



**UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE CUENCA**

CARRERA DE INGENIERÍA ELECTRÓNICA

**IMPLEMENTACIÓN DE PROTOCOLOS DE ENRUTAMIENTO PARA REDES DE
SENSORES INALÁMBRICOS FIJOS BASADAS EN ZIGBEE, UTILIZANDO
DIMENSIONAMIENTO ÓPTIMO DE REDES**

Trabajo de titulación previo a la obtención del
título de Ingeniero Electrónico

AUTOR: JORDI CASTEL TAPIA

TUTOR: ING. JUAN PAÚL INGA ORTEGA, MgT.

Cuenca – Ecuador

2022

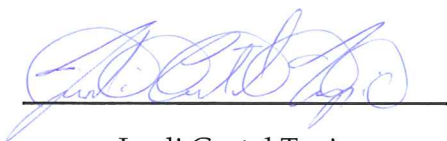
**CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA DEL TRABAJO DE
TITULACIÓN**

Yo, Jordi Castel Tapia con documento de identificación N° 1754641429 manifiesto que:

Soy el autor y responsable del presente trabajo; y, autorizo a que sin fines de lucro la Universidad Politécnica Salesiana pueda usar, difundir, reproducir o publicar de manera total o parcial el presente trabajo de titulación.

Cuenca, 28 de febrero del 2022

Atentamente,



Jordi Castel Tapia

1754641429

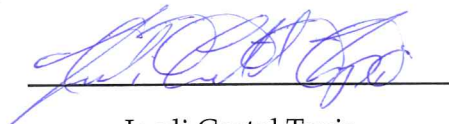
**CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE
TITULACIÓN A LA UNIVERSIDAD POLITÉCNICA SALESIANA**

Yo, Jordi Castel Tapia con documento de identificación No. 1754641429, expreso mi voluntad y por medio del presente documento cedo a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que soy autor del proyecto técnico con enfoque investigativo: "Implementación de protocolos de enrutamiento para redes de sensores inalámbricos fijos basadas en ZigBee, utilizando dimensionamiento óptimo de redes", el cual ha sido desarrollado para optar por el título de: Ingeniero Electrónico, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En concordancia con lo manifestado, suscribo este documento en el momento que hago la entrega del trabajo final en formato digital a la Biblioteca de la Universidad Politécnica Salesiana..

Cuenca, 28 de febrero del 2022

Atentamente,



Jordi Castel Tapia

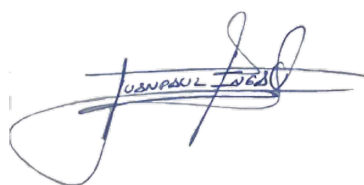
1754641429

CERTIFICADO DE DIRECCIÓN DEL TRABAJO DE TITULACIÓN

Yo, Juan Paúl Inga Ortega con documento de identificación N° 0104166491, docente de la Universidad Politécnica Salesiana, declaro que bajo mi tutoría fue desarrollado el trabajo de titulación: IMPLEMENTACIÓN DE PROTOCOLOS DE ENRUTAMIENTO PARA REDES DE SENSORES INALÁMBRICOS FIJOS BASADAS EN ZIGBEE, UTILIZANDO DIMENSIONAMIENTO ÓPTIMO DE REDES, realizado por Jordi Castel Tapia con documento de identificación N° 1754641429, obteniendo como resultado final el trabajo de titulación bajo la opción proyecto técnico con enfoque investigativo que cumple con todos los requisitos determinados por la Universidad Politécnica Salesiana.

Cuenca, 28 de febrero del 2022

Atentamente,

A handwritten signature in blue ink, appearing to read 'JUAN PAUL INGA ORTEGA', is written over a horizontal line.

Juan Paúl Inga Ortega

01041664941

AGRADECIMIENTOS

Agradezco primeramente a mi familia por el constante apoyo que me permitió culminar esta tesis, también agradecerle al Ing. Juan Inga quien fue una guía y mentor durante todo el proceso, por último a mis amigos quienes ayudaron con la implementación de la red.

DEDICATORIA

Esta tesis va dedicada a mis familiares y amigos quienes me han apoyado durante todo el trayecto de mi carrera, en especial a mi madre Julia y mi hermano Marek quienes han estado siempre presentes a pesar de la distancia.

Índice General

Índice General	I
Índice de Figuras	III
Índice de Tablas	V
Resumen	VI
Abstract	VIII
Objetivos	X
Antecedentes o Problema de Estudio	XI
Justificación	XIV
Introducción	1
1. Conceptos Teóricos	3
1.1. Redes de Sensores Inalámbricas	3
1.2. Protocolos de Enrutamiento	9
1.3. Dimensionamiento de Redes	13
1.4. ZigBee	14
2. Problema de estudio	19
2.1. WSN Implementada	20
2.2. Protocolos de Enrutamiento	26
2.2.1. Dijkstra	31

<i>ÍNDICE GENERAL</i>	II
2.2.2. Prim	35
2.2.3. Low-Energy adaptative Clustering Hierachy (LEACH)	38
2.2.4. Power-Efficient Gathering in Sensor Information Systems (PEGASIS)	41
2.3. Dimensionamiento de la Red	45
3. Resultados	48
3.1. Escenario sin dimensionamiento	49
3.2. Escenario Aplicando Dimensionamiento	54
4. Conclusiones Recomendaciones y Trabajos Futuros	62
Glosario	66
Referencias	67

Índice de Figuras

1.1. Componentes de un nodo sensor.	4
1.2. Topología de las redes WSN	8
1.3. Clasificación de los protocolos de enrutamiento para WSN.	13
1.4. Capas del estándar Zigbee	16
2.1. Ubicación de los nodos en el campus universitario.	21
2.2. Módulo XBEE Click.	22
2.3. Esquemas de conexión de los nodos sensores usando un microcontrolador STM32F103C8T6 y STM32F401CxUx	23
2.4. Esquemas de conexión de los nodos sensores usando Arduino UNO	23
2.5. Módulo ETH WIZ Click.	24
2.6. Nodo Coordinador/Concentrador: Pasarela de transferencia de datos.	24
2.7. Programación en LabView para monitoreo de datos recibidos del coordinador.	25
2.8. Comunicación entre nodos en la fase de configuración y estado estable.	28
2.9. Diagrama de flujo del proceso de descubrimiento de los nodos sensores.	29
2.10. Conformación de la trama para la transmisión de los datos.	31
2.11. Ejemplo de ejecución del algoritmo Dijkstra.	32
2.12. Diagrama de Flujo del algoritmo Dijkstra.	33
2.13. Diagrama de Flujo del funcionamiento del nodo sensor con el protocolo Dijkstra.	34
2.14. Ejemplo de ejecución del algoritmo Prim.	35
2.15. Diagrama de Flujo del funcionamiento del nodo sensor con el protocolo Prim.	36
2.16. Diagrama de Flujo del algoritmo Prim.	37
2.17. Topología lógica del protocolo LEACH.	38
2.18. Diagrama de Flujo del funcionamiento del nodo sensor con el protocolo LEACH.	40

2.19. Topología lógica del protocolo PEGASIS.	42
2.20. Diagrama de Flujo del funcionamiento del nodo sensor con el protocolo PEGASIS. . .	44
2.21. Dimensionamiento y enrutamiento georreferenciados de redes de sensores inalámbricos.	47
3.1. Paquetes recibidos por nodo.	50
3.2. Latencia promedio en la transmisión de cada nodo.	51
3.3. Paquetes recibidos por nodo con eventos de desconexión de nodos.	52
3.4. Latencia promedio en la transmisión de cada nodo con eventos de desconexión.	53
3.5. Identificación de las ubicaciones para optimización de la red.	56
3.6. Paquetes recibidos por nodo.	58
3.7. Latencia promedio en la transmisión de cada nodo.	59
3.8. Paquetes recibidos por nodo.	60
3.9. Latencia promedio en la transmisión de cada nodo.	61

Índice de Tablas

1.1. Composición de trama para transmisión - Solicitud de transmisión (0x10)	18
1.2. Composición de trama para recepción - Recibir paquete (0x90)	18
2.1. Puntos Georreferenciados de los nodos de la red.	21
3.1. Puntos Georreferenciados de los sitios candidatos.	54
3.2. Resultados del dimensionamiento.	55
3.3. Enlaces entre coordinadores y nodos sensores.	57

Resumen

En una red de sensores inalámbrica (*WSN*), para una transmisión eficiente en cuanto al consumo energético sin ralentizar su operación y con la menor pérdida de paquetes, el procesamiento de la información en cada nodo y sobre todo el análisis y cálculo de las mejores rutas para enviar información desde un nodo origen al punto de agregación de datos (*DAP*) son los problemas más desafiantes para este tipo de redes. Por otro lado, el dimensionamiento óptimo de una red busca garantizar que los recursos físicos de las tecnologías que se usen en *WSN* sean los óptimos permitiendo incluso reducir los costos de implementación al considerar por ejemplo restricciones de capacidad y cobertura en cada nodo en la red. Por esto, el presente trabajo propone el estudio de la influencia del dimensionamiento óptimo en los protocolos de enrutamiento en *WSN*, en este caso al evaluar algoritmos de cálculo de rutas como LEACH, PEGASIS, Dijkstra y Prim usando la tecnología de comunicación inalámbrica ZigBee que permite el despliegue de múltiples topologías de red para *WSN*. Entonces, se plantean dos escenarios, el primero consiste en el despliegue de la red ubicando los nodos en sitios por conveniencia y el segundo en un escenario optimizado aplicando el modelo lineal de optimización con restricciones de capacidad y cobertura. En ambos escenarios se analizó la latencia y tasa de pérdida de paquetes existentes en cada protocolo. Este trabajo muestra que la planificación y despliegue óptimo de una *WSN* puede traer mejoras en el enrutamiento de los paquetes optimizando la cantidad de enlaces a través de la ubicación adecuada de los nodos *DAP* optimizando los enlaces entre los nodos de tal forma que se reduce la latencia y la tasa de pérdidas de paquetes. La evaluación considera datos de medición inteligente identificando que los protocolos PEGASIS y Prim, presentan una menor tasa de pérdida de paquetes siendo este un requisito de alto interés en aplicaciones de medición inteligente y monitoreo.

Palabras clave: Dijkstra; Dimensionamiento Óptimo; Enrutamiento; Latencia; LEACH; PEGASIS; Prim; WSN; ZigBee

Abstract

In a wireless sensor network (*WSN*), for an efficient transmission in terms of energy consumption without slowing down its operation and with the least packet loss, the processing of information at each node and especially the analysis and calculation of the best routes to send information from a source node to the data aggregation point (*DAP*) are the most challenging problems for this type of network. On the other hand, the optimal sizing of a network seeks to ensure that the physical resources of the technologies used in *WSN* are optimal, even allowing to reduce implementation costs by considering, for example, capacity and coverage constraints at each node in the network. Therefore, this paper proposes the study of the influence of optimal dimensioning on routing protocols in *WSN*, in this case by evaluating routing algorithms such as LEACH, PEGASIS, Dijkstra and Prim using ZigBee wireless communication technology that allows the deployment of multiple network topologies for *WSNs*. Then, two scenarios are proposed, the first one consists in the deployment of the network by placing the nodes in sites for convenience and the second one in an optimized scenario applying the linear optimization model with capacity and coverage constraints. In both scenarios, the existing latency and packet loss rate for each protocol were analyzed. This work shows that the optimal planning and deployment of a *WSN* can bring improvements in packet routing by optimizing the number of links through the proper placement of *DAP* nodes by optimizing the links between nodes in such a way that latency and packet loss rate are reduced. The evaluation considers smart metering data identifying that PEGASIS and Prim protocols, present a lower packet loss rate being this a requirement of high interest in smart metering and monitoring applications.

Keywords: Dijkstra; Optimal Dimensioning; Routing; Latency; LEACH; PEGASIS; Prim; WSN; ZigBee

Objetivos

Objetivo General

- Implementar protocolos de enrutamiento para redes WSN fijas basadas en ZigBee, utilizando dimensionamiento óptimo de redes aplicado a Smart Grids.

Objetivos específicos:

- Realizar el estado del arte de los principales protocolos de enrutamiento para WSN fijas usando ZigBee para identificar los protocolos más promocionados en aplicaciones orientadas a Smart Grids y Smart Cities.
- Implementar una red de sensores inalámbricos fija con tecnología ZigBee y con topología mesh que permita implementar los protocolos de enrutamiento identificados en el estado del arte.
- Probar los protocolos de enrutamiento utilizando la teoría del dimensionamiento de redes óptimo y sin considerar este criterio, esto para determinar el rendimiento y eficiencia de cada red en base a los parámetros definidos en el estado del arte.

Antecedentes

La revolución tecnológica que acarrea la cuarta revolución industrial (Industria 4.0) ha permitido que diversos conceptos tecnológicos aceleren su desarrollo, por ejemplo, la posibilidad de conectar prácticamente todo tipo de dispositivo a Internet ha derivado el concepto llamado “Internet de las Cosas” (IoT, del inglés Internet of Things) [1, 2, 3]. Así, las redes de sensores inalámbricas (WSN, del inglés Wireless Sensor Network), por su bajo costo y versatilidad en diferentes escenarios de aplicación como el monitoreo de diversas variables de entorno llevan siendo objeto de interés en la investigación e implementación para el despliegue de IoT [1], [4].

Existe una amplia gama tecnológica relacionada a las WSN y de acuerdo al escenario de aplicación ha generado el desarrollo de soluciones tecnológicas heterogéneas, por lo cual, la interoperabilidad entre estas tecnologías son desafíos importantes. Así, en la implementación de estas redes, suele ser necesario el uso de puertas o pasarelas de enlace (llamados *gateways*) que permitan la interoperabilidad entre estándares de comunicación heterogéneos y así aplicar una abstracción entre las capas de hardware y de aplicación de usuario [3], [5], [6]. De esta manera, se busca asegurar que la información de cualquier nodo de una WSN pueda escalar hacia los servicios de Internet. Toda esta interacción entre capas permite que las cosas puedan conectarse a Internet siendo la base del fundamento de IoT.

Muchas de las redes de IoT pueden desplegarse con diferentes tipos de topologías adicionales que pueden ser desplegadas y/o combinadas, siendo las de tipo árbol o en malla las más usadas en lo que respecta a perfiles de bajo consumo de energía o transporte de datos; por ejemplo, [7] propone una WSN que permita múltiples saltos en aplicación de infraestructuras de medición avanzada (AMI, del inglés Advanced Measurement Infrastructure) a través de una topología jerárquica.

En cualquier caso, aprovechar multi saltos para que la información recorra desde un nodo sensor hasta un nodo considerado como punto de agregación de datos (**DAP**, del inglés Data Aggregation Point) de la red es muy importante en aplicaciones con estas topologías identificando cuál es el nodo origen de la información. Así, para el despliegue de redes de **IoT**, la gestión eficiente de la topología de la **WSN** es un requisito crucial y en particular, en las implementaciones a gran escala [8]. Entonces, diversas topologías pueden ser implementadas dependiendo de las aplicaciones, del tipo de tráfico y de las restricciones de la fuente de información [9].

En lo que respecta a las tecnologías que permiten la aplicación de **IoT** a través de medios inalámbricos, en general existen dos grandes grupos que son: las redes de baja potencia de corto alcance y de largo alcance [10]. En el primer caso, está la propuesta de usar **IPv6** para el direccionamiento de datos a través de redes de área personal (**PAN**, del inglés Personal Area Network) donde protocolos como *Bluetooth* o los basados en el estándar IEEE 802.15.4 como *ZigBee* o *Z-Wave* son los principales representantes. Para el segundo grupo, los protocolos que muestran, principalmente, mayor despunte en los últimos años son *LoRaWAN* y *SigFox*.

No obstante, es menester indicar que, si bien *ZigBee* a través de su propulsor *ZigBee Alliance*, actualmente busca el dar soporte al uso de **IPv6** y por tanto de **6LoWPAN**, no todos los módulos *ZigBee*, en la actualidad, disponen de una pila que permita implementar el protocolo, aunque sí permite el uso de direccionamiento basado en el formato de trabajo de **TCP/IP** a través de direccionamientos de red **PAN** conocidos como direcciones de vecinos **PAN**. Por esto el uso de pasarelas intermedias se reduciría a unas cuantas para convertir paquetes *ZigBee* a paquetes que puedan, por ejemplo, usar el mismo **IPv6** y así subir los datos correspondientes a Internet a través de redes **LAN**, **WAN**, **MAN**. Para el caso de **LPWAN**, es necesario de múltiples pasarelas en el camino hasta llegar a un servidor que permita subir los datos adquiridos por la **WSN** a la Internet. En este sentido, es importante considerar un despliegue para **IoT**, basado en **IP**, debido a las bondades que este protocolo brinda [11].

Entonces, frente a todo lo mencionado, el enrutamiento es un problema muy desafiante y de preocupación directa en **WSN**, [12] porque existen diversas problemáticas o parámetros que se deben considerar para realizar una transmisión

eficiente reduciendo el consumo energético y el procesamiento por parte de los nodos de la red.

Entre las problemáticas del enrutamiento se encuentra la escalabilidad, la tolerancia a fallos, la heterogeneidad de energía y capacidad computacional, el consumo energético, la conectividad, cobertura, calidad de servicio, entre otros [13]. De acuerdo con esto, el consumo energético permite definir la vida útil de la WSN, por lo que es imperativo administrar la energía de los dispositivos para mejorar su rendimiento y vida útil, de tal forma que el despliegue masivo de dispositivos remotos pueda tener energía disponible para sus funciones y se reduzca la sustitución frecuente de baterías ya que representan un gasto significativo en el ámbito operativo [14].

Justificación

La implementación de [WSN](#) aplicando modelos de dimensionamiento óptimo puede traer mejoras en el enrutamiento de los paquetes, ya que, al involucrar la capacidad, cobertura, interferencia y número de canales para la ubicación de los nodos, se puede optimizar la cantidad de enlaces y disminuir el consumo energético, así como el procesamiento computacional en cada nodo. Es por esto que, se plantea estudiar diferentes protocolos de enrutamiento en una [WSN](#) aplicando criterios de dimensionamiento en contraste con un escenario donde no se lo aplique, analizando métricas como latencia, vida útil de la red, consumo energético de la red, rendimiento (throughput), entre otras, para de esta manera identificar los protocolos que permiten optimizar los recursos en la red.

Las aplicaciones de [IoT](#) son diversas y con varios requisitos, por ejemplo, la distancia de alcance (rango), velocidad de transmisión de datos, nivel de seguridad, demanda de energía, latencia, nivel de confiabilidad. Cada tecnología tiene sus características clave y únicas además de otras características como conectividad, área de cobertura, tecnología de radio, capacidad de canal, consumo de energía, seguridad, movilidad, confiabilidad y latencia. Siendo un verdadero desafío, a largo plazo, la coexistencia exitosa de todas las tecnologías disponibles. En este sentido, para garantizar esta coexistencia de tecnologías, trabajar con redes que permitan enrutamiento y el manejo de múltiples saltos en el traslado de información, brinda a la red y a las aplicaciones flexibilidad y ubicuidad. No obstante, como se mencionó en el apartado anterior, evaluar los protocolos de enrutamiento para [WSN](#) combinados con un dimensionamiento de red óptimo puede garantizar el uso adecuado de los recursos de red y mejorar características de las aplicaciones en términos de costos y energía garantizando la cobertura mínima de usuarios y permitiendo una planificación

escalable y sostenible de la misma en el tiempo [7], [15], [16].

Por lo tanto, el presente trabajo pretende evaluar esta combinación del dimensionamiento óptimo de WSN con diversos protocolos de enrutamiento y así contrastar, a través de una implementación, los resultados obtenidos en trabajos previos desarrollados por los grupos de investigación GIREI (Grupo de Investigación en Redes Eléctricas Inteligentes) de la Universidad Politécnica Salesiana y GITEL (Grupo de Investigación en Telecomunicaciones y Telemática).

Finalmente, se debe indicar que la tecnología a usar en la red WSN evaluada en este trabajo es ZigBee debido a que permite el uso de enrutadores PAN y que, al estar soportada por el estándar IEEE 802.15.4, la frecuencia de trabajo corresponde a la banda de radio industriales, científicas y médicas (ISM) por lo que el uso del espectro radio eléctrico es libre. Además, es una tecnología que presenta una solución de ahorro de energía de bajo costo que incluye prevención de colisiones y permite formar redes con una gran cantidad de nodos y usa *spread spectrum* para evitar interferencias por la transmisión de datos en bandas similares [15], [17], [18].

Introducción

La necesidad de monitorear diferentes variables ambientales y físicas y buscar el control sobre prácticamente todo, la implementación de los sistemas IoT a nivel mundial se ha incrementado en forma exponencial y en este sentido, las WSN tienen una importancia crítica por su flexibilidad y ubicuidad en la implementación de la red según la necesidad del escenario.

Un caso particular de aplicación de WSN es en el despliegue de la red de comunicaciones que de soporte al segmento de medición inteligente (AMI) propio de las redes inteligentes o Smart Grid [7]. Dentro de este caso en particular, surgen diversas propuestas de red debido a que el paradigma de Smart Grid demanda el uso eficiente de los recursos y así se establece la necesidad de evaluar cada detalle referente al funcionamiento de la red de comunicaciones de este caso en particular [15], [19]. En este sentido, se debe mencionar que los datos tomados de los medidores inteligentes, en general, son muestreados cada 7 o cada 15 minutos [20], [21]. Sin embargo, por diversas circunstancias pueden no necesariamente transmitirse hacia la empresa eléctrica de distribución al instante en el cual se aplica el muestreo. También puede darse situaciones en las cuales solo se transmita una vez al día los datos de consumo eléctrico o cada siete días, incluso más tiempo, por ejemplo, cada quince o hasta treinta días si es que presentan problemas de conexión de la red o por congestión de la misma [22], razón por la cual este escenario muestra la necesidad de establecer un protocolo de comunicación adecuado de acuerdo a la topología de la red.

Por todo esto, para cualquier situación de aplicación de WSN y como se ha comentado, incluso para aplicaciones de Smart Grids, para un correcto despliegue de los nodos en la WSN es imprescindible dimensionar y planificar la red con la finalidad de reducir los costos de instalación, mejorar la eficiencia y rendimiento de la red y

aumentar el número de posibles usuarios [14], [22] además de que se busca garantizar que los parámetros físicos de la tecnología usada, como cobertura y capacidad, puedan cumplirse. Este tema ha generado varias investigaciones que abordan los modelos matemáticos óptimos y que se basan en la cobertura, capacidad, interferencia y número de canales dentro de la red, que son parámetros principales en el dimensionamiento óptimo de las redes de comunicación [14], [15].

Por lo tanto, el despliegue y enrutamiento de una WSN cuya función es la transmisión de información hacia una estación base u oficina central, debe garantizar la interconexión con cada nodo sensor de tal forma que cada nodo pueda retransmitir su información a otro nodo generando una cadena multisalto con la finalidad de reducir el consumo energético del nodo. Esta situación indica que es necesario optimizar los recursos de la red, de tal forma que el número de transmisiones entre nodos sea el adecuado de acuerdo a la aplicación de la red [23].

Con todo lo mencionado, es posible indicar que el dimensionamiento de red y el enrutamiento de los paquetes tienen una estrecha relación, por lo tanto el objetivo del presente trabajo es analizar el efecto del dimensionamiento óptimo en las métricas de latencia y tasa de pérdida de paquetes de los protocolos de enrutamiento LEACH, PEGASIS, Dijkstra y Prim implementados en una WSN fija con tecnología ZigBee con aplicación a Smart Grids. En el escenario planteado se realiza la transmisión de paquetes con los datos obtenidos a partir del sistema de medición inteligente de la Universidad Politécnica Salesiana considerando una WSN sin criterios de dimensionamiento y otra aplicando el modelamiento matemático de la capacidad y cobertura.

Capítulo 1

Conceptos Teóricos

El presente capítulo presenta una introducción a los conceptos teóricos tratados en este trabajo, como es la definición de las redes de sensores inalámbricas, los tipos de protocolos de enrutamiento utilizados en las WSN, la teoría del dimensionamiento de redes utilizado para optimizar el tráfico de datos por la red y para finalizar se aborda sobre la tecnología ZigBee utilizada para la implementación de la red.

1.1. Redes de Sensores Inalámbricas

Las WSN son sistemas que constan de grandes cantidades de nodos multifuncionales compuestos por un microprocesador, transceptores de radio y los sensores de acuerdo a las variables analizadas. Estos nodos por lo general son de tamaño pequeño pero mediante todos sus elementos tiene la capacidad de medir condiciones ambientales y transformarlas a una señal eléctrica para luego ser enviada entre ellos o a una estación de procesamiento centralizada de manera inalámbrica, por lo tanto, un nodo no solo tiene un componente de detección, sino también capacidad de procesamiento, comunicación y almacenamiento de los datos [2], [22].

Para conformar una WSN los nodos se encuentran distribuidos en una región para que de manera remota transmitan los datos recopilados por un medio inalámbrico [2]. La Figura 1.1 presenta el diagrama esquemático de los componentes del nodo sensor y la arquitectura de comunicación de la WSN [24]. Las WSN son específicas de la aplicación y, por lo tanto, deben involucrar tanto software como hardware,

permitiéndoles utilizar diferentes tecnologías y estándares en base a su aplicación [2].

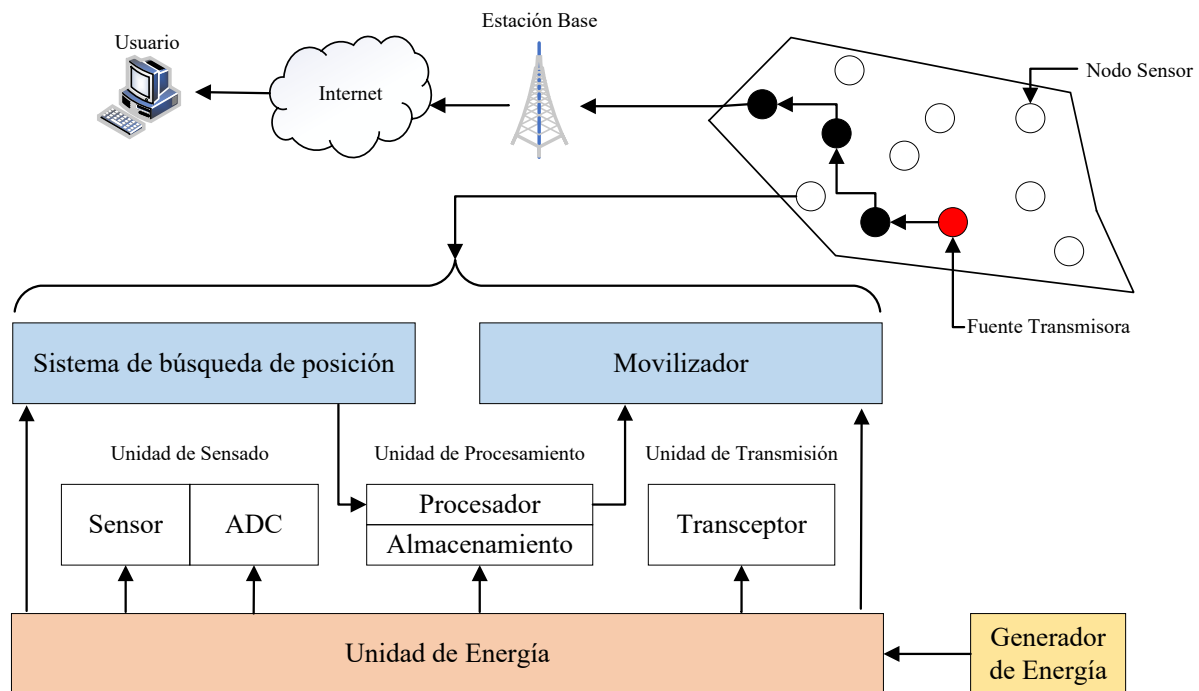


Figura 1.1: Componentes de un nodo sensor.

A diferencia de las redes inalámbricas convencionales, las WSN tienen limitaciones y características únicas lo que les permite ser flexibles a los escenarios de implementación [25]. Entre las características más relevantes de este tipo de red se encuentran la densidad de nodos desplegados en el área de monitoreo la cual puede ser muy grande con respecto a las redes móviles, la red es flexible a los cambios de topología por la pérdida de conectividad de los nodos [25] y además al ser una red extensa no es posible realizar un esquema de direccionamiento global que involucre a todos los nodos ya que se generaría una sobrecarga del mantenimiento de la identificación de cada nodo, por lo tanto, los protocolos tradicionales basados en IP no son aplicables a las WSN [24], [26]. Por otro lado, las limitaciones principales que tienen las WSN es el almacenamiento de los datos censados, la comunicación inalámbrica entre nodos, el procesamiento de los paquetes recibidos y por transmitir [27], y principalmente el uso de batería lo que conlleva una restricción de energía para los nodos y pérdida de información en el momento que un nodo se desconecta de la red por su nivel de batería, también al ser colocados en sitios remotos se corre el riesgo

de daños o fallos disminuyendo la confiabilidad de los nodos, por último este tipo de redes por lo general son planificadas para una aplicación en específico y para ciertas características del entorno [25].

Para el diseño de la red, se deben considerar diversos objetivos que permitan delimitar las características de la red. De acuerdo con [25], los objetivos que se deben considerar para este tipo de redes son los siguientes:

1. **Consumo energético:** para prolongar la vida útil de la red y contar con un mayor flujo de paquetes, se debe reducir el consumo energético de los nodos. Para esto se debe considerar diferentes parámetros como la tecnología a utilizar, el protocolo de enrutamiento, entre otros.
2. **Escalabilidad:** la **WSN** debe tener la capacidad de crecer aumentando la cantidad de nodos, para de esta forma brindar cobertura a una mayor cantidad de usuarios.
3. **Tamaño del nodo:** la disminución del tamaño de los nodos facilita su implementación en cada uno de los puntos remotos y a su vez reducen el costo del nodo.
4. **Tolerancia a fallos:** frente a cualquier estímulo externo donde el funcionamiento de la red se encuentre comprometido, la red debe tener la capacidad de reestructurar la trayectoria de la información con la finalidad de disminuir la pérdida de paquetes.
5. **Soporte de Calidad de Servicio (QoS):** dependiendo de la aplicación de la red, la latencia y la pérdida de paquetes se tendrá que disminuir hasta que la cantidad de paquetes recibidos sean los adecuados y sean recibidos en el tiempo esperado.
6. **Utilización del canal:** dependiendo de la tecnología utilizada, los nodos tendrán recursos de ancho de banda limitados. Por lo tanto, los protocolos de enrutamiento deben optimizar este recurso para mejorar el uso del canal.
7. **Seguridad:** al igual que las redes tradicionales, en las **WSN** se debe tener mecanismos de seguridad para evitar el acceso no autorizado a la información recopilada.

Además de los objetivos de diseño descritos anteriormente, en las WSN es imperativo gestionar la topología de interconexión de los nodos sensores ya que a partir de esta se podrá garantizar la conservación de energía mientras se mantienen los enlaces de comunicación de los nodos. Esta gestión consiste en conocer las conexiones inalámbricas de los nodos dependientes a la línea de vista de los mismos, y las relaciones lógicas entre ellos [28]. La Figura 1.2 presentan las topologías que se pueden implementar en una WSN, cuyas características son las siguientes [13]:

- **Estrella:** en este tipo de topología la información es centralizada hacia un nodo que tiene la función de concentrador o directamente hacia la estación base. La ventaja de esta topología es reducir la latencia en la recepción de los paquetes ya que se disminuyen la cantidad de multisaltos, por otro lado, la desventaja es la pérdida de conexión de los nodos más alejados a la estación base que cuentan con bajos niveles energéticos impidiendo alcanzar la potencia requerida de transmisión.
- **Anillo:** se trata de una topología utilizada cuando la cantidad de nodos de la red es pequeña, ya que para recibir el mensaje desde el transmisor, los paquetes deberán recorrer todos los nodos que se encuentren en la conexión anillo entre el transmisor y receptor. Por este motivo, no es recomendable en las WSN ya que al aumentar la cantidad de nodos el canal empieza a degradarse y a aumentar la latencia.
- **Árbol:** con esta topología se busca generar una jerarquía entre los nodos, donde por lo general los nodos hijos se dedican principalmente a la transmisión de la información, mientras que los nodos padre tendrán que recibir los paquetes para agregar su información y retransmitirlos hacia la estación base.
- **Mallado con conexión total:** con esta topología se consigue la interconexión entre todos los nodos de la red. En una malla con n nodos se requieren de $n(n - 1)/2$ enlaces directos. Debido a esto, esta topología es muy costosa y compleja de implementar porque los nodos deben contar con línea de vista directa con los demás nodos de la red, por otro lado, esta topología es muy confiable y se garantiza recibir la mayor cantidad de paquetes transmitidos.

- **Mallado con conexión parcial:** para esta topología de red, cada nodo se conectará a cierta cantidad de nodos generando en algunos casos una malla completa. Esta topología es menos costosa a la de mallado con conexión total pero por supuesto llega a ser menos confiable porque se reducen los enlaces redundantes.

Para un correcto funcionamiento de una WSN, se deben medir e interpretar algunas métricas que definen el rendimiento de la red, estas métricas están en función de las características de la red y de la aplicación de la misma [29]. Las métricas más comunes que son medidas en una WSN de acuerdo con [29] son:

- **Vida útil de la red:** esta definida por la eficiencia energética que tiene la red, es decir, el tiempo de funcionamiento de los nodos hasta que estos cuenten con un nivel energético muy bajo imposibilitando alcanzar los niveles de potencia necesarios para la transmisión de los datos. Esta métrica se puede medir con parámetros genéricos como el tiempo hasta que cierto porcentaje de nodos dejan de funcionar, o con métricas dirigidas a la aplicación como es cuando los nodos dejan de proporcionar a la aplicación la información deseada sobre el entorno, o también se puede medir en base a la tasa de pérdida de paquetes la cual si supera un umbral se determina que la red a culminado su vida útil. En base a esto, el diseño de la red debe buscar la eficiencia en el consumo energético de cada nodo ocupando diferentes estrategias.
- **Consumo de energía:** en esta métrica se analiza la energía total utilizada por toda la red, ya que cada nodo requiere cierta cantidad de energía para realizar todos sus procesos como la transmisión, recepción y procesamiento. Para disminuir este consumo, la estrategia empleada en las WSN es la inactividad de los nodos frente a la red, es decir, eliminar los procesos internos de cada nodo por un periodo de tiempo o hasta que suceda un evento.
- **Latencia:** es la métrica referente al retraso de tiempo desde el envío de un paquete desde el transmisor hasta la recepción exitosa del mismo en el receptor. Para esta métrica se debe considerar los efectos de propagación en los diferentes escenarios y a su vez del tiempo de procesamiento de los procesadores de cada nodo.

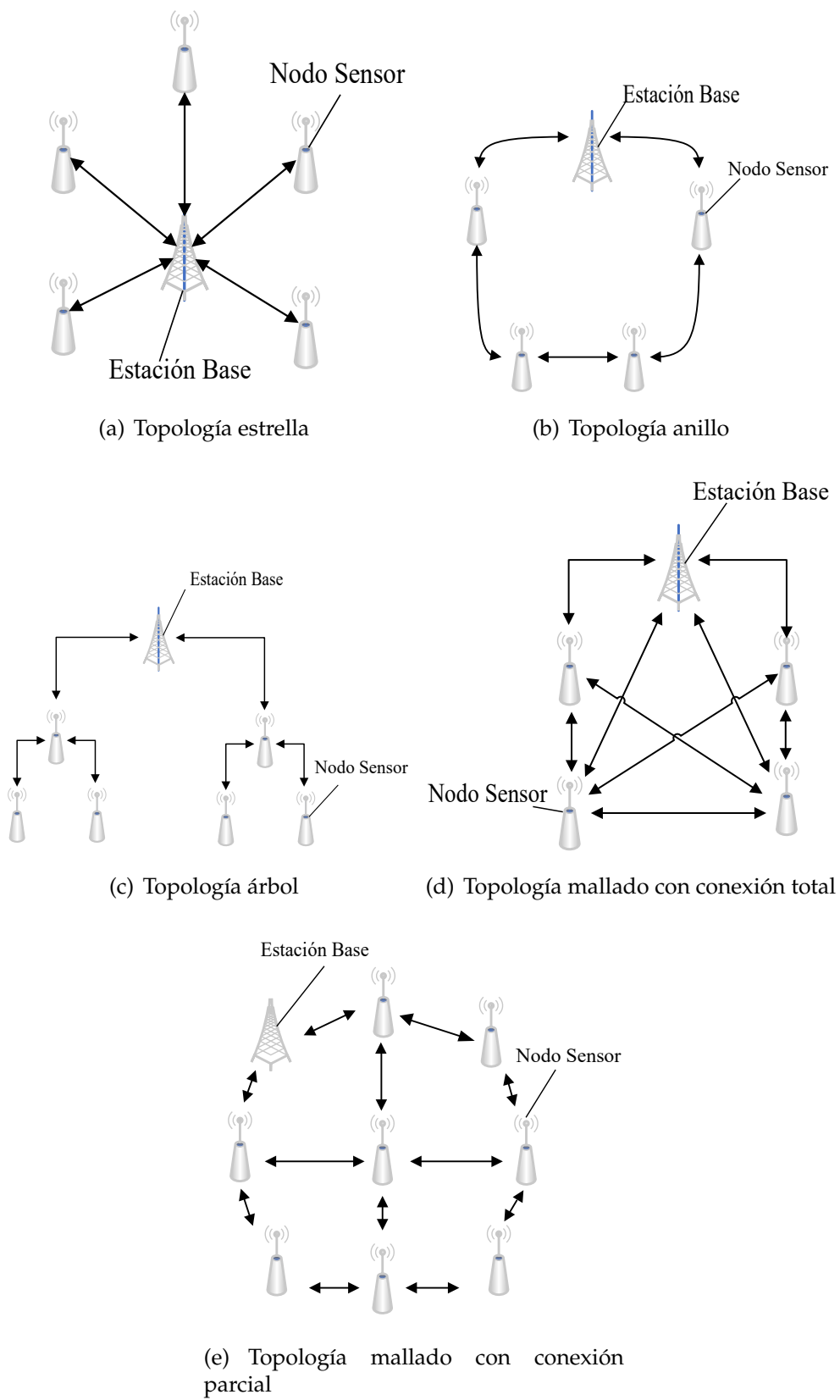


Figura 1.2: Topología de las redes WSN

Dependiendo de la aplicación de la *WSN*, la latencia tendrá cierto umbral para determinar si la red cumple con los requerimientos de la aplicación.

- **Exactitud:** esta definida por la semejanza de los datos transmitidos y recibidos, es decir, si el paquete transmitido no sufrió ninguna alteración mientras se transmitía hacia el receptor. Una *WSN* debe ser adaptativa, de tal modo que su desempeño logre la exactitud y latencia deseada con un gasto mínimo de energía. Esta métrica está fuertemente influenciada por los fenómenos físicos presentes en la propagación de las señales electromagnéticas.
- **Tolerancia a fallos:** el despliegue de las *WSN* en sitios remotos aumenta el riesgo de fallos de los nodos por las condiciones físicas del entorno o cuando se agota la batería de los mismos, es por esto que las *WSN* deben ser capaces de adaptarse a la desconexión de un nodo de tal forma que se disminuya la tasa de pérdida de paquetes.
- **Escalabilidad:** una *WSN* debe soportar el aumento de la carga de trabajo, es decir, incluir más nodos sensores de lo que se planificó en el diseño de la red. Una red escalable es aquella que puede expandir la cantidad de nodos sin afectar las necesidades de latencia, consumo energético y exactitud que se consideraron para el diseño, sabiendo que la topología de red será más compleja y el costo de implementación aumentará con respecto a la cantidad de nodos implementados.
- **Rendimiento de la red:** para esta métrica se analiza el volumen de tráfico que existe en la red, de tal forma que se puede cuantificar la cantidad de paquetes recibidos por unidad de tiempo y determinar la tasa de pérdida de paquetes.

1.2. Protocolos de Enrutamiento

Los protocolos de enrutamiento tienen la función de transmitir paquetes realizando un descubrimiento de rutas diferentes y óptimas para la entrega de datos desde el nodo origen hasta el nodo destino tratando de garantizar eficiencia energética [14], [26], [30]. También buscan mantener la comunicación entre nodos

incluso si algunas rutas se encuentran inhabilitadas [14]. El principal objetivo de los protocolos de enrutamiento consiste en desviar el tráfico desde la fuente hacia el destino minimizando el costo de la transmisión de los paquetes y mejorando el rendimiento de la red [24], [26].

El enrutamiento es considerado uno de los problemas más desafiantes en las WSN ya que se deben considerar diferentes parámetros o problemáticas para seleccionar las rutas de transmisión de los paquetes de manera eficiente [26]. Una de las problemáticas que existe en este tipo de redes surge por la gran cantidad de nodos sensores que conforman la red, por lo que al ser una red extensa no es posible realizar un esquema de direccionamiento global que involucre a todos los nodos ya que se generaría una sobrecarga del mantenimiento de la identificación de cada nodo, por lo tanto, los protocolos tradicionales basados en IP no son aplicables a las WSN [24], [26].

Debido a las diferentes aplicaciones y requisitos de las WSN se han desarrollado diferentes algoritmos que intentan resolver el problema del enrutamiento [24], [26]. Para administrar los recursos de la red especialmente la energía de los nodos, los protocolos de enrutamiento emplean algunas técnicas utilizadas en otro tipo de redes y otras diseñadas únicamente para este tipo, por ejemplo la agregación de datos, la agrupación de los nodos, la asignación de roles diferentes a cada nodo, agregación del procesamiento en la red, entre otros [24].

Los protocolos de enrutamiento en las WSN se pueden clasificar en dos grupos, de acuerdo a la estructura de la red o al criterio de encaminamiento utilizado [31]. La Figura 1.3 muestra una clasificación de los tipos de protocolos de enrutamiento para WSN. Dependiendo de la estructura de la red tenemos:

1. **Enrutamiento plano:** en este tipo de enrutamiento todos los nodos desempeñan la misma función, a los cuales se les asigna un identificador global, este proceso se lo puede considerar como una tarea no trivial [31]. Para solucionar este problema, en los protocolos se utiliza un esquema de nombres basado en atributos con la finalidad de evitar los datos redundantes que podrían enviar los nodos [26]. Algunos de los principales protocolos de enrutamiento perteneciente a este tipo son: Directed Diffusion, Sensor Protocols for Information via Negotiation (SPIN), Gradient-Based Routing (GBR), Active Query forwarding In sensor networks

(ACQUIRE) [26], [32].

2. **Enrutamiento jerárquico:** en este tipo de protocolos se tiene como objetivo agrupar los nodos para que los jefes del grupo puedan hacer la agregación y reducción de datos con el fin de ahorrar energía [31]. En estos protocolos, el nodo que contenga el máximo nivel de energía es seleccionado como jefe de grupo, (CH, del inglés Clusterhead), el cual se encarga de recopilar los datos de los miembros de su grupo o cluster para posteriormente transmitirla a la estación base con la finalidad de reducir la transmisión de información redundante [26]. Un protocolo principal de esta clasificación es LEACH en el cual los nodos se agrupan de manera jerárquica en función de su energía residual. A partir de este protocolo se han generado diversas mejoras y han surgido variantes como A-LEACH, S-LEACH, Q-LEACH, entre otros. Otro de los protocolos principales es PEGASIS el cual es una mejora al protocolo LEACH enfocada en que los nodos sensores interactúen con sus nodos sensores vecinos circundantes para generar una comunicación por multisalto hasta llegar a la estación base [26], [32].
3. **Enrutamiento basado en localización:** en este tipo de enrutamiento se utiliza la información de posición para transmitir los datos hasta las regiones deseadas de la red [31]. La ruta se selecciona utilizando la intensidad de la señal recibida (RSS), cuyo valor le permite identificar al nodo de origen que el nodo de destino está dentro de una distancia de comunicación mínima o viceversa [26]. Algunos de los principales protocolos de enrutamiento de esta clasificación son Geographic Adaptive Fidelity (GAF), Geographic and Energy-Aware Routing (GEAR), Bounded Voronoi-Greedy Forwarding (BVGF), Stateless-Protocol for End-to-End Delay (SPEED), Geographic Random Forwarding (GeRaF), Minimum Energy Communication Network (MECN) [26], [32].

Según el criterio de enrutamiento, los protocolos pueden clasificarse en:

1. **Basados en múltiples rutas:** estos protocolos de enrutamiento en lugar de tener una única ruta para la transmisión de los paquetes hacia la estación base utilizan

múltiples rutas con la finalidad de aumentar la tolerancia a fallas, por lo tanto, estos protocolos generan rutas alternativas entre el transmisor y receptor en el momento que la ruta principal falle o exista un tráfico excesivo en los enlaces de la ruta [24].

2. **Consultas:** en este tipo de enrutamiento, los nodos generan una consulta de datos a un nodo concentrador, el cual se encarga de pedir la información requerida al nodo que la contenga y generar la ruta de comunicación entre los dos nodos. En este caso, todos los nodos cuentan con tablas que contienen cada una de las consultas y los datos que deben enviar como respuesta a cada una de ellas [24].
3. **Negociación:** estos protocolos utilizan descriptores de alto nivel con la finalidad de eliminar las transmisiones de datos redundantes mediante la negociación. Las transmisiones redundantes consumen más energía y aumentan el procesamiento por parte de los nodos para realizar la transmisión de la misma información. Por lo tanto, la idea principal de este grupo de enrutamientos es eliminar la información duplicada mediante mensajes de negociación entre el nodo transmisor y su próximo salto [24].
4. **Calidad de servicio (QoS):** estos protocolos de enrutamiento se encargan de equilibrar el consumo energético de los nodos y mejorar la calidad de la recepción de los datos, es decir, la red debe satisfacer ciertos parámetros como reducir la latencia, el ancho de banda, etc., para otorgar una buena calidad de servicio [24].
5. **Basados en coherencia:** en este tipo de enrutamiento los datos son procesados con anterioridad a ser enviados, el nivel de procesamiento es mínimo y se enfoca principalmente en la supresión de duplicados. Mediante estos protocolos de enrutamiento se puede generar una eficiencia energética ya que se puede realizar procesos de compresión de datos de tal manera que se añada al paquete la información del nodo inferior en la jerarquía de la red [24].

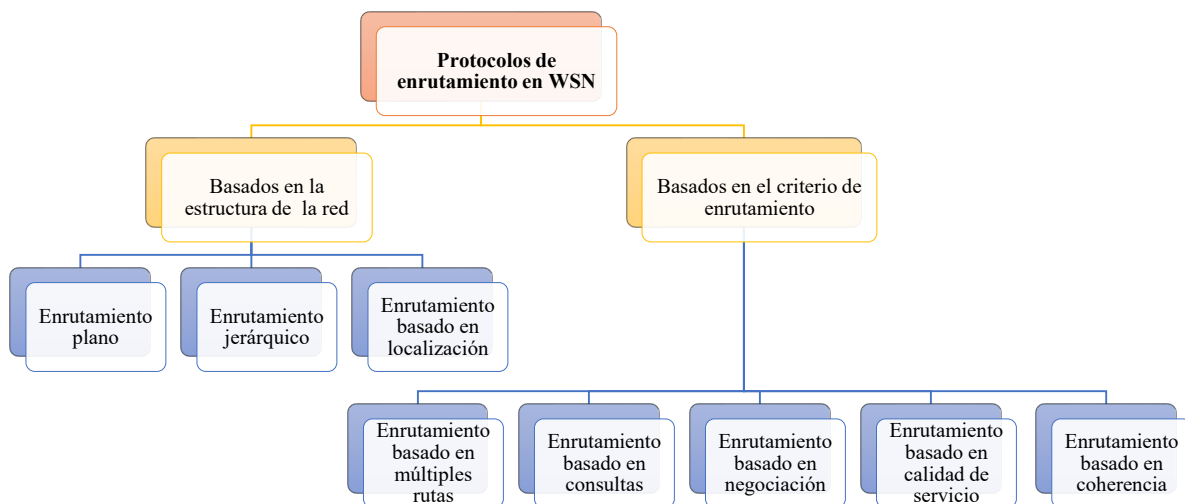


Figura 1.3: Clasificación de los protocolos de enrutamiento para WSN.

1.3. Dimensionamiento de Redes

El dimensionamiento de las redes, permite establecer una ubicación óptima de los sitios candidatos para los nodos de la red, lo cual ayuda a disminuir los costos involucrados en el despliegue de la infraestructura de la red, mejorar el rendimiento y eficiencia de la red, y permitir una escalabilidad para abarcar más usuarios. Los algoritmos de dimensionamiento se enfocan en generar soluciones heurísticas sub-óptimas donde consideran diversos parámetros de la red como la cobertura, capacidad, interferencia y número de canales [13]. A continuación se presenta en que consiste cada uno de los parámetros indicados anteriormente:

- **Cobertura:** El dimensionamiento de las redes, permite establecer una ubicación óptima de los sitios candidatos para los nodos de la red, lo cual ayuda a disminuir los costos involucrados en el despliegue de la infraestructura de la red, mejorar el rendimiento y eficiencia de la red, y permitir una escalabilidad para abarcar más usuarios. Los algoritmos de dimensionamiento se enfocan en generar soluciones heurísticas sub-óptimas donde consideran diversos parámetros de la red como la cobertura, capacidad, interferencia, número de canales, entre otros [22, 23], [33, 34].

- **Capacidad:** La capacidad hace referencia a la cantidad de enlaces que el concentrador puede soportar en la región establecida, para esto se busca que los usuarios sean atendidos al menos por un concentrador y la cantidad de usuarios atendidos supere el porcentaje de cobertura preestablecidos. En este criterio se debe considerar un porcentaje de crecimiento de los usuarios para poder aumentar la cantidad de enlaces [22, 23], [33, 34].
- **Interferencia:** El criterio de interferencia indica la manera de gestionar el servicio a un usuario que se encuentra en una zona donde más de un concentrador puede dar cobertura. Este criterio se fundamenta en la cantidad de usuarios que se conectan a cada uno de los concentradores y la densidad de los usuarios en los radios de cobertura de cada concentrador [33, 34].
- **Número de canales:** Al realizar las transmisiones por el espectro radioeléctrico, se encuentra vulnerable a la interferencia en la comunicación al momento de transmitir la información desde diferentes fuentes en el mismo rango de frecuencia. Para disminuir los niveles de interferencia en los usuarios, el número de canales permite dimensionar y distribuir las frecuencias de operación de los concentradores para que los concentradores adyacentes transmitan y reciban la información a frecuencias diferentes. Con este criterio, se realiza el proceso de reutilización de frecuencias con la finalidad de utilizar un rango de frecuencias del espectro radioeléctrico [33, 34].

1.4. ZigBee

ZigBee es un estándar que define un conjunto de protocolos de comunicación para redes inalámbricas de corto alcance y baja velocidad de datos. Los dispositivos inalámbricos basados en ZigBee funcionan en bandas de frecuencia de 868 MHz, 915 MHz y 2,4 GHz. La velocidad máxima de datos es de 250 Kbps. ZigBee está diseñado principalmente para aplicaciones que funcionan con baterías donde los requisitos principales son la baja velocidad de datos, el bajo costo y la duración de la batería. En muchas aplicaciones de ZigBee, el tiempo total que el dispositivo inalámbrico está

involucrado en cualquier tipo de actividad es muy limitado; el dispositivo pasa la mayor parte del tiempo en un modo de ahorro de energía, también conocido como modo de suspensión [17], [35].

A las redes ZigBee se las conoce como redes de área personal (PAN), a las cuales se las define con un identificador PAN (PAN ID), que es común entre todos los dispositivos pertenecientes a la red. Para esto, cada dispositivo debe estar preconfigurado con un PAN ID o en su defecto puede realizar un descubrimiento de las redes cercanas y unirse a una de ellas. ZigBee admite que el PAN ID sea tanto de 64 bits como de 16 bits, en donde el PAN ID de 16 bits es utilizado como campo de direccionamiento de la capa MAC en todas las transmisiones de paquetes entre dispositivos de la misma red, mientras que el PAN ID de 64 bits se utiliza para evitar la semejanza del PAN ID de 16 bits entre dos redes ZigBee, ya que se tiene una mayor cantidad de redes disponibles [18].

Los protocolos de red dividen los diferentes componentes y funciones en capas que se pueden relacionar de diferentes maneras. En el estándar Zigbee la división de capas se realiza en cuatro, la primera es la capa física (PHY) y la segunda es la capa de control de acceso medio (MAC) las cuales son descritas en el estándar IEEE 802.15.4 [18]. En estas capas se manejan las operaciones de red como el direccionamiento, la transmisión y recepción de los paquetes, estas operaciones son consideradas de bajo nivel. La tercera capa es conocida como la capa de red (NWK) y la cuarta es la capa de aplicación (APL); en la capa de red se maneja los conceptos de topología, enrutamiento y seguridad de la red, mientras que en la capa de aplicación se maneja las aplicaciones definidas por el usuarios para que cada uno de los nodos de la red tengan su funcionalidad específica [18]. La Figura 1.4 presenta un resumen de las capas del estándar Zigbee.

Una de las relaciones existentes entre la capa de red y la capa de control de acceso al medio es durante el direccionamiento de los paquetes en donde ZigBee utiliza dos tipos de direcciones, la primera es una dirección de 64 bits la cual es única por cada dispositivo físico [18]. Esta dirección es también conocida como MAC y es asignada en el proceso de fabricación del equipo. Los primeros tres bytes representan un identificador único organizativo (OUI, del inglés Organizationally Unique Identifier)

asignado al fabricante por el Instituto de Ingenieros Eléctricos y Electrónicos (IEEE). Mientras que la dirección de 16 bits es asignada al momento que un dispositivo se une a una red ZigBee siendo la dirección 0x0000 la reservada para el coordinador. Todas la transmisiones se envían utilizando las direcciones de 16 bits de origen y destino [18].

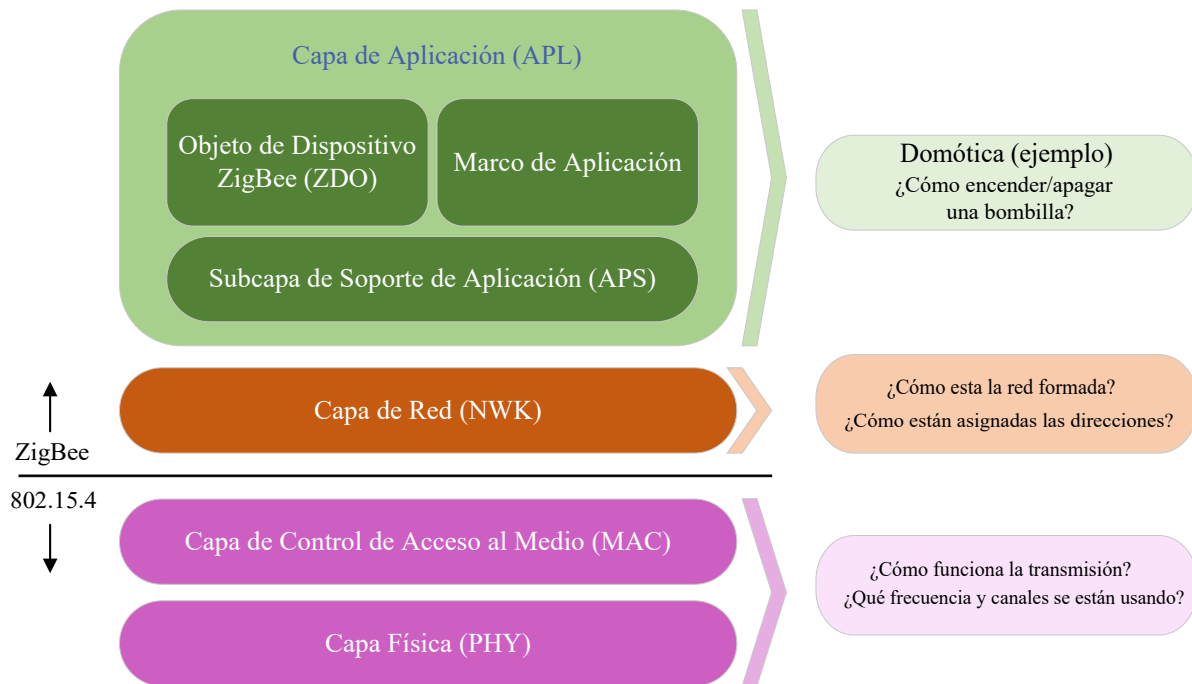


Figura 1.4: Capas del estándar Zigbee

En una red ZigBee, existen tres tipos de nodos los cuales realizan diferentes funciones en la red. Los tipos de nodos son los siguientes:

1. **Coordinador:** el coordinador es el nodo encargado de generar la red dependiendo el ID de la misma, es decir, su función es la de establecer el canal operativo entre todos los nodos para que estos puedan realizar la transmisión de sus datos. Luego de realizar este proceso, el coordinador puede funcionar como enrutador y a su vez recibir paquetes desde el resto de nodos de la red. Cabe indicar que solo puede existir un nodo concentrador por cada ID de la red PAN [18].
2. **Enrutador:** el nodo enrutador (en inglés Router), tiene la función de establecer la mejor ruta para la transmisión de paquetes, para esto estos nodos crean y

mantienen información referente a los enlaces entre los nodos de la red [18].

3. **Dispositivo Final:** los dispositivos finales son considerados las fuentes de transmisión y recepción de los paquetes, es decir, la función de estos nodos es generar tramas con la información censada por ese nodo para posteriormente transmitirla a otro dispositivo final de la red. Estos nodos no participan en el proceso de enrutamiento por el hecho de que permanecen por periodos de tiempo inactivos con la finalidad de disminuir el consumo energético para prolongar el tiempo de vida del nodo y a su vez aumentar la cantidad de paquetes transmitidos [18].

Para que todos los tipos de nodos entiendan la información recibida, estos deben operar dentro de los tres modos de funcionamiento. El primero es el modo Transparente, donde los datos a transmitir son enviados sin ser incluidos en una trama, es decir, los datos son transmitidos sin la necesidad de agregarle una cabecera o cola. Para el identificar el nodo receptor, se necesita únicamente su dirección **MAC**. El segundo modo de operación es el modo API en donde los datos a transmitir son encapsulados en una trama con la finalidad de detectar posibles modificaciones ejercidas por el canal. Las tramas para transmisión y recepción son las siguientes de acuerdo con [18]:

- **Solicitud de Transmisión (0x10):** este tipo de trama se utiliza para enviar datos de carga útil hacia un destino específico. La trama esta conformada de acuerdo a lo que se detalla en la Tabla 1.1.
- **Recepción de paquete (0x90):** esta trama se genera en consecuencia de la recepción de la solicitud de transmisión (0x10). La trama esta conformada de acuerdo a lo que se detalla en la Tabla 1.2.

Tabla 1.1: Composición de trama para transmisión - Solicitud de transmisión (0x10)

Posición	Tamaño	Campo de la Trama	Descripción
0	8-bits	Delimitador de Inicio	Indica el inicio de un marco API.
1	16-bits	Longitud	Número de bytes entre la longitud y la suma de comprobación.
3	8-bits	Tipo de Trama	Solicitud de transmisión - 0x10.
4	8-bits	ID de la Trama	Identifica la trama de datos para que el host se correlacione con una trama de respuesta posterior.
5	64-bits	Dirección de 64-bits de Destino	Dirección IEEE de 64-bits del dispositivo de destino.
13	16-bits	Dirección de 16-bits de Destino	Dirección de red de 16-bits del dispositivo de destino.
15	8-bits	Radio de Broadcast	Establece el número máximo de saltos en una transmisión por difusión.
16	8-bits	Opciones de Transmisión	
17-n	variable	Carga Útil (Datos)	Datos a enviar al dispositivo de destino.
EOF	8-bits	Suma de Comprobación (Checksum)	0xFF menos la suma de bytes de 8-bits desde la posición 3 de la trama hasta este byte (entre la longitud y la suma de comprobación).

Tabla 1.2: Composición de trama para recepción - Recibir paquete (0x90)

Posición	Tamaño	Campo de la Trama	Descripción
0	8-bits	Delimitador de Inicio	Indica el inicio de un marco API.
1	16-bits	Longitud	Número de bytes entre la longitud y la suma de comprobación.
3	8-bits	Tipo de Trama	Recibir Paquete - 0x90.
4	8-bits	Dirección de 64-bits del Transmisor	Dirección de 64-bits del dispositivo transmisor.
12	64-bits	Dirección de 16-bits del Transmisor	Dirección de red de 16-bits del dispositivo transmisor.
14	16-bits	Opciones de Recepción	
15-n	8-bits	Datos Recibidos	Carga útil recibida.
EOF	8-bits	Suma de Comprobación (Checksum)	0xFF menos la suma de bytes de 8-bits desde la posición 3 de la trama hasta este byte (entre la longitud y la suma de comprobación).

Capítulo 2

Problema de estudio

Con lo mencionado en el Capítulo 1, en WSN el problema más desafiante es el enrutamiento de los paquetes [12], porque existen diversas problemáticas o parámetros que se deben considerar para realizar una transmisión eficiente reduciendo el consumo energético y el procesamiento por parte de los nodos de la red [13]. Es por esto por lo que se han desarrollado diversos algoritmos que buscan emplear diversas técnicas para aumentar la vida útil de la red a su vez que mejoran la eficiencia energética de los nodos, disminuyen la latencia y reducen la tasa de pérdida de paquetes.

Por otro lado, en el proceso del diseño de redes se ha visto imprescindible realizar un dimensionamiento para un óptimo despliegue de la red de acuerdo con la aplicación que se le da a la red. Con un correcto diseño, dimensionamiento y despliegue se consigue reducir costos de instalación, mejorar la eficiencia y rendimiento de la red garantizando una escalabilidad de esta [14], [22].

La implementación de una WSN aplicando modelos de dimensionamiento óptimo puede traer mejoras en el enrutamiento de los paquetes, ya que, al involucrar la capacidad, cobertura, interferencia y número de canales para la ubicación de los nodos, se puede optimizar la cantidad de enlaces y disminuir el consumo energético, así como el procesamiento computacional en cada nodo.

Dicho lo anterior, en el presente trabajo de titulación se realiza el estudio de los protocolos de enrutamiento LEACH, PEGASIS, Dijkstra y Prim en una red desplegada sin criterios de dimensionamiento y en otra aplicando los modelos matemáticos de cobertura y capacidad en cuyos escenarios se analiza las métricas de latencia y

rendimiento (throughput) de tal forma que se pueda determinar la influencia del dimensionamiento de red en los protocolos de enrutamiento.

Para este estudio se a generado una WSN fija con tecnología ZigBee debido a que permite el uso de enrutadores PAN y que, al estar soportada por el estándar IEEE 802.15.4, la frecuencia de trabajo corresponde a la banda de radio industriales, científicas y médicas (ISM) por lo que el uso del espectro radio eléctrico es libre. Además, es una tecnología que presenta una solución de ahorro de energía de bajo costo que incluye prevención de colisiones y permite formar redes con una gran cantidad de nodos [15], [17].

Por lo tanto, el presente trabajo pretende evaluar esta combinación del dimensionamiento óptimo de WSN con diversos protocolos de enrutamiento y así contrastar, a través de una implementación, los resultados obtenidos en ambos escenarios. En el presente capítulo se explica la metodología utilizada para el diseño de la red, la programación de los protocolos de enrutamiento y la aplicación de los modelos matemáticos para el dimensionamiento de la red.

2.1. WSN Implementada

Para el diseño de la red se consideraron 10 nodos, de los cuales uno sería el coordinador de la red ZigBee y por lo tanto sería considerado como la estación base. El resto de nodos tendrán la función de enrutadores para que sean capaces de direccionar el tráfico de los paquetes de acuerdo al protocolo de enrutamiento implementado en la red.

Estos nodos fueron desplegados por el campus universitario de la Universidad Politécnica Salesiana sede Cuenca en los puntos georreferenciados descritos en la Tabla 2.1. En la Figura 2.1 se muestra su ubicación desde una vista satelital.

Tabla 2.1: Puntos Georreferenciados de los nodos de la red.

Nodos	Coordenadas	
	Latitud	Longitud
1	-2,886998	-78,990127
2	-2,886958	-78,990198
3	-2,886268	-78,990986
4	-2,886887	-78,989327
5	-2,885932	-78,990657
6	-2,886229	-78,990059
7	-2,886776	-78,990125
8	-2,887240	-78,990000
9	-2,886895	-78,989738
10	-2,886761	-78,990411

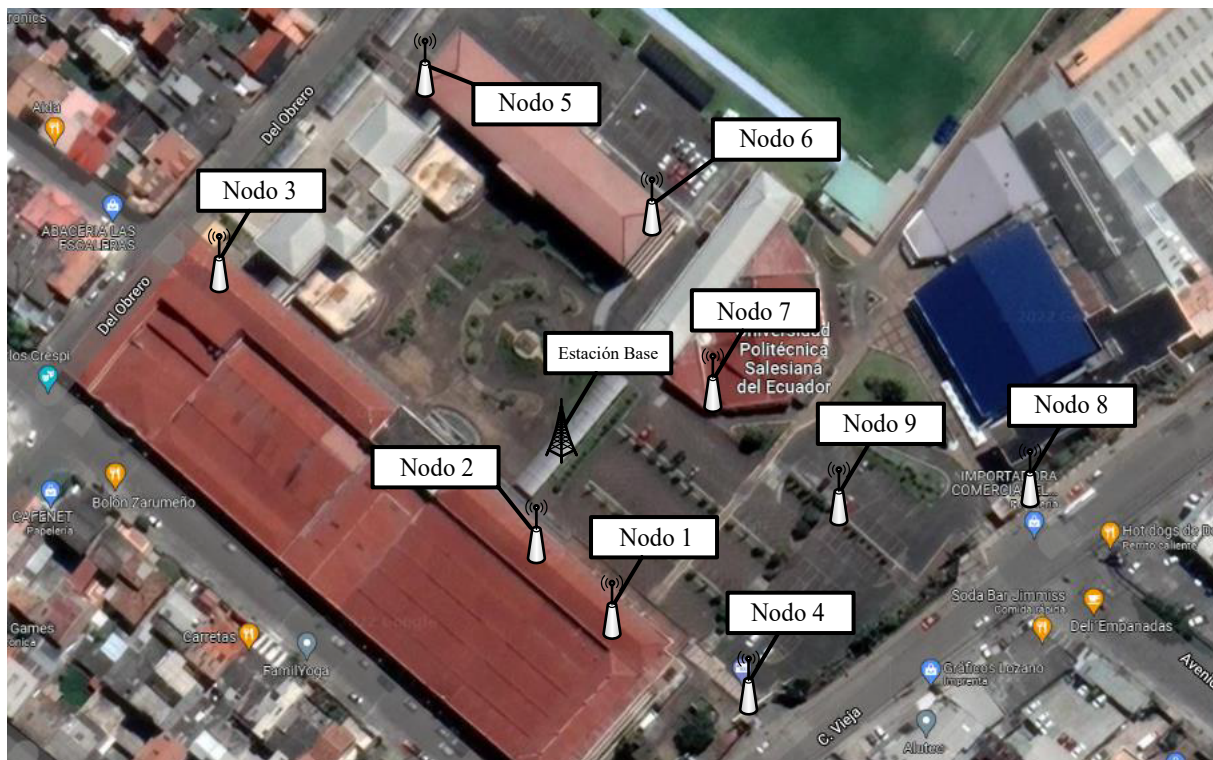


Figura 2.1: Ubicación de los nodos en el campus universitario.

El módulo ZigBee utilizado en cada nodo de esta red es el XBEE Click de MikroE que utiliza un dispositivo XBEE S2C ya que permite manejar el modo API de ZigBee y permite un rango de alcance de 60m en interiores y hasta 1200m en exteriores con línea de vista absoluta y tasa de transmisión que alcanza los 250,000 Kbps. La Figura 2.2 presenta el módulo XBEE Click usado en cada nodo de la red implementada; por

supuesto, cada uno tiene su propia dirección **MAC**.

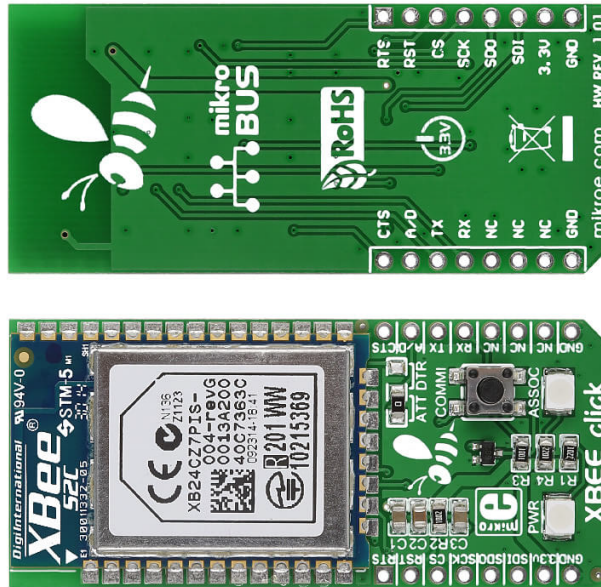


Figura 2.2: Módulo XBEE Click.

En lo que respecta a la inteligencia para el cálculo uso de los algoritmos de enrutamiento implementados, cada nodo estaba compuesto por su microcontrolador capaz de implementar los algoritmos de enrutamiento evaluados. Así, nodo concentrador que al mismo tiempo operó como una pasarela (*gateway*) de datos para ser registrados en un computador fue implementado con un microcontrolador STM32F407VG, los nodos 1 y 2 usaron del microcontrolador STM32F103C8T6, los nodos 3 y 4 con un STM32F401CxUx y el resto de nodos se implementaron con Arduino UNO. La razón de uso de estos microcontroladores ha sido debido a la facilidad de conseguirlos en el mercado y disposición de los mismos. Debe mencionarse que cada nodo fue ubicado en una caja industrial para aplicaciones **IoT** con nivel de protección IP67 permitiendo colocar los nodos al aire libre.

En cuanto a la alimentación de voltaje, el caso de todos los nodos que usaron los microcontroladores STM32 se conectaron con un *power bank* o batería de litio y para los nodos implementados con Arduino UNO se usó una batería de 9V. Las Figuras 2.3 y 2.4 se resume la conexión de los componentes de los nodos sensores usados en la implementación de este trabajo.

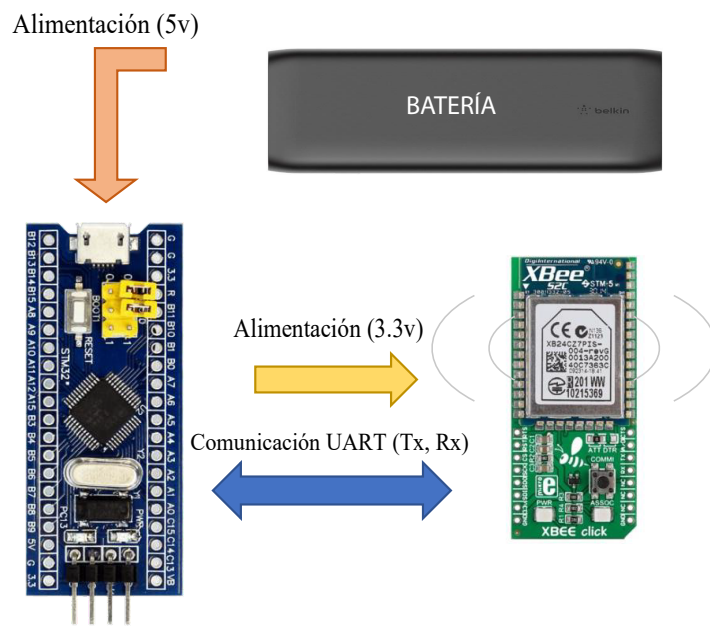


Figura 2.3: Esquemas de conexión de los nodos sensores usando un microcontrolador STM32F103C8T6 y STM32F401CxUx

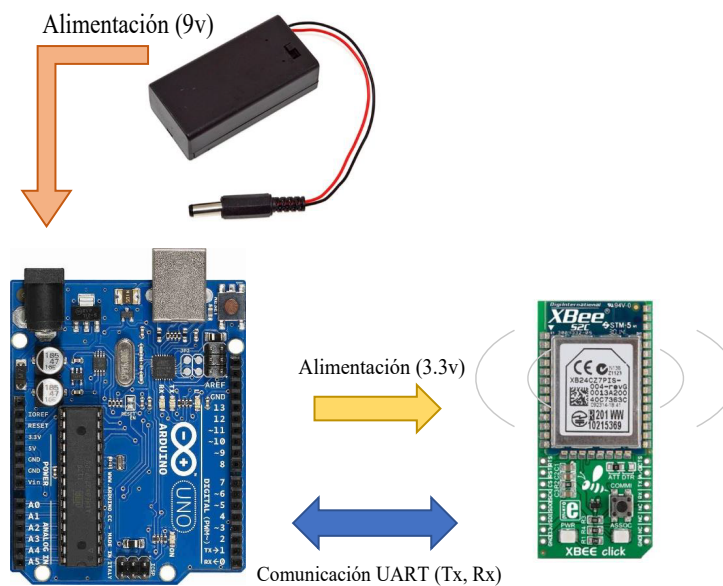


Figura 2.4: Esquemas de conexión de los nodos sensores usando Arduino UNO

Debido a que ZigBee usa un nodo coordinador que es el nodo al cual llega la información de cada uno de los otros nodos de la red, el gateway usa entonces doble tecnología, una tarjeta de comunicación XBEE, y un módulo Ethernet para poder transmitir los datos hacia un computador. La transferencia de datos hacia el

computador es a través de mensajes con formato *sockets* de acuerdo al protocolo de datagramas de usuario (**UDP**, del inglés User Datagram Protocol). Ya que el enfoque del trabajo no es el desarrollo de la plataforma de monitoreo sino analizar los datos que circulan a través de la red, el monitoreo y registro de datos transmitidos a través de la red ha sido implementado mediante LabView debido a la facilidad que se tenía para la captura de datos del puerto Ethernet. Aquí los datos se almacenan en un archivo con extensión *.xls* para posteriormente analizar los datos recibidos. El módulo ETH WIZ Click es el ocupado para esta operación y en la Figura 2.5 se presenta este módulo. La Figura 2.6 presenta el uso del microcontrolador STM32F407VG conectándose con las tarjetas de comunicación a través de una tarjeta de expansión.

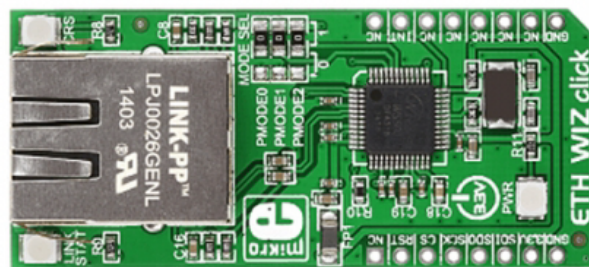


Figura 2.5: Módulo ETH WIZ Click.

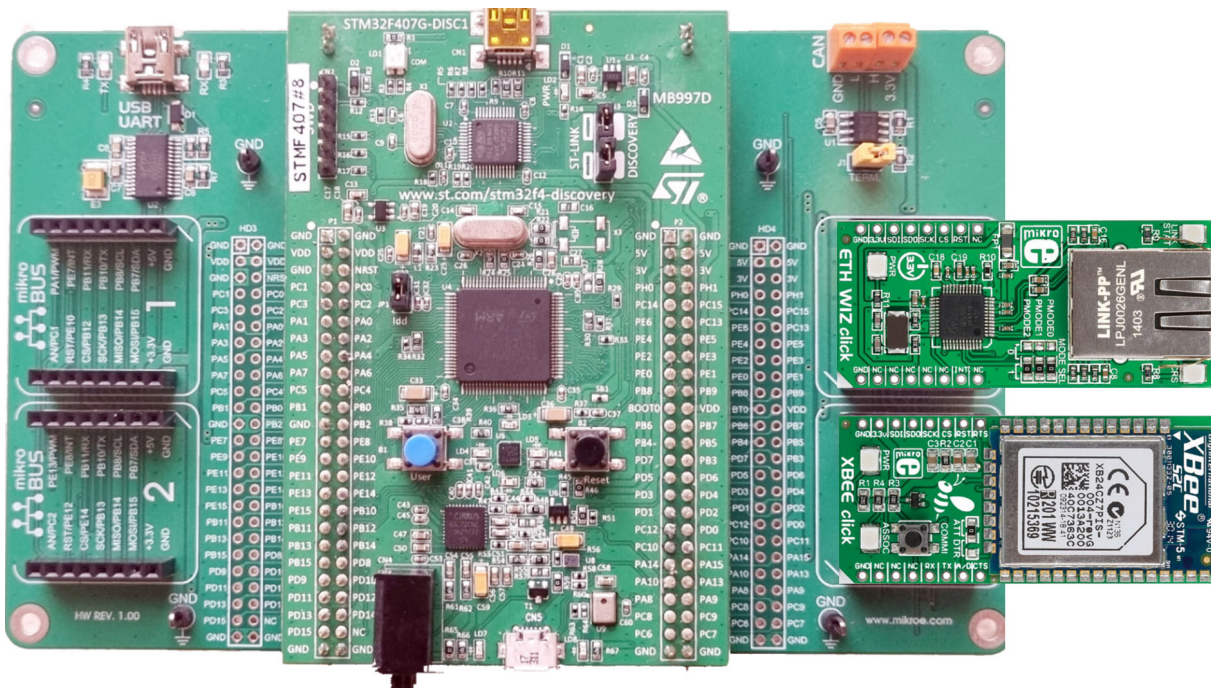
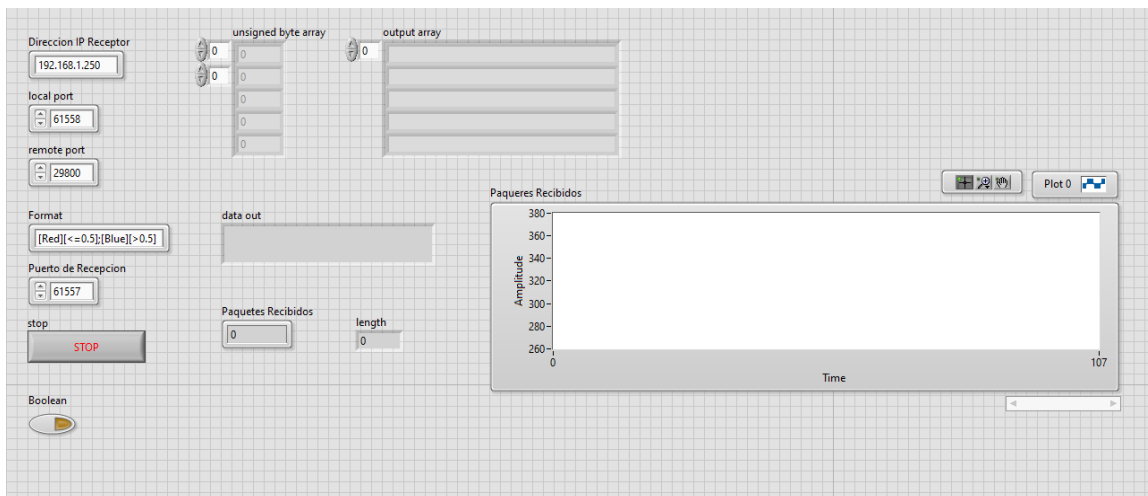
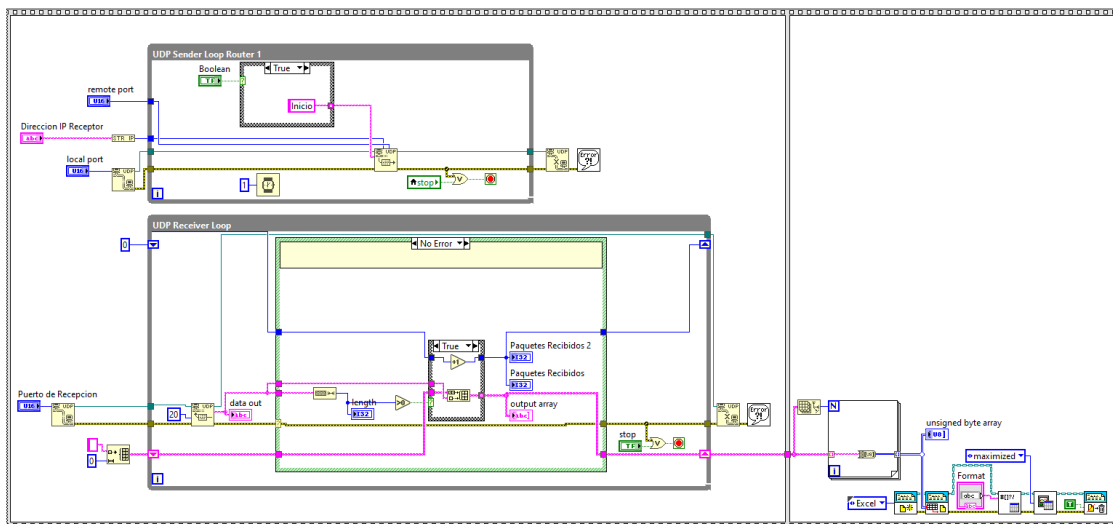


Figura 2.6: Nodo Coordinador/Concentrador: Pasarela de transferencia de datos.

En el sistema de monitoreo se realiza un conteo de los paquetes recibidos y cada paquete es almacenado en un vector para que al momento de la finalización de la prueba se realice la respectiva conversión a un documento como se explicó más arriba. Para dar inicio al proceso de recepción, se debe enviar un comando desde el computador de monitoreo por Ethernet para que el coordinador habilite el puerto UART en donde está conectado su módulo XBEE. La Figura 2.7(a) presenta el panel frontal y la Figura 2.7(b) muestra el diagrama de bloques del programa desarrollado en LabView.



(a) Panel Frontal del Programa de Monitoreo



(b) Diagrama de Bloques del Programa de Monitoreo

Figura 2.7: Programación en LabView para monitoreo de datos recibidos del coordinador.

2.2. Protocolos de Enrutamiento

Para la programación de los protocolos de enrutamiento se realizó un estudio previo de los enlaces existentes entre cada uno de los nodos, es decir, se determinó la línea de vista que tiene cada nodo para según esto establecer las funciones de cada nodo. En base a este estudio se determinó una matriz de conectividad G . La ecuación 2.1 es la matriz de conectividad usada en el presente trabajo de manera que la diagonal principal mantiene 0 para asegurarse que no existan lazos ya que por ejemplo la posición $G_{1,1}$ significa la conectividad del nodo 1 con el nodo 1. El lector puede apreciar en la ecuación 2.1 que todos los elementos de la parte triangular superior de G es un espejo de todos los elementos de la parte inferior, por ejemplo el elemento $G_{3,1}$ tiene el mismo valor de $G_{1,3}$.

$$\mathbf{G} = \begin{bmatrix} 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 0 \end{bmatrix} \quad (2.1)$$

Esta matriz de conectividad ha sido construida a partir de la verificación del radio máximo de cobertura que la tecnología nos brinda considerando que no todas las ubicaciones brinda una línea de vista adecuada y que por el escenario físico del campus universitario donde se desplegaron los nodos, ciertos nodos no se podían “ver” entre sí, en cuyo caso, el elemento correspondiente a la conectividad de un par de nodos se marca con 0 y con 1 para el caso de que se compruebe que exista conectividad entre estos. Cabe mencionar que los algoritmos de cálculo de rutas como Prim o Dijkstra usan una matriz de distancias para estimar la mejor ruta, no obstante, al multiplicar

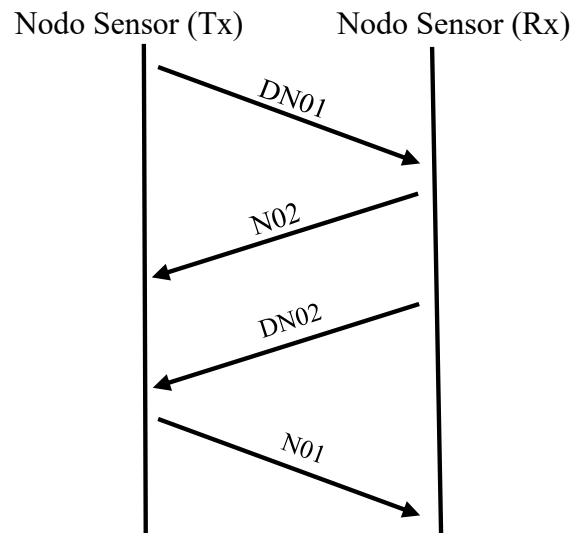
por la matriz de conectividad, es posible eliminar rutas que en la práctica no se pueden dar y estos algoritmos solo usen aquellas rutas que son factibles.

Entonces, a través de esta matriz de conectividad, se determina que una topología en malla será muy complicada de realizar porque muchos nodos comparten un único enlace con los demás nodos sensores. Como solución se determinó utilizar una topología en árbol o jerárquica de tal manera que todos los nodos puedan tener un enlace con un nodo padre que este a su vez cuenta con línea de vista hacia el concentrador.

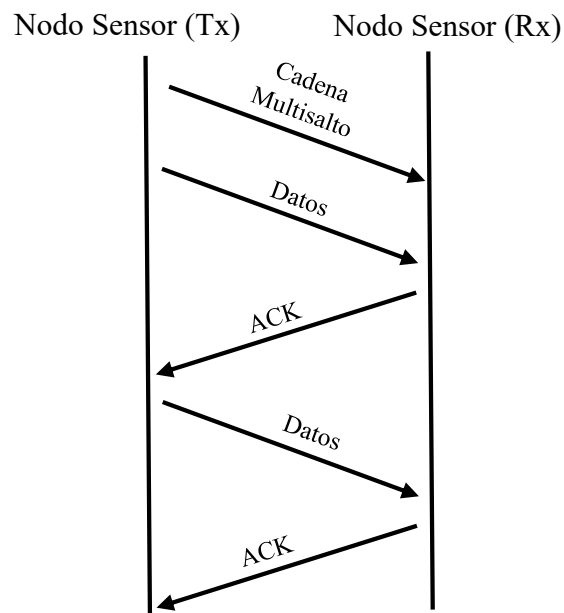
Convencionalmente los protocolos de enrutamiento en las WSN están compuestos por dos fases, la primera es de configuración en la cual se realiza un descubrimiento de los nodos de la red para establecer los enlaces entre cada nodo y organizar los grupos en caso de ser un protocolo de enrutamiento jerárquico y se logra a través de la transmisión de mensajes por difusión (*broadcast*) tal como se resume en la Figura 2.8(a); la segunda fase es conocida como fase de estado estable que es cuando ocurren cada una de las transmisiones de paquetes desde cada nodo hacia la estación base y se resumen en la Figura 2.8(b) [36]. En general, la Figura 2.8 presenta los paquetes con los mensajes necesarios para establecer la configuración y el estado estable. Cabe indicar que estas fases se repiten cada cierto tiempo o cuando se detecta la pérdida de conexión de algún nodo siendo necesario la reestructuración de la ruta de transmisión de los paquetes.

Entonces, al programar los protocolos de enrutamiento analizados y descritos en las siguientes subsecciones de este capítulo, se estableció que el proceso de descubrimiento se realice de la misma manera, este proceso consiste en que cada nodo sensor contará con una ventana de tiempo para realizar su descubrimiento, para conseguir esto se configuró un temporizador en cada nodo para establecer los tiempos de cada proceso, a la variable que almacena los conteos de este temporizador se la denominó *Tiempo*. Además se creó una variable booleana *Tx* para determinar si ese nodo ya realizó este proceso y no se ejecute de manera repetitiva. Para finalizar con las variables de entrada, en un vector denominado *EnlacesNodos* con una longitud de 10 posiciones en donde cada posición de ese vector representa un nodo de la red, se almacena los posibles enlaces de transmisión con los que cuenta ese nodo, si existe línea

de vista entre el nodo transmisor y el nodo que recibió su paquete de descubrimiento se colocará el valor de 1 en su posición correspondida.



(a) Comunicación entre nodos para descubrimiento por Broadcast



(b) Comunicación entre nodos para transmisión de datos

Figura 2.8: Comunicación entre nodos en la fase de configuración y estado estable.

Dentro del tiempo de descubrimiento, cada nodo podrá transmitir su paquete de descubrimiento por broadcast pasado un tiempo de 100 segundos después del

inicio de funcionamiento. Posteriormente en un periodo de 150 segundos el nodo estará a la espera de recibir una contestación de los demás nodos. Los nodos que ya hayan realizado su descubrimiento o aún no cumplen con el tiempo de guarda para su inicio deberán permanecer en constante monitoreo del puerto UART para detectar cualquier mensaje de descubrimiento de parte de los otros nodos sensores y enviar su contestación. La Figura 2.9 muestra el diagrama de flujo de este proceso.

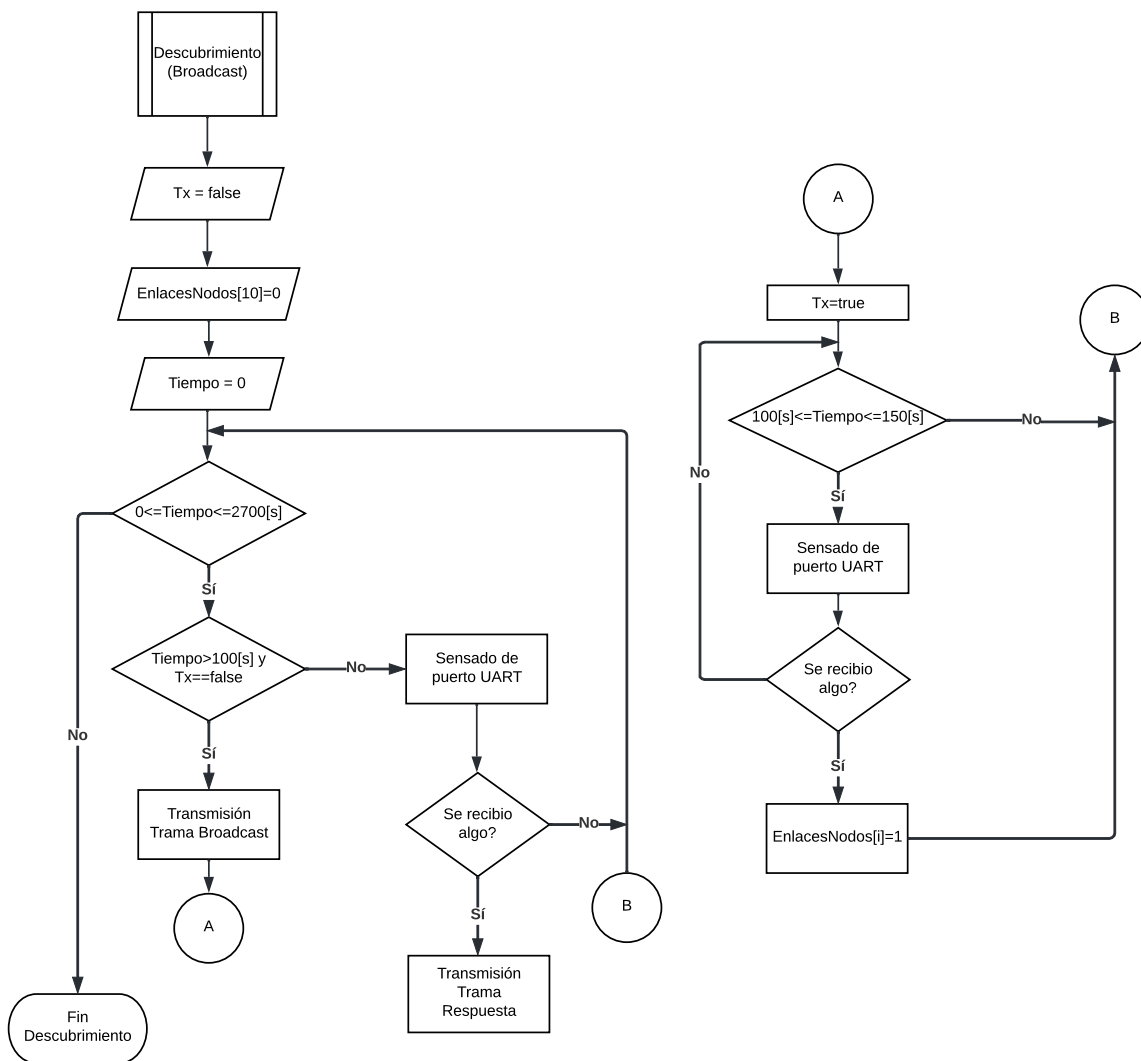


Figura 2.9: Diagrama de flujo del proceso de descubrimiento de los nodos sensores.

Para conocer los nodos pertenecientes a la cadena de transmisión por multsalto, los nodos generan un vector donde se almacena los nodos por los que a transcurrido ese paquete, de tal forma que para el cálculo del siguiente salto no se repita

ningún nodo perteneciente a la cadena. Cada nodo sensor previo a su transmisión de información deberá transmitir esta cadena a su siguiente salto. Por lo tanto, el cálculo del próximo salto se realiza de manera descentralizada, con esto garantizamos que todos los nodos decidan la mejor ruta considerando los enlaces existentes en ese momento. Si bien es cierto que al realizar el cálculo del siguiente salto de manera descentralizada aumentamos el procesamiento en los nodos sensores y a su vez incrementamos el consumo energético de los mismos, pero en la otra mano disminuimos el riesgo de desconocimiento de los nodos pertenecientes a la cadena de multisalto que estos pertenecen a la misma, ya que por el hecho de que el coordinador no tiene un enlace directo a todos los nodos no se podrá hacer una difusión por broadcast de los nodos pertenecientes a la cadena multisalto [37].

Los nodos receptores del paquete de descubrimiento responden con una trama indicando el número de nodo que son de tal forma que el nodo que está realizando el descubrimiento puede incluir a ese nodo en su tabla de enrutamiento. Por otro lado en el proceso de transmisión de los datos, se a considerado que cada nodo tenga almacenado variables de voltajes previamente capturadas en el proyecto de Smart Metering de la Universidad Politécnica Salesiana sede Cuenca. Por lo tanto, la carga útil de la trama esta compuesta inicialmente por una etiqueta de la longitud de los datos para posteriormente abstraer esa cantidad de bytes en el coordinador y transmitir por Ethernet únicamente los datos y no la trama completa.

Además, cada nodo receptor añade a la trama recibida su tiempo de procesamiento desde el momento que recibió el paquete hasta retransmitirlo, la trama queda estructurada como se muestra en la Figura 2.10. Cabe indicar que el coordinador no realiza ninguna contestación de los procesos descritos anteriormente, de tal forma que todos los nodos tienen conciencia de los enlaces directos hacia el coordinador.

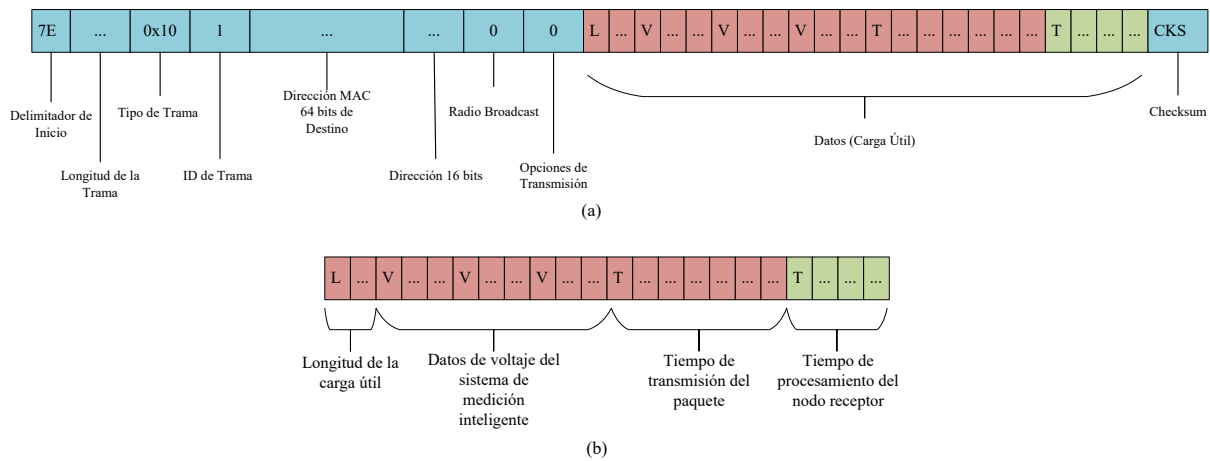


Figura 2.10: Conformación de la trama para la transmisión de los datos.

2.2.1. Dijkstra

El algoritmo Dijkstra fue diseñado para resolver el problema de los caminos más cortos de fuente única. En este algoritmo se busca determinar el camino más corto entre dos nodos sin recorrer dos veces el mismo nodo. Para la decisión de que camino tomar, Dijkstra compara las distancias entre el nodo transmisor y todos sus posibles receptores y selecciona la menor como mejor opción. Este proceso se repite en cada nodo hasta que el paquete transmitido llegue a su destino [38]. Este proceso de cálculo del camino más corto se presenta en la Figura 2.11 donde cada nodo toma la decisión de transmitir al nodo que tiene menor distancia.

Para la implementación del algoritmo Dijkstra en un nodo sensor se establecieron las dos fases descritas al inicio de este capítulo, en donde el subproceso de descubrimiento por broadcast se presentó previamente en la Figura 2.9 como diagrama de flujo. Después de realizar la fase de configuración, se establece una multiplexación TDM, en donde cada nodo sensor tiene una ventana de tiempo para realizar la transmisión de sus datos. En este protocolo se tienen como variables de entrada un vector de distancias entre cada nodo denominado *Dist* y una variable booleana *Tx* para asegurar que los nodos solo tengan un espacio para la transmisión de sus datos. Al igual que el proceso de descubrimiento, después de 100 segundos cada nodo empezará a realizar su transmisión de datos, para lo cual debe aplicar el algoritmo Dijkstra y determinar su próximo salto considerando el vector *Dist*.

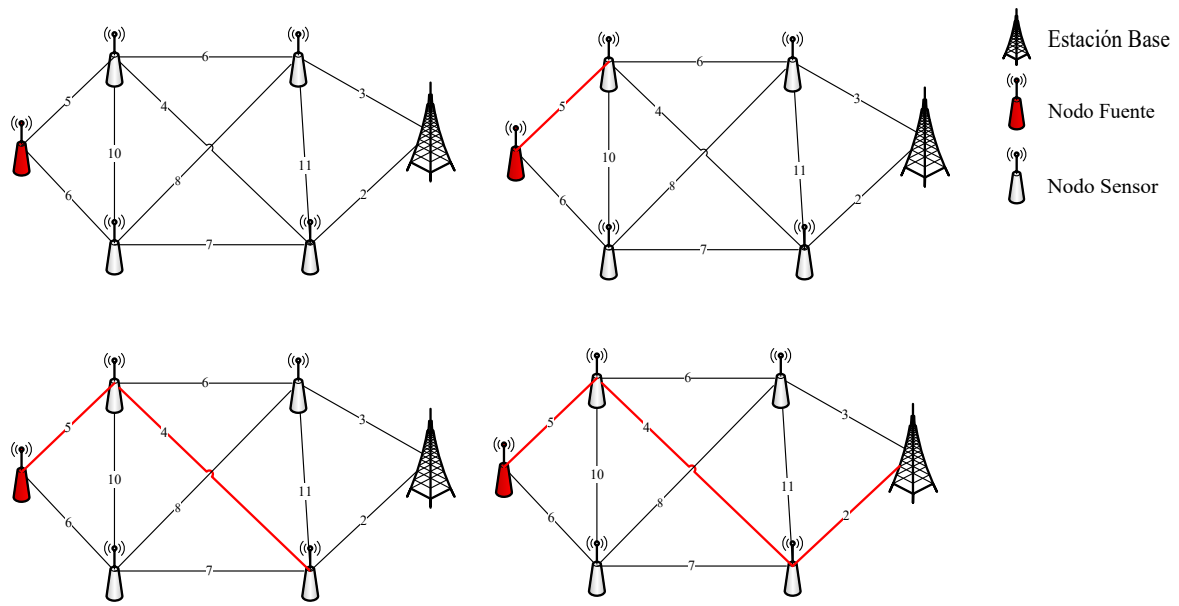


Figura 2.11: Ejemplo de ejecución del algoritmo Dijkstra.

La Figura 2.12 se presenta un diagrama de flujo con el funcionamiento del algoritmo Dijkstra en el cual en primera instancia se establece una distancia muy alta para los nodos que no se tiene línea de vista, esta modificación de distancias se realiza a través del vector *EnlacesNodos* inicializado en el subproceso de descubrimiento. A continuación se compara las distancias de los nodos con enlaces activos y a la par se detecta si en la cadena multisalto se encuentra el nodo analizado, si el nodo candidato a ser receptor ya pertenece a la cadena de multisalto significa que ese nodo fue una solución de un nodo anterior, por lo que ese nodo no puede ser recorrido dos veces. Una vez comparadas todas las distancias se guarda la menor en la variable *DistNH* y el próximo salto en la variable *NextHope*.

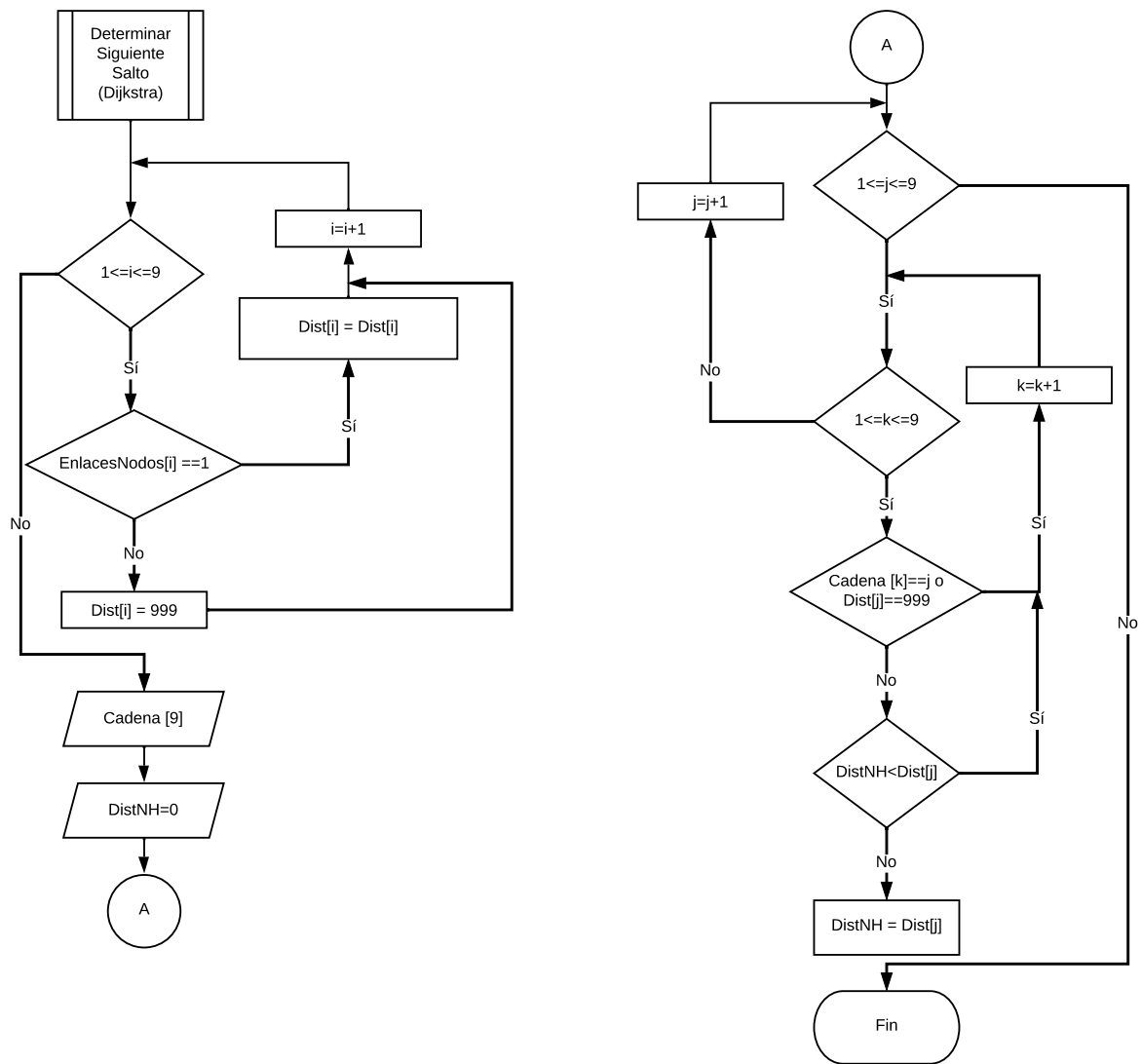


Figura 2.12: Diagrama de Flujo del algoritmo Dijkstra.

Una vez finalizado este proceso el nodo sensor empieza a transmitir sus datos al siguiente nodo como se presentó en la Figura 2.8. Los demás nodos permanecerán en constante monitoreo del puerto UART para detectar cualquier paquete proveniente de los demás nodos. En caso de recibir la cadena de multisalto, significa que ese nodo pertenecerá a la misma y deberá realizar el cálculo del siguiente salto con Dijkstra para continuar con la transmisión de los datos. Todo este algoritmo es resumido en el diagrama de flujo presentado en la Figura 2.13. Cabe indicar que este proceso se repite cada vez que un nodo se encuentra en su ventana TDM asignada.

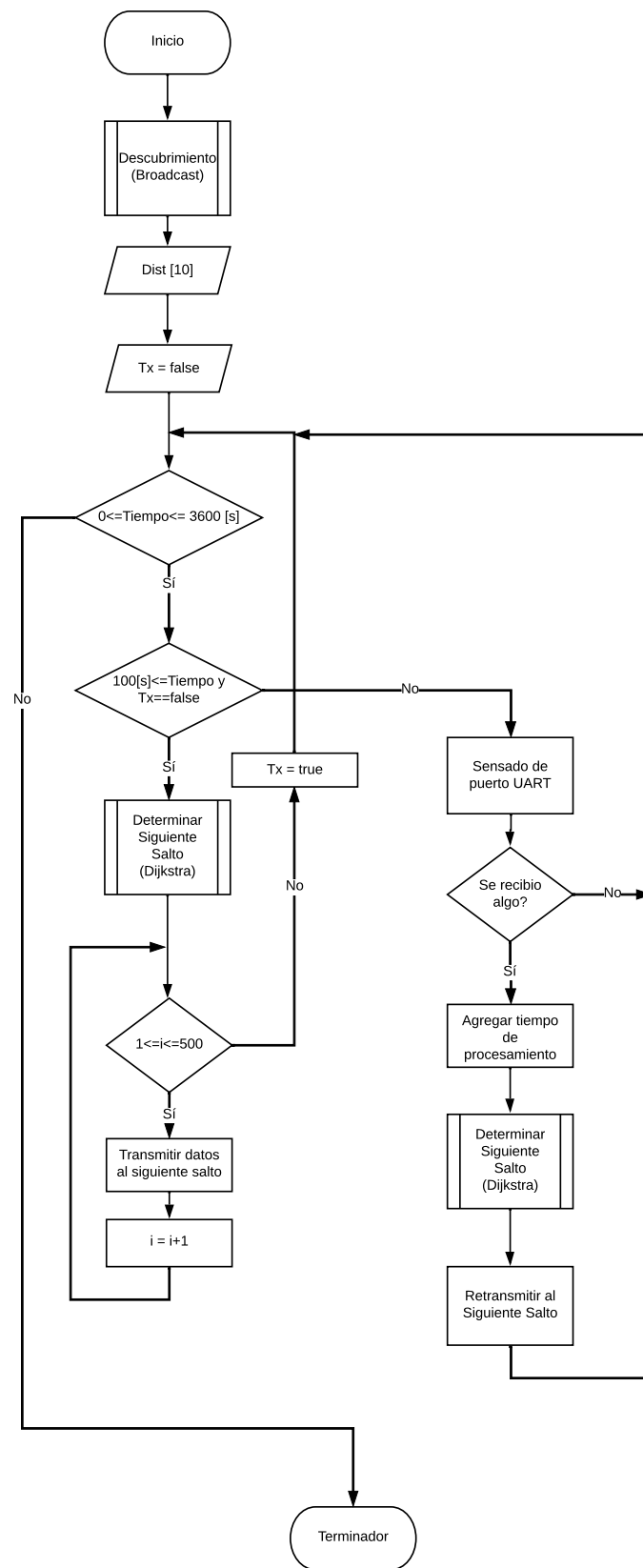


Figura 2.13: Diagrama de Flujo del funcionamiento del nodo sensor con el protocolo Dijkstra.

2.2.2. Prim

El algoritmo Prim es una posible solución al problema del árbol de mínima expansión (MST del inglés Minimum Spanning Tree) que permite conectar los nodos minimizando el costo total sin que existan lazos o *loops*. Este algoritmo implementado en una red tiene un funcionamiento similar al de Dijkstra para el cálculo de las rutas porque en el protocolo Prim se busca encontrar la ruta más corta para la transmisión de un paquete considerando además de una matriz de métricas, una matriz de pesos que establece una prioridad de uso de cada uno de los enlaces del nodo [39]. Este proceso se demuestra en la Figura 2.14 en donde el número de la izquierda en negro de cada enlace representa la distancia y el de la derecha en verde representa el peso. Para determinar la distancia más corta considerando el peso de cada enlace, se realiza una multiplicación entre la distancia y el peso, por lo tanto a mayor valor del peso menor probabilidad de uso de ese enlace.

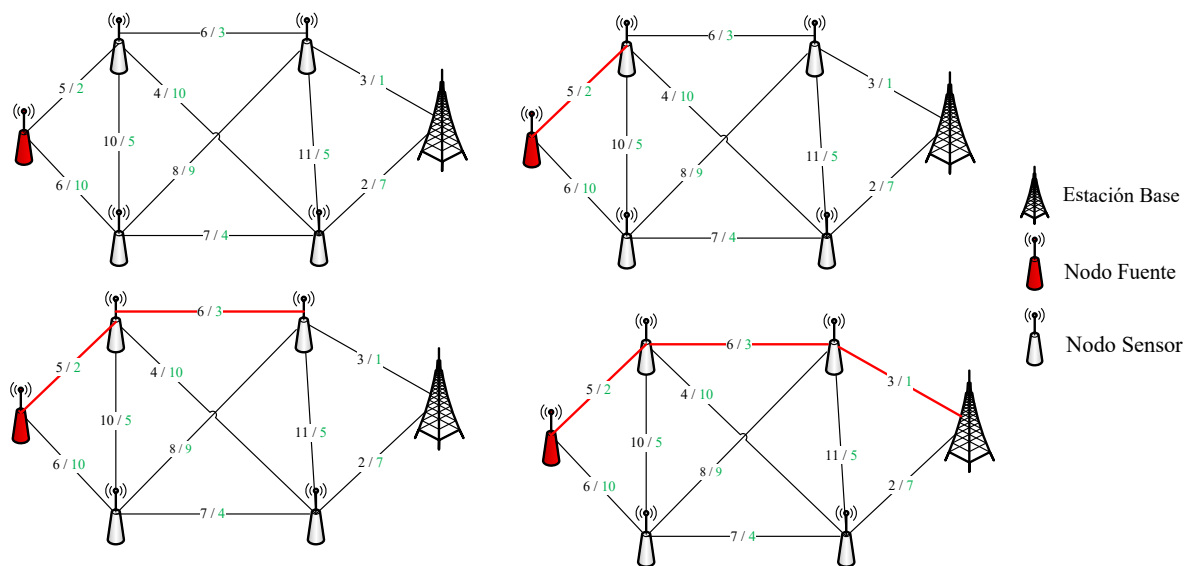


Figura 2.14: Ejemplo de ejecución del algoritmo Prim.

Para la implementación de este algoritmo en la WSN se debe interpretar que el vector de pesos debe estar acorde con la calidad de enlace que existe entre los nodos. Al igual que la programación del protocolo Dijkstra, para iniciar el funcionamiento de este algoritmo es pertinente realizar un descubrimiento de los nodos descrito en la Figura 2.9.

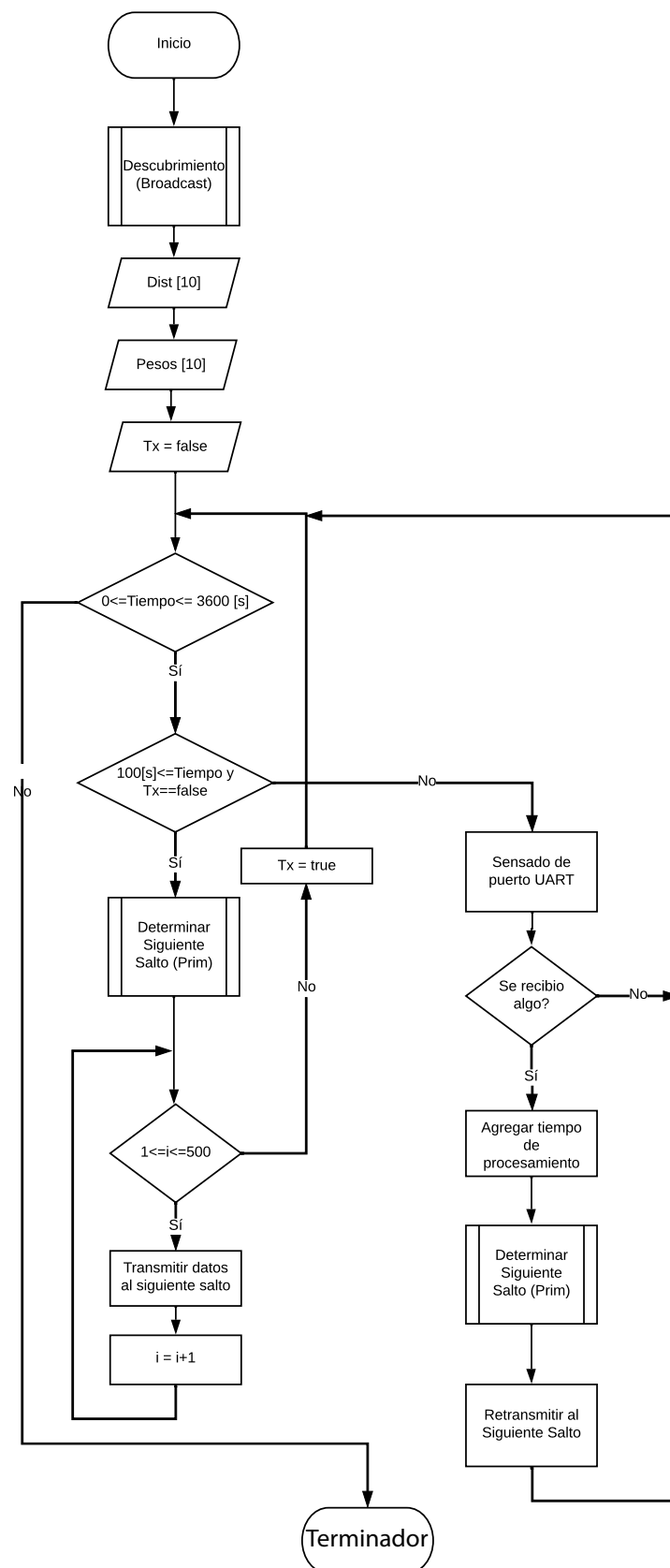


Figura 2.15: Diagrama de Flujo del funcionamiento del nodo sensor con el protocolo Prim.

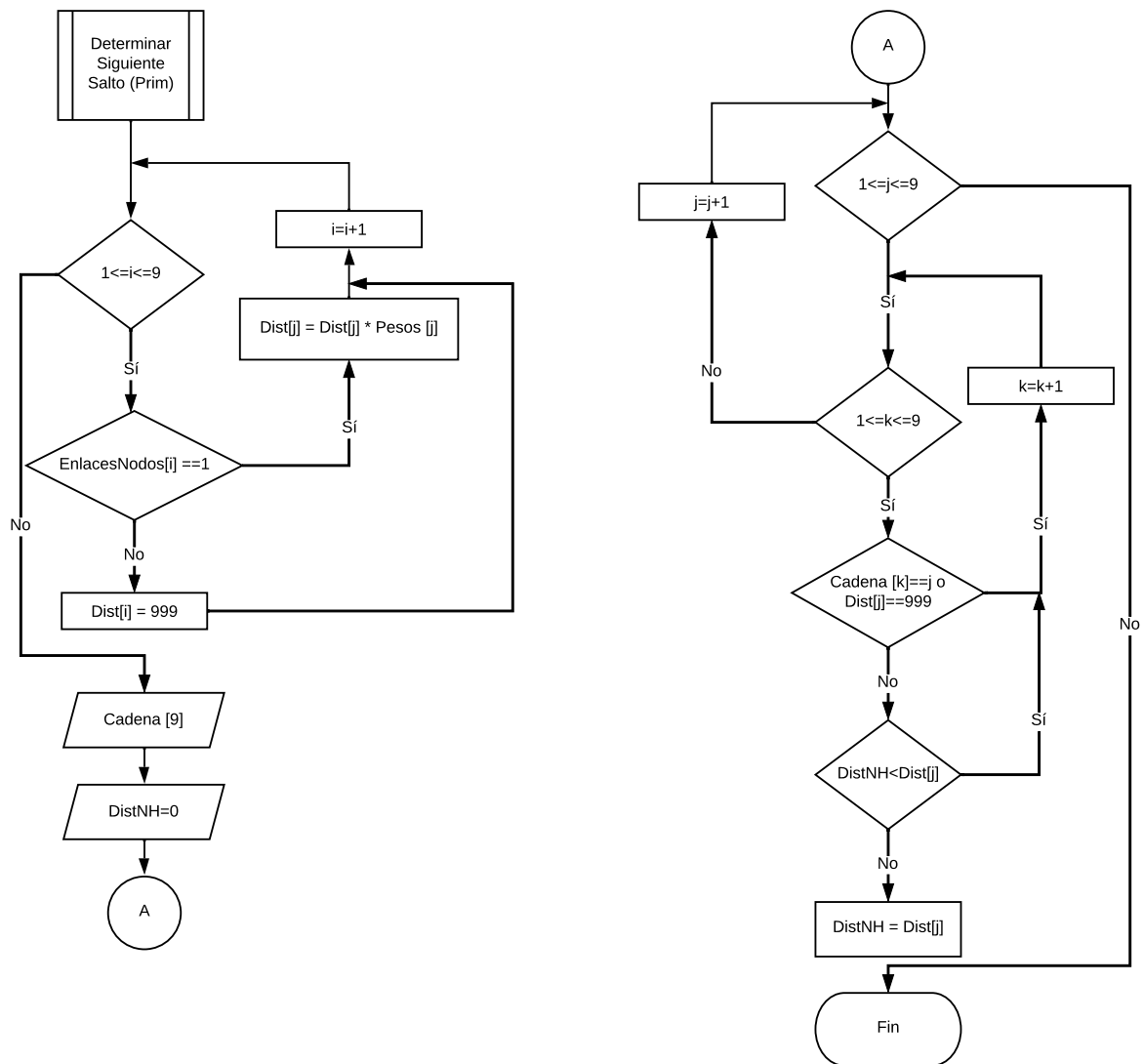


Figura 2.16: Diagrama de Flujo del algoritmo Prim.

Posterior a la fase de configuración inicia la fase de estado estable en donde el funcionamiento de este algoritmo es similar al de Dijkstra con la particularidad de que se debe recalcular el valor de la distancia de acuerdo a los pesos establecidos. El funcionamiento del algoritmo Prim en forma de diagrama de flujo se presenta en la Figura 2.15 y la del protocolo de forma completa en la Figura 2.16 en la que se ocupan las mismas variables que el protocolo Dijkstra.

2.2.3. Low-Energy adaptative Clustering Hierachy (LEACH)

El protocolo LEACH [36] se basa en la agrupación por clusters donde la selección del CH se genera mediante una rotación aleatoria para distribuir uniformemente la carga de energía entre los sensores de la red, donde los CH son los encargados en la transmisión directa con la estación base y los nodos sensores realizan la transmisión de sus paquetes a los CH [36], [40]. La arquitectura que permite el uso de este protocolo es de tipo estrella extendida tal como se puede apreciar en la Figura 2.17. Además, este protocolo incorpora una compresión de datos para reducir la cantidad de datos que se transmiten hacia la estación base. El sistema puede llegar a determinar a priori, el número óptimo de clusters necesarios para abarcar la cobertura de todos los nodos sensores, esto dependerá de varios factores como son la topología de la red y los costos relativos de computación vs. comunicación [36].

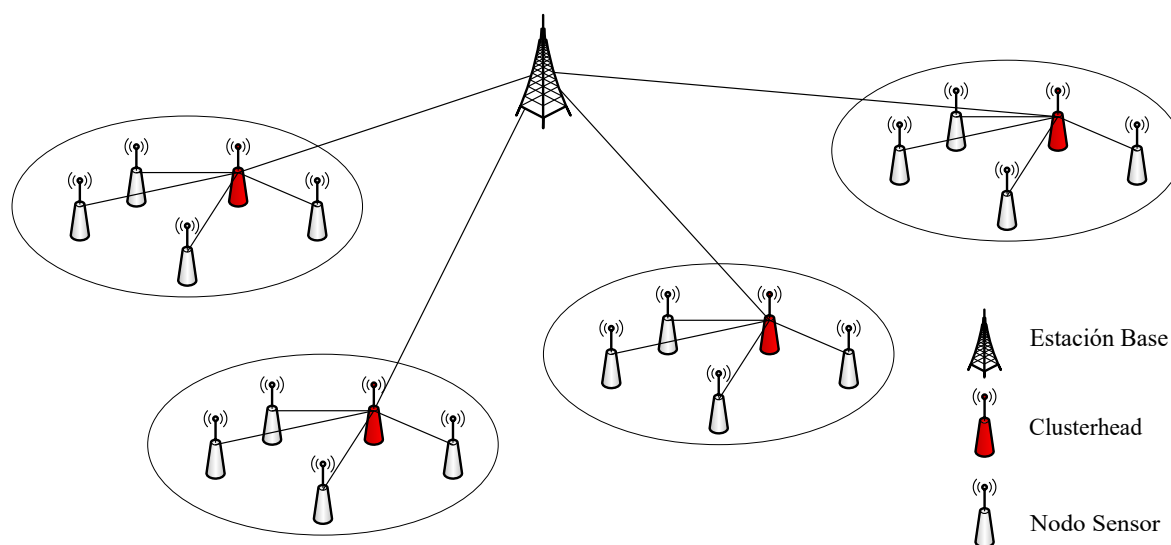


Figura 2.17: Topología lógica del protocolo LEACH.

Con la selección aleatoria del CH, LEACH resuelve el problema de la desconexión de un grupo de sensores los cuales se encontraban conectados a un nodo que termino su tiempo de vida útil, realizando un agrupamiento adaptativo y auto organizado que permite no agotar la batería de un único nodo sensor [36]. El proceso de generación de los cluster consiste en la determinación del CH la cual se basa en el porcentaje sugerido de CH para la red y el número de veces que ese nodo ya ha sido

considerado CH hasta ese momento, esta decisión la toma el nodo eligiendo un número aleatorio entre 0 y 1, si el número es menor que un umbral el nodo se convierte en CH [36], [40]. El umbral con el cual se comparará el número aleatorio se puede calcular con la ecuación 2.2.

$$T(n) = \begin{cases} \frac{P}{1 - P \times (r \bmod \frac{1}{p})} & \text{si } n \in G \\ 0 & \text{otro caso} \end{cases} \quad (2.2)$$

Donde P es el porcentaje deseado de CH, r es la ronda actual, n es el nodo el cual decidió ser CH, G es el conjunto de nodos que no han sido CH en las últimas $\frac{1}{p}$ rondas. Posteriormente los sensores seleccionan a que cluster desean conformar en base a la energía de comunicación mínima entre el sensor y el CH. Cuando todos los nodos sensores pertenecen a un cluster, cada CH genera una programación TDM para que los nodos de su cluster solo puedan realizar la transmisión de sus datos en un instante de tiempo y a su vez encienda sus componentes de radio para la transmisión con la finalidad de minimizar la energía disipada en los sensores individuales [36]. Cada una de las programaciones son enviadas desde cada CH a la estación base, por lo que solo pocos sensores de la red deben realizar transmisiones que requieran alto consumo energético.

La aleatoriedad de la selección del CH se realiza de manera autónoma, es decir, cada nodo se selecciona a sí mismo como CH independientemente del resto de nodos de la red considerando su energía residual, el primer conjunto de nodos se mantendrán como CH en un tiempo t_0 y para un tiempo t_1 se seleccionan nuevos nodos para de esta manera los nodos con mayor energía restante realicen las funciones que consumen mayor energía en la red. Cada nodo que se ha elegido a sí mismo como CH transmite un mensaje de advertencia con la misma potencia, los demás nodos sensores permanecen en un proceso de recepción para captar los mensajes de advertencia de cada CH, y en base a la intensidad de la señal recibida del mensaje el nodo sensor selecciona a que cluster pertenecer [36]. Este proceso tiene la ventaja de eliminar la negociación entre los CH reduciendo las transmisiones entre estos nodos.

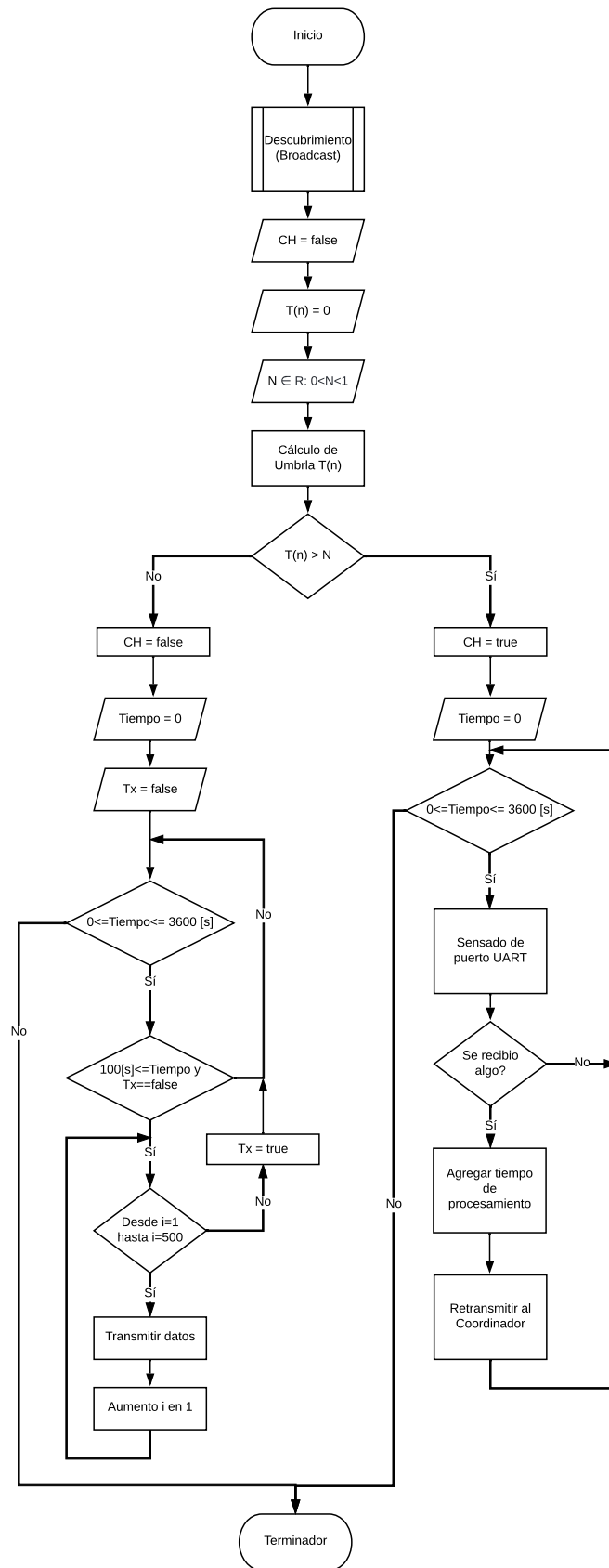


Figura 2.18: Diagrama de Flujo del funcionamiento del nodo sensor con el protocolo LEACH.

A diferencia de los protocolos descritos anteriormente, en el protocolo LEACH después del proceso de descubrimiento cada nodo sensor debe determinar si esa ronda tiene la función de Clusterhead o de nodo sensor, para esto se calcula un umbral con la ecuación 2.2 la cual debe ser mayor a un número aleatorio N que va desde 0 y 1. Si el umbral es mayor al número N entonces el nodo es considerado como CH, caso contrario el nodo se dedicará únicamente a la transmisión de sus datos y no realizará el procedimiento de recepción.

Los procesos de transmisión y recepción son iguales a los protocolos descritos anteriormente, en donde se realiza una multiplexación TDM para que cada nodo sensor tenga la oportunidad de transmitir sus datos hacia el CH y este a su vez a la estación base. El diagrama de flujo de este protocolo esta presentado en la Figura 2.18.

2.2.4. Power-Efficient Gathering in Sensor Information Systems (PEGASIS)

PEGASIS [41] es una mejora al protocolo LEACH en donde cada nodo sensor se comunica únicamente con sus vecinos cercanos formando una cadena entre los nodos para que a través de ellos se pueda transmitir y recibir la información. Los nodos cercanos a la estación base, tienen un determinado tiempo para la transmisión de sus paquetes para conseguir reducir la energía promedio gastada por cada nodo en cada ronda y para distribuir la carga energética de manera uniforme entre los nodos de la red [41].

La organización de los nodos en forma de cadena se realiza a partir de un algoritmo goloso, en inglés Greedy, el cual se ejecuta desde el nodo más alejado de la estación base para garantizar que ese nodo lejano cuente con vecinos para poder realizar su transmisión y, además, porque en el algoritmo Greedy la distancia con respecto a los vecinos aumentará gradualmente ya que los nodos que ya pertenezcan a una cadena no podrán pertenecer a otra [41]. Para seleccionar el vecino más cercano, cada nodo evalúa la intensidad de la señal del nodo vecino. Cuando uno de los nodos de la cadena agota su energía y se presenta como inactivo para el resto de sensores, se reconstruye la cadena realizando el proceso anterior para eliminar el nodo inactivo

[40], [41]. La Figura 2.19 presenta la topología que se genera con el algoritmo PEGASIS.

Para la transmisión de la información, cada nodo excepto el nodo final de la cadena fusiona sus datos con los provenientes del nodo vecino y los transmite al siguiente nodo de la cadena. Para identificar que nodo debe realizar su transmisión, el nodo final de la cadena envía un token hacia el nodo inicial para que transmita sus datos por los nodos de la cadena [41]. Cada nodo de la red estará situado en una posición aleatoria de la cadena, por lo tanto, los nodos que se comunicarán con la estación base consumirán con mayor velocidad su energía en ubicaciones aleatorias permitiendo que la red sea resistente a fallos [40].

Para evitar que los nodos que se encuentren muy distantes a la estación base se conviertan en los nodos finales de la cadena, se establece un umbral en la distancia del vecino para ser el nodo final [41].

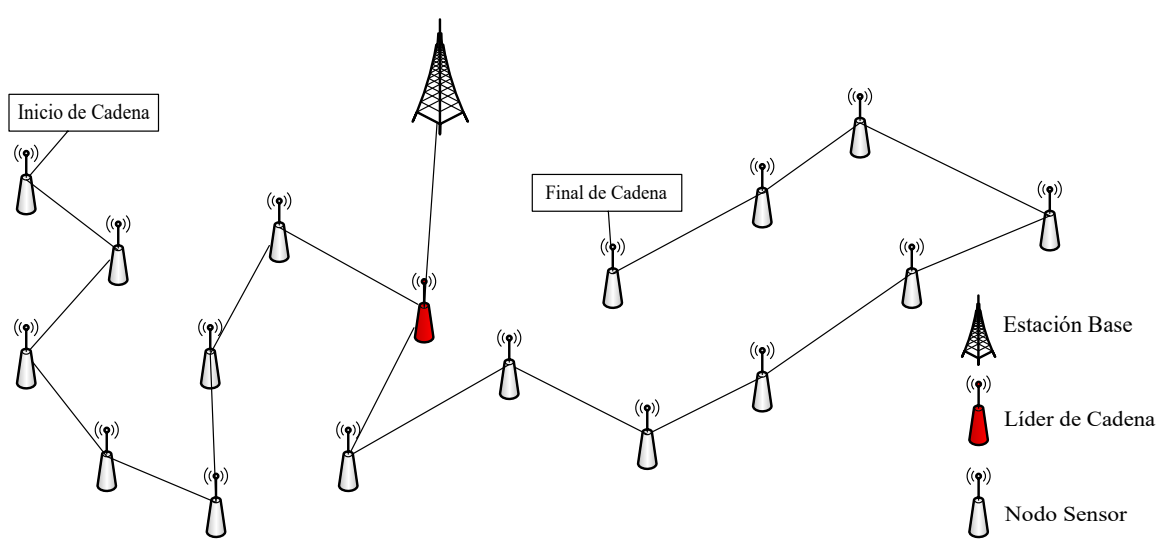


Figura 2.19: Topología lógica del protocolo PEGASIS.

Las ventajas que presenta este protocolo frente a LEACH consiste en la disminución de la distancia para la transmisión de los nodos hacia su próximo salto, que en el caso de LEACH era directamente el CH [40], [41]. Además, se reduce la cantidad de datos que debe recibir el nodo final o CH, ya que en PEGASIS recibirá el paquete del nodo vecino mientras que en LEACH el CH recibida una cantidad de paquetes equivalente a la cantidad de nodos pertenecientes a ese cluster [41].

Para la programación de este protocolo, se debe considerar que los nodos tienen

un conocimiento global de la red, por lo tanto para la selección de los nodos líderes de la cadena de multisalto se tomó en cuenta a los nodos que tienen línea directa al coordinador de la red y que a su vez tienen más de un nodo dentro de su cobertura y que comparten línea de vista. En base a lo descrito anteriormente, los nodos que cumplen estas características son los nodos sensor 1, 2 y 9, por lo tanto estos nodos realizarán la misma función que un Clusterhead.

Cabe indicar que el proceso de descubrimiento se continúa realizando con la finalidad de establecer los niveles jerárquicos de la red, es decir, conocer al líder de la cadena a la que pertenece cada nodo sensor y a su vez determinar los enlaces con el resto de nodos sensores.

En la fase de estado estable, los nodos sensores tendrán la oportunidad de transmisión dentro de su ventana de tiempo y realizarán una cadena de multisalto a través del algoritmo Dijkstra descrito en la Figura 2.12, y a su vez el resto de nodos monitorean sus enlaces para cuando surja el evento de recepción de un paquete estos puedan determinar su próximo salto y retransmitir los datos.

Los nodos líderes de la cadena realizarán su retransmisión directamente al coordinador tal como lo realiza un CH. En la Figura 2.20 presenta el funcionamiento de este protocolo en su totalidad.

2.3. Dimensionamiento de la Red

Para el dimensionamiento de la red, se asume un área cuadrada en metros de L por L en un espacio abierto. En esta región se colocan los nodos de sensores para generar una conectividad entre ellos, sabiendo que los módulos Xbee permiten una cobertura de cierto radio en metros; en consecuencia, los nodos pueden ser colocados en cualquier sitio de esta área. Pero para facilitar el proceso de instalación de los nodos, se tomaron en cuenta los puntos georreferenciados del estudio Levantamiento de los sensores IoT implementado en la Universidad Politécnica Salesiana sede Cuenca. De esta manera disminuimos el costo de implementación de los nodos [23].

Para los modelos matemáticos del dimensionamiento se considera un conjunto de nodos N distribuidos en la región y se considera un conjunto de M ubicaciones posibles o sitios candidatos donde se implemente al nodo coordinador. El modelo define que un nodo sensor está cubierto si se encuentra a una distancia R de al menos un coordinador, para esto se utiliza la distancia de Haversine de tal forma que se considere la curvatura de la Tierra entre los puntos georreferenciados [23].

Los sitios candidatos son considerados como sitios activos si el nodo coordinador está instalado en esa ubicación. Dentro del diagrama de red, cada coordinador cuenta con una capacidad de nodos sensores conectados al unísono, tras realizar pruebas se determinó que esta capacidad es de tres nodos funcionando de manera simultánea y se denota con la variable C . A partir de los detalles anteriores, el problema de optimización tiene como objetivo encontrar el número mínimo de sitios activos tal que al menos un porcentaje P de los sensores estén cubiertos. Para esto es necesario definir un conjunto $S = \{S_1, S_2, S_3, \dots, S_M\}$ de sitios candidatos, donde la j -ésima posición viene dada por (x_{s_j}, y_{s_j}) ; y un conjunto de nodos sensores $D = \{D_1, D_2, D_3, \dots, D_M\}$ donde la posición i -ésimo está dada por (x_{d_i}, y_{d_i}) . Además, se define la cantidad $\alpha_{j,i} \in \{0, 1\}$, lo que indica que si el sensor i tiene conexión con el coordinador j , entonces el valor es 1; caso contrario, el valor es 0. Así, para cada sitio candidato, se define la cantidad $Z_j \in \{0, 1\}$ la cual implica que el valor es 1 cuando el sitio candidato j es un sitio activo. De la misma manera, para cada nodo sensor d_i la cantidad $Y_i \in \{0, 1\}$ está definida cuando el valor es 1 el nodo sensor está cubierto por

al menos un sitio candidato [23]. El modelo de optimización para el dimensionamiento se presenta a continuación [23]:

Función objetivo:

$$\min \sum_{j=1}^M Z_j, \quad (2.3)$$

que está sujeto a:

$$Y_i = \sum_{j=1}^M X_{j,i}; \quad \forall i \in D; \quad (2.4)$$

$$\sum_{i=1}^N X_{j,i} \leq C \cdot Z_j; \quad \forall j \in S; \quad (2.5)$$

$$\sum_{i=1}^N Y_i \geq N \cdot P; \quad \forall i \in D; \quad (2.6)$$

$$X_{j,i} \leq \alpha_{j,i} \cdot Z_j; \quad \forall j \in S; \forall i \in D; \quad (2.7)$$

A partir del modelado matemático descrito anteriormente, la WSN puede ser dimensionada de manera óptima considerando la cobertura y capacidad de cada nodo sensor. Para determinar la influencia del dimensionamiento en los protocolos de enrutamiento se debe estudiar dos escenarios, el primero considerando un despliegue de los nodos de la WSN en ubicaciones consideradas por conveniencia, y el segundo con un despliegue de la WSN con el coordinador en una ubicación considerada como sitio candidato. Para solucionar los modelos matemáticos del dimensionamiento se utilizó los software Openstreetmap, Matlab y LPSolve. En ambos escenarios se analiza los protocolos de enrutamiento descritos anteriormente y se analiza las métricas de latencia y pérdidas de paquetes. Este procedimiento es descrito en la Figura 2.21 donde se indica la motivación del estudio, el método realizado y por último las métricas a analizar.

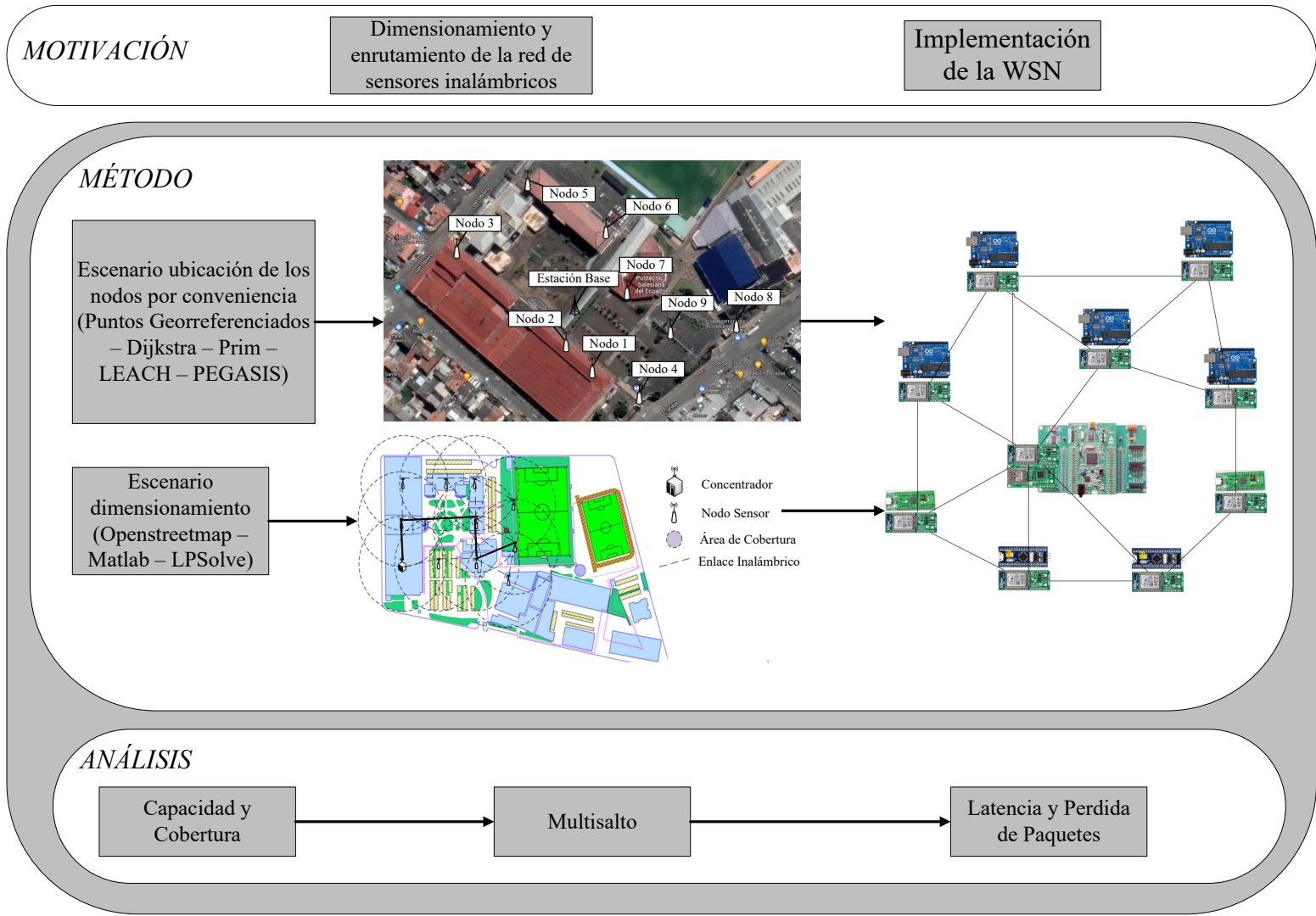


Figura 2.21: Dimensionamiento y enrutamiento georreferenciados de redes de sensores inalámbricos.

Capítulo 3

Resultados

De acuerdo a lo descrito en el final del Capítulo 2, para la obtención de los resultados se realizaron pruebas en dos escenarios diferentes. En el primer escenario se realizó el despliegue de la WSN sin criterios de dimensionamiento y cuyas ubicaciones fueron preestablecidas en el despliegue de una red ZigBee de 8 nodos en el Campus el Vecino en la ciudad de Cuenca de la Universidad Politécnica Salesiana para la evaluación de diversas variables y que por diversos motivos ha dejado de usarse en la actualidad. Debido a que la red no estuvo operativa, no se consideró el mismo contexto en cuanto al tipo de variables que se estuvieron monitorizando y transmitiendo además se desconoce cada cuanto tiempo se transmitían dichos datos.

En lo posterior, se aplicó dimensionamiento óptimo para identificar la ubicación más adecuada del nodo concentrador considerando el radio máximo de cobertura y la capacidad del coordinador para recibir mensajes de varios nodos enrutadores si estos transmiten en un instante de tiempo similar. Entonces, para el análisis de este trabajo, se consideró la transferencia de datos de medición inteligente correspondiendo a 3 variables de voltaje con un tamaño de 2 bytes cada una y en un contexto en el que cada nodo tiene una ventana de tiempo de hasta 5 minutos antes de realizar otra medición y transmitirla, datos no menores al momento de construir las tramas de transmisión dentro del protocolo de enrutamiento. No obstante, al momento de realizar las pruebas y evaluar la capacidad del concentrador se optó por considerar una ventana de tiempo de recepción de datos de parte del nodo coordinador más pequeña, concretamente 1 minuto, esto para generalizar el contexto de la aplicación de la red debido a que

se usaron 10 nodos en esta evaluación. Además, si se tratase de una aplicación de medición inteligente urbano, por la cantidad de nodos transmitiendo, el número de mensajes sería mucho mayor, dándole sentido a la consideración de usar una ventana de tiempo de 1 minuto entre cada recepción de datos de parte del coordinador.

En lo que respecta las pruebas realizadas, se analizó la latencia y pérdida de paquetes de cada protocolo descrito en el capítulo anterior. Además, en el escenario de la red no dimensionada se estudio el comportamiento de los protocolos de enrutamiento cuando empieza a existir eventos de desconexión de los nodos sensores. El segundo escenario corresponde al despliegue de la red de sensores inalámbricos luego de analizar la mejor ubicación del coordinador de acuerdo al modelo de dimensionamiento para, en teoría garantizar las características físicas de la red por lo cual en este escenario se volvió a analizar las métricas de latencia y perdida de paquetes.

Así, este capítulo describe los resultados obtenidos en el desarrollo del presente trabajo.

3.1. Escenario sin dimensionamiento

Para las pruebas de transmisión, cada nodo sensor tenía que transmitir un total de 500 paquetes en su ventana temporal omitiendo los paquetes de descubrimiento y de las cadenas multisalto. por lo tanto, en los protocolos Dijkstra y Prim se esperaba recibir un total de 4500 paquetes, mientras que en los protocolos LEACH y PEGASIS 3500 paquetes ya que se debe omitir los nodos que tuvieron la función de CH siendo estos el 20 % del total de nodos.

La Figura 3.1 presenta la cantidad de paquetes recibidos de manera exitosa en el coordinador de acuerdo al nodo sensor transmisor en la ejecución de cada protocolo de enrutamiento. Las etiquetas en la figura indican el nodo sensor con mayor transmisiones exitosas en cada protocolo de enrutamiento. Al analizar los paquetes recibidos por cada nodo en el primer escenario, se conoce que en el protocolo Dijkstra la estación base recibió 3509 paquetes, con el protocolo Prim 3564, con el protocolo PEGASIS 2659 y con LEACH se recibió un total de 2719. Esto demuestra que mediante

el protocolo Prim se consigue recibir más paquetes que el protocolo Dijkstra porque los nodos sensores priorizan los enlaces con mejor calidad.

Por otro lado, entre los protocolos LEACH y PEGASIS, con LEACH existe una mayor cantidad de paquetes recibidos en la estación base porque se reducen la cantidad de multisaltos disminuyendo la posibilidad de pérdida de paquetes durante la transmisión entre nodos sensores. En este escenario en concreto, los nodos sensores 2 y 9 tuvieron la función de CH en el protocolo LEACH y en el protocolo PEGASIS los nodos 2 y 9 fueron los líderes de cadena, es por esto que estos nodos tienen un valor de 0 ya que estos no fueron la fuente del paquete sino realizaron únicamente la retransmisión. Al analizar Prim y LEACH, se determina que con Prim la estación base fue capaz de capturar el 79,2%, mientras que con el protocolo LEACH se recibió el 77.68%, con esto se deduce que el protocolo Prim presenta una mejora al priorizar los enlaces con mejor calidad.

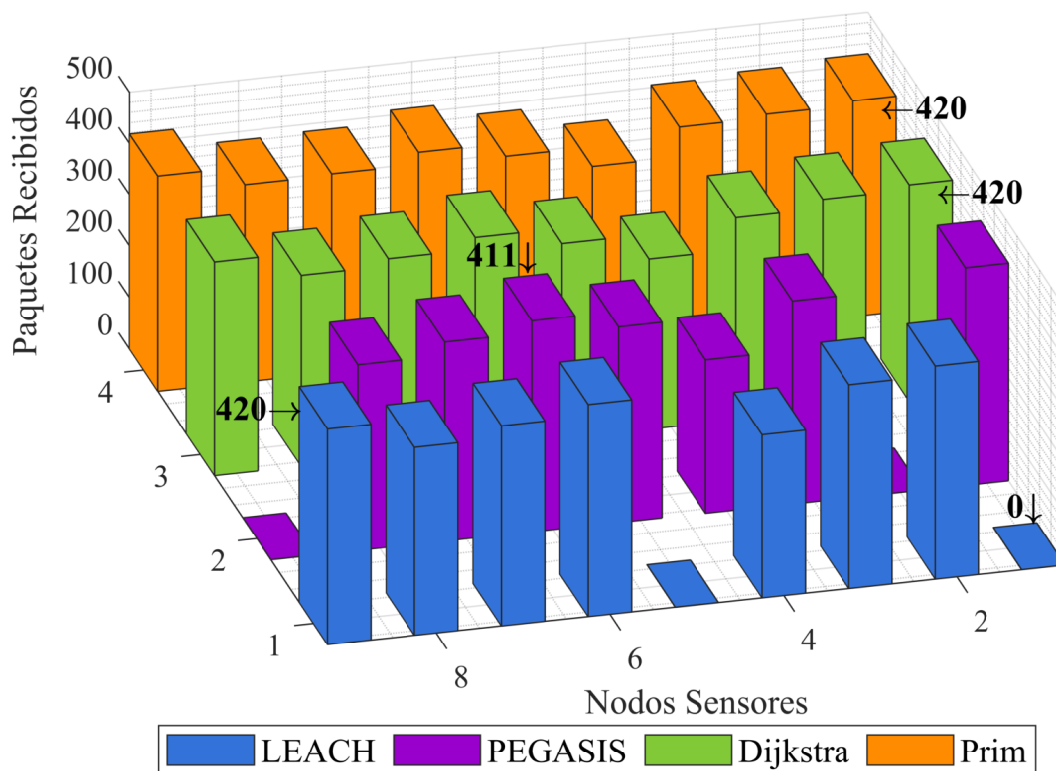


Figura 3.1: Paquetes recibidos por nodo.

En el mismo escenario, la latencia existente entre la transmisión de los paquetes aumenta debido al multisalto, por lo tanto, los protocolos Dijkstra, Prim y PEGASIS

presenta una mayor latencia frente al protocolo LEACH. Además, en los nodos más lejanos del coordinador la latencia aumenta porque necesitan recorrer varios nodos hasta que el paquete llegue a su destino. La Figura 3.2 presenta la latencia promedio en el proceso de transmisión de cada nodo sensor con los diferentes protocolos de enrutamiento en funcionamiento. Las etiquetas de la figura indican la mayor latencia presente en cada protocolo. Con esta figura se observa que en el protocolo Dijkstra existe una mayor latencia debido a su cálculo de próximo salto donde considera únicamente la distancia entre nodos por lo que la transmisión de paquetes se puede realizar en un enlace de baja calidad, es por esto que el protocolo Prim disminuye la latencia frente a Dijkstra.

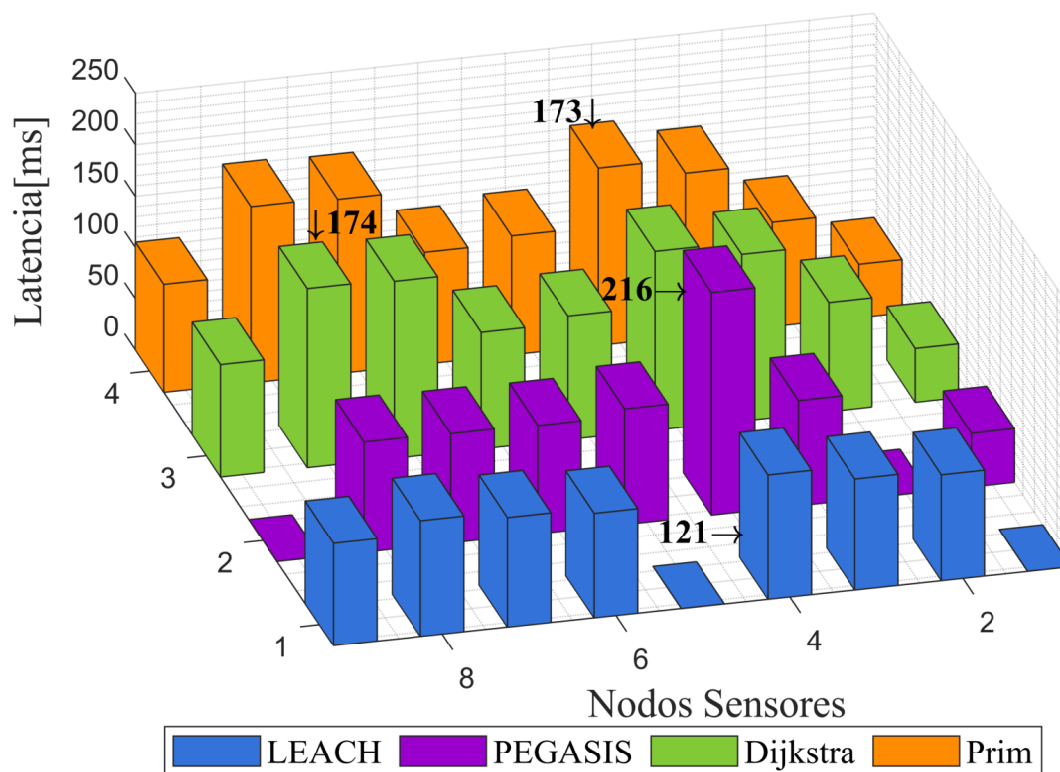


Figura 3.2: Latencia promedio en la transmisión de cada nodo.

Para determinar el comportamiento de los protocolos de enrutamiento cuando existe el fallo o la desconexión de un nodo de la red, se estudió un escenario donde los nodos se desconectaban después de un tiempo aleatorio. Este escenario se analizó únicamente cuando el concentrador se ubicaba en el sitio candidato D, ya que al no modificar la topología de red los resultados tanto en un escenario dimensionado y sin

dimensionar no presentarían una gran diferencia. En este escenario, la cantidad de paquetes recibidos en cada protocolo disminuye por el hecho que se reduce la cantidad de posibilidades de establecer un camino hasta llegar a la estación base o líder de cadena. Tras este estudio con el protocolo Dijkstra la estación base alcanzó a recibir 2339 paquetes, con Prim 2457 paquetes, con LEACH un total de 1076 paquetes y con PEGASIS 1076 paquetes. La pérdida de aproximadamente del 50 % de paquetes en los protocolos LEACH y PEGASIS se debe principalmente a que cuando existe la pérdida de conexión de un CH, todos los nodos sensores que lo consideraban en una posición jerárquica superior llegan a presentarse como inactivos para la red, ya que muchos de ellos no son capaces de conectarse a otro CH porque no cuentan con línea de vista, esto sucede en el protocolo LEACH con los nodos sensores 4 y 5 los cuales no contaban con línea de vista hacia los CH que en este escenario fueron los nodos sensores 2 y 9. Este no es el caso de PEGASIS, ya que los nodos realizan un descubrimiento jerárquico en su mismo nivel, es decir, además de conocer los líderes de cadena descubren a los demás nodos sensores, de tal forma que son capaces de redirigir sus paquetes hacia otro nodo sensor en caso de no poder transmitirlo a un líder de cadena.

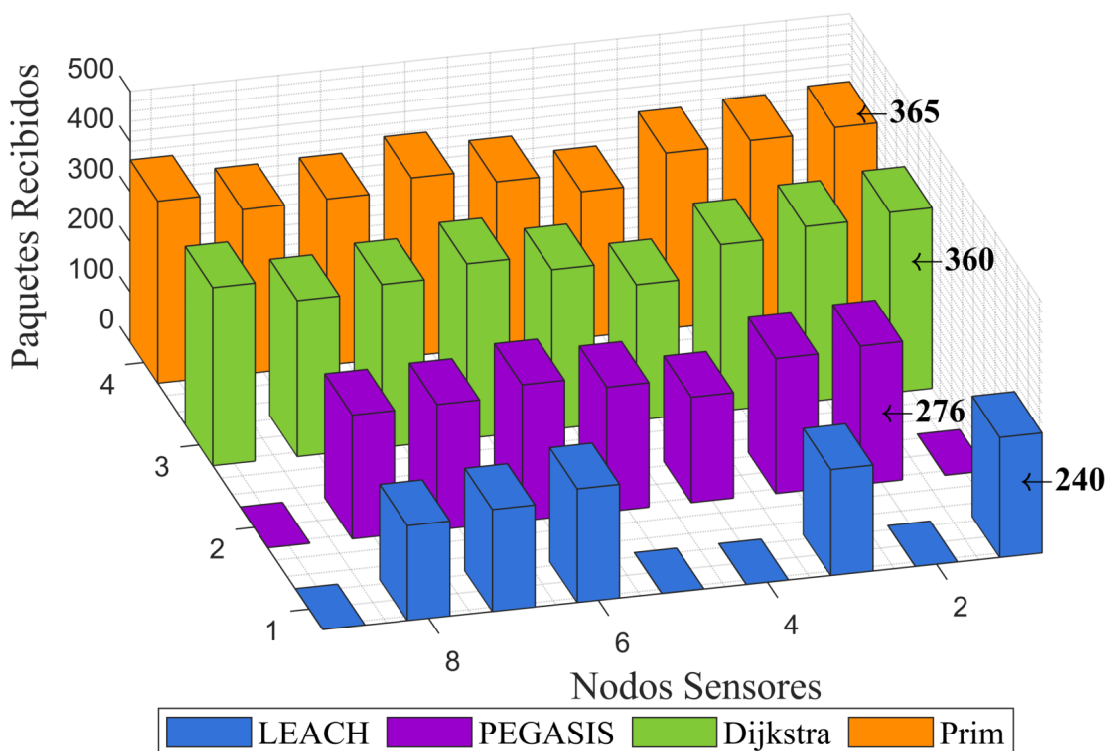


Figura 3.3: Paquetes recibidos por nodo con eventos de desconexión de nodos.

Por otro lado, en los protocolos Dijkstra y Prim la estación base recibió el 65,97% y el 68.2% respectivamente, con esto podemos intuir que la posibilidad de varios caminos para la transmisión de paquetes disminuye el riesgo de pérdida de paquetes. La Figura 3.3 presenta la relación entre la cantidad de paquetes recibidos y los nodos sensores fuentes de la transmisión en función del protocolo de enrutamiento en estudio. Además, se indica la mayor cantidad de paquetes recibidos en cada protocolo de enrutamiento.

Al igual que en el escenario donde no existe una desconexión de los nodos, la latencia es mayor en los protocolo con transmisión por multisalto, siendo en este caso PEGASIS el que demuestra una mayor latencia porque además de realizar su cadena de multisalto, los nodos sensores fuentes de la transmisión deben encontrar a un nodo líder de la cadena para que los paquetes sean recibidos por el coordinador. La latencia promedio existente en la transmisión de cada nodo se presenta en la Figura 3.4 donde esta latencia es distribuida de acuerdo a los protocolos de enrutamiento analizados. Las etiquetas en la figura indica la mayor latencia presente en cada protocolo.

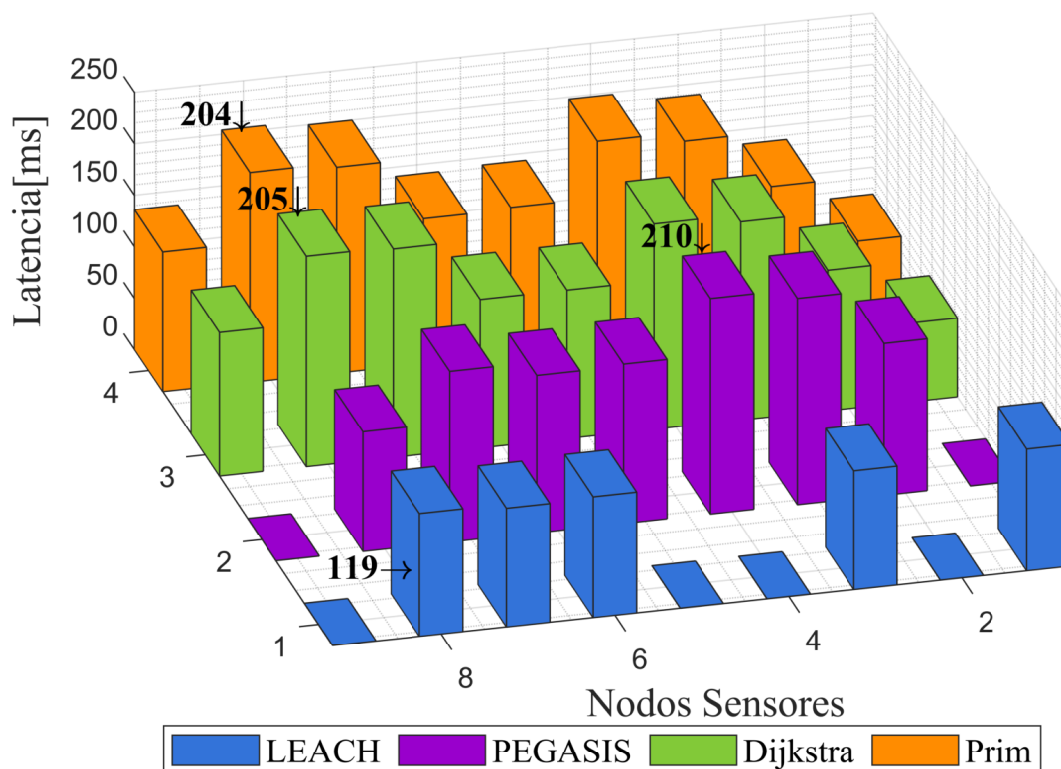


Figura 3.4: Latencia promedio en la transmisión de cada nodo con eventos de desconexión.

3.2. Escenario Aplicando Dimensionamiento

Para el dimensionamiento de red se establecieron 7 sitios candidatos descritos en la tabla 3.1, a partir de los cuales, los sitios *E*, *F* y *G* se encontraban a una altura superior al resto de nodos de aproximadamente 15 [m], por lo tanto el cálculo de la distancia con respecto a estos nodos se realizó a través de la ecuación 3.1.

Tabla 3.1: Puntos Georreferenciados de los sitios candidatos.

Sitios Candidatos	Coordenadas	
	Latitud	Longitud
A	-2,886998	-78,990127
B	-2,886958	-78,990198
C	-2,886895	-78,989738
D	-2,886761	-78,990411
E	-2,886742	-78,990694
F	-2,886216	-78,990813
G	-2,886736	-78,989365

$$d \approx \sqrt{2rh} \quad (3.1)$$

Donde r es el radio de la tierra 6371 [m] y h es la diferencia de altura entre el nodo sensor y el sitio candidato. Los enlaces entre nodos que no tienen línea de vista son establecidos con una distancia de infinito de tal forma que los sitios candidatos omitan ese enlace de conexión y por ende su cobertura.

Para el desarrollo de los modelos matemáticos, se consideró una capacidad de funcionamiento simultaneo de 3 nodos sensores, un porcentaje de cobertura entre el 70 % y el 100 % y un radio de cobertura entre 110 [m] y 170 [m]. Se conoce que el radio de cobertura del módulo XBEE utilizado es de 1200 [m] en espacios abiertos, pero por la densidad del escenario su cobertura alcanza hasta los 180 [m], es por esto que para el modelamiento matemático se decidió disminuir el radio de cobertura para garantizar el alcance de todos los nodos.

Los modelos matemáticos descritos anteriormente son implementados en MATLAB para que a través de este se genere un archivo **.lp**, para que el software LPSolve resuelva a través de programación lineal entera mixta (MILP por su siglas en inglés de Mixed Integer Linear Programming) el problema de optimización lineal. Es

decir, LPSolve permite resolver el problema de optimización mas no MATLAB. Se usó este “*solver*” ya que es de licencia abierta [42].

Los resultados obtenidos del análisis de la capacidad y cobertura son presentados en la tabla 3.2 los cuales indican que los sitios candidatos *A*, *B* y *D* son los sitios óptimos donde se debe implementar un nodo coordinador para alcanzar una cobertura del 70 % y 80 %.

Tabla 3.2: Resultados del dimensionamiento.

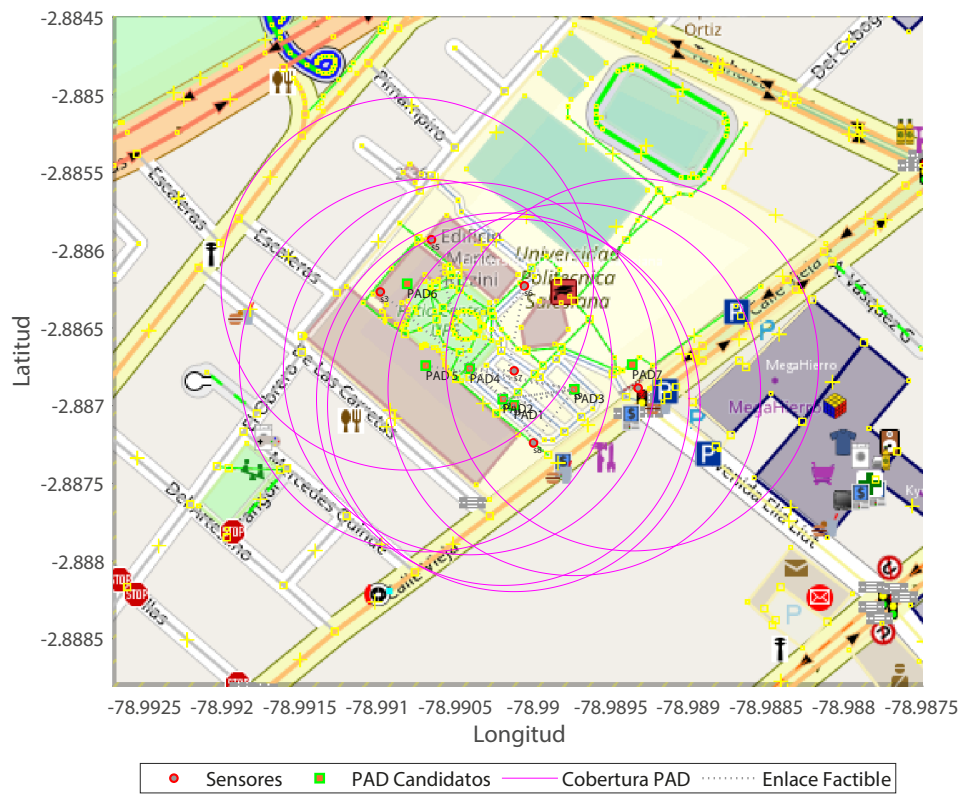
Radio de Cobertura	70 %	80 %	90 %	100 %
110	A B D	A B D	-	-
120	A B D	A B D	-	-
130	A B D	A B D	-	-
140	A B D	A B D	-	-
150	A B D	A B D	-	-
160	A B D	A B D	-	-
170	A B D	A B D	-	-

Del total de 7 sitios candidatos, se determinó que con 3 coordinadores se puede alcanzar una cobertura de hasta el 80 % de los nodos generando una WSN optimizada de tal forma que cada nodo sensor cuente con un enlace directo a un coordinador consiguiendo reducir la latencia y la pérdida de paquetes por el multisalto.

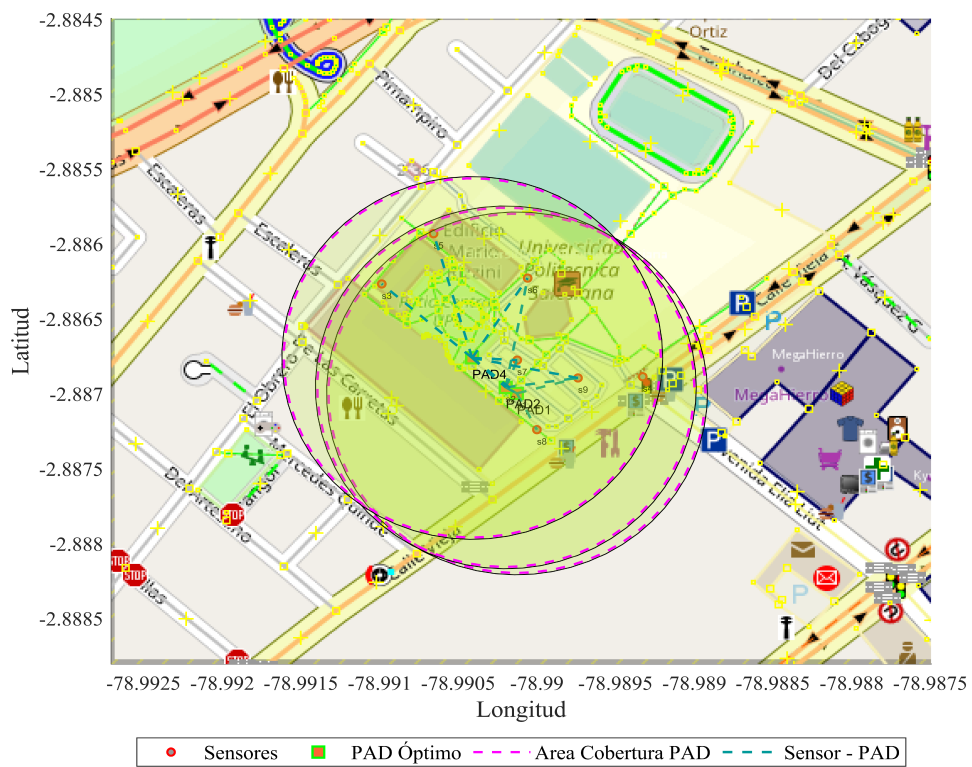
La Figura 3.5(a) presenta el escenario original de estudio con los radios de cobertura de cada nodo sensor.

La Figura 3.5(b) presenta el escenario óptimo con los enlaces entre los coordinadores y los nodos sensores. Estas gráficas se realizaron considerando un radio de cobertura de 170 [m] y un porcentaje de cobertura del 80 %.

El resto de resultados no presentan ya que tanto los enlaces, como los puntos óptimos de la red no se modifican.



(a) Escenario original



(b) Escenario óptimo

Figura 3.5: Identificación de las ubicaciones para optimización de la red.

Para una mejor interpretación en la tabla 3.3 se indican las conexiones entre coordinadores y nodos sensores. A partir de este resultado se puede conocer que en los análisis con una cobertura del 90 % y 100 % de nodos sensores no fueron resueltos porque el nodo sensor 4 es el único que no cuenta con línea de vista directa hacia uno de los sitios candidatos. Para la implementación de este escenario se modificó la posición del coordinador al del nodo 1 y 2 ya que se buscaba determinar la ubicación óptima para el coordinador a partir de los tres sitios activos obtenidos mediante el modelado matemático de la capacidad y cobertura. Con el coordinador ubicado en estos puntos, se ejecutaron los protocolos de enrutamiento para analizar la tasa de pérdida de paquetes y la latencia.

Tabla 3.3: Enlaces entre coordinadores y nodos sensores.

Coordinador	Nodos Suscritos
A	8 9
B	1 6 7
D	2 3 5

Cuando el coordinador se ubicó en la posición del nodo sensor 1 la estación base recibió 1652 paquetes cuando se ejecutó el algoritmo LEACH, 2919 paquetes con PEGASIS, 3593 paquetes con el algoritmo Dijkstra y 3748 paquetes con Prim, siendo Prim el algoritmo con menor tasa de pérdida de paquetes logrando recibir el 83.28 % de los paquetes transmitidos. La cantidad de paquetes recibidos de cada nodo sensor en este escenario es presentado en la Figura 3.6, en donde se agrupa la transmisión de todos los nodos sensores en función del protocolo de enrutamiento. Además, las etiquetas apuntan el nodo sensor fuente de la mayor cantidad de paquetes recibidos por el coordinador en cada protocolo de enrutamiento.

En cuanto a la latencia, la variación no es muy evidente porque los nodos sensores siguen realizando la comunicación por multisalto, es por esto que LEACH sigue presentado una menor latencia frente al resto de protocolos. La comparación de la latencia existente en cada nodo en función del protocolo de enrutamiento es presentado La Figura 3.7 presenta la latencia promedio existente en la transmisión de cada nodo sensor en función del protocolo de enrutamiento implementado en la red, donde a su vez se indica la mayor latencia existente en un nodo sensor de acuerdo al

protocolo de enrutamiento.

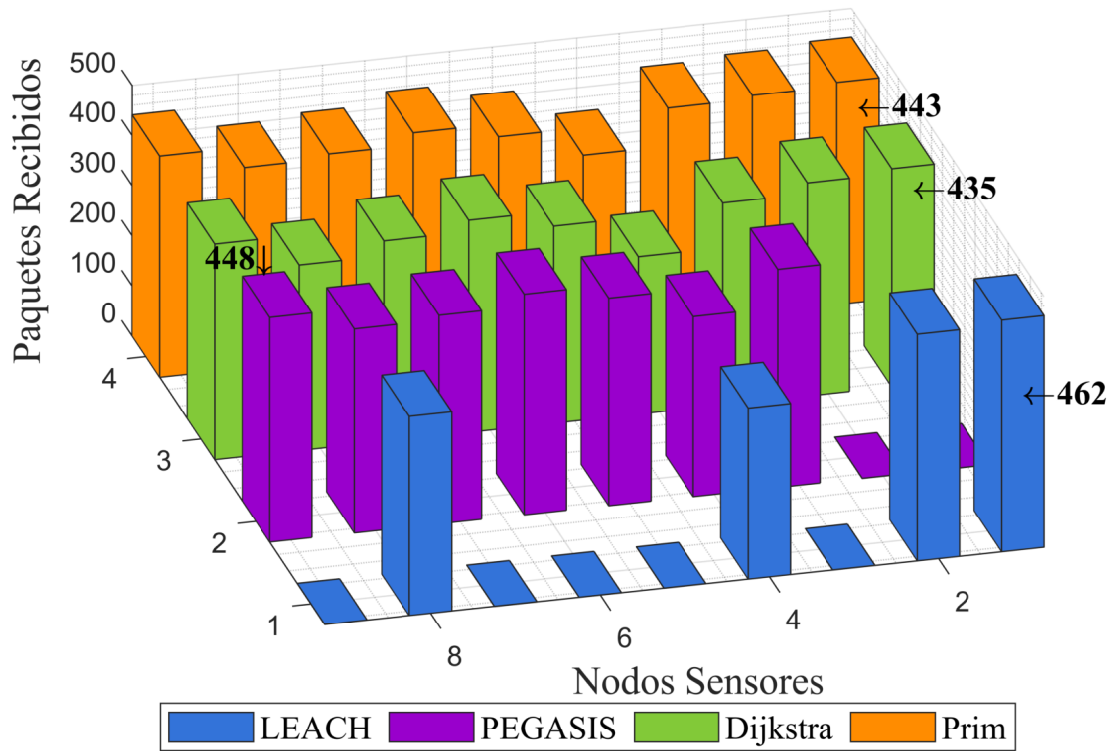


Figura 3.6: Paquetes recibidos por nodo.

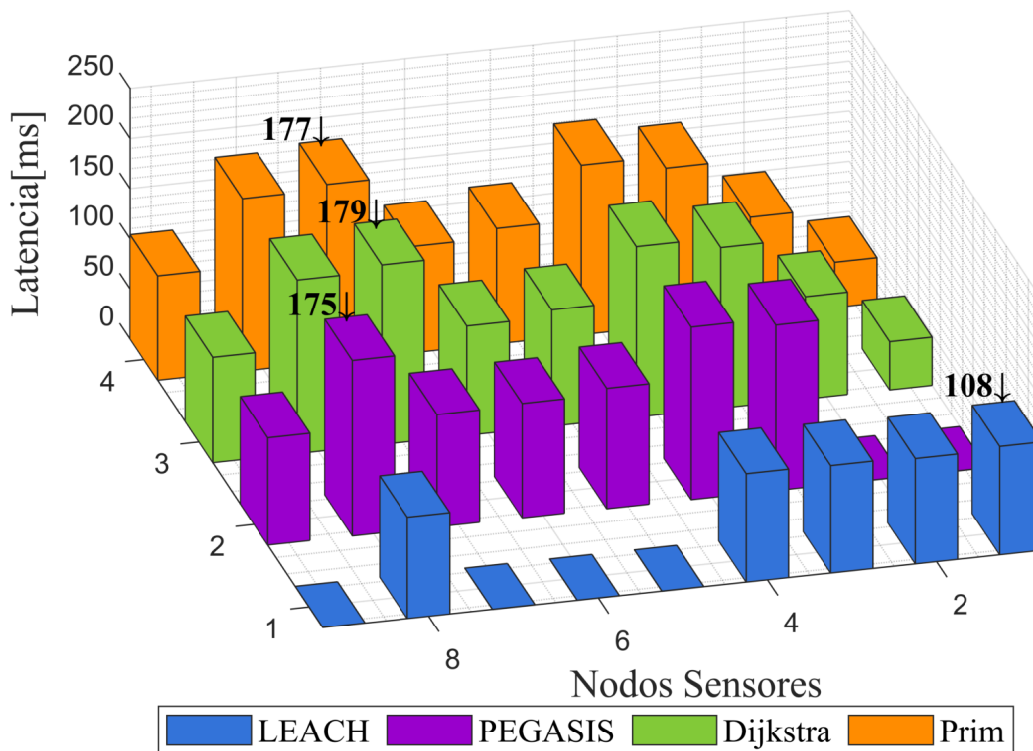


Figura 3.7: Latencia promedio en la transmisión de cada nodo.

Al colocar el coordinador en la segunda posición existe un incremento de paquetes recibidos por la estación base, en donde con el protocolo LEACH se recibieron 1813 paquetes, con PEGASIS 3126, con el algoritmo Dijkstra 3770 paquetes y con Prim un total de 3969 paquetes, es decir, el 51.8%, 89.31%, 83.77% y 88.2% de paquetes transmitidos respectivamente. Con esto observamos que con PEGASIS se disminuye la tasa de pérdida de paquetes deduciendo que la posición del nodo sensor 2 es más centralizada y por lo tanto disminuye los multisaltos. En la Figura 3.8 se presenta la cantidad de paquetes recibidos por cada nodo sensor transmisor durante el funcionamiento de cada protocolo de enrutamiento. Las etiquetas presentes en la figura indica la mayor cantidad de paquetes recibidos transmitidos por un nodo sensor en cada protocolo.

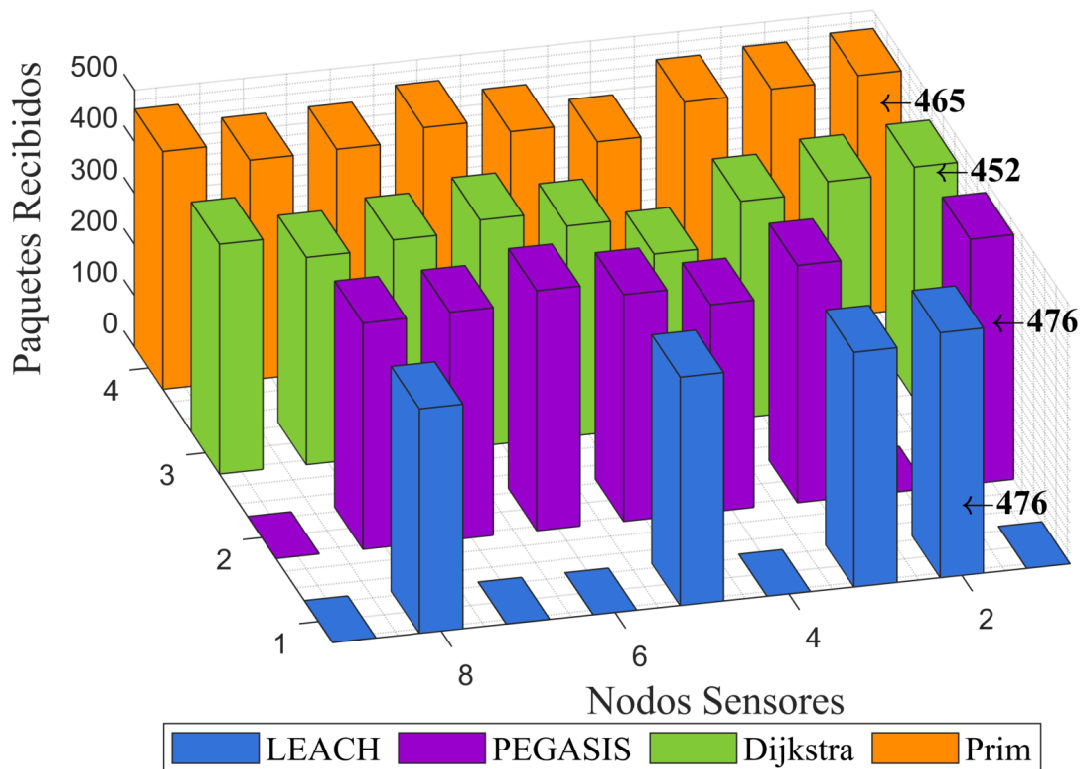


Figura 3.8: Paquetes recibidos por nodo.

Los beneficios de la centralización del coordinador se demuestran también con el análisis de la latencia la cual disminuye en función de los multisaltos que se deben realizar para la recepción del paquetes en el coordinador. Los protocolos más beneficiados son PEGASIS, Dijkstra y Prim, mientras que LEACH mantiene su estructura jerárquica y por ende la cantidad de multisaltos. La Figura 3.9 presenta la relación de la latencia promedio existente entre la transmisión de paquetes con cada nodo sensor agrupados de acuerdo al protocolo de enrutamiento implementado en la red. A su vez en esta figura se indica mediante etiquetas la mayor latencia existente en cada protocolo de enrutamiento y el nodo sensor con esta cantidad de latencia.

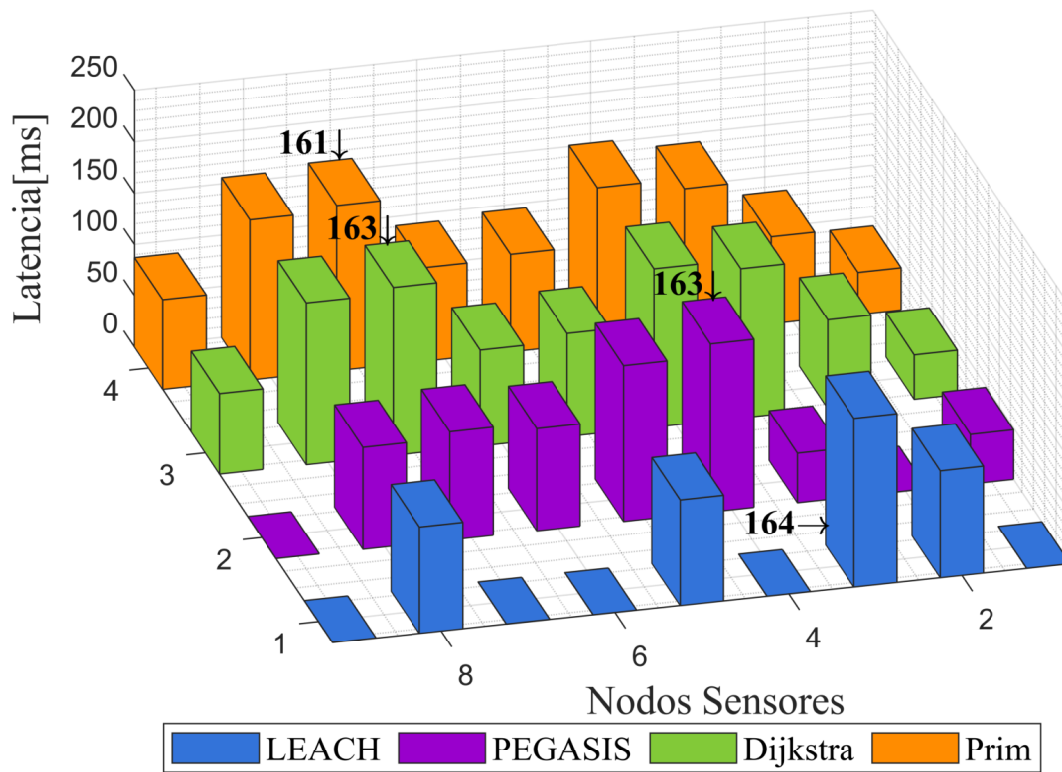


Figura 3.9: Latencia promedio en la transmisión de cada nodo.

Con estos resultados, se intuye que al cambiar de posición al coordinador se disminuyó tanto la latencia como la tasa de pérdida de paquetes frente al primer escenario donde el coordinador se ubicó por conveniencia. Por lo tanto, la centralización del coordinador determinada por el dimensionamiento de la red demuestra una mejora sustancial en el funcionamiento de los protocolos.

Capítulo 4

Conclusiones Recomendaciones y Trabajos Futuros

Conclusiones:

Con la investigación realizada, a través de los resultados mostrados, se llega a concluir que el dimensionamiento óptimo de redes puede mejorar métricas del funcionamiento de la red como es la disminución de la latencia y la tasa de pérdida de paquetes, por lo tanto es correcto afirmar que para el diseño de una [WSN](#) se debe analizar los sitios candidatos para obtener la mejor ubicación de los concentradores o estaciones base, de la forma que se optimicen los recursos de la red y costos de implementación. Siendo el objetivo de la [WSN](#) la transmisión de datos censados utilizando la menor cantidad de recursos, el dimensionamiento de red llega a ser una herramienta que permite optimizar estos recursos.

Considerando los protocolos de enrutamiento analizados, para la aplicación de una [WSN](#) en aplicaciones donde se tienen que transmitir varios datos en topologías jerárquicas como pudiese ser en el caso de medición inteligente, los protocolos recomendados en base al estudio son PEGASIS y Prim, los cuales demostraron tener una menor tasa de pérdida de paquetes siendo este un requisito para los sistemas con medidores inteligentes cuyo formato de transmisión es de un dato cada 15 minutos de acuerdo a los medidores inteligentes del mercado. Además, en concreto con el protocolo Prim se demostró que a pesar de la pérdida de conexión de los nodos su tasa de transmisión supera el 50 % indicando que es un protocolo estable y tolerante

a fallos. Si bien los protocolos con transmisión por multisalto tienen problemas de latencia, estos logran aumentar la vida útil de la red porque permiten a los nodos sensores utilizar menor potencia y por ende menor consumo energético en cada una de sus transmisiones.

Recomendaciones:

Según la aplicación de la WSN, se recomienda analizar las propiedades de los protocolos de enrutamiento para determinar los protocolos que cumplen las necesidades de la aplicación, considerando que los protocolos de enrutamiento plano como Dijkstra y Prim pueden llegar a ser más estables y tolerante a fallos ya que pueden generar diferentes rutas para la transmisión de los paquetes, por otro lado los protocolos jerárquicos como los analizados LEACH y PEGASIS disminuyen la cantidad de multisaltos reduciendo la probabilidad de pérdida de paquetes y latencia como es el caso de LEACH.

Para el desarrollo de una red ZigBee, se recomienda la configuración de los modos en el modo API de tal forma que los datos a transmitir sean encapsulados en tramas que permitan adquirir información extra a la carga útil transmitida, como es la dirección MAC de origen o destino, la longitud de la trama y la suma de comprobación. Con esta información, se pueden implementar diferentes protocolos de enrutamiento en base a las direcciones MAC y en los microcontroladores se puede determinar con mayor facilidad el inicio de la trama para sustraer los datos transmitidos y también se puede comprobar si su contenido no fue afectado por los efectos del canal inalámbrico a través de la suma de comprobación.

Con el hardware implementado de la red se consiguió transmitir tramas con un a carga útil promedio de 30 bytes, donde 9 bytes son los datos de voltaje de línea adquiridos por los medidores inteligentes. En el contexto de que se necesite almacenar datos ya sea por una pérdida de conexión para transmitir en cuanto se pueda reconectar el nodo a la red o porque se planifique transmisiones de datos acumulados, el uso de tarjetas como de desarrollo como Arduino Uno que tienen menor capacidad de memoria para las variables y código del programa no es recomendado, mientras las tarjetas SMT32 muestran mucho mejores prestaciones; por tanto, deberá considerarse

la capacidad máxima de almacenamiento al momento de elegir el microcontrolador.

Antes de realizar el estudio del dimensionamiento de la red, es necesario determinar la capacidad de funcionamiento simultaneo de los nodos, es decir, la cantidad de nodos que pueden transmitir al mismo tiempo tramas hacia el concentrador de tal forma que no se pierda información. Esta capacidad dependerá de la ventana de tiempo de transmisión de cada nodo. Este parámetro es importante de identificar ya que a partir de este se podrá determinar la funcionalidad de la red de acuerdo a las variables censadas y transmitidas.

Trabajos Futuros:

Se plantea el análisis del escenario optimizado considerando la implementación de tres coordinadores e incrementando la densidad de la red para estudiar métricas como la tasa de perdida de paquetes, latencia, vida útil de la red y eficiencia energética. Además, se buscaría la implementación de la red en conjunto con el sistema de medición inteligente para iniciar con pruebas de funcionamiento y tráfico de la red en tiempo real.

También, queda por verificar que al optimizar los posibles enlaces y en consecuencia mejorar el proceso del cálculo de rutas independiente del protocolo usado, se puede obtener una disminución del consumo energético ya que el tiempo de proceso se ve reducido.

Glosario

6LoWPAN IPv6 Sobre Redes de Área Personal Inalámbricas de Baja Potencia – IPv6 over Low power Wireless Personal Area Networks.

AMI Infraestructura de Medición Avanzada – Advanced Measurement Infrastructure.

CH Jefe de Grupo – Clusterhead.

DAP Punto de Agregación de Datos – Data Aggregation Point.

IEEE Instituto de Ingenieros Eléctricos y Electrónicos – Institute of Electrical and Electronics Engineers.

IoT Internet de las Cosas – Internet of Things.

IP Protocolo de Internet – Internet Protocol.

IPv6 Protocolo de Internet versión 6 – Internet Protocol version 6.

LAN Red de Área Local – Local Area Network.

LoRaWAN Red de Área Amplia de Largo Alcance – Long Range Wide Area Network.

LPWAN Red de Área Amplia de Bajo Consumo – Low Power Wide Area Network.

MAC Control de Acceso al Medio – Media Access Control.

MAN Red de Área Metropolitana – Metropolitan Area Network.

MST Árbol de Mínima Expansión – Minimum Spanning Tree.

OUI Identificador Único Organizativo – Organizationally Unique Identifier.

PAN Red de Área Personal – Personal Area Network.

QoS Calidad de Servicio – Quality of Service.

RSS Fuerza de la Señal Recibida – Received Signal Strength.

TCP Protocolo de Control de Transmisión – Transmission Control Protocol.

TDM Multiplexación por División de Tiempo – Time-Division Multiplexing.

UART Transmisor-Receptor Asíncrono Universal – Universal Asynchronous Receiver-Transmitter.

UDP Protocolo de Datagramas de Usuario – User Datagram Protocol.

WAN Red de Área Amplea – Wide Area Network.

WSN Redes de Sensores Inalámbricas – Wireless Sensor Network.

Bibliografía

- [1] U. Sendler, *The internet of things: Industrie 4.0 unleashed*. Springer, 2017.
- [2] M. Sonthalia, A. Jha, U. Gupta, and J. Thyagarajan, "A real time implementation of hierarchical routing protocol for IoT based wireless sensor network," *2019 International Conference on Wireless Communications, Signal Processing and Networking, WiSPNET 2019*, pp. 512–516, 2019.
- [3] L. Mainetti, L. Patrono, and A. Vilei, "Evolution of wireless sensor networks towards the Internet of Things: A survey," *2011 International Conference on Software, Telecommunications and Computer Networks, SoftCOM 2011*, pp. 16–21, 2011.
- [4] L. Atzori, A. Iera, and G. Morabito, "From "smart objects"to "social objects": The next evolutionary step of the internet of things," *IEEE Communications Magazine*, vol. 52, no. 1, pp. 97–105, 2014.
- [5] M. Zorzi, A. Gluhak, S. Lange, and A. Bassi, "From Today's INTRANet of Things to a Future Internet of Things: A Wireless-and Mobility-Related View," *IEEE Internet of Things Magazine*, vol. 2, no. October, pp. 17–23, 2017.
- [6] J. Tan and S. G. Koo, "A survey of technologies in internet of things," *Proceedings - IEEE International Conference on Distributed Computing in Sensor Systems, DCOSS 2014*, pp. 269–274, 2014.
- [7] A. Peralta-Sevilla, E. Inga, R. Cumbal, and R. Hincapié, "Optimum deployment of FiWi Networks using wireless sensors based on Universal Data Aggregation Points," *2015 IEEE Colombian Conference on Communications and Computing, COLCOM 2015 - Conference Proceedings*, 2015.

- [8] A. Shahraki, A. Taherkordi, O. Haugen, and F. Eliassen, "A Survey and Future Directions on Clustering: From WSNs to IoT and Modern Networking Paradigms," *IEEE Transactions on Network and Service Management*, vol. 18, no. 2, pp. 2242–2274, 2021.
- [9] S. Liu, Y. Tang, and Y. Liu, "A survey of transport protocol for wireless sensor networks," *2012 2nd International Conference on Consumer Electronics, Communications and Networks, CECNet 2012 - Proceedings*, vol. 2, pp. 2338–2341, 2012.
- [10] H. A. Al-Kashoash and A. H. Kemp, "Comparison of 6LoWPAN and LPWAN for the Internet of Things," *Australian Journal of Electrical and Electronics Engineering*, vol. 13, no. 4, pp. 268–274, 2016.
- [11] Z. Shelby and C. Bormann, *6LoWPAN : The Wireless Embedded Internet Companion Lecture Slides The Book 6LoWPAN : The Wireless Embedded Internet*, 2009, no. c. [Online]. Available: <http://library.books24x7.com.zorac.aub.aau.dk/toc.aspx?site=G45VZ&bookid=36453>
- [12] R. Saxena, V. Rishiwal, and O. Singh, "Performance Evaluation of Routing Protocols in Wireless Sensor Networks," *Proceedings - 2018 3rd International Conference On Internet of Things: Smart Innovation and Usages, IoT-SIU 2018*, no. 2, pp. 1–6, 2018.
- [13] A. M. Palma, "Análisis de protocolos de enrutamiento para redes de sensores inalámbricas," 2009.
- [14] M. Sana and L. Noureddine, "Multi-hop energy-efficient routing protocol based on Minimum Spanning Tree for anisotropic Wireless Sensor Networks," *Proceedings of International Conference on Advanced Systems and Emergent Technologies, IC_ASET 2019*, pp. 209–214, 2019.
- [15] C. Ganán, E. Inga, and R. Hincapié, "Óptimo despliegue y enrutamiento de UDAP para infraestructura de medición avanzada basada en el algoritmo MST Optimal deployment and routing geographic of UDAP for advanced metering infrastructure based on MST algorithm," *Revista chilena de ingeniería*, vol. 25, no. 1, pp. 106–115, 2017.
- [16] R. A. Michael, Z. P. Carlos, and I. O. Juan, "Optimum design and dimensioning model of a Mesh-WiFi network for emergency services in protected areas," *2017 IEEE 2nd Ecuador Technical Chapters Meeting, ETCM 2017*, vol. 2017-January, pp. 1–6, 2018.

- [17] L. Garcia, J. M. Jiménez, M. Taha, and J. Lloret, "Wireless Technologies for IoT in Smart Cities," *Network Protocols and Algorithms*, vol. 10, no. 1, p. 23, 2018.
- [18] D. International, "Xbee® / xbee-pro® zb smt rf modules: Zigbee rf module datasheet," 2010. [Online]. Available: <https://pdfslide.net/documents/datasheet-xbee.html>
- [19] J. Inga, E. Inga, C. Gómez, and R. Hincapié, "Evaluación de la Infraestructura de Medición y la Respuesta de la Demanda," *Revista Técnica .Energía*, vol. 12, no. 1, pp. 262–269, 2016.
- [20] S. S. Refaat, O. Ellabban, S. Bayhan, H. Abu-Rub, F. Blaabjerg, and M. M. Begovic, *Smart Grid and Enabling Technologies (Wiley - IEEE)*. John Wiley & Sons, 2021.
- [21] J. Inga-Ortega, E. Inga-Ortega, C. Gómez, and R. Hincapié, "Electrical load curve reconstruction required for demand response using compressed sensing techniques," *2017 IEEE PES Innovative Smart Grid Technologies Conference - Latin America, ISGT Latin America 2017*, vol. 2017-January, pp. 1–6, 2017.
- [22] E. Inga, R. Hincapié, C. Suárez, and G. Arévalo, "Shortest path for optimal routing on Advanced Metering Infrastructure using cellular networks," *2015 IEEE Colombian Conference on Communications and Computing, COLCOM 2015 - Conference Proceedings*, 2015.
- [23] E. Inga, J. Inga, and A. Ortega, "Novel approach sizing and routing of wireless sensor networks for applications in smart cities," *Sensors*, vol. 21, no. 14, pp. 1–17, 2021.
- [24] J. N. Al-Karaki and A. E. Kamal, "Routing techniques in wireless sensor networks: A survey," *IEEE Wireless Communications*, vol. 11, no. 6, pp. 6–27, 2004.
- [25] J. Zheng and A. Jamalipour, *Wireless Sensor Networks: A Networking Perspective*, 2008.
- [26] B. Raja, R. Rajakumar, P. Dhavachelvan, and T. Vengattaraman, "A survey on classification of network structure routing protocols in wireless sensor networks," *2016 IEEE International Conference on Computational Intelligence and Computing Research, ICCIC 2016*, pp. 4–8, 2017.
- [27] H. M. Ammari, *The Art of Wireless Sensor Networks*, 2014, vol. 1.
- [28] R. M. Abarca, *Guide to Wireless Sensor Network*, 2021.
- [29] L. Vinet and A. Zhedanov, *Wireless Networks Sensor Concepts, Applications, Experimentation and Analysis*, 2011, vol. 44, no. 8.

- [30] T. D. Nguyen, J. Y. Khan, and D. T. Ngo, "A Distributed Energy-Harvesting-Aware Routing Algorithm for Heterogeneous IoT Networks," *IEEE Transactions on Green Communications and Networking*, vol. 2, no. 4, pp. 1115–1127, 2018.
- [31] N. Kaur, S. Verma, and Kavita, "A survey of routing protocols in wireless sensor networks," *International Journal of Engineering and Technology(UAE)*, vol. 7, no. 4.12 Special Issue 12, pp. 20–25, 2018.
- [32] K. Akkaya and M. Younis, "A survey on routing protocols for wireless sensor networks," *Ad Hoc Networks*, vol. 3, no. 3, pp. 325–349, 2005.
- [33] L. E. Agustín-Blas, S. Salcedo-Sanz, P. Vidales, G. Urueta, and J. A. Portilla-Figueras, "Near optimal citywide WiFi network deployment using a hybrid grouping genetic algorithm," *Expert Systems with Applications*, vol. 38, no. 8, pp. 9543–9556, 2011.
- [34] T. Wakefield, D. McNally, D. Bowler, and A. Mayne, *Introduction to mobile communications: technology, services, markets: technology, services, markets*. Auerbach publications, 2007.
- [35] G. National and H. Pillars, *Zigbee Wireless Networks and Transceivers*, 2008.
- [36] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," *Proceedings of the Annual Hawaii International Conference on System Sciences*, vol. 2000-January, no. c, pp. 1–10, 2000.
- [37] J. A. Fraire and E. L. Gasparini, "Centralized and Decentralized Routing Solutions for Present and Future Space Information Networks," *IEEE Network*, vol. 35, no. 4, pp. 110–117, 2021.
- [38] R. E. Tarjan, "Algorithm design," pp. 204–212, 1987.
- [39] T. H. Cormen, C. R. Leiserson, R. L. Rivest, and C. Stein, "Introduction to algorithms," p. 1180, 2003.
- [40] I. Sharma, R. Singh, and M. Khurana, "Comparative study of LEACH, LEACH-C and PEGASIS routing protocols for wireless sensor network," *Conference Proceeding - 2015 International Conference on Advances in Computer Engineering and Applications, ICACEA 2015*, pp. 842–846, 2015.

- [41] S. Lindsey and C. S. Raghavendra, "PEGASIS: Power-efficient gathering in sensor information systems," *IEEE Aerospace Conference Proceedings*, vol. 3, pp. 1125–1130, 2002.
- [42] M. Berkelaar, "Lp solve reference guide." [Online]. Available: <http://lpsolve.sourceforge.net/5.5/index.htm>~<http://web.mit.edu/lpsolve/doc/>