los rendimientos obtenidos:

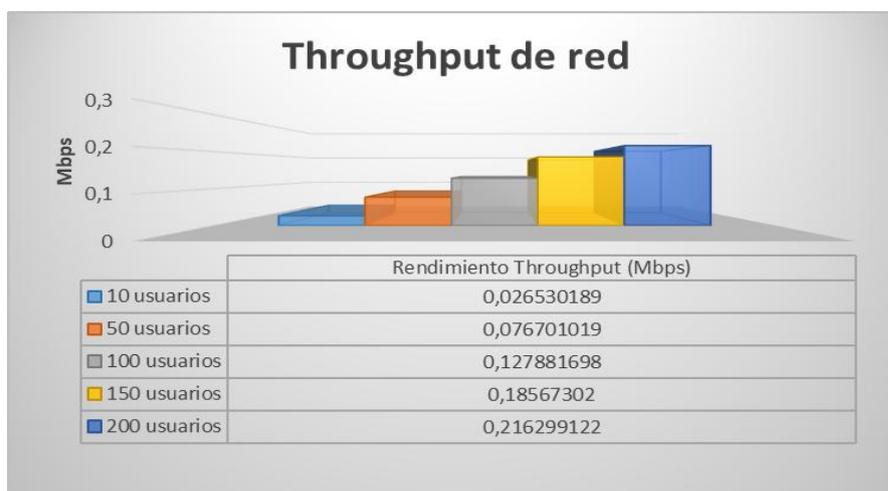


Fig. 104. Throughput de red Escenario 2 – Simulación C.

Fuente: Investigador.

Según aumentan el número de clientes el rendimiento de la red aumenta, el usuario deberá tener un ancho de banda mínimo dedicado para el sistema de presión arterial equivalente a 0,18567 Mbps para 150 usuarios. En relación a la simulación B este valor disminuye, debido a que el tiempo de retardo promedio aumenta y el throughput es inversamente proporcional a este.

Calidad del canal inalámbrico

La simulación C permite realizar un reporte y análisis del nivel la calidad de la señal en el enlace de subida y de bajada de todos los usuarios conectados a los *eNodeBs*. Los dispositivos de usuarios envían indicadores de CQI a las radio bases de LTE, valores que van del 1 al 15 y mientras más alto sea, quiere decir que el canal de comunicación es óptimo y que el nivel de señal recibida y la relación de señal a ruido son altos. En la Fig. 105 se muestran los CQI reportados para el enlace de bajada de 150 usuarios:

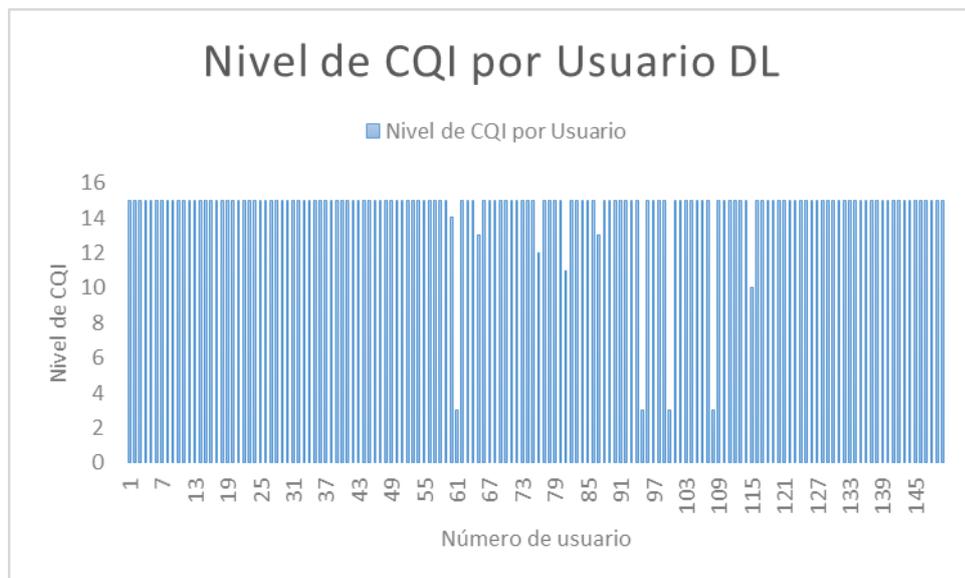


Fig. 105. CQI enlace de bajada de 150 usuarios Escenario 2 – Simulación C.

Fuente: Investigador.

Mientras mayor sea el CQI el esquema de modulación base y la tasa de código son mejorados, indicando que la conexión es buena entre el nodo de usuario y la radio estación, este efecto se asemeja a lo indicado en la Fig. 106:

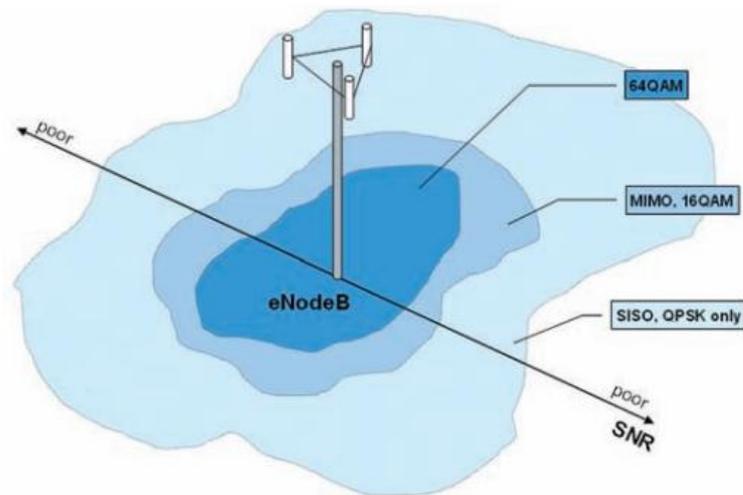


Fig. 106. Asignación de técnica de modulación según nivel de señal en LTE [82].

Para el enlace de subida el nivel de CQI es bajo en la mayoría de usuarios, entonces el *eNodeB* determina la mejor alternativa de los parámetros mencionados anteriormente para asegurar que la comunicación sea satisfactoria, los niveles tomados para 150 usuarios se muestran en la Fig. 107:

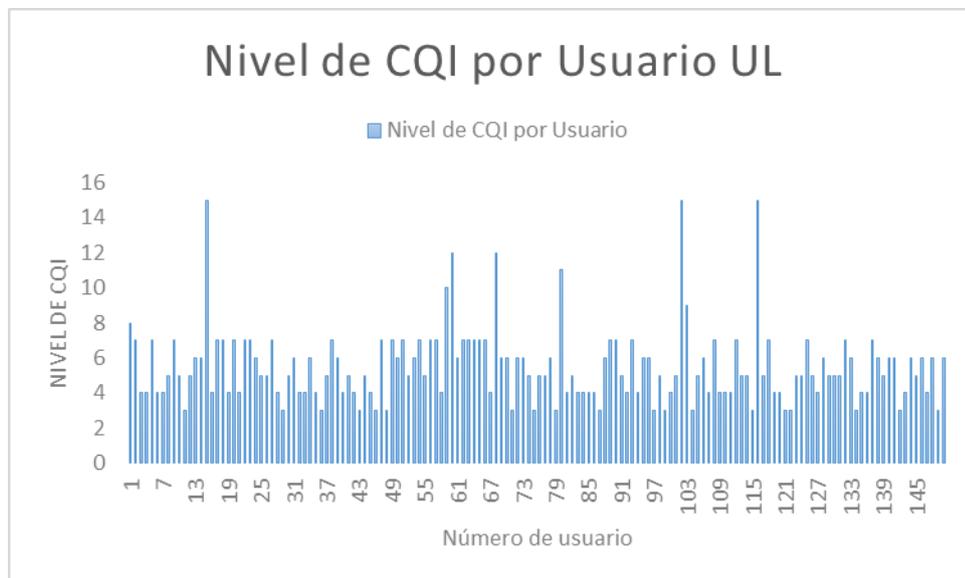


Fig. 107. CQI enlace de bajada de 150 usuarios Escenario 2 – Simulación C.

Fuente: Investigador.

Según los resultados un alto porcentaje de nodos reportan un CQI en el enlace de bajada de 15 seleccionándose una modulación 64-QAM con tasa de codificación de 0,9258, mientras para el enlace de subida existe una fluctuación entre un CQI de 3 y 4

aunque existen picos de 15 y 12, estos niveles indican una modulación QPSK con tasa de codificación entre 0,1885 y 0,3008. A pesar de que los mecanismos de planificación de radio de LTE pretendan evitar problemas de comunicación, algunos paquetes se pierden y otros llegan al destino con errores [83].

CAPÍTULO V

Conclusiones y Recomendaciones

5.1 Conclusiones

- El trabajo investigativo analiza el funcionamiento del prototipo, las tecnologías y protocolos BLE, Wi-Fi y LTE utilizados en el Sistema Integrado Ambulatorio de presión arterial para la comunicación y entrega de información entre los usuarios y el servidor médico. Determinando que el sistema es confiable en un 96%.
- Un paquete de información que viaja desde el dispositivo de adquisición de datos hasta el servidor médico, en el primer escenario que utiliza Wi-Fi llega en un tiempo promedio de 0,908 segundos. Mientras tanto, en el segundo escenario que emplea LTE, la información llega en aproximadamente en 1,35 segundos. Concluyendo que la latencia en red de área local inalámbrica es inferior que al utilizar una red LTE para el caso de estudio.
- El análisis de consumo energético del dispositivo ambulatorio electrónico para la adquisición de signos vitales reveló que el dispositivo logra estar encendido 18 horas y 15 minutos sin necesidad de ser cargado por una fuente externa, considerando el consumo en general del dispositivo y el consumo de radio.
- En el escenario que utiliza tecnología Wi-Fi y BLE, no existen interferencias en el medio inalámbrico compartido, esto se logra debido a protocolos como el IEEE 802.15.2 incluidos en BLE y que han sido desarrollados para permitir coexistencia entre redes WLAN y WPAN en un mismo entorno. En el

escenario LTE los niveles reportados de CQI indican un nivel de calidad bajo en el canal de subida entre el usuario final y la radio base, pero no existen pérdidas considerables de paquetes.

5.2 Recomendaciones

- Se requiere que en una futura implementación se utilice un protocolo como el Echo compatible con datos UDP, con esto el servidor podrá enviar una copia exacta del dato recibido al cliente para verificar que el paquete no se perdió y reducir el índice de pérdidas de información del estado de salud de los usuarios, de esta manera se logrará que el porcentaje de confiabilidad del sistema aumente.
- Mientras exista la disponibilidad de acceso a una red de área local inalámbrica para la conexión del Smartphone o Tablet hacia internet, el uso de datos móviles debe ser deshabilitado para que los datos lleguen con mayor rapidez al servidor médico.
- En necesario un algoritmo que permita apagar el dispositivo cuando su nivel de batería llegue a aproximadamente el 24,32%, realizando este proceso se logrará aumentar la vida útil de las baterías de litio.
- Se recomienda a los desarrolladores del sistema que en la WPAN entre el dispositivo móvil y el dispositivo de adquisición de datos se tome en cuenta las directivas del protocolo IEEE 802.15.2, el cual se maneja para evitar interferencias entre las WLAN y las WPAN.
- En el dispositivo de adquisición de datos que se ubicará en el brazo del paciente, es recomendable instalar un botón de emergencia que permita enviar a cualquier instante al doctor de cabecera un paquete de información de las variables físicas en caso de presentarse una alteración en la presión arterial o en el ritmo cardiaco.
- La plataforma OMNeT++ 5.0 es la más estable para simular redes LTE y es la que más se ajusta a los requerimientos de los módulos para evaluar este tipo de topologías, ya que las plataformas más actualizadas como OMNET++ 5.2 aún no cuentan con el soporte completo para protocolos de redes móviles.

- Existen en la web múltiples frameworks disponibles para ser descargados y utilizados en OMNeT++, entre ellos Mixim e INETMANET. Es recomendable utilizar la última versión del framework INET que en la actualidad ha absorbido y mejorado estos frameworks obsoletos.
- Para simulaciones con gran cantidad de eventos discretos se debe establecer una prioridad alta para la ejecución de la plataforma, esto es posible realizarlo desde el administrador de tareas y así reducir considerablemente el tiempo que se demora el ordenador en procesar la simulación.

BIBLIOGRAFÍA

- [1] Universo, El, «Pan American Health Organization,» 21 Enero 2014. [En línea]. Available: http://www.paho.org/ecu/index.php?option=com_content&view=article&id=1115:enero-21-2014&Itemid=356. [Último acceso: 4 Marzo 2017].
- [2] K. Jamil, Y. Mehmet R y F. Karami, «Performance evaluation of a Wireless Body Area sensor network for remote patient monitoring,» *Engineering in Medicine and Biology Society, 2008. EMBS 2008. 30th Annual International Conference of the IEEE*, pp. 1266 - 1269, 2008.
- [3] J. Averos, ESTUDIO DEL ESTÁNDAR IEEE 802.15.6 Y SIMULACIÓN DE LOS PARÁMETROS DE TRANSMISIÓN EN UNA RED DE ÁREA CORPORAL EN LA BANDA DE FRECUENCIA DE 2.4 GHz., Quito, 2017.
- [4] D. Tobón y N. Gaviria, «Análisis de métricas de calidad de servicio para la configuración del protocolo CSMA/CA en redes de sensores inalámbricas de área corporal,» *Ingeniería y Desarrollo Barranquilla*, vol. 30, nº 1, 2012.
- [5] D. M. Barakah y M. Ammad-Uddin, «A Survey of Challenges and Applications of Wireless Body Area Network (WBAN) and Role of a Virtual Doctor Server in Existing Architecture,» *Intelligent Systems, Modelling and Simulation (ISMS), 2012 Third International Conference on*, pp. 214 - 219, 2012.
- [6] M. Tirado, RED INALÁMBRICA DE SENSORES PARA EL MONITOREO DE LA CALIDAD DEL AGUA DE LA MICROCUENCA DEL RÍO QUERO, Ambato, 2015.
- [7] J. Andreu, «Google Libros,» [En línea]. Available: books.google.com.ec/books?isbn=8490030901. [Último acceso: 1 Abril 2017].

- [8] G.-Z. Yang, «Body Sensor Networks,» 2014. [En línea]. Available: https://books.google.com.ec/books?id=MoApBAAQBAJ&pg=PT317&lpg=PT317&dq=802.15.1+ieee+tdma&source=bl&ots=snvIawjFWx&sig=4wxkjT5tRR4j2PHvQwWvGZm2bnE&hl=es&sa=X&ved=0ahUKEwius5u3ls_TAhXL4yYKHdTZAGgQ6AEIMjAC#v=onepage&q=802.15.1%20ieee%20tdma&f=false. [Último acceso: 9 Julio 2017].
- [9] «Sparkfun,» [En línea]. Available: <https://learn.sparkfun.com/tutorials/bluetooth-basics>. [Último acceso: 7 Junio 2017].
- [10] C. Gomez, J. Oller y J. Paradells, «Overview and Evaluation of Bluetooth Low Energy: An Emerging Low-Power Wireless Technology,» *sensors*, vol. 12, n° 11734-11753, pp. 16-17, 29 Agosto 2012.
- [11] [En línea]. Available: <http://biling.us.es/proyectos/abreproy/11761/fichero/Volumen1%252F6-Cap%C3%ADtulo2+-+Redes+inal%C3%A1mblicas+de+%C3%A1rea+personal+%28WPAN%29.pdf>. [Último acceso: 9 Julio 2017].
- [12] I. C. Society, Part 15.1: Wireless medium access control (MAC) and physical layer (PHY) specifications for wireless personal area networks (WPAN's), New York, 2015, pp. 17-19.
- [13] J. Decuir, Bluetooth 4.0: Low Energy, 2013.
- [14] C. Beard y W. Stallings, BLUETOOTH AND IEEE 802.15, vol. 1st edition, 2016, pp. 12-13.
- [15] J. Ensworth, «Every Smart Phone is a Backscatter Reader: Modulated Backscatter Compatibility with Bluetooth 4.0 Low Energy (BLE) Devices,» *2015 IEEE International Conference on RFID (RFID)*, pp. 80-84, 2015.
- [16] C. Keuchul, J. Changsu, K. Jimbae, Y. Yongtak y H. Kijun, «Modeling and Analysis of Performance based on Bluetooth Low Energy,» p. 2.

- [17] M. Gast, 802.11 Wireless Networks: The Definitive Guide: The Definitive Guide, Segunda ed., O'Reilly, 2005.
- [18] I. Rodriguez, «VIKTORIVAN,» 9 Abril 2016. [En línea]. Available: <http://viktorivan.github.io/gfsk/dpsk/sdr/gnuradio/2016/04/09/dpsk-y-gfsk-en-gnu-radio.html>. [Último acceso: 18 Julio 2017].
- [19] B. Forouzan, Data Communications and Networking, Cuarta ed., MacGraw-Hill, 2007.
- [20] N. Mithila, Study and Analysis of Protocols of Wireless Sensor Network, Dhaka, 2012.
- [21] B. SIG, BLUETOOTH SPECIFICATION Version 4.2, 2014, pp. 115-116.
- [22] «RF Wireless World,» [En línea]. Available: <http://www.rfwireless-world.com/Tutorials/Bluetooth-MAC-layer.html>. [Último acceso: 19 Julio 2017].
- [23] J. Treurniet, C. Sarkar, V. Prasad y W. de Boer, «Energy Consumption and Latency in BLE Devices under Mutual Interference: An Experimental Study».
- [24] L. Jia, C. Canfeng, M. Yan y X. Yiung, «Energy Analysis of Device Discovery for Bluetooth Low Energy,» pp. 1-5.
- [25] K. Mikhaylov, «Simulation of Network-Level Performance for Bluetooth Low Energy,» pp. 1-6, 2014.
- [26] A. Escudero, «Estándares en Tecnologías Inalámbricas,» 2012, pp. 11-19.
- [27] «CCM,» [En línea]. Available: <http://es.ccm.net/contents/791-modos-de-funcionamiento-wifi-802-11-o-wi-fi>. [Último acceso: 2017 Septiembre 2017].
- [28] TRENDnet, «Punto de acceso inalámbrico Wireless N a 300Mbps TEW-638APB (A1.0R),» Manhattan, pp. 1-2.

- [29] J. Sánchez, «Telecomundo,» 12 Octubre 2016. [En línea]. Available: <https://telecomundo.wordpress.com/2016/10/16/modulaciones-m-arias-qam-y-qpsk/>. [Último acceso: 7 Septiembre 2017].
- [30] «Universidad de Colima,» [En línea]. Available: http://docente.ucol.mx/al980347/public_html/csma.htm. [Último acceso: 7 Septiembre 2017].
- [31] «Universidad Carlos III de Madrid - Departamento de Ingeniería Telemática,» [En línea]. Available: <http://www.it.uc3m.es/lpgonzal/protocolos/transporte.php>. [Último acceso: 8 Septiembre 2017].
- [32] «Hipermegared,» [En línea]. Available: <http://www.hipermegared.net/cuadro-comparativo-de-conexiones-a-internet-de-redes-moviles/>. [Último acceso: 12 Septiembre 2017].
- [33] W. Rendón, Evaluación de métodos para realizar mediciones de calidad de servicio VoIP en redes móviles de cuarta generación LTE en ambientes urbanos de la ciudad de Guayaquil, Guayaquil, 2017, pp. 34-40.
- [34] M. Tolsprump, Indoor Radio Planning, A practical Guide for GSM, DSC, UMTS, HSPA and LTE, WILEY, 2011.
- [35] ELUNIVERSO, «El Universo,» Febrero 2015. [En línea]. Available: <http://www.eluniverso.com/noticias/2015/02/26/nota/4592656/mas-espaciosbandas->. [Último acceso: 25 Julio 2017].
- [36] A. 3. L. T. Evolution, «System Overview, Product Development and Test Challenges,» 2009.
- [37] «Sharetechnote,» [En línea]. Available: http://www.sharetechnote.com/html/Handbook_LTE_RAType.html. [Último acceso: 12 Septiembre 2017].

- [38] L. Oñate, Estudio e investigación de las Femto Celdas para aumentar la cobertura y capacidad de las redes móviles, Guayaquil, 2014.
- [39] A. Piedra, Redes Móviles LTE: Herramienta de Planificación basada en Sistemas de Información Geográfica, Cuenca, 2013.
- [40] «Artiza Networks,» [En línea]. Available: http://www.artizanetworks.com/resources/white/han_pro.html. [Último acceso: 12 Septiembre 2017].
- [41] A. Fourie, «Linkedin,» [En línea]. Available: <https://www.linkedin.com/pulse/what-minimum-rssi-needed-3g-lte-andre-fourie/>. [Último acceso: 15 Septiembre 2017].
- [42] «LteWorld,» [En línea]. Available: <http://lteworld.org/specification/medium-access-control-mac>. [Último acceso: 12 Septiembre 2017].
- [43] H. Facchini, Rendimiento del estándar 802.11n - Estrategias de migración.
- [44] M. Stambaugh, «MobileDev & Design,» [En línea]. Available: <http://mobiledevdesign.com/standards/harq-process-boosts-lte-communications>. [Último acceso: 12 Septiembre 2017].
- [45] «LteBasics,» 21 Septiembre 2014. [En línea]. Available: <https://ltebasics.wordpress.com/2014/09/21/use-of-rank-indication-in-lte/>. [Último acceso: 12 Septiembre 2017].
- [46] «RF Wireless World,» 2012. [En línea]. Available: <http://www.rfwireless-world.com/Terminology/BER-vs-BLER-vs-DBLER.html>. [Último acceso: 13 Septiembre 2017].
- [47] F. Montealegre, Instalación, puesta en funcionamiento de tarjeta BER en generador de alta frecuencia y elaboración de prácticas de laboratorio de telecomunicaciones, 2007.

- [48] «Electrical Engineering,» [En línea]. Available: <https://electronics.stackexchange.com/questions/243286/what-is-the-core-concept-of-sinr-or-snr-model>. [Último acceso: 15 Septiembre 2017].
- [49] J. Drake, Ruidose interferencias: técnicas de reducción, Santander, 2005.
- [50] A. Varga, INET Framework for OMNeT++ Manual, 2016.
- [51] «Universidad Nacional de Quilmes,» [En línea]. Available: <http://iaci.unq.edu.ar/materias/telecomunicaciones/archivos/2008/introduccion.pdf>. [Último acceso: 15 Septiembre 2017].
- [52] E. Coimbra, «Coimbraweb,» 25 Agosto 2010. [En línea]. Available: http://www.coimbraweb.com/documentos/digital/4.3_transmision_digital_base.pdf. [Último acceso: 15 Septiembre 2017].
- [53] «telecomHall ES,» [En línea]. Available: <http://www.telecomhall.com/es/ques-es-isi-interferencia-entre-simbolos-en-lte.aspx>. [Último acceso: 18 Septiembre 2017].
- [54] «Wireless Communication,» [En línea]. Available: <http://www.wirelesscommunication.nl/reference/chaptr03/>. [Último acceso: 18 Septiembre 2017].
- [55] Cisco, «Cisco,» [En línea]. Available: <http://www.cisco.com/c/en/us/products/ios-nx-os-software/quality-of-service-qos/index.html>. [Último acceso: 19 Julio 2017].
- [56] S. Villacrés, Análisis del desempeño de una red WPAN Basado en el estándar IEEE 802.15.4 utilizando Network Simulator 2, Sangolquí, 2009.
- [57] J. García, «LA TELEMEDICINA Y EL BIENESTAR CIUDADANO,» *Revistaesalud.com*, vol. 2, nº 8, 2015.
- [58] J. Gásquez, F. Guillen, N. Novas y M. Berenguel , «DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE COMUNICACIÓN EN TIEMPO REAL Y TOLERANTE A FALLOS PARA MONITORIZACIÓN

MÉDICA EN VEHÍCULO EN MOVIMIENTO,» *XXV Jornadas de Automática*, 2004.

- [59] C. Abramovich y K. Pozzobon, DISEÑO E IMPLEMENTACIÓN DE UN PROTOTIPO DE SISTEMA REMOTO AUTOMATIZADO PARA LA MONITORIZACIÓN DE LA PRESIÓN ARTERIAL, Caracas, 2013.
- [60] «Taringa,» [En línea]. Available: <http://www.taringa.net/posts/salud-bienestar/1735720/Sabes-cual-es-tu-presion-arterial.html>. [Último acceso: 18 Abil 2017].
- [61] H. Laura, «Fundación española del Corazón,» [En línea]. Available: <http://www.fundaciondelcorazon.com/informacion-para-pacientes/metodos-diagnosticos/electrocardiograma.html>. [Último acceso: 13 Septiembre 2017].
- [62] J. Esteves, Simulation of Algorithms for Mobile Ad-Hoc Networks, Porto, 2015, pp. 35-44.
- [63] A. Varga, OMNET++ User Guide Version 5.0, 2016, pp. 1-3.
- [64] A. Varga, OMNET++ Simulation Manual Version 5.0, 2016, pp. 1-18.
- [65] V. Antonio, «SIMUlte,» 2015. [En línea]. Available: <http://simulte.com/>. [Último acceso: 2017 Septiembre 2017].
- [66] «Laleo.com,» [En línea]. Available: <http://www.laleo.com/brazalete-welchallyn-para-obeso-no-12-p-19676.html>. [Último acceso: 17 Agosto 2017].
- [67] «EVolta,» [En línea]. Available: http://www.evolta.cl/product_info.php?products_id=4281&osCsid=b8ea1d308509cb842fd6f34b73e979ba. [Último acceso: 11 Agosto 2017].
- [68] J. T. Company, Bluetooth 4.0 BLE module Datasheet, Shandong, 2014, pp. 4-6.

- [69] «Prometec,» [En línea]. Available: <http://www.prometec.net/consumos-arduino/>. [Último acceso: 12 Agosto 2017].
- [70] T. O. Technologies, Estudio sobre los requisitos técnicos que permitan caracterizar la cobertura con tecnología LTE necesaria para proporcionar determinados servicios de datos, pp. 9-15.
- [71] A. 5G, Internet de las cosas en América Latina, 2016, pp. 35-45.
- [72] A. Casanova, «Digimodes,» 25 Diciembre 2016. [En línea]. Available: <https://digimodes.wordpress.com/2016/12/25/seguridad-en-comunicaciones-lte-parte-i/>. [Último acceso: 8 Agosto 2017].
- [73] TP-Link, «TP-Link,» [En línea]. Available: http://www.tp-link.com/ar/products/details/cat-9_TL-WR841N.html#specifications. [Último acceso: 8 Agosto 2017].
- [74] S. Pita, «Fisterra,» [En línea]. Available: http://www.fisterra.com/mbe/investiga/conc_numerica/conc_numerica.asp. [Último acceso: 20 Septiembre 2017].
- [75] A. Jiménez, Entrenamiento personal: bases, fundamentos y aplicaciones, vol. Segunda Edición, Barcelona: INDE, 2005.
- [76] A. Varga, OMNeT ++ Installation Guide Version 5, 2016.
- [77] «Velocímetro web,» [En línea]. Available: <http://www.velocimetro.org/velocimetro/pages/main.jsp?c=t&t=c&d=l&e=lancyResults>. [Último acceso: 20 Junio 2017].
- [78] «Grupo TV Cable,» [En línea]. Available: <https://www.grupotvcable.com/medidor-de-velocidad/>. [Último acceso: 20 Junio 2017].
- [79] CNT, «Coorpración Nacional de Telecomunicaciones CNT EP,» [En línea]. Available: <https://www.cnt.gob.ec/internet/plan/fibra-optica-para-hogares/>. [Último acceso: 10 Agosto 2017].

- [80] Cisco, «Cisco,» [En línea]. Available: https://www.cisco.com/c/en/us/td/docs/wireless/asr_5000/20/P-GW/b_20_PGW_Admin/b_19_PGW_Admin_chapter_01.pdf. [Último acceso: 3 Septiembre 2017].
- [81] A. Viridis, «Omnet.org,» 2017. [En línea]. Available: https://summit.omnetpp.org/archive/2016/assets/pdf/OMNET-2016-Session_4-01-Presentation.pdf. [Último acceso: 4 Septiembre 2017].
- [82] R. & S. España, «LTE (Long Term Evolution): El siguiente nivel,» *Destino 4G*, pp. 82-86, Octubre 2010.
- [83] T. Bruggen, «Sistema de prueba R&S@TS8980 verifica los indicadores de calidad LTE CQI, PMI y RI,» *NOVEDADES*, n° 203/11, pp. 6-9.
- [84] D. Cruz, Evaluación de redes de sensores inalámbricos mediante el Simulador OMNeT++, Valencia, 2014.
- [85] M. Yuce y J. Khan, *Wireless Body Area Networks. Technology*, Ed. Boca Ratón: CRC, 2012.

ANEXOS

Anexo A

Código de programación del archivo redBLE.ned de descripción de red del Escenario 1 Simulación A y B.

```
//Paquete contenedor del archivo NED
package inet.examples.sistema_pa_ecg.escenario1;
//Importación de librerías y módulos
import inet.networklayer.configurator.ipv4.IPv4NetworkConfigurator;
import inet.node.ethernet.Eth100M;
import inet.node.inet.Router;
import inet.node.inet.StandardHost;
import inet.node.inet.WirelessHost;
import inet.node.internetcloud.InternetCloud;
import inet.node.wireless.AccessPoint;
import
inet.physicallayer.ieee80211.packetlevel.Ieee80211ScalarRadioMedium;
import inet.visualizer.integrated.IntegratedCanvasVisualizer;
import inet.common.misc.ThruputMeteringChannel;
import inet.common.figures.DelegateSignalConfigurator;
import inet.node.inet.INetworkNode;
import inet.physicallayer.contract.packetlevel.IRadioMedium;
import inet.environment.common.PhysicalEnvironment;
import inet.common.lifecycle.LifecycleController;

//Red de la Simulación A del Escenario 1
network simulacion_A
{
    //parámetros y señales a ser desplegadas en el ambiente de
    simulación
    parameters:
        string mediumType = default("IdealRadioMedium");

    @display("bgb=451.53848,238.46155;bgu=m;bgg=10,0,grey;bg1=2;i=backgr
    ound/streetmap");
        @figure[title](type=label; pos=0,-1; anchor=sw;
    color=darkblue);
        //declaración de señal para mostrar el número de paquetes
    que llegan al servidor médico mientras se ejecuta la simulación
        @figure[rcvdPkText](type=indicatorText; pos=50,8; anchor=w;
    font=,15; textFormat="Paquete recibidos en el Servidor: %g";
    initialValue=0);
        @statistic[rcvdPk](source=HostServidorMedico_rcvdPk;
    record=figure(count); targetFigure=rcvdPkText);
        @signal[HostServidorMedico_rcvdPk];

    @delegatesignal[rcvdPk](source=HostServidorMedico.udpApp[0].rcvdPk;
    target=HostServidorMedico_rcvdPk);

    types:
        //creación del canal de fibra óptica
        channel fiberline extends ThruputMeteringChannel
        {
            delay = 1us;
            datarate = 100Mbps;
            thruputDisplayFormat = "u";
        }
}
```

```

submodules:
    //inclusión de los submódulos de la red con su ubicación
    (x,y) y tipo de ícono
    visualizer: IntegratedCanvasVisualizer {
        parameters:
            @display("p=411.53848,63.076927");
    }
    configurator: IPv4NetworkConfigurator {
        parameters:
            assignDisjunctSubnetAddresses = false;
            @display("p=411.53848,26.153847;is=s");
    }
    radioMedium: <mediumType> like IRadioMedium {
        parameters:
            @display("p=410.76926,135.38461");
    }
    HostBLEMovil: WirelessHost {
        parameters:
            @display("p=50,50;i=device/pocketpc;r=50,black");
    }
    HostServidorMedico: StandardHost {
        parameters:
            @display("p=346.15387,208.46155;i=device/server");
    }
    ISPClienteRouter: Router {
        parameters:
            @display("p=220.00002,28.46154");
    }
    AP: AccessPoint {
        parameters:
            @display("p=114.24671,111.97088;r=130,wheat");
    }
    HostBLEdpa: WirelessHost {
        parameters:
            @display("p=50,50;i=abstract/person;r=50,red");
    }
    ISPServidorRouter: Router {
        parameters:
            @display("p=346.15387,123.07693");
    }

    internetCloud: InternetCloud {
        @display("p=346.15387,27.692308;is=v1");
    }

    figureHelper: DelegateSignalConfigurator {
        @display("p=411.53848,102.30769");
    }
    physicalEnvironment: PhysicalEnvironment {
        @display("p=411.53848,169.23077");
    }
    lifecycleController: LifecycleController {
        parameters:
            @display("p=411.53848,203.07693");
    }

connections:
    //conexiones cableadas entre las interfaces de los módulos
    HostServidorMedico.ethg++ <--> Eth100M <-->
    ISPServidorRouter.ethg++;

```



```

@display("p=228.735,452.4975,matrix,15,200,200;i=old/cloud");
}
ISPServidorRouter: Router {
    parameters:
        @display("p=2237.4766,128.67278");
}
internetCloud: InternetCloud {
    @display("p=224.50874,100.86625;is=1");
}

figureHelper: DelegateSignalConfigurator {
    @display("p=3731.5105,128.67278");
}

connections:
    //conexiones cableadas entre las interfaces de los módulos
    HostServidorMedico.ethg++ <--> Eth100M <-->
ISPServidorRouter.ethg++;
    ISPServidorRouter.pppg++ <--> fiberline <-->
internetCloud.pppg++;
    //ciclo para conectar N cantidad de interfaces ppp según el
    número de clientes
    for i=0..numClientes-1 {
        redHostCliente[i].pppg++ <--> fiberline <-->
internetCloud.pppg++;
    }
}

```

Anexo B

Código de programación del archivo de inicialización omnetpp.ini del Escenario

1 Simulación A y B.

```
#declaración de nombre de la configuración
[Config simulacion_A]
network = simulacion_A #red a la que pertenece
sim-time-limit = 60s #tiempo límite a simular

#####
####CONFIGURACIONES CAPA APLICACION###
#####
#Número de aplicaciones UDP, puerto a utilizar, tipo de aplicación
**.addDefaultRoutes = false #sin rutas por defecto
**.Host*.numUdpApps = 1
**.Host*.udpApp[0].localPort = 1000
**.Host*.udpApp[0].typename = "UDPBasicApp"
#destino de los mensajes UDP enviados, puerto y tamaño del mensaje
de capa de aplicación
**.HostBLEdpa.udpApp[0].destAddresses = "HostBLEMovil" #
**.HostBLEMovil.udpApp[0].destAddresses = "HostServidorMedico"
**.Host*.udpApp[0].destPort = 1000
**.Host*.udpApp[0].messageLength = 125B
#intervalo de envío del mensaje y tiempo inicio y paro de las
aplicaciones UDP
**.Host*.udpApp[0].sendInterval = 1s
**.HostBLEMovil.udpApp[0].startTime = 1.544s
**.HostBLEdpa.udpApp[0].startTime = 1s
**.Host*.udpApp[0].stopTime = 48.8s

#####
####CONFIGURACIONES DIRECCIONAMIENTO EN LA NUBE###
#####
#llamada al archivo xml que contiene las configuraciones del módulo
cloud
**.internetCloud.networkLayer.delayer.config =
xmlDoc("internetCloud_A.xml")

#####
####CONFIGURACIONES INTERFACES DE RED###
#####
#número de tarjetas inalámbricas en el móvil
*.HostBLEMovil.numRadios = 2
#MAC TDMA BLE para el DPA y MÓVIL
*.HostBLEdpa.wlan[0].typename = "WirelessNic"
*.HostBLEdpa.wlan[0].macType = "BMacLayer"
*.HostBLEMovil.wlan[0].typename = "WirelessNic"
*.HostBLEMovil.wlan[0].macType = "BMacLayer"
*.HostBLEMovil.wlan[1].typename = "Ieee80211Nic"
#Tasa de bits según la interfaz
*.HostBLEdpa.**.bitrate = 0.7Mbps
*.HostBLEMovil.wlan[0]**.bitrate = 0.7Mbps
*.HostBLEMovil.wlan[1]**.bitrate = 11Mbps

#####
####CONFIGURACIONES MAC###
```

```

#####
#Dirección MAC automática tamaño de cabecera y unidad máxima de
transferencia
*.HostBLE*.wlan[0].mac.address = "auto"
*.HostBLE*.wlan[0].mac.headerLength = 10B
*.HostBLE*.wlan[0].mac.mtu = 0B
#Ack entre el DPA y el MÓVIL
*.HostBLE*.wlan[0].mac.useMACAcks = true
*.HostBLE*.wlan[0].mac.slotDuration = 0.1s
*.HostBLE*.wlan[0].mac.checkInterval = 0.01s
*.HostBLE*.wlan[0].mac.queueLength = 20
*.HostBLE*.wlan[0].mac.animation = false #animación deshabilitada
*.HostBLE*.wlan[0].mac.macMaxFrameRetries = 3 #número máximo de
retransmisiones de MAC

#####
###CONFIGURACIONES CONSUMO ENERGÉTICO###
#####
#tipo de consumidor de energía y cantidad de consumo de energía
según el modo de operación de radio
*.HostBLE*.wlan[0].radio.energyConsumerType =
"StateBasedEnergyConsumer"
*.HostBLE*.wlan[0].radio.energyConsumer.offPowerConsumption = 0mW
#0mA radio
*.HostBLE*.wlan[0].radio.energyConsumer.sleepPowerConsumption =
1.32mW #0.4mA radio
*.HostBLE*.wlan[0].radio.energyConsumer.switchingPowerConsumption =
28.05mW #8.5mA radio
*.HostBLE*.wlan[0].radio.energyConsumer.receiverIdlePowerConsumption
= 28.05mW #8.5mA radio
*.HostBLE*.wlan[0].radio.energyConsumer.receiverBusyPowerConsumption
= 28.05mW #8.5mA radio
*.HostBLE*.wlan[0].radio.energyConsumer.receiverReceivingPowerConsum
ption = 28.05mW #8.5mA radio
*.HostBLE*.wlan[0].radio.energyConsumer.transmitterIdlePowerConsumpt
ion = 28.05mW #8.5mA radio
*.HostBLE*.wlan[0].radio.energyConsumer.transmitterTransmittingPower
Consumption = 28.05mW #8.5mA radio
# Modulo que detecta la energía residual en el DPA
*.HostBLE*.energyStorageType = "IdealEnergyStorage"
**.hasStatus = true

#####
###CONFIGURACIONES DE MOVILIDAD###
#####
#intervalo de actualización de posición de los nodos
**.updateInterval = 1s
#tipo de movilidad
**.HostBLE*.mobilityType = "TractorMobility"
#establecimiento de puntos máximos y mínimos para la movilidad de
los nodos
**.HostBLEDpa.mobility.x1 = 50m
**.HostBLEDpa.mobility.y1 = 50m
**.HostBLEDpa.mobility.x2 = 190m
**.HostBLEDpa.mobility.y2 = 190m
#establecimiento de puntos máximos y mínimos para la movilidad de
los nodos
**.HostBLEMovil.mobility.x1 = 50m
**.HostBLEMovil*.mobility.y1 = 50m
**.HostBLEMovil.mobility.x2 = 190m

```

```

**.HostBLEMovil.mobility.y2 = 190m
#número de filas para la trayectoria de TractorMobility
**.HostBLE*.mobility.rowCount = 2
#velocidad del nodo
**.HostBLE*.mobility.speed = 10mps

#####
####CONFIGURACIONES DE OBSTÁCULOS FÍSICOS Y ENTORNO###
#####
*.physicalEnvironment.config = xmldoc("paredes.xml")
*.radioMedium.obstacleLossType = "DielectricObstacleLoss"
*.physicalEnvironment.groundType = "FlatGround"
*.physicalEnvironment.ground.elevation = 0m

#####
####CONFIGURACIONES RADIO###
#####
#tipo de medio para el cálculo de propagación en BLE
*.mediumType = "Ieee802151BLERadioMedium"
#ruido de fondo y frecuencia portadora para BLE
*.radioMedium.backgroundNoise.power = -90dBm
*.radioMedium.mediumLimitCache.carrierFrequency = 2.45GHz
#parámetros característicos de BLE
*.HostBLE*.wlan[0].radioType = "Ieee802151BLERadio"
*.HostBLE*.wlan[0].radio.carrierFrequency = 2.45GHz
*.HostBLE*.wlan[0].radio.bandwidth = 2MHz
*.HostBLE*.wlan[0].radio.transmitter.power = 1mW
*.HostBLE*.wlan[0].radio.transmitter.preambleDuration = 10us
*.HostBLE*.wlan[0].radio.receiver.sensitivity = -87dBm
*.HostBLE*.wlan[0].radio.receiver.energyDetection = -87dBm
*.HostBLE*.wlan[0].radio.receiver.snirThreshold = -8dB
*.HostBLE*.wlan[0].radio.antennaType = "IsotropicAntenna"
*.HostBLE*.wlan[0].radio.antenna.numAntennas = 1
*.HostBLE*.wlan[0].radio.*.headerBitLength = 54b
#modelo de error en recepcion de paquetes
*.HostBLE*.wlan[0].radio.receiver.errorModelType = "ErrorModelBase"
#modulación GFSK en el trasmisor y receptor
*.HostBLE*.wlan[0].radio.transmitter.modulation = "GFSK"
*.HostBLE*.wlan[0].radio.receiver.modulation = "GFSK"
#técnica de modulación y modo de operación para la conexión Wifi
MÓVIL AP
*.HostBLEMovil.wlan[1].radio.receiver.modulation = "QPSK"
*.HostBLEMovil.wlan[1].radio.transmitter.modulation = "QPSK"
*.AP.wlan[0].radio.receiver.modulation = "QPSK"
*.AP.wlan[0].radio.transmitter.modulation = "QPSK"
*.AP.wlan[0].opMode = "b"

#####
####CONFIGURACIONES DE PÉRDIDAS Y MODELO DE PROPAGACIÓN###
#####
*.radioMedium.pathLossType = "RayleighFading" #pérdidas por
desvanecimiento
**.propagationType = "ConstantSpeedPropagation" #propagación
constante
**.analogModelType = "ScalarAnalogModel" #calcula la potencia de la
señal recibida
**.backgroundNoiseType = "IsotropicScalarBackgroundNoise" #incluye
ruido de fondo

```

```

#####
####CONFIGURACIONES DE ANIMACIONES###
#####
*.visualizer.sceneVisualizer.descriptionFigure = "title" #pone
título en el gráfico
*.visualizer.mediumVisualizer.signalPropagationUpdateInterval = 1s
#intervalo para visualizar propagacion de señal
*.visualizer.physicalLinkVisualizer.packetNameFilter =
"UDPBasicApp*" #visualiza el nombre de esos paquetes
*.visualizer.mediumVisualizer.displaySignals = false
*.visualizer.mobilityVisualizer.displayMovementTrail = true #muestra
un trazo de la trayectoria del nodo

#declaración de nombre de la configuración
[Config simulacion_B]
network = simulacion_B #red a la que pertenece
sim-time-limit = 60s #tiempo límite a simular

#####
####CONFIGURACIONES CAPA APLICACION###
#####
#Número de aplicaciones UDP, puerto a utilizar, tipo de aplicación
**.addDefaultRoutes = false
**.Host*.numUdpApps = 1
**.Host*.udpApp[0].localPort = 1000
**.Host*.udpApp[0].typename = "UDPBasicApp"
#destino de los mensajes UDP enviados, puerto y tamaño del mensaje
de capa de aplicación
**.redHostCliente[*].udpApp[0].destAddresses = "HostServidorMedico"
**.Host*.udpApp[0].destPort = 1000
**.Host*.udpApp[0].messageLength = 125B
#intervalo de envío del mensaje y tiempo de inicio y paro de las
aplicaciones UDP
**.Host*.udpApp[0].sendInterval = 1s
**.Host*.udpApp[0].startTime = 1s
**.Host*.udpApp[0].stopTime = 48.99s

#####
####CONFIGURACIONES DIRECCIONAMIENTO EN LA NUBE###
#####
#llamada al archivo xml que contiene las configuraciones del módulo
cloud
*.internetCloud.networkLayer.delayer.config =
xmlDoc("internetCloud_B.xml")

#####
####CONFIGURACIONES DE ANIMACIONES###
#####
*.visualizer.sceneVisualizer.descriptionFigure = "title" #pone
título en el gráfico
*.visualizer.mediumVisualizer.signalPropagationUpdateInterval =
100ns #intervalo para visualizar propagacion de señal
*.visualizer.physicalLinkVisualizer.packetNameFilter =
"UDPBasicApp*" #visualiza el nombre de esos paquetes

```

Anexo C

Código de programación del archivo de direccionamiento en la nube internetCloud_A.xml del Escenario 1 Simulación A.

```
<!-- Nombre del módulo y tráfico simétrico -->
<internetCloud symmetric="true">
  <parameters name="good">
    <!-- Nombre la entrada y salida de información, retardo generado,
    velocidad y probabilidad de paquetes descartados en el módulo -->
    <traffic src="ISPClienteRouter" dest="ISPServidorRouter"
    delay="20ms+truncnormal(200ms,60ms)"
    datarate="uniform(10Mbps,100Mbps)" drop="uniform(0,1) &lt; 0.01 " />
    <!--
    <traffic src="*" dest="*" delay="10ms+truncnormal(100ms,20ms)"
    datarate="uniform(10Mbps,100Mbps)" drop="0 &lt; 0" />
    -->
  </parameters>
</internetCloud>
```

Anexo D

Código de programación del archivo de direccionamiento en la nube internetCloud_B.xml del Escenario 1 Simulación B.

```
<!-- Nombre del módulo y tráfico simétrico -->
<internetCloud symmetric="true">
  <parameters name="good">
    <!-- Nombre la entrada y salida de información, retardo generado,
    velocidad y probabilidad de paquetes descartados en el módulo -->
    <traffic src="redHostCliente[*]" dest="ISPServidorRouter"
    delay="20ms+truncnormal(200ms,60ms)"
    datarate="uniform(10Mbps,100Mbps)" drop="uniform(0,1) &lt; 0.01" />
    <!--
    <traffic src="*" dest="*" delay="10ms+truncnormal(100ms,20ms)"
    datarate="uniform(10Mbps,100Mbps)" drop="0 &lt; 0" />
    -->
  </parameters>
</internetCloud>
```

Anexo E

Código de programación del archivo de obstáculos físicos paredes.xml del Escenario 1 Simulación A.

```
<!-- Objetos físicos a incluir en la simulación posición,  
orientación, estructura, material, color y opacidad -->  
  
<environment>  
  
<object position="min 20 20 0" orientation="0 0 0" shape="cuboid 5  
200 4" material="concrete" fill-color="203 65 84" opacity="0.4"/>  
  
<object position="min 20 20 0" orientation="0 0 0" shape="cuboid 200  
5 4" material="concrete" fill-color="203 65 84" opacity="0.4"/>  
  
<object position="min 220 20 0" orientation="0 0 0" shape="cuboid 5  
200 4" material="concrete" fill-color="203 65 84" opacity="0.4"/>  
  
<object position="min 20 220 0" orientation="0 0 0" shape="cuboid  
200 5 4" material="concrete" fill-color="203 65 84" opacity="0.4"/>  
  
<object position="min 20 140 0" orientation="0 0 0" shape="cuboid  
120 5 4" material="glass" fill-color="203 65 84" opacity="0.4"/>  
  
<object position="min 155 100 0" orientation="0 0 0" shape="cuboid 5  
120 4" material="glass" fill-color="203 65 84" opacity="0.4"/>  
  
</environment>
```

Anexo F

Código de programación del archivo redLTE.ned de descripción de red del Escenario 2 Simulación C.

```
//Paquete contenedor del archivo NED
package inet.examples.sistema_pa_ecg.escenario2;
//Importación de librerías y módulos
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 lte.corenetwork.binder.LteBinder;
import lte.corenetwork.nodes.eNodeB;
import lte.corenetwork.nodes.Ue;
import lte.corenetwork.nodes.ExtCell;
import lte.epc.PgwStandardSimplified;
import lte.world.radio.LteChannelControl;
import inet.node.ethernet.Eth10M;
import inet.common.misc.ThruputMeteringChannel;
import inet.node.internetcloud.InternetCloud;
import inet.node.ethernet.Eth100M;
import inet.visualizer.integrated.IntegratedCanvasVisualizer;
import inet.common.figures.DelegateSignalConfigurator;
import lte.corenetwork.nodes.ExtCell;

//Red de la Simulación C del Escenario 2
network simulacion_C
{
    //parámetros y señales a ser desplegadas en el ambiente de
    simulación
    parameters:
        int numUe1;
        int numUe2;
        int numUe3;
        int numExtCells = default(0);

    @display("i=block/network2;bgb=2412.1863,2199.3462;bgi=background/te
rrain,t;bgg=100,,#00FF40;bgu=m");
        @figure[title](type=label; pos=0,-1; anchor=sw;
color=darkblue);
        //declaración de señal para mostrar el número de
        paquetes que llegan al servidor médico mientras se ejecuta la
        simulación
        @figure[rcvdPkText](type=indicatorText; pos=300,50;
anchor=w; font=,45; textFormat="Paquete recibidos en el Servidor:
%g"; initialValue=0);
        @statistic[rcvdPk](source=serverHostMedico_rcvdPk;
record=figure(count); targetFigure=rcvdPkText);
        @signal[serverHostMedico_rcvdPk];

    @delegatesignal[rcvdPk](source=serverHostMedico_udpApp[0].rcvdPk;
target=serverHostMedico_rcvdPk);

    types:
        //creación del canal de fibra óptica
        channel fiberline extends ThruputMeteringChannel
```

```

    {
        delay = 1us;
        datarate = 100Mbps;
        thruputDisplayFormat = "u";
    }

submodules:
    //inclusión de los submódulos de la red con su ubicación
    (x,y) y tipo de ícono
    channelControl: LteChannelControl {
        @display("p=2195.1729,392.29324;is=s");
    }
    routingRecorder: RoutingTableRecorder {
        @display("p=2199.3462,734.50653;is=s");
    }
    configurator: IPv4NetworkConfigurator {
        @display("p=2199.3462,575.91986");
        config = xmldoc("demo.xml");
    }
    binder: LteBinder {
        @display("p=2199.3462,888.9198;is=s");
    }
    server: StandardHost {
        @display("p=125.199974,87.639984;is=n;i=device/server");
    }
    pgw: PgwStandardSimplified {
        nodeType = "PGW";
        @display("p=121.02664,392.29324;is=l");
    }
    router1: Router {
@display("p=438.19992,655.2132;i=device/smallrouter;is=s");
    }
    router2: Router {
@display("p=959.86646,605.13324;i=device/smallrouter;is=s");
    }
    router3: Router {
@display("p=509.14658,1364.6797;i=device/smallrouter;is=s");
    }
    routerX2: Router {
@display("p=934.8265,1110.1064;i=device/smallrouter;is=s");
    }
    eNodeB1: eNodeB {
        @display("p=517.4932,872.2265;is=l;r=500,wheat");
    }
    eNodeB2: eNodeB {
        @display("p=1377.1997,872.2265;is=l;r=500,yellow");
    }
    eNodeB3: eNodeB {
        @display("p=926.4798,1594.213;is=l;r=500,red");
    }
    ue1[numUe1]: Ue {
        @display("p=267.0933,872.2265;is=vs");
    }
    ue2[numUe2]: Ue {
        @display("p=1377.1997,1110.1064;is=vs");
    }
}

```

```

ue3[numUe3]: Ue {
    @display("p=1164.3597,1752.7997;is=vs");
}
figureHelper: DelegateSignalConfigurator {
    @display("p=2199.3462,58.426655");
}
internetCloud: InternetCloud {
    @display("p=1802.8796,204.49329;is=v1");
}

ISPServidorRouter: Router {
    parameters:
        @display("p=1982.3329,1076.7197");
}

serverHostMedico: StandardHost {
    parameters:
        @display("p=1982.3329,1473.1864;i=device/server");
}
visualizer: IntegratedCanvasVisualizer {
    parameters:
        @display("p=2199.3462,262.91995");
}
router: Router {
    @display("p=592.6132,208.66663");
}

connections:
    //conexiones cableadas entre las interfaces de los
módulos
router.pppg++ <--> Eth10G <--> pgw.filterGate;
server.pppg++ <--> Eth10G <--> router.pppg++;
pgw.pppg++ <--> Eth10G <--> router1.pppg++;
pgw.pppg++ <--> Eth10G <--> router2.pppg++;
pgw.pppg++ <--> Eth10G <--> router3.pppg++;
router1.pppg++ <--> Eth10G <--> eNodeB1.ppp;
router2.pppg++ <--> Eth10G <--> eNodeB2.ppp;
router3.pppg++ <--> Eth10G <--> eNodeB3.ppp;
ISPServidorRouter.pppg++ <--> fiberline <-->
internetCloud.pppg++;
serverHostMedico.ethg++ <--> Eth100M <-->
ISPServidorRouter.ethg++;
ISPServidorRouter.pppg++ <--> fiberline <-->
internetCloud.pppg++;
    //conexiones de interfaces X2
eNodeB1.x2++ <--> Eth10G <--> routerX2.pppg++;
eNodeB2.x2++ <--> Eth10G <--> routerX2.pppg++;
eNodeB3.x2++ <--> Eth10G <--> routerX2.pppg++;
router.pppg++ <--> fiberline <--> internetCloud.pppg++;
}

```

Anexo G

Código de programación del archivo de inicialización omnetpp.ini del Escenario

2 Simulación C.

```
#declaración de las configuraciones generales para una red LTE
[General]
image-path=../../images #carpeta de imágenes
tkenv-plugin-path = ../../../../inet/etc/plugins #carpeta de plugins
output-scalar-file-append = false
sim-time-limit = 60s #tiempo límite a simular
**.vector-recording = true #habilitar la creacion de vectores de
reultados

#####
####CONFIGURACIONES DE SIMULACIÓN###
#####
tkenv-default-config =

#####
####CONFIGURACIONES DEL CANAL###
#####
**.channelControl.pMax = 10W #potencia máxima de transmisión en la
red
**.channelControl.alpha = 1.0
**.channelControl.carrierFrequency = 1980e+6Hz #frecuencia portadora
**.channelControl.sat = -110dBm #umbral de atenuación de la señal
**.channelControl.propagationModel = "FreeSpaceModel" #modelo de
propagación

#####
####CONFIGURACIONES MAC###
#####
**.mac.queueSize = 1MiB #cantidad de bytes para detectar otrs
dispositivos
**.mac.maxBytesPerTti = 1KiB #cantidad máxima de bytes por tti
**.mac.macDelay.result-recording-modes = all #almacenamiento de
resultados
**.mac.macThroughput.result-recording-modes = all #almacenamiento de
resultados
# Schedulers o programadores, controlan el ancho de banda que se
asigna a cada UE segun
#la información recibida MAXCI utilizado para tráfico UL DL y D2D
**.mac.schedulingDisciplineDl = "MAXCI"
**.mac.schedulingDisciplineUl = "MAXCI"

#####
####CONFIGURACIONES DE CAPA FÍSICA###
#####
#habilitar retardos de propagación e inclusión del archivo de
configuracion de capa física
**.nic.phy.usePropagationDelay = true
**.nic.phy.channelModel=xmldoc("config_channel.xml")

#####
####CONFIGURACIONES DEL CANAL DE RETROALIMENTACIÓN###
#####
#inclusión del archvo de configuracion de canal de retroalimentación
```

```

**.feedbackComputation = xmldoc("config_channel.xml")
#intervalo de tiempo entre el sensado y transmisión en el intervalo
de tiempo
#de trasmisiónTTI
**.fbDelay = 1

#####
###CONFIGURACIONES DE MOVILIDAD###
#####
#area de movimiento del nodo en el eje Z
**.mobility.constraintAreaMinZ = 0m
**.mobility.constraintAreaMaxZ = 0m
**.mobility.initFromDisplayString = true
# no se habilita el Handover
**.enableHandover = false

#####
###CONFIGURACIONES DE DESPLIEGUE DE RED###
#####
#intervalo de reporte de posicion de nodos hacia el eNB e intervalo
de mensajes broadcast
**.deployer.positionUpdateInterval = 0.001s
**.deployer.broadcastMessageInterval = 1s #usado en unidades remotas
(RU o DAS) y en handover
#número de bloques de recurso de enlace DL y UL
**.deployer.numRbDl = 6
**.deployer.numRbUl = 6
#número de subportadoras por RB
**.deployer.rbyDl = 12 #pueden existir hasta 12 subportadoras por RB
**.deployer.rbyUl = 12
#número de símbolos OFDM
**.deployer.rbxDl = 7
**.deployer.rbxUl = 7
#número de símbolos de señalización
**.deployer.signalDl = 1
**.deployer.signalUl = 1
#número de bandas
**.deployer.numBands = 1

#####
###CONFIGURACIONES DE MÓDULO AMC###
#####
#tipo de asignación de recursos y MAC automática
**.rbAllocationType = "localized"
**.mac.amcMode = "AUTO"
#la retroalimentacion se habilita para todas las bandas
**.feedbackType = "ALLBANDS"
**.maxHarqRtx = 3

#####
###CONFIGURACIONES DE POTENCIAS DE TRANSMISIÓN###
#####
#potencias de transmisión del los UE y los eNB
**.ueTxPower = 26
**.microTxPower = 20
**.eNodeBTxPower = 46

#declaración de nombre de la configuración
[Config simulacion_C]

```

```

network= simulacion_C #red a la que pertenece

#####
####CONFIGURACIONES DE MOVILIDAD Y ASOCIACIÓN###
#####
#cada UE se asocia a su respectiva celda LTE
**.ue1[*].masterId = 1
**.ue1[*].macCellId = 1
**.ue2[*].masterId = 2
**.ue2[*].macCellId = 2
**.ue3[*].masterId = 3
**.ue3[*].macCellId = 3
#puntos entre los cuales los UE se pueden mover
*.ue1[*].mobility.constraintAreaMinX = 300m
*.ue1[*].mobility.constraintAreaMinY = 600m
*.ue1[*].mobility.constraintAreaMaxX = 800m
*.ue1[*].mobility.constraintAreaMaxY = 1200m
*.ue2[*].mobility.constraintAreaMinX = 1200m
*.ue2[*].mobility.constraintAreaMinY = 600m
*.ue2[*].mobility.constraintAreaMaxX = 1700m
*.ue2[*].mobility.constraintAreaMaxY = 1200m
*.ue3[*].mobility.constraintAreaMinX = 700m
*.ue3[*].mobility.constraintAreaMinY = 1400m
*.ue3[*].mobility.constraintAreaMaxX = 1200m
*.ue3[*].mobility.constraintAreaMaxY = 2000m
#intervalo de actualización de ubicación y tipo de movilidad
**.updateInterval = 1s
**.ue*[*].mobilityType = "GaussMarkovMobility"
#parámetros de la movilidad aleatoria
**.ue*[*].mobility.alpha = 0.9
**.ue*[*].mobility.speed = 1mps #velocidad
**.ue*[*].mobility.angle = 0deg
**.ue*[*].mobility.variance = 40
**.ue*[*].mobility.margin = 30m

#####
####CONFIGURACIONES DE CAPA DE APICACIÓN###
#####
#Número de aplicaciones UDP, puerto a utilizar, tipo de aplicación
**.addDefaultRoutes = false
**.ue*[*].numUdpApps = 1
**.server*.numUdpApps = 1
**.ue*[*].udpApp[0].typename = "UDPBasicApp"
**.ue*[*].udpApp[*].localPort = 1000
**.server*.udpApp[0].typename = "UDPBasicApp"
**.server*.udpApp[*].localPort = 1000
#destino de los mensajes UDP enviados, puerto y tamaño del mensaje
de capa de aplicación
#intervalo de envío del mensaje y tiempo de paro de las aplicaciones
UDP
**.ue*[*].udpApp[0].destAddresses = "serverHostMedico"
**.server*.udpApp[0].destPort = 1000
**.server*.udpApp[0].messageLength = 125B
**.server*.udpApp[0].sendInterval = 1s
**.server*.udpApp[0].stopTime = 48.5s
**.ue*[*].udpApp[0].destPort = 1000
**.ue*[*].udpApp[0].messageLength = 125B
**.ue*[*].udpApp[0].sendInterval = 1s
**.ue*[*].udpApp[0].stopTime = 48.9s

```

```
#####  
###CONFIGURACIONES DE DIRECCIONAMIENTO EN LA NUBE###  
#####  
#llamada al archivo xml que contiene las configuraciones del módulo  
cloud  
**.internetCloud.networkLayer.delayer.config =  
xmlDoc("internetCloud_C.xml")  
  
#####  
###CONFIGURACIONES DE ANIMACIONES###  
#####  
*.visualizer.sceneVisualizer.descriptionFigure = "title" #pone título  
en el gráfico
```

Anexo H

Código de programación del archivo de configuración de capa física config_channel.xml del Escenario 2 Simulación C.

```
<?xml version="1.0" encoding="UTF-8"?>
<root>
    <!-- Channel Model Type (REAL, DUMMY) -->
    <ChannelModel type="REAL">
        <!-- Enable/disable shadowing -->
        <parameter name="shadowing" type="bool" value="true"/>
        <!-- Pathloss scenario from ITU -->
        <parameter name="scenario" type="string"
value="URBAN_MACROCELL"/>
        <!-- eNodeB height -->
        <parameter name="nodeb-height" type="double"
value="25"/>
        <!-- Building height -->
        <parameter name="building-height" type="double"
value="20"/>
        <!-- Carrier Frequency (GHz) -->
        <parameter name="carrierFrequency" type="double"
value="1.98"/>
        <!-- Target bler used to compute feedback -->
        <parameter name="targetBler" type="double"
value="0.001"/>
        <!-- HARQ reduction -->
        <parameter name="harqReduction" type="double"
value="0.2"/>
        <!-- Rank indicator tracefile -->
        <parameter name="lambdaMinTh" type="double"
value="0.02"/>
        <parameter name="lambdaMaxTh" type="double"
value="0.2"/>
        <parameter name="lambdaRatioTh" type="double"
value="20"/>
        <!-- Antenna Gain of UE -->
        <parameter name="antennaGainUe" type="double"
value="0"/>
        <!-- Antenna Gain of eNodeB -->
        <parameter name="antennGainEnB" type="double"
value="18"/>
        <!-- Antenna Gain of Micro node -->
        <parameter name="antennGainMicro" type="double"
value="5"/>
        <!-- Thermal Noise for 10 MHz of Bandwidth -->
        <parameter name="thermalNoise" type="double" value="-
104.5"/>
        <!-- Ue noise figure -->
        <parameter name="ue-noise-figure" type="double"
value="7"/>
        <!-- eNodeB noise figure -->
        <parameter name="bs-noise-figure" type="double"
value="5"/>
        <!-- Cable Loss -->
        <parameter name="cable-loss" type="double" value="2"/>
    </ChannelModel>
</root>
```

```

        <!-- If true enable the possibility to switch
dynamically the LOS/NLOS pathloss computation -->
        <parameter name="dynamic-los" type="bool"
value="false"/>
        <!-- If dynamic-los is false this parameter, if true,
compute LOS pathloss otherwise compute NLOS pathloss -->
        <parameter name="fixed-los" type="bool" value="false"/>
        <!-- Enable/disable fading -->
        <parameter name="fading" type="bool" value="true"/>
        <!-- Fading type (JAKES or RAYGHLEY) -->
        <parameter name="fading-type" type="string"
value="RAYGHLEY"/>
        <!-- If jakes fading this parameter specify the number
of path (tap channel) -->
        <parameter name="fading-paths" type="int" value="6"/>
        <!-- if true, enables the inter-cell interference
computation -->
        <parameter name="extCell-interference" type="bool"
value="true"/>
        <!-- if true, enables the multi-cell interference
computation -->
        <parameter name="multiCell-interference" type="bool"
value="false"/>
        </ChannelModel>

        <!-- Feedback Type (REAL, DUMMY) -->
        <FeedbackComputation type="REAL">
            <!-- Target bler used to compute feedback -->
            <parameter name="targetBler" type="double"
value="0.001"/>
            <!-- Rank indicator tracefile -->
            <parameter name="lambdaMinTh" type="double"
value="0.02"/>
            <parameter name="lambdaMaxTh" type="double"
value="0.2"/>
            <parameter name="lambdaRatioTh" type="double"
value="20"/>
        </FeedbackComputation>
    </root>

```

Anexo I

Código de programación del archivo de direccionamiento en la nube internetCloud_C.xml del Escenario 2 Simulación C.

```
<!-- Nombre del módulo y tráfico simétrico -->
<internetCloud symmetric="true">
  <parameters name="good">
    <!-- Nombre la entrada y salida de información, retardo generado,
    velocidad y probabilidad de paquetes descartados en el módulo -->
    <traffic src="router" dest="ISPServidorRouter"
    delay="20ms+truncnormal(200ms,60ms)"
    datarate="uniform(10Mbps,100Mbps)" drop="0 &lt; 0" />
    <!--
    <traffic src="*" dest="*" delay="10ms+truncnormal(100ms,20ms)"
    datarate="uniform(10Mbps,100Mbps)" drop="uniform(0,1) &lt; 0.01" />
    -->
  </parameters>
</internetCloud>
```