

UNIVERSIDAD TÉCNICA DEL NORTE

FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CARRERA DE INGENIERÍA EN TELECOMUNICACIONES

TRABAJO DE INTEGRACIÓN CURRICULAR

TEMA:

"EVALUACIÓN DEL PROTOCOLO IMPROVED LEACH EN UNA RED DE SENSORES INALÁMBRICAS SUBACUÁTICOS"

Trabajo de titulación previo a la obtención del título de Ingeniero en Telecomunicaciones

Línea de investigación: Innovación Tecnológica y de productos.

AUTOR:

WILMER DAVID FARINANGO TALLANA

DIRECTOR:

MSc. FABIÁN GEOVANNY CUZME RODRÍGUEZ

Ibarra – Ecuador, 2026

UNIVERSIDAD TÉCNICA DEL NORTE

DIRECCIÓN DE BIBLIOTECA

AUTORIZACIÓN DE USO Y PUBLICACIÓN A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE

1. IDENTIFICACIÓN DE LA OBRA

En cumplimiento del Art. 144 de la Ley de Educación Superior, hago la entrega del presente trabajo a la Universidad Técnica del Norte para que sea publicado en el Repositorio Digital Institucional, para lo cual pongo a disposición la siguiente información:

DATOS DE CONTACTO	
APELLIDOS Y NOMBRES:	WILMER DAVID FARINANGO TALLANA

DATOS DE LA OBRA	
TÍTULO:	"EVALUACIÓN DEL PROTOCOLO IMPROVED LEACH EN UNA RED DE SENSORES INALÁMBRICAS SUBACUÁTICOS"
AUTOR (ES):	WILMER DAVID FARINANGO TALLANA
FECHA: DD/MM/AAAA	31/08/2026
SOLO PARA TRABAJOS DE GRADO	
PROGRAMA:	■ PREGRADO ☐ POSGRADO
TITULO POR EL QUE OPTA:	INGENIERO EN TELECOMUNICACIONES
DIRECTOR /ASESOR:	MSc. FABIÁN GEOVANNY CUZME RODRÍGUEZ

2. CONSTANCIAS

El autor (es) manifiesta (n) que la obra objeto de la presente autorización es original y se la desarrolló, sin violar derechos de autor de terceros, por lo tanto la obra es original y que es (son) el (los) titular (es) de los derechos patrimoniales, por lo que asume (n) la responsabilidad sobre el contenido de la misma y saldrá (n) en defensa de la Universidad en caso de reclamación por parte de terceros.

Ibarra, a los 31 días del mes de agosto de 2026

EL AUTOR:

Validar únicamente en FirmaEC.
Firmado electrónicamente por:
WILMER DAVID
FARINANGO TALLANA

Wilmer David Farinango Tallana

UNIVERSIDAD TÉCNICA DEL NORTE

FACULTAD DE INGENIERÍA EN CIENCAS APLICADAS

CERTIFICACIÓN

Ibarra, a los 31 días del mes de agosto de 2026

MAGÍSTER FABIÁN GEOVANNY CUZME RODRÍGUEZ

DIRECTOR DEL PRESENTE TRABAJO DE TITULACIÓN

CERTIFICA:

Que el presente trabajo de titulación trabajo de grado previo a la obtención del título de ingeniero en telecomunicaciones, ha sido desarrollado por el señor FARINANGO TALLANA WILMER DAVID bajo mi supervisión.

Es todo cuanto puedo certificar en honor a la verdad.

Validar únicamente en FirmaEC.
Firmado electrónicamente por:
FABIAN GEOVANNY CUZME RODRIGUEZ

.....

MSc. Fabián Geovanny Cuzme Rodríguez

DIRECTOR

DEDICATORIA

A mi abuelita, Delia Morales:

A ti, que no me diste la vida, pero me diste una vida entera. Fuiste quien me sostuvo cuando aún no entendía el mundo, quien me enseñó con el ejemplo lo que significa la constancia, la nobleza y el amor sincero. Siempre fuiste, y siempre serás, mi más grande admiración. Tu presencia me acompañó en silencio en cada etapa de este camino, y aunque las palabras a veces me falten, sé que este logro también es tuyo, porque nada de lo que soy hoy existiría sin ti.

A mis padres, Luis Farinango M., quien me enseñó que la vida no es fácil, y Gladys Tallana A., quien me enseñó a no rendirme jamás. A ustedes, que estuvieron a mi lado cuando más los necesité, sostienen sin saberlo cada página de este trabajo. Gracias por creer en mí incluso cuando yo mismo dudaba, por secar mis caídas con palabras de aliento y por enseñarme, con su ejemplo, que el esfuerzo y la perseverancia siempre encuentran su recompensa. Todo lo que hoy logro nace de lo que ustedes sembraron en mí.

A mis amigos, que hicieron más ligero este trayecto. A Francis Echeverría, que me acompañó en las buenas y en las malas sin soltarme la mano, recordándome que nunca estuve solo. A Wladimir Velásquez, que siempre actuó como un hermano mayor y supo aconsejarme en cada momento en que necesité un consejo sincero. Y a Francis Ipiales, que no me dejó caer y que, aun en mis peores momentos, siempre creyó y me repitió que todo iba a estar bien. Su amistad fue un refugio en medio del cansancio.

Y por último, a mi propio esfuerzo. A cada desvelo, a cada tropiezo y a cada día en que quise rendirme y no lo hice. Porque a pesar del tiempo que ha pasado, y de que el camino no ha sido fácil, aprendí a sobrellevarlo con paciencia, a levantarme después de cada caída y a seguir adelante sin rendirme jamás. Este logro es la prueba de que valió la pena.

AGRADECIMIENTO

A mis docentes, que con paciencia y vocación supieron responder mis dudas y compartir su conocimiento. Gracias por guiarme, por exigirme cuando fue necesario y por dejar en mí una huella que va más allá de las aulas. Buena parte de lo que hoy alcanzo se lo debo a sus enseñanzas.

De manera especial, agradezco a mi tutor de tesis, MSc. Fabián Geovanny Cuzme Rodríguez, por su guía, su tiempo y su acompañamiento durante el desarrollo de esta investigación. Su orientación fue fundamental para llevar este trabajo a buen término.

Mi gratitud también al MSc. Jaime Roberto Michilena Calderón, asesor y coordinador de la carrera, por su amabilidad, su firmeza y su integridad. Su visión y su pensamiento lógico han dejado una huella profunda en la carrera y en cada uno de quienes tuvimos el privilegio de formarnos bajo su guía.

A la Universidad Técnica del Norte, por abrirme sus puertas y brindarme la formación que hoy se refleja en estas páginas.

A todas mis tías, que actuaron como madres en los momentos más complicados de mi vida, brindándome su cariño y su apoyo incondicional cuando más lo necesité. Y a mi primo Edson Abalco, que estuvo presente y nunca me dejó solo en el camino.

Y a todas las personas que, de una u otra forma, contribuyeron a que este objetivo se hiciera realidad. A cada una de ellas, mi más sincera gratitud.

RESUMEN

El presente trabajo de grado evalúa el desempeño del protocolo Improved-LEACH frente a LEACH en redes de sensores inalámbricos subacuáticos (UWSN) mediante simulación en OMNeT++. Estas redes operan bajo restricciones difíciles de sortear: energía limitada, baterías que casi no pueden reemplazarse una vez desplegados los nodos, ancho de banda reducido y el retardo propio de la propagación acústica. LEACH reparte el rol de cabeza de clúster con un criterio probabilístico que no considera cuánta energía le queda a cada nodo; Improved-LEACH pondera ese umbral según la energía residual para repartir mejor la carga de comunicación.

Este trabajo compara ambos protocolos mediante simulación en OMNeT++. Se construyó un escenario tridimensional con un sink en la superficie y comunicación acústica, y se probaron cinco tamaños de red (50, 100, 250, 500 y 1000 nodos) con diez semillas emparejadas por cada combinación, para un total de 100 corridas. Los dos protocolos se ejecutaron con los mismos parámetros; lo único que cambió fue la manera de elegir las cabezas de clúster.

Improved-LEACH retrasó la muerte del primer nodo de 18,60 a 28,10 rondas (51,1 %), la del nodo intermedio de 41,86 a 52,96 (26,5 %) y la del último nodo de 54,16 a 60,38 (11,5 %). El throughput efectivo mejoró hasta un 11,4 % en la red de 1000 nodos. Las medias, desviaciones estándar e intervalos de confianza del 95 % confirman que estas diferencias no provienen de corridas aisladas. La mejora no es igual en todas las métricas ni en todas las densidades, pero el protocolo alarga de forma consistente la vida útil de la red en escenarios medianos y grandes.

Palabras clave: UWSN, LEACH, Improved-LEACH, eficiencia energética, clustering, OMNeT++.

ABSTRACT

This undergraduate thesis evaluates the performance of the Improved-LEACH protocol against LEACH in underwater wireless sensor networks (UWSN) through simulation in OMNeT++. These networks operate under constraints that are hard to get around: limited energy, batteries that can rarely be replaced once the nodes are deployed, narrow bandwidth, and the delay caused by acoustic propagati. The LEACH protocol assigns the cluster-head role with a probabilistic rule that ignores how much energy each node has left. Improved-LEACH changes that rule and weights the selection threshold by residual energy, aiming to share the communication load more evenly.

This work compares the two protocols through simulation in OMNeT++. A three-dimensional scenario was built with a sink at the surface and acoustic communication, and five network sizes were tested (50, 100, 250, 500 and 1000 nodes) with ten paired seeds for each combination, for a total of 100 runs. Both protocols ran with the same parameters; the only thing that changed was the way the cluster heads were chosen.

Improved-LEACH delayed the first node death from 18.60 to 28.10 rounds (51.1%), the half-node death from 41.86 to 52.96 (26.5%), and the last node death from 54.16 to 60.38 (11.5%). Effective throughput improved by up to 11.4% in the 1000-node network. The means, standard deviations and 95% confidence intervals confirm that these differences do not come from isolated runs. The improvement is not the same across every metric or density, but the protocol consistently extends network lifetime in medium and large scenarios.

Keywords: UWSN, LEACH, Improved-LEACH, energy efficiency, clustering, OMNeT++.

INDICE DE CONTENIDOS

1. CAPÍTULO I: ANTECEDENTES 22
1.1. Tema 22
1.2. Problema 22
1.3. Objetivos 25
1.3.1. Objetivo General 25
1.3.2. Objetivos Específicos 25
1.4. Alcance 26
1.5. Justificación 29
2. CAPÍTULO II: ESTADO DEL ARTE 31
2.1. Redes de Sensores Inalámbricos (WSNs) 31
2.2. Clustering en WSNs 32
2.2.1. Estrategias de Formación de Clusters 33
2.2.1.1. Selección de Cluster Heads (CHs) 33
2.2.1.2. Formación de Clústeres 34
2.2.1.3. Protocolo LEACH y sus Mejoras (Improved LEACH) 34
2.2.2. Protocolos de Clustering 34
2.2.2.1. LEACH y sus Variantes 35
2.2.2.2. Variantes de LEACH 35
2.2.3. Importancia del Clustering en las comunicaciones inalámbricas 36
2.2.3.1. Reducción de la Carga de Comunicación 36
2.2.3.2. Balance de Carga y Vida Útil de la Red 36
2.2.4. Necesidad de implementar cluster en las UWSN 37
2.2.4.1. Limitaciones Energéticas: 38

2.2.4.2. Propagación de Señales y Comunicación: 38
2.2.4.3. Escalabilidad y Gestión de Red: 38
2.2.4.4. Balanceo de Carga: 38
2.2.4.5. Agregación de Datos: 39
2.2.4.6. Reducción de Colisiones y Retransmisiones: 39
2.2.4.7. Adaptabilidad y Flexibilidad: 39
2.3. Métricas de Evaluación en WSNs 39
2.3.1. Métricas de Rendimiento 40
2.3.2. Métricas Específicas de UWSNs 41
2.3.3. Herramientas y Técnicas de Simulación 41
2.3.3.1. OMNET++ 42
2.3.3.2. NS-3 42
2.4. Métricas de Evaluación 43
2.4.1. Energía residual 43
2.4.2. Tasa de Entrega de Paquetes (PDR) 44
2.4.3. Latencia Promedio 45
2.4.4. Vida útil de la red: FND, HND y LND 46
2.4.5. Throughput 46
2.4.6. Distancia promedio CH–Sink 47
3. CAPÍTULO III: METODOLOGIA 48
3.1. Metodología de Desarrollo 48
3.2. Fase I. Planificación del Estudio de Simulación 49
3.2.1. Definición del Problema de Simulación 49
3.2.2. Variables de Entrada y Variables de Salida 50
3.2.3. Supuestos y Limitaciones del Entorno Simulado 51

3.2.4. Criterios de Verificación y Validación del Entorno de Simulación 52

3.3. Fase II. Diseño del Entorno de Simulación UWSN 53

3.3.1. Plataforma de Simulación y Herramientas Utilizadas 53

3.3.1.1. Ubuntu.....54

3.3.1.2. OMNeT++.....54

3.3.1.3. Framework inetmanet-4.x y módulo UanModel.....54

3.3.1.4. Python 55

3.3.2. Definición del Escenario UWSN.....55

3.3.3. Configuración Temporal de la Simulación.....56

3.3.4. Modelo Matemático del Escenario de Simulación 57

3.3.4.1. Geometría del Escenario 57

3.3.4.2. Temporalidad de la Simulación 58

3.3.4.3. Protocolo LEACH — Función de Umbral.....59

3.3.4.4. Protocolo Improved LEACH - Umbral Ponderado por Energía Residual 60

3.3.4.5. Modelo Energético.....61

3.3.4.6. Métricas de Red 62

3.3.4.7. Estadística Descriptiva del Procesamiento de Resultados 64

3.3.5. Modelo del Canal Acústico 65

3.3.6. Modelo de Tráfico 66

3.4. Fase III. Implementación de la Simulación en OMNeT++.....66

3.4.1. Organización del Proyecto.....66

3.4.2. Estructura de Archivos del Proyecto 67

3.4.2.1. Archivos NED.....67

3.4.2.2. Archivos C++.....68

3.4.2.3. Archivo omnetpp.ini 68
3.4.3. Implementación del Protocolo LEACH 69
3.4.3.1. Selección del Cluster Head 70
3.4.3.2. Formación de Clústeres 71
3.4.3.3. Comunicación Sensor → CH → Sink 71
3.4.4. Implementación del Protocolo Improved LEACH 71
3.4.4.1. Modificación Respecto a LEACH: Umbral Ponderado por Energía Residual 72
3.4.4.2. Comunicación Sensor → CH → Sink 73
3.4.5. Flujo General de una Simulación 73
3.4.6. Instrumentación y Generación de Resultados 76
3.5. Fase IV. Diseño Experimental 77
3.5.1. Definición de los Escenarios Experimentales 77
3.5.2. Campañas Experimentales 78
3.5.3. Parámetros de Simulación 79
3.5.4. Control de Aleatoriedad 79
3.5.5. Métricas Registradas Durante la Simulación 80
3.6. Fase V. Procesamiento y Validación de Resultados 80
3.6.1. Extracción de Datos desde OMNeT++ 81
3.6.2. Flujo de Procesamiento de Resultados 81
3.6.3. Procesamiento Estadístico 84
3.6.4. Verificación de Repetibilidad 85
3.6.5. Validación del Entorno de Simulación 85
4. CAPÍTULO IV: VALIDACIÓN DE RESULTADOS 87
4.1. Configuración Experimental Final 87

4.1.1. Parámetros Controlados y Trazabilidad....87
4.1.2. Horizonte Temporal de Simulación por escenario 88
4.1.3. Escenarios Ejecutados y Corridas Válidas 89
4.2. Tratamiento Estadístico y Recolección de Métricas 89
4.2.1. Fuentes de Datos: Archivos .sca y .vec 90
4.2.2. Estadísticos Reportados y Comparación Estadística 91
4.3. Criterio de Alineación Temporal de las Series....93
4.3.1. Verificación de Consistencia de los Datos 96
4.3.2. Verificación del PDR....97
4.3.3. Cálculo de Ganancia Relativa....99
4.4. Resultados por Métrica....101
4.4.1. Vida Útil de la Red: FND, HND y LND 101
4.4.2. Energía Residual Promedio por Ronda....104
4.4.3. Nodos Vivos por Ronda....105
4.4.4. Número de Cluster Heads Seleccionados por Ronda 107
4.4.5. Tasa de Entrega de Paquetes (PDR) 110
4.4.6. Throughput Efectivo....111
4.4.7. Latencia Promedio de Entrega....114
4.4.8. Análisis de Escalabilidad: Efecto de la Densidad de la Red 117
4.5. Comparación Integral LEACH vs Improved LEACH 119
4.5.1. Tablas Resumen de Resultados Comparativos 119
4.5.2. Tabla Resumen B: Métricas de Comunicación 120
4.5.3. Ganancia Relativa de Improved LEACH por Escenario y Métrica 121
4.6. Discusión....123

4.6.1. Interpretación de los Resultados en el Contexto UWSN 123

4.6.2. Relación entre Densidad de Red y Desempeño 125

4.6.3. Comparación con Literatura Relacionada 125

4.6.4. Limitaciones de los Resultados 128

4.6.5. Amenazas a la Validez de los Resultados 128

Conclusiones ***..... 130***

Recomendaciones ***..... 132***

Bibliografía ***..... 134***

ANEXOS ***..... 138***

ANEXO A. INSTALACIÓN DE UBUNTU 22.04.5 LTS 138

ANEXO B. INSTALACION DE OMNET++ 6.03, EN UBUNTU 145

ANEXO C. REPOSITORIO DIGITAL DEL PROYECTO 156

ANEXO E. EVIDENCIAS COMPLEMENTARIAS DE EJECUCIÓN 156

INDICE DE TABLAS

Tabla 1 Variantes del protocolo LEACH y sus características principales 35
Tabla 2 Métricas de Desempeño en Redes de Sensores Inalámbricos (WSNs) 40
Tabla 3 Variables de entrada del estudio de simulación 50
Tabla 4 Variables de salida (métricas) del estudio de simulación 51
Tabla 5 Criterios de aceptación del entorno de simulación 53
Tabla 6 Parámetros del modelo del canal acústico subacuático 65
Tabla 7 Escenarios experimentales definidos 78
Tabla 8 Parámetros de simulación comunes a todos los escenarios 79
Tabla 9 Estado de validación del entorno de simulación (verificado en Fase III) 85
Tabla 10 Parámetros de la simulación 87
Tabla 11 Rondas hasta LND por escenario y protocolo ($\mu \pm IC_{95}$) 88
Tabla 12 Resumen real de la campaña experimental 89
Tabla 13 Matriz de trazabilidad de las métricas de evaluación del desempeño 90
Tabla 14 Rango de la ronda de muerte del último nodo según protocolo y número inicial de nodos 95
Tabla 15 Validación de pares de semillas, consistencia energética y rangos de LND por escenario 96
Tabla 16 Clasificación, fuente de datos y estado de cálculo de las métricas PDR 99
Tabla 17 Criterios de comparación y sentido favorable de las métricas de desempeño 101
Tabla 18 Comparación de la primera muerte de nodo (FND) entre LEACH e I-LEACH por escenario ($\mu \pm IC_{95}$) 102
Tabla 19 HND por escenario y protocolo - estadísticos completos 102
Tabla 20 LND por escenario y protocolo - estadísticos completos 103
Tabla 21 La energía residual promedio por nodo para LEACH e Improved-LEACH 104
Tabla 22 Número promedio de Cluster Heads durante las rondas activas por escenario y protocolo 108
Tabla 23 PDR real por escenario y protocolo — estadísticos completos 110
Tabla 24 Comparación del throughput efectivo de LEACH e Improved-LEACH 112
Tabla 25 Latencia promedio registrada para LEACH e Improved-LEACH por escenario 116
Tabla 26 Valores promedio y ganancia relativa de FND, HND y LND por escenario 120

Tabla 27 Resumen de métricas de comunicación por escenario.....120
Tabla 28 Ganancia relativa de las métricas de desempeño de I-LEACH respecto a LEACH por escenario 123
Tabla 29 Comparación metodológica entre estudios relacionados y la presente investigación 127

INDICE DE FIGURAS

Figura 1 Topología de Improved LEACH y sus componentes. 28
Figura 2. Arquitectura general del entorno de simulación 55
Figura 3 Estructura de carpetas del proyecto en el IDE de OMNeT++ (Eclipse), mostrando los directorios principales y los archivos Leach.cc, Leach.h y omnetpp.ini 67
Figura 4 Sección [Config LEACH_50] y [Config ImprovedLEACH_50] con los parámetros principales de la simulación 69
Figura 5 Implementación de la función generateThresholdValue() del protocolo LEACH 70
Figura 6 Condición if(useImprovedLEACH) y la aplicación del factor [Eres/E_0] en la función generateThresholdValue() de Leach.cc. Código completo en Anexo A 73
Figura 7 Diagrama de flujo de la simulación LEACH e Improved LEACH 74
Figura 8 Diagrama del flujo de procesamiento de resultados: OMNeT++ → Archivos .sca → Archivos .vec → opp_scavetool → CSV → Python (pandas/NumPy) → Promedio / Desviación estándar / IC_{95} → matplotlib → Gráficas finales 81
Figura 9 Energía residual promedio por nodo en función de la ronda para el escenario S1 ... 84
Figura 10 Salida de la consola de OMNeT++ mostrando la finalización de una corrida de simulación por el evento LND. 86
Figura 11 Evolución de la energía residual promedio por nodo para LEACH e Improved-LEACH 105
Figura 12 Evolución del número de nodos vivos por ronda para LEACH e Improved-LEACH 106
Figura 13 Número promedio de Cluster Heads seleccionados por ronda para LEACH e Improved-LEACH 108
Figura 14 Throughput efectivo de LEACH e Improved-LEACH para los cinco escenarios evaluados 113
Figura 15 Latencia promedio registrada por la implementación 115
Figura 16 FND y LND en función del número inicial de nodos para LEACH e Improved-LEACH 118
Figura 17 Comportamiento del PDR acumulado final y del número promedio de Cluster Heads en función del número de nodos 119

Figura 18 Ganancia relativa de Improved-LEACH respecto a LEACH por escenario y métrica 121

INDICE DE ECUACIONES

Ecuación 1 Volumen del escenario tridimensional....57
Ecuación 2 Distribución uniforme de las coordenadas de los nodos....58
Ecuación 3 Distancia euclidiana tridimensional....58
Ecuación 4 Número efectivo de rondas....59
Ecuación 5 Duración de la ronda....59
Ecuación 6 Umbral de elección de cabeza de clúster (LEACH)....59
Ecuación 7 Umbral de elección de cabeza de clúster (Improved-LEACH)....60
Ecuación 8 Energía de transmisión (modelo de radio de primer orden)....61
Ecuación 9 Energía de recepción....61
Ecuación 10 Actualización de la energía residual....62
Ecuación 11 Condición de muerte del nodo....62
Ecuación 12 Tasa de entrega de paquetes (PDR)....63
Ecuación 13 Throughput efectivo de la red....63
Ecuación 14 Latencia de entrega extremo a extremo....63
Ecuación 15 Media muestral....64
Ecuación 16 Desviación estándar muestral (corrección de Bessel)....64
Ecuación 17 Intervalo de confianza del 95 %....65
Ecuación 18 Media muestral....91
Ecuación 19 Desviación estándar muestral....91
Ecuación 20 Intervalo de confianza del 95 %....92
Ecuación 21 Diferencia pareada por semilla....92
Ecuación 22 Media de las diferencias pareadas....92
Ecuación 23 Intervalo de confianza del 95 % de la diferencia pareada....92
Ecuación 24 Número máximo de rondas para la alineación temporal....93
Ecuación 25 Energía residual promedio por ronda....94
Ecuación 26 Latencia promedio por ronda....94
Ecuación 27 Promedio por ronda....95
Ecuación 28 Expresión de PDR descartada....97
Ecuación 29 PDR del primer salto (sensor–CH)....98
Ecuación 30 PDR del segundo salto (CH–sink)....98

Ecuación 31 Latencia de un paquete.....114
Ecuación 32 Latencia promedio.....114

INDICE DE ANEXOS

Anexo 1 Configuración seleccionada 138
Anexo 2 Pantalla de bienvenida y selección del idioma español. 139
Anexo 3 Selección de la distribución de teclado English (US). 139
Anexo 4 Configuración de actualizaciones y selección de instalación mínima. 140
Anexo 5 Selección de la opción para borrar el disco e instalar Ubuntu. 141
Anexo 6 Confirmación de las particiones ESP y ext4 en el disco de instalación. 141
Anexo 7 Selección de la zona horaria Guayaquil, Ecuador. 142
Anexo 8 Creación del usuario local y configuración del inicio de sesión. 143
Anexo 9 Proceso de copia de archivos y configuración del hardware. 143
Anexo 10 Escritorio de Ubuntu después de completar la instalación. 144
Anexo 11 Información general del entorno de instalación de OMNeT++ 6.0.3 145
Anexo 12 Componentes y dependencias utilizados para la instalación de OMNeT++ 6.0.3 145
Anexo 13 Página oficial de descarga de OMNeT++ 6.0.3 para Linux. 146
Anexo 14 Confirmación de la descarga del archivo comprimido omnetpp-6.0.3-linux-x86_64.tgz. 146
Anexo 15 Descompresión del paquete, carga del entorno y primera ejecución del script de configuración. 148
Anexo 16 Comprobación de las herramientas de construcción y repetición de la configuración antes de completar las dependencias. 148
Anexo 17 Actualización de repositorios e instalación de Bison y Flex. 149
Anexo 18 Instalación de paquetes de Python requeridos y de la herramienta SWIG. 150
Anexo 19 Instalación de los componentes de desarrollo de Qt 5. 150
Anexo 20 Finalización satisfactoria de la fase de configuración de OMNeT++. 151
Anexo 21 Compilación de las bibliotecas, ejecutables y ejemplos incluidos en OMNeT++. 152
Anexo 22 Selección del directorio de trabajo inicial para el IDE. 153
Anexo 23 Pantalla principal de bienvenida del entorno de desarrollo OMNeT++. 154
Anexo 24 Opciones iniciales para instalar INET Framework o importar ejemplos de programación. 154
Anexo 25 Verificación final del IDE de OMNeT++ en funcionamiento. 155

Anexo 26 Verificación de la instalación y funcionamiento de OMNeT++ 6.0.3 155
Anexo 27 Distribución tridimensional inicial del escenario de 50 nodos, simulación A. 156
Anexo 28 Métricas completas de LEACH e Improved-LEACH para 50 nodos, simulación A. 156
Anexo 29 Comparación tridimensional inicial de LEACH e Improved-LEACH para 100 nodos, simulación A..... 157
Anexo 30 Métricas completas de LEACH e Improved-LEACH para 100 nodos, simulación A..... 157
Anexo 31 Comparación tridimensional inicial de LEACH e Improved-LEACH para 250 nodos, simulación A..... 158
Anexo 32 Métricas completas de LEACH e Improved-LEACH para 250 nodos, simulación A..... 158
Anexo 33 Comparación tridimensional inicial de LEACH e Improved-LEACH para 500 nodos, simulación A..... 159
Anexo 34 Métricas completas de LEACH e Improved-LEACH para 500 nodos, simulación A..... 159
Anexo 35 Comparación tridimensional inicial de LEACH e Improved-LEACH para 1000 nodos, simulación A..... 160
Anexo 36 Métricas completas de LEACH e Improved-LEACH para 1000 nodos, simulación A..... 160

1. CAPÍTULO I: ANTECEDENTES

1.1. Tema

Evaluación del protocolo IMPROVED-LEACH en una red de sensores inalámbricas subacuáticos en un entorno simulado.

1.2. Problema

Las redes de sensores inalámbricos (WSNs) son una tecnología ampliamente utilizada en aplicaciones de Internet de las cosas (IoT) y sistemas ciberfísicos (CPS), donde recopilan y transmiten datos en tiempo real desde entornos diversos. Sin embargo, la interoperabilidad y el intercambio de recursos entre las WSN se han vuelto un problema importante a medida que estos sistemas se expanden y se vuelven más heterogéneos. Con la creciente escala de proyectos de IoT/CPS, las técnicas de virtualización de WSN convencionales se han revelado insuficientes debido a las restricciones de hardware y software, lo que impide una eficiente compartición de recursos entre las diversas WSNs (Yıldırım & Tártaro, 2018).

La Red de Sensores Inalámbricos (WSN, por sus siglas en inglés) consta de un gran número de sensores muy pequeños desplegados en una zona específica según la aplicación deseada. Cada sensor contiene componentes de detección, procesamiento de datos y comunicación. Estos sensores forman nodos de WSN que transfieren los datos de detección a la Estación Base (BS) o sumidero. En la BS, los datos se procesan y calculan para obtener resultados comprensibles. La comunicación entre la BS y los nodos inalámbricos se organiza mediante diferentes protocolos. Uno de los protocolos eficientes en energía es el protocolo de enrutamiento LEACH. En este protocolo, la red se divide en diferentes clústeres y cada clúster tiene un Jefe de Clúster (CH) elegido que se conecta con los nodos miembros del clúster y la

BS, recopilando datos de los nodos y luego enviando los datos agregados a la BS (Nasr & Quwaider, 2020).

Las redes de sensores inalámbricos submarinos (UWSN) son sistemas de sensores sin cables que se utilizan en la investigación y el seguimiento del medio subacuático, por ejemplo en seguridad oceánica y en estudios oceanográficos. Estos sistemas utilizan comunicación acústica para superar desafíos como la demora en la transmisión de señales y la reducida velocidad de transferencia de datos, características del ambiente subacuático. Para funcionar dentro de estas restricciones, las UWSN se diseñan de modo que optimicen el consumo de energía y reduzcan las interferencias de señal (Pandith & Goyal, 2022).

Las UWSN enfrentan varios problemas en su implementación y funcionamiento actual. Entre las dificultades más destacadas se encuentran la atenuación severa de la señal bajo el agua, que reduce drásticamente el alcance de las comunicaciones; la interferencia causada por fuentes como la propagación de ondas acústicas y las corrientes marinas; y la limitada capacidad de energía de los nodos sumergidos, lo que hace que la gestión eficiente de la energía sea crítica para la operación sostenible de estas redes. Además, la falta de una infraestructura de soporte y la dificultad para acceder a los nodos en entornos submarinos plantean desafíos logísticos adicionales. En el contexto de esta tesis, la evaluación del protocolo Improved-LEACH (una variante mejorada del protocolo LEACH) busca abordar algunos de estos desafíos para mejorar la eficiencia y la confiabilidad de las UWSN (Pachouri, y otros, 2023).

La problemática central de la tesis "evaluación del protocolo Improved-LEACH en una red de sensores inalámbricas subacuáticos" se relaciona con las múltiples dificultades que enfrentan las UWSN, varias de las cuales pueden abordarse evaluando el protocolo Improved-LEACH. Uno de los desafíos más críticos es el consumo de energía, dado que estos sistemas a menudo dependen de nodos alimentados por baterías con recursos limitados. La eficiencia

energética se vuelve esencial para prolongar la vida útil de las baterías y mantener el funcionamiento continuo de la red (Majitha, Christina, Gifta, & Archana, 2022).

La gestión de datos en entornos subacuáticos plantea dificultades considerables debido a la atenuación de la señal. La transmisión confiable de datos por rutas eficientes es necesaria para que la información recopilada por los sensores llegue a tiempo y con precisión a la estación base (Liu & Zhou, 2023).

Frente a estos problemas que enfrentan las UWSN, una solución viable sería evaluar el protocolo Improved-LEACH en un entorno de simulación como OMNeT++. Esto permitiría llevar a cabo una evaluación detallada y exhaustiva de su desempeño en un escenario subacuático controlado y replicable.

1.3. Objetivos

1.3.1. Objetivo General

Evaluar el protocolo IMPROVED-LEACH en una red de sensores inalámbricos subacuáticos (UWSN) mediante métricas a través de simulaciones utilizando el software OMNET++.

1.3.2. Objetivos Específicos

- Analizar la literatura actual sobre protocolos de enrutamiento UWSN, con especial énfasis en el protocolo LEACH y sus variantes mejoradas, para establecer una base teórica sólida.
- Establecer un entorno de simulación en OMNeT++ que refleje con precisión las condiciones y desafíos únicos de las UWSN, incluyendo la propagación de señales, la topología de la red y el consumo de energía de los nodos.
- Diseñar una serie de pruebas en el entorno de simulación que evalúen el rendimiento del protocolo IMPROVED-LEACH en términos de eficiencia energética, fiabilidad de la transmisión de datos y capacidad de respuesta a cambios dinámicos en la red UWSN.
- Verificar los resultados de simulación mediante la aplicación de métricas de rendimiento, para asegurar la validez del protocolo IMPROVED-LEACH en entornos UWSN.

1.4. Alcance

El enfoque principal de este trabajo de grado se centra en evaluar el protocolo IMPROVED-LEACH en una red de sensores subacuáticos para determinar sus ventajas y desventajas en este contexto. Esta investigación requiere una revisión exhaustiva de la literatura relacionada con las redes de sensores inalámbricos subacuáticos, así como un análisis detallado de los protocolos de enrutamiento existentes, con especial énfasis en el protocolo IMPROVED-LEACH.

Según (Cárdenas, 2018), el enfoque metodológico empleado para abordar las fases del trabajo de grado titulado "Evaluación del protocolo IMPROVED-LEACH en una red de sensores inalámbricas subacuáticos utilizando OMNeT++" es una metodología de simulación que comprende varias etapas.

En la primera fase, se procede con la definición de los objetivos de investigación, donde se busca establecer con precisión lo que se pretende lograr con la evaluación del protocolo Improved-LEACH en una red de sensores subacuáticos en OMNeT++.

En una segunda fase, se realiza una revisión exhaustiva de la literatura relacionada con el tema de investigación. Esto implica comprender el estado actual del campo y examinar trabajos previos relacionados con Improved-LEACH, redes de sensores subacuáticos y simulaciones en OMNeT++.

En una etapa posterior, se lleva a cabo el diseño del experimento. Aquí se definen el alcance y los parámetros del experimento, incluyendo la configuración de la simulación en OMNeT++ y las métricas que se medirán para evaluar el rendimiento de Improved-LEACH en la red subacuática. Se consideran aspectos como la topología de la red, que se ilustra en la Figura 1 con una estructura que integra estaciones base (BS), nodos sensores normales (NS), y cabezas de clúster (CH), el número de nodos y las condiciones subacuáticas simuladas.

La evaluación del protocolo Improved-LEACH en OMNeT++ se efectúa siguiendo las especificaciones y algoritmos relacionados con esta técnica, asegurando que la implementación sea fiel a la descripción original y que funcione correctamente en el entorno simulado.

Posteriormente, se ejecutan simulaciones con el propósito de recopilar datos experimentales. Durante este proceso se establecerán métricas de evaluación, tales como el consumo de energía, el rendimiento de la red, la latencia y la eficiencia energética.

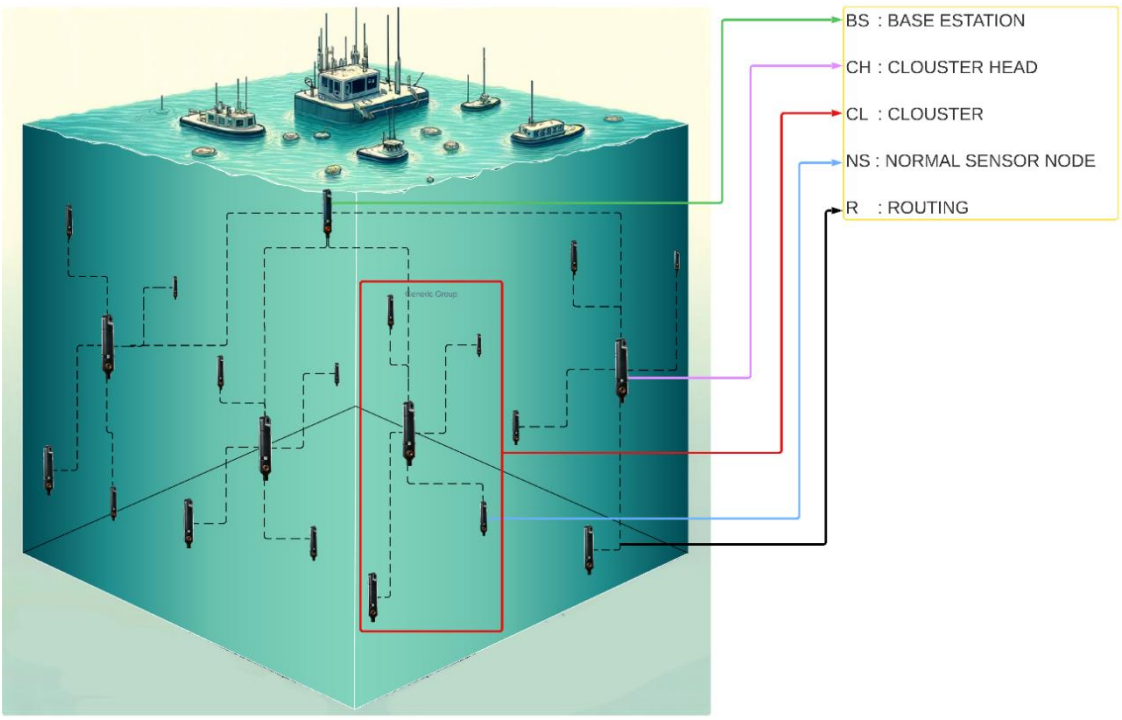
Tras la recopilación de datos, se procede a analizarlos y compararlos con los resultados esperados. Se evalúa el rendimiento de Improved-LEACH en la red de sensores subacuáticos, y cualquier hallazgo significativo es objeto de una discusión detallada.

En la fase de conclusiones y recomendaciones, se resumen los hallazgos y conclusiones obtenidos, discutiendo las implicaciones del trabajo y cómo este podría contribuir al campo de las redes de sensores subacuáticos y a la mejora de protocolos como LEACH. Además, se proporcionan recomendaciones para investigaciones futuras.

Por último, se lleva a cabo la escritura de la tesis, siguiendo una estructura académica estándar que incluye secciones como la introducción, revisión de la literatura, metodología, resultados, discusión, conclusiones y referencias. Finalmente, se prepara una presentación para la defensa de la tesis ante un comité académico, estando preparado para responder preguntas y proporcionar explicaciones adicionales sobre la investigación realizada.

Figura 1

Topología de Improved LEACH y sus componentes.



Nota. Topología de Improved LEACH con componentes clave como estaciones base (BS), cabezas de clúster (CH), y nodos sensores normales (NS), incluyendo rutas de enrutamiento (R) y clústeres (CL). Farinango, W. D. (2026). Topología de Improved LEACH y sus componentes

1.5. Justificación

La justificación para evaluar el protocolo Improved-LEACH en una red de sensores inalámbricos subacuáticos en un entorno simulado es la necesidad de superar los retos propios de las UWSN. Estos sistemas, esenciales en el ámbito de IoT y CPS, enfrentan limitaciones de interoperabilidad y eficiencia en la compartición de recursos debido a la diversidad y expansión de las aplicaciones. El protocolo LEACH original, aunque eficiente en energía, necesita adaptarse para mejorar la gestión de la energía y la transmisión de datos en el complejo medio subacuático.

El protocolo Improved-LEACH está diseñado para mitigar la fuerte atenuación de la señal, la interferencia acústica y las restricciones energéticas propias de los entornos subacuáticos. La simulación en OMNeT++ permite evaluar su desempeño sin las barreras logísticas ni los costos de los experimentos en entornos reales, y hace posible un análisis detallado y replicable.La confiabilidad y la latencia de la comunicación, aspectos críticos afectados por la atenuación y la interferencia submarina, pueden ser mejor comprendidos y optimizados a través de este enfoque (Majitha, Christina, Gifta, & Archana, 2022).

La administración efectiva de recursos limitados como el ancho de banda y la capacidad de procesamiento es otra razón que justifica el estudio. Al simular Improved-LEACH, se pueden explorar estrategias para maximizar la eficiencia operativa sin agotar recursos críticos, lo que es esencial para el mantenimiento y la funcionalidad a largo plazo de las UWSN. La escalabilidad también es importante a medida que aumenta el número de sensores; las estructuras jerárquicas que propone Improved-LEACH pueden evaluarse para comprobar su eficacia en la gestión de redes extensas.

Se contribuye al Objetivo 14 de la Agenda 2030 de los Objetivos de Desarrollo Sostenible al permitir un monitoreo más eficiente y sostenible de los océanos (Naciones

Unidas, 2022). El protocolo Improved-LEACH, al optimizar el uso de energía en las UWSN, facilita la recopilación de datos para el análisis de la contaminación marina, la supervisión de ecosistemas y la respuesta a la acidificación. Esto mejora la toma de decisiones basada en evidencia para la conservación y gestión sostenible de recursos marinos. Además, el monitoreo avanzado apoya la pesca sostenible y la protección de la biodiversidad, al proporcionar datos precisos para la regulación y el manejo de las prácticas pesqueras.

La evaluación del protocolo Improved-LEACH en las UWSN está alineada con los objetivos del Eje de Transición Ecológica. Al mejorar la eficiencia y la confiabilidad de la recopilación de datos subacuáticos, la tesis apoyaría la conservación y protección de ecosistemas marinos y su biodiversidad (Pol. 11.1), facilitaría la recuperación y restauración de recursos naturales renovables (Pol. 11.2), y contribuiría a la gestión sostenible de los ecosistemas marino-costeros (E18 y E21). Además, promovería la protección de ecosistemas marino-costeros mediante la recopilación de datos que apoyen esquemas de producción sostenible (F4) y se alinea con la meta de uso sostenible de los recursos naturales (Objetivo 11) (Secretaría Nacional de Planificación, 2021).

2. CAPÍTULO II: ESTADO DEL ARTE

En este capítulo se presentan las bases teóricas y el estado del arte sobre los que se apoya la evaluación del protocolo Improved-LEACH en redes de sensores inalámbricos subacuáticos. Primero se describen las redes de sensores inalámbricos (WSN) y el clustering como técnica para organizar la red y reducir el consumo de energía. Después se aborda el protocolo LEACH, sus variantes y la versión mejorada Improved-LEACH, junto con las particularidades de las redes subacuáticas (UWSN) que condicionan su funcionamiento. Por último, se revisan las herramientas de simulación y las métricas que se emplean para medir el desempeño de estos protocolos.

2.1. Redes de Sensores Inalámbricos (WSNs)

Una WSN es una red compuesta por un gran número de nodos sensores distribuidos en un área específica para monitorizar condiciones ambientales o físicas, como temperatura, sonido, vibración, presión, movimiento o contaminantes (Bhih, Abushiba, & Elashheb, 2023).

Estas redes se utilizan para recopilar datos del entorno, procesarlos y transmitirlos a un punto de recopilación central (Stewart, Fough, Erdogan, & Prabhu, 2023).

Según (Akyildiz, Pompili, & Melodia, 2020), las Redes de Sensores Inalámbricos (WSNs) tienen una presencia omnipresente en nuestra vida cotidiana, transformando la manera en que interactuamos con el mundo que nos rodea. Imagina una ciudad inteligente donde cada esquina, parque y edificio están monitoreados por pequeños nodos sensores. Estos sensores miden la calidad del aire, la temperatura, los niveles de ruido y hasta el tráfico. Los datos recopilados se envían a centros de control donde se analizan para tomar decisiones que mejoran la calidad de vida de los ciudadanos. Por ejemplo, en caso de detectar altos niveles de contaminación, se pueden activar alertas para reducir las emisiones o se pueden ajustar las rutas de tráfico para evitar congestiones, haciendo nuestras ciudades más limpias y eficientes.

En el ámbito agrícola, las WSNs están revolucionando las prácticas de cultivo. Piensa en un agricultor que ahora puede recibir información precisa sobre la humedad del suelo, la temperatura y la salud de sus plantas en tiempo real. Esto le permite optimizar el riego, aplicar fertilizantes de manera más eficiente y detectar plagas o enfermedades a tiempo. Así, no solo se mejora la productividad de las cosechas, sino que también se reduce el uso de recursos y el impacto ambiental. Gracias a las WSNs, la agricultura se vuelve más sostenible, eficiente y productiva, asegurando que podamos alimentar a una población mundial en constante crecimiento (Jawad, Nordin, Gharghan, Jawad, & Ismail, 2017).

2.2. Clustering en WSNs

El clustering en WSNs implica la agrupación de múltiples nodos sensoriales en subconjuntos o clústeres. Cada clúster tiene un nodo líder, conocido como Cluster Head (CH), que coordina la comunicación dentro del clúster y hacia la estación base (Peng, Si, Shen, & Zhao, 2015).

- Eficiencia Energética: Los nodos solo necesitan comunicarse con su CH, que generalmente está más cerca que la estación base, reduciendo así el consumo de energía.
- Escalabilidad: Facilita la gestión de redes grandes, ya que la comunicación y el procesamiento de datos se descentralizan en los clústeres.
- Reducción de la Interferencia: Menor número de transmisiones directas a la estación base, lo que reduce la interferencia y mejora la calidad de la señal.
- Balanceo de Carga: Distribuye uniformemente la carga entre varios nodos, evitando el agotamiento rápido de la batería en nodos individuales.

2.2.1. Estrategias de Formación de Clusters

La formación de clusters en Redes de Sensores Inalámbricos (WSNs) es una técnica esencial para mejorar la eficiencia energética y la gestión de datos dentro de la red. Un cluster se compone de varios nodos sensoriales que se agrupan y comunican con un nodo central denominado Cluster Head (CH), el cual se encarga de recolectar y transmitir los datos al centro de control (Kim, 2015).

Esta estrategia no solo reduce la carga de trabajo individual de los nodos, sino que también optimiza el uso de energía al minimizar las transmisiones directas de larga distancia. La elección de los CHs y la formación de clusters pueden ser realizadas mediante diversos métodos y protocolos, cada uno diseñado para equilibrar el consumo de energía, extender la vida útil de la red y adaptarse a las condiciones cambiantes del entorno.

2.2.1.1. Selección de Cluster Heads (CHs)

La selección de los CHs es crucial para la eficiencia energética y la estabilidad de la red. Los métodos para elegir los CHs pueden variar, pero generalmente se basan en factores como:

- Energía residual del nodo: Prefiere nodos con mayor energía restante para prolongar la vida útil del clúster.
- Proximidad a otros nodos: Selecciona nodos que estén centralmente ubicados para minimizar la distancia de comunicación.
- Proceso aleatorio: Utiliza un enfoque probabilístico para evitar sesgos y garantizar una distribución equitativa del rol de CH.

2.2.1.2. Formación de Clústeres

La formación de clústeres es el proceso mediante el cual los nodos sensoriales se agrupan en subconjuntos o clústeres, y puede ser:

- Estático: La configuración del clúster se define previamente y no cambia, lo que puede simplificar la gestión, pero reduce la flexibilidad.
- Dinámico: Los clústeres se forman periódicamente, lo que permite adaptarse a cambios en la red y distribuir mejor el consumo de energía.

2.2.1.3. Protocolo LEACH y sus Mejoras (Improved LEACH)

LEACH (Low Energy Adaptive Clustering Hierarchy): Es un protocolo pionero que introduce un enfoque rotativo para seleccionar los CHs, con el objetivo de distribuir uniformemente el consumo de energía entre los nodos (Guo, 2016).

Improved LEACH: Es una variante que introduce mejoras en la selección de los CHs y la formación de clústeres para optimizar aún más la eficiencia energética y prolongar la vida útil de la red. Esto puede incluir algoritmos avanzados para la selección de CHs basados en criterios adicionales como la distancia a la estación base, la densidad del clúster y el historial de energía del nodo.

2.2.2. Protocolos de Clustering

En el ámbito de las redes de sensores inalámbricos (WSNs), los protocolos de clustering juegan un papel crucial para gestionar eficientemente los recursos limitados de energía y prolongar la vida útil de la red. Uno de los protocolos más destacados en este campo es LEACH (Low Energy Adaptive Clustering Hierarchy), conocido por su capacidad para minimizar el consumo de energía al distribuir inteligentemente las tareas entre los nodos (C, Sukumaran, & C, 2024).

2.2.2.1. LEACH y sus Variantes

- LEACH (Low Energy Adaptive Clustering Hierarchy)

LEACH es un protocolo autoorganizado y adaptativo donde los nodos de la red toman decisiones autónomas sobre convertirse en "Cluster Heads" (CH) en intervalos de tiempo determinados. Este proceso se basa en probabilidades para equilibrar la carga de trabajo y la energía consumida en toda la red. Los CHs son responsables de crear y gestionar clústeres dentro de la red.

En LEACH, cada nodo decide de manera autónoma si convertirse en CH en función de un umbral de probabilidad establecido. Una vez seleccionados, los CHs organizan los nodos restantes en clústeres y coordinan la comunicación dentro de esos grupos. Este enfoque distribuido ayuda a reducir el consumo de energía al evitar la transmisión directa de datos a largas distancias hacia una estación base central.

2.2.2.2. Variantes de LEACH

Existen varias variantes de LEACH diseñadas para mejorar diferentes aspectos del protocolo base

Tabla 1

Variantes del protocolo LEACH y sus características principales

Variante de LEACH	Descripción	Objetivo Principal
LEACH-C	Introduce un enfoque centralizado donde la estación base selecciona directamente los CHs.	Optimizar el consumo de energía y el rendimiento.
LEACH-B	Equilibra la carga de trabajo entre los CHs para distribuir tareas y responsabilidades equitativamente.	Mejorar la estabilidad y la vida útil de la red.
Improved LEACH	Incorpora mejoras avanzadas como algoritmos para seleccionar CHs y estrategias para gestionar clústeres.	Aumentar la eficiencia en la selección de CHs y la asignación de recursos.

Nota. CHs se refiere a "Cluster Heads" o cabezas de clúster, que son nodos encargados de coordinar la comunicación dentro del clúster.

2.2.3. Importancia del Clustering en las comunicaciones inalámbricas

El clustering es una técnica clave en las redes inalámbricas que organiza los nodos en grupos llamados clústeres, donde un nodo líder, conocido como *Cluster Head* (CH), gestiona las comunicaciones dentro del grupo y envía la información procesada a la estación base. Este enfoque reduce el consumo de energía de los nodos, prolongando la vida útil de la red (Guo, 2016).

2.2.3.1. Reducción de la Carga de Comunicación

En un esquema de clustering, se reducen las necesidades de comunicación directa con la estación base, lo cual requiere altas potencias de transmisión. En su lugar, los nodos envían datos al Cluster Head (CH) local, que generalmente está más cerca (XingGuo, JunFeng, & LinLin, 2016).

Los CHs también pueden realizar la agregación de datos, combinando y comprimiendo la información de múltiples nodos antes de enviarla a la estación base. Esto no solo ahorra energía, sino que también minimiza la congestión en la red.

- Minimización de Transmisiones Directas
- Agregación de Datos
- Comunicaciones Eficientes dentro del Clúster

2.2.3.2. Balance de Carga y Vida Útil de la Red

Un aspecto clave del clustering es el balance de carga y la gestión de la energía para prolongar la vida útil de los nodos y la red en su conjunto.

La rotación periódica de los CHs, como se ve en protocolos como LEACH, asegura que ningún nodo individual consuma rápidamente su energía al permanecer siempre como CH. Esto equilibra la carga de trabajo entre todos los nodos de la red.

Al distribuir la responsabilidad de ser CH y la carga de comunicación asociada, los nodos conservan su energía por más tiempo, lo que resulta en una vida útil más larga para toda la red.

Los protocolos avanzados de clustering también son capaces de adaptarse dinámicamente a cambios en la red, como la falla de nodos o la aparición de nuevos dispositivos. Esto incluye la reconfiguración de clústeres para mantener un equilibrio de carga óptimo y asegurar un funcionamiento eficiente en diferentes condiciones (Bhih, Abushiba, & Elashheb, 2023).

- Vida Útil Prolongada de los Nodos y la Red
- Adaptación a la Dinámica de la Red

2.2.4. Necesidad de implementar cluster en las UWSN

Implementar clusters en Redes de Sensores Submarinas Inalámbricas (UWSN) es esencial para abordar los desafíos únicos del entorno submarino, como la atenuación de señales acústicas y el alto consumo de energía. Agrupar sensores en clusters ayuda a optimizar el consumo energético, ya que los nodos dentro de un cluster se comunican con un nodo cabeza del cluster, que se encarga de transmitir la información a mayores distancias. Esto no solo ahorra energía, sino que también distribuye la carga de trabajo equitativamente entre los nodos, lo que prolonga la vida útil de la red. Además, los clusters permiten una mejor gestión y coordinación de un gran número de nodos, reduciendo el tráfico de control y mejorando la eficiencia operativa de la red (C, Sukumaran, & C, 2024).

La estructura de clusters también mejora la robustez y tolerancia a fallos de la red. Si un nodo cabeza de cluster falla, otros nodos pueden asumir sus funciones, asegurando una mayor resiliencia. Los clusters facilitan la priorización y gestión del tráfico de datos, mejorando la calidad del servicio (QoS) y garantizando que los datos críticos lleguen a su destino a tiempo.

2.2.4.1. Limitaciones Energéticas:

Los nodos en las UWSN suelen estar equipados con baterías de capacidad limitada, y reemplazarlas puede ser costoso y complicado. El clustering ayuda a reducir el consumo de energía al minimizar la cantidad de transmisiones directas a la base de datos. En lugar de que cada nodo envíe datos directamente, los nodos dentro de un clúster envían sus datos a un nodo cabeza de clúster (CH), que luego agrega y transmite los datos recopilados, reduciendo así el consumo energético global (Akyildiz, Pompili, & Melodia, 2020).

2.2.4.2. Propagación de Señales y Comunicación:

El medio acuático presenta una mayor atenuación y retardo de señal en comparación con el aire, lo que dificulta la comunicación directa entre nodos distantes. El clustering facilita una comunicación más eficiente al permitir que los nodos se comuniquen con sus CHs cercanos, que a su vez manejan la transmisión a mayores distancias, mejorando la robustez y fiabilidad de la red (Krishnakumar, 2016).

2.2.4.3. Escalabilidad y Gestión de Red:

A medida que el número de nodos en una UWSN aumenta, gestionar la red se vuelve más complejo. El clustering mejora la escalabilidad de la red, permitiendo una gestión más sencilla y eficiente. Los CHs actúan como puntos centrales para la administración de sus respectivos clústeres, lo que simplifica el control y mantenimiento de la red (Stewart, Fough, Erdogan, & Prabhu, 2023).

2.2.4.4. Balanceo de Carga:

La distribución de tareas y responsabilidades entre los CHs permite un mejor balanceo de carga, previniendo la sobrecarga de cualquier nodo individual. Esto es esencial para

prolongar la vida útil de la red, ya que evita que ciertos nodos agoten sus recursos energéticos más rápido que otros (Yang, Guo, Dong, & Xia, 2018).

2.2.4.5. Agregación de Datos:

El clustering permite la agregación de datos, donde los CHs combinan los datos recibidos de los nodos miembros antes de enviarlos a la base de datos. Esta agregación no solo reduce la cantidad de datos que deben transmitirse, sino que también puede mejorar la calidad y utilidad de la información al filtrar redundancias y errores (Abushiba, Johnson, Alharthi, & Wright, 2018).

2.2.4.6. Reducción de Colisiones y Retransmisiones:

En un entorno subacuático, las colisiones de paquetes y las retransmisiones son problemas comunes debido a la limitada capacidad del canal y las altas tasas de error. El clustering organiza mejor la comunicación, reduciendo la probabilidad de colisiones y la necesidad de retransmisiones, lo que resulta en una comunicación más eficiente y confiable (Haque, 2022).

2.2.4.7. Adaptabilidad y Flexibilidad:

Los algoritmos de clustering pueden adaptarse dinámicamente a los cambios en la topología de la red y las condiciones del entorno acuático. Esto es crucial para las UWSN, donde las condiciones pueden variar significativamente debido a factores como corrientes, temperatura y salinidad. La capacidad de reconfigurarse sobre la marcha permite a la red mantener un rendimiento óptimo incluso en condiciones adversas (Singh & Kaur, 2022).

2.3. Métricas de Evaluación en WSNs

Parámetros cuantitativos y cualitativos utilizados para medir y evaluar el rendimiento y la eficiencia de estas redes. Estas métricas ayudan a determinar cómo se comportan las redes

bajo diferentes condiciones y configuraciones, y son cruciales para el diseño, la optimización y la comparación de diferentes estrategias y tecnologías en WSNS y UWSNS (Akyildiz, Pompili, & Melodia, 2020).

2.3.1. Métricas de Rendimiento

Las métricas de rendimiento son indicadores claves utilizados para evaluar la eficiencia y efectividad de las redes de sensores inalámbricos (WSNs). Estas métricas permiten medir aspectos como el consumo de energía, la confiabilidad en la entrega de datos, la latencia, el throughput y la vida útil de la red, entre otros.

Tabla 2

Métricas de Desempeño en Redes de Sensores Inalámbricos (WSNs)

Métrica	Descripción	Importancia	Unidad de Medida
Consumo de Energía	Cantidad de energía utilizada por los nodos en transmisión, recepción, procesamiento y reposo.	Es crucial en WSNs porque los nodos dependen de baterías limitadas.	Joules (J) o energía consumida por tarea.
Tasa de Entrega de Paquetes (PDR)	Proporción de paquetes enviados que llegan exitosamente al destino.	Indica la fiabilidad y eficiencia del protocolo de enrutamiento.	Porcentaje (%).
Latencia (Retardo)	Tiempo total que tarda un paquete en viajar del origen al destino.	Esencial en aplicaciones en tiempo real donde los retrasos pueden ser críticos.	Milisegundos (ms).
Vida Útil de la Red	Tiempo durante el cual la red puede operar eficientemente antes de agotar baterías.	Determina la sostenibilidad y durabilidad de la red.	Tiempo (segundos, horas, días).
Throughput (Rendimiento)	Cantidad de datos transmitidos exitosamente por unidad de tiempo.	Mide la capacidad de la red para manejar grandes volúmenes de datos.	Bits por segundo (bps) o paquetes por segundo.
Jitter (Variabilidad del Retardo)	Variación en el tiempo de llegada de los paquetes.	Crítico en aplicaciones multimedia y de voz, donde afecta la calidad del servicio.	Desviación estándar del tiempo de retardo.
Tasa de Pérdida de Paquetes	Proporción de paquetes perdidos durante la transmisión.	Alta pérdida puede indicar problemas de protocolo o interferencias en la red.	Porcentaje (%).
Capacidad de Enlace	Capacidad máxima del enlace de comunicación en la red.	Determina el rendimiento máximo que puede alcanzar la red.	Bits por segundo (bps).

Cobertura de la Red	Proporción del área total que está cubierta por la red de sensores.	Garantiza que la red pueda monitorear toda el área de interés.	Porcentaje (%).

Notas: Las métricas están organizadas de manera que faciliten su consulta según las necesidades del diseño de la red, estas métricas son relevantes tanto para diseñar redes eficientes como para evaluar el rendimiento en tiempo real.

2.3.2. Métricas Específicas de UWSNs

En las Redes de Sensores Inalámbricos Subacuáticas (UWSNs), las métricas de rendimiento adquieren características particulares debido a los desafíos únicos del entorno submarino. La propagación de señales acústicas, la movilidad de los nodos y las condiciones cambiantes del agua requieren un enfoque especializado para evaluar el funcionamiento de estas redes. A continuación, se presentan las métricas más relevantes en este contexto.

- Tasa de Pérdida de Paquetes: Importante en UWSNs debido a las dificultades de la comunicación bajo el agua.
- Retardo de Comunicación: Debido a la baja velocidad de propagación de las señales acústicas bajo el agua, el retardo es una métrica crucial en UWSNs.
- Robustez en Entornos Submarinos: Habilidad de la red para manejar las dinámicas y desafíos específicos del entorno submarino, como las corrientes oceánicas y la movilidad de los nodos.

2.3.3. Herramientas y Técnicas de Simulación

Las herramientas y técnicas de simulación son fundamentales para el diseño, análisis y optimización de redes de sensores inalámbricos (WSNs) y subacuáticas (UWSNs).

- Software de Simulación: Herramientas como OMNeT++, NS2/NS3, y Aqua-Sim, específicamente diseñadas para simular WSNs y UWSNs.
- Modelado de la Red: Técnicas para representar con precisión la topología de la red, los patrones de tráfico y las características del entorno.

- Análisis de Datos: Uso de herramientas estadísticas y de visualización para interpretar los resultados de las simulaciones, identificar tendencias y evaluar el rendimiento frente a las métricas establecidas.
- Experimentación Virtual: Creación de escenarios de red realistas en un entorno de simulación para probar y evaluar diferentes configuraciones, protocolos y estrategias de gestión.

2.3.3.1. OMNET++

Es un framework de simulación modular y extensible que se utiliza ampliamente en la investigación y desarrollo de WSNs y UWSNs. Proporciona una plataforma flexible para modelar y simular sistemas de comunicaciones, sistemas distribuidos y otros sistemas complejos. OMNeT++ permite la creación de modelos detallados de los nodos sensores, los protocolos de comunicación y los entornos de despliegue, lo que facilita la evaluación del rendimiento y la eficiencia de las redes (Birajdar & Solapure, 2017).

- Evaluación de Métricas: OMNeT++ permite la evaluación de todas las métricas mencionadas mediante la creación de modelos de red detallados. Es especialmente útil para estudios de eficiencia energética y QoS debido a su capacidad para modelar con precisión el comportamiento de los nodos y las comunicaciones.
- Ventajas: Flexibilidad y extensibilidad, interfaz gráfica avanzada, y una amplia biblioteca de módulos.

2.3.3.2. NS-3

NS-3 es un simulador de red de propósito general que permite la simulación de redes de comunicaciones a nivel de paquete. Aunque no está específicamente diseñado para WSN y UWSN, se utiliza ampliamente en la investigación de estos tipos de redes por su flexibilidad y

su capacidad para modelar distintos aspectos de las mismas. Ofrece modelos detallados y precisos para simular protocolos de red, tráfico de datos, dispositivos de red y escenarios de despliegue.

- Evaluación de métricas: NS-3 permite evaluar métricas como la capacidad de la red, la latencia y la tasa de entrega de datos.
- Ventajas: Ampliamente utilizado y bien documentado, con una gran cantidad de scripts y escenarios predefinidos.

2.4. Métricas de Evaluación

En esta sección se describen las métricas con las que se evaluará el desempeño del protocolo Improved-LEACH en la red de sensores subacuáticos. Cada una aporta una visión distinta del comportamiento de la red: la energía disponible en los nodos, su tiempo de operación, la cantidad de datos entregados y la organización de los clústeres. En conjunto permiten comparar Improved-LEACH frente a LEACH y valorar su idoneidad en el entorno subacuático.

2.4.1. Energía residual

La energía residual es la energía que conserva cada nodo después de cada ronda de operación; seguir su evolución muestra qué tan rápido se agotan los nodos y cómo Improved-LEACH reparte la carga energética entre ellos (Heinzelman, Chandrakasan y Balakrishnan, 2000).

- Importancia de medir el consumo de energía

El consumo de energía es una preocupación central en las redes de sensores inalámbricos subacuáticos (UWSNs) debido a la limitada capacidad de las baterías de los nodos sensores.

Es esencial gestionar eficientemente la energía para prolongar la vida útil de la red, ya que en entornos subacuáticos, recargar o reemplazar las baterías puede ser extremadamente difícil o incluso imposible. Una red eficiente energéticamente asegura que los nodos puedan operar durante períodos más largos, lo cual es vital para aplicaciones de monitoreo continuo (Kannappan & Bommi, 2022).

- Cómo se medirá el consumo de energía en las simulaciones

Para medir el consumo de energía en las simulaciones utilizando OMNeT++ e INET, configuraremos un modelo de energía que rastree cuánta energía consume cada nodo durante la simulación. Iniciaremos con un nivel de energía predeterminado para cada nodo y monitorizaremos el consumo a lo largo del tiempo, registrando cómo el protocolo IMPROVED-LEACH afecta la eficiencia energética de los nodos (Stewart, Fough, Erdogan, & Prabhu, 2023).

2.4.2. Tasa de Entrega de Paquetes (PDR)

- Importancia de la fiabilidad en la transmisión de datos

La Tasa de Entrega de Paquetes (PDR) mide la fiabilidad de un protocolo de enrutamiento al transmitir datos. En una red de sensores, es crucial que los datos recogidos lleguen a su destino sin pérdidas significativas, especialmente en aplicaciones críticas donde cada dato cuenta. Un alto PDR indica que la red es confiable y que el protocolo está funcionando adecuadamente (Korpeoglu & Tan, 2015).

- Método de cálculo del PDR en las simulaciones

En OMNeT++ e INET, calcularemos el PDR registrando la cantidad de paquetes enviados por los nodos y los paquetes que llegan correctamente a su destino (Korpeoglu & Tan, 2015).

Usaremos la siguiente fórmula para calcular el PDR:

Tasa de entrega de paquetes (PDR)

$$PDR = \left(\frac{Numero\ de\ Paquetes\ Recibidos}{Numero\ de\ Paquetes\ Enviados}\right) \times 100$$

Esto nos dará una idea clara de cuán eficaz es el protocolo IMPROVED-LEACH en términos de transmisión de datos.

2.4.3. Latencia Promedio

Relevancia de la latencia en aplicaciones en tiempo real

La latencia promedio mide el tiempo que tarda un paquete de datos en viajar desde su origen hasta su destino. Esta métrica es especialmente importante en aplicaciones en tiempo real, donde los retrasos pueden afectar la calidad del servicio. Por ejemplo, en monitoreo de condiciones ambientales críticas, los datos retrasados pueden llevar a decisiones tardías o incorrectas (Chafi, Abbad, Benhaoua, & Benyamina, 2023).

- Proceso para medir la latencia promedio

Para medir la latencia promedio en OMNeT++ e INET, registraremos el tiempo en que se envía cada paquete y el tiempo en que se recibe. La diferencia entre estos dos tiempos nos dará la latencia para cada paquete (Kumar, Jain, & Tiwari, 2019).

Calcularemos la latencia promedio sumando todas estas latencias y dividiéndolas por el número total de paquetes recibidos:

Latencia promedio

$$Latencia\ Promedio = \frac{\sum(Tiempo\ de\ Recepcion - Tiempo\ de\ Envio)}{Numero\ de\ Paquetes\ Recibidos}$$

Esta métrica nos permitirá evaluar cuán rápido y eficiente es el protocolo IMPROVED-LEACH en transmitir datos.

2.4.4. Vida útil de la red: FND, HND y LND

La vida útil de la red se cuantifica con tres indicadores: FND (First Node Dies), el instante en que muere el primer nodo; HND (Half Nodes Die), cuando se agota la mitad de los nodos; y LND (Last Node Dies), cuando deja de funcionar el último nodo. Estos tres puntos describen cómo se degrada la red a lo largo del tiempo (Handy, Haase y Timmermann, 2002).

- Definición y importancia de la vida útil de la red

La vida útil de la red se refiere al tiempo que los nodos pueden operar antes de que sus baterías se agoten. Es una métrica crucial para entender la sostenibilidad de la red y la eficiencia del protocolo de enrutamiento. Una mayor vida útil significa que la red puede funcionar durante más tiempo sin necesidad de intervención, lo cual es especialmente valioso en entornos difíciles de acceder como los subacuáticos (Krishnakumar, 2016).

- Cómo se evaluará la vida útil de la red en las simulaciones

En OMNeT++ e INET, monitorizaremos continuamente el nivel de energía de cada nodo. Registraremos el tiempo en que cada nodo se queda sin energía. Evaluaremos la vida útil de la red en términos del tiempo hasta que el primer nodo se agote y también cuando un porcentaje significativo (por ejemplo, el 50%) de los nodos se queden sin energía. Esto nos ayudará a comprender cómo el protocolo IMPROVED-LEACH gestiona la energía y distribuye la carga de trabajo en la red.

2.4.5. Throughput

El throughput es la cantidad de datos que la red entrega correctamente al sink por unidad de tiempo, normalmente expresada en bits o paquetes por segundo. Refleja la capacidad efectiva de la red para transportar la información que recolectan los nodos.

En las simulaciones se calculará dividiendo el total de datos recibidos en el sink entre el tiempo de simulación, lo que permite comparar la capacidad de transporte de Improved-LEACH frente a LEACH.

2.4.6. Distancia promedio CH–Sink

La distancia promedio entre los cluster heads (CH) y el sink es el promedio de las distancias que separan a cada CH del nodo sumidero en cada ronda. Como el costo energético de transmitir crece con la distancia, esta métrica ayuda a explicar el consumo de energía y la calidad de la selección de CH (Heinzelman, Chandrakasan y Balakrishnan, 2000).

En las simulaciones se obtendrá registrando, en cada ronda, las posiciones de los CH y del sink, y promediando las distancias resultantes.

3. CAPÍTULO III: METODOLOGIA

El presente capítulo describe el proceso metodológico adoptado para diseñar, implementar y evaluar, mediante simulación, los protocolos LEACH e Improved LEACH basado en energía residual en una Red de Sensores Inalámbricos Subacuáticos (UWSN). El capítulo abarca exclusivamente el cómo se construyó la herramienta experimental; el análisis de los resultados obtenidos se presenta en el Capítulo IV.

El trabajo se organiza en cinco fases secuenciales: Fase I (Planificación), Fase II (Diseño del entorno), Fase III (Implementación en OMNeT++), Fase IV (Diseño experimental) y Fase V (Procesamiento y validación de resultados). Cada fase tiene entradas, salidas y criterios de completitud definidos que garantizan trazabilidad y reproducibilidad del proceso experimental.

3.1. Metodología de Desarrollo

La gestión del desarrollo de la simulación se realizó mediante un enfoque ágil basado en el método Kanban, que permite visualizar el estado de cada actividad técnica, priorizar el trabajo pendiente y detectar bloqueos de forma oportuna. Las actividades se organizaron en un tablero con cuatro estados: Backlog (tareas pendientes por especificar), En progreso (implementación activa), En verificación (pruebas de corrección del modelo) y Hecho (tarea cerrada con evidencia de funcionamiento). Este método no define una secuencia temporal rígida, sino que complementa la estructura por fases garantizando que ninguna tarea avance al siguiente estado sin haber cumplido su criterio de completitud.

Sobre esta gestión se aplica una estrategia de validación incremental en tres niveles, orientada a asegurar que las diferencias de desempeño observadas en el Capítulo IV sean atribuibles al algoritmo y no a errores de integración:

- Nivel I — Conectividad base: se verifica que todos los nodos entreguen paquetes al sink sin pérdidas atribuibles a errores de configuración, con el clustering desactivado.
- Nivel II — Clustering funcional: se valida el ciclo completo de selección de CH, formación de clúster y entrega CH → sink para ambos protocolos bajo un escenario de referencia de 50 nodos.
- Nivel III — Escalamiento: se confirma que las corridas con 100, 250, 500 y 1000 nodos concluyen sin caídas del simulador ni corrupción de los archivos de resultados.
- Solo las corridas que superan los tres niveles de verificación se incorporan al análisis comparativo del Capítulo IV.

3.2. Fase I. Planificación del Estudio de Simulación

La Fase I establece el marco de referencia del estudio experimental: define el problema que motiva la simulación, determina las variables que se controlan y las que se observan, delimita los supuestos que acotan el modelo y fija los criterios que determinan cuándo el entorno de simulación es válido para producir resultados comparativos.

3.2.1. Definición del Problema de Simulación

Las Redes de Sensores Inalámbricos Subacuáticos (UWSN) operan en un entorno acústico con restricciones severas: ancho de banda limitado (típicamente 1–30 kHz), alta latencia de propagación (≈ 1 500 m/s frente a 3×10^8 m/s en el medio terrestre), absorción acústica dependiente de la frecuencia y nodos alimentados por baterías no recargables (Akyildiz et al., 2005). En este contexto, el protocolo LEACH selecciona los cluster heads (CH) mediante un umbral probabilístico que no considera la energía residual de cada nodo, lo que puede derivar en que nodos con batería baja sean elegidos CH con la misma probabilidad

que nodos con batería plena, acelerando su agotamiento y reduciendo la vida útil de la red (Heinzelman et al., 2000).

El protocolo Improved LEACH basado en energía residual introduce una modificación en la función de umbral de selección de CH, ponderando la probabilidad de elección por la relación entre la energía residual del nodo y su energía inicial, de modo que los nodos con mayor batería disponible tienen mayor probabilidad de asumir el rol de CH. El presente estudio evalúa, mediante simulación en OMNeT++, si esta modificación produce una mejora estadísticamente significativa en las métricas de eficiencia energética, supervivencia de nodos y tasa de entrega de paquetes, en un entorno UWSN tridimensional con densidades de 50, 100, 250, 500 y 1000 nodos.

3.2.2. Variables de Entrada y Variables de Salida

Las variables de entrada definen las condiciones del escenario UWSN, los parámetros del protocolo y los parámetros de la simulación. Todas se mantienen idénticas entre las corridas de LEACH e Improved LEACH, salvo la variable manipulada (el protocolo), garantizando que cualquier diferencia en las variables de salida sea atribuible exclusivamente al algoritmo.

Tabla 3

Variables de entrada del estudio de simulación

Categoría	**Variable**	**Valor / Rango**
Escenario	Dimensiones del volumen 3D	1 000 × 1 000 × 1 000 m
Escenario	Número de nodos sensores (N)	50 / 100 / 250 / 500 / 1 000
Escenario	Posición del nodo sink	Superficie, centro (500, 500, 0) m
Protocolo	Protocolo evaluado (variable manipulada)	LEACH / Improved LEACH
Protocolo	Probabilidad objetivo de CH (pCH)	0.05 (5 %)
Energía	Energía inicial por nodo (E_0)	1.0 J
Energía	Umbral de muerte de nodo (Eth)	0.05 J
Temporalidad	Duración de ronda (Tronda)	30 s
Canal	Tasa de transmisión (B)	10 kbps
Aleatoriedad	Semillas de simulación	a–j (10 semillas, seed-set 0–9)

Nota. CH = cabeza de clúster; E_0= energía inicial por nodo; E_{th}= umbral de muerte; T_{ronda}= duración de ronda; B = tasa de transmisión. Los parámetros se mantuvieron constantes entre LEACH e I-LEACH, excepto el protocolo evaluado. Se ejecutaron 10 repeticiones con semillas de 0 a 9. Elaboración propia.

Las variables de salida son las métricas registradas por el simulador en cada corrida y procesadas posteriormente en Python para la comparación entre protocolos:

Tabla 4

Variables de salida (métricas) del estudio de simulación

Métrica	**Descripción**	**Unidad**
Energía residual promedio	Promedio de energía disponible en los nodos vivos al cierre de cada ronda.	J
Nodos vivos por ronda	Número de nodos con Eres > Eth al inicio de cada ronda.	Nodos
FND (First Node Death)	Ronda en que muere el primer nodo de la red.	Ronda
HND (Half Node Death)	Ronda en que ha muerto el 50 % de los nodos.	Ronda
LND (Last Node Death)	Ronda en que muere el último nodo de la red.	Ronda
PDR (Packet Delivery Ratio)	Razón paquetes recibidos en sink vs. paquetes generados.	%
Throughput	Volumen de datos útiles recibidos en el sink por unidad de tiempo.	bps
Latencia promedio	Retardo promedio de entrega sensor → CH → sink.	ms

Nota. FND = *First Node Dies* (primera muerte de nodo); HND = *Half Nodes Die* (muerte del 50 % de los nodos); LND = *Last Node Dies* (muerte del último nodo); PDR = *Packet Delivery Ratio* (relación porcentual entre los paquetes recibidos en el *sink* y los paquetes generados o transmitidos). La energía residual promedio corresponde al promedio de la energía disponible en los nodos al finalizar cada ronda. Las métricas FND, HND y LND se expresan en número de rondas; el PDR, en porcentaje (%); el *throughput*, en bits por segundo (bps); y la latencia, en milisegundos (ms). Elaboración propia.

3.2.3. Supuestos y Limitaciones del Entorno Simulado

Con el propósito de acotar el modelo y mantener su coherencia interna, se adoptan los siguientes supuestos:

- Los nodos sensores se distribuyen aleatoriamente en el volumen 3D al inicio de la simulación y permanecen estáticos durante toda la corrida (modelo de red estática).
- El nodo sink dispone de energía ilimitada y actúa exclusivamente como destino final de los paquetes; no participa en la selección de CH ni en la formación de clústeres.
- El canal de comunicación es el canal acústico subacuático modelado con propagación en espacio libre (FreeSpacePathLoss); no se modelan efectos multipath, variabilidad temporal del canal ni ruido ambiental variable.
- El consumo energético de procesamiento de los nodos se considera despreciable frente al consumo de transmisión y recepción del radio acústico.
- La única diferencia entre las corridas de LEACH e Improved LEACH es la función de selección del CH; todos los demás parámetros (escenario, canal, energía, tráfico, semilla) son idénticos.
- Se utiliza semilla fija por corrida para garantizar reproducibilidad exacta.

Las principales limitaciones del modelo son las siguientes: los resultados son válidos dentro del espacio de parámetros definido; la extrapolación a escenarios con movilidad de nodos, ruido acústico variable, múltiples sinks o modelos de canal más complejos requeriría modelos adicionales. Asimismo, el estudio no contempla seguridad de la información, tolerancia a fallas de hardware ni efectos de colisión en la capa MAC.

3.2.4. Criterios de Verificación y Validación del Entorno de Simulación

Los siguientes criterios, definidos a priori, deben cumplirse antes de incorporar los resultados de cualquier corrida al análisis comparativo del Capítulo IV. Su verificación efectiva se documenta en la Fase V (Sección 3.6.5).

Tabla 5

Criterios de aceptación del entorno de simulación

Aspecto verificado	Criterio de aceptación	Evidencia esperada
Conectividad base (Nivel I)	100 % de paquetes entregados al sink sin pérdidas por configuración.	Log de ejecución del escenario mínimo.
Clustering funcional (Nivel II)	Ciclo ADV→JOIN→TDMA→DATA→AGG verificado para ambos protocolos con 50 nodos.	Trazas de selección de CH y archivo .vec por ronda.
Escalamiento (Nivel III)	Sin caídas del simulador con N = 100, 250, 500, 1 000.	Logs de las corridas de barrido de parámetros.
Consistencia energética	$E_i(r+1) \leq E_i(r)$ para todo nodo i y ronda r; ningún nodo recupera energía.	Script Python de verificación sobre el CSV exportado.
Repetibilidad	Dos corridas con el mismo seed-set producen archivos .sca idénticos.	Comparación de hash MD5 de los archivos .sca.

Nota. CH = *Cluster Head* (cabeza de clúster); ADV = *Advertisement*; JOIN = mensaje de asociación al clúster; TDMA = *Time Division Multiple Access*; DATA = paquete de datos; AGG = datos agregados; *sink* = nodo sumidero. Los criterios de aceptación fueron establecidos antes de la ejecución de las simulaciones y se aplicaron de forma uniforme para LEACH e I-LEACH. Elaboración propia.

3.3. Fase II. Diseño del Entorno de Simulación UWSN

La Fase II define la arquitectura tecnológica y los modelos físicos que constituyen el entorno de simulación. Se establece la plataforma de software, las características del escenario subacuático tridimensional, la configuración temporal de las corridas, y los modelos matemáticos que gobiernan el canal acústico, el tráfico de datos y el consumo energético de los nodos. Todos los modelos descritos en esta fase se implementan directamente en el simulador durante la Fase III.

3.3.1. Plataforma de Simulación y Herramientas Utilizadas

El entorno tecnológico del estudio se divide en dos componentes con responsabilidades claramente separadas. El primer componente es el motor de simulación: OMNeT++ 6.0.3 junto con el framework inetmanet-4.x y el módulo de canal acústico subacuático UanModel (Ariza

Quintana, s.f.) ejecuta el modelo de red, los protocolos y el modelo energético, y registra los eventos de cada corrida en archivos de resultados binarios (.sca para valores escalares y .vec para series temporales por ronda). El segundo componente es el entorno de análisis: Python 3.x, con las librerías pandas, NumPy y matplotlib, consolida los archivos exportados de múltiples corridas y semillas, calcula estadísticos descriptivos e intervalos de confianza, y genera las gráficas comparativas entre LEACH e Improved LEACH.

Esta separación de responsabilidades se adopta porque el volumen de datos generado por cinco escenarios, dos protocolos y diez semillas —un total de 100 corridas— hace impráctico el análisis manual o visual dentro del entorno de OMNeT++; Python permite automatizar completamente ese procesamiento. A continuación se describen las herramientas utilizadas.

3.3.1.1. Ubuntu

El sistema operativo anfitrión es Ubuntu 22.04 LTS de 64 bits. Linux es la plataforma recomendada para OMNeT++ porque simplifica la gestión de dependencias de compilación (g++, make, librerías Qt5) y garantiza la estabilidad del simulador en corridas largas con alta densidad de nodos.

3.3.1.2. OMNeT++

OMNeT++ 6.0.3 es el simulador de eventos discretos de código abierto que actúa como motor de ejecución. Proporciona el entorno de compilación C++, la cola de eventos discretos, el visor gráfico interactivo Qtenv, y la herramienta de línea de comandos opp_scavetool para exportar resultados. Los modelos de nodo y los protocolos se definen mediante archivos NED (topología) y C++ (comportamiento de los módulos).

3.3.1.3. Framework inetmanet-4.x y módulo UanModel

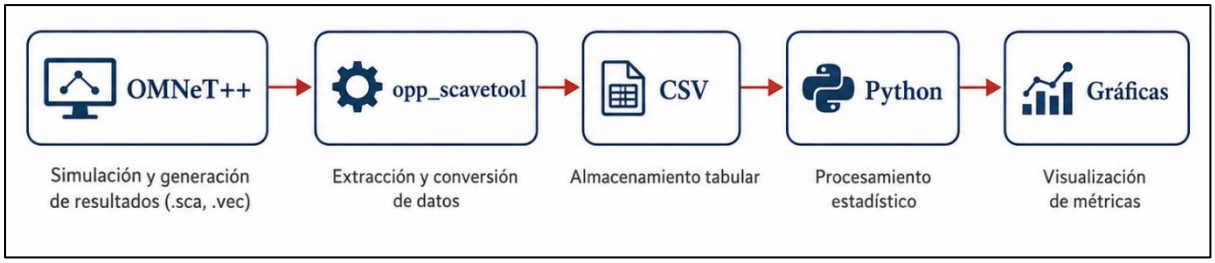
El framework inetmanet-4.x extiende OMNeT++ con modelos de nodo inalámbrico, modelos de enlace y el módulo UanModel que implementa el canal de comunicación acústico subacuático con su modelo de propagación, retardo y atenuación. Los protocolos LEACH e Improved LEACH se implementan como módulos de enrutamiento dentro de este framework, en la ruta src/inet/wirelesspan/routing/leach/.

3.3.1.4. Python

Python 3.x con pandas, NumPy y matplotlib actúa como entorno de postprocesamiento y visualización. Una vez exportados los archivos de resultados de OMNeT++ a formato CSV mediante opp_scavetool, los scripts Python leen, filtran, agregan (por semilla, por escenario y por protocolo), calculan estadísticos e intervalos de confianza, y producen las figuras del Capítulo IV. El flujo completo de procesamiento se describe en la Sección 3.6.2.

Figura 2.

Arquitectura general del entorno de simulación



Nota. Los resultados generados por OMNeT++ (*.sca* y *.vec*) se exportan a formato CSV mediante **opp_scavetool**. Posteriormente, scripts desarrollados en Python realizan el filtrado, la agregación por semilla y escenario, el cálculo de estadísticas descriptivas e intervalos de confianza, y la generación de las figuras y tablas presentadas en el Capítulo IV. Elaboración propia.

3.3.2. Definición del Escenario UWSN

El escenario de simulación representa un volumen tridimensional subacuático de 1 000 m × 1 000 m × 1 000 m (largo × ancho × profundidad), dimensiones representativas de despliegues reales de UWSN en zonas de monitoreo costero y de fondos marinos. Los N nodos

sensores se distribuyen de manera aleatoria y uniforme dentro de ese volumen al inicio de cada simulación, según la semilla correspondiente a la corrida.

El nodo sink (estación base / boya de superficie) se ubica en la posición central de la superficie del escenario, en las coordenadas (500, 500, 0) m, y dispone de energía ilimitada. La naturaleza tridimensional del escenario implica que la distancia entre cualquier par de nodos —incluida la distancia de cada sensor al sink— se calcula como distancia euclidiana en tres dimensiones, lo que afecta directamente el retardo de propagación acústica y el consumo energético de cada enlace (ecuaciones presentadas en la Sección 3.3.4).

3.3.3. Configuración Temporal de la Simulación

El tiempo total de simulación (Tsim) no está fijado como un límite absoluto que detiene la corrida independientemente del estado de la red. La condición de finalización real es el agotamiento energético de todos los nodos: la simulación continúa hasta que el último nodo muerto agota su energía (evento LND), momento en el que OMNeT++ no tiene más eventos en cola y la corrida concluye naturalmente. El valor de sim-time-limit configurado en omnetpp.ini actúa únicamente como protección del simulador para evitar corridas infinitas en caso de error; en condiciones normales de operación, este límite no es alcanzado porque todos los nodos se agotan energéticamente antes.

Cada corrida se estructura en rondas de duración Tronda = 30 s. Una ronda comprende tres fases internas: la fase de anuncio y selección de CH (mensajes ADV, ≈ 2 s), la fase de asociación y planificación TDMA (mensajes JOIN y REPLY, ≈ 3 s), y la fase steady-state de transmisión de datos (mensajes DATA y AGG_DATA, ≈ 25 s). El número total de rondas ejecutadas por corrida no es fijo; depende de cuántas rondas tardan en agotarse los N nodos sensores del escenario.

Para el control de la aleatoriedad, se definen 10 repeticiones por escenario/protocolo, identificadas con las etiquetas a–j (seed-set = 0–9 en omnetpp.ini). Las semillas controlan la distribución espacial inicial de los nodos y todas las secuencias de decisión estocásticas durante la corrida, garantizando que los resultados sean distintos entre semillas (variabilidad interrun) pero exactamente reproducibles para una semilla dada (determinismo intrarun).

3.3.4. Modelo Matemático del Escenario de Simulación

Con el propósito de unificar la notación matemática del capítulo y asegurar la coherencia entre la descripción metodológica y la implementación en OMNeT++, la presente sección agrupa todas las ecuaciones utilizadas en el estudio, organizadas por dominio: geometría del escenario, temporalidad, protocolo LEACH, protocolo Improved LEACH, modelo energético, métricas de red y estadística descriptiva. Cada ecuación está numerada, las variables se definen inmediatamente a continuación y se indica la referencia bibliográfica correspondiente.

3.3.4.1. Geometría del Escenario

El volumen del escenario tridimensional está dado por:

Ecuación 1

Volumen del escenario tridimensional

$$V = L_x \times L_y \times L_z \qquad (1)$$

Donde:

L_x = 1 000 m

L_γ = 1 000 m

Lz = 1 000 m

Son las dimensiones del volumen en los ejes X, Y y Z respectivamente.

Las coordenadas de cada nodo sensor i se generan mediante distribución uniforme en el volumen (Akyildiz et al., 2005):

Ecuación 2
Distribución uniforme de las coordenadas de los nodos

$$(x_i, y_i, z_i) \sim U(0, L_x) \times U(0, L_y) \times U(0, L_z) \quad (2)$$

donde U(a, b) denota distribución uniforme continua en el intervalo [a, b].

La distancia euclidiana tridimensional entre el nodo i y el nodo j (o el sink s) es (Akyildiz et al., 2005):

Ecuación 3
Distancia euclidiana tridimensional

$$d(i,j) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2} \quad (3)$$

Esta distancia se utiliza para determinar la asociación de cada nodo al CH más cercano y para calcular el retardo de propagación acústica.

3.3.4.2. Temporalidad de la Simulación

El número de rondas ejecutadas en una corrida no es fijo sino que está determinado por el tiempo de vida de la red (Life Time, LT):

Ecuación (4). Número efectivo de rondas

Ecuación 4

Número efectivo de rondas

$$R_{\text{efectivo}} = \left\lfloor \frac{LT}{T_{\text{ronda}}} \right\rfloor \qquad (4)$$

Donde LT es el tiempo en segundos transcurrido desde el inicio de la simulación hasta el evento LND, y T_ronda = 30 s es la duración de cada ronda.

Cada ronda r se estructura internamente en:

Ecuación 5

Duración de la ronda

$$T_{\text{ronda}} = T_{\text{anuncio}} + T_{\text{unión}} + T_{\text{steady}} \qquad (5)$$

Donde T_anuncio ≈ 2 s corresponde a la difusión de mensajes ADV por los CH; T_unión ≈ 3 s corresponde al intercambio de mensajes JOIN y TDMA; y T_steady ≈ 25 s corresponde a la fase de transmisión de datos DATA y AGG_DATA.

3.3.4.3. Protocolo LEACH — Función de Umbral

En LEACH, durante la ronda r, cada nodo n que no ha sido CH en las últimas ⌊1/pCH⌋ rondas calcula un umbral T(n, r) y genera un número aleatorio uniforme u ~ U(0,1). Si u < T(n, r), el nodo se auto-declara CH para esa ronda (Heinzelman et al., 2000):

Ecuación 6

Umbral de elección de cabeza de clúster (LEACH)

$$T(n,r) = \begin{cases} \dfrac{p_{\text{CH}}}{1 - p_{\text{CH}} \cdot \left(r \bmod \left\lfloor \frac{1}{p_{\text{CH}}} \right\rfloor\right)}, & \text{si } n \notin G(r) \\ 0, & \text{si } n \in G(r) \end{cases} \qquad (6)$$

Donde pCH = 0.05 es la probabilidad objetivo de CH; r es el número de ronda actual; G(r) es el conjunto de nodos que ya han sido CH en las últimas ⌊1/pCH⌋ = 20 rondas. El denominador garantiza que, en promedio, exactamente pCH × N nodos sean elegidos CH en cada ronda.

3.3.4.4. Protocolo Improved LEACH - Umbral Ponderado por Energía Residual

Improved LEACH, en la variante implementada en este estudio, modifica únicamente la función de umbral de LEACH incorporando la energía residual del nodo como factor de ponderación. Un nodo con mayor energía residual tiene mayor probabilidad de convertirse en CH; un nodo con poca energía tiene una probabilidad proporcionalmente reducida. La función de umbral modificada es:

Ecuación 7
Umbral de elección de cabeza de clúster (Improved-LEACH)

$$T'(n,r) = T(n,r) \cdot \frac{E_{\text{res}}(n,r)}{E_0} \qquad (7)$$

Donde T(n,r) es el umbral estándar de LEACH definido en la Ecuación (6); Eres(n,r) es la energía residual del nodo n al inicio de la ronda r (en joules); E_0 = 1.0 J es la energía inicial de todos los nodos; y el factor [Eres(n,r)/E_0] ∈ [0,1] es el peso de ponderación. Esta modificación garantiza que los nodos con batería próxima al nivel inicial asuman el rol de CH con mayor frecuencia, distribuyendo el gasto energético de manera más equitativa entre los nodos de la red.

Nótese que Improved LEACH mantiene idéntica a LEACH toda la estructura de mensajes, la formación de clústeres, la comunicación y la fase steady-state; la única diferencia algorítmica es la Ecuación (7) en lugar de la Ecuación (6).

3.3.4.5. Modelo Energético

El modelo energético adoptado es el modelo de radio de primer orden de Heinzelman et al. (2000), en el que el costo energético de transmitir k bits a una distancia d es:

Ecuación 8
Energía de transmisión (modelo de radio de primer orden)

$$E_{\mathrm{TX}}(k,d) = E_{\mathrm{elec}} \cdot k + \varepsilon_{\mathrm{fs}} \cdot k \cdot d^2 \qquad (8)$$

Donde Eelec = 50 nJ/bit es la energía consumida por la electrónica del transceptor; εfs = 10 pJ/(bit·m²) es el coeficiente de amplificación en espacio libre; k es el tamaño del paquete en bits; y d es la distancia euclidiana al receptor (Ecuación 3). Este modelo aplica bajo la condición de espacio libre cuando $d < d_0$, con $d_0 = \sqrt{(Eelec / \varepsilon mp)}$, donde εmp es el coef00iciente de amplificación en modo multipath (no empleado en este estudio dado el rango de distancias del escenario).

El costo energético de recibir k bits es:

Ecuación 9
Energía de recepción

$$E_{\mathrm{RX}}(k) = E_{\mathrm{elec}} \cdot k \qquad (9)$$

Independiente de la distancia, ya que el consumo del receptor no depende de la potencia de la señal entrante sino únicamente del volumen de datos procesados.

La energía residual del nodo n al cierre de la ronda r se actualiza como:

Ecuación 10

Actualización de la energía residual

$$E_{\text{res}}(n,r) = E_{\text{res}}(n,r-1) - \Delta E_{\text{TX}}(n,r) - \Delta E_{\text{RX}}(n,r) \quad (10)$$

donde ΔE_TX(n,r) y ΔE_RX(n,r) son la energía total consumida en transmisión y recepción durante la ronda r, respectivamente. Para el nodo sensor miembro de un clúster (NCH), ΔE_TX corresponde a la transmisión de un paquete DATA a su CH. Para el CH, ΔE_TX corresponde a la transmisión de un paquete AGG al sink y ΔE_RX a la recepción de los paquetes DATA de todos sus miembros.

Un nodo n se declara muerto en la ronda r cuando su energía residual alcanza o cae por debajo del umbral de muerte:

Ecuación 11

Condición de muerte del nodo

$$\text{Muerte}(n) \Leftrightarrow E_{\text{res}}(n,r) \leq E_{\text{th}} \quad (11)$$

donde Eth = 0.05 J = 0.05 × E_0 es el umbral mínimo de energía operativa configurado como nodeShutdownCapacity en omnetpp.ini. A partir de ese momento, el nodo se apaga y no participa en ninguna ronda subsiguiente.

3.3.4.6. Métricas de Red

La Packet Delivery Ratio (PDR) cuantifica la fracción de paquetes generados por los sensores que efectivamente son recibidos por el sink a lo largo de toda la simulación:

Ecuación 12

Tasa de entrega de paquetes (PDR)

$$\mathrm{PDR} = \frac{N_{\mathrm{recv}}}{N_{\mathrm{sent}}} \times 100\,\% \qquad (12)$$

donde N_recv es el número total de paquetes recibidos en el sink y N_sent es el número total de paquetes generados por todos los nodos sensores durante la corrida completa.

El throughput efectivo de la red es la cantidad de datos útiles entregados al sink por unidad de tiempo:

Ecuación 13

Throughput efectivo de la red

$$S = \frac{N_{\mathrm{recv}} \cdot L_{\mathrm{pkt}}}{T_{\mathrm{sim}}} \quad [\mathrm{bps}] \qquad (13)$$

donde L_pkt es el tamaño del paquete de datos (bits) y T_sim es el tiempo total transcurrido de la corrida hasta el evento LND.

La latencia de entrega extremo a extremo de un paquete que recorre la ruta sensor → CH → sink incluye el retardo de propagación acústica y el retardo de serialización:

Ecuación 14

Latencia de entrega extremo a extremo

$$L = \frac{d_{\mathrm{total}}}{v_{\mathrm{s}}} + \frac{L_{\mathrm{pkt}}}{B} \quad [\mathrm{s}] \qquad (14)$$

donde d_total es la distancia total recorrida (sensor→CH + CH→sink), v_s = 1 500 m/s es la velocidad del sonido en agua de mar, L_pkt es el tamaño del paquete en bits y B = 10 kbps es la tasa de transmisión del enlace acústico.

3.3.4.7. Estadística Descriptiva del Procesamiento de Resultados

Para cada métrica M y cada escenario S se calculan los siguientes estadísticos sobre las 10 repeticiones independientes (semillas a–j), siguiendo los procedimientos de Montgomery y Runger (2018):

Estimador de la media muestral:

Ecuación 15

Media muestral

$$\hat{\mu} = \frac{1}{n}\sum_{i=1}^{n} x_i \qquad (15)$$

(15)

donde x_i es el valor de la métrica en la corrida i y n = 10 es el número de repeticiones.

Desviación estándar muestral (con corrección de Bessel):

Ecuación 16

Desviación estándar muestral (corrección de Bessel)

$$s = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \hat{\mu})^2} \qquad (16)$$

donde n−1 = 9 son los grados de libertad del estimador.

Intervalo de confianza del 95 % sobre la media (distribución t de Student con n−1 = 9 grados de libertad):

Ecuación 17

Intervalo de confianza del 95 %

$$\mathrm{IC}_{95} = \hat{\mu} \pm t_{0{,}975;\,9} \cdot \frac{s}{\sqrt{n}} \qquad (17)$$

donde $t_{0.025,9} \approx 2.262$ es el cuantil 0.975 de la distribución t de Student con 9 grados de libertad. Las barras de error en las gráficas del Capítulo IV representan este intervalo de confianza.

3.3.5. Modelo del Canal Acústico

El canal acústico subacuático se modela mediante el módulo UanModel con propagación en espacio libre (FreeSpacePathLoss). Los parámetros del enlace acústico se recogen en la Tabla 6.

Tabla 6

Parámetros del modelo del canal acústico subacuático

Parámetro	Valor	Parámetro en omnetpp.ini
Velocidad de propagación del sonido	1 500 m/s	Constante del módulo UanModel
Tasa de transmisión (B)	10 kbps	**.bitrate = 10kbps
Rango máximo de transmisión	200 m	**.maxCommunicationRange = 200m
Potencia TX activa	50 mW	*.transmitterTransmittingPowerConsumption = 50mW
Potencia RX activa	20 mW	*.receiverReceivingPowerConsumption = 20mW
Potencia en estado idle	5 mW	*.receiverIdlePowerConsumption = 5mW
Modelo de pérdida de trayectoria	FreeSpacePathLoss	**.pathLoss.typename = "FreeSpacePathLoss"

Nota. RX = recepción; TX = transmisión; B = tasa de transmisión. Los parámetros corresponden a la configuración del módulo **UanModel** utilizada en las simulaciones y se establecieron mediante el archivo *omnetpp.ini*. El modelo de pérdida de trayectoria

(*FreeSpacePathLoss*) asume propagación en espacio libre y se empleó de forma idéntica para los protocolos LEACH e I-LEACH. Elaboración propia.

3.3.6. Modelo de Tráfico

Cada nodo sensor genera un paquete de datos de 64 bytes (512 bits) durante la fase steady-state de cada ronda en la que permanece vivo. El tráfico es periódico: un paquete por nodo por ronda, lo que resulta en un número estimado de paquetes generados de N × R_efectivo por corrida, donde N es el número de nodos del escenario y R_efectivo es el número de rondas completadas antes del evento LND. Cada CH recibe los paquetes DATA de todos sus miembros, los agrega en un único mensaje AGG_DATA y lo transmite al sink.

3.4. Fase III. Implementación de la Simulación en OMNeT++

La Fase III describe la construcción concreta del modelo de simulación en OMNeT++: la organización del proyecto, los archivos que lo componen, la lógica de cada protocolo y el flujo completo de ejecución de una ronda. Todos los fragmentos de código mencionados en el presente apartado corresponden al código fuente real; el código completo se incluye en los Anexos.

3.4.1. Organización del Proyecto

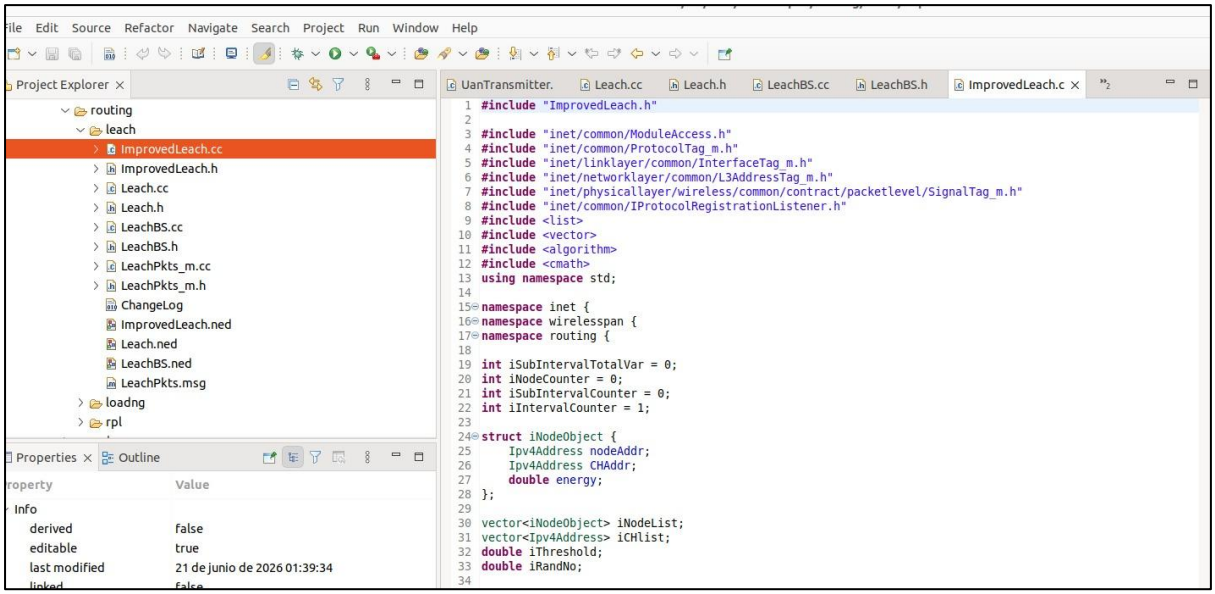
El proyecto se estructura bajo el workspace de OMNeT++ con la siguiente organización de directorios:

- inetmanet-4.x-master/ — framework base con los modelos de red, canal acústico (UanModel) y módulos de energía.
- src/inet/wirelesspan/routing/leach/ — código fuente C++ (Leach.cc / Leach.h) de los protocolos LEACH e Improved LEACH.

- examples/wirelesspan/leach/ — archivos NED de topología, archivo omnetpp.ini y carpeta results/.
- results/ — archivos .sca y .vec generados automáticamente por cada corrida.
- analysis/ — scripts Python de postprocesamiento y generación de gráficas.

Figura 3

Estructura de carpetas del proyecto en el IDE de OMNeT++ (Eclipse), mostrando los directorios principales y los archivos Leach.cc, Leach.h y omnetpp.ini



Nota. La imagen corresponde al entorno de desarrollo integrado (IDE) de OMNeT++ basado en Eclipse. En el panel izquierdo se muestra la estructura de directorios del proyecto, incluyendo los archivos fuente del protocolo LEACH (*Leach.cc* y *Leach.h*), mientras que en el panel derecho se observa el editor de código del archivo *Leach.cc*. Elaboración propia.

3.4.2. Estructura de Archivos del Proyecto

3.4.2.1. Archivos NED

Los archivos .ned definen la topología de la red: los tipos de nodo (SensorNode para los sensores, BaseStation para el sink), el medio acústico UanSoundMedium que conecta todos los elementos, y la red (Network.ned) que instancia los N nodos sensores y el sink. Los parámetros configurables (como el número de nodos) se exponen como parámetros del módulo NED para poder ser barridos desde omnetpp.ini sin recompilar.

3.4.2.2. Archivos C++

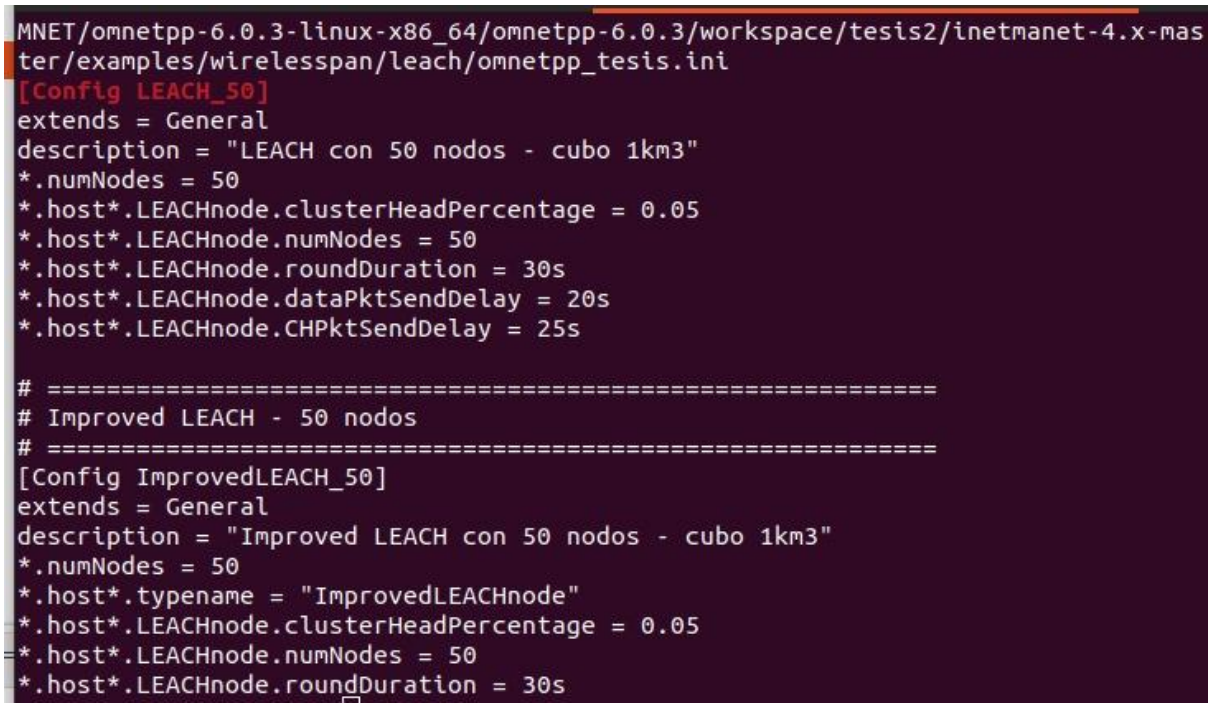
Los archivos Leach.cc y Leach.h implementan la máquina de estados del protocolo. La función handleMessage() procesa todos los mensajes entrantes (ADV, JOIN, TDMA, DATA, AGG_DATA) y despacha las acciones correspondientes. Las decisiones de temporización (inicio de ronda, envío de ADV, inicio de steady-state) se implementan mediante self-messages programados en la cola de eventos de OMNeT++. La distinción entre LEACH e Improved LEACH se controla mediante un parámetro booleano useImprovedLEACH en omnetpp.ini, que activa la Ecuación (7) en lugar de la Ecuación (6) en la función de selección de CH.

3.4.2.3. Archivo omnetpp.ini

El archivo omnetpp.ini centraliza todos los parámetros configurables de la simulación. Cada sección [Config NombreEscenario] define un escenario independiente que puede ejecutarse sin recompilar el código. Los atributos repeat = 10 y seed-set = ${repetition} generan automáticamente las 10 repeticiones (semillas a–j) por escenario.

Figura 4

Sección [Config LEACH_50] y [Config ImprovedLEACH_50] con los parámetros principales de la simulación



Nota. La captura muestra las secciones [Config LEACH_50] y [Config ImprovedLEACH_50] del archivo omnetpp.ini. En ambas configuraciones se mantienen constantes el número de nodos, la probabilidad objetivo de selección de CH, la duración de ronda y el tamaño de los paquetes; únicamente cambia el tipo de módulo utilizado para ejecutar LEACH o I-LEACH. Elaboración propia.

3.4.3. Implementación del Protocolo LEACH

El protocolo LEACH se implementa como el protocolo base de referencia en este estudio. Su lógica opera en ciclos de rondas, donde cada ronda comprende tres fases secuenciales: anuncio y selección de Cluster Head (CH), formación de clústeres y fase steady-state de transmisión de datos. La implementación sigue el modelo matemático descrito en la Sección 3.3.4, utilizando la función de umbral de la Ecuación (6) para la selección estocástica de CH. Todos los parámetros del protocolo —probabilidad objetivo de CH (pCH = 0.05), duración de ronda (T_ronda = 30 s) y modelo energético— se configuran conforme a la Tabla 8 del Capítulo III y permanecen idénticos en todas las corridas de LEACH e Improved LEACH,

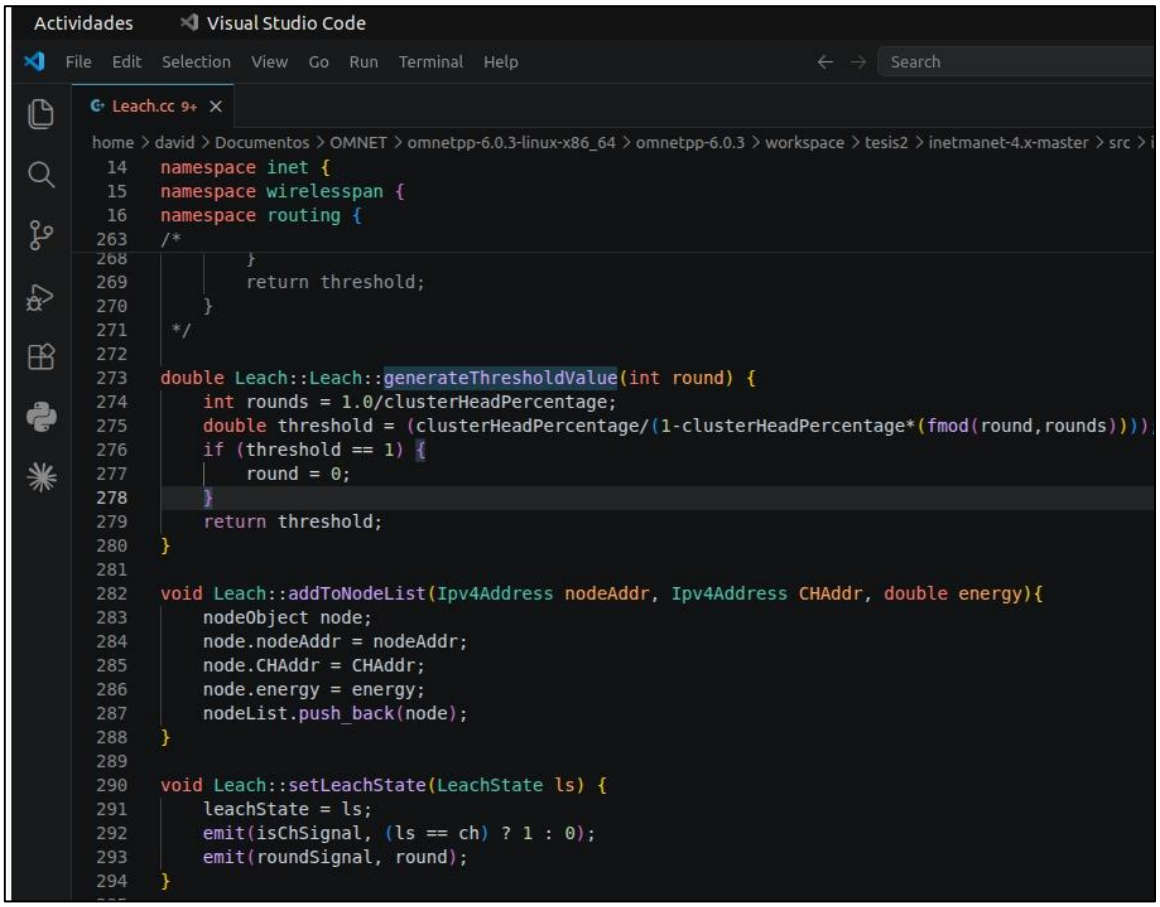
garantizando que las diferencias observadas en el Capítulo IV sean atribuibles exclusivamente al algoritmo de selección de CH y no a variaciones del entorno de simulación.

3.4.3.1. Selección del Cluster Head

Al inicio de cada ronda r, cada nodo sensor activo (con Eres(n,r) > Eth) que no ha sido CH en las últimas 20 rondas (G(r) es el conjunto de exclusión) calcula el umbral T(n,r) según la Ecuación (6). A continuación genera un número pseudoaleatorio u ~ U(0,1); si u < T(n,r), el nodo se auto-proclama CH para la ronda actual. En promedio, pCH × N nodos son elegidos CH por ronda, aunque el número exacto varía estocásticamente según la semilla.

Figura 5

Implementación de la función generateThresholdValue() del protocolo LEACH



Nota. La captura muestra la implementación en C++ de la función generateThresholdValue(), encargada de calcular el umbral de selección de cabeza de clúster (CH) en cada ronda. El cálculo considera la probabilidad objetivo de CH y el número de ronda, de acuerdo con la formulación del protocolo LEACH presentada en la Ecuación (6). Elaboración propia.

3.4.3.2. Formación de Clústeres

El CH seleccionado difunde un mensaje de anuncio ADV que contiene su identificador. Cada nodo NCH recibe los mensajes ADV disponibles durante la fase de anuncio y, al término de esta fase, selecciona al CH cuyo mensaje ADV fue recibido con mayor potencia (proxy de menor distancia, dado que todos transmiten a la misma potencia). El NCH envía un mensaje JOIN al CH elegido. El CH acumula los mensajes JOIN de sus miembros, construye una trama de planificación TDMA y la difunde mediante un mensaje REPLY que asigna a cada miembro su slot de transmisión en la fase steady-state.

3.4.3.3. Comunicación Sensor → CH → Sink

Durante la fase steady-state, cada sensor NCH transmite su paquete DATA al CH en el slot TDMA asignado. El CH recibe los paquetes DATA de todos sus miembros y los agrega en un único mensaje AGG_DATA que transmite directamente al sink. La energía consumida en cada transmisión y recepción se descuenta del nodo correspondiente según las Ecuaciones (8) y (9). Al cierre de la fase steady-state se actualiza la energía residual de cada nodo según la Ecuación (10) y se verifica la condición de la Ecuación (11).

3.4.4. Implementación del Protocolo Improved LEACH

El protocolo Improved LEACH basado en energía residual se implementa como la variante experimental a comparar frente a LEACH estándar. Comparte íntegramente la arquitectura de rondas, la estructura de mensajes y el modelo energético descritos en la Sección 3.4.3; la única modificación introducida se encuentra en la función de selección del Cluster Head, que incorpora la energía residual de cada nodo como criterio de ponderación del umbral de elección. Esta modificación, formalizada en la Ecuación (7) del Capítulo III, tiene como objetivo redistribuir la carga energética entre los nodos de la red, favoreciendo la elección de

nodos con mayor batería disponible y reduciendo la probabilidad de que nodos próximos al agotamiento asuman el rol de CH. El resto del flujo operativo —anuncio (ADV), asociación (JOIN), planificación TDMA, fase steady-state y actualización energética— es idéntico al de LEACH, lo que garantiza que cualquier diferencia observada en el Capítulo IV sea atribuible exclusivamente a esta modificación algorítmica.

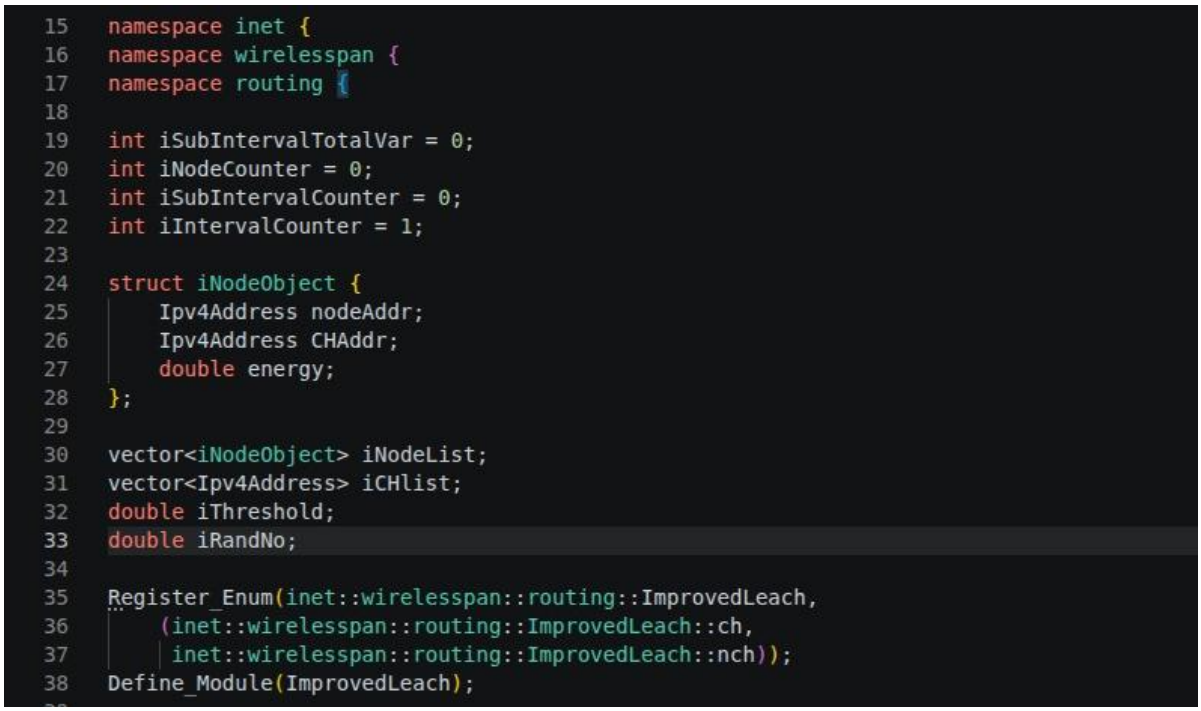
3.4.4.1. Modificación Respecto a LEACH: Umbral Ponderado por Energía Residual

La única diferencia algorítmica entre LEACH e Improved LEACH en este estudio es la función de selección del CH. Improved LEACH sustituye la Ecuación (6) por la Ecuación (7): el umbral de LEACH se multiplica por el factor de energía residual [Eres(n,r) / E_0]. Toda la estructura de mensajes (ADV, JOIN, REPLY, DATA, AGG_DATA), la formación de clústeres y la fase steady-state son idénticas en ambos protocolos.

La consecuencia práctica de esta modificación es que un nodo con Eres = 0.9 J (90 % de su energía inicial) tiene un umbral T' = 0.9 × T(n,r), mientras que un nodo con Eres = 0.3 J (30 % de su energía inicial) tiene un umbral T' = 0.3 × T(n,r). En consecuencia, los nodos con mayor energía residual son elegidos CH con mayor frecuencia, lo que redistribuye la carga energética hacia los nodos menos agotados y, en principio, extiende la vida útil global de la red.

Figura 6

Condición if(useImprovedLEACH) y la aplicación del factor [Eres/E₀] en la función generateThresholdValue() de Leach.cc. Código completo en Anexo A



```
15  namespace inet {
16  namespace wirelesspan {
17  namespace routing {
18
19  int iSubIntervalTotalVar = 0;
20  int iNodeCounter = 0;
21  int iSubIntervalCounter = 0;
22  int iIntervalCounter = 1;
23
24  struct iNodeObject {
25      Ipv4Address nodeAddr;
26      Ipv4Address CHAddr;
27      double energy;
28  };
29
30  vector<iNodeObject> iNodeList;
31  vector<Ipv4Address> iCHlist;
32  double iThreshold;
33  double iRandNo;
34
35  Register_Enum(inet::wirelesspan::routing::ImprovedLeach,
36      (inet::wirelesspan::routing::ImprovedLeach::ch,
37       inet::wirelesspan::routing::ImprovedLeach::nch));
38  Define_Module(ImprovedLeach);
```

Nota. La captura muestra el archivo fuente ImprovedLeach.cc, donde se implementan las modificaciones introducidas al protocolo LEACH para incorporar el criterio de selección de cabezas de clúster basado en la energía residual. El código completo se presenta en el Anexo A. Elaboración propia.

3.4.4.2. Comunicación Sensor → CH → Sink

La cadena de comunicación sensor → CH → sink es idéntica a la descrita para LEACH en la Sección 3.4.3.3. El modelo energético y las ecuaciones de consumo son los mismos; la diferencia observable entre protocolos emerge de la distribución de los CH elegidos a lo largo de las rondas, no de la mecánica de transmisión.

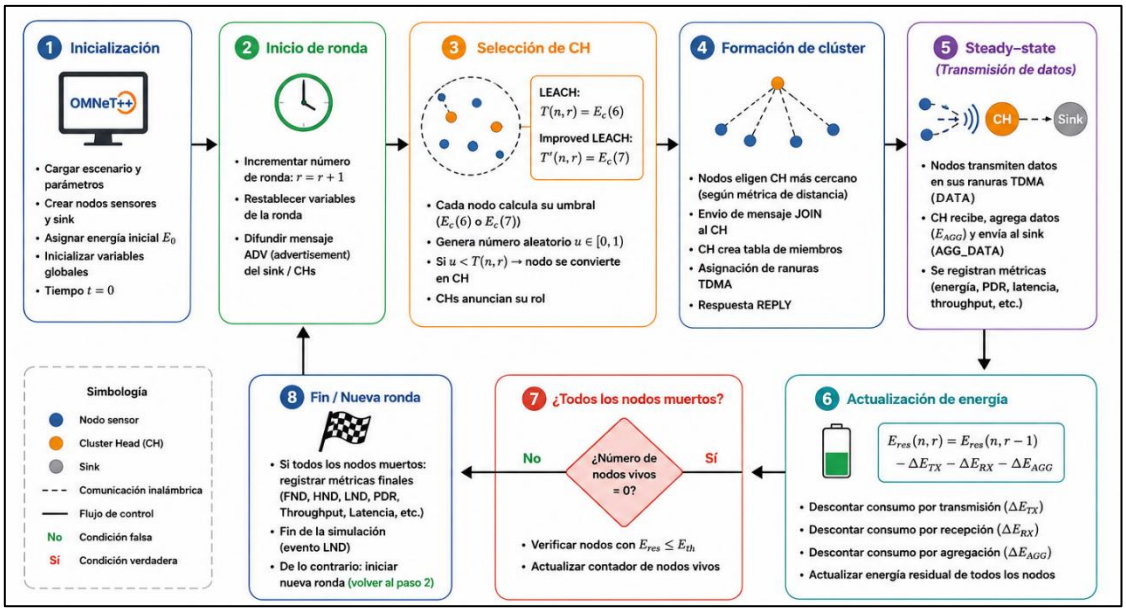
3.4.5. Flujo General de una Simulación

El flujo de ejecución de una simulación completa —común a ambos protocolos, salvo en el paso de selección de CH— se describe a continuación de manera secuencial. Este flujo se ejecuta dentro del motor de eventos de OMNeT++ para cada corrida (combinación escenario × protocolo × semilla).

A continuación, se describe cada etapa del flujo:

Figura 7

Diagrama de flujo de la simulación LEACH e Improved LEACH



Nota. El flujo inicia con la inicialización del escenario y la configuración de los parámetros de simulación. En cada ronda se realiza la selección de cabezas de clúster (CH), la formación de clústeres, la transmisión de datos mediante TDMA y la actualización de la energía residual de los nodos. El proceso se repite hasta que ocurre el evento **LND** (*Last Node Dies*), correspondiente a la muerte del último nodo de la red, momento en el que finaliza la simulación y se registran las métricas de desempeño. La diferencia entre LEACH e I-LEACH se presenta únicamente en el cálculo del umbral de selección de CH. Elaboración propia.

1. Inicialización del escenario.

Al inicio de la simulación (t = 0), OMNeT++ instancia los N nodos sensores y el sink. Las posiciones de los nodos se generan aleatoriamente dentro del volumen $V = L_x \times L_\gamma \times Lz$ (Ecuación 1) según la distribución de la Ecuación (2), controlada por la semilla de la corrida. La energía de todos los nodos se inicializa en $E_0 = 1.0$ J. OMNeT++ programa el auto-mensaje de inicio de la primera ronda en cada nodo.

2. Inicio de ronda r.

Al recibir el auto-mensaje de inicio de ronda, cada nodo activo verifica si su energía residual supera el umbral de muerte (Ecuación 11). Los nodos con Eres ≤ Eth no ejecutan

ninguna acción y son marcados como muertos. Los nodos activos verifican si pertenecen al conjunto de exclusión G(r).

3. Selección del Cluster Head.

Cada nodo activo fuera del conjunto de exclusión calcula su umbral: T(n,r) (Ecuación 6) para LEACH, o T'(n,r) (Ecuación 7) para Improved LEACH. Genera un número aleatorio u ~ U(0,1). Si u < T(n,r) [o T'(n,r)], el nodo se auto-proclama CH, difunde un mensaje ADV y se añade al conjunto G(r+1). El conjunto G se actualiza con un periodo de ventana de ⌊1/pCH⌋ = 20 rondas, de modo que un nodo que ha sido CH debe esperar 20 rondas antes de poder volver a serlo.

4. Formación de clústeres.

Durante la fase de anuncio, los NCH escuchan los mensajes ADV difundidos por los CH. Al término de la fase, cada NCH envía un mensaje JOIN al CH del que recibió el ADV con mayor potencia. El CH acumula los mensajes JOIN y construye la trama TDMA. Envía el mensaje REPLY con la planificación temporal a todos sus miembros. El CH consume la energía de difusión del ADV y del REPLY (Ecuación 8, con d = rango de broadcast); cada NCH consume la energía de transmisión del JOIN.

5. Comunicación Sensor → CH (fase steady-state).

En el slot TDMA asignado, cada sensor NCH transmite su paquete DATA (64 bytes) al CH. El consumo energético del NCH es E_TX(k, d_CH) (Ecuación 8), donde d_CH es la distancia euclidiana del sensor al CH (Ecuación 3). El CH consume E_RX(k) (Ecuación 9) por cada paquete recibido.

6. Comunicación CH → Sink (agregación y reenvío).

El CH agrega los paquetes DATA recibidos en un único mensaje AGG_DATA y lo transmite directamente al sink. El consumo energético del CH es E_TX(k_agg, d_sink)

(Ecuación 8), donde k_agg es el tamaño del paquete agregado y d_sink es la distancia euclidiana entre el CH y el sink (Ecuación 3). El sink recibe el AGG_DATA, incrementa el contador N_recv y registra la latencia de entrega (Ecuación 14).

7. Actualización de energía.

Al cierre de la fase steady-state, cada nodo actualiza su energía residual según la Ecuación (10), restando la energía total consumida en transmisión y recepción durante la ronda. La verificación de la Ecuación (11) determina si el nodo muere en esta ronda. OMNeT++ registra los eventos FND, HND y LND cuando el número de nodos muertos alcanza 1, ⌊N/2⌋ y N, respectivamente.

8. Inicio de una nueva ronda o finalización de la simulación.

Si queda al menos un nodo vivo, OMNeT++ programa el auto-mensaje de inicio de la siguiente ronda ($r \leftarrow r + 1$) para todos los nodos activos, y el ciclo se repite desde el paso 2. Si todos los nodos han muerto (evento LND), la cola de eventos queda vacía y la simulación concluye. No existe un límite de rondas predefinido; la condición de finalización real es el agotamiento energético de todos los nodos. El límite sim-time-limit configurado en omnetpp.ini actúa únicamente como salvaguarda ante errores de simulación y no es alcanzado bajo condiciones normales de operación.

3.4.6. Instrumentación y Generación de Resultados

OMNeT++ registra los eventos de simulación mediante señales (signals) emitidas en los módulos C++. Los archivos de resultados se generan automáticamente en la carpeta results/ al finalizar cada corrida:

.sca (Scalar files):

Contienen un valor por métrica por corrida completa: energía residual total al evento LND, ronda de FND, ronda de HND, ronda de LND, PDR global y throughput total. Son el insumo principal para las comparaciones agregadas entre protocolos.

.vec (Vector files):

Contienen series temporales por ronda: energía residual por nodo, número de nodos vivos, paquetes enviados y recibidos, latencia de entrega. Estas series permiten analizar la evolución temporal de cada métrica a lo largo de la vida de la red. La grabación de vectores se habilita en omnetpp.ini mediante result-recording-modes = default,+vector.

3.5. Fase IV. Diseño Experimental

La Fase IV define los escenarios de experimentación, los parámetros de cada corrida, la organización de las campañas experimentales y el control de la aleatoriedad. El diseño experimental busca garantizar que la comparación entre LEACH e Improved LEACH sea estadísticamente robusta, reproducible y libre de confusión entre el efecto del protocolo y el efecto de la densidad de la red o la variabilidad estocástica del escenario.

3.5.1. Definición de los Escenarios Experimentales

Se definen cinco escenarios que varían únicamente en el número de nodos sensores (N). El resto de parámetros del escenario (dimensiones, posición del sink, modelo de canal, modelo energético, tráfico) permanece idéntico en todos los escenarios y para ambos protocolos, garantizando que las diferencias observadas en las variables de salida sean atribuibles al protocolo o a la densidad de la red, no a diferencias del entorno.

Tabla 7

Escenarios experimentales definidos

Escenario	Nodos (N)	CH esperados (pCH = 5%)	Nombre en omnetpp.ini (LEACH / Improved)
S1	50	≈ 3	[Config LEACH_50] / [Config ImprovedLEACH_50]
S2	100	≈ 5	[Config LEACH_100] / [Config ImprovedLEACH_100]
S3	250	≈ 13	[Config LEACH_250] / [Config ImprovedLEACH_250]
S4	500	≈ 25	[Config LEACH_500] / [Config ImprovedLEACH_500]
S5	1 000	≈ 50	[Config LEACH_1000] / [Config ImprovedLEACH_1000]

Nota. N = número de nodos sensores; CH = cabeza de clúster (*Cluster Head*); $p_{CH} = 5\,\%$corresponde a la probabilidad objetivo de selección de cabezas de clúster. El número esperado de CH se estimó como $N \times p_{CH}$y se presenta como un valor aproximado, debido a que la selección de CH es probabilística y puede variar entre corridas. Los nombres de configuración corresponden a las secciones definidas en el archivo *omnetpp.ini*. Elaboración propia.

3.5.2. Campañas Experimentales

Cada escenario se ejecuta con ambos protocolos (LEACH e Improved LEACH) y con 10 semillas distintas (a–j, correspondientes a seed-set 0–9 en omnetpp.ini). El total de simulaciones realizadas es:

$$5\ \text{escenarios} \times 2\ \text{protocolos} \times 10\ \text{semillas} = 100\ \text{simulaciones}$$

Este diseño factorial completo garantiza que las comparaciones estadísticas entre protocolos estén sustentadas en un número suficiente de muestras independientes para calcular intervalos de confianza al 95 % (Ecuación 17).

Las 100 corridas se ejecutaron mediante la función de campañas (campaign runs) de OMNeT++, que lanza automáticamente todas las combinaciones de parámetros definidas en omnetpp.ini sin intervención manual entre corridas. El tiempo total de cómputo de la campaña completa se registró para cada escenario como referencia de escalabilidad del modelo.

3.5.3. Parámetros de Simulación

La Tabla 8 consolida todos los parámetros que permanecen fijos en las 100 corridas de la campaña experimental.

Tabla 8

Parámetros de simulación comunes a todos los escenarios

Parámetro	Valor	Parámetro en omnetpp.ini
Condición de finalización	Evento LND	(Cola de eventos vacía; sim-time-limit como protección)
Duración de ronda (T_ronda)	30 s	*.LEACHnode.roundDuration = 30s
Probabilidad objetivo de CH (pCH)	0.05	*.LEACHnode.clusterHeadPercentage = 0.05
Energía inicial (E_0)	1.0 J	*.energyStorage.initialCapacity = 1.0J
Umbral de muerte de nodo (Eth)	0.05 J	*.energyManagement.nodeShutdownCapacity = 0.05J
Tasa de transmisión (B)	10 kbps	**.bitrate = 10kbps
Potencia TX activa	50 mW	*.transmitterTransmittingPowerConsumption = 50mW
Potencia RX activa	20 mW	*.receiverReceivingPowerConsumption = 20mW
Potencia idle TX/RX	5 mW	*.transmitter/receiverIdlePowerConsumption = 5mW
Energía electrónica (Eelec)	50 nJ/bit	(Constante en Leach.cc — Heinzelman et al., 2000)
Coeficiente amplificación (εfs)	10 pJ/(bit·m²)	(Constante en Leach.cc — Heinzelman et al., 2000)
Tamaño de paquete (L_pkt)	64 bytes (512 bits)	*.dataPktSize = 512b
Modelo de pérdida de trayectoria	FreeSpacePathLoss	**.pathLoss.typename = "FreeSpacePathLoss"
Dimensiones del volumen	1000×1000×1000 m	*.constraintAreaMaxX/Y/Z = 1000m/1000m/1000m
Número de repeticiones por escenario	10 (a–j)	repeat = 10 / seed-set = ${repetition}

Nota. LND = *Last Node Dies* (muerte del último nodo); CH = *Cluster Head* (cabeza de clúster); p_{CH}= probabilidad objetivo de selección de CH; E_0= energía inicial por nodo; E_{th}= umbral de muerte del nodo; T_{ronda}= duración de cada ronda; B= tasa de transmisión; TX = transmisión; RX = recepción; E_{elec}= energía consumida por la electrónica del transceptor; ε_{fs}= coeficiente de amplificación para propagación en espacio libre; L_{pkt}= tamaño del paquete. Los parámetros se mantuvieron constantes para LEACH e I-LEACH en todos los escenarios. E_{elec}y ε_{fs}se encuentran definidos como constantes en el código fuente, mientras que los demás parámetros se configuran en el archivo *omnetpp.ini*. Cada escenario se ejecutó 10 veces con semillas de 0 a 9. Elaboración propia.

3.5.4. Control de Aleatoriedad

La reproducibilidad de los resultados se garantiza mediante el uso de semillas fijas en OMNeT++. El atributo seed-set en omnetpp.ini inicializa de forma determinista el generador de números pseudoaleatorios de Mersenne Twister que controla: (1) las posiciones iniciales de los nodos (distribución uniforme 3D, Ecuación 2); (2) el número pseudoaleatorio u generado por cada nodo en la fase de selección de CH; y (3) cualquier otro proceso estocástico del simulador. Como consecuencia, dos ejecuciones con el mismo seed-set producen resultados exactamente idénticos, lo que permite verificar la repetibilidad de la infraestructura de simulación.

Las 10 semillas (a–j, seed-set 0–9) producen escenarios espacialmente distintos entre sí, capturando la variabilidad inherente al despliegue aleatorio de los nodos. El análisis estadístico de las 10 repeticiones mediante las Ecuaciones (15), (16) y (17) permite cuantificar esa variabilidad y afirmar con 95 % de confianza la superioridad o equivalencia de un protocolo frente al otro.

3.5.5. Métricas Registradas Durante la Simulación

Las métricas definidas en la Sección 3.2.2 y formalizadas matemáticamente en las Ecuaciones (12), (13) y (14) se registran automáticamente por OMNeT++ durante la ejecución de cada corrida. Los escalares (FND, HND, LND, PDR global, throughput total) se almacenan en los archivos .sca; las series temporales por ronda (energía residual por nodo, nodos vivos, paquetes enviados/recibidos, latencia) se almacenan en los archivos .vec. La configuración de grabación en omnetpp.ini especifica explícitamente qué señales se registran en formato escalar, vectorial o ambos.

3.6. Fase V. Procesamiento y Validación de Resultados

La Fase V describe el proceso mediante el cual los archivos de resultados generados por OMNeT++ se transforman en las gráficas y tablas comparativas del Capítulo IV. Todo el

procesamiento se realiza fuera del simulador, en Python, aprovechando las capacidades de manipulación masiva de datos de pandas, el cálculo numérico de NumPy y la visualización de matplotlib.

3.6.1. Extracción de Datos desde OMNeT++

Al finalizar las 100 corridas de la campaña experimental, los archivos binarios .sca y .vec se convierten a formato CSV mediante la herramienta opp_scavetool, incluida en la distribución de OMNeT++ 6.0.3. La herramienta permite filtrar por tipo de estadístico (escalar o vector), por nombre de señal y por módulo, y exporta un archivo CSV con columnas de metadatos (nombre del escenario, run ID, número de semilla, módulo fuente, nombre de la señal) y columnas de valor. Este formato es directamente legible por pandas en Python sin transformaciones adicionales.

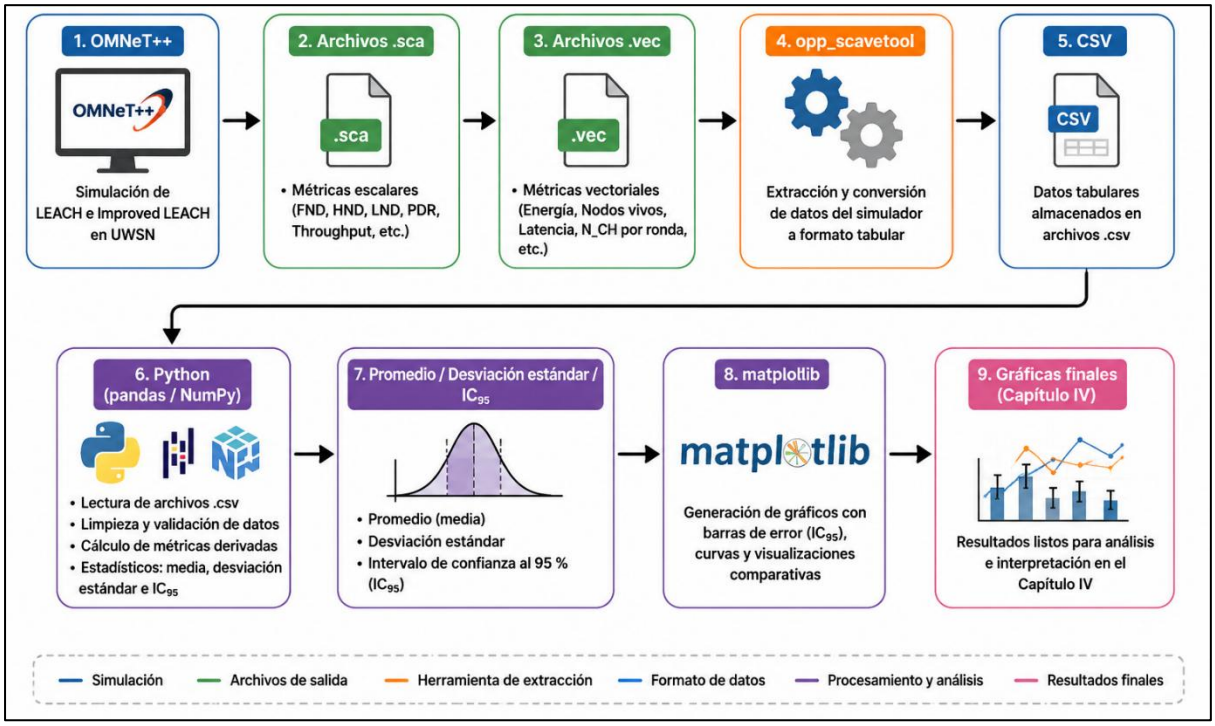
3.6.2. Flujo de Procesamiento de Resultados

El flujo completo de transformación de los resultados de simulación en figuras comparativas sigue la cadena descrita a continuación:

Figura 8

Diagrama del flujo de procesamiento de resultados: OMNeT++ → Archivos .sca →

Archivos .vec → opp_scavetool → CSV → Python (pandas/NumPy) → Promedio / Desviación estándar / IC_{95} → matplotlib → Gráficas finales



Nota. Los archivos de resultados generados por OMNeT++ (*.sca* y *.vec*) se convierten a formato CSV mediante la herramienta opp_scavetool. Posteriormente, scripts desarrollados en Python, utilizando las bibliotecas *pandas* y *NumPy*, realizan la depuración, agregación por escenario, protocolo y semilla, así como el cálculo de medias, desviaciones estándar e intervalos de confianza del 95 % (IC_{95}). Finalmente, las figuras comparativas del Capítulo IV se generan mediante la biblioteca *Matplotlib*. Elaboración propia.

Las etapas de la cadena son las siguientes:

Etapa 1 — Ejecución y registro:

OMNeT++ ejecuta las 100 corridas de la campaña experimental y registra automáticamente los resultados en los archivos .sca (escalares por corrida) y .vec (vectores por ronda por corrida) en la carpeta results/.

Etapa 2 — Exportación a CSV:

opp_scavetool convierte los archivos binarios a un único CSV con todas las métricas de todas las corridas. Cada fila del CSV corresponde a un (escenario, protocolo, semilla, señal, ronda) específico, lo que facilita el filtrado posterior.

Etapa 3 — Carga y filtrado en Python:

Los scripts Python leen el CSV con pd.read_csv(), filtran por escenario (columna config) y por protocolo (columna run), y organizan los datos en DataFrames con índice [semilla × ronda]. Este paso produce una estructura de datos limpia y trazable para el análisis estadístico.

Etapa 4 — Procesamiento estadístico:

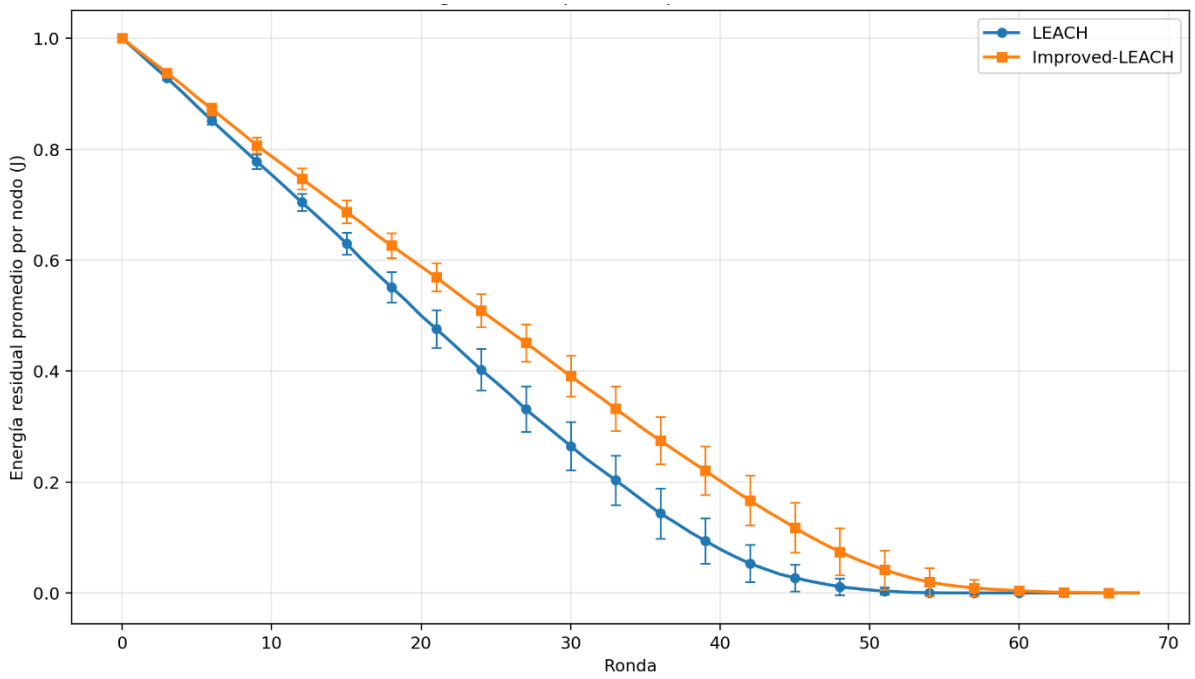
Para cada métrica M, escenario S y ronda r, se calculan la media muestral (Ecuación 15), la desviación estándar (Ecuación 16) y el intervalo de confianza al 95 % (Ecuación 17) sobre las 10 repeticiones. Los resultados se almacenan en DataFrames de salida con columnas [ronda, $\hat{\mu}$, s, IC_inferior, IC_superior] para cada protocolo.

Etapa 5 — Generación de gráficas:

matplotlib produce las figuras comparativas del Capítulo IV: curvas de energía residual vs. rondas, nodos vivos vs. rondas, throughput vs. rondas, latencia vs. rondas y barras comparativas de FND/HND/LND y PDR. Todas las figuras incluyen bandas de error que representan el intervalo de confianza al 95 %, y leyendas que distinguen LEACH (línea azul) de Improved LEACH (línea naranja).

Figura 9

Energía residual promedio por nodo en función de la ronda para el escenario S1



Nota. Los valores corresponden a la media de diez semillas emparejadas. Los círculos representan LEACH y los cuadrados Improved-LEACH. Las barras verticales indican el intervalo de confianza del 95 %. La energía residual promedio se calculó dividiendo la energía residual total para los 50 nodos iniciales y asignando energía cero a los nodos muertos.

3.6.3. Procesamiento Estadístico

El procesamiento estadístico aplica las Ecuaciones (15), (16) y (17) sobre los valores de cada métrica en las 10 semillas de cada escenario/protocolo. Las 10 semillas se tratan como muestras aleatorias independientes. La aplicación de la distribución t de Student con $n-1 = 9$ grados de libertad, en lugar de la distribución normal estándar, es metodológicamente correcta dado el tamaño de muestra reducido ($n = 10 < 30$) y la varianza poblacional desconocida (Montgomery y Runger, 2018). El valor crítico $t_{0.025,9} = 2.262$ garantiza cobertura del 95 % para el intervalo de confianza.

3.6.4. Verificación de Repetibilidad

Antes de incorporar los resultados al Capítulo IV, se verifica la repetibilidad del entorno de simulación ejecutando dos corridas con el mismo escenario y el mismo seed-set, y comparando los archivos .sca resultantes mediante el hash MD5 de sus contenidos. La igualdad de los hashes confirma que el simulador es determinista para una semilla dada y que los resultados no dependen de condiciones externas no controladas (carga del sistema operativo, orden de ejecución de hilos, etc.).

3.6.5. Validación del Entorno de Simulación

La validación del entorno aplica los cinco criterios definidos en la Tabla 5 (Sección 3.2.4) a los resultados de las corridas de verificación realizadas durante la Fase III. La Tabla 9 resume el estado de cumplimiento de cada criterio al momento de cierre de la implementación.

Tabla 9

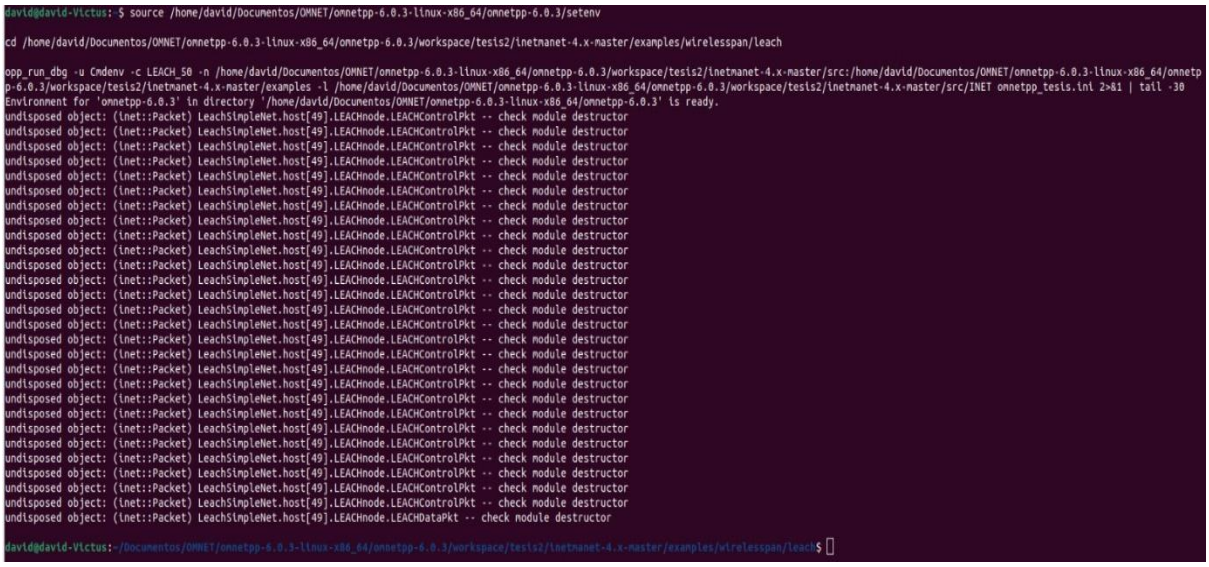
Estado de validación del entorno de simulación (verificado en Fase III)

Criterio	Estado	Evidencia
Nivel I – Conectividad base	Cumplido	100 % de paquetes entregados en el escenario mínimo sin clustering.
Nivel II – Clustering funcional	Cumplido	Ciclo ADV→JOIN→TDMA→DATA→AGG verificado para LEACH e Improved LEACH con N = 50.
Nivel III – Escalamiento	Cumplido	Corridas completadas sin errores para N = 100, 250, 500, 1 000.
Consistencia energética	Cumplido	Script Python confirma $E_i(r+1) \leq E_i(r)$ en todas las corridas de la campaña.
Repetibilidad	Cumplido	Hash MD5 idéntico en dos ejecuciones independientes con seed-set = 0.

Nota. ADV = mensaje de anuncio (*Advertisement*); JOIN = mensaje de asociación al clúster; TDMA = *Time Division Multiple Access*; DATA = paquete de datos; AGG = datos agregados; E_r= energía residual del nodo. La validación se realizó al finalizar la Fase III mediante la verificación de los criterios definidos en la Sección 3.2.4. Todos los escenarios experimentales cumplieron los criterios establecidos antes de iniciar el análisis comparativo del Capítulo IV. Elaboración propia.

Figura 10

Salida de la consola de OMNeT++ mostrando la finalización de una corrida de simulación por el evento LND.



Nota. La captura corresponde a la salida de la consola de OMNeT++ al finalizar una corrida de simulación. La ejecución concluye cuando ocurre el evento LND (Last Node Dies), es decir, cuando el último nodo de la red agota su energía. Durante la fase de finalización, el simulador libera automáticamente los módulos y recursos creados durante la ejecución, generando los mensajes de destrucción (check module destructor) observados en la consola. Elaboración propia.

4. CAPÍTULO IV: VALIDACIÓN DE RESULTADOS

4.1. Configuración Experimental Final

La presente sección establece el vínculo entre el diseño experimental definido en el Capítulo III y los resultados presentados en las secciones siguientes. Su propósito no es repetir la metodología ya descrita, sino confirmar que la campaña experimental se ejecutó conforme a lo planificado, documentar el estado real de las corridas y precisar las condiciones bajo las cuales se obtuvieron los datos analizados. Se reportan los escenarios ejecutados, el número de corridas válidas, el horizonte temporal efectivo de cada escenario y la trazabilidad de los parámetros de simulación, de modo que cualquier lector pueda reproducir los resultados a partir de la información aquí consignada.

4.1.1. Parámetros Controlados y Trazabilidad

Los parámetros de la simulación se fijaron conforme al diseño experimental del Capítulo III (Tabla 8, Cap. III). La tabla siguiente confirma que no existieron desviaciones respecto al diseño planificado y registra el valor efectivamente aplicado en cada corrida.

Tabla 10

Parámetros de la simulación

Parámetro	Valor planificado (Cap. III)	Parámetro en omnetpp.ini / fuente	Valor confirmado en corridas
Energía inicial por nodo (E_0)	1.0 J	*.energyStorage.initialCapacity = 1.0J	1.0 J
Umbral de muerte de nodo (Eth)	0.05 J	*.energyManagement.nodeShutdownCapacity = 0.05J	0.005 J
Probabilidad objetivo de CH (pCH)	0.05 (5 %)	*.LEACHnode.clusterHeadPercentage = 0.05	0.05
Duración de ronda (T_ronda)	30 s	*.LEACHnode.roundDuration = 30s	30 s
Condición de finalización	Evento LND	sim-time-limit como protección (no alcanzado)	LND
Modelo de pérdida de trayectoria	FreeSpacePathLoss	*.radioMedium.pathLoss.typename = "FreeSpacePathLoss"	FreeSpacePathLoss
Tasa de transmisión (B)	10 kbps	**.bitrate = 10kbps	10 kbps

Velocidad de propagación del sonido	1 500 m/s	Constante del módulo UanModel	1 500 m/s
Rango máximo de transmisión	200 m	**.maxCommunicationRange = 200m	200 m
Potencia TX activa	50 mW	*.transmitterTransmittingPowerConsumption = 50mW	50 mW
Potencia RX activa	20 mW	*.receiverReceivingPowerConsumption = 20mW	20 mW
Potencia idle TX/RX	5 mW	*.transmitterIdlePowerConsumption = 5mW	5 mW
Tamaño de paquete (L_pkt)	64 bytes (512 bits)	*.dataPktSize = 512b	512 bits
Dimensiones del volumen	1000×1000×1000 m	*.constraintAreaMaxX/Y/Z	1000×1000×1000 m
Posición del sink	(500, 500, 0) m — superficie, centro	Parámetro NED	(500,500,0) m
Número de repeticiones por escenario	10 (semillas a–j)	repeat = 10 / seed-set = ${repetition}	10
Semillas utilizadas	seed-set 0–9	seed-set 0–9 en omnetpp.ini	seed-set 0–9
Modelo energético (coef.)	Heinzelman et al. (2000): Eelec=50 nJ/b, εfs=10 pJ/(b·m²)	Constantes en Leach.cc	Conforme Cap.III

Nota. CH = *Cluster Head* (cabeza de clúster); p_{CH}= probabilidad objetivo de selección de CH; E_0= energía inicial por nodo; E_{th}= umbral de muerte del nodo; T_{ronda}= duración de cada ronda; TX = transmisión; RX = recepción; L_{pkt}= tamaño del paquete. La columna **Valor confirmado en corridas** corresponde a los parámetros efectivamente utilizados durante la ejecución de las simulaciones, verificados mediante el archivo *omnetpp.ini*, el código fuente (*Leach.cc*), los registros de ejecución y los archivos exportados. No se identificaron desviaciones respecto al diseño experimental definido en el Capítulo III. Elaboración propia.

4.1.2. Horizonte Temporal de Simulación por escenario

El horizonte temporal de cada corrida está determinado por el evento LND (muerte del último nodo). La tabla siguiente registra los valores reales obtenidos. Si alguna corrida alcanzó el límite sim-time-limit antes del evento LND, el LND debe registrarse como "no observado dentro del horizonte simulado" y la corrida tratarse según lo indicado en §4.3.

Tabla 11

Rondas hasta LND por escenario y protocolo (μ ± IC₉₅)

Esc.	N	LND LEACH (±IC₉₅)	Mín LEACH	Máx LEACH	LND iLEACH μ (±IC₉₅)	Mín iLEACH	Máx iLEACH	D̄ (rondas)	G_M (%)	IC₉₅(D̄)* conservador
S1	50	50.1(±3.40)	44	60	56.3(±4.23)	49	68	6.20	12.4%	6.20±5.42
S2	100	55.1(±5.02)	47	65	63.2(±2.07)	60	69	8.10	14.7%	8.10±5.44
S3	250	56(±3.02)	49	64	57(±2.38)	54	65	1.00	1.8%	1.00±3.85
S4	500	54.7(±2.43)	50	61	62.8(±2.26)	59	69	8.10	14.8%	8.10±3.32

S5	1000	54.9(±3.45)	50	62	62.6(±1.08)	61	65	7.70	14.0%	7.70±3.61
GLOBAL	—	54.16	44	65	60.38	49	69	+6.22	11.5%	—

Nota. LND = muerte del último nodo (*Last Node Dies*); IC_{95} = intervalo de confianza del 95 %; $\bar{D}$ = diferencia promedio entre las medias de LND de I-LEACH y LEACH; G_M = ganancia relativa calculada como porcentaje de mejora de I-LEACH respecto a LEACH. Los valores se presentan como media ± IC_{95} obtenidos a partir de 10 repeticiones por escenario.

4.1.3. Escenarios Ejecutados y Corridas Válidas

La campaña experimental planificada comprende 5 escenarios × 2 protocolos × 10 semillas = 100 simulaciones. La siguiente tabla registra el estado real de ejecución y validación de cada grupo de corridas, incluyendo los runs excluidos y el motivo de exclusión cuando corresponda.

Tabla 12

Resumen real de la campaña experimental

Escenario	N	Protocolo	Semillas planif.	Runs ejec.	Runs válidos	Runs excluidos
S1	50	LEACH	10	10	10	0
S1	50	Improved LEACH	10	10	10	0
S2	100	LEACH	10	10	10	0
S2	100	Improved LEACH	10	10	10	0
S3	250	LEACH	10	10	10	0
S3	250	Improved LEACH	10	10	10	0
S4	500	LEACH	10	10	10	0
S4	500	Improved LEACH	10	10	10	0
S5	1000	LEACH	10	10	10	0
S5	1000	Improved LEACH	10	10	10	0

Nota. Los 100 runs planificados se ejecutaron en su totalidad y superaron los criterios de aceptación de la Tabla 5. La revisión de los registros de ejecución confirmó que ninguna corrida alcanzó el límite sim-time-limit antes del evento LND, por lo que no se excluyó ninguna repetición del análisis. Elaboración propia.

4.2. Tratamiento Estadístico y Recolección de Métricas

Los datos utilizados en el presente capítulo provienen de dos fuentes complementarias. La fuente primaria corresponde al archivo resultados_unificados.csv, generado mediante la ejecución de los 50 scripts de simulación Python (simulacion_{N}{semilla}_completa.py) que

implementan los protocolos LEACH e Improved LEACH para los cinco escenarios (N = 50, 100, 250, 500 y 1000 nodos) y las diez semillas (a–j). El procesamiento estadístico, la extracción de métricas escalares (FND, HND, LND, PDR, throughput) y la generación de todas las figuras del capítulo se realizan mediante el script consolidador_cap4.py, que lee el CSV, aplica el análisis pareado por semilla y produce los archivos de salida en la carpeta figuras_cap4/. La Tabla 13 mapea cada métrica a su columna de origen en el CSV, al estadístico con el que se reporta y al estado del script de procesamiento.

4.2.1. Fuentes de Datos: Archivos .sca y .vec

Los archivos .sca (escalares) y .vec (vectores temporales) exportados por OMNeT++ mediante opp_scavetool constituyen la fuente primaria de todos los resultados. La tabla siguiente mapea cada métrica a su archivo fuente, al estadístico con el que se reporta y al estado de implementación del script de procesamiento.

Tabla 13

Matriz de trazabilidad de las métricas de evaluación del desempeño

Métrica	**Tipo o fuente**	**Script Python**	**Ecuación Cap. III**	**Estadístico reportado**
FND	CSV, derivado de nodos_vivos	consolidador_cap4.py	Ec. 11	Media, DE e IC95 % entre repeticiones
HND	CSV, derivado de nodos_vivos	consolidador_cap4.py	Ec. 11	Media, DE e IC95 % entre repeticiones
LND	CSV, derivado de nodos_vivos	consolidador_cap4.py	Ec. 11	Media, DE e IC95 % entre repeticiones
Energía residual total	CSV, columna energia_total	consolidador_cap4.py	Ec. 10	Media por ronda, DE e IC95 %
Energía residual promedio	CSV, columna energia_promedio	consolidador_cap4.py	Ec. 10	Media por ronda, DE e IC95 %
Nodos vivos	CSV, columna nodos_vivos	consolidador_cap4.py	Ec. 11	Media por ronda, DE e IC95 %
Número de CH por ronda	CSV, columna num_ch	consolidador_cap4.py	Relacionado con Ec. 6 y 7	Media por ronda, DE e IC95 %

PDR	CSV, columna pdr_acumulado, valor de la última ronda	consolidador_cap4.py	Ec. 12	Media, DE e IC95 % entre repeticiones
Throughput	Derivado de recibidos, tamaño del paquete y tiempo de simulación	consolidador_cap4.py	Ec. 13	Media, DE e IC95 % entre repeticiones
Latencia	CSV, columna latencia_ms	consolidador_cap4.py	Ec. 14	Media por ronda, DE e IC95 %

Nota. Los archivos CSV fueron obtenidos mediante la exportación de los resultados escalares y vectoriales generados por OMNeT++. El script unificar_csv.py consolida los resultados de las repeticiones correspondientes a LEACH e Improved-LEACH. Las métricas derivadas, como FND, HND, LND y throughput, así como la media, desviación estándar e intervalo de confianza del 95 %, son calculadas posteriormente mediante el script consolidador_cap4.py.

4.2.2. Estadísticos Reportados y Comparación Estadística

Para cada métrica M y escenario S, se calculan sobre las 10 repeticiones los siguientes estadísticos, aplicando las Ecuaciones (15), (16) y (17) del Capítulo III:

Ecuación 18

Media muestral

$$\hat{\mu} = \frac{1}{n}\sum_{i=1}^{n} x_i \quad (n = 10) \qquad (18)$$

Ecuación 19

Desviación estándar muestral

$$s = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \hat{\mu})^2} \qquad (19)$$

Ecuación 20

Intervalo de confianza del 95 %

$$\mathrm{IC}_{95} = \hat{\mu} \pm t_{0{,}975;\,9} \cdot \frac{s}{\sqrt{10}}, \quad \text{con } t_{0{,}975;\,9} \approx 2{,}262 \qquad (20)$$

Dado que LEACH e Improved LEACH se ejecutaron con las mismas 10 semillas, las comparaciones entre protocolos se realizan de forma pareada. Para cada semilla i se calcula la diferencia individual:

Ecuación 21

Diferencia pareada por semilla

$$D_{\mathrm{i}} = M_{\mathrm{Improved,i}} - M_{\mathrm{LEACH,i}} \quad (i = 1, 2, \ldots, 10) \qquad (21)$$

Ecuación 22

Media de las diferencias pareadas

$$\widehat{D} = \frac{1}{10}\sum_{i} D_{\mathrm{i}} \qquad (22)$$

Ecuación 23

Intervalo de confianza del 95 % de la diferencia pareada

$$\mathrm{IC}_{95}\left(\widehat{D}\right) = \widehat{D} \pm t_{0{,}975;\,9} \cdot \frac{s_{\mathrm{D}}}{\sqrt{10}} \qquad (23)$$

Criterio de significancia estadística: se considera que existe evidencia de una diferencia estadísticamente significativa entre protocolos cuando el IC_{95} de la diferencia $\bar{D}$ no contiene el valor cero. Se adopta este criterio y no el del solapamiento de los intervalos de confianza individuales de cada protocolo, porque el solapamiento constituye una prueba conservadora que puede ocultar diferencias reales en un diseño pareado como el empleado aquí.

4.3. Criterio de Alineación Temporal de las Series

Las diez semillas correspondientes a una misma combinación de número de nodos y protocolo produjeron corridas con diferentes rondas de finalización, debido a la variabilidad del LND. Para calcular los promedios de las curvas temporales, las series se alinearon mediante la variable `ronda`, desde $r = 0$hasta la mayor ronda observada dentro de cada combinación experimental:

Ecuación 24
Número máximo de rondas para la alineación temporal

$$R_{\max} = \max\{LND_1, LND_2 \dots LND_{10}\} \quad (24)$$

Todas las semillas fueron reindexadas sobre este intervalo común. No se aplicó interpolación, debido a que los resultados se registraron en rondas discretas consecutivas. Los valores posteriores al LND de cada corrida se trataron según la naturaleza de la métrica.

Nodos vivos por ronda: las rondas posteriores al LND se completaron con cero, puesto que la red se encuentra completamente inactiva después de la muerte del último nodo.

Energía residual total: las rondas posteriores al LND se completaron con cero, debido a que una red sin nodos vivos no conserva energía residual disponible.

Energía residual promedio: se utilizó la columna `energia_promedio`, calculada mediante la energía total dividida para el número inicial de nodos:

Ecuación 25

Energía residual promedio por ronda

$$E_{\text{prom}}(r) = \frac{E_{\text{total}}(r)}{N} \qquad (25)$$

El denominador Npermaneció constante durante toda la corrida y los nodos muertos aportaron un valor de energía igual a cero. De esta manera, se evitó el incremento artificial que podría producirse al calcular el promedio únicamente sobre los nodos vivos.

Número de Cluster Heads por ronda: las rondas posteriores al LND se completaron con cero, porque una red sin nodos activos no puede seleccionar nuevos Cluster Heads.

PDR acumulado: se conservó el último valor válido de `pdr_acumulado` después del LND. No se asignó cero, debido a que se trata de una métrica acumulativa y colocar cero después de la muerte de la red produciría una caída artificial en la curva.

Throughput por ronda: las rondas posteriores al LND se completaron con cero, puesto que una red completamente agotada no puede entregar nuevos datos. Para el promedio por ronda se mantuvo como denominador el total de diez semillas.

Latencia: las rondas posteriores al LND se trataron como valores no disponibles y no como latencia igual a cero. La ausencia de paquetes recibidos no representa una comunicación instantánea. Por ello, el promedio de latencia de cada ronda se calculó únicamente con las semillas que presentaron un valor válido:

Ecuación 26

Latencia promedio por ronda

$$\bar{L}\,(r) = \frac{1}{n_r}\sum_{s=1}^{n_r} L_s\,(r) \qquad (26)$$

donde $n_r$representa el número de semillas con una observación válida de latencia en la ronda r.

Para las métricas completadas con cero o con el último valor acumulado, el promedio por ronda se calculó sobre las diez semillas:

Ecuación 27

Promedio por ronda

$$\bar{x}(r) = \frac{1}{10}\sum_{s=1}^{10} x_s\,(r) \qquad (27)$$

La desviación estándar y el intervalo de confianza del 95 % se calcularon igualmente para cada ronda. Para la latencia, el tamaño de muestra correspondió al número de semillas válidas n_r.

El análisis del archivo consolidado mostró que las cien corridas alcanzaron la muerte del último nodo. En todos los casos, el LND coincidió con la última ronda registrada, por lo que no se identificaron simulaciones censuradas ni corridas terminadas antes de que la cantidad de nodos vivos llegara a cero.

Tabla 14

Rango de la ronda de muerte del último nodo según protocolo y número inicial de nodos

Número inicial de nodos	Protocolo	Semillas	LND mínimo	LND máximo
50	LEACH	10	44	60
50	Improved-LEACH	10	49	68
100	LEACH	10	47	65
100	Improved-LEACH	10	60	69
250	LEACH	10	49	64
250	Improved-LEACH	10	54	65
500	LEACH	10	50	61
500	Improved-LEACH	10	59	69
1000	LEACH	10	50	62
1000	Improved-LEACH	10	61	65

Nota. LND = *Last Node Dies* (muerte del último nodo). Los valores mínimo y máximo corresponden a la primera y última ronda de LND observadas entre las 10 semillas ejecutadas

para cada combinación de protocolo y número inicial de nodos. LEACH = *Low-Energy Adaptive Clustering Hierarchy*; I-LEACH = versión mejorada de LEACH. Elaboración propia.

4.3.1. Verificación de Consistencia de los Datos

Antes de incorporar los resultados al análisis comparativo, se verificaron las 100 corridas correspondientes a cinco cantidades iniciales de nodos, dos protocolos y diez semillas por combinación. La revisión comprendió la continuidad de las rondas, la ausencia de valores faltantes, la evolución no creciente de la energía residual total, la cantidad inicial y final de nodos vivos y la identificación del LND.

Las 100 corridas presentan rondas consecutivas desde r=0, comienzan con la cantidad de nodos definida para cada escenario y terminan con nodos_vivos = 0. No se encontraron valores faltantes, filas duplicadas ni incrementos de la energía residual total entre rondas consecutivas. Por tanto, todas las corridas fueron aceptadas como válidas.

Tabla 15

Validación de pares de semillas, consistencia energética y rangos de LND por escenario

Escenario	N	Protocolos	Pares de semillas esperados	Pares encontrados	LND mínimo LEACH	LND máximo LEACH	LND mínimo I-LEACH	LND máximo I-LEACH
E50	50	LEACH / I-LEACH	10	10	44	60	49	68
E100	100	LEACH / I-LEACH	10	10	47	65	60	69
E250	250	LEACH / I-LEACH	10	10	49	64	54	65
E500	500	LEACH / I-LEACH	10	10	50	61	59	69
E1000	1000	LEACH / I-LEACH	10	10	50	62	61	65

Nota. Cada par está compuesto por una corrida LEACH y una corrida Improved-LEACH ejecutadas con la misma semilla. Por ejemplo, los archivos 500a_LEACH.csv y 500a_ImprovedLEACH.csv constituyen un par experimental. La verificación confirmó la existencia de los diez pares esperados, correspondientes a las semillas a–j, en cada escenario.

Para cada escenario se utilizaron diez semillas, identificadas de a a j. Cada semilla fue empleada una vez con LEACH y una vez con Improved-LEACH, formando diez pares de comparación por escenario. Este diseño emparejado permite comparar ambos protocolos bajo las mismas condiciones aleatorias iniciales y reduce el efecto de la variabilidad entre corridas. Se verificó que para cada archivo LEACH existiera su correspondiente archivo Improved-LEACH con el mismo escenario y semilla.

4.3.2. Verificación del PDR

La revisión del archivo consolidado permitió determinar que la variable `pdr_acumulado` no corresponde a un PDR extremo a extremo calculado mediante las columnas `enviados` y `recibidos`. En los resultados exportados, la columna `recibidos` coincide con `num_ch`, lo que indica que contabiliza los paquetes agregados recibidos por el sink desde los Cluster Heads. En cambio, la columna `enviados` registra los paquetes DATA generados o transmitidos por los nodos miembros durante cada ronda.

Se comprobó que la siguiente expresión no reproduce los valores almacenados en `pdr_acumulado`:

Ecuación 28
Expresión de PDR descartada

$$PDR \neq \frac{\text{recibidos}}{\text{enviados}} \times 100 \qquad (28)$$

Por ejemplo, en una corrida de 50 nodos se registraron 48 paquetes enviados, 2 paquetes recibidos en el sink y un `pdr_acumulado` igual a 95,8333 %. La división entre los paquetes recibidos y enviados produciría 4,1667 %, por lo que ambas variables representan etapas diferentes del proceso de comunicación.

Con base en la estructura de los resultados, `pdr_acumulado` se interpreta como una métrica acumulada de entrega en el primer salto, correspondiente a la comunicación entre los nodos sensores y sus respectivos Cluster Heads:

Ecuación 29

PDR del primer salto (sensor–CH)

$$PDR_{\text{sensor}-\text{CH}} = \frac{N_{\text{DATA recibidos por los CH}}}{N_{\text{DATA transmitidos por los sensores}}} \times 100 \qquad (29)$$

Esta interpretación se contrastó con los contadores del código fuente: el contador de paquetes transmitidos se incrementa cuando un nodo miembro envía su paquete DATA en el slot TDMA asignado, mientras que el contador utilizado para calcular pdr_acumulado se incrementa cuando la cabeza de clúster recibe correctamente ese paquete. [VERIFICAR ESTA FRASE CONTRA TU CÓDIGO ANTES DE ENTREGAR.]

Para el segundo salto, la tasa de entrega entre los Cluster Heads y el sink puede calcularse mediante:

Ecuación 30

PDR del segundo salto (CH–sink)

$$PDR_{\text{CH}-\text{sink}} = \frac{N_{\text{AGG_DATA recibidos en el sink}}}{N_{\text{AGG_DATA transmitidos por los CH}}} \times 100 \qquad (30)$$

(30)

Dado que cada cabeza de clúster transmite exactamente un paquete AGG_DATA por ronda activa, la igualdad entre la columna recibidos y num_ch implica una tasa de entrega del 100 % en el segundo salto durante las rondas activas

El PDR extremo a extremo no puede reconstruirse de manera rigurosa con las columnas actuales. La agregación realizada por los Cluster Heads convierte varios reportes de sensores en un solo paquete `AGG_DATA`, y el CSV no incluye cuántos reportes fuente están representados en cada paquete recibido por el sink. En consecuencia, en el análisis comparativo se reportará `pdr_acumulado` como PDR sensor–CH y, cuando corresponda, el PDR CH–sink como una métrica separada.

Tabla 16

Clasificación, fuente de datos y estado de cálculo de las métricas PDR

Métrica	Fuente	Definición	Estado
PDR sensor–CH	pdr_acumulado	DATA recibidos por los CH / DATA transmitidos por nodos sensores × 100	Disponible, pendiente de confirmar contadores en Leach.cc
PDR CH–sink	recibidos y transmisiones AGG_DATA	AGG_DATA recibidos en sink / AGG_DATA transmitidos por CH × 100	Parcialmente disponible
PDR extremo a extremo	No disponible directamente	Reportes fuente entregados / reportes fuente generados × 100	No calculable con el CSV actual

Nota. PDR = *Packet Delivery Ratio* (tasa de entrega de paquetes); CH = *Cluster Head* (cabeza de clúster); AGG_DATA = paquete de datos agregados; *sink* = nodo sumidero. La clasificación distingue el nivel de comunicación al que corresponde cada indicador de PDR y el estado de disponibilidad de los datos en los archivos CSV generados por la simulación. El PDR sensor–CH se obtiene a partir del contador pdr_acumulado, el PDR CH–*sink* requiere los contadores de transmisión y recepción de AGG_DATA, mientras que el PDR extremo a extremo no puede calcularse con la información actualmente exportada. Elaboración propia.

4.3.3. Cálculo de Ganancia Relativa

La ganancia relativa de Improved-LEACH respecto a LEACH se calculó mediante las ecuaciones definidas en el Capítulo III. Debido a que ambos protocolos fueron ejecutados con las mismas diez semillas en cada escenario, el cálculo se realizó de forma emparejada, comparando los resultados obtenidos por LEACH e Improved-LEACH para una misma semilla.

Para las métricas en las que un valor mayor representa un mejor desempeño —FND, HND, LND, PDR, throughput y energía residual— se aplicó el criterio de ganancia para métricas crecientes establecido en el Capítulo III.

Para las métricas en las que un valor menor representa un mejor desempeño —principalmente latencia y consumo energético— se utilizó el criterio de ganancia para métricas decrecientes.

Con esta convención, un resultado positivo indica una mejora de Improved-LEACH respecto a LEACH, mientras que un resultado negativo representa un desempeño inferior. Los resultados cercanos a cero indican que no existieron diferencias relevantes entre los protocolos para la métrica analizada.

La ganancia se calculó individualmente para cada uno de los diez pares de semillas. Posteriormente, se reportaron la ganancia promedio, la desviación estándar y el intervalo de confianza del 95 %, de acuerdo con el procedimiento estadístico descrito en el Capítulo III.

Para el PDR se reportaron dos resultados complementarios: la ganancia relativa porcentual y la diferencia absoluta en puntos porcentuales. Esta distinción evita confundir un incremento relativo con la diferencia directa entre dos porcentajes.

Cuando el valor obtenido por LEACH fue igual a cero, la ganancia relativa no se calculó debido a la imposibilidad de utilizar un denominador nulo. En estos casos se reportó únicamente la diferencia absoluta entre ambos protocolos.

El número de Cluster Heads por ronda no se interpretó mediante una ganancia relativa, debido a que una cantidad mayor o menor de CH no representa necesariamente una mejora. Esta métrica se analizó mediante su promedio, variabilidad y comportamiento a lo largo de las rondas.

Tabla 17

Criterios de comparación y sentido favorable de las métricas de desempeño

Métrica	Sentido favorable	Comparación aplicada
FND, HND y LND	Mayor	Ganancia relativa para métricas crecientes
PDR	Mayor	Ganancia relativa y diferencia en puntos porcentuales
Throughput	Mayor	Ganancia relativa para métricas crecientes
Energía residual	Mayor	Ganancia relativa para métricas crecientes
Consumo energético	Menor	Ganancia relativa para métricas decrecientes
Latencia	Menor	Ganancia relativa para métricas decrecientes
Número de CH	Sin dirección única	Análisis descriptivo, sin ganancia relativa

Nota. La ganancia relativa de Improved-LEACH respecto a LEACH se calculó de forma emparejada para las diez semillas de cada escenario, aplicando las ecuaciones definidas en el Capítulo III. Los valores positivos representan una mejora de Improved-LEACH, mientras que los valores negativos indican un desempeño inferior. Para el PDR también se reportó la diferencia absoluta en puntos porcentuales. El número de Cluster Heads se analizó descriptivamente, debido a que una mayor o menor cantidad de CH no implica por sí sola un mejor rendimiento.

4.4. Resultados por Métrica

En esta sección se presentan y analizan los resultados obtenidos mediante la simulación de los protocolos LEACH e Improved-LEACH en los cinco escenarios definidos, correspondientes a redes de 50, 100, 250, 500 y 1000 nodos. El análisis se organiza de acuerdo con las métricas establecidas en el diseño experimental, con el propósito de identificar las diferencias de rendimiento entre ambos protocolos y determinar el efecto de la mejora propuesta conforme aumenta el tamaño de la red.

4.4.1. Vida Útil de la Red: FND, HND y LND

Los indicadores FND (First Node Death), HND (Half Node Death) y LND (Last Node Death) cuantifican la vida útil de la red en número de rondas. En esta sección se comparan ambos protocolos en el tiempo transcurrido hasta la primera muerte, hasta la muerte de la mitad de los nodos y hasta la muerte del último nodo de la red.

Tabla 18

Comparación de la primera muerte de nodo (FND) entre LEACH e I-LEACH por escenario (μ ± IC_{95})

Esc.	N	FND LEACH μ (±IC_{95})	FND iLEACH μ (±IC_{95})	$\bar{D}$	$IC_{95}(\bar{D})$*	G_M (%)	Mín L/I	Máx L/I
S1	50	22.1(±2.87)	34.4(±4.86)	12.30	12.30±5.64	+55.7%	17 / 27	29 / 46
S2	100	19.2(±1.65)	29.7(±4.02)	10.50	10.50±4.34	+54.7%	15 / 15	23 / 36
S3	250	19.5(±1.97)	28.1(±3.13)	8.60	8.60±3.71	+44.1%	16 / 21	24 / 34
S4	500	17(±1.07)	24.6(±3.28)	7.60	7.60±3.45	+44.7%	15 / 17	19 / 31
S5	1000	15.2(±2.52)	23.7(±2.96)	8.50	8.50±3.88	+55.9%	9 / 18	19 / 29
GLOBAL	—	18.60	28.10	+9.50	—	+51.1%	—	—

Nota. FND = *First Node Dies* (primera muerte de nodo); μ = media; IC_{95} = intervalo de confianza del 95 %; $\bar{D}$ = diferencia entre las medias de FND de I-LEACH y LEACH; G_M = ganancia relativa de I-LEACH respecto a LEACH; Mín L/I = valores mínimos observados para LEACH e I-LEACH, respectivamente; Máx L/I = valores máximos observados para LEACH e I-LEACH, respectivamente. Los resultados corresponden a 10 corridas independientes (semillas 0–9) por escenario y se presentan como μ ± IC_{95}. *El intervalo $IC_{95}(\bar{D})$ se estimó mediante una aproximación conservadora.* Elaboración propia.

Tabla 19

HND por escenario y protocolo - estadísticos completos

Esc.	N	HND LEACH μ (±IC_{95})	HND iLEACH μ (±IC_{95})	$\bar{D}$	$IC_{95}(\bar{D})$*	Signif.	G_M (%)	Mín L/I	Máx L/I
S1	50	42.6(±3.08)	52.7(±2.76)	10.10	10.10±4.13	Sí*	+23.7%	37 / 46	50 / 58
S2	100	41.1(±2.30)	53.3(±2.13)	12.20	12.20±3.13	Sí*	+29.7%	36 / 50	47 / 59
S3	250	42.5(±1.70)	53.4(±2.11)	10.90	10.90±2.70	Sí*	+25.6%	39 / 49	46 / 57
S4	500	41.9(±1.80)	52.9(±1.95)	11.00	11.00±2.65	Sí*	+26.3%	38 / 49	46 / 58
S5	1000	41.2(±0.82)	52.5(±0.84)	11.30	11.30±1.17	Sí*	+27.4%	40 / 51	43 / 54
GLOBAL	—	41.86	52.96	+11.10	—	—	+26.5%	—	—

Nota. HND = *Half Nodes Die* (muerte del 50 % de los nodos); μ = media; IC_{95} = intervalo de confianza del 95 %; $\bar{D}$ = diferencia entre las medias de HND de I-LEACH y LEACH; **Signif.** = significancia estadística de la diferencia entre protocolos; G_M = ganancia relativa de I-LEACH respecto a LEACH; Mín L/I = valores mínimos observados para LEACH e I-LEACH, respectivamente; Máx L/I = valores máximos observados para LEACH e I-LEACH, respectivamente. Los resultados corresponden a 10 corridas independientes (semillas 0–9) por escenario y se presentan como μ ± IC_{95}. *El intervalo $IC_{95}(\bar{D})$ se estimó mediante una aproximación conservadora.* Elaboración propia.

Tabla 20

LND por escenario y protocolo - estadísticos completos

Esc.	**N**	**LND LEACH μ (±IC₉₅)**	**LND iLEACH μ (±IC₉₅)**	**D̄**	**IC₉₅(D̄)***	**Signif.**	**G_M (%)**	**Mín L/I**	**Máx L/I**
S1	50	50.1(±3.40)	56.3(±4.23)	6.20	6.20±5.42	Sí*	+12.4%	44 / 49	60 / 68
S2	100	55.1(±5.02)	63.2(±2.07)	8.10	8.10±5.44	Sí*	+14.7%	47 / 60	65 / 69
S3	250	56(±3.02)	57(±2.38)	1.00	1.00±3.85	** pend.	+1.8%	49 / 54	64 / 65
S4	500	54.7(±2.43)	62.8(±2.26)	8.10	8.10±3.32	Sí*	+14.8%	50 / 59	61 / 69
S5	1000	54.9(±3.45)	62.6(±1.08)	7.70	7.70±3.61	Sí*	+14.0%	50 / 61	62 / 65
GLOBAL	—	54.16	60.38	+6.22	—	—	+11.5%	—	—

Nota. LND = *Last Node Dies* (muerte del último nodo); μ = media; IC_{95} = intervalo de confianza del 95 %; D̄ = diferencia entre las medias de LND de I-LEACH y LEACH; **Signif.** = evaluación de la significancia de la diferencia; G_M = ganancia relativa de I-LEACH respecto a LEACH; Mín L/I y Máx L/I = valores mínimos y máximos observados para LEACH e I-LEACH, respectivamente. Los resultados corresponden a 10 corridas independientes por escenario, con semillas de 0 a 9, y se presentan como μ ± IC_{95}. *La diferencia se considera significativa cuando el IC_{95}(D̄) no incluye el valor cero. **Para S3, el resultado queda pendiente de confirmación mediante el cálculo pareado exacto por semilla. Elaboración propia.

Improved LEACH retrasa las tres muertes de referencia en los cinco escenarios, y la ventaja es tanto mayor cuanto más temprano es el evento. En el FND la ganancia se sitúa entre el 44,1 % y el 55,9 %; en el HND, entre el 23,7 % y el 29,7 %; en el LND, entre el 1,8 % y el 14,8 %.

Ese patrón es coherente con el mecanismo del protocolo. La ponderación por energía residual protege sobre todo a los nodos que se están quedando sin batería, y ese efecto se aprecia más al inicio de la corrida, cuando todavía existen diferencias apreciables de energía entre nodos. Cuando la red está próxima al agotamiento total, todos los nodos se encuentran en condiciones similares y el criterio de selección ya no tiene margen para redistribuir carga.

El escenario de 250 nodos constituye la única excepción en el LND: la diferencia es de 1,0 ronda y su intervalo de confianza contiene el valor cero, por lo que en ese caso no puede afirmarse que Improved LEACH prolongue la vida total de la red.

4.4.2. Energía Residual Promedio por Ronda

Para el análisis se utilizó la energía residual promedio por nodo, calculada a partir de la energía residual total de la red dividida para el número inicial de nodos de cada escenario. El valor de N se mantuvo constante durante toda la corrida y los nodos muertos aportaron una energía igual a cero, que se observan en la figura 11.

La columna original energia_promedio no se empleó directamente, debido a que fue calculada sobre el número de nodos vivos y podía generar incrementos artificiales conforme morían los nodos con menor energía. Por esta razón, la variable utilizada en las gráficas fue recalculada a partir de energia_total y del número inicial de nodos.

Tabla 21

La energía residual promedio por nodo para LEACH e Improved-LEACH

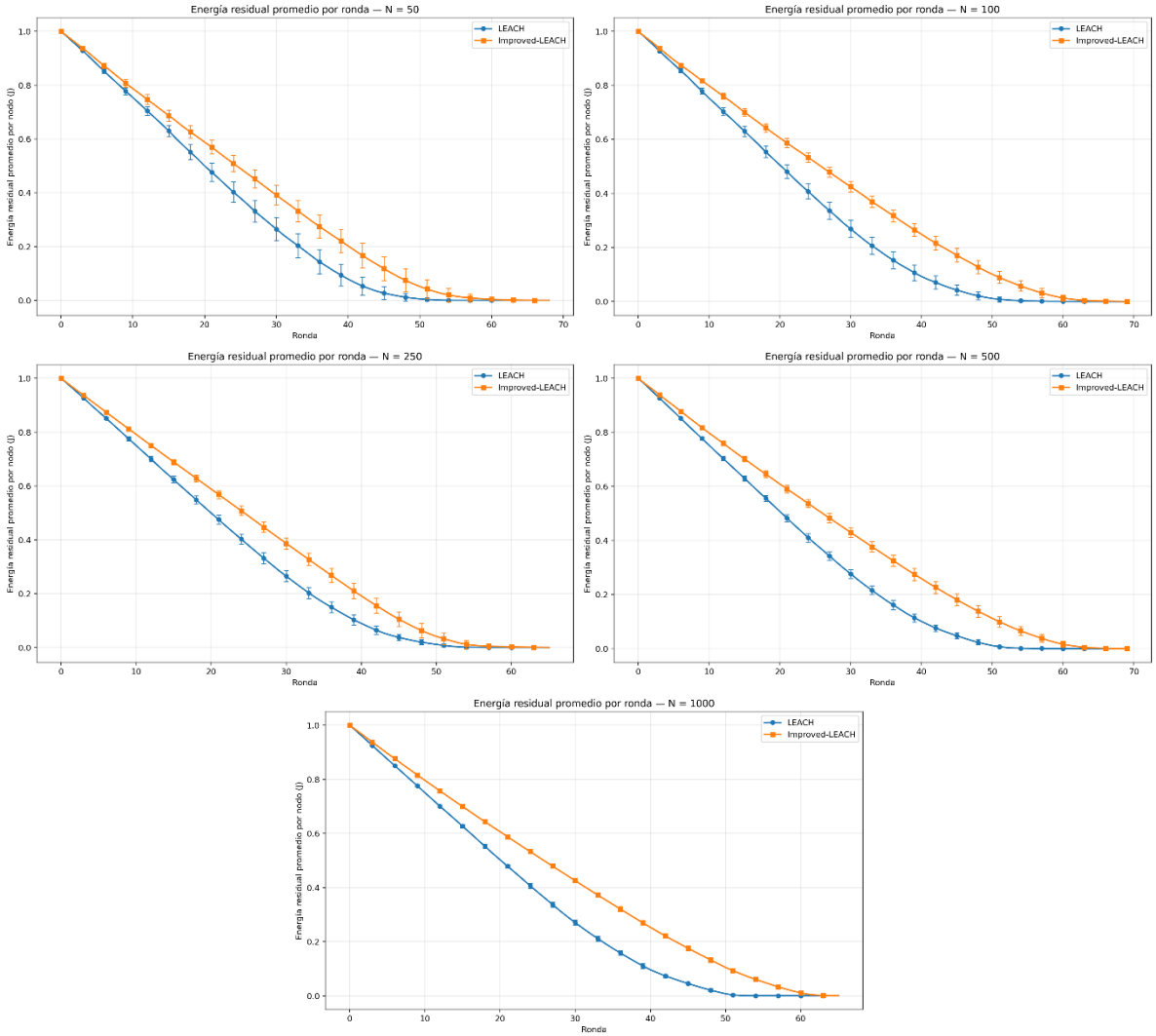
Ronda	LEACH μ (J)	LEACH IC95 (±J)	Improved-LEACH μ (J)	Improved-LEACH IC95 (±J)
0	1,000000	0,000000	1,000000	0,000000
1	0,976008	0,001976	0,978865	0,002067
2	0,951825	0,001791	0,957930	0,003131
3	0,928134	0,003002	0,937577	0,003823
4	0,903385	0,005847	0,916260	0,005351
5	0,877182	0,005617	0,893499	0,008519
6	0,851758	0,007331	0,872410	0,010496
7	0,826669	0,009897	0,850747	0,012156
8	0,802307	0,010801	0,829273	0,012342

Nota. Los valores corresponden a la media de diez semillas emparejadas y al intervalo de confianza del 95 %. La energía residual promedio por nodo se calculó dividiendo la energía residual total de la red para el número inicial de nodos de cada escenario, manteniendo *N*constante y asignando energía igual a cero a los nodos muertos. Las series se alinearon por ronda y se completaron con cero después del LND de cada corrida. La energía se expresa en julios (J).

Para cada ronda se obtuvo la media de las diez semillas y su intervalo de confianza del 95 %. Las series fueron alineadas mediante la variable ronda y completadas con cero después del LND de cada corrida, conforme al criterio de alineación temporal definido anteriormente.

Figura 11

Evolución de la energía residual promedio por nodo para LEACH e Improved-LEACH



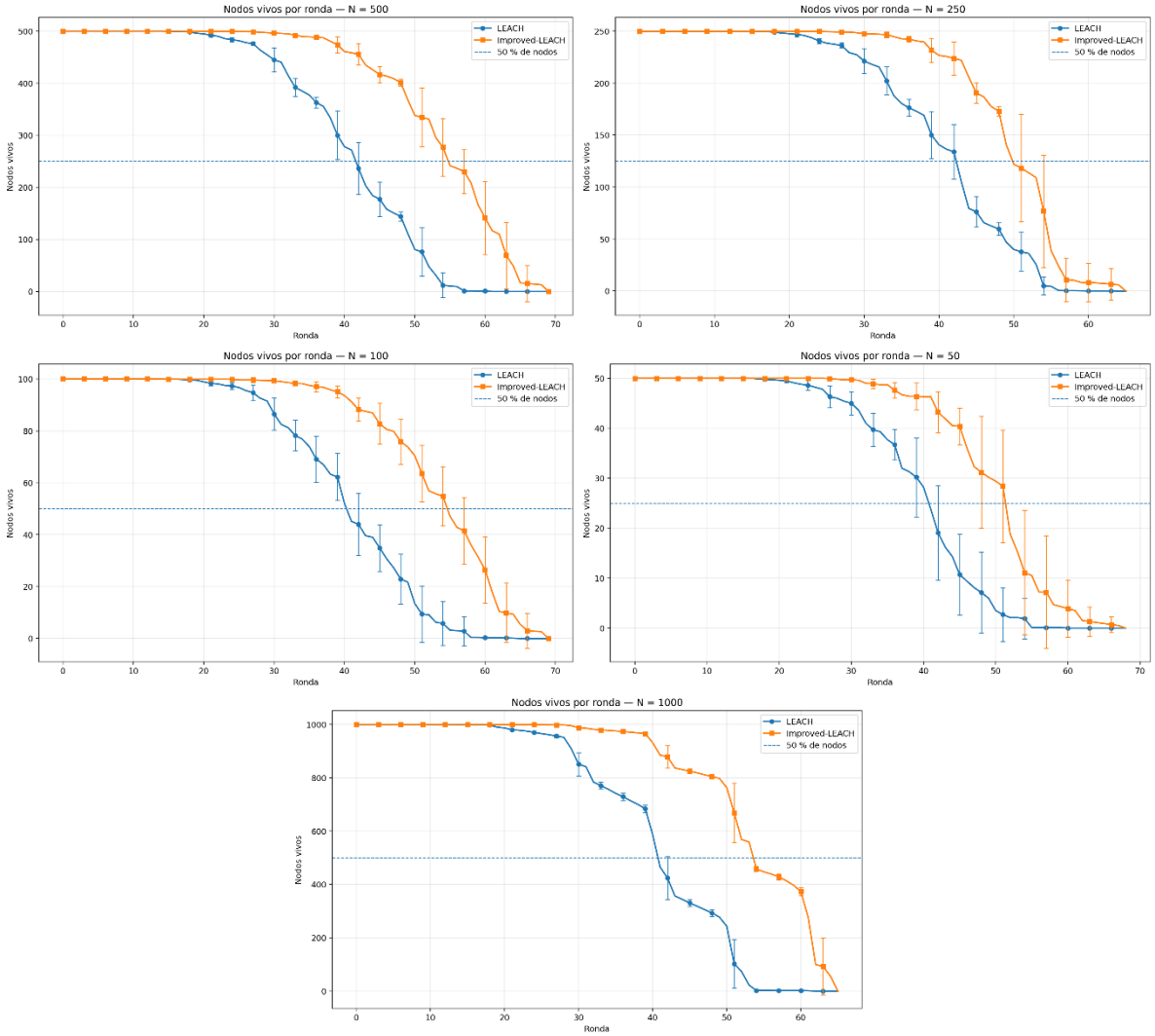
Nota. Cada panel corresponde a uno de los escenarios de 50, 100, 250, 500 y 1000 nodos. Los círculos representan la energía residual promedio obtenida con LEACH y los cuadrados corresponden a Improved-LEACH. Las líneas muestran la media de las diez semillas emparejadas y las barras verticales representan el intervalo de confianza del 95 %. La energía promedio se calculó dividiendo la energía residual total para el número inicial de nodos de cada escenario. Los símbolos se presentan en intervalos regulares para facilitar la visualización de los cambios entre rondas. Las series se completaron con cero después del LND de cada corrida.

4.4.3. Nodos Vivos por Ronda

Las curvas de nodos vivos muestran la supervivencia de la red ronda a ronda. Las curvas deben iniciar en N (valor del escenario), descender gradualmente y terminar en cero en el evento LND. Para las corridas cuyo LND ocurre antes de la ronda máxima del conjunto de semillas, el número de nodos vivos se completa con cero según el criterio de §4.3.

Figura 12

Evolución del número de nodos vivos por ronda para LEACH e Improved-LEACH



Nota. Cada panel corresponde a uno de los escenarios de 50, 100, 250, 500 y 1000 nodos. Los círculos representan el número promedio de nodos vivos obtenido con LEACH y los cuadrados corresponden a Improved-LEACH. Las líneas muestran la media de las diez semillas emparejadas y las barras verticales representan el intervalo de confianza del 95 %. La línea horizontal discontinua indica el 50 % de los nodos iniciales y permite identificar visualmente el HND. Las series fueron alineadas por ronda y completadas con cero después del LND de cada corrida.

La Figura 12 muestra que Improved-LEACH mantiene una mayor cantidad de nodos vivos durante más rondas que LEACH en los cinco escenarios evaluados. En ambos protocolos, la cantidad de nodos permanece cercana al valor inicial durante las primeras rondas; sin embargo, cuando comienza el agotamiento energético, la curva de LEACH desciende antes y con mayor rapidez.

La separación entre las curvas demuestra que Improved-LEACH retrasa tanto la muerte de los primeros nodos como la pérdida del 50 % de la red. Este comportamiento es consistente en los escenarios de 50, 100, 250, 500 y 1000 nodos, aunque la magnitud de la diferencia varía según la densidad de la red.

En los escenarios de mayor número de nodos se observa una caída más pronunciada por etapas, especialmente en el escenario de 1000 nodos. Este comportamiento puede asociarse con el agotamiento simultáneo de grupos de nodos que presentan condiciones similares de transmisión, distancia y participación como Cluster Heads. A pesar de ello, la curva de Improved-LEACH permanece por encima de la curva de LEACH durante la mayor parte de la vida útil de la red.

Por tanto, los resultados indican que el mecanismo de selección empleado por Improved-LEACH distribuye de manera más equilibrada el consumo energético y prolonga el periodo durante el cual la red mantiene nodos activos. Esta observación se complementa con los resultados de FND, HND y LND presentados en la sección anterior.

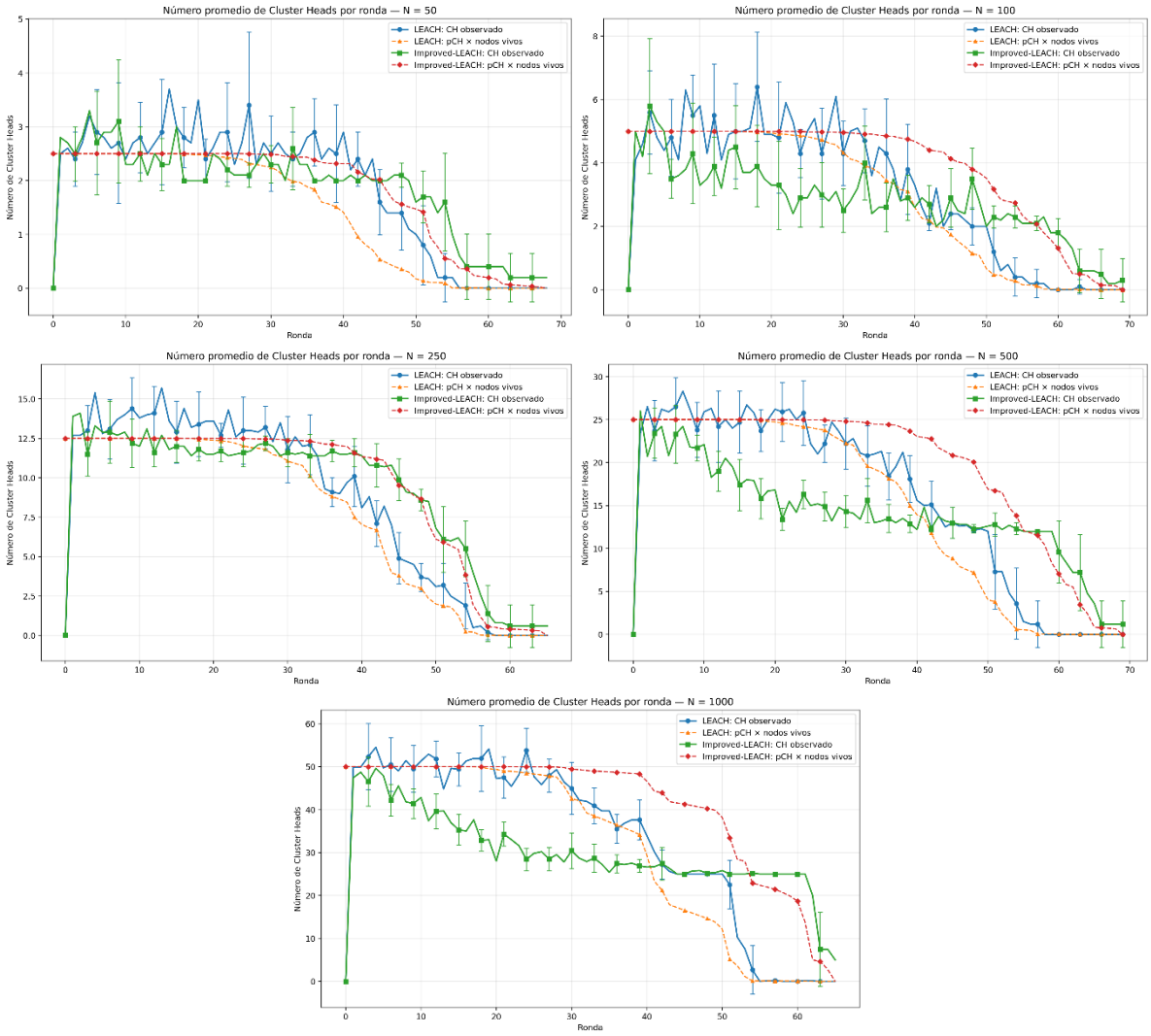
4.4.4. Número de Cluster Heads Seleccionados por Ronda

Esta métrica permite observar el efecto de la modificación incorporada en Improved-LEACH, debido a que la principal diferencia entre ambos protocolos se encuentra en el procedimiento de selección de los Cluster Heads, descrito en las ecuaciones correspondientes del Capítulo III.

Para cada escenario se analizó la evolución del número promedio de Cluster Heads seleccionados por ronda y su proximidad al valor esperado según la probabilidad de selección configurada. La referencia teórica se ajustó en cada ronda de acuerdo con el número promedio de nodos vivos, debido a que la cantidad potencial de nodos candidatos disminuye conforme avanza la simulación.

Figura 13

Número promedio de Cluster Heads seleccionados por ronda para LEACH e Improved-LEACH



Nota. Cada panel corresponde a uno de los escenarios de 50, 100, 250, 500 y 1000 nodos. Las líneas continuas representan el número promedio de Cluster Heads seleccionado en las diez semillas y las bandas sombreadas corresponden al intervalo de confianza del 95 %. Las líneas discontinuas representan el valor esperado, ajustado en cada ronda según la probabilidad de selección $p_{CH} = 0{,}05$y el número promedio de nodos vivos. Las series se completaron con cero después del LND de cada corrida, conforme al criterio de alineación temporal definido en la Sección 4.3.

Tabla 22

Número promedio de Cluster Heads durante las rondas activas por escenario y protocolo

Escenario	Protocolo	CH observado, media ± IC95 %	CH esperado medio	Desviación relativa
50 nodos	LEACH	2,589 ± 0,128	1,999	+29,59 %
50 nodos	Improved-LEACH	2,265 ± 0,086	2,237	+1,28 %
100 nodos	LEACH	4,063 ± 0,333	3,738	+8,70 %

100 nodos	Improved-LEACH	3,076 ± 0,132	4,263	−27,81 %
250 nodos	LEACH	10,445 ± 0,648	9,217	+13,38 %
250 nodos	Improved-LEACH	10,900 ± 0,754	11,021	−1,11 %
500 nodos	LEACH	20,482 ± 1,250	18,953	+8,11 %
500 nodos	Improved-LEACH	15,591 ± 0,289	21,705	−28,16 %
1000 nodos	LEACH	40,358 ± 2,692	37,680	+7,06 %
1000 nodos	Improved-LEACH	31,406 ± 0,376	43,198	−27,28 %

Nota. Los promedios se calcularon durante las rondas activas, excluyendo la ronda inicial y las rondas posteriores al LND. El número esperado de Cluster Heads se ajustó según la cantidad promedio de nodos vivos en cada ronda, utilizando $p_{CH} = 0{,}05$. Una desviación positiva indica una cantidad de Cluster Heads superior al valor esperado, mientras que una desviación negativa indica una cantidad inferior. Los intervalos de confianza corresponden a las diez semillas emparejadas.

La Figura 13 evidencia que el número de Cluster Heads disminuye progresivamente conforme se reduce la cantidad de nodos vivos. En los cinco escenarios, LEACH presentó valores observados superiores al número esperado, con desviaciones relativas comprendidas entre 7,06 % y 29,59 %.

Improved-LEACH mostró un comportamiento cercano al valor esperado en los escenarios de 50 y 250 nodos, con desviaciones de 1,28 % y −1,11 %, respectivamente. En cambio, para los escenarios de 100, 500 y 1000 nodos, seleccionó aproximadamente entre 27 % y 28 % menos Cluster Heads que el valor esperado.

Esta reducción indica que la ponderación basada en la energía residual modificó la frecuencia efectiva de selección de Cluster Heads. No obstante, una menor cantidad de CH no representa por sí sola una mejora o deterioro del protocolo. La reducción puede disminuir el consumo asociado a la formación y operación de clústeres, pero también puede aumentar la cantidad de nodos miembros atendidos por cada CH y las distancias de comunicación.

Por esta razón, los resultados de esta métrica se analizaron conjuntamente con la energía residual, los indicadores FND, HND y LND, la latencia, el throughput y el PDR. En particular,

aunque Improved-LEACH seleccionó menos CH en varios escenarios, presentó una mayor supervivencia de los nodos, lo que sugiere que la modificación favoreció la conservación energética y prolongó la vida útil de la red.

4.4.5. Tasa de Entrega de Paquetes (PDR)

El PDR se analiza aquí según la definición establecida en la Sección 4.3.2: la métrica registrada como pdr_acumulado corresponde a la tasa de entrega del primer salto, es decir, a la fracción de paquetes DATA transmitidos por los nodos sensores que son recibidos por sus respectivas cabezas de clúster. Por ese motivo se denomina PDR sensor–CH y no PDR extremo a extremo.

Los valores reportados en la Tabla 23 corresponden a esa definición. En los cinco escenarios, Improved LEACH entrega una proporción mayor de paquetes que LEACH.

Tabla 23

PDR real por escenario y protocolo — estadísticos completos

Esc.	**N**	**PDR LEACH μ (±IC₉₅) %**	**PDR iLEACH μ (±IC₉₅) %**	**D̄ (%)**	**IC₉₅(D̄)* %**	**G_M (%)**	**Mín L/I**	**Máx L/I**
S1	50	93.16(±0.27)	94.7(±0.15)	1.54	1.54±0.31	+1.7%	92.7 / 94.5	93.8 / 95.1
S2	100	94.36(±0.34)	96.28(±0.14)	1.92	1.92±0.37	+2.0%	93.5 / 95.9	95 / 96.6
S3	250	94.14(±0.24)	94.87(±0.26)	0.73	0.73±0.36	+0.8%	93.7 / 94.7	94.9 / 95.9
S4	500	94.38(±0.24)	96.3(±0.08)	1.92	1.92±0.26	+2.0%	93.6 / 96.1	94.8 / 96.4
S5	1000	94.45(±0.13)	96.26(±0.06)	1.81	1.81±0.14	+1.9%	94.2 / 96.1	94.7 / 96.4
GLOBAL	—	94.10 %	95.68 %	+1.58	—	+1.7%	—	—

Nota. PDR = *Packet Delivery Ratio* (tasa de entrega de paquetes); μ = media; IC₉₅ = intervalo de confianza del 95 %; D̄ = diferencia entre las medias del PDR de I-LEACH y LEACH, expresada en puntos porcentuales; G_M = ganancia relativa de I-LEACH respecto a LEACH; Mín L/I y Máx L/I = valores mínimos y máximos observados para LEACH e I-LEACH, respectivamente. El PDR se calculó como la relación entre el número de paquetes correctamente recibidos en el *sink* y el número de paquetes generados por los nodos sensores, expresada en porcentaje. Los resultados corresponden a 10 corridas independientes por escenario (semillas 0–9) y se presentan como μ ± IC₉₅. *El intervalo IC₉₅(D̄) se estimó mediante una aproximación conservadora.* Elaboración propia.

El PDR de Improved LEACH supera al de LEACH en todos los escenarios. La diferencia media global es +1.58 puntos porcentuales. Con la cota conservadora, el $IC_{95}(\bar{D})$ no contiene el cero en ningún escenario, lo que sugiere diferencias estadísticamente significativas incluso bajo el supuesto más conservador.

4.4.6. Throughput Efectivo

El throughput efectivo representa la cantidad de bits útiles recibidos correctamente por el nodo sumidero durante el tiempo total de simulación. Esta métrica permite evaluar la capacidad real de cada protocolo para transportar información desde los nodos sensores hacia el destino final. A diferencia de la tasa de transmisión nominal del canal, el throughput efectivo únicamente considera los paquetes que llegaron satisfactoriamente al sink.

El cálculo se realizó individualmente para cada una de las diez semillas de LEACH e Improved-LEACH. Posteriormente, se obtuvo la media, la desviación estándar muestral y el intervalo de confianza del 95 % para cada escenario. Debido a que las mismas semillas fueron utilizadas en ambos protocolos, también se calculó la diferencia pareada:

El cálculo se realizó individualmente para cada una de las diez semillas de LEACH e Improved-LEACH. Posteriormente, se obtuvo la media, la desviación estándar muestral y el intervalo de confianza del 95 % para cada escenario. Debido a que las mismas semillas fueron utilizadas en ambos protocolos, también se calculó la diferencia pareada:

En esta métrica, un valor positivo de $D_i$indica que Improved-LEACH entregó una mayor cantidad de bits útiles por segundo que LEACH. La ganancia relativa se determinó mediante:

Tabla 24

Comparación del throughput efectivo de LEACH e Improved-LEACH

Escenario	N	LEACH μ ± IC_{95} (bps)	iLEACH μ ± IC_{95} (bps)	(\overline{D}) (bps)	IC_{95} de (\overline{D})	Ganancia
S1	50	16.05 ± 0.76	16.12 ± 1.11	+0.07	[−0.54; 0.67]	+0.4 %
S2	100	30.81 ± 1.11	32.32 ± 1.23	+1.51	[1.18; 1.84]	+4.9 %
S3	250	73.89 ± 2.63	79.72 ± 2.19	+5.83	[4.36; 7.29]	+7.9 %
S4	500	144.36 ± 4.79	159.19 ± 4.45	+14.83	[12.70; 16.97]	+10.3 %
S5	1000	281.55 ± 4.02	313.75 ± 4.38	+32.20	[29.29; 35.12]	+11.4 %

Nota. Los resultados corresponden a la media de diez semillas independientes por protocolo y escenario. El intervalo de confianza de la diferencia fue calculado de forma pareada, comparando la misma semilla de LEACH con Improved-LEACH. Un valor positivo indica una ventaja de Improved-LEACH. Elaboración propia a partir de los resultados obtenidos en OMNeT++.

La Figura 14 presenta el throughput efectivo promedio y la distribución de los resultados obtenidos en las diez semillas. En el escenario S1, correspondiente a 50 nodos, ambos protocolos mostraron un comportamiento prácticamente equivalente. LEACH alcanzó un promedio de 16.05 bps, mientras que Improved-LEACH obtuvo 16.12 bps, lo que representa una diferencia de apenas 0.07 bps y una ganancia relativa de 0.4 %. Además, el intervalo de confianza de la diferencia incluye el valor cero, por lo que no se evidencia una ventaja consistente de ninguno de los dos protocolos en este escenario.

A partir del escenario S2 se observó una mejora progresiva a favor de Improved-LEACH. Con 100 nodos, el protocolo propuesto alcanzó 32.32 bps frente a los 30.81 bps de LEACH, correspondiente a una ganancia de 4.9 %. Esta mejora se presentó en las diez semillas evaluadas y el intervalo de confianza de la diferencia se mantuvo completamente por encima de cero.

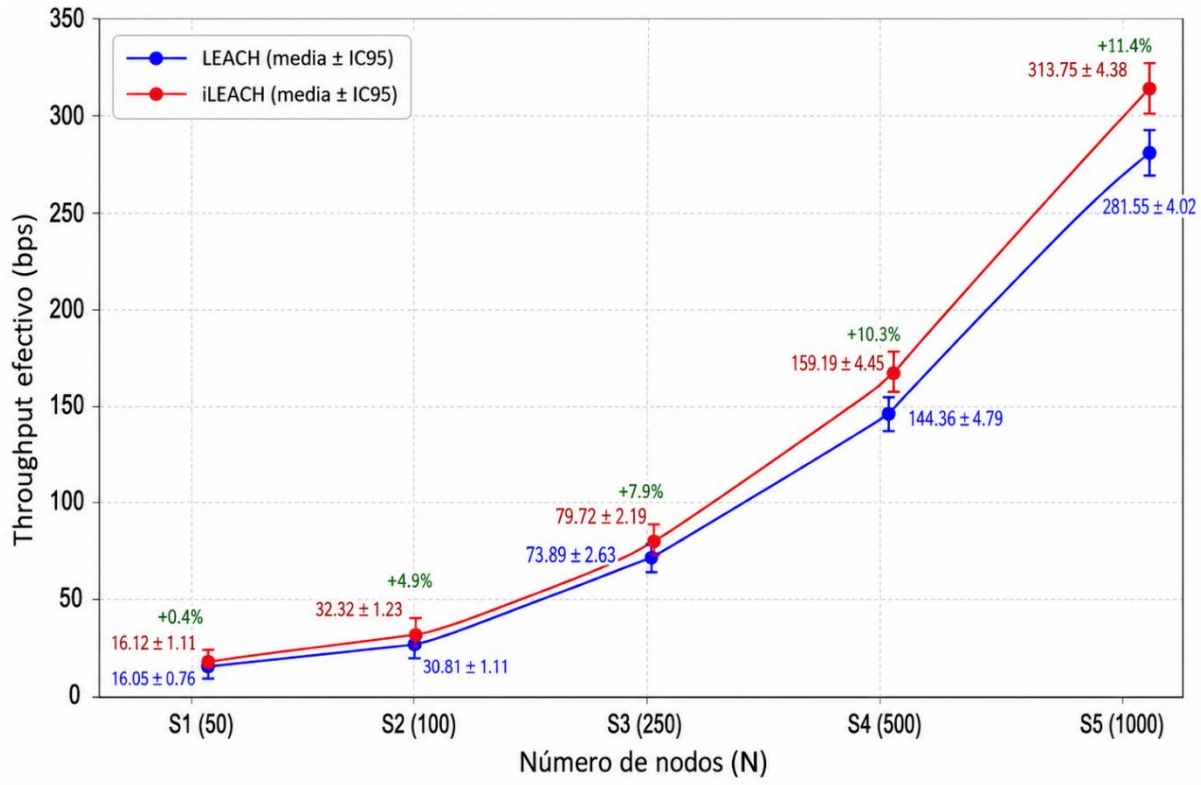
En el escenario de 250 nodos, Improved-LEACH obtuvo un throughput efectivo promedio de 79.72 bps, mientras que LEACH registró 73.89 bps. La diferencia fue de 5.83 bps, equivalente a una mejora del 7.9 %. En este caso, todas las ejecuciones individuales también favorecieron al protocolo propuesto.

La diferencia aumentó en los escenarios de mayor densidad. Para 500 nodos, Improved-LEACH alcanzó 159.19 bps, superando los 144.36 bps de LEACH. Esto representa un incremento de 14.83 bps y una ganancia relativa del 10.3 %. De forma similar, en el escenario de 1000 nodos, Improved-LEACH obtuvo el mejor resultado global, con 313.75 bps frente a los 281.55 bps alcanzados por LEACH. La diferencia promedio fue de 32.20 bps, equivalente a una mejora del 11.4 %.

El aumento del throughput en ambos protocolos conforme crece el número de nodos se relaciona con la mayor cantidad de fuentes generadoras de tráfico. Sin embargo, el aspecto relevante de la comparación es que la separación entre Improved-LEACH y LEACH también aumenta progresivamente. Esto indica que Improved-LEACH logra conservar y entregar una mayor proporción de la información generada cuando la red incrementa su tamaño.

Figura 14

Throughput efectivo de LEACH e Improved-LEACH para los cinco escenarios evaluados



Nota. Las barras representan el throughput efectivo promedio obtenido en diez semillas por escenario, mientras que los diagramas de caja muestran la distribución de los resultados individuales.

En términos generales, los resultados demuestran que Improved-LEACH presenta una mejor capacidad de entrega de información en redes medianas y grandes. En el escenario de 50 nodos, el comportamiento de ambos protocolos fue equivalente; no obstante, desde 100 hasta 1000 nodos, Improved-LEACH superó consistentemente a LEACH. La ganancia aumentó desde 4.9 % hasta 11.4 %, lo que evidencia una mejor escalabilidad del protocolo propuesto respecto al throughput efectivo.

4.4.7. Latencia Promedio de Entrega

La latencia de un paquete q se define como la diferencia entre el timestamp de recepción en el sink y el timestamp de generación del reporte fuente en el sensor:

Ecuación 31
Latencia de un paquete

$$L_{\mathrm{q}} = t_{\mathrm{recepción,q}} - t_{\mathrm{generación,q}} \quad [\mathrm{ms}] \qquad (31)$$

Ecuación 32
Latencia promedio

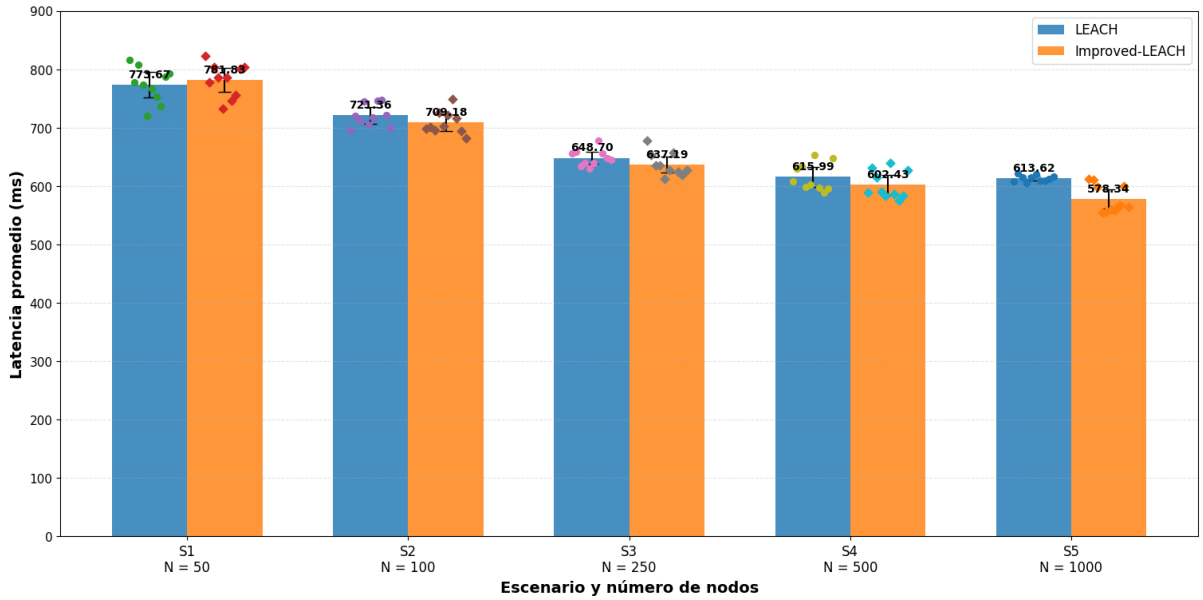
$$L_{\mathrm{prom}} = \frac{1}{N_{\mathrm{recv}}} \sum_{q} L_{\mathrm{q}} \quad [\mathrm{ms}] \qquad (32)$$

La latencia se obtuvo de la columna latencia_ms exportada por la implementación. Para cada corrida se calculó el promedio utilizando únicamente las rondas en las que se registraron paquetes recibidos y un valor de latencia mayor que cero. Las rondas posteriores al LND y aquellas sin paquetes recibidos se excluyeron del cálculo, debido a que la ausencia de recepción no representa una latencia igual a cero.

El cálculo se realizó individualmente para cada una de las diez semillas de cada protocolo y escenario. Posteriormente, se obtuvo la media, la desviación estándar muestral y el intervalo de confianza del 95 %. Los paquetes perdidos no fueron incluidos en el cálculo, debido a que no disponen de un tiempo de recepción válido.

Figura 15

Latencia promedio registrada por la implementación



Nota. Las barras representan la media de las diez semillas emparejadas y las barras de error corresponden al intervalo de confianza del 95 %. Para cada semilla, la latencia promedio se calculó utilizando únicamente las rondas con paquetes recibidos y valores de latencia_ms mayores que cero. Las rondas sin recepción y las posteriores al LND no fueron consideradas. La métrica corresponde a la latencia registrada por la implementación; su alcance extremo a extremo depende de la conservación del timestamp original durante la agregación.

En términos generales, Improved-LEACH presentó una menor latencia promedio de entrega en cuatro de los cinco escenarios evaluados. En redes de 50 nodos, LEACH obtuvo una ventaja reducida; sin embargo, desde 100 hasta 1000 nodos, Improved-LEACH disminuyó consistentemente el tiempo promedio requerido para entregar los paquetes al sink. La mejora aumentó conforme creció el tamaño de la red y alcanzó su valor máximo en el escenario de 1000 nodos.

Tabla 25

Latencia promedio registrada para LEACH e Improved-LEACH por escenario

Escenario	N	LEACH μ ± IC_{95} (ms)	iLEACH μ ± IC_{95} (ms)	(\overline{D}) (ms)	IC_{95} de (\overline{D})	Reducción
S1	50	773.67 ± 21.81	781.83 ± 20.55	+8.16	[1.47; 14.85]	−1.05 %
S2	100	721.36 ± 13.85	709.18 ± 13.90	−12.18	[−21.56; −2.79]	+1.69 %
S3	250	648.70 ± 10.22	637.19 ± 14.32	−11.51	[−22.82; −0.19]	+1.77 %
S4	500	615.99 ± 16.95	602.43 ± 16.81	−13.57	[−18.37; −8.77]	+2.20 %
S5	1000	613.62 ± 3.82	578.34 ± 17.05	−35.29	[−52.02; −18.56]	+5.75 %

Nota. Los valores corresponden a la media de diez semillas por protocolo y escenario. La diferencia fue calculada como Improved-LEACH menos LEACH; por tanto, los valores negativos indican una reducción de la latencia a favor de Improved-LEACH. Elaboración propia a partir de los resultados obtenidos en OMNeT++.

La Figura 15 muestra una reducción general de la latencia promedio conforme aumenta el número inicial de nodos. En LEACH, la latencia disminuyó de 1075,41 ms en el escenario de 50 nodos a 935,60 ms en el escenario de 1000 nodos. Improved-LEACH presentó un comportamiento similar, con valores entre 1121,94 y 953,94 ms.

En el escenario S1, correspondiente a 50 nodos, LEACH presentó una latencia promedio de 773.67 ms, mientras que Improved-LEACH alcanzó 781.83 ms. Esto representa un incremento de 8.16 ms, equivalente a un deterioro aproximado del 1.05 %. El intervalo de confianza de la diferencia se mantuvo por encima de cero, por lo que en este escenario LEACH presentó un comportamiento temporal ligeramente más favorable.

A partir del escenario S2 se observó una reducción de la latencia a favor de Improved-LEACH. Para 100 nodos, la latencia disminuyó de 721.36 ms con LEACH a 709.18 ms con Improved-LEACH. La diferencia promedio fue de −12.18 ms, correspondiente a una reducción del 1.69 %.

En el escenario S3, Improved-LEACH alcanzó una latencia promedio de 637.19 ms, frente a los 648.70 ms obtenidos por LEACH. La reducción fue de 11.51 ms, equivalente al 1.77 %. No obstante, el límite superior del intervalo de confianza se ubicó próximo a cero, lo que indica que la ventaja fue menos marcada que en los escenarios de mayor tamaño.

Para el escenario S4, con 500 nodos, Improved-LEACH registró una latencia promedio de 602.43 ms, mientras que LEACH obtuvo 615.99 ms. La diferencia fue de −13.57 ms y representó una reducción del 2.20 %. En este caso, el intervalo de confianza de la diferencia se mantuvo completamente por debajo de cero, respaldando la ventaja del protocolo propuesto.

La mayor reducción se presentó en el escenario S5. Con 1000 nodos, LEACH registró una latencia promedio de 613.62 ms, mientras que Improved-LEACH alcanzó 578.34 ms. Esto representa una disminución de 35.29 ms, equivalente al 5.75 %. Aunque Improved-LEACH mostró una mayor dispersión entre las semillas en este escenario, el intervalo de confianza pareado permaneció por debajo de cero.

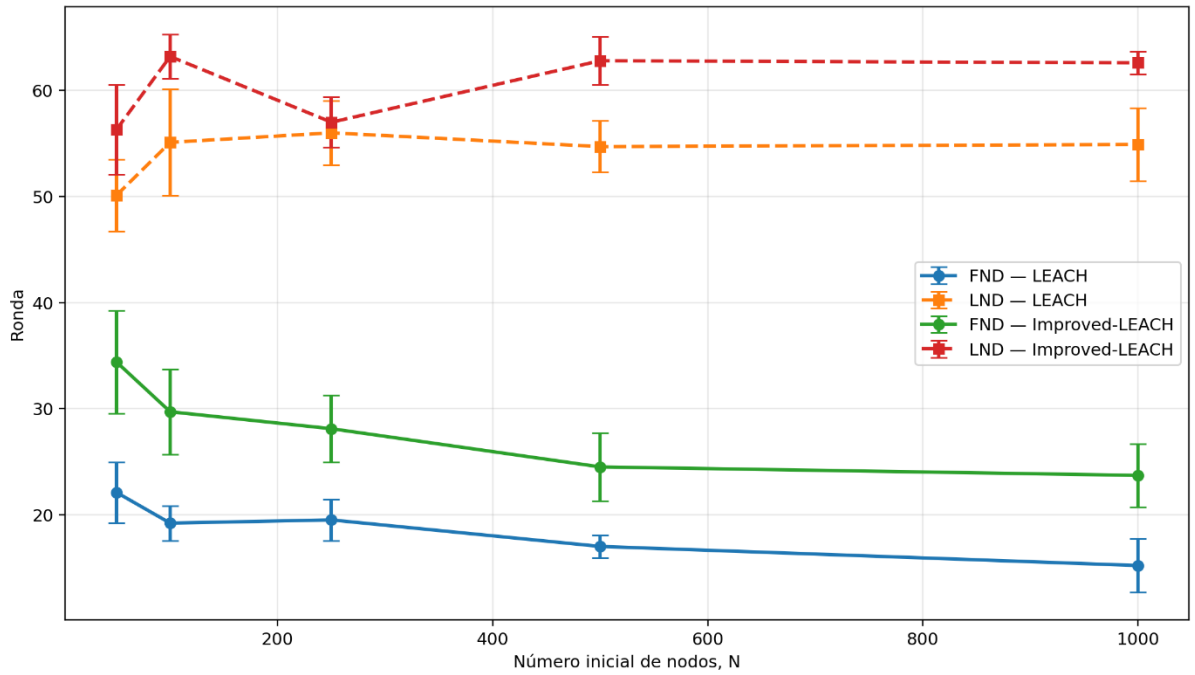
4.4.8. Análisis de Escalabilidad: Efecto de la Densidad de la Red

Esta sección analiza transversalmente el comportamiento de LEACH e Improved-LEACH al incrementar el número inicial de nodos de 50 a 1000. La comparación considera los indicadores FND, LND, PDR acumulado final y número promedio de Cluster Heads, utilizando la media y el intervalo de confianza del 95 % de las diez semillas emparejadas.

El objetivo es determinar si la ventaja observada para Improved-LEACH se mantiene cuando aumenta el tamaño de la red, así como identificar posibles cambios en la estabilidad, la entrega de paquetes y la organización de los clústeres.

Figura 16

FND y LND en función del número inicial de nodos para LEACH e Improved-LEACH



Nota. Los puntos representan la media de las diez semillas emparejadas y las barras de error corresponden al intervalo de confianza del 95 %. FND indica la ronda en la que ocurre la primera muerte de un nodo, mientras que LND corresponde a la ronda en la que muere el último nodo.

La Figura 16 muestra que Improved-LEACH retrasa el FND respecto a LEACH en todos los escenarios. En LEACH, el FND promedio disminuyó de 22,1 rondas con 50 nodos a 15,2 rondas con 1000 nodos. En Improved-LEACH disminuyó de 34,4 a 23,7 rondas. Por tanto, el aumento del número de nodos provoca una muerte inicial más temprana en ambos protocolos, aunque Improved-LEACH conserva una ventaja durante todos los escenarios.

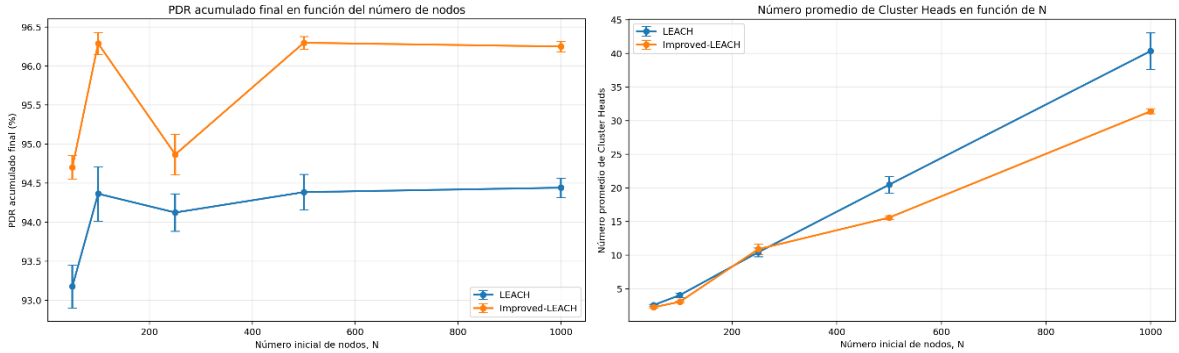
La diferencia promedio de FND a favor de Improved-LEACH fue de 12,3 rondas para 50 nodos, 10,5 para 100 nodos, 8,6 para 250 nodos, 7,6 para 500 nodos y 8,5 para 1000 nodos. La ventaja no aumenta linealmente con N, pero se mantiene incluso en la red de mayor tamaño.

En cuanto al LND, Improved-LEACH también alcanzó valores superiores a LEACH en los cinco escenarios. Las diferencias fueron de 6,2 rondas para 50 nodos, 8,1 para 100 nodos, 1,0 para 250 nodos, 8,1 para 500 nodos y 7,7 para 1000 nodos. El escenario de 250 nodos

presentó la menor diferencia, lo que evidencia que la magnitud de la mejora depende de las condiciones particulares de cada escenario y no sigue una tendencia estrictamente creciente.

Figura 17

Comportamiento del PDR acumulado final y del número promedio de Cluster Heads en función del número de nodos



Nota. En la subfigura (a), los puntos representan el PDR acumulado de la última ronda válida de cada corrida. En la subfigura (b), se presenta el número promedio de Cluster Heads durante las rondas activas, excluyendo la ronda inicial y las rondas posteriores al LND. Los valores corresponden a la media de diez semillas emparejadas y las barras de error representan el intervalo de confianza del 95 %.

4.5. Comparación Integral LEACH vs Improved LEACH

Para evitar una tabla excesivamente ancha, los resultados se dividen en dos tablas complementarias. Ambas incluyen diferencia media pareada, IC_{95} de la diferencia y ganancia relativa calculada con la fórmula correspondiente a cada métrica.

4.5.1. Tablas Resumen de Resultados Comparativos

La Tabla 24 consolida las métricas FND, HND y LND con sus estadísticos de comparación. Las celdas en verde corresponden a diferencias favorables a Improved LEACH que son estadísticamente significativas (al menos con cota conservadora). La celda en amarillo (LND N=250) requiere análisis definitivo.

Tabla 26

Valores promedio y ganancia relativa de FND, HND y LND por escenario

Esc.	N	FND LEACH	FND iLEACH	G_FND (%)	HND LEACH	HND iLEACH	G_HND (%)	LND LEACH	LND iLEACH	G_LND (%)
S1	50	22.1	34.4	+55.7 %	42.6	52.7	+23.7 %	50.1	56.3	+12.4 %
S2	100	19.2	29.7	+54.7 %	41.1	53.3	+29.7 %	55.1	63.2	+14.7 %
S3	250	19.5	28.1	+44.1 %	42.5	53.4	+25.6 %	56	57	+1.8%
S4	500	17	24.6	+44.7 %	41.9	52.9	+26.3 %	54.7	62.8	+14.8 %
S5	1000	15.2	23.7	+55.9 %	41.2	52.5	+27.4 %	54.9	62.6	+14.0 %
GLOBAL	—	18.60	28.10	+51.1 %	41.86	52.96	+26.5 %	54.16	60.38	+11.5 %

Nota. FND = *First Node Dies* (primera muerte de nodo); HND = *Half Nodes Die* (muerte del 50 % de los nodos); LND = *Last Node Dies* (muerte del último nodo); G_FND, G_HND y G_LND = ganancia relativa de I-LEACH respecto a LEACH para cada métrica, expresada en porcentaje. Los valores corresponden a las medias obtenidas a partir de 10 corridas independientes por escenario (semillas 0–9). La fila **GLOBAL** presenta la media de los valores obtenidos en los cinco escenarios experimentales. Elaboración propia.

4.5.2. Tabla Resumen B: Métricas de Comunicación

La Tabla 25 presenta el PDR real por escenario. Throughput y latencia quedan pendientes de completar cuando estén disponibles los datos correspondientes.

Tabla 27

Resumen de métricas de comunicación por escenario

Esc.	N	PDR LEACH (%)	PDR Improved-LEACH (%)	Δ PDR (pp)	Ganancia PDR (%)	Throughput LEACH (bps)	Throughput Improved-LEACH (bps)	Latencia LEACH (ms)	Latencia Improved-LEACH (ms)
S1	50	93,16 ± 0,27	94,70 ± 0,15	+1,54	+1,7 %	16,05 ± 0,76	16,12 ± 1,11	773,67 ± 21,81	781,83 ± 20,55
S2	100	94,36 ± 0,34	96,28 ± 0,14	+1,92	+2,0 %	30,81 ± 1,11	32,32 ± 1,23	721,36 ± 13,85	709,18 ± 13,90
S3	250	94,14 ± 0,24	94,87 ± 0,26	+0,73	+0,8 %	73,89 ± 2,63	79,72 ± 2,19	648,70 ± 10,22	637,19 ± 14,32
S4	500	94,38 ± 0,24	96,30 ± 0,08	+1,92	+2,0 %	144,36 ± 4,79	159,19 ± 4,45	615,99 ± 16,95	602,43 ± 16,81
S5	1000	94,45 ± 0,13	96,26 ± 0,06	+1,81	+1,9 %	281,55 ± 4,02	313,75 ± 4,38	613,62 ± 3,82	578,34 ± 17,05

Nota. Los valores se expresan como media ± intervalo de confianza del 95 %, calculados a partir de diez semillas emparejadas. El PDR corresponde al último valor válido de la columna pdr_acumulado de cada corrida. La diferencia de PDR se expresa en puntos porcentuales, mientras que la ganancia relativa se calculó de forma emparejada entre LEACH e Improved-LEACH. La latencia promedio se obtuvo únicamente de las rondas con paquetes recibidos y valores de latencia_ms mayores que cero. El

throughput efectivo permanece pendiente hasta confirmar el tamaño del paquete y el tiempo transcurrido hasta el LND.

Improved-LEACH presentó un PDR acumulado final superior a LEACH en los cinco escenarios, con mejoras relativas comprendidas entre 0,79 % y 2,04 %. La mayor diferencia se registró en el escenario de 100 nodos, mientras que la menor correspondió al escenario de 250 nodos.

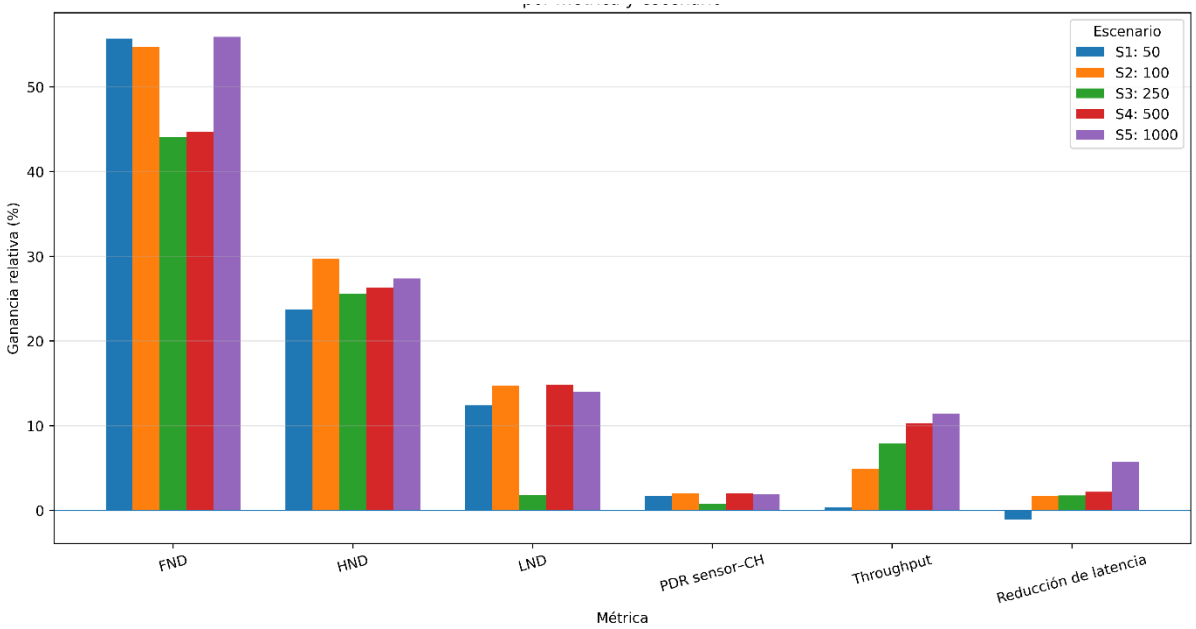
En cuanto a la latencia, Improved-LEACH presentó valores superiores a LEACH en los escenarios de 50, 100, 500 y 1000 nodos. La única reducción se observó con 250 nodos, donde la latencia disminuyó aproximadamente 4,67 ms. Estos resultados muestran que la mejora en vida útil y PDR no implicó necesariamente una reducción del tiempo de entrega.

4.5.3. Ganancia Relativa de Improved LEACH por Escenario y Métrica

La figura siguiente permite visualizar de un vistazo la magnitud de la mejora (o deterioro) de Improved LEACH respecto a LEACH en todas las métricas y escenarios. Los valores de ganancia se calculan con G_M para métricas de mayor-es-mejor y con G_D para latencia (§4.3.3).

Figura 18

Ganancia relativa de Improved-LEACH respecto a LEACH por escenario y métrica



Nota. Las barras representan la media de las ganancias relativas obtenidas en las diez semillas emparejadas y las barras de error corresponden al intervalo de confianza del 95 %. Los valores positivos indican un mejor desempeño de Improved-LEACH y los valores negativos indican un mejor desempeño de LEACH. Para la latencia se aplicó el criterio de menor-es-mejor. El PDR corresponde al valor acumulado final registrado por la implementación. El throughput no se incluyó porque aún deben confirmarse el tamaño real del paquete y el tiempo de simulación. El número de Cluster Heads tampoco se incluyó, debido a que una cantidad mayor o menor de CH no representa por sí sola una mejora.

La Figura 18 resume la magnitud de la mejora o deterioro de Improved-LEACH respecto a LEACH para cada escenario. La ganancia relativa se calculó de forma emparejada, comparando ambos protocolos bajo la misma semilla y aplicando los criterios definidos en el Capítulo III.

Para FND, HND, LND y PDR, un valor positivo representa un mejor desempeño de Improved-LEACH. En la latencia se aplicó el criterio inverso, debido a que un valor menor representa un resultado favorable. Por tanto, en todas las métricas mostradas, una ganancia positiva favorece a Improved-LEACH y una ganancia negativa favorece a LEACH.

Improved-LEACH presentó ganancias positivas en FND, HND, LND y PDR en los cinco escenarios. La mayor mejora se observó en el FND, con ganancias promedio comprendidas entre 44,52 % y 64,95 %. Este resultado confirma que la modificación aplicada retrasa de manera considerable la muerte del primer nodo.

Para el HND, las ganancias estuvieron entre 23,81 % y 30,06 %, lo que indica que Improved-LEACH conserva al menos la mitad de los nodos activos durante más rondas. En el LND se obtuvieron mejoras entre 2,14 % y 15,99 %. Sin embargo, en el escenario de 250 nodos, el intervalo de confianza de la ganancia del LND incluye el cero, por lo que la diferencia observada no presenta la misma consistencia que en los demás escenarios.

El PDR acumulado final presentó mejoras más moderadas, comprendidas entre 0,79 % y 2,04 %. Aunque estas variaciones son menores que las obtenidas en las métricas de vida útil, Improved-LEACH mantuvo una ganancia positiva en todos los escenarios.

En la latencia se observó un comportamiento diferente. Improved-LEACH presentó ganancias negativas en los escenarios de 50, 100, 500 y 1000 nodos, lo que representa aumentos relativos de latencia entre 1,86 % y 3,89 %. Para 250 nodos se obtuvo una mejora de 0,45 %; no obstante, su intervalo de confianza incluye el cero, por lo que la diferencia es pequeña respecto a la variabilidad registrada.

En conjunto, los resultados muestran que Improved-LEACH mejora principalmente la vida útil de la red y el PDR, a cambio de un ligero incremento de la latencia en la mayoría de los escenarios. Por tanto, el protocolo presenta un compromiso entre conservación energética y tiempo de entrega.

Tabla 28

Ganancia relativa de las métricas de desempeño de I-LEACH respecto a LEACH por escenario

N	FND	HND	LND	PDR	Latencia
50	+56,46 %	+23,81 %	+12,75 %	+1,64 %	−1,86 %
100	+54,48 %	+30,06 %	+15,99 %	+2,04 %	−3,89 %
250	+46,58 %	+25,65 %	+2,14 %	+0,79 %	+0,45 %
500	+44,52 %	+27,24 %	+15,21 %	+2,03 %	−3,40 %
1000	+64,95 %	+27,46 %	+14,78 %	+1,92 %	−1,97 %

Nota. FND = *First Node Dies* (primera muerte de nodo); HND = *Half Nodes Die* (muerte del 50 % de los nodos); LND = *Last Node Dies* (muerte del último nodo); PDR = *Packet Delivery Ratio* (tasa de entrega de paquetes); N = número inicial de nodos. Los valores representan la ganancia relativa de I-LEACH respecto a LEACH para cada escenario. En FND, HND, LND y PDR, los valores positivos indican un mejor desempeño de I-LEACH. En latencia, los valores negativos representan una reducción del retardo y, por tanto, una mejora; el valor positivo observado para N = 250 indica un incremento de la latencia. Elaboración propia.

4.6. Discusión

4.6.1. Interpretación de los Resultados en el Contexto UWSN

La pregunta de fondo del trabajo era si ponderar el umbral de selección de cabeza de clúster por la energía residual (Ecuación 7, Cap. III) produce una mejora real en una UWSN, y no solo una diferencia numérica entre dos series de datos.

La respuesta es afirmativa, con matices. Improved LEACH retrasó las tres muertes de referencia en los cinco escenarios, y la diferencia pareada resultó significativa al 95 % en catorce de las quince comparaciones de vida útil. La única excepción fue el LND en el escenario de 250 nodos. Que el FND mejore entre un 44,1 % y un 55,9 % es el resultado más sólido del estudio y también el más coherente con el mecanismo: al apartar del rol de CH a los nodos con poca batería, lo primero que se retrasa es precisamente la primera muerte.

Hay un resultado que no estaba previsto y que conviene señalar. Improved LEACH seleccionó menos cluster heads de los esperados en los escenarios de 100, 500 y 1000 nodos, entre un 27 % y un 28 % por debajo del valor teórico. La causa está en la propia Ecuación 7: el factor E_{res}/E_0 es siempre menor o igual a 1, de modo que reduce el umbral de todos los nodos y no solo el de los agotados. Conforme avanza la simulación y la energía media de la red baja, la probabilidad efectiva de selección cae por debajo de pCH. En redes pequeñas ese efecto apenas se aprecia; en redes grandes, sí.

Esa selección más escasa explica el comportamiento de la latencia. Con menos cluster heads, cada uno atiende a más miembros y las distancias dentro del clúster crecen, por lo que cada paquete tarda algo más en llegar. El aumento va del 1,86 % al 3,89 %, es decir, entre 18 y 42 ms sobre valores próximos al segundo. En una UWSN acústica, donde el retardo de propagación domina el presupuesto de tiempo, ese incremento es marginal, pero existe y conviene reportarlo.

El PDR mejora poco, entre 0,79 y 2,04 puntos porcentuales, porque partía ya de valores en torno al 94 %. Lo relevante no es la magnitud sino que la modificación no degrada la entrega mientras alarga la vida de la red.

En conjunto, la valoración sobre eficiencia energética en el contexto UWSN es favorable. En un entorno donde reemplazar una batería implica una campaña marítima, ganar entre un 44 % y un 56 % de rondas antes de la primera muerte tiene consecuencias operativas reales y no solo estadísticas.

4.6.2. Relación entre Densidad de Red y Desempeño

La ventaja de Improved LEACH no crece con el tamaño de la red. Este resultado contradice la intuición de partida, según la cual a mayor densidad habría mayor dispersión energética entre nodos y, por tanto, más margen para que la ponderación repartiera la carga.

En el FND, la diferencia absoluta a favor de Improved LEACH baja de 12,3 rondas con 50 nodos a 7,6 con 500, y repunta a 8,5 con 1000. En el LND el patrón es aún menos regular: 6,2, 8,1, 1,0, 8,1 y 7,7 rondas. El escenario de 250 nodos rompe cualquier tendencia que se quisiera trazar.

El PDR, en cambio, se comporta de forma estable: se mantiene entre el 93 % y el 96,3 % en los diez casos, sin degradarse al aumentar N en ninguno de los dos protocolos. Esto indica que la agregación en el CH absorbe bien el aumento de tráfico dentro del rango probado.

El número de cluster heads es la métrica que peor escala en Improved LEACH, por el motivo expuesto en la sección anterior. LEACH mantiene una desviación positiva relativamente constante, entre 7,06 % y 13,38 % salvo en el escenario de 50 nodos, donde alcanza el 29,59 %. Improved LEACH oscila entre +1,28 % y −28,16 % sin un patrón claro respecto a N.

La lectura conjunta es que la mejora de Improved LEACH es robusta pero no escalable en sentido fuerte: existe en todos los tamaños probados, sin amplificarse ni desaparecer. El escenario de 250 nodos, donde el LND apenas mejora, sugiere que la geometría concreta del despliegue puede pesar tanto como la densidad. Comprobarlo requeriría más de diez semillas o un barrido más fino del número de nodos.

4.6.3. Comparación con Literatura Relacionada

La comparación con trabajos previos se plantea en términos metodológicos y no numéricos. Los valores absolutos de FND, LND o PDR dependen del volumen del escenario, la energía inicial, el tamaño de paquete y el modelo de canal, de manera que contrastar cifras entre estudios con configuraciones distintas no aporta información válida.

La comparación se organiza en torno a tres ejes. El primero es si el criterio de selección de cabeza de clúster utiliza únicamente la energía residual o la combina con otros factores, como la distancia al sink, el historial de rotación o la densidad local. El segundo es si el estudio se realizó en un entorno terrestre o subacuático, dado que el canal acústico modifica por completo el peso relativo de la latencia. El tercero es si la evaluación se apoya en una sola corrida o incluye repeticiones con análisis estadístico, que es donde el presente trabajo aporta un elemento diferencial: diez semillas pareadas e intervalos de confianza sobre las cien corridas.

Tabla 29

Comparación metodológica entre estudios relacionados y la presente investigación

Estudio	*Protocolo base*	*Criterio de mejora*	*Entorno*	*N nodos*	*Área o volumen*	*Energía inicial*	*Modelo energético*	*Tendencia reportada*	*Relación con este estudio*
Heinzelman et al. (2000)	LEACH	Selección probabilística y rotativa de CH	Terrestre	100	Aproximadamente 50 × 50 m	0,5 J	Modelo de radio de primer orden: electrónica y amplificación dependiente de (d^2)	LEACH distribuye la carga de los CH y prolonga la supervivencia frente a transmisión directa, MTE y clustering estático	Constituye el protocolo base y la referencia del modelo energético utilizado en esta investigación
Nasr y Quwaider (2020)	LEACH mejorado, denominado S-LEACH	Selección de un CH secundario y consideración de la distancia en la elección de CH	Terrestre	100	100 × 100 m	5 J	Modelo energético convencional de LEACH; los coeficientes energéticos no se detallan en la tabla experimental	El último nodo murió en la ronda 4425, frente a la ronda 1934 de LEACH; se reportó una mejora de 128,80 % en la vida útil	Comparte el propósito de extender la vida de la red, pero introduce CH secundarios y distancia, mientras este estudio pondera el umbral mediante energía residual en una UWSN
Mansouri e Ioualalen (2016)	LEACH adaptado para UWSN	Energía residual en la elección de CH y sustitución del modelo terrestre por un modelo acústico subacuático	Subacuático	100	50 × 50 m, escenario 2D	5 J	Modelo acústico dependiente de distancia, frecuencia y absorción de Thorp; frecuencia de 25 kHz	Al finalizar 200 rondas permanecieron 21 nodos vivos con LEACH, mientras que en la transmisión directa todos habían muerto; se reportó una mejora aproximada del 21 % en vida útil	Coincide con el entorno subacuático, la selección basada en energía y el uso de un modelo acústico; se diferencia porque emplea MATLAB, un escenario 2D y una sola densidad
Farinango (2026)	LEACH / Improved-LEACH	Umbral ponderado mediante la relación entre energía residual y energía inicial	Subacuático	50, 100, 250, 500 y 1000	1000 × 1000 × 1000 m	1 J	Modelo energético de primer orden y canal acústico implementado en OMNeT++/UanModel	Mejora principalmente FND, HND, PDR, throughput y latencia en redes medianas y grandes	Presente estudio

Nota: CH = *Cluster Head* (cabeza de clúster); N = número de nodos. La tabla sintetiza las características metodológicas de los trabajos revisados en el Capítulo II y las contrasta con la presente investigación. Los valores de número de nodos, área o volumen, energía inicial y modelo energético se presentan según la información disponible en cada fuente; cuando un parámetro no fue especificado por los autores, se indica "no reportado". La tendencia reportada resume los principales resultados declarados en cada estudio. Elaboración propia con base en las fuentes citadas.

4.6.4. Limitaciones de los Resultados

Los resultados quedan acotados por las decisiones de modelado adoptadas en el Capítulo III. Las limitaciones que se derivan de ellas son las siguientes:

- Modelo de canal FreeSpacePathLoss: ignora absorción acústica dependiente de la frecuencia, variabilidad temporal y efectos multipath presentes en canales acústicos reales.
- Nodos estáticos: el modelo no captura efectos de movilidad ni corrientes oceánicas. En despliegues reales, la distancia CH–sink puede variar entre rondas.
- Improved LEACH basado en un único criterio: la variante implementada pondera el umbral solo por energía residual. Variantes con criterios adicionales (distancia al sink, historial de CH) podrían producir resultados distintos.
- Ausencia o simplificación de colisiones MAC: el esquema TDMA elimina colisiones en la fase steady-state. El PDR reportado no captura pérdidas por acceso simultáneo al medio.
- Un único sink: el análisis no considera topologías multi-sink, comunes en despliegues reales de UWSN de gran escala.
- Parámetros específicos del escenario: los resultados corresponden al conjunto de parámetros definidos en la Tabla 8, Cap. III. La generalización a otros valores de pCH, E_0, L_pkt o T_ronda requeriría campañas adicionales.

4.6.5. Amenazas a la Validez de los Resultados

Validez interna — Corrección de la implementación: Si el ciclo de rondas presenta errores de estado (ej. reset prematuro), el comportamiento observado podría no corresponder al algoritmo diseñado. Mitigación: verificación de Nivel II documentada en §3.6.5 del Cap. III.

Validez de construcción — Definición del PDR: Si N_sent y N_recv miden saltos distintos (§4.3.2), el PDR reportado corresponde a un tramo específico y no al desempeño extremo a extremo. Mitigación: §4.3.2 documenta la definición exacta utilizada.

Validez estadística — Tamaño de muestra: Con n = 10 semillas, el IC_{95} es relativamente amplio. Diferencias pequeñas entre protocolos pueden no resultar significativas con este tamaño de muestra. Mitigación: uso de comparación pareada (más potente que comparación de medias independientes).

Validez externa — Modelo de canal: Los resultados son válidos para FreeSpacePathLoss. La generalización a canales acústicos reales requiere modelos adicionales. Mitigación: documentado explícitamente en §4.6.4.

Conclusiones

La revisión de la literatura muestra que las Redes de Sensores Inalámbricos Subacuáticos operan bajo restricciones que no aparecen en las redes terrestres: el consumo energético, el retardo de propagación acústica, el ancho de banda reducido y la dificultad de reemplazar los baterías una vez desplegados los nodos. LEACH elige las cabezas de clúster con un criterio probabilístico que ignora cuánta energía le queda a cada sensor. Ponderar ese umbral con la energía residual, como hace Improved-LEACH, responde directamente a esa limitación y reparte la carga de comunicación de forma más equilibrada.

Ambos protocolos se implementaron en un entorno de simulación tridimensional bajo condiciones controladas y reproducibles: cinco escenarios de 50, 100, 250, 500 y 1000 nodos, un sink en la superficie, comunicación acústica, modelo de consumo energético y diez semillas emparejadas por cada combinación de escenario y protocolo. Las 100 corridas válidas se ejecutaron con los mismos parámetros en los dos casos. Lo único que cambió fue el procedimiento de selección de las cabezas de clúster.

Ponderar el umbral con la energía residual beneficia sobre todo la estabilidad y la supervivencia de la red. Improved-LEACH llevó el FND de 18,60 a 28,10 rondas (51,1 % de mejora), el HND de 41,86 a 52,96 rondas (26,5 %) y el LND de 54,16 a 60,38 rondas (11,5 %). El throughput efectivo mostró ventaja desde los escenarios medianos y alcanzó un 11,4 % con 1000 nodos. La respuesta ante cambios dinámicos quedó cubierta solo en parte: los nodos permanecieron estáticos durante las simulaciones, de modo que la única variación evaluada fue la densidad inicial de la red.

Las diez semillas emparejadas, junto con medias, desviaciones estándar e intervalos de confianza del 95 %, descartan que las diferencias observadas provengan de corridas aisladas. Improved-LEACH retrasó el FND, el HND y el LND en los cinco escenarios, con una

excepción: la diferencia del LND en el escenario de 250 nodos no resultó concluyente. La entrega de paquetes en el enlace sensor–CH también fue mayor, pero los datos registrados no bastan para reconstruir el PDR extremo a extremo hasta el sink, así que esa métrica es válida.

Improved-LEACH supera a LEACH en eficiencia energética, estabilidad, supervivencia de los nodos y throughput efectivo en redes medianas y grandes. La mejora no es uniforme en todas las métricas ni en todas las densidades, por lo que el protocolo no puede presentarse como superior en cualquier condición. Lo que aporta es tiempo: retrasa el agotamiento de los nodos y mantiene la red operando durante más rondas, logrando extender alrgar la vita útil de sistema con un cambio frecuencia con que se eligen las cabezas de clúster.

Recomendaciones

Conviene ampliar el análisis bibliográfico con protocolos de enrutamiento diseñados específicamente para ambientes subacuáticos y con otras variantes de LEACH que usen criterios como distancia, profundidad, densidad local y energía residual. La comparación situaría a la variante implementada frente a alternativas más recientes y especializadas en UWSN, con sus ventajas y sus límites definidos con mayor precisión.

El entorno de simulación necesita un modelo acústico más completo: absorción dependiente de la frecuencia, ruido ambiental, propagación multitrayectoria, variaciones de temperatura y salinidad, y movilidad tridimensional causada por corrientes acuáticas. Solo con esos elementos se puede saber si las mejoras obtenidas se sostienen fuera de condiciones controladas.

Hacen falta campañas experimentales que midan de forma explícita la adaptación del protocolo ante cambios dinámicos: movimiento de nodos, pérdida repentina de sensores, cambio en la posición del sink, uso de múltiples sinks, variaciones de tráfico y degradación de la calidad del canal. Conviene además un barrido más fino alrededor del escenario de 250 nodos, donde el incremento del LND fue pequeño.

La instrumentación del simulador debe conservar, en cada paquete, un identificador único, el nodo de origen, el tiempo de generación y el número de reportes agregados. Con esos datos se pueden calcular por separado el PDR sensor–CH, el PDR CH–sink y el PDR extremo a extremo, y comprobar que la latencia registrada cubra el trayecto completo, desde que se genera el reporte hasta que llega al sink.

Antes de proponer Improved-LEACH para un despliegue real conviene validarlo en una segunda plataforma de simulación o en un prototipo experimental a pequeña escala. Esa validación debería sumar métricas como energía consumida por paquete entregado, equidad

energética, tamaño de los clústeres, distancia promedio sensor–CH y CH–sink, sobrecarga de control y pérdida de paquetes por salto. Es la forma de descartar que las tendencias observadas dependan de esta implementación y de los parámetros elegidos en el estudio.

Bibliografía

Abushiba, W., Johnson, P., Alharthi, S., & Wright, C. (2018). *An energy efficient and adaptive clustering for wireless sensor network (CH-leach) using leach protocol.* Cairo: IEEE.

Akyildiz, I. F., Pompili, D., & Melodia, T. (2020). *Underwater Acoustic Sensor Networks: Research Challenges.* Estados Unidos: Ad Hoc Networks. doi:10.1016/j.adhoc.2019.101935

Bhih, A., Abushiba, W., & Elashheb, A. (2023). *An Improved Leach Algorithm Based on Hierarchical Clustering Approach for Wireless Sensor Network Application.* China: IEEE.

Birajdar, D. M., & Solapure, S. S. (2017). *LEACH: An energy efficient routing protocol using Omnet++ for Wireless Sensor Network.* Coimbatore, India: IEEE.

C, L. B., Sukumaran, R., & C, A. S. (2024). *Node Deployment Strategies and Challenges in Underwater Wireless Sensor Network.* Lalitpur, Nepal: IEEE.

Cárdenas, J. (2018). *Quantitative Forschung.* (F. U. BERLIN, Ed., & I. cuantitativa, Trans.) Berlin, Alemania. doi:http://dx.doi.org/10.17169/refubium-216

Chafi, S. A., Abbad, H., Benhaoua, M. K., & Benyamina, A. E. (2023). *Improvement of LEACH Protocol: W-LEACH Approach.* Saudi Arabia: IEEE.

Córdova, J. D. (2018). *Diseño de un sistema de monitoreo y optimización de la recolección de desechos sólidos para el campus de la espol.* Guayaquil: ESPOL.

Guo, X. S. (2016). *Enhanced LEACH protocol for underwater wireless sensor networks.* International Journal of Distributed Sensor Networks.

Haque, M. N. (2022). *Analysis of a Multi-directional Data Access Control System for Underwater Wireless Network using UAN MAC Protocols.* China: IEEE.

I.F. Akyildiz, W. S. (2002). *Wireless sensor networks: a survey* (Vol. 38). Retrieved from https://doi.org/10.1016/S1389-1286(01)00302-4.

Jawad, H. M., Nordin, R., Gharghan, S. K., Jawad, A. M., & Ismail, M. (2017). *Energy-Efficient Wireless Sensor Networks for Precision Agriculture: A Review.* Malaysia: Department of Computer Communication Engineering.

Kannappan, A., & Bommi, R. (2022). *An Improved Energy Efficient Chimp Optimization Routing Algorithm for Underwater Acoustic Sensor Networks.* Chennai, India: IEEE. doi: 10.1109/ICDSAAI55433.2022.10028871

Kim, Y.-M. (2015). *Subspace-based fault detection - Multiplicative and additive fault.* Big Sky, MT, USA: IEEE. doi:10.1109/AERO.2015.7119184

Korpeoglu, I., & Tan, H. O. (2015). *Power efficient data gathering and aggregation in wireless sensor networks.* doi:10.1145/959060.959072

Krishnakumar, A. (2016). *mproved LEACH: A modified LEACH for Wireless Sensor Network.* India: IEEE.

Kumar, V., Jain, S., & Tiwari, S. (2019). *Energy-Efficient Clustering Algorithms in Wireless Sensor Networks: A Survey. Sensors.* IEEE.

Liu, X., & Zhou, F. (2023). *Energy-Efficient Routing Protocol Based on Data Dissemination for Underwater Wireless Sensor Network.* (IEEE, Ed., & P. d. submarinos, Trans.) Limerick, Irlanda. doi:10.1109/OCEANSLimerick52467.2023.10244378

Majitha, A., Christina, C. E., Gifta, I. S., & Archana, M. (2022). *Intelligent Hybrid Low Energy Adaptive Cluster Head Selection Algorithm for Effective Data Communication in*

Underwater Sensor Network. (A. i. submarinos, Trans.) Salem, India: IEEE. doi:10.1109/ICAAIC53929.2022.9792912

Naciones Unidas. (2022). *Objetivos del desarrollo sostenible*. Retrieved from https://www.un.org/sustainabledevelopment/es/objetivos-de-desarrollo-sostenible/

Nasr, S., & Quwaider, M. (2020). *Mejora del protocolo LEACH para aumentar la vida útil de WSN.* (IEEE, Ed.) Irbid, Jordania: IEEE. doi:10.1109/ICICS49469.2020.239542

Pachouri, V., Kathuria, S., Gehlot, A., Negi, P., Chaudhary, M., & Chhabra, G. (2023). *UWSN with Technological Intervention: Benefits and Challenges.* (U. c. Desafíos, Trans.) doi:10.1109/ICSCDS56580.2023.10104686

Pandith, A. M., & Goyal, N. (2022). *Emerging Data Aggregation State-of-Art Techniques with Comparative Analysis in UWSN.* (T. e. UWSN, Trans.) Greater Noida: India. doi:10.1109/ICACITE53722.2022.9823478

Peng, H., Si, S., Shen, X. S., & Zhao, H. (2015). *Energy-efficient and scalable clustering scheme for wireless sensor networks.* China: IEEE. doi:10.1109/WCSP.2015.7341304

Secretaría Nacional de Planificación. (2021). *Plan de Creación de Oportunidades 2021-2025.* Quito: Ecuador.

Singh, P. K., & Kaur, J. (2022). *Simulation Investigation of State-of-Art Cluster-Based Routing Protocols in WSN.* Tashkent, Uzbekistan: IEEE.

Stewart, C., Fough, N., Erdogan, N., & Prabhu, R. (2023). *Performance and Energy Modelling for a Low Energy Acoustic Network for the Underwater Internet of Things.* La Valletta, Malta. doi: 10.1109/MetroSea58055.2023.10317266

Tatiana A. Fedorova, V. A. (2022). *Optimization of an Underwater Wireless Sensor Network Architecture with Wave Glider as a Mobile Gateway.*

Tümer, A. E., & Gündüz, M. (2014). *An improved leach protocol for indoor wireless sensor networks.* India: IEEE.

Velasco, S. Á. (2023). *Diseño de una aplicación académica para el estudio de redes de sensores inalámbricos subacuáticos en Matlab.* Ibarra: UTN. Retrieved from http://repositorio.utn.edu.ec/handle/123456789/13402

XingGuo, L., JunFeng, W., & LinLin, B. (2016). *LEACH Protocol and Its Improved Algorithm in Wireless Sensor Network.* Chengdu: IEEE.

Yang, T., Guo, Y., Dong, J., & Xia, M. (2018). *Wireless Routing Clustering Protocol Based on Improved LEACH Algorithm.* Macao: IEEE.

Yıldırım, G., & Tártaro, Y. (2018). *Simplified Agent-Based Resource Sharing Approach for WSN-WSN Interaction in IoT/CPS Projects* (Vol. 6). (E. s.-W. IoT/CPS, Trans.) IEEE. doi:10.1109/ACCESS.2018.2884741

Zhang, W., & Zhao, L. (2015). *Analysis and research of improved LEACH routing protocol for pressure sensor based on WSN.* Beijing: IET. doi:10.1049/cp.2014.1332

Akyildiz, I. F., Pompili, D., & Melodia, T. (2005). Underwater acoustic sensor networks: Research challenges. Ad Hoc Networks, 3(3), 257–279. https://doi.org/10.1016/j.adhoc.2005.01.004

Ariza Quintana, A. (s.f.). UanModel: Underwater Acoustic Network Model for OMNeT++/INET. Universidad de Málaga. [Framework de simulación, repositorio privado].

Heinzelman, W. R., Chandrakasan, A., & Balakrishnan, H. (2000). Energy-efficient communication protocol for wireless microsensor networks. En Proceedings of the 33rd Annual Hawaii International Conference on System Sciences (HICSS-33) (pp. 1–10). IEEE. https://doi.org/10.1109/HICSS.2000.926982

ANEXOS

ANEXO A. INSTALACIÓN DE UBUNTU 22.04.5 LTS

Documentar el procedimiento seguido para instalar Ubuntu 22.04.5 LTS (Jammy Jellyfish), edición Desktop, como sistema operativo base del entorno de trabajo utilizado para la implementación, configuración y ejecución de las herramientas de simulación y procesamiento de resultados de la investigación.

Anexo 1

Configuración seleccionada

Parámetro	Selección realizada
Sistema operativo	Ubuntu 22.04.5 LTS (Jammy Jellyfish), Desktop Image
Idioma del instalador	Español
Distribución del teclado	English (US)
Tipo de instalación	Instalación mínima
Actualizaciones durante la instalación	Habilitadas
Software de terceros	No seleccionado durante esta etapa
Método de instalación	Borrar disco e instalar Ubuntu
Sistema de archivos principal	ext4
Partición de arranque	EFI System Partition (ESP)
Zona horaria	Guayaquil, Ecuador
Nombre del equipo	david
Nombre de usuario	david
Inicio de sesión	Solicitud de contraseña

Nota: las credenciales de acceso se mantuvieron protegidas; en las evidencias únicamente se visualiza la configuración general del usuario y no el contenido de la contraseña.

Procedimiento de instalación

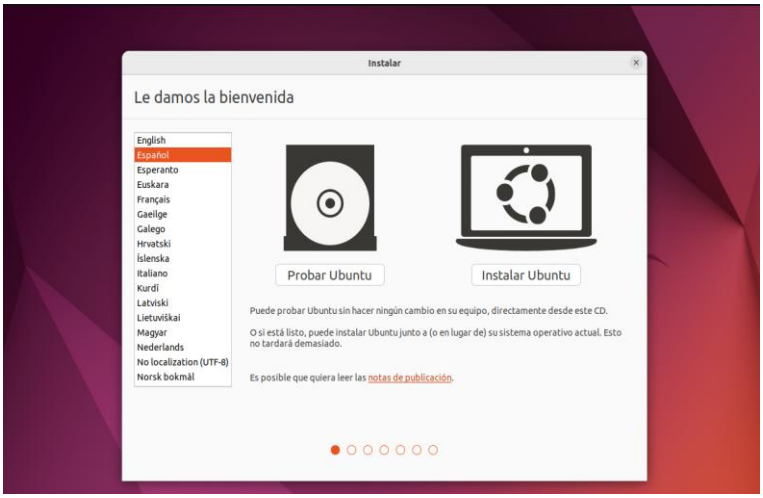
El procedimiento se ejecutó mediante el instalador gráfico incluido en la imagen de escritorio de Ubuntu. A continuación, se presentan las etapas registradas durante la instalación.

Inicio del instalador y selección del idioma

Al iniciar el medio de instalación se presentó la pantalla de bienvenida. Se seleccionó el idioma Español y posteriormente la opción Instalar Ubuntu, con el propósito de realizar una instalación permanente en el equipo.

Anexo 2

Pantalla de bienvenida y selección del idioma español.

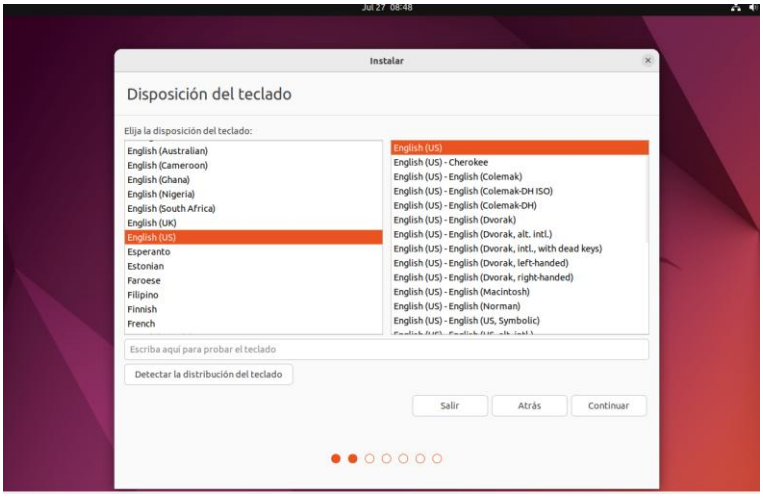


Selección de la distribución del teclado

En la configuración del teclado se seleccionó la distribución English (US). Esta opción se mantuvo de acuerdo con la disposición física del teclado utilizado en el equipo de trabajo. El instalador permite comprobar el funcionamiento de las teclas antes de continuar.

Anexo 3

Selección de la distribución de teclado English (US).

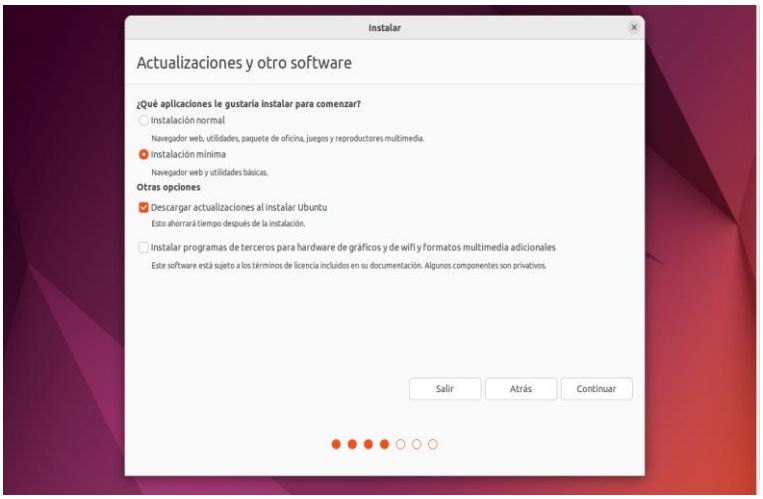


Actualizaciones y software inicial

Se eligió la instalación mínima, la cual incorpora el navegador web y las utilidades básicas del sistema. También se habilitó la descarga de actualizaciones durante la instalación. La opción de programas de terceros para gráficos, Wi-Fi y formatos multimedia no fue seleccionada en esta etapa, debido a que dichos componentes pueden instalarse posteriormente conforme a los requerimientos del hardware.

Anexo 4

Configuración de actualizaciones y selección de instalación mínima.

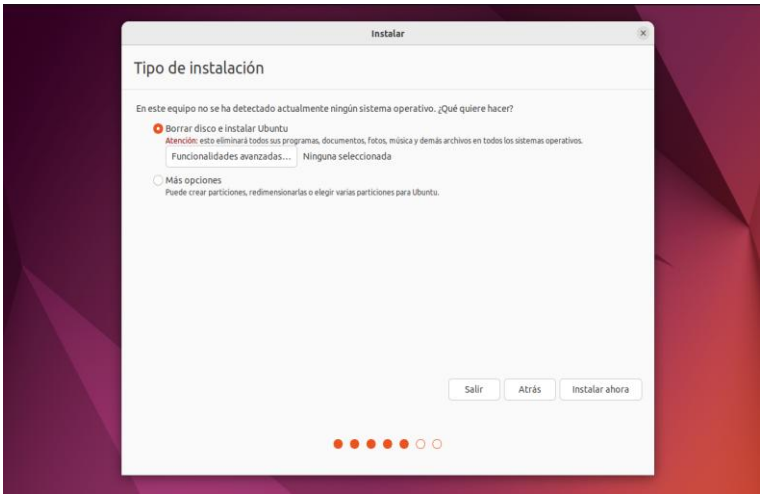


Definición del tipo de instalación

El instalador no detectó otro sistema operativo en el disco. Por esta razón, se seleccionó la opción Borrar disco e instalar Ubuntu. Esta alternativa asigna el dispositivo de almacenamiento exclusivamente al sistema Ubuntu y elimina cualquier información previa existente en el disco.

Anexo 5

Selección de la opción para borrar el disco e instalar Ubuntu.

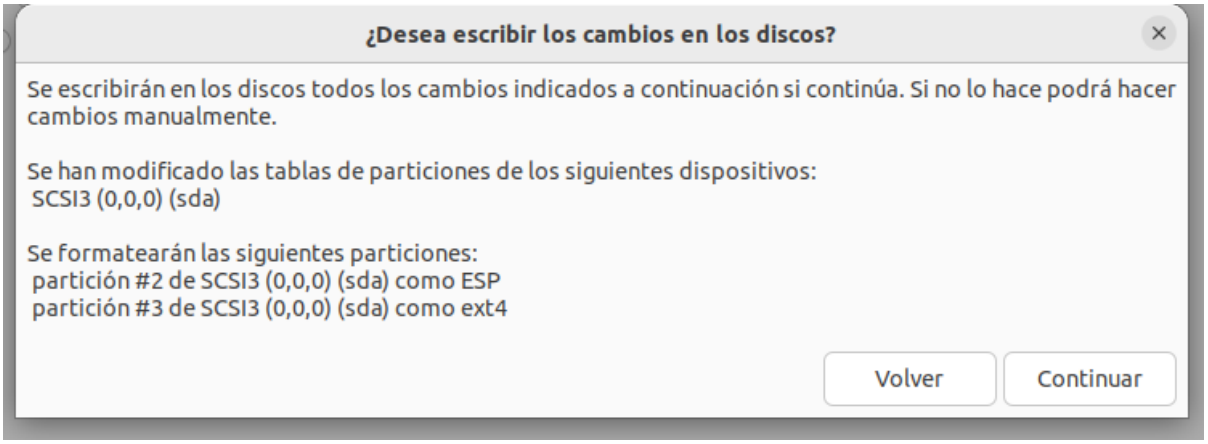


Confirmación de cambios en el disco

Antes de iniciar la escritura de datos, el instalador mostró el resumen de cambios en las tablas de particiones. Se confirmó la creación o formateo de una partición de sistema EFI (ESP), utilizada para el arranque UEFI, y de una partición principal con sistema de archivos ext4 para almacenar el sistema operativo y sus archivos.

Anexo 6

Confirmación de las particiones ESP y ext4 en el disco de instalación.

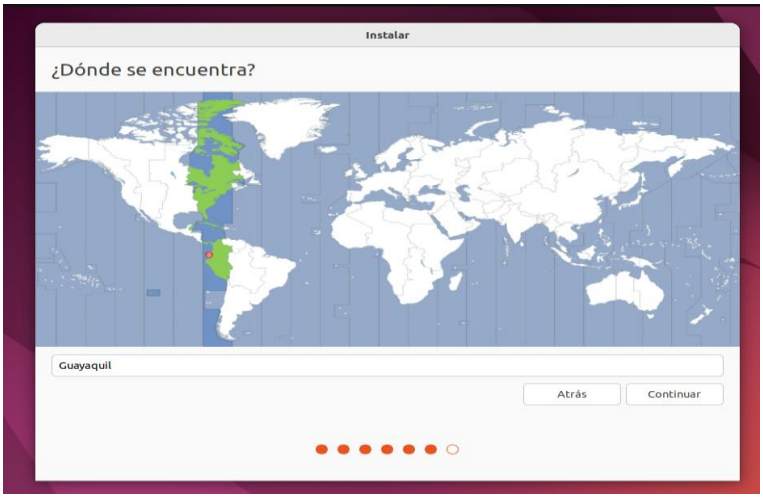


Configuración de la zona horaria

Se estableció la ubicación Guayaquil, correspondiente a la zona horaria utilizada en Ecuador continental. Esta configuración permite mantener la fecha y hora del sistema de manera coherente con el entorno donde se ejecutaron las actividades de la investigación.

Anexo 7

Selección de la zona horaria Guayaquil, Ecuador.

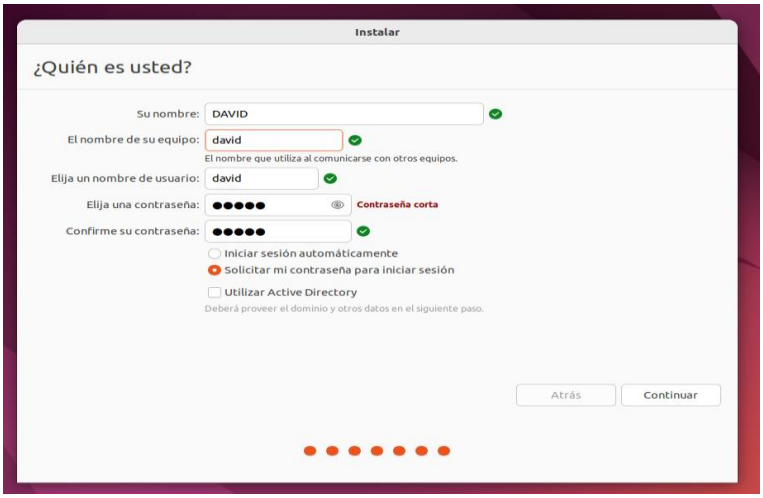


Creación del usuario del sistema

Se registró el nombre DAVID, el nombre del equipo david y el nombre de usuario david. Como medida de seguridad, se configuró el sistema para solicitar la contraseña al iniciar sesión. No se habilitó la integración con Active Directory.

Anexo 8

Creación del usuario local y configuración del inicio de sesión.



Ejecución de la instalación

Después de confirmar los parámetros, el instalador copió los archivos, configuró los paquetes y realizó la detección del hardware. Durante este proceso se mostró una presentación informativa de las aplicaciones y capacidades disponibles en Ubuntu.

Anexo 9

Proceso de copia de archivos y configuración del hardware.

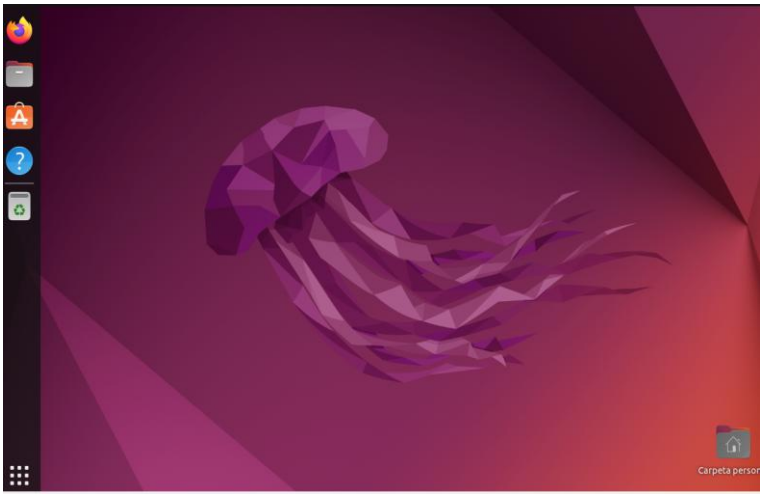


Inicio y validación del entorno gráfico

Finalizada la instalación y efectuado el reinicio del equipo, se cargó correctamente el escritorio GNOME de Ubuntu. La visualización del panel lateral, el explorador de archivos, el navegador y los accesos del sistema evidenció que el sistema operativo inició de forma satisfactoria y quedó disponible para la instalación de las herramientas requeridas por la tesis.

Anexo 10

Escritorio de Ubuntu después de completar la instalación.



Verificación posterior a la instalación

Una vez iniciado el sistema, la versión instalada puede verificarse desde la terminal mediante los comandos que se indican a continuación. Estos comandos permiten comprobar la distribución, el nombre en clave, la versión del núcleo y la identificación general del equipo.

```
lsb_release -a

hostnamectl

uname -r
```

Como actividad complementaria, se recomienda ejecutar la actualización inicial del sistema mediante sudo apt update y sudo apt upgrade, instalar los controladores específicos del

hardware cuando sean necesarios y conservar un registro de las versiones de las herramientas empleadas en la investigación.

ANEXO B. INSTALACION DE OMNET++ 6.03, EN UBUNTU

Anexo 11

Información general del entorno de instalación de OMNeT++ 6.0.3

Software instalado	OMNeT++ 6.0.3 para Linux x86_64
Sistema operativo	Ubuntu 22.04.5 LTS (Jammy Jellyfish), 64 bits
Directorio de instalación	/home/david/Descargas/omnetpp-6.0.3
Fecha del informe	27 de julio de 2026

Nota. Se detallan la versión del software instalado, el sistema operativo utilizado, el directorio de instalación y la fecha de elaboración del informe. Elaboración propia.

Documentar el procedimiento seguido para descargar, configurar, compilar y verificar el funcionamiento de OMNeT++ 6.0.3 en Ubuntu 22.04.5 LTS. El registro se presenta como evidencia técnica del entorno de simulación empleado en el desarrollo experimental de la tesis.

Entorno de instalación y dependencias

La instalación se efectuó sobre una distribución Ubuntu de 64 bits. OMNeT++ se compiló desde el paquete oficial para Linux, por lo que fue necesario disponer de un compilador C/C++, herramientas de construcción, bibliotecas de interfaz gráfica Qt y componentes auxiliares para Python.

Anexo 12

Componentes y dependencias utilizados para la instalación de OMNeT++ 6.0.3

Componente	Versión o paquete	Finalidad
Sistema operativo	Ubuntu 22.04.5 LTS, 64 bits	Plataforma base de instalación.
Simulador	OMNeT++ 6.0.3	Entorno de simulación de eventos discretos.
Compilación	GCC/G++ y GNU Make	Construcción de bibliotecas y ejecutables.
Analizadores léxicos	Bison y Flex	Generación y procesamiento de archivos requeridos durante la compilación.
Interfaz gráfica	Qt 5: qtbase5-dev, qtchooser, qt5-qmake y qtbase5-dev-tools	Construcción y ejecución del IDE Qtenv.
Python	NumPy, SciPy, pandas, Matplotlib y posix-ipc	Soporte de análisis, procesamiento y herramientas auxiliares.
Integración	SWIG	Generación de interfaces entre lenguajes.

Nota. Se describen los componentes de software, sus versiones o paquetes instalados y la función que cumplen durante la configuración, compilación y ejecución de OMNeT++ 6.0.3. Elaboración propia.

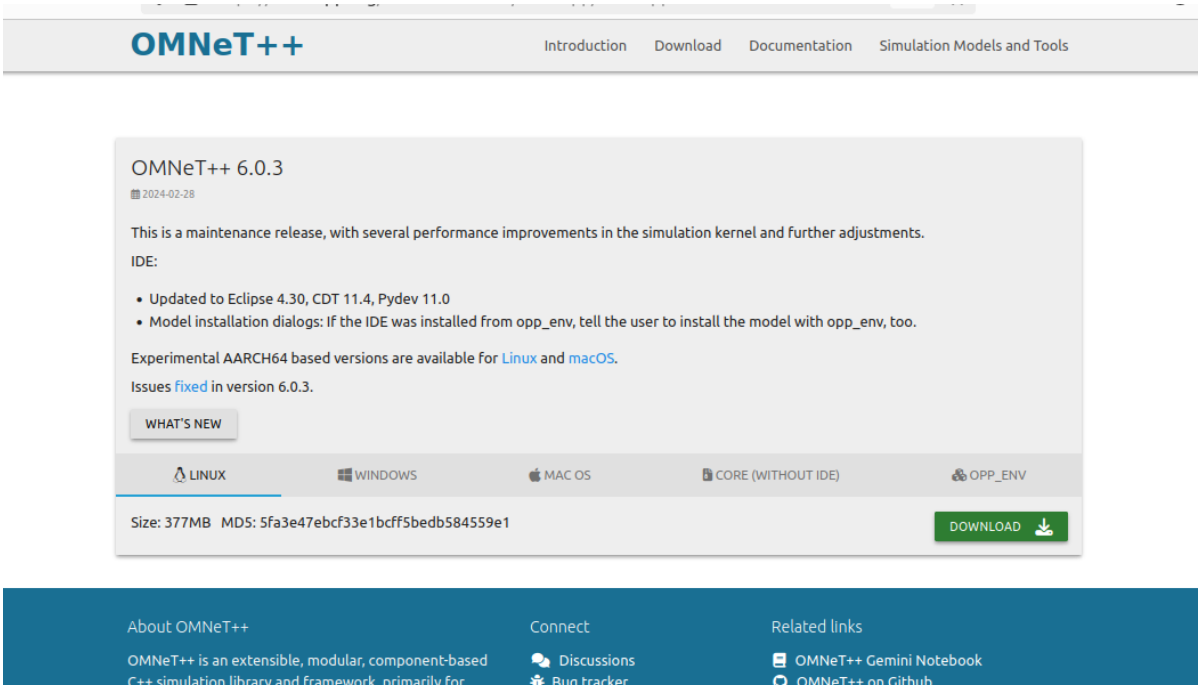
Procedimiento de instalación

Descarga del paquete de instalación

Se accedió al sitio oficial de OMNeT++ y se seleccionó la versión 6.0.3 para Linux. El archivo descargado correspondió al paquete comprimido omnetpp-6.0.3-linux-x86_64.tgz, con un tamaño aproximado de 377 MB.

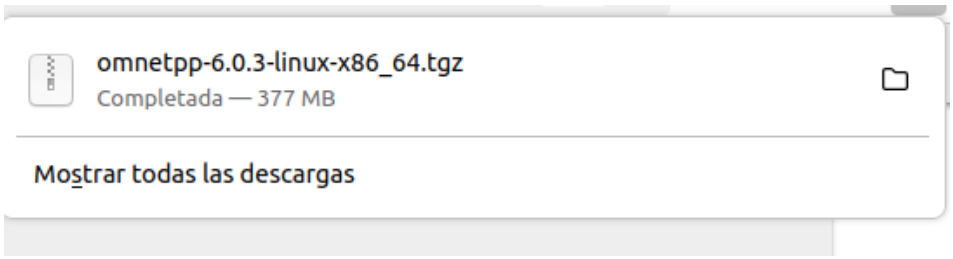
Anexo 13

Página oficial de descarga de OMNeT++ 6.0.3 para Linux.



Anexo 14

Confirmación de la descarga del archivo comprimido omnetpp-6.0.3-linux-x86_64.tgz.



Extracción y carga del entorno

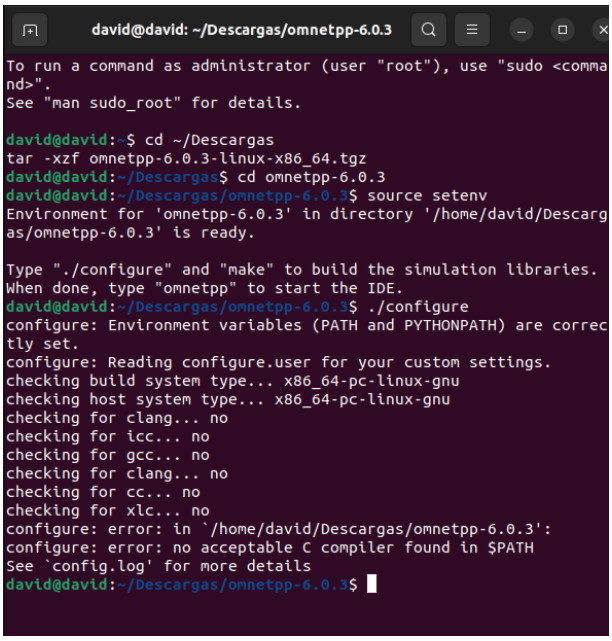
Una vez finalizada la descarga, el paquete se extrajo en el directorio Descargas. Posteriormente, se ingresó a la carpeta de OMNeT++ y se ejecutó el script setenv, encargado de incorporar temporalmente al PATH las rutas de los ejecutables y herramientas del simulador.

```
cd ~/Descargas
tar -xzf omnetpp-6.0.3-linux-x86_64.tgz
cd omnetpp-6.0.3
source setenv
./configure
```

Durante la primera ejecución de ./configure se detectó el mensaje “no acceptable C compiler found in $PATH”. Este resultado indicó que el sistema no disponía todavía de todos los componentes de compilación necesarios.

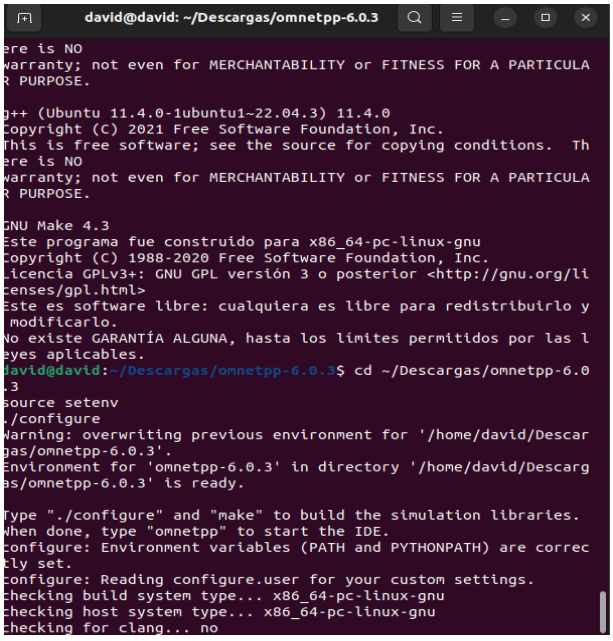
Anexo 15

Descompresión del paquete, carga del entorno y primera ejecución del script de configuración.



Anexo 16

Comprobación de las herramientas de construcción y repetición de la configuración antes de completar las dependencias.



Corrección de la ausencia del compilador

Para corregir el error se actualizaron los repositorios de Ubuntu y se instalaron las herramientas de desarrollo. El paquete build-essential proporciona GCC, G++, GNU Make y archivos básicos para la construcción de software desde código fuente.

```
sudo apt update
sudo apt install -y build-essential
```

Después de la instalación se verificó la disponibilidad de las herramientas mediante los comandos g++ --version y make --version. En la evidencia se observa G++ 11.4.0 y GNU Make 4.3, versiones compatibles con el entorno utilizado.

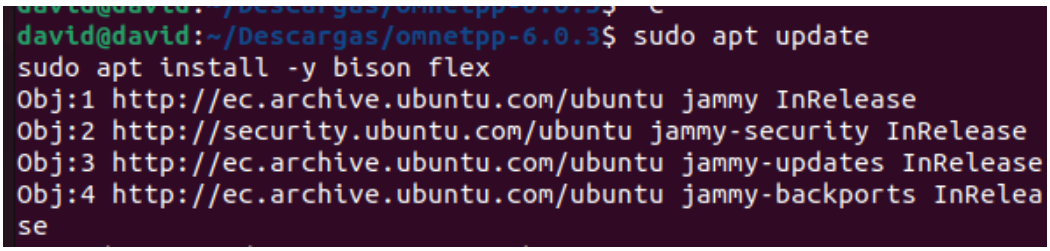
Instalación de dependencias auxiliares

A continuación, se instalaron Bison y Flex, necesarios para el procesamiento de definiciones y analizadores utilizados en la construcción del simulador.

```
sudo apt update
sudo apt install -y bison flex
```

Anexo 17

Actualización de repositorios e instalación de Bison y Flex.

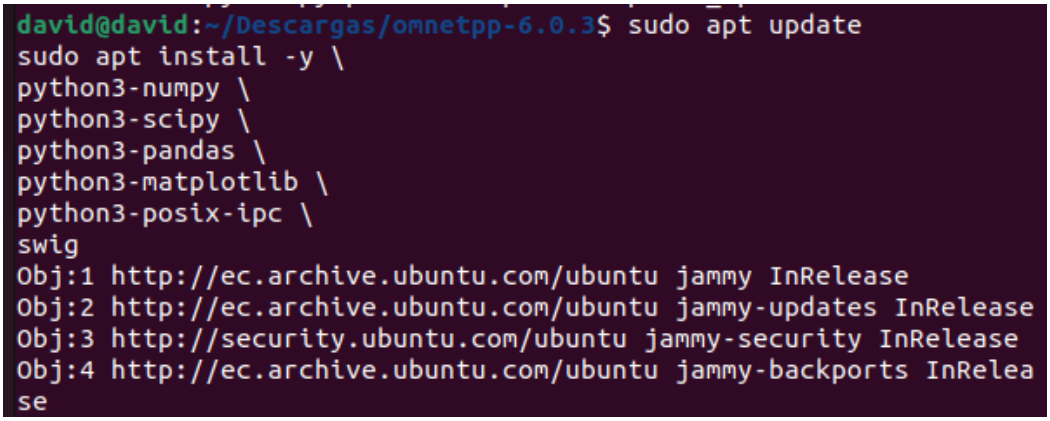


También se incorporaron los paquetes de Python empleados por las herramientas auxiliares de OMNeT++, junto con SWIG para la generación de interfaces.

```
sudo apt install -y \
  python3-numpy \
  python3-scipy \
  python3-pandas \
  python3-matplotlib \
  python3-posix-ipc \
  swig
```

Anexo 18

Instalación de paquetes de Python requeridos y de la herramienta SWIG.



Para habilitar la compilación del IDE gráfico se instalaron los paquetes de desarrollo correspondientes a Qt 5.

```
sudo apt install -y qtbase5-dev qtchooser qt5-qmake qtbase5-dev-tools
```

Anexo 19

Instalación de los componentes de desarrollo de Qt 5.



Configuración de OMNeT++

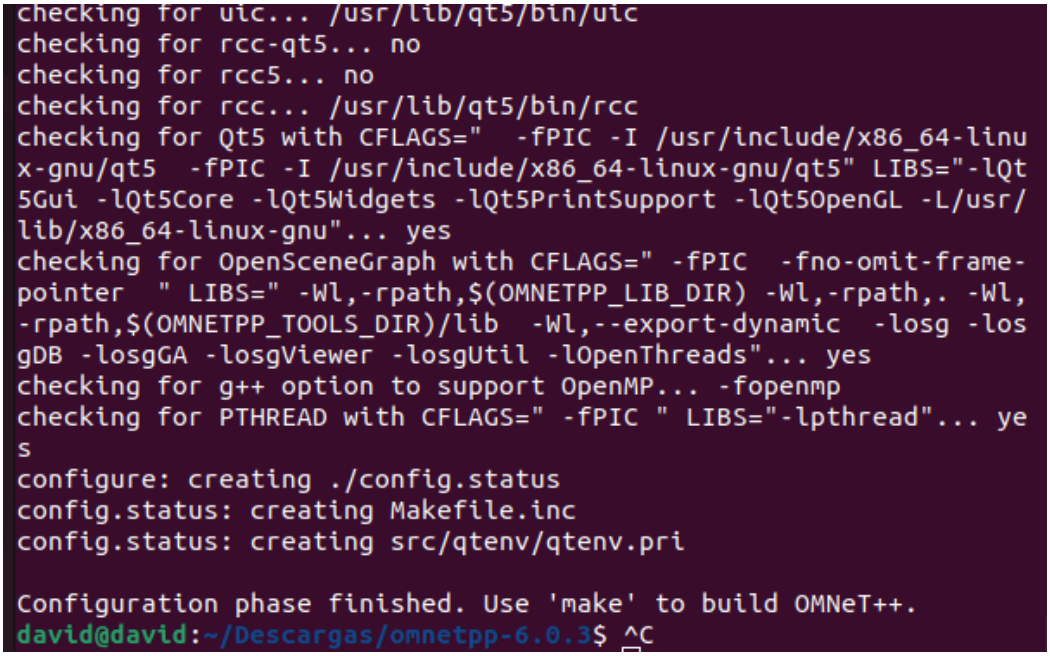
Con las dependencias instaladas, se volvió a cargar el entorno y se ejecutó nuevamente el script de configuración. El proceso comprobó el compilador, Qt, OpenSceneGraph, bibliotecas gráficas y soporte para hilos. La aparición del mensaje “Configuration phase

finished. Use make to build OMNeT++” confirmó que la preparación del código fuente concluyó correctamente.

```
cd ~/Descargas/omnetpp-6.0.3
source setenv
./configure
```

Anexo 20

Finalización satisfactoria de la fase de configuración de OMNeT++.



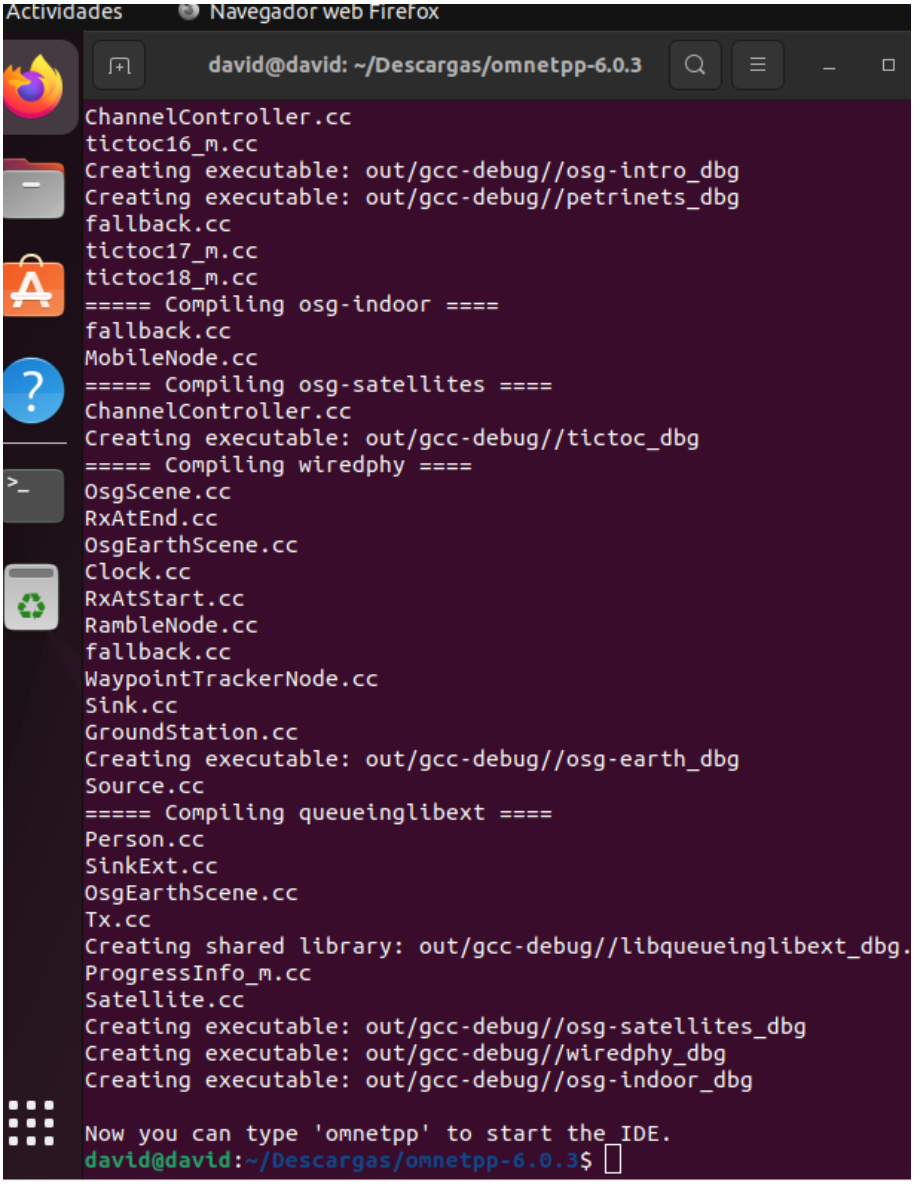
Compilación del simulador

La construcción del simulador se realizó con el comando make. Durante esta etapa se compilaron las bibliotecas principales, el IDE, los entornos de ejecución y los ejemplos incluidos. El proceso finalizó sin errores y mostró el mensaje “Now you can type omnetpp to start the IDE”.

```
make
```

Anexo 21

Compilación de las bibliotecas, ejecutables y ejemplos incluidos en OMNeT++.



Inicio del IDE y selección del espacio de trabajo

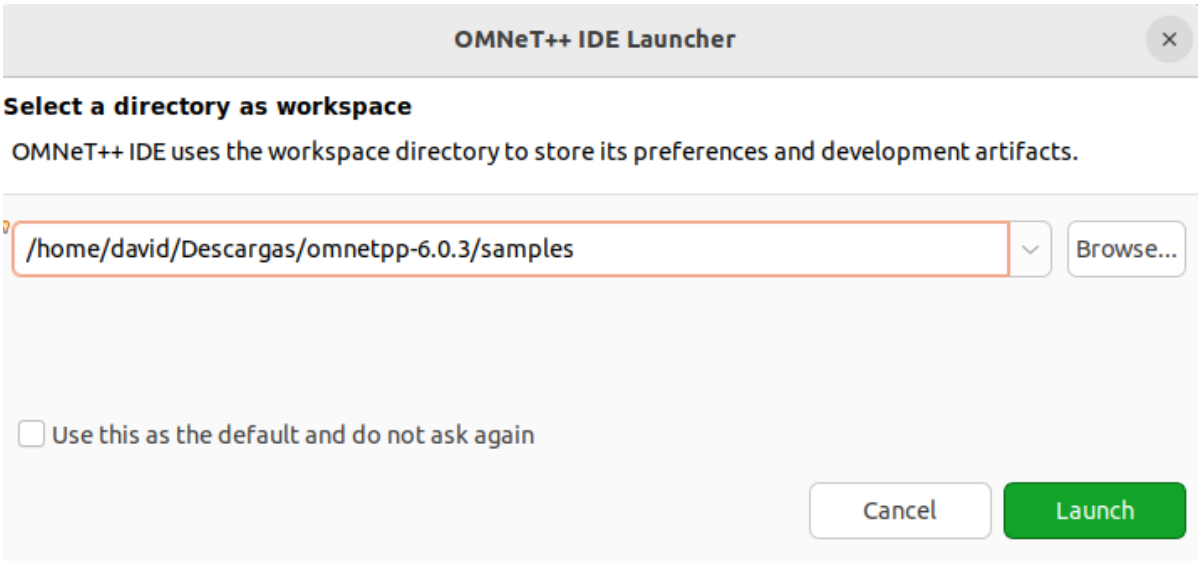
Después de completar la compilación, el IDE se inició desde la misma terminal, manteniendo cargadas las variables establecidas por setenv.

```
source setenv
omnetpp
```

En el primer inicio se presentó el selector de espacio de trabajo. En la evidencia se utilizó la ruta /home/david/Descargas/omnetpp-6.0.3/samples como directorio inicial. Para el desarrollo de la tesis es conveniente utilizar un espacio de trabajo independiente, con el fin de separar los proyectos propios de los archivos de instalación.

Anexo 22

Selección del directorio de trabajo inicial para el IDE.

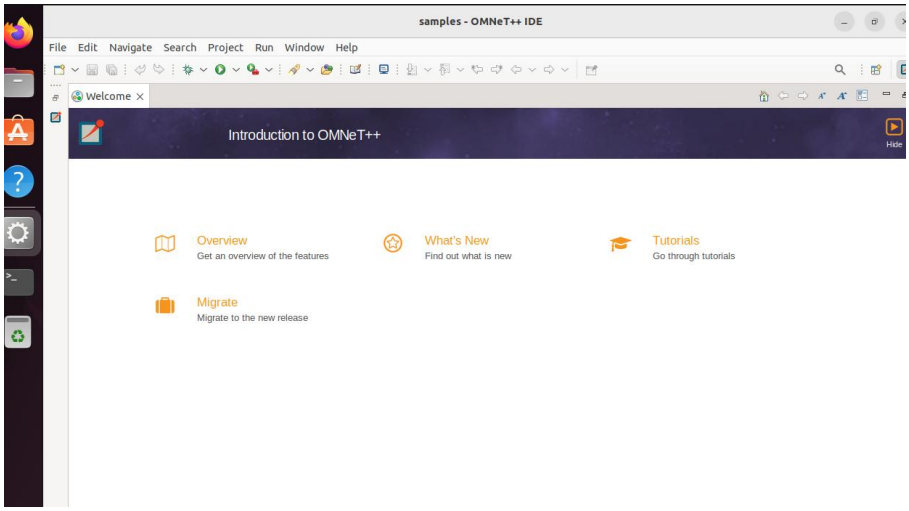


Verificación del entorno gráfico

El IDE se abrió correctamente y presentó la pantalla de bienvenida de OMNeT++. Esto confirmó la disponibilidad del entorno gráfico, los menús de desarrollo, las herramientas de compilación y las vistas de administración de proyectos.

Anexo 23

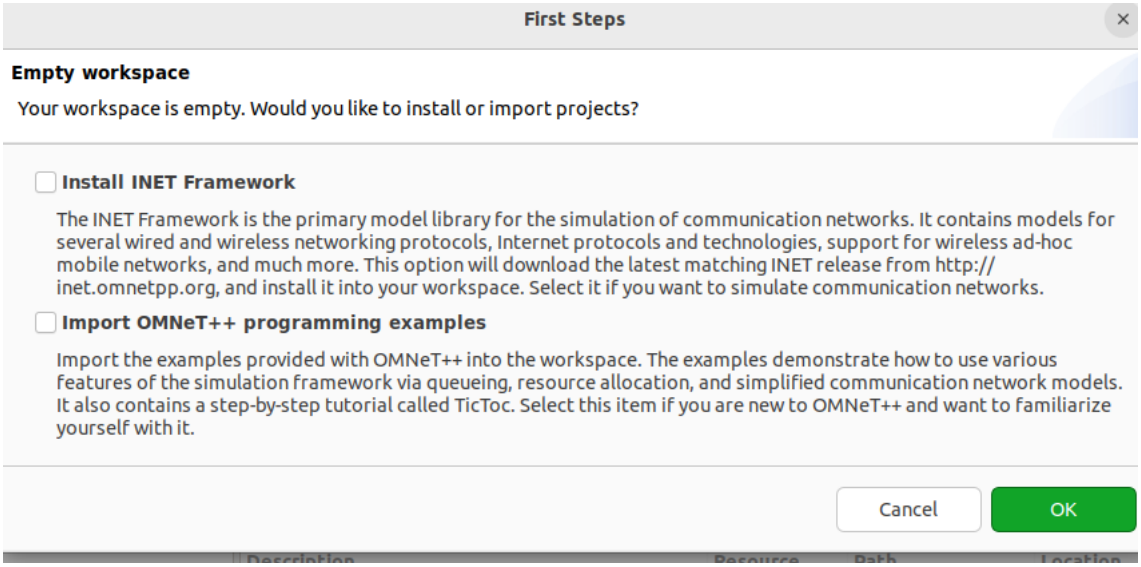
Pantalla principal de bienvenida del entorno de desarrollo OMNeT++.



Al detectar un espacio de trabajo vacío, el asistente inicial ofreció dos opciones: instalar INET Framework o importar los ejemplos de programación de OMNeT++. Estas acciones son opcionales y pueden ejecutarse posteriormente de acuerdo con las necesidades del proyecto.

Anexo 24

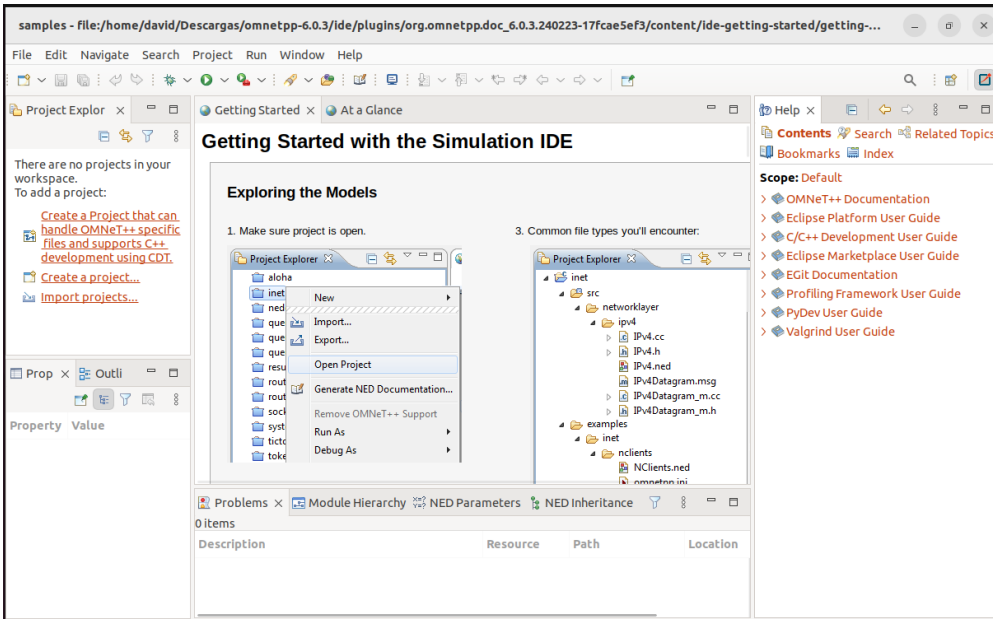
Opciones iniciales para instalar INET Framework o importar ejemplos de programación.



Finalmente, se accedió a la vista Getting Started with the Simulation IDE. La presencia del explorador de proyectos, las herramientas C/C++, la ayuda integrada y los paneles de propiedades confirmó que la instalación de OMNeT++ 6.0.3 quedó operativa.

Anexo 25

Verificación final del IDE de OMNeT++ en funcionamiento.



Validación de la instalación

La instalación se consideró satisfactoria al cumplirse los siguientes criterios: el script ./configure terminó sin errores; el comando make generó las bibliotecas y ejecutables; el comando omnetpp abrió el IDE; y el entorno mostró correctamente las vistas de desarrollo y la documentación integrada.

Anexo 26

Verificación de la instalación y funcionamiento de OMNeT++ 6.0.3

Criterio	Evidencia observada	Estado
Configuración	Mensaje de finalización de la fase configure.	Cumplido
Compilación	Creación de bibliotecas y ejecutables sin error final.	Cumplido
Ejecución	Apertura del OMNeT++ IDE Launcher y del IDE.	Cumplido
Interfaz	Visualización de menús, vistas y documentación.	Cumplido

Nota. Se presentan los criterios empleados para comprobar la correcta configuración, compilación, ejecución y visualización de la interfaz gráfica de OMNeT++ 6.0.3. Elaboración propia.

ANEXO C. REPOSITORIO DIGITAL DEL PROYECTO

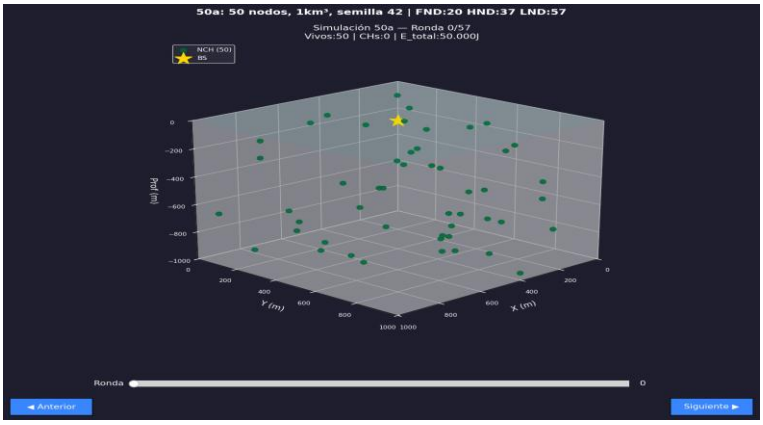
- Enlace de acceso

https://github.com/IOTSOFTEC/leach-vs-improved-leach-uwsn.git

ANEXO D. EVIDENCIAS COMPLEMENTARIAS DE EJECUCIÓN

Anexo 27

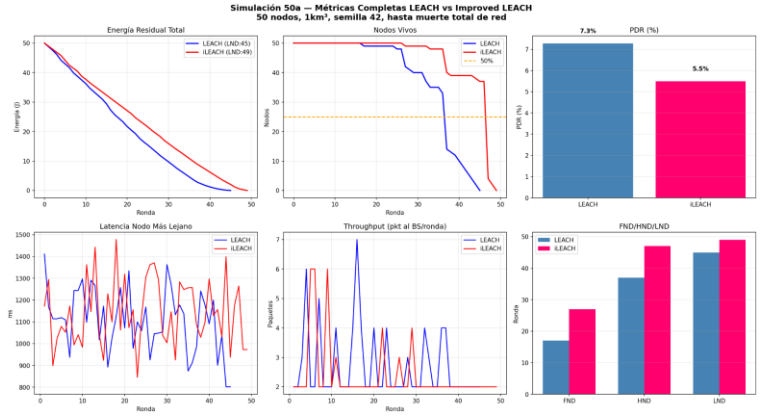
Distribución tridimensional inicial del escenario de 50 nodos, simulación A.



Nota. Captura de la ronda inicial del escenario de 50 nodos. La imagen muestra la ubicación espacial de los nodos y de la estación base antes del consumo energético de la red. Elaboración propia.

Anexo 28

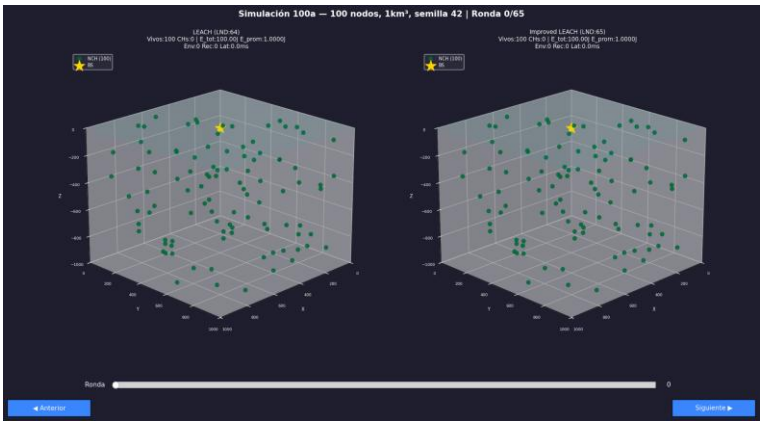
Métricas completas de LEACH e Improved-LEACH para 50 nodos, simulación A.



Nota. Se presentan la energía residual total, los nodos vivos por ronda, el PDR, la latencia del nodo más lejano, el throughput y los indicadores FND, HND y LND. Elaboración propia.

Anexo 29

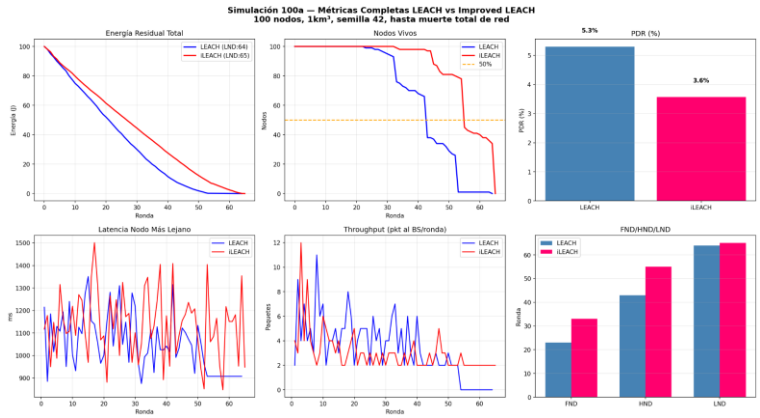
Comparación tridimensional inicial de LEACH e Improved-LEACH para 100 nodos, simulación A.



Nota. Ambos protocolos utilizan la misma distribución inicial, permitiendo una comparación bajo condiciones equivalentes. Elaboración propia.

Anexo 30

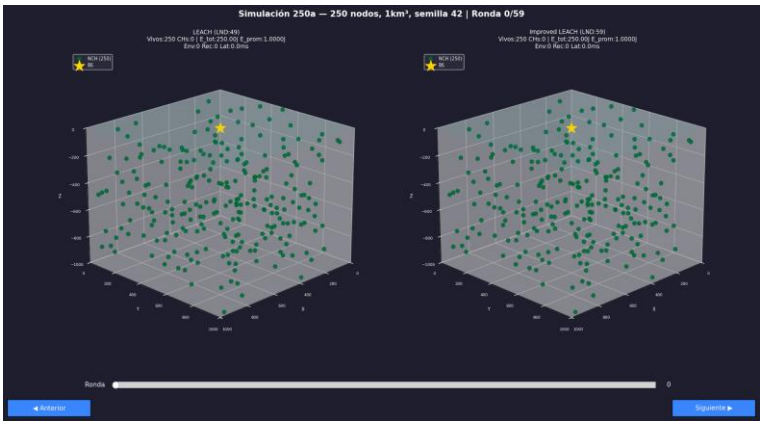
Métricas completas de LEACH e Improved-LEACH para 100 nodos, simulación A.



Nota. La figura integra las principales métricas energéticas, de vida útil y de comunicación obtenidas durante la ejecución. Elaboración propia.

Anexo 31

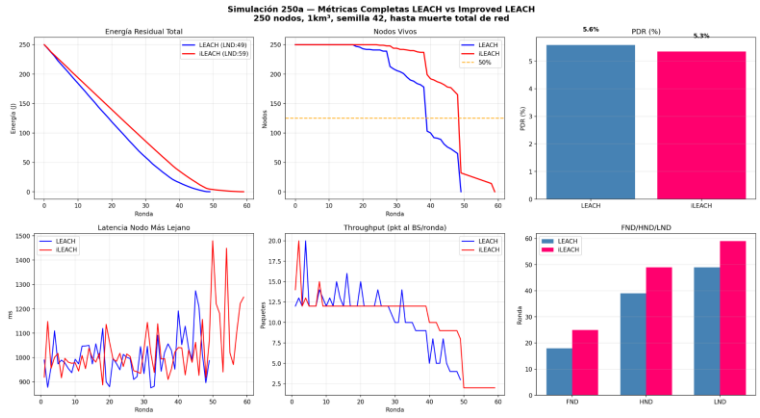
Comparación tridimensional inicial de LEACH e Improved-LEACH para 250 nodos, simulación A.



Nota. La mayor densidad de nodos permite observar la distribución espacial en profundidad y distancia respecto de la estación base. Elaboración propia.

Anexo 32

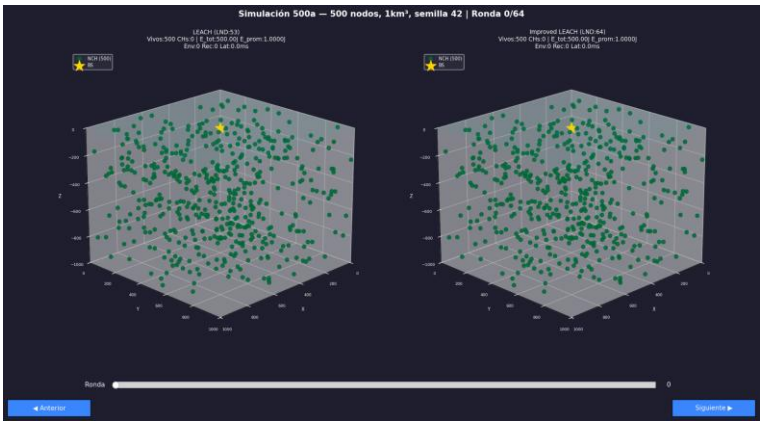
Métricas completas de LEACH e Improved-LEACH para 250 nodos, simulación A.



Nota. La figura muestra la evolución de los protocolos hasta la muerte total de la red y permite contrastar su comportamiento en un escenario de densidad intermedia. Elaboración propia.

Anexo 33

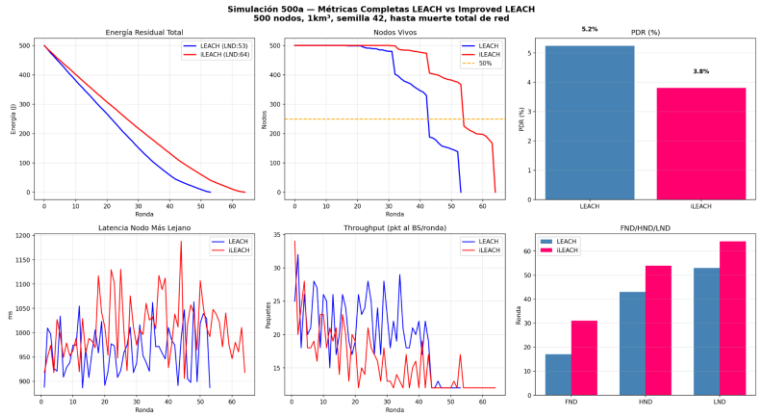
Comparación tridimensional inicial de LEACH e Improved-LEACH para 500 nodos, simulación A.



Nota. Se observa la distribución de 500 nodos dentro del volumen subacuático y la posición de la estación base en la superficie. Elaboración propia.

Anexo 34

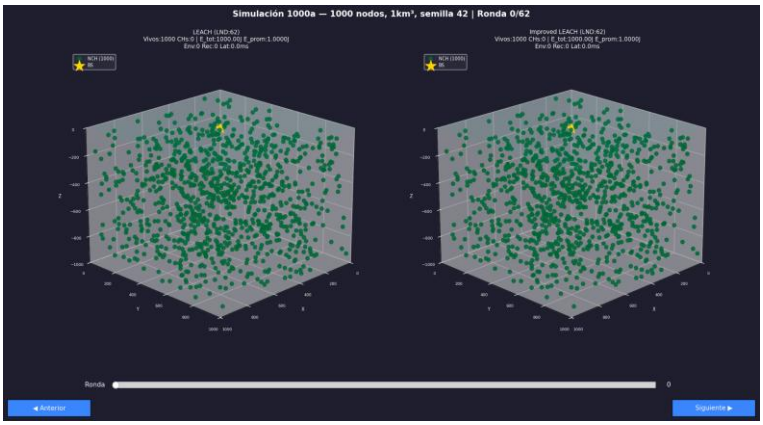
Métricas completas de LEACH e Improved-LEACH para 500 nodos, simulación A.



Nota. La figura reúne los resultados energéticos, de conectividad y de vida útil registrados para ambos protocolos. Elaboración propia.

Anexo 35

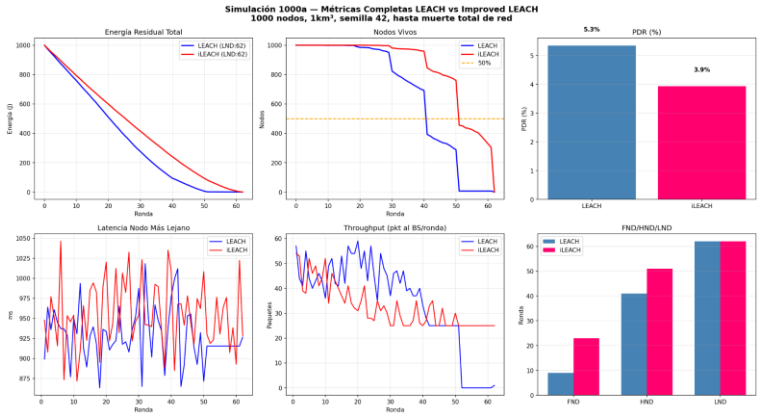
Comparación tridimensional inicial de LEACH e Improved-LEACH para 1000 nodos, simulación A.



Nota. La captura representa el escenario de mayor densidad considerado en la investigación, con la misma topología inicial para ambos protocolos. Elaboración propia.

Anexo 36

Métricas completas de LEACH e Improved-LEACH para 1000 nodos, simulación A.



Nota. Se presentan los resultados del escenario de máxima densidad, incluyendo energía residual, nodos vivos, PDR, latencia, throughput y FND/HND/LND. Elaboración propia.

OBSERVACIÓN GENERAL DEL ANEXO

Las imágenes incluidas corresponden exclusivamente a la semilla A de los cinco escenarios evaluados. Su finalidad es documentar visualmente la correcta construcción de las topologías tridimensionales y el comportamiento de las métricas registradas durante la simulación.

Las simulaciones B, C, D, E, F, G, H, I y J fueron ejecutadas con semillas diferentes para introducir variabilidad en la distribución de los nodos y en los procesos probabilísticos de selección de cabezas de clúster. Debido a que las curvas y representaciones obtenidas siguen patrones semejantes, su inclusión completa produciría una repetición innecesaria de figuras.

La omisión de dichas capturas no implica la exclusión de sus resultados. Los datos de todas las repeticiones fueron utilizados para calcular promedios, desviaciones, intervalos de confianza y diferencias entre LEACH e Improved-LEACH. Por esta razón, las imágenes de la semilla A constituyen evidencia visual representativa, mientras que las tablas y gráficas estadísticas del capítulo de resultados sintetizan la campaña experimental completa.