

Desarrollo de una red de internet de
las cosas para la comunicación de un
robot modular

Desarrollo de una red de internet de las cosas para la comunicación de un robot modular

Luis Fernando Pedraza Martínez
Henry Alberto Hernández Martínez
Ernesto Cadena Muñoz



*Los autores agradecen a la Universidad Distrital Francisco José de Caldas por
apoyar esta investigación y a los estudiantes del programa de Ingeniería en
Telecomunicaciones que se vincularon al proyecto.*



UD
Editorial

E2
ESPACIOS

© Universidad Distrital Francisco José de Caldas
© Oficina de Investigaciones
© Luis Fernando Pedraza Martínez, Henry Alberto Hernández Martínez,
Ernesto Cadena Muñoz

ISBN: 978-958-787-522-5
ISBN digital: 978-958-787-523-2
ISBN ePub: 978-958-787-524-9
Primera edición, septiembre del 2023

Líder Proyecto de Publicaciones
Rubén Eliécer Carvajalino C.

Gestión editorial
Andrés Delgado Darnalt

Corrección de estilo
Proceditor

Diagramación
Astrid Prieto Castillo

Editorial UD
Universidad Distrital Francisco José de Caldas
Carrera 24 n.º 34-37, Bogotá, D. C., Colombia
Teléfono: 6013239300 ext. 6202
Correo electrónico: publicaciones@udistrital.edu.co

*Sistema de Bibliotecas de la Universidad Distrital
Francisco José de Caldas*

Catalogación en la publicación (CEP)

Pedraza Martínez, Luis Fernando

Desarrollo de una red de internet de las cosas para la comunicación de un robot modular / Luis Fernando Pedraza Martínez, Henry Alberto Hernández Martínez, Ernesto Cadena Muñoz. -- Primera edición. -- Bogotá : Universidad Distrital Francisco José de Caldas, 2023.

126 páginas : gráficas, fotografías ; 24 cm. (Colección Espacios)

ISBN: 978-958-787-522-5 ISBN digital: 978-958-787-523-2

ISBN ePub: 978-958-787-524-9

1. Robótica -- Procesamiento de datos 2. Procesamiento de datos en tiempo real 3. Sistemas de transmisión de datos 4. Sistemas de control de autómatas 5. Comunicaciones digitales. I. Hernández Martínez, Henry Alberto, autor II. Cadena Muñoz, Ernesto, autor III. Serie.

629.892: CDD 21 edición

Todos los derechos reservados.

Esta obra no puede ser reproducida sin el permiso previo escrito de la Unidad de Publicaciones de la Universidad Distrital Francisco José de Caldas. Hecho en Colombia

Contenido

Listado de abreviaturas	17
Introducción	21
Objetivos del proyecto de investigación	25
Financiamiento del proyecto de investigación	25
Organización del libro de investigación	26
Capítulo 1. Antecedentes	27
Comunicación con ultrasonido	27
Comunicación con infrarrojo	28
Comunicación óptica en espacio libre	28
Comunicación Bluetooth	28
Comunicación ZigBee	29
Comunicación Wi-Fi	30
Comunicaciones móviles	30
Comunicación con redes de área extensa de baja potencia	31
Capítulo 2. Robots modulares	33
Descripción de los robots modulares	33
Categorías de sistema de los robots modulares	34
Estructura tipo cadena	35
Estructura tipo retícula	37
Estructura híbrida	38
Control de movimiento en robots modulares	41
Locomoción en robots	41
Sistemas de control	42

Técnicas de control de movimiento en robots modulares	43
Modos de navegación	45
Capítulo 3. Internet de las cosas (IoT)	49
Redes de área extensa de baja potencia	50
NarrowBand-IoT	51
LoRa	56
Telensa	60
Sigfox	62
Aplicaciones de internet de las cosas con redes de área extensa de baja potencia	65
Iluminación inteligente	66
Automatización y monitoreo	67
Sistemas de localización	68
Plataforma ThingSpeak	69
Parámetros de calidad de servicio	69
Pérdida de paquetes	70
Indicador de fuerza de la señal recibida	70
<i>Throughput</i>	70
<i>Jitter</i>	71
Capítulo 4. Metodología	73
Robot modular EMERGE	76
Algoritmo para la coordinación y control de los movimientos del robot	80
Sistema de comunicación de Internet de las cosas con una red de área extensa de baja potencia	83
Rata de bit en LoRa	83
Transceptor de internet de las cosas	84
Desarrollo del programa para el dispositivo que transmite información al robot	85
Desarrollo del programa para el dispositivo que recibe información en el robot	86
Configuración y almacenamiento de la información en la plataforma ThingSpeak	87
Capítulo 5. Evaluación y análisis de resultados	89
Evaluación y análisis de los parámetros de calidad de servicio del sistema de comunicaciones IoT	93

Conclusiones	103
Referencias	107
Sobre los autores	125

Listado de figuras

Figura 1.	Diagrama de clasificación de robots modulares según su estructura	34
Figura 2.	Robot modular ModRED: configuración y estructura	35
Figura 3.	Robot modular Microtub y sus partes	36
Figura 4.	Robot Odín: módulos (a) y estructura (b)	37
Figura 5.	Movimiento pivotante del robot modular M-Blocks: pivote de los módulos (a) y estructura (b)	38
Figura 6.	Partes de un módulo y morfología tipo carro del robot modular ATRON: módulo (a) y morfología (b)	39
Figura 7.	Autosoldado: morfologías del robot SolderCubes	40
Figura 8.	Comparación de los rangos de alcance, consumo de energía y velocidad de transmisión de datos en distintas tecnologías inalámbricas	51
Figura 9.	Arquitectura de red de NB-IoT	53
Figura 10.	Protocolos de NB-IoT para el control de acceso al medio (plano de control y plano de usuario)	55
Figura 11.	Estructura de red LoRaWAN	57
Figura 12.	Arquitectura y protocolos usados en LoRaWAN	57
Figura 13.	Estructura de un paquete LoRa y los factores de la modulación en el paquete	58
Figura 14.	Estación base UNB Telensa	60
Figura 15.	Controladores para alumbrado público	61
Figura 16.	Urban IQ para ciudades inteligentes	61
Figura 17.	Arquitectura general de Sigfox	63
Figura 18.	Principio de transmisión única o múltiple en Sigfox	64
Figura 19.	Dispositivos para la prueba de una red LoRa: <i>gateway</i> de referencia LoRa IoT modelo 0X80400AC (a) y nodos de referencia LoRa (b)	66
Figura 20.	Dispositivo instalado para monitoreo de sistema de riego con tecnología LPWAN	68

Figura 21.	Esquema general del sistema desarrollado	75
Figura 22.	Diagrama de flujo del sistema desarrollado	76
Figura 23.	Robot modular EMERGE: ejemplo de ensamble (a) y módulo del robot (b)	77
Figura 24.	Controlador del robot EMERGE	78
Figura 25.	Rangos de movimiento del actuador de un módulo	79
Figura 26.	Paquete del parámetro de movimiento	80
Figura 27.	Arquitectura del sistema de comunicación para enviar parámetros de movimiento al robot EMERGE	82
Figura 28.	Segmento del archivo de texto plano generado en la nube	82
Figura 29.	Dispositivo Wi-Fi LoRa 32 marca Heltec	85
Figura 30.	Diagrama de flujo para programar el dispositivo que transmite información al robot	86
Figura 31.	Diagrama de flujo para programar el dispositivo que recibe información en el robot	87
Figura 32.	Inicialización del programa para el dispositivo que transmite información hacia el robot	90
Figura 33.	Ejemplo de la segunda parte del programa para el dispositivo que transmite información hacia el robot	90
Figura 34.	Ejemplo del programa para el dispositivo que recibe información en el robot	91
Figura 35.	Almacenamiento y visualización de la posición angular de los módulos del robot en la plataforma ThingSpeak	92
Figura 36.	Módulos del robot EMERGE moviéndose a la última posición almacenada y descargada de la nube	92
Figura 37.	<i>Throughput</i> para BW de 125 kHz	97
Figura 38.	<i>Throughput</i> para BW de 250 kHz	97
Figura 39.	Rata de bit para BW de 125 kHz	97
Figura 40.	Rata de bit para BW de 250 kHz	98
Figura 41.	Porcentaje de paquetes perdidos para BW de 125 kHz	98
Figura 42.	Porcentaje de paquetes perdidos para BW de 250 kHz	98
Figura 43.	<i>Jitter</i> para BW de 125 kHz	99
Figura 44.	<i>Jitter</i> para BW de 250 kHz	99
Figura 45.	Nivel de potencia recibida para BW de 125 kHz	99
Figura 46.	Nivel de potencia recibida para BW de 250 kHz	100

Listado de tablas

Tabla 1.	Otros ejemplos de robots modulares	40
Tabla 2.	Especificaciones técnicas de LoRaWAN para diferentes regiones	59
Tabla 3.	Características de las principales tecnologías basadas en LPWAN	73
Tabla 4.	Segmento de la rutina de control de movimiento	79
Tabla 5.	Parámetros de QoS para una distancia de transceptores de 20 m	94
Tabla 6.	Parámetros de QoS para una distancia de transceptores de 100 m	95
Tabla 7.	Parámetros de QoS para una distancia de transceptores de 150 m	96

Listado de abreviaturas

Abreviatura	Término
2G	Segunda generación
3D-UNB	Triple diversidad ultrabanda angosta
3G	Tercera generación
3GPP	Proyecto de asociación de tercera generación
4G	Cuarta generación
5G	Quinta generación
6LowPAN	IPv6 sobre redes de área personal inalámbricas de potencia baja
AES	Estándar de cifrado avanzado
bps	Bits por segundo
BPSK	Modulación por desplazamiento de fase binaria
BR	Rata básica
BW	Ancho de banda
CAN	<i>Controller area network</i>
CE	Mejora de cobertura
CIoT	Internet de las cosas celular
CPG	Generadores centrales de patrones
CR	Rata de código

CSS	<i>Chirp spread spectrum</i>
dB	Decibelio
dBm	Decibelio-milivatio
DC	Corriente directa
DL	Enlace de bajada
DOF	Grados de libertad
EDR	Rata de datos mejorada
EMERGE	<i>Easy modular embodied robot generator</i>
EPS	Sistema de paquetes evolucionado
FSK	Modulación por desplazamiento de frecuencia
GHz	Gigahertz
GPRS	Servicio general de paquetes vía radio
GSM	Sistema global para las comunicaciones móviles
Hz	Hertz
IEEE	Instituto de Ingenieros Eléctricos y Electrónicos
IoRT	Internet de las cosas robóticas
IoT	Internet de las cosas
IP	Protocolo de Internet
IR	Comunicación infrarroja
ISM	Bandas industriales, científicas y médicas
kHz	Kilohertz
LE	Baja energía
LoRA	Largo alcance
LoRaWAN	Red de área amplia de largo alcance
LOS	Línea de vista

LPWAN	Red de área extensa de baja potencia
LTE	<i>Long term evolution</i>
MHz	Megahertz
MME	Entidad de gestión de movilidad
NAS	Estrato sin acceso
NB-IoT	IoT de banda angosta
NFC	Comunicación de campo cercano
PGW	Puerta de enlace de red de paquetes de datos
PoE	Potencia sobre Ethernet
PSK	Modulación por desplazamiento de fase
QoS	Calidad de servicio
QPSK	Modulación por desplazamiento de fase por cuadratura
RACH	Procedimiento de canal de acceso aleatorio
RC	Radio cognitiva
RSSI	Indicador de fuerza de la señal recibida
SDR	Radio definida por <i>software</i>
SF	Factor de ensanchamiento
SGW	Puerta de enlace de servicio
Sub-GHz	Subgigahertz
UE	Equipo de usuario
UIT	Unión Internacional de Telecomunicaciones
UL	Enlace de subida
UNB	Ultrabanda angosta
USB	Bus Serie Universal
Wi-Fi	Tecnología de fidelidad inalámbrica

Introducción

Las tecnologías de la información forman parte de la revolución de la sociedad, ya que son capaces de promover cambios sociales y económicos. Este tipo de soluciones se convierte en una herramienta de apoyo a los desarrolladores de sistemas automatizados que ofrece diversidad, calidad y bajos costos.

La necesidad de expandir las redes para transferir información de relevancia entre diferentes dispositivos permite el avance cada vez más dinámico de las tecnologías vinculadas a dicho proceso. La gran cantidad de tecnologías hace difícil que el usuario pueda seleccionar la más adecuada para sus necesidades; en algunos casos, la selección errada de los dispositivos suscita vulnerabilidades de seguridad o que la información no se transmita de manera conveniente, lo que ocasiona retrasos y errores en la comunicación y en el envío de paquetes.

Con la llegada de las tecnologías de internet de las cosas (IoT) y la creciente demanda para que los dispositivos electrónicos tengan cierta autonomía en la realización de sus tareas, se hace esencial evaluar el uso de las tecnologías basadas en IoT para la transmisión de información y la mejora de sistemas que funcionen en tiempo continuo. Es el caso, por ejemplo, de los robots modulares.

El concepto de IoT lo utilizó el británico Kevin Ashton [1] para hacer referencia a la interconexión digital de objetos de uso cotidiano con internet con el objeto de recolectar y enviar información de distintas variables a determinados centros de datos, realizar el procesamiento de esta y generar acciones entre los objetos conectados a la red [2, 3].

Las tecnologías de IoT pueden desarrollarse en una variedad de aspectos de la vida cotidiana, donde las empresas y los gobiernos pueden beneficiarse de dicha tecnología. Es posible servirse del IoT en la industria para automatizar y controlar los procesos. También se puede usar para integrar las operaciones de la empresa en la cadena de suministro y crear un flujo de datos continuo desde los pedidos de los clientes hasta las operaciones y el cumplimiento de los mismos. El IoT se puede utilizar para controlar todo el proceso de producción mediante el empleo de la realidad aumentada y el monitoreo a distancia con el fin de evitar el envío de personal a zonas remotas para medir parámetros de operación [4, 5]. Lo anterior constituye un desafío para la industria colombiana.

Una solución parcial a esta limitación es la implementación de sistemas robóticos para automatizar procesos, lo que ha traído enormes cambios socioeconómicos en los últimos tiempos. Los robots industriales se han implementado y utilizado ampliamente en todo tipo de industrias para ejecutar tareas repetitivas, tediosas, críticas o peligrosas, como ensamble de productos, pintura de automóviles, empaque de cajas y soldadura, entre otras. Estos robots han tenido mucho éxito en las aplicaciones industriales gracias a su alta precisión, resistencia y velocidad, y se han integrado con las tecnologías de red existentes para ampliar las funcionalidades de estos robots [6, 7].

Las comunidades de IoT y robótica, que están conformadas por empresas líderes en estas áreas, se fomentan con diferentes objetivos, entre ellos, apoyar los servicios de información para la detección, el seguimiento y el monitoreo; así como la producción, la interacción y el comportamiento autónomo. Por esta razón, se afirma que la creación de una internet de las cosas robóticas (IoRT) que combine los resultados de las dos comunidades tendrá un gran valor agregado [8]. Los primeros pasos en la convergencia IoT-robótica se observan en paradigmas de control de robots heterogéneos y distribuidos, como los enfoques de robótica ubicua [9, 10] y robótica en la nube [11]-[13].

El Comité Técnico de Robots en Red de la Sociedad de Robótica y Automatización del Instituto de Ingenieros Eléctricos y Electrónicos (IEEE) definió el sistema robótico en red como una colección de dispositivos robóticos que están conectados por medio de una red de comunicación por

cable o inalámbrica. Las aplicaciones robóticas en red [14, 15] se pueden clasificar como robots teleoperados, es decir, robots posicionados de forma remota y controlados por los comandos enviados por un humano mediante la red de comunicación, o como un sistema de múltiples robots, esto es, como un grupo de robots en red distribuidos para realizar una tarea asignada, intercambiando datos de detección e información a través de la red de comunicación de manera cooperativa. La robótica en red adolece de limitaciones físicas inherentes, como ejecución de instrucciones a baja velocidad, memoria pequeña, latencia de la red, calidad de servicio variable, tiempo de inactividad y falta de inteligencia [7, 16].

Las limitaciones han motivado a los investigadores a pensar en una nueva forma de sistemas robóticos eficientes, es decir, “robótica en la nube”. La robótica en la nube puede describirse como un sistema que adapta la infraestructura de “computación en la nube” [17]; así, aumenta su poder de procesamiento y datos para respaldar su funcionamiento [18]. Eso significa que no toda la detección, la computación y la memoria están integradas en un solo sistema autónomo como lo estaba en el caso de la robótica en red. Los sistemas robóticos en la nube a menudo incluyen una parte de su capacidad de procesamiento local para respuestas de baja latencia cuando el acceso a la red no está disponible o no es confiable, es decir, sin conexión. Un ejemplo de robótica en la nube es el automóvil autónomo de Google que utiliza los mapas, las imágenes y otra información relevante de Google recopilada por los satélites y las fuentes generadas por personas para facilitar la localización. Aunque la robótica en la nube se beneficia del análisis del *big data*, la computación en la nube, la computación humana y el aprendizaje colaborativo de robots tiene varios problemas como interoperabilidad, heterogeneidad, latencia variable de la red, seguridad, administración de múltiples robots, diseño de infraestructura común, calidad de servicio (QoS) y estandarización [11, 19]. Por las virtudes inherentes del IoRT se prevé que se superarán los problemas mencionados anteriormente, lo que conducirá a redes robóticas más inteligentes, colaborativas, heterogéneas, eficientes, autoadaptativas, conscientes del contexto y más económicas [7, 8, 20].

En Colombia el uso de varias de estas tecnologías está en auge debido a que las personas y las empresas se encuentran en una etapa exploratoria que

les permitirá vincular este tipo de tecnologías al desarrollo de sus actividades cotidianas. Estas prácticas, en su mayoría, se han empezado a emplear en el sector productivo para verificar y validar el estado de un producto; sin embargo, la utilización de estas tecnologías para conocer el estado de un sistema a partir de la información de los sensores es aún básico. Las ventajas de implementar un sistema realimentado a partir de la información de los sensores es la reducción del margen de error en el funcionamiento del sistema.

Como se observa, la robótica ha sido ampliamente investigada desde diferentes áreas del conocimiento, dada la cantidad de ventajas que representa la automatización parcial o total de un proceso. No obstante, la autonomía de un robot modular aún es un tema en estudio, ya que las estructuras de múltiples módulos están sujetas a fallos de *hardware* durante su funcionamiento continuo, o de *software*, cuando hay ausencia de suministro eléctrico para mantener de forma coordinada los movimientos que genera el controlador del robot. Por lo tanto, se hace necesario proponer una solución parcial a esta problemática, a través de un controlador remoto que le comunique al robot los parámetros de control de movimiento, así se presenten fallas durante su funcionamiento continuo. Lo anterior conduce a plantearse los siguientes interrogantes:

- ¿Cómo utilizar una tecnología basada en IoT para comunicar los parámetros de un robot de múltiples módulos?
- ¿Cómo medir los parámetros de QoS de la tecnología IoT utilizada?

La solución a estas preguntas implica el desarrollo de una red de comunicaciones con dispositivos de IoT para supervisar y coordinar un robot de múltiples módulos. Para este caso se aprovecha una tecnología emergente para la red de comunicaciones IoT en el área de la robótica, como lo es la red de área extensa de baja potencia (LPWAN). Este tipo de tecnología es adecuada para comunicar el robot de múltiples módulos de este proyecto, ya que no se requiere una alta velocidad de transmisión; además, es de largo alcance y bajo consumo energético.

Por tanto, los principales aportes de esta investigación consisten en el uso de la tecnología de LPWAN como red de comunicaciones para un robot modular de hasta cinco módulos, y la medición y análisis de los parámetros

de QoS para este tipo de red de comunicaciones. Es así que el texto está orientado a personas que desean profundizar en desarrollos de sistemas IoT aplicados en la robótica.

Este tipo de proyectos conlleva el estudio de nuevas alternativas para la comunicación de los robots y la interoperabilidad con otras redes para acceder a la información, que es una de las propuestas de las redes de nueva generación para que organismos como la Unión Internacional de Telecomunicaciones (UIT) emitan recomendaciones en lo que han llamado “redes futuras”.

Objetivos del proyecto de investigación

El objetivo general de este proyecto de investigación es desarrollar un esquema de comunicación con base en una tecnología de IoT para supervisar y coordinar las tareas de un robot de múltiples módulos. Los objetivos específicos son los siguientes:

- Implementar un algoritmo de control de manera local que permita realizar tareas de supervisión y control en un robot de múltiples módulos.
- Ejecutar un sistema de comunicaciones de IoT para transmitir y visualizar los parámetros de control del robot con un usuario a través de la red de internet.
- Analizar parámetros de QoS de la comunicación entre el robot y el usuario conectado a través de la red de internet.

Financiamiento del proyecto de investigación

El presente libro de investigación es producto de los resultados del proyecto de investigación —que se presentó a la convocatoria de proyectos sin recursos financieros— denominado “Desarrollo de un esquema de comunicación con base en una tecnología de internet de las Cosas para supervisar y coordinar las tareas de un robot de múltiples módulos”, institucionalizado por el Consejo de Facultad de la Facultad Tecnológica y el Centro de Investigaciones y Desarrollo Científico (CIDC) de la Universidad Distrital Francisco José de Caldas, en el 2018. El proyecto fue realizado con los recursos humanos, materiales y físicos que posee el grupo de investigación GIDENUTAS dentro de la universidad.

Organización del libro de investigación

El libro está estructurado como sigue. En el capítulo 1 se presentan los antecedentes de los sistemas de comunicación usados en los robots. En el capítulo 2 se describen los robots modulares. En el capítulo 3 se explica el IoT en lo que se refiere a las redes de área extensa de baja potencia. En el capítulo 4 se especifica la metodología de la investigación de este trabajo y en el capítulo 5 se exhiben los resultados. Al final se incluyen las conclusiones.

Capítulo 1

Antecedentes

Los sistemas de comunicación aplicados en la robótica han evolucionado con el desarrollo de las nuevas tecnologías; sin embargo, su utilidad depende de la aplicación y de la necesidad del robot. Entre estas necesidades se destacan la cobertura, la fiabilidad y el tiempo de respuesta. A continuación se presentan los antecedentes de las tecnologías de comunicaciones inalámbricas que mayoritariamente se han usado para transmitir la información de sistemas robóticos.

Comunicación con ultrasonido

La comunicación ultrasónica requiere transductores para convertir las señales eléctricas en ondas ultrasónicas. Las ondas ultrasónicas son ondas mecánicas, cuya atenuación es proporcional a la distancia y depende del medio de propagación. Como resultado de esto, las ondas ultrasónicas son adecuadas para comunicaciones de corto alcance [21].

Las aplicaciones de las comunicaciones con ultrasonido en la robótica son extensas; por ejemplo, el uso de robots para efectuar exámenes médicos de ultrasonido ha ido creciendo en la última década. En [22] se define el sistema de ultrasonido robótico como la combinación de imágenes de ultrasonido con un sistema robótico en intervenciones médicas. Aquí se destacan los dispositivos de diagnóstico extracorporal, los sistemas de guía de agujas y los sistemas quirúrgicos intraoperatorios. Dentro de las recientes aplicaciones sobresale el uso de robots ultrasónicos para tomar ecografías a distancia y, así, minimizar el riesgo por contagio del

COVID-19 [23]. La comunicación ultrasónica también se emplea en robots submarinos, especialmente en el seguimiento del entorno submarino, la vigilancia de apoyo y la observación militar y oceánica [24, 25].

Comunicación con infrarrojo

La comunicación con infrarrojos utiliza la transmisión de las ondas de luz a través de la propagación en el espacio libre en la banda infrarroja [26]. Lo anterior conlleva que la distancia de transmisión sea de corto alcance y la señal no atraviese obstáculos. En la robótica se usa para comunicar enjambres de robots [27, 28], que consisten en un gran número de robots simples.

Comunicación óptica en espacio libre

En estos sistemas de comunicación el espacio libre actúa como un canal de comunicación entre transceptores que están en línea de vista (LOS) para la transmisión de señales ópticas. El canal puede ser la atmósfera, el espacio o el vacío, cuyas características determinan la transmisión y recepción de señales ópticas para diseñar sistemas de comunicación fiables y eficientes. Este tipo de comunicación ofrece una alta velocidad para satisfacer la enorme demanda del creciente tráfico de datos [29].

Esta tecnología se ha aprovechado para la comunicación de equipos de robots móviles donde se presentaron resultados con un rendimiento satisfactorio [30]. También para comunicar enjambres de robots terrestres [31, 32] y submarinos [33], así como para robots que eliminan artefactos explosivos [34].

Comunicación Bluetooth

La tecnología de radio Bluetooth fue desarrollada por L. M. Ericsson en 1994. En la actualidad es usada ampliamente para crear redes de área personal, permitiendo a los usuarios conectarse y desconectarse dinámicamente al entrar y salir del área de cobertura. Bluetooth opera de dos formas: a tasa básica/rata de datos mejorada (BR/EDR) o con baja energía (LE) [35].

El Bluetooth BR/EDR funciona en el rango de frecuencias 2.4-2.4835 GHz, con 79 canales espaciados a 1 MHz más las bandas de guarda. Se definen

dos modos de transmisión de datos: el modo BR, que utiliza una modulación de frecuencia binaria para minimizar la complejidad del transceptor, y el modo EDR, que usa modulación por desplazamiento de fase (PSK) con dos variantes adicionales [35, 36].

La capa física de Bluetooth LE es una versión reducida y optimizada de Bluetooth BR. Mientras que el BR salta sobre 79 canales e implementa el descubrimiento en 32 canales, el LE toma 37 canales, al tiempo que realiza el descubrimiento de dispositivos en tres canales. Dado que LE pasa por menos canales mientras efectúa el descubrimiento, se logra un establecimiento de conexión rápido [36].

En la comunicación de robots, la tecnología Bluetooth se ha empleado para la comunicación en múltiples aplicaciones, como lo son la interacción con dispositivos celulares [37, 38], las labores de asistencia y monitorización [39], el control de brazos robóticos [40] y los robots autónomos [41], entre otros.

Comunicación ZigBee

ZigBee e IEEE 802.15.4 son protocolos basados en estándares que proporcionan la infraestructura de red necesaria para las aplicaciones de redes de sensores inalámbricos. 802.15.4 define las capas físicas y de control de acceso al medio, y ZigBee define las capas de red y aplicación [42].

Para las aplicaciones de redes de sensores los requisitos se apoyan en una batería de larga duración, bajo costo, tamaño reducido y redes en malla con el fin de soportar la comunicación entre una gran cantidad de dispositivos en un entorno interoperable y de múltiples aplicaciones [42].

La capa física opera en dos rangos de frecuencia: 868/915 MHz y 2.4 GHz. La frecuencia más baja de la capa física cubre tanto la banda europea de 868 MHz como la banda de 915 MHz, utilizada en países como Estados Unidos y Australia. La mayor frecuencia de la capa física se utiliza prácticamente en todo el mundo [43].

ZigBee ha aumentado sus usos en la robótica respecto a otras tecnologías como Bluetooth o Wi-Fi debido al bajo costo de sus transceptores. Las aplicaciones más destacadas de este protocolo para los robots son: la búsqueda

y rescate urbano [44]-[47], la extinción de incendios [48], la comunicación submarina [49], la comunicación de múltiples robots [50, 51], los robots que interactúan con múltiples plataformas de servicios [52], la seguridad del hogar [53], la rehabilitación robótica de miembros superiores [54] y la detección de fallas en robots industriales [55].

Comunicación Wi-Fi

La tecnología Wi-Fi fue definida en el estándar IEEE 802.11 [56], en donde se especifican los protocolos de control de acceso al medio y de capa física para implementar las redes de área local inalámbrica. La evolución de este estándar es constante y para la fecha alcanza más de 500 millones de redes instaladas. La Wi-Fi Alliance es quien se encarga de certificar los equipos desarrollados [57].

Wi-Fi es una de las tecnologías de mayor uso para el acceso a internet. Por lo tanto, es común su utilidad en la robótica, como por ejemplo en acciones como las siguientes: para determinar el posicionamiento de robots móviles [58]-[63], para crear mapas de entornos [64], para evitar los obstáculos [65], para establecer anomalías al interior de una tubería [66], para teleoperación de robots [67], para las operaciones de búsqueda y rescate [68] y para la limpieza del hogar [69].

Comunicaciones móviles

Las comunicaciones móviles celulares son uno de los principales catalizadores de la conectividad en la sociedad actual, dada su extensa cobertura. En la década de los noventa el principal servicio utilizado por los usuarios era la voz a través del empleo del estándar de segunda generación (2G) GSM (sistema global para las comunicaciones móviles). Posteriormente, a comienzos del siglo XXI se implantó el sistema de tercera generación (3G), sistema universal de telecomunicaciones móviles, cuyo objetivo fue aumentar la velocidad en la transmisión de datos. Luego, en la década del 2010 comenzó la transición a la técnica de cuarta generación (4G), conocida como LTE (*long term evolution*), con la que se logró aumentar las velocidades de transmisión [70]. Actualmente existe el estándar de quinta generación (5G), conocido

como “nueva radio”, que por su rendimiento permite ofrecer servicios para el IoT.

Diferentes aplicaciones de los robots han usado para la comunicación las plataformas de los estándares de las redes móviles creados hasta la fecha, entre las que se pueden resaltar las siguientes: los robots móviles con plataformas de comunicación GPRS (servicio general de paquetes vía radio) [71]-[73], de 3G [74, 75] y 5G [76]; los robots teleoperados con tecnologías de 2G [77], 4G [78] y 5G [79]; la búsqueda y rescate de seres humanos con tecnología de 3G [80, 81]; la detección de explosivos mediante el estándar 3G [82]; la comunicación de drones con tecnología de 4G [83]; la localización de robots con LTE [84]; la robótica industrial con 5G [85]; la telecografía con 4G [86]; y la telecirugía con 5G [87].

Comunicación con redes de área extensa de baja potencia

Estas redes poseen recursos limitados y se caracterizan principalmente por la batería de larga duración, la amplia cobertura, la alta capacidad y los bajos costos en los dispositivos y el despliegue. Por lo anterior, son una solución prometedora para aplicaciones de comunicación máquina a máquina y de IoT [88], como la que desarrolla la presente investigación.

Al ser una tecnología reciente, su uso en la robótica se encuentra en fase inicial; sin embargo, se evidencian algunas aplicaciones: en un robot de monitoreo de agua de río [89], en un robot de seguimiento [90] y para comunicar enjambres de robots [91]. Además, las redes de área extensa de baja potencia están mostrando un enorme potencial para la amplia gama de aplicaciones en automatización industrial y fabricación inteligente, de donde forma parte la robótica [88].

Por último, al hacer una comparación de costos en el despliegue de infraestructura de tecnologías inalámbricas para una misma cobertura con base en [92]-[94], se puede observar que con un par de módulos LPWAN de aproximadamente 14 USD se necesitarían: 50 puntos de acceso Wi-Fi, con un precio de 25 USD cada uno; 500 nodos Bluetooth, con un precio de 3 USD cada uno; y 50 nodos ZigBee, por un precio de 5 USD cada uno.

Capítulo 2

Robots modulares

Los robots modulares están compuestos por módulos que pueden reconfigurarse adaptando diferentes morfologías, estas requieren de una sincronización entre sus módulos para generar una locomoción y desplazarse en una superficie. Los movimientos para generar un desplazamiento del robot están directamente relacionados con la mecánica y la estructura del robot modular, como, por ejemplo, el diseño de los módulos, los grados de libertad (DOF) de los actuadores y la cantidad de módulos. El desarrollo de estos sistemas modulares ha permitido que se implementen diferentes tipos de robots, con distintas estructuras y diseños. En este capítulo se exponen sus características básicas [95, 96].

Descripción de los robots modulares

El robot modular acopla más de un robot y a través de dispositivos electrónicos puede tomar morfologías arbitrarias para adaptarse a un entorno respectivo, lo que le facilita su desplazamiento. Estos robots pueden configurarse manual o autónomamente, y estas reconfiguraciones dependen de ciertas características ligadas a la estructura de los módulos:

- *Las caras*: determinan el diseño de cada módulo y facilitan la unión y ensamblaje de los módulos entre sí. El diseño de estas es variable: pueden ser cuadradas, curvas o más complejas.
- *Escalabilidad*: la funcionalidad del robot modular depende de varios factores tales como la mecánica, el control y la comunicación entre el conjunto de elementos robóticos. El robot es escalable cuando, a pesar de tener

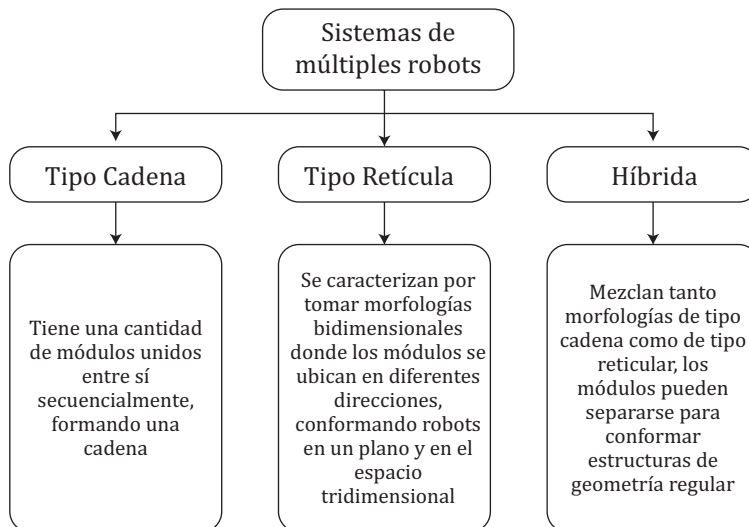
un gran número de módulos, consigue adaptarse a las complejas tareas, manteniendo la funcionalidad del sistema estable.

- *DOF*: establecen los ejes de rotación de cada módulo y determinan los tipos de movimientos que puede ejecutar el sistema. Estos movimientos pueden ser rotativos o lineales, dependen de los DOF del actuador y de la mecánica en general.
- *Comunicación*: los robots modulares se sirven de un protocolo de comunicación que permite enviar y recibir entre módulos información asociada a la posición de los actuadores, para generar una locomoción u otras variables proporcionadas por sensores que utiliza el robot.

Categorías de sistema de los robots modulares

Existen varias formas de clasificar los robots modulares; entre ellas están las nuevas adaptaciones y los desarrollos en los sistemas modulares tanto en su estructura como en su morfología [95]. Con base en la estructura, existen tres categorías fundamentales de robots modulares, como se muestra en la figura 1. El robot dispuesto para el desarrollo de este proyecto es de tipo cadena, cuyos módulos y cadenas cinemáticas se construyeron previamente.

Figura 1. Diagrama de clasificación de robots modulares según su estructura



Fuente: tomada de [97].

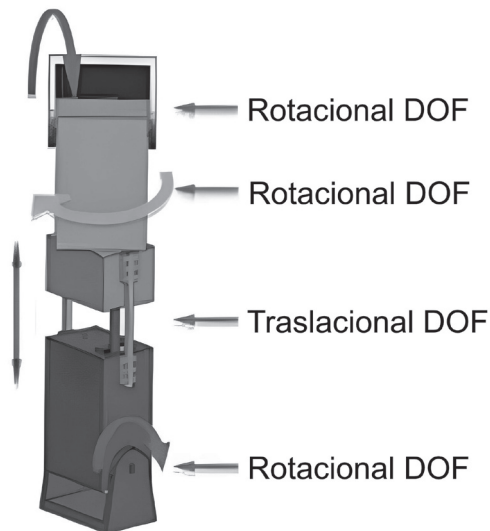
Estructura tipo cadena

Los robots modulares tipo cadena son un conjunto de módulos unidos secuencialmente que forman estructuras de gusano o serpiente. Tienen una estructura básica y sencilla que permite realizar una locomoción de forma sinusoidal o con movimientos a ritmos repetitivos.

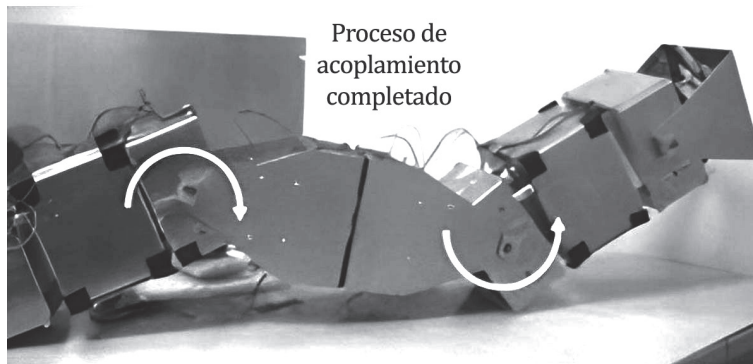
ModRED

ModRED presenta un diseño de sistema de robot modular homogéneo basado en cuatro DOF por módulo, donde tres son rotacionales y uno es prismático (figura 2). Esta técnica aumenta la versatilidad de reconfiguración entre módulos [98]. Cada módulo está conformado por tres cajas que albergan los actuadores y el sistema electrónico; en sus extremos se encuentran las placas de acoplamiento con un diseño plano y circular como el de un plato, que gira en sentido de las manecillas del reloj para asegurarse con la placa de otro módulo. Estas placas, a su vez, contienen ocho pines que ayudan en el acoplamiento e interconectan líneas de comunicación de datos y de alimentación.

Figura 2. Robot modular ModRED: configuración y estructura



Continúa en la siguiente página

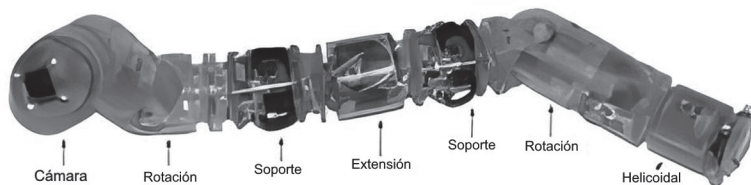


Fuente: tomada de [98].

Microtub

Es un robot modular tipo cadena o serpiente heterogénea. Esto quiere decir que cada uno de sus módulos tiene un diseño y función específica. Es multiconfigurable, característica que hace posible el intercambio de los módulos fácilmente, según la tarea requerida, sin necesidad de reprogramar el robot [96]. El diámetro de cada módulo es de solo 27 mm y el espesor de algunas piezas es inferior a 1 mm; por ello, su tamaño lo clasifica como un micro-robot. Microtub, como se presenta en la figura 3, posee una cámara que le facilita al robot identificar fugas o roturas dentro de una tubería, su módulo de rotación cuenta con dos DOF rotacionales. Los dos módulos de rotación permiten que el robot se pueda desplazar, realizando movimientos ondulares tal como lo haría una serpiente; los módulos de soporte y los módulos de extensión ayudan al robot a girar a la derecha o izquierda; por último, el módulo helicoidal tiene un DOF rotacional capaz de impulsar otros módulos.

Figura 3. Robot modular Microtub y sus partes



Fuente: tomada de [96].

Estructura tipo retícula

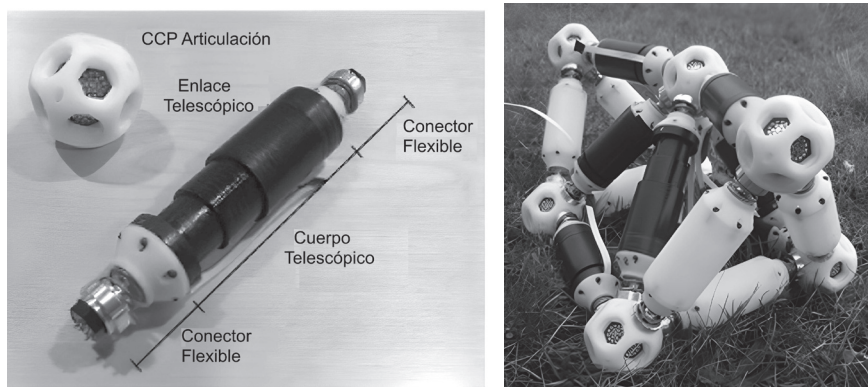
Son robots capaces de formar morfologías bidimensionales o tridimensionales al ser reconfigurados. La intención de estas estructuras es que se puedan autoconfigurar, generando, de esta forma, una cierta autonomía adaptativa.

Odín

Es un robot modular heterogéneo, reconfigurable manualmente y que posee una jerarquización de módulos que dotan al robot de alguna propiedad. Los de nivel más bajo se asocian a la estructura del robot, a módulos con sensores y a módulos encargados de la alimentación del sistema [99]. Los módulos de jerarquía superior facilitan la generación de movimiento y dan una estabilidad a la estructura.

La estructura del robot Odín, como se aprecia en la figura 4(b), se deforma a causa de fuerzas externas, permitiendo así una morfología adaptativa con una reconfiguración. Esta deformación la permiten los actuadores telescópicos, como el presentado en la figura 4(a), que ayudan a variar la distancia de las articulaciones.

Figura 4. Robot Odín: módulos (a) y estructura (b)



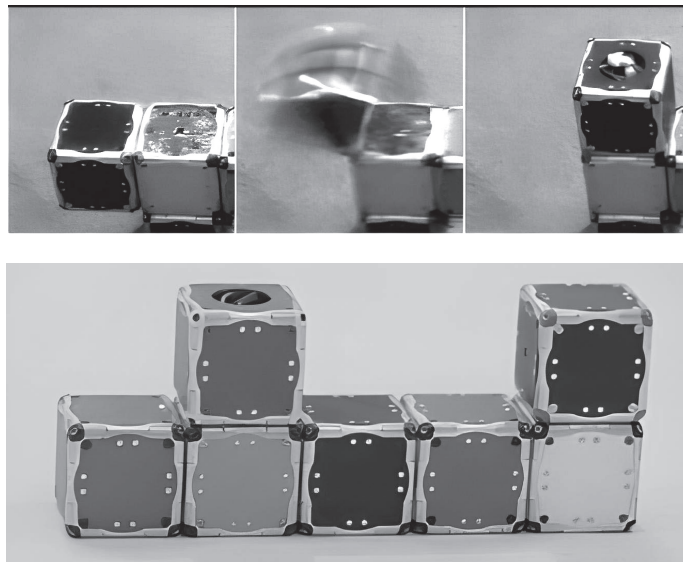
Fuente: tomada de [99].

M-Blocks

Es un robot modular autoensamblable conformado por cubos de 5 cm que están unidos magnéticamente entre sí. Los módulos logran estos movimientos transfiriendo rápidamente la energía al cuerpo del robot por medio de

un momento angular, generando así un movimiento pivotante que consigue que los módulos cambien su posición y se unan a otros, formando estructuras aleatorias. Un volante ubicado en el interior de cada módulo se usa para almacenar la energía o momento angular, el volante emplea un mecanismo para desacelerar y generar el momento inercial que permite que el módulo rompa sus enlaces magnéticos y realice un giro sobre sí mismo, como se observa en la figura 5 [100].

Figura 5. Movimiento pivotante del robot modular M-Blocks: pivote de los módulos (a) y estructura (b)



Fuente: tomada de [100].

Estructura híbrida

Los robots de categoría híbrida combinan configuraciones morfológicas de tipo cadena y retícula, formando estructuras irregulares y adaptables a un entorno. Su tecnología suele facilitar la autoconfiguración y admitir distribuciones con mayores DOF.

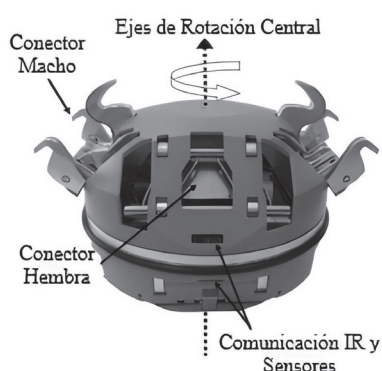
ATRON

ATRON es un sistema modular autoconfigurable donde cada módulo tiene un DOF rotacional. El actuador es capaz de rotar 360° , cada módulo mide

11 cm de diámetro y tiene la capacidad de conectarse a ocho módulos vecinos, cuatro en cada hemisferio [101]. Las uniones entre módulos se dan debido a los conectores tipo macho en forma de tres ganchos que se fijan a los conectores tipo hembra.

El sistema es capaz de conectar y desconectar mecánicamente sus módulos y conformar morfologías arbitrarias, como las de un carro, como se observa en la figura 6, y rodar como lo haría este. Otra de sus morfologías es la de tipo serpiente. ATRON usa un protocolo de comunicación infrarroja (IR) para transportar información asociada a la posición y rotación de cada actuador e información de proximidad proporcionada por los sensores IR.

Figura 6. Partes de un módulo y morfología tipo carro del robot modular ATRON: módulo (a) y morfología (b)

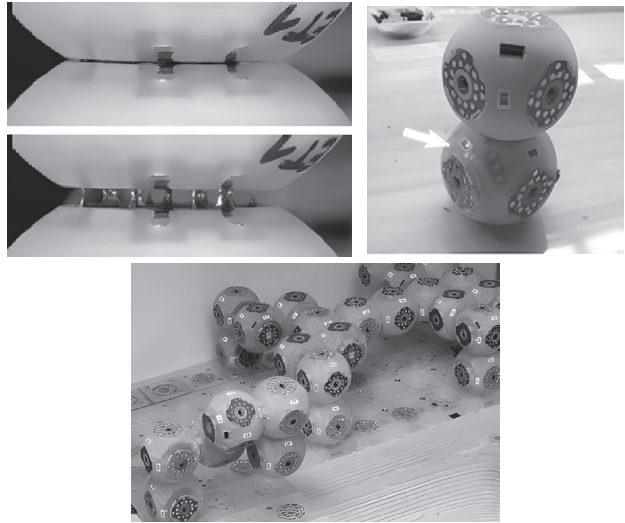


Fuente: tomada de [101].

SolderCubes

Es un robot autoconfigurable de un DOF por cada módulo. Cada uno de estos tiene 5.5 cm de alto y ancho, y tiene la posibilidad de conectarse con seis módulos vecinos, uno por cada cara de su forma cúbica. Cada módulo SolderCube [102] expone seis conectores autosoldables en su superficie que están constituidos por una tarjeta de circuito impreso de dos caras, cuya cara exterior contiene una almohadilla de aleación de bajo punto de fusión, que, al ser calentada por unas resistencias a 62 °C, permite que las almohadillas de ambos módulos se suelden y se conecten mecánica y eléctricamente, como se observa en la figura 7.

Figura 7. Autosoldado: morfologías del robot SolderCubes



Fuente: tomada de [102].

Otros ejemplos de robots modulares se muestran en la tabla 1. Estos se organizan de acuerdo con el año y se indica en qué tipo de estructura se clasifica cada uno.

Tabla 1. Otros ejemplos de robots modulares

Robot	Clasificación	Año	Cita
Polypod	Cadena	1998	[103]
TeleCubes	Retícula	1998	[104]
MoleCube	Retícula	1998	[105]
CONRO	Cadena	1998	[106]
SuperBot	Híbridos	2005	[107]
YaMoR	Cadena	2005	[108]
Miche	Retícula	2006	[109]
HexaMob	Híbridos	2017	[110]

Fuente: elaboración propia.

Control de movimiento en robots modulares

Existen diferentes técnicas que facilitan a los robots modulares realizar movimientos periódicos para lograr su desplazamiento. Entre estas técnicas se destacan la cinemática inversa, la tabla de control de movimientos y los generadores sinusoidales. Estas estrategias han sido implementadas en simuladores para ser estudiadas y analizadas, y luego ejecutadas por el robot modular; sin embargo, una limitación de los entornos virtuales es que la autonomía del robot se reduce, ya que el simulador no estima todas las variables o interacciones físicas que puede tener un robot con su entorno. Esto se debe a que los entornos virtuales son fijos y los reales están sujetos a cambios como, por ejemplo, el cambio de lugar de los obstáculos, la temperatura y la humedad del ambiente, o el estado del *hardware* del robot.

Los generadores centrales de patrones (CPG) [111] son una estrategia que ha solucionado parcialmente las limitaciones de autonomía del robot. Un CPG emula movimientos rítmicos de animales, generando secuencias de locomoción adaptativas a espacios invariantes. A continuación se describen el concepto de *locomoción* y las características básicas de las técnicas de control de movimiento en un robot. Así mismo, se presentan algunos modos de navegación.

Locomoción en robots

La locomoción es una propiedad que por medio de movimientos repetitivos hace que el sistema genere un desplazamiento de un punto a otro. En robótica la locomoción cumple un papel importante, ya que con ella el robot podrá cumplir las misiones de navegación recibidas.

Diferentes tipos de mecanismos permiten una locomoción, ya sea por medio de patas (robots bípedos, cuadrúpedos, hexápodos, entre otros) o por ruedas que pueden producir una locomoción diferencial, síncrona, Ackerman, entre otras. La de tipo diferencial está constituida por dos ruedas fijas, es un sistema sencillo pero que carece de estabilidad, por lo cual se incorpora una tercera rueda orientable no controlada. El diseño de tipo síncrono, por lo general, tiene tres ruedas, que son tanto de dirección como motrices; este sistema es complejo y difícil de implementar. Por último, la

locomoción Ackerman tiene el mismo mecanismo que el de un carro de cuatro ruedas [112]. Estos mecanismos tienen ciertas desventajas a la hora de implementarse sobre superficies irregulares o terrenos desconocidos, ya que su estructura y diseño no faculta para un desplazamiento estable y son susceptibles a volcarse y presentar daños. La necesidad de crear dispositivos robóticos adaptables a un espacio llevó a los investigadores a diseñar estructuras flexibles que proporcionaran versatilidad y estabilidad. Por tanto, se desarrolló una variedad de sistemas que cumplieran con estas propiedades, entre estos, robots ápodos (sin patas) y robots modulares, que cuentan con la capacidad de desplazarse por superficies rugosas e irregulares. La locomoción que deben generar estos sistemas depende directamente de su estructura física y mecánica, en muchos casos para estos es bioinspirada y simula movimientos que los seres vivos suelen realizar. En el caso de los robots ápodos y los robots modulares ápodos, se implementan los movimientos sinusoidales, giratorio, de escalar y en espiral [113].

Los robots modulares pueden reconfigurar sus módulos con base en un terreno específico y, así, formar diferentes estructuras de geometría regular e irregular; sin embargo, la locomoción se torna compleja y debe poseer un sistema de control específico que brinde una buena funcionalidad. Se describen aquí los tipos de sistemas de control y las técnicas de control de movimiento para robots modulares.

Sistemas de control

Son un conjunto de elementos para influir en el comportamiento y funcionamiento de un sistema [113]. Para robots modulares el sistema de control se encarga de administrar y procesar tareas particulares asociadas a la locomoción y a los procesos lógicos.

Control centralizado

En este sistema de control todas las tareas son procesadas por un único controlador, que usualmente puede estar en uno de los módulos del robot o en un dispositivo externo. Una de las ventajas del control centralizado es que se puede reprogramar fácilmente; no obstante, si el controlador presenta alguna falla, afectará a todo el sistema.

Control descentralizado

En los controladores de este tipo se establece más de un dispositivo capaz de ordenar y comunicar a todos los módulos la realización de las funciones correspondientes [114]. Los controladores pueden estar en dos o más módulos, lo que conlleva tener una redundancia en el caso de que uno de los controladores falle, para garantizar el funcionamiento del robot modular, brindando estabilidad y funcionalidad al sistema.

Técnicas de control de movimiento en robots modulares

Las técnicas de control de movimiento permiten al sistema de control generar una respuesta en cuanto a la locomoción y la ejecución de tareas. Estas técnicas han ayudado a la comunidad que investiga estos robots a realizar los patrones de movimiento. A continuación se describen algunas de las técnicas aplicadas a la robótica modular.

Cinemática inversa

Esta técnica ayuda a establecer un movimiento para que una articulación final vaya a una posición deseada mediante el uso de ecuaciones cinemáticas [97]; así, en la robótica modular se ha aprovechado la cinemática inversa para determinar las posiciones deseadas en las articulaciones de los módulos. Por ejemplo, en [113] se utilizó la cinemática inversa en un robot serpiente cuyas coordenadas son calculadas por un computador y enviadas a los módulos del robot. Una limitación de esta técnica es que, entre mayor número de DOF presente el robot, más difícil será establecer un modelo matemático para el movimiento de dicho robot.

Tablas de control de movimiento

Estas tablas contienen todos los parámetros de movimiento que se ejecutarán en las articulaciones, ya que cada parámetro está formado por un vector que contiene las posiciones de las articulaciones en los módulos. El sistema de control inicia con el primer parámetro en la tabla de control de movimiento, luego procesa dicha información y envía la posición de la articulación al módulo correspondiente. Una vez la articulación ha terminado de posicionarse, el sistema de control seguirá con el siguiente parámetro, y así hasta procesar

el último parámetro de la tabla. Finalizado esto, se repetirá todo el proceso para generar un movimiento sucesivo y repetitivo.

El robot Polypod, uno de los primeros robots modulares, construido por Mark Yim [115], empleó tablas de control de movimiento para generar movimientos de tipo circular, en forma de 'S' y araña. Aquí se demostró que el uso de tablas de movimiento funciona de manera independiente de la cantidad de módulos.

Generadores sinusoidales

Es un controlador que estima la posición de todas las articulaciones presentes en el robot a partir del cálculo de una ecuación sinusoidal. Este cálculo arroja el valor de la posición de un módulo y estima la posición de los siguientes, de forma que se produce un movimiento sinusoidal.

En robótica modular el uso de esta técnica puede tornarse complejo a la hora de implementarse en robots con morfologías arbitrarias [113]; sin embargo, en robots modulares tipo cadena en una dimensión, presenta pocos cálculos para el controlador y genera una locomoción simple y adaptable a superficies no uniformes, que hace empleo de la ecuación (1).

$$x=A \times \text{sen} (\theta \times t+\varphi) \quad (1)$$

La ecuación (1) representa el cálculo hecho por el controlador, donde x es la posición de cada módulo, A es la amplitud o apertura máxima de cada módulo, θ es un ángulo que varía con respecto al tiempo y φ es la diferencia de fase que se asigna a cada módulo.

Generadores centrales de patrones y aplicaciones en robótica

Los CPG son redes neuronales presentes en la biología de los animales, realizan patrones rítmicos para llevar a cabo actividades motoras repetitivas [111]. Estas actividades están asociadas a la locomoción y respiración; por ello, se ha estudiado su comportamiento para originar movimientos bioinspirados en sistemas robóticos.

En robótica los CPG suponen una técnica de control de movimiento que presenta robustez, versatilidad y autonomía, debido a que, al cambiar los parámetros de un CPG, se puede obtener una gran cantidad de movimientos.

El CPG está construido por un conjunto de redes que se interconectan entre sí y que generan una secuencia de tareas repetitiva. Los siguientes son algunos tipos de robots que se sirven de esta técnica:

- *Bípedos*: son robots que usan dos extremidades para desplazarse, los CPG imitan una locomoción bioinspirada en animales de dos piernas. Para realizar un movimiento, el robot estima la posición de las articulaciones con base en su posición inicial. Su movimiento está ligado a las articulaciones, una a nivel de la cadera y la otra a nivel de las rodillas [113].
- *Cuadrípedos*: son robots que utilizan cuatro patas, que deben coordinarse entre sí haciendo uso de osciladores acoplados para cada una de las extremidades. El CPG sincroniza movimientos e imita la locomoción de un animal de cuatro patas, apoyando siempre tres patas en el suelo para caminar. Este tipo de estructuras proporciona estabilidad y omnidireccionalidad [116].
- *Hexápodos*: son robots que usan seis patas para brindar estabilidad y adaptarse a superficies no uniformes. Los movimientos que pueden hacer son avanzar, retroceder o girar.
- *Serpientes*: son robots que generan movimientos repetitivos en forma de ‘S’, ya sea usando CPG o generadores sinusoidales. En el caso de robots modulares tipo serpiente que hacen uso de CPG, estiman el movimiento de un módulo a partir de la posición de sus módulos vecinos.
- *Modulares con morfologías arbitrarias*: en este tipo de robots, el CPG [117] usa osciladores acoplados para generar movimientos rítmicos en sus extremidades; con todo, debido a la compleja locomoción de estos sistemas, se utiliza un computador para determinar las posiciones de los módulos.

Modos de navegación

La navegación corresponde a un conjunto de métodos implementados para crear una ruta de desplazamiento con base en un espacio. Su objetivo principal es ir de un punto *A* a un punto *B* [113]. En robótica se ejecuta una serie de tareas para realizar la navegación, entre estas se encuentran la percepción, la planificación y la operación. La percepción es la forma en que el robot percibe la realidad o el entorno que lo rodea, generalmente mediante

sensores. La planificación es una estrategia que permite trazar una trayectoria según la información recibida por los sensores. Por último, la operación es el conjunto de movimientos o acciones para efectuar una ruta estimada. Algunos de los modos de navegación que se implementan en la robótica son:

- *Control reactivo*: este tipo de estrategia brinda una respuesta de acción según la información de percepción sensorial. Las respuestas se modifican continuamente de acuerdo con el cambio de estado en los sensores. Este es un sistema simple que favorece la ejecución de tareas de forma rápida [118].
- *Grafos de visibilidad*: aquí el espacio es representado como un plano bi-dimensional, con obstáculos simbolizados por polígonos. El objetivo de esta estrategia de navegación es ir de un punto a otro, trazando la ruta más corta y evitando que esta se intercepte con los obstáculos [118].
- *Diagramas de Voronoi*: estos son similares a la estrategia de grafos de visibilidad, aunque aquí se busca la ruta más larga entre dos puntos, representando la trayectoria mediante líneas rectas o curvas [118].
- *Descomposición de celdas*: esta estrategia divide el espacio en celdas, donde se desarrolla un algoritmo capaz de encontrar y determinar las celdas que unen dos puntos sin que estas sean obstruidas por un obstáculo [118].
- *Campos potenciales*: trazan una trayectoria en la que el robot es considerado una partícula atraída por el punto final y repelida por los obstáculos [118].

Las operaciones de navegación de los robots se pueden ejecutar de forma global o local: la global se da cuando un dispositivo externo traza la ruta que debe seguir el robot, y la local se refiere a cuando el sistema de control del robot construye una trayectoria de navegación.

En robótica modular se suelen usar simuladores que recrean un espacio de trabajo en el que un robot virtual (con las mismas características físicas y mecánicas del robot modular real) se puede desplazar e interactúa con el entorno. Algunas de las ventajas de estos simuladores son la predicción e ilustración de los movimientos del robot para construir morfologías que el robot real pueda ejecutar. Esto permite realizar pruebas de locomoción y dar una idea de las posibilidades que tiene el robot para generar diferentes movimientos, dependiendo de su configuración estructural. También se pueden enviar los

movimientos realizados por el robot virtual al robot real para su ejecución. Una desventaja de estos simuladores es que no permiten al robot adaptarse a entornos cambiantes. Estos simuladores ayudan a trazar una trayectoria de navegación con base en las condiciones físicas de dicho entorno virtual.

Capítulo 3

Internet de las cosas (IoT)

Las aplicaciones basadas en lo que hoy se conoce como IoT abarcan profundamente toda clase de necesidades en casi cualquier ámbito (salud, domótica, industria, entre muchas otras). Es común hablar hoy día de aplicaciones basadas en *smarththings*, que no son más que la instauración del IoT en el entorno tecnológico actual. IoT está continuamente en cambio y en la actualidad no existe una estandarización que regule o especifique de forma estricta un modelo único para su aplicación [119]; es de esperar, entonces, que día a día se produzcan distintas tecnologías, protocolos y soluciones diversas para cada tipo de problemas o situaciones concretas, donde se requiera una solución basada en las grandes posibilidades que ofrece el IoT, esto es, la ejecución de diversos protocolos y tecnologías relacionados con el propósito de la aplicación de un modelo IoT o la selección de un estándar. Esto es un verdadero reto y una tarea difícil para cualquier organización.

Conceptualmente, el modelo de IoT no fue creado como una red para objetos físicos. En realidad, el IoT es una red de dispositivos que interactúa con objetos físicos [120]. Teniendo claro este concepto, en el panorama actual las tecnologías más adecuadas para trabajar sobre IoT son [121]: la comunicación de campo cercano (NFC); la identificación por radiofrecuencia; los sistemas basados en IEEE 802.15.4 (ZigBee, 6LoWPAN, entre otros); los dispositivos Bluetooth y Bluetooth de baja energía; los sistemas propietarios como Z-Wave, CSRMesh y EnOcean; y los principales sistemas basados en IEEE 802.11/Wi-Fi. Si lo que se requiere son aplicaciones con características de largo alcance y bajo consumo de potencia, estas tecnologías no son relevantes.

Otras perspectivas estudian el uso de redes licenciadas de telefonía celular para IoT, como lo son 2G/GSM/GPRS; sin embargo, además del costo de utilizar las bandas licenciadas, este tipo de tecnología está enfocada en la transmisión de largo alcance, pero con un alto consumo de energía [121]. Lo mismo sucede con las tecnologías 3G/4G a una mayor velocidad de transmisión [122].

Pese a que la gran mayoría de problemas resueltos con IoT están basados en aplicaciones de corto alcance aprovechando tecnologías como ZigBee/Wi-Fi, ciertas aplicaciones no pueden ser suplidas por tecnologías basadas en corto alcance o alto consumo de recursos.

Las tecnologías emergentes que se apoyan en las redes de área extensa de baja potencia ofrecen una alternativa adecuada cuando se requiere envío de datos a un largo alcance, con un consumo energético y una velocidad limitados [123].

A continuación, se revisan las tecnologías dispuestas para implementar IoT sobre LPWAN. Dado su reciente auge, las LPWAN se presentan como una solución prometedora para las comunicaciones, debido a su bajo costo y consumo reducido de energía, ya que están diseñadas para ofrecer conectividad a escala masiva y en zonas de amplia cobertura, donde se demanda baja transmisión de datos con dispositivos inalámbricos asequibles [124].

Redes de área extensa de baja potencia

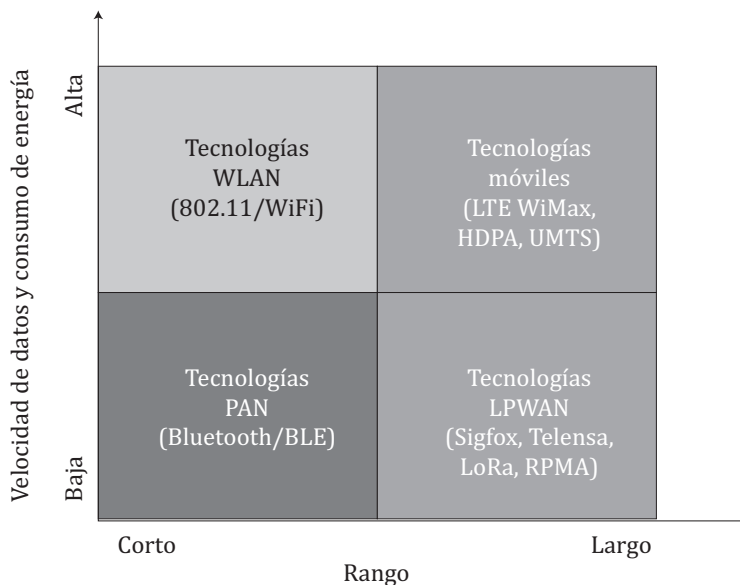
LPWAN es un término genérico que se da a un grupo de tecnologías que habilita las comunicaciones en redes de área extensa a un bajo costo y con un racional consumo de energía [123].

La gran mayoría de tecnologías de LPWAN opera en el rango de frecuencias de subgigahertz (Sub-GHz); esto significa que se tiene mejor calidad de señal y menor costo en áreas de amplio rango en comparación con las bandas de 2.4 GHz y 5 GHz, que son empleadas en protocolos para tecnologías Wi-Fi, Bluetooth, telefonía celular, entre otras [122].

Como se observa en la figura 8, el uso de las tecnologías de LPWAN frente a otras tecnologías inalámbricas dependerá de la aplicación específica. Las LPWAN se utilizan cuando se requieren amplios rangos de alcance con

un consumo de energía bajo. La figura 8 es una visión general de la clasificación de las diferentes tecnologías inalámbricas; no obstante, dependiendo de la aplicación, se deben evaluar otros aspectos.

Figura 8. Comparación de los rangos de alcance, consumo de energía y velocidad de transmisión de datos en distintas tecnologías inalámbricas



Fuente: tomada de [125].

Adicionalmente, las LPWAN se implementan sobre bandas licenciadas y no licenciadas, por lo que no están sujetas a un único modelo o prototipo, ya que cada una ofrece distintas posibilidades, así como ventajas y desventajas. A causa de esto, se analizan las principales tecnologías que en la actualidad se desarrollan en este extenso campo de comunicaciones, además de sus principales características. Ellas son: NB-IoT (IoT de banda angosta), LoRa (largo alcance), Telensa y Sigfox. Este sector presenta un cambio continuo, por lo tanto, en el corto plazo es probable que incursionen nuevas tecnologías en el mercado [122, 124].

NarrowBand-IoT

Este tipo de tecnología enfocada en aplicaciones para IoT sobre la banda licenciada promete una vida útil de más de diez años en sus dispositivos, además de un amplio alcance. NarrowBand-IoT (NB-IoT) define un nuevo

tipo de tecnología de radio que puede integrarse en el estándar LTE, es decir, trabaja sobre las mismas bandas licenciadas y se fundamenta en LTE; no obstante, varias características se han modificado con el fin de minimizar el consumo de batería, reducir el costo y, por supuesto, darle mayor alcance de cobertura [124].

Los costos iniciales de implementar NB-IoT son equiparables a los de la tecnología GSM/GPRS; sin embargo, debido al aumento de la demanda de estos dispositivos de uso exterior e interior y a su sencillez, los costos cada vez son más pequeños y, por ende, los dispositivos más asequibles. Esto convierte a NB-IoT en una tecnología altamente flexible que puede subsistir en armonía con otras tecnologías de redes móviles como las de 2G, 3G y 4G [126]. Se describen ahora sus principales características [127].

Características físicas

Las siguientes son las características físicas de NB-IoT:

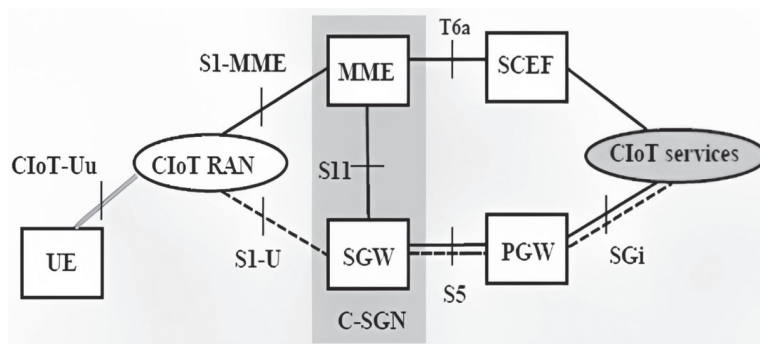
- Se sirve de las mismas frecuencias asignadas para LTE [128].
- Emplea modulación por desplazamiento de fase por cuadratura (QPSK) [129].
- Usa tres formas de operación en las bandas de frecuencia: modo *stand-alone*, modo *in-guard band* y modo *in-band* [130].
- Hay doce subportadoras de 15 kHz en el enlace descendente, usando acceso múltiple por división de frecuencias ortogonales y portadoras de 3.75/15 kHz en el enlace ascendente, usando acceso múltiple por división de frecuencia de portadora única.

Arquitectura de red

El núcleo de la red de NB-IoT está basado en el sistema de paquetes evolucionado (EPS) y dos optimizaciones para el internet de las cosas celular (CIoT), que corresponden a la optimización CIoT EPS del plano de usuario y a la optimización CIoT EPS del plano de control, como se presenta en la figura 9. Ambos planos eligen la mejor ruta para los paquetes de datos de control y de usuario en los enlaces ascendente y descendente. La ruta de optimización para el plano seleccionado es flexible para el paquete de datos generado por

el conjunto de móviles. El procedimiento de acceso a la celda de un usuario de NB-IoT es similar al de LTE. En el plano de control de optimización CIoT EPS, la red de acceso de radio terrestre universal evolucionada maneja las comunicaciones de radio entre el equipo de usuario (UE) y la entidad de gestión de movilidad (MME), y consta de las estaciones base evolucionadas, denominadas eNodeB. Luego, los datos se transmiten a la puerta de enlace de red de paquetes de datos (PGW) a través de la puerta de enlace de servicio (SGW). Para los datos que no son IP, se transfieren al nodo recién definido, función de exposición de capacidad de servicio, que puede entregar datos tipo máquina sobre el plano de control y proporcionar una interfaz abstracta para los servicios. Con la optimización CIoT EPS del plano de usuario, tanto los datos de protocolo de internet (IP) como los que no son IP se pueden transmitir por medio de túneles de radio por la SGW y la PGW al servidor de aplicaciones [127].

Figura 9. Arquitectura de red de NB-IoT



Fuente: tomada de [127].

Adicionalmente, en la figura 9 aparecen los siguientes componentes:

- CIoT RAN: red de acceso a radio del CIoT.
- CIoT-Uu, S1-MME, T6a, S1-U, S5 y SGi: interfaces para la comunicación entre los bloques.

El objetivo es que el UE se conecte a las tecnologías que soportan los servicios del CIoT, haciendo uso de esta arquitectura.

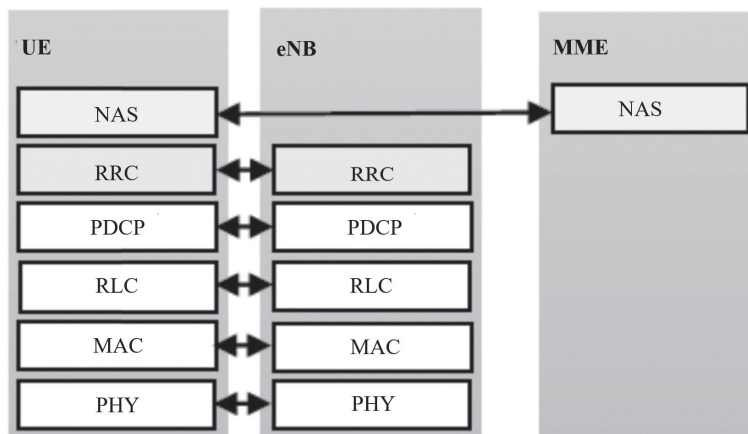
Protocolo de control de acceso al medio

La pila de protocolos para NB-IoT es la pila de protocolos fundamental de LTE, que se reduce al mínimo y se mejora para reutilizar y evitar que NB-IoT sobrecargue el LTE no empleado. La pila de protocolos NB-IoT se considera una nueva interfaz aérea para LTE. Como se muestra en la figura 10, la estructura del protocolo NB-IoT se ha dividido en el plano de control y en el plano de usuario. El protocolo de convergencia de paquetes de datos es de la capa 2 con un tamaño de 1600 bytes. El estrato sin acceso (NAS) de la pila de protocolos transmite señales que no son de radio entre el UE y la red central. El NAS realiza acciones de autenticación, control de seguridad, gestión de movilidad y gestión de transporte. El estrato de acceso es una capa funcional debajo del NAS entre el UE y la red de radio. Se aprovecha para administrar recursos de radio en NB-IoT. La capa de control de recursos de radio minimiza la señalización por suspensión/reanudación de la operación del plano de usuario. La seguridad de la capa 2 provee cifrado de la autenticación y la señalización NAS entre el UE y la red central. La gestión de movilidad del usuario en modo de conectividad se ajusta en este protocolo.

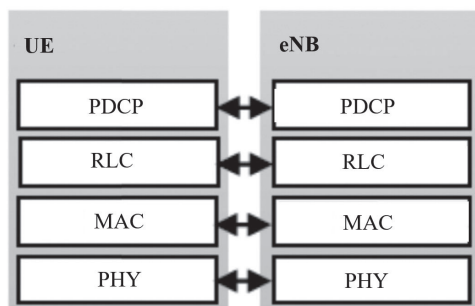
Para NB-IoT, el procedimiento de canal de acceso aleatorio (RACH) siempre se basa en la contención y comienza con la transmisión de un preámbulo. Si la transmisión del preámbulo falla, el UE retransmitirá hasta que se alcance el número máximo de retransmisiones, que depende del nivel de mejora de cobertura (CE). Luego, el UE pasará al siguiente nivel de CE. Si la eNodeB recibe con éxito el preámbulo, enviará la respuesta de acceso aleatorio asociada al UE. Después de eso, se transmite un mensaje programado para iniciar el proceso de resolución de contenciones. El procedimiento de RACH se completa cuando el mensaje de resolución de contenciones se transmite al UE [127].

Figura 10. Protocolos de NB-IoT para el control de acceso al medio (plano de control y plano de usuario)

Plano de Control



Plano de Usuario



Fuente: tomada de [127].

Vida útil de la batería y latencia

Debido a la poca sincronización, pero regular, el dispositivo NB-IoT consume energía adicional; además, la multiplexación por división de frecuencia ortogonal y el acceso múltiple por división de frecuencia requieren más corriente pico para el transmisor lineal, siendo de hasta 130 mA con una latencia menor a 10 s [9]. Este consumo extra de energía comporta una mayor tasa de transmisión de datos y menor latencia, en comparación con tecnologías como LoRa, lo que permite la mejora de estos parámetros de QoS [131].

Cobertura y rango de red

La cobertura de dispositivos con tecnología NB-IoT se mejora hasta en 23 dB para entornos interiores con respecto a otras tecnologías LPWAN; sin embargo, el despliegue de esta tecnología (en estructura física) se circunscribe a la existencia de estaciones base 4G/LTE, lo que no hace adecuado a NB-IoT para aplicaciones rurales o en donde no haya una infraestructura de cobertura 4G instalada. A pesar de esto, en condiciones óptimas, el rango de la señal puede alcanzar distancias de hasta 25 km [9]. No obstante, NB-IoT puede experimentar una interferencia significativa debido al uso de las mismas frecuencias que LTE [131].

LoRa

LoRa alude a la capa física o a la modulación inalámbrica que se utiliza para crear enlaces de comunicación de largo alcance con baja potencia. Aunque en muchas fuentes bibliográficas se hace referencia de forma indistinta a LoRa y LoRaWAN (red de área amplia de largo alcance), existen ciertas diferencias entre ellas.

LoRaWAN define el protocolo de comunicación y el sistema de arquitectura de red, mientras que LoRa, como ya se indicó, establece la capa de red física que posibilita la comunicación en largo alcance [132].

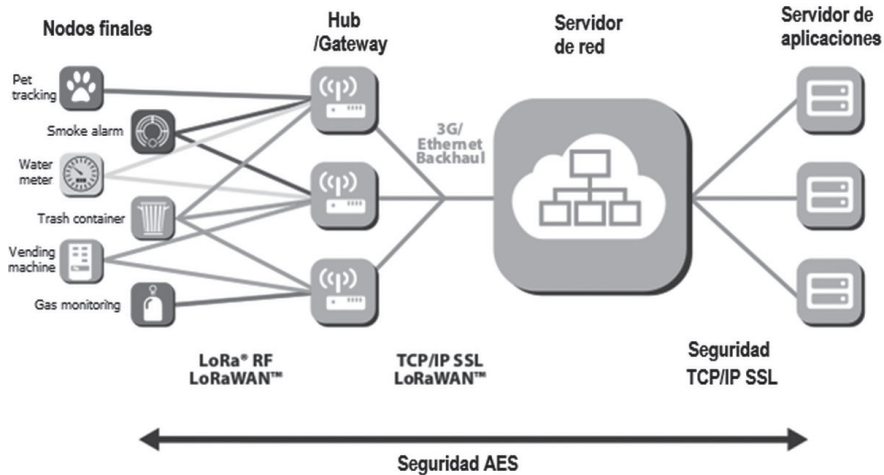
Una forma adecuada de evaluar las ventajas que tiene LoRaWAN frente a otras tecnologías emergentes de baja potencia es analizar sus parámetros de desempeño. LoRaWAN demuestra un buen rendimiento en situaciones que impliquen transmisión de baja cantidad de paquetes de datos a grandes distancias. Esto se debe a que usa modulación *chirp spread spectrum* (CSS), que es robusta frente a la interferencia y el ruido presentes en los sistemas de comunicaciones, llegando a tener una cobertura de hasta 21 km y una transmisión de 11 paquetes/min. También evidencia un alto desempeño cuando hay una gran cantidad de dispositivos conectados simultáneamente (hasta 500 paquetes/min con 800 dispositivos) [133].

A continuación se analizan las características del protocolo a partir de lo indicado por la LoRa-Alliance.

Arquitectura de red

Su implementación se basa en una estructura de red de largo alcance tipo estrella como la que se expone en la figura 11.

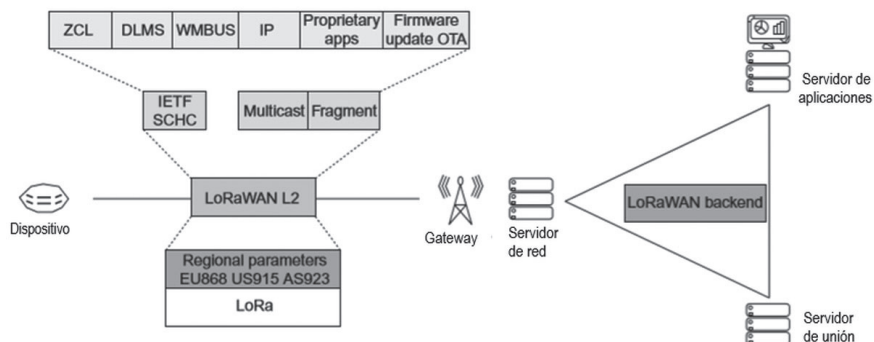
Figura 11. Estructura de red LoRaWAN



Fuente: tomada de [132].

En cuanto a la arquitectura de LoRa, en la figura 12 se muestra un esquema de la misma junto con los protocolos usados. Esta arquitectura se inspira en futuras mejoras, lo que quiere decir que esta tecnología está en constante cambio y su construcción se orienta por la tecnología desarrollada por LoRa Alliance [134].

Figura 12. Arquitectura y protocolos usados en LoRaWAN



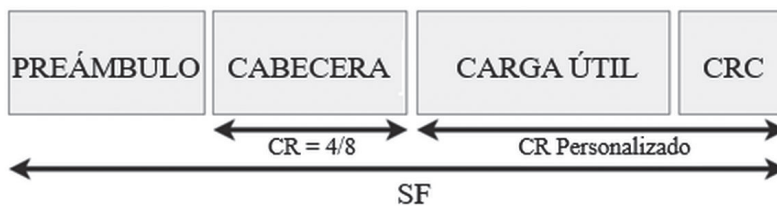
Fuente: tomada de [134].

Capa física

En función del modelo de interconexión de sistemas abierto, la solución brindada por LoRaWAN es la más popular en cuanto al protocolo de control de acceso al medio de las múltiples alternativas diseñadas para las redes de área extensa. Los paquetes transmitidos por medio de LoRa pueden tener diferentes parámetros en la modulación, que son: ancho de banda (BW), rata de código (CR) y factor de ensanchamiento (SF).

En general, un paquete de LoRa contiene un preámbulo, una cabecera, la carga útil y los bits de código de redundancia cíclica [135]. Su estructura se muestra en la figura 13.

Figura 13. Estructura de un paquete LoRa y los factores de la modulación en el paquete



Fuente: tomada de [135].

Vida útil de la batería

En LoRaWAN los nodos de la red son asíncronos y se comunican cuando tienen datos para enviar o en algún evento específico o previamente programado. Este tipo de protocolo es conocido como protocolo ALOHA, lo que reduce considerablemente el consumo de energía respecto a tecnologías inalámbricas convencionales.

Capacidad de la red

Dada la alta escalabilidad de LoRaWAN, el despliegue de esta tecnología es de bajo costo en infraestructura y depende directamente de la cantidad de puertas de enlace (*gateways*) necesarias para soportar el número de usuarios; es decir, si se requiere más capacidad en la red, se agregan más *gateways* a la estructura. Esta escalabilidad que presenta LoRaWAN no es habitual en otras tecnologías LPWAN.

Seguridad

LoRaWAN utiliza dos capas dedicadas a la seguridad: una de red y otra de aplicación. En la capa de red se verifica la autenticidad del nodo en la red y la capa de aplicación certifica que el operador de la red no tiene acceso a los datos de la aplicación del usuario final.

Se emplea encriptación tipo estándar de cifrado avanzado (AES) con un intercambio de claves que utiliza un identificador IEEE EUI-64.

Especificaciones técnicas

De manera más formal, las características de esta tecnología se presentan en la tabla 2 y se comparan con las demás en la tabla 3 (en el siguiente capítulo).

Tabla 2. Especificaciones técnicas de LoRaWAN para diferentes regiones

Especificación técnica	Europa	Norte-américa	China	Corea	Japón	India
Banda de frecuencia	867-869 MHz	902-928 MHz	470-510 MHz	920-925 MHz	920-925 MHz	865-867 MHz
Canales	10	64 + 8 + 8	En definición por el comité técnico			
Ancho de banda de canal de subida	125/250 kHz	125/500 kHz				
Ancho de banda de canal de bajada	125 kHz	500 kHz				
Potencia de transmisión de subida	+ 14 dBm	+ 20 dBm típico				
Potencia de transmisión de bajada	+ 14 dBm	+ 27 dBm				
SF subida	7-12	7-10				
Rata de datos	250 bps-50 kbps	980 bps-21.9 kbps				
Presupuesto del enlace de subida	155 dB	154 dB				
Presupuesto del enlace de bajada	155 dB	157 dB				

Fuente: elaboración propia con base en [127].

Telensa

Telensa es una tecnología de ultrabanda angosta (UNB) basada en LPWAN y desarrollada por Telensa Inc. Su principal servicio es la iluminación inteligente, que aprovecha el espectro no licenciado, al igual que la mayoría de tecnologías LPWAN. Dada la poca información técnica encontrada sobre esta tecnología, se describen aquí algunos de sus dispositivos [14, 16].

Estación base de ultrabanda angosta

Estas estaciones base proporcionan conectividad inalámbrica para la conexión de grandes cantidades de dispositivos a largas distancias. Según Telensa [136], las estaciones tienen un alcance por encima de los 16 km y cada estación base puede conectar más de 5000 dispositivos. La estación base UNB de Telensa, que se muestra en la figura 14, se combina con el sistema de planificación que usa un mapa topográfico para la ubicación interactiva de las luces de la calle.

Figura 14. Estación base UNB Telensa



Fuente: tomada de [136].

Controlador universal de alumbrado

Estos controladores tienen un alcance de 16 km, consumen cuatro veces menos que los esquemas de red tipo malla, tienen sistema de posicionamiento global y NFC [137]. En la figura 15 se muestran dichos controladores.

Figura 15. Controladores para alumbrado público



Fuente: tomada de [137].

Urban IQ

Este es un dispositivo multipropósito que brinda acceso a variedad de sensores, entre los cuales están los siguientes: alumbrado público, temperatura de la carretera, Wi-Fi, velocidad del viento, sensores de ambiente (PM2.5, Ozono, NO₂), tráfico, ruido y radar. Entre sus especificaciones técnicas se encuentran procesadores de arquitectura ARM, puertos alámbricos e inalámbricos de conexión para PoE, USB, serial, Bluetooth, Wi-Fi y 802.15.4 [138]. El dispositivo se muestra en la figura 16.

Figura 16. Urban IQ para ciudades inteligentes



Fuente: tomada de [138].

Las principales características técnicas de los dispositivos de la tecnología Telensa se presentan en la tabla 3, en el siguiente capítulo.

Sigfox

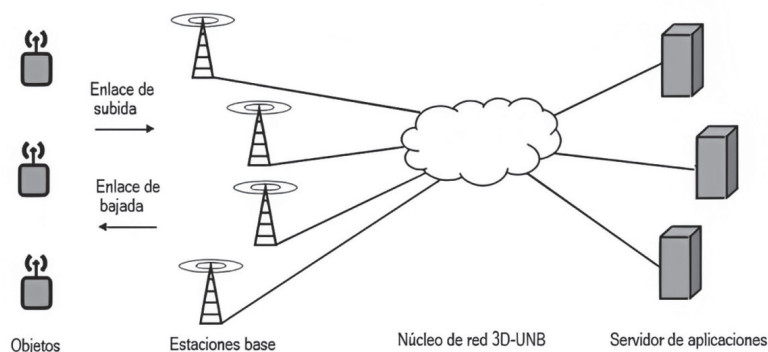
Sigfox es un operador de red que ofrece conexión de extremo a extremo para soluciones de IoT basadas en sus tecnologías. Las estaciones base son propiedad privada de la empresa y están equipadas con tecnologías de radio cognitiva (RC) y radio definida por *software* (SDR) [139]. Además, están conectadas a los servidores por medio de redes IP [123]. Paralelamente, los dispositivos finales que están conectados a dichas estaciones base usan modulación por desplazamiento de fase binaria (BPSK) en una banda portadora de Sub-GHz, con una UNB de BW igual a 100 Hz.

Sigfox usa las bandas industriales, científicas y médicas (ISM) no licenciadas; en la tabla 3 se dan más detalles de las frecuencias de ISM en diferentes regiones. Gracias a que emplea la UNB, aprovecha de manera más eficiente el BW, experimentando muy bajos niveles de ruido, un consumo de potencia menor, alta sensibilidad en el transmisor, un bajo costo en la antena y una velocidad de solo 100 bits por segundo (bps).

Sigfox aprovecha como solución una serie de reglas de comunicación conocidas como 3D-UNB (triple diversidad ultrabanda angosta), que se refiere a diversidad en tiempo, frecuencia y espacio. Este sistema diseñado por Sigfox es muy simple e implica un nivel de baja complejidad para los objetos conectados. Esta forma dinámica de emplear distintos protocolos de comunicación involucra la combinación de SDR y RC, dando la capacidad de diversas configuraciones en un esquema de red y distintas posibilidades en cada una de las capas de red. Sigfox es más un conjunto de “reglas de comunicación” que un protocolo en sí mismo [140]. En la figura 17 se observa la arquitectura general de la solución propuesta por Sigfox.

Arquitectura de red

En su capa física emplea UNB, mientras que los protocolos de capa de red son privados, por lo que no existe mucha información sobre cómo funcionan [121]. Sigfox explica sus servicios y dispositivos a partir de las aplicaciones en IoT que puede brindar, sin que se especifiquen detalles técnicos.

Figura 17. Arquitectura general de Sigfox

Fuente: tomada de [140].

Escalabilidad y longitud de carga útil

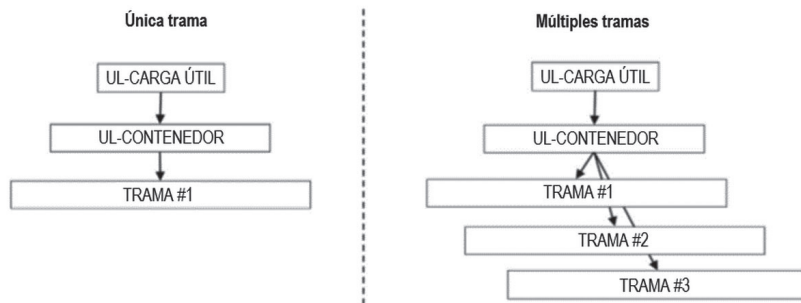
Aunque inicialmente Sigfox solo soportaba comunicación unidireccional del enlace de subida (UL), después la tecnología se adaptó para que fuese bidireccional; sin embargo, para que exista comunicación en el enlace de bajada (DL) debe haber una comunicación en el UL. En cuanto al UL, por cada mensaje un objeto puede transmitir de una a tres tramas, lo que comprende una disminución en el consumo de energía o el incremento de la velocidad y cantidad de tramas enviadas [140]. Este principio de transmisión se muestra en la figura 18. El número de mensajes en el UL se limita a 140 por día, pero el número de mensajes en el DL está limitado a tan solo 4 por día. La máxima longitud de carga útil para cada mensaje en el DL es de 8 *bytes*.

Como las estaciones base pueden recibir simultáneamente múltiples mensajes de todos los canales, los dispositivos pueden seleccionar aleatoriamente la frecuencia de un canal para transmitir su información. Esto simplifica en gran medida el diseño y el costo de los dispositivos finales.

Cobertura

Es de las más altas en las tecnologías LPWAN, alcanzando teóricamente una distancia de hasta 40 km. Otros detalles sobre el alcance y la cobertura se ofrecen en la tabla 3.

Figura 18. Principio de transmisión única o múltiple en Sigfox



Fuente: tomada de [140].

Vida útil de la batería y latencia

De manera similar a LoRa, Sigfox emplea un modo de reposo que le permite una reducción considerable en el consumo de energía, con lo cual se sacrifica la latencia de la red.

Modelo de despliegue

Sigfox es una tecnología madura que se ha expandido en el mundo. En su sitio web oficial se ofrece una herramienta virtual que cuenta con un mapa interactivo para saber en qué lugares existe cobertura o se está realizando un despliegue de la tecnología, así como un conjunto de enlaces para contactar a los distribuidores en cada país [141]. Sigfox se encuentra en aproximadamente 31 países, superada por LoRa, que cuenta con un despliegue en 42 países; pese a ello, estos datos son susceptibles a cambios debido a la rapidez con la que se masifica este tipo de tecnologías [123].

Seguridad

Aunque los protocolos de Sigfox son propietarios, de forma general se puede decir que Sigfox enfoca su seguridad en su misma red, dejando los mecanismos de seguridad a los usuarios finales, es decir, en el lado del transmisor. La seguridad queda rezagada al nodo de Sigfox, y, en el lado del receptor, a las aplicaciones enlazadas a la nube de Sigfox vía interfaces de programación de aplicaciones [121].

Aplicaciones de internet de las cosas con redes de área extensa de baja potencia

En general, el escenario actual para las redes LPWAN es de crecimiento favorable y continuo. Por tanto, la tendencia es a fortalecer y regular estas tecnologías [121, 122, 125]. La empresa analista Machina Research afirma que para el 2025 habrá tres billones de conexiones a redes LPWAN y que para el 2023 el 59% de las conexiones activas a redes LPWAN utilizará el espectro licenciado. Strategy Analytics estima que para el 2023 los operadores de red podrían recibir más de trece billones de dólares por concepto de tecnologías LPWAN, así como ingresos adicionales en servicios de valor agregado, tales como análisis de datos y seguridad [142].

A continuación se describen las principales aplicaciones de las tecnologías LPWAN.

Ciudades inteligentes

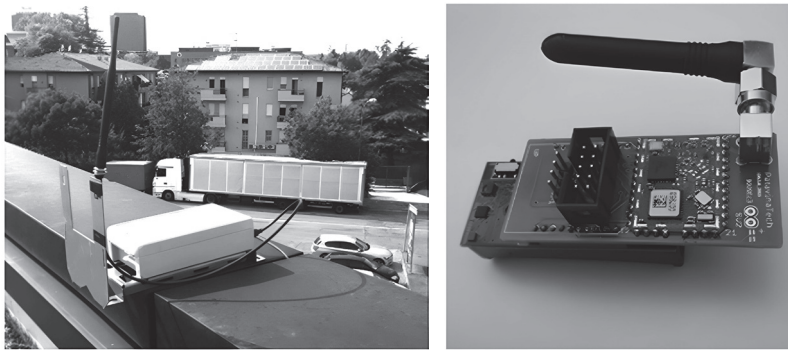
Las ciudades inteligentes utilizan tecnologías digitales integradas en todas sus funciones para mejorar el bienestar de los ciudadanos. Las ciudades con estas tecnologías registran mejoras en la eficiencia energética, el uso del agua, el tráfico de vehículos, la protección ambiental, la reducción de la contaminación, el cuidado de las personas mayores, la seguridad pública, las tasas de alfabetización, entre otros aspectos [143]. Lo anterior obliga a disponer de tecnologías de comunicación robustas como las LPWAN.

En general, en este diverso campo de estudio la elección de la tecnología está dada por el manejo de datos y la escalabilidad que se requiera dar. NB-IoT está implementada sobre bandas licenciadas, ofreciendo alta escalabilidad a las redes creadas. Varios desarrollos coinciden en que LoRa tiene uno de los mejores desempeños en esta área; por ejemplo, en [127] se compara la cobertura entre LoRaWAN, NB-IoT y Sigfox, destacando las ventajas que presenta el despliegue de LoRaWAN en una ciudad de Bélgica con una superficie total de 30 500 km², en la que se logró una cobertura total con tan solo siete estaciones base.

Por su parte, en [121] se propone un interesante estudio para hacer varias pruebas sobre la cobertura y despliegue de la tecnología LoRa. En la primera

prueba sobre el test de despliegue se evaluó la efectividad y alcance de una red privada LoRa instalada en un edificio de 19 pisos ubicado al norte de Italia, usando nodos y una *gateway* LoRa como los presentados en la figura 19. El objetivo del proyecto fue adaptar la red para capturar variables medioambientales (temperatura y humedad) dentro de varios pisos del edificio con el fin de reducir el costo de la ventilación, la calefacción y el aire acondicionado.

Figura 19. Dispositivos para la prueba de una red LoRa: *gateway* de referencia LoRa IoT modelo 0X80400AC (a) y nodos de referencia LoRa (b)



Fuente: tomada de [121].

Con una estación base y 32 nodos se dio cobertura a todos los sitios deseados; incluso se instaló un servidor web con una aplicación de monitoreo y bases de datos. Los resultados fueron favorables; aunque se probaron otras tecnologías y estructuras diferentes, no se alcanzaron resultados equiparables a los de la red LoRa [3].

Enseguida se reseñan las ventajas de las redes LoRa en el escenario de aplicación de ciudades inteligentes, por lo que se observa una preferencia en esta aplicación del IoT; sin embargo, esto no implica que otras tecnologías y esquemas no se implementen en este tipo de aplicaciones [144]-[146].

Iluminación inteligente

Es importante aclarar que para algunos autores la iluminación inteligente es un conjunto de aplicaciones inmersas en las soluciones de ciudades inteligentes, por sus diversas utilidades. Para este caso, las aplicaciones de ciudades inteligentes e iluminación inteligente se analizan de forma independiente.

Tecnologías como Telensa tienen un gran campo de acción en la aplicación de iluminación inteligente. En Gloucestershire (Reino Unido) Telensa desplegó un sistema de control para 55 000 paneles led para alumbrar la calle, cubriendo cerca de 1 000 millas cuadradas [147]. Esto redujo el consumo anual de energía en lo equivalente a 7 000 toneladas de carbón. El sistema consiste en fotoceldas conectadas a los postes de luz, que se comunican de forma inalámbrica y se controlan desde la nube [125, 147].

Automatización y monitoreo

En el campo de la robótica y automatización, el empleo de las LPWAN se enfoca especialmente en el monitoreo de sistemas complejos debido a la cantidad de procesos, las dimensiones, la arquitectura, el manejo de datos y el control a largas distancias de múltiples dispositivos simultáneamente. Esto se da gracias al alcance de esta tecnología para el control de múltiples dispositivos con un bajo consumo de energía, lo que implica, a su vez, una disminución de los costos de despliegue y del mantenimiento de los proyectos [148].

Es común encontrar aplicaciones orientadas al monitoreo de sistemas de ventilación, calidad de aire y sistemas de riego. Por ejemplo, en [149] y [150] se usa tecnología LPWAN para evaluar la calidad del aire en un recinto y en un área en campo abierto, respectivamente. Estos estudios son significativos por sus impactos en la metodología de evaluación de la calidad del aire. En ambos se utiliza el índice de calidad del aire, que establece una escala para evaluar qué tan insalubre puede ser la calidad del aire según una escala numérica. Para ello se dispone de sensores de humedad relativa, gases, entre otros, que permiten valorar distintas variables medioambientales relacionadas con la calidad del aire. Respecto al uso de tecnologías LPWAN, en [149] se aprovecha un módulo Arduino con conectividad LoRa, y en [150] un transceptor tipo LoRa RFM95. En ambos la información se direcciona a una *gateway* y de ahí a un dispositivo Android.

Por otra parte, algunos autores destacan la utilidad de la tecnología LPWAN en la automatización de procesos en el agro. En [151] se implementa un sistema de riego inteligente basado en *machine learning* con una red LoRa. Aquí se controlan válvulas solenoides para el riego en un cultivo.

El parámetro a controlar es la apertura y cerrado de la válvula, junto con el tiempo de este estado.

En [152] se emplea la tecnología NB-IoT, tras evaluar distintas alternativas de comunicación a larga distancia (otras tecnologías de LPWAN, ZigBee y GSM) en el monitoreo de un sistema de riego. El método que se ejecuta es similar al anterior: aquí se inicializan los módulos, luego se hace una lectura de los distintos sensores que permiten evaluar la necesidad de riego y las condiciones del cultivo, y finalmente se procesan los datos obtenidos. El resultado en este caso destaca el bajo consumo de energía y el alcance de cada uno de los nodos de la red implementada. En la figura 20 se presenta el dispositivo que sirvió para esta tarea.

Figura 20. Dispositivo instalado para monitoreo de sistema de riego con tecnología LPWAN



Fuente: tomada de [152].

Sistemas de localización

Algunas aplicaciones están encaminadas a la utilización o implementación de sistemas de localización. En [153] se ejecuta un sistema de localización por huella dactilar usando tecnología Sigfox; el proyecto resalta el bajo consumo de energía que tienen las redes LPWAN frente a otros métodos comunes de

localización. El resultado indica que con los algoritmos aplicados un transmisor Sigfox puede ser localizado con un error promedio de 320 m, y el error decrece a medida que se ponen en marcha más estaciones base.

Plataforma ThingSpeak

ThingSpeak es una plataforma que permite almacenar datos en la nube, de amplio beneficio para la comunidad de IoT. En ThingSpeak es posible agregar, visualizar y analizar información directamente en la nube. Entre sus características están las siguientes:

- Crea canales para subir, observar y analizar en vivo los datos en la nube.
- Es compatible con diferentes dispositivos.
- Es una plataforma muy estable y eficiente.
- Ejecuta análisis automáticamente con base en horarios o eventos.

Un canal está compuesto por los siguientes elementos:

- Ocho campos para almacenar datos de cualquier tipo: estos se pueden aprovechar para almacenar los datos de un sensor o de un dispositivo integrado.
- Tres campos de ubicación: se pueden utilizar para almacenar la latitud, la longitud y la elevación. Son muy útiles para rastrear un dispositivo en movimiento.
- Un campo de estado: es un mensaje corto para describir los datos almacenados en el canal [154, 155].

Parámetros de calidad de servicio

La UIT define la QoS como la totalidad de las características de un servicio de telecomunicaciones, que determinan su capacidad para satisfacer las necesidades explícitas e implícitas del usuario del servicio [156].

Estos parámetros de calidad dentro de las telecomunicaciones sirven para determinar la fiabilidad, confiabilidad y buen desempeño de una comunicación o red. Con los parámetros de QoS se puede fijar la cantidad de información o paquetes que se pierden, la cantidad de información que llega en

un lapso específico o la variación temporal en la recepción de los paquetes, entre otros. Con estos parámetros es posible establecer fallos en la comunicación o la confiabilidad de la misma. Los parámetros de QoS que se evalúan en la red IoT con tecnología LPWAN de esta investigación son: la pérdida de paquetes, el indicador de fuerza de la señal recibida (RSSI) o nivel de potencia recibida, el *throughput* y el *jitter*. Se describen ahora estos parámetros de QoS [157].

Pérdida de paquetes

Este parámetro de calidad mide la cantidad de paquetes de información que sale del extremo transmisor y no llega al extremo receptor, que es uno de los problemas frecuentes en una red. En protocolos como el de control de transmisión, que son orientados a la conexión, existen campos en las cabeceras para confirmar si la información ha llegado o no a su destino, con el fin de determinar si se reenvían o no los paquetes, por lo que su impacto puede ser mínimo. En tecnologías como la LoRaWAN la pérdida de paquetes puede convertirse en una dificultad relevante.

Indicador de fuerza de la señal recibida

Si el nivel de potencia de la señal que llega al receptor no alcanza el nivel de sensibilidad del receptor, la información no podrá ser detectada. Para determinar este parámetro de QoS basta con medir la potencia en el receptor. Este parámetro generalmente se mide en unidades de decibelio-milivatio (dBm).

Throughput

El *throughput* es un parámetro de QoS que indica la cantidad real de información por unidad de tiempo transferida en una comunicación. Por tanto, se mide la cantidad de bits que se transmiten o se reciben en un segundo; su unidad son los bps. En esta investigación, el *throughput* se calcula de la siguiente forma:

$$\text{Throughput} = \frac{(\text{Tamaño de la trama})}{(\text{Tiempo de ida y vuelta})} \quad (2)$$

Jitter

El *jitter* es la variación en el tiempo de recepción de los paquetes. Se calcula de la siguiente forma:

$$Jitter = \frac{\sum_{i=0}^n R_i - R_{i-1}}{n} \quad (3)$$

Donde R_i es el tiempo de recepción del paquete actual, R_{i-1} es el tiempo de recepción del paquete anterior y n es la cantidad de paquetes recibidos.

Capítulo 4

Metodología

En la tabla 3 se resumen las principales características de las tecnologías LPWAN estudiadas en el capítulo anterior. Aquí se observa que las principales diferencias se encuentran a nivel de rango de cobertura, en la que LoRa es un 500 % mayor que NB-IoT y un 50 % menor a Sigfox y Telensa para el entorno urbano, que es donde se desarrolla el proyecto. En lo que respecta a la velocidad, LoRa logra ser 500 veces más veloz que Sigfox y Telensa, pero cuatro veces más lenta que NB-IoT. En pruebas de campo, los resultados de estabilidad en la comunicación de estas tecnologías son cercanos [131]. Adicionalmente, para determinar qué tecnología de comunicación se utiliza en el desarrollo del proyecto se analizaron los costos de los dispositivos, la facilidad de adquirirlos en el país y la modularidad con otras tecnologías inalámbricas. En esto la tecnología LoRa tiene una amplia ventaja; por lo tanto, se selecciona esta tecnología para realizar la comunicación LPWAN, cuyas características de operación en Colombia, según la frecuencia, están definidas en la Resolución 105 de 2020 de la Agencia Nacional del Espectro [158].

Tabla 3. Características de las principales tecnologías basadas en LPWAN

Tecnología / Criterio	NB-IoT	LoRa	Telensa	Sigfox
Rango de transmisión	1 km (urbano) 25 m (rural)	20 km (urbano) 20 km (rural)	Hasta 10 km aproximadamente	10 km (urbano) 40 km (rural)
Licencia	Banda licenciada	Bandas no licenciadas	Bandas no licenciadas	Bandas no licenciadas

Continúa en la siguiente página

Tecnología / Criterio	NB-IoT	LoRa	Telensa	Sigfox
Ancho de banda	200 kHz	150, 250 y 500 kHz	1.1 kHz	100 Hz
Banda de frecuencias	Banda LTE, GSM	Sub-GHz ISM (868 MHz en Europa, 915 MHz en Norteamérica y 433 MHz en Asia)	Sub-GHz ISM (868 MHz en Europa, 915 MHz en Norteamérica y 433 MHz en Asia) / Espacios blancos de la TV	Sub-GHz ISM (868 MHz en Europa, 915 MHz en Norteamérica y 433 MHz en Asia) / Espacios blancos de la TV
Modulación	QPSK	CSS	UNB 2-FSK	BPSK
Velocidad de transmisión de datos	200 kbps	50 kbps	100 bps	100 bps
Máximas pérdidas de acoplamiento	164 dB	157 dB	154 dB	160 dB
Máxima potencia de transmisión	23/35 dBm	20 dBm	20 dBm	14/27 dBm
Inmunidad a interferencia	Baja	Alta	Alta	Alta
Comunicación bidireccional	Sí / <i>Half-duplex</i>	Dependiendo de la clase / <i>Half-duplex</i>	Sí	Limitada / <i>Half-duplex</i>
Estandarización	3GPP (GSM Association)	Lora-Alliance	A pesar de ser de arquitectura abierta trabaja de manera similar a Microsoft	Sigfox Company trabaja con el Instituto Europeo de Normas de Telecomunicaciones en la estandarización de las redes basadas en Sigfox
Duración de la batería	+ 10 años	+ 10 años	+ 10 años	+ 10 años
Aplicaciones sobre IoT	Ciudades inteligentes, parqueaderos inteligentes, agrónoma	Ciudades inteligentes, parqueaderos inteligentes, iluminación inteligente	Ciudades inteligentes, parqueaderos inteligentes, iluminación inteligente	Ciudades inteligentes, alarmas inteligentes, medidores inteligentes

Continúa en la siguiente página

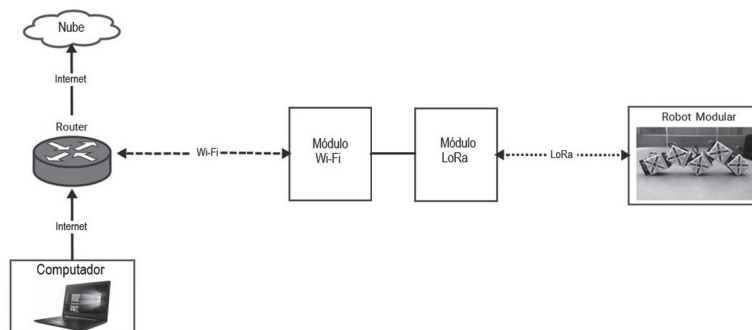
Tecnología / Criterio	NB-IoT	LoRa	Telensa	Sigfox
Costo en Colombia de dispositivo (incluidos los impuestos)	Medio	Bajo	Alto	Medio
Número de proveedores de dispositivos en Colombia	Pocos	Muchos	No hay	Pocos
Modularidad de dispositivos con otras tecnologías	Baja	Alta	Nula	Baja

Fuente: elaboración propia con base en [122], [125] y [131].

En la figura 21 se presentan los componentes del sistema implementado: el robot modular y dos dispositivos de comunicaciones marca Heltec (que contienen los módulos Wi-Fi y LoRa), uno que está ubicado en el robot modular para recibir y enviar la información por LoRa, y el otro que remite la información desde el computador a través del *router* usando la conexión Wi-Fi, y una vez llega esta información al módulo Heltec, la envía al robot usando el módulo LoRa. Este módulo también lleva la información de la posición del robot hacia la nube a través de la red de internet, actuando como una *gateway*.

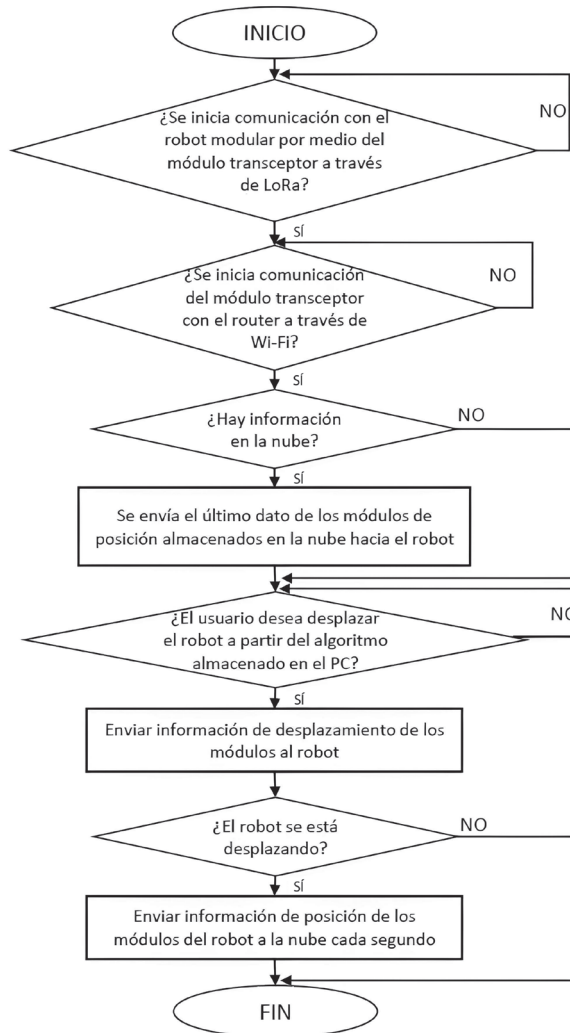
En la figura 22 se observa el diagrama de flujo que describe el funcionamiento de los componentes del sistema implementado. En las siguientes secciones se detallan estos componentes.

Figura 21. Esquema general del sistema desarrollado



Fuente: elaboración propia.

Figura 22. Diagrama de flujo del sistema desarrollado



Fuente: elaboración propia.

Robot modular EMERGE

La flexibilidad de adoptar morfologías arbitrarias en robótica modular está directamente relacionada con el diseño y estructura de los módulos; algunos tipos de robots tienen más limitaciones a la hora de formar configuraciones estructurales bidimensionales y tridimensionales debido a que sus módulos solo pueden conectarse a máximo dos módulos vecinos. El robot modular

EMERGE (*easy modular embodied robot generator*) es una alternativa de robot tipo cadena, reconfigurable manualmente, que puede formar estructuras de tipología 1D, 2D y 3D. Enseguida se explican las principales características de este robot.

El robot EMERGE que se observa en la figura 23(a) es un prototipo de uso libre; es decir, los materiales, circuitos eléctricos y el procedimiento necesario para su ensamblaje están disponibles en un repositorio [159]. Además, este prototipo es flexible, lo que le facilita al usuario adaptar los circuitos a sus necesidades particulares [159, 160]. Por ejemplo, en este caso se agregó al robot un circuito impreso con un dispositivo que posee las tecnologías Wi-Fi y LoRa incorporadas.

Básicamente, este robot se compone de varios módulos, como el de la figura 23(b), cuya agrupación depende del usuario. Cada módulo tiene cuatro caras con imanes, lo que le permite conectarse con otros módulos. La información se comparte usando el protocolo *controller area network* (CAN), que le ayuda al controlador a empaquetar, enviar y recibir la información hacia o desde un módulo o dispositivo específico. En la parte inferior de cada circuito impreso se encuentra un controlador local que controla el motor del módulo a partir de los parámetros recibidos por el controlador [159, 160]. Los programas y algoritmos de este robot pueden ser consultados en [161] y [162], y en [163] se proporcionan algunos videos sobre su funcionamiento.

Figura 23. Robot modular EMERGE: ejemplo de ensamble (a) y módulo del robot (b)



Fuente: tomada de [164].

La forma de control tradicional del robot EMERGE se basa en un controlador centralizado como el que muestra la figura 24, que se implementa sobre

la plataforma; es decir, el controlador tiene un algoritmo de control embebido que se ejecuta en tiempo real durante el funcionamiento de la plataforma [165]. Sin embargo, cuando la plataforma se apaga o algún módulo se desconecta, el controlador pierde la sincronización y el robot deja de moverse; en este caso, lo anterior se contrarresta al cargar los parámetros de control del robot a la nube, para que estén disponibles aun en ausencia de suministro eléctrico, lo que permite a la plataforma enviar los parámetros al robot una vez corregido el fallo de *hardware*.

Figura 24. Controlador del robot EMERGE



Fuente: tomada de [164].

Como se mencionó, el controlador centralizado coordina los módulos de la estructura conformada, lo que se consigue al generar secuencias de movimientos a partir de tablas de control de movimiento. Estas tablas se definen en el *firmware* del controlador y se construyen al descomponer puntos de la señal sinusoidal y traducirlos a las posiciones de cada módulo, como se muestra en los ejemplos de la tabla 4, cuyos parámetros fundamentales son la amplitud, la frecuencia y la diferencia de fase para cada uno de los módulos. Como resultado de esta función, se obtienen valores entre 0 y π , y luego se efectúa la conversión para la puesta en marcha del actuador de cada módulo, que recibe valores entre 0 y 1024, como se muestra en la figura 25. La conversión

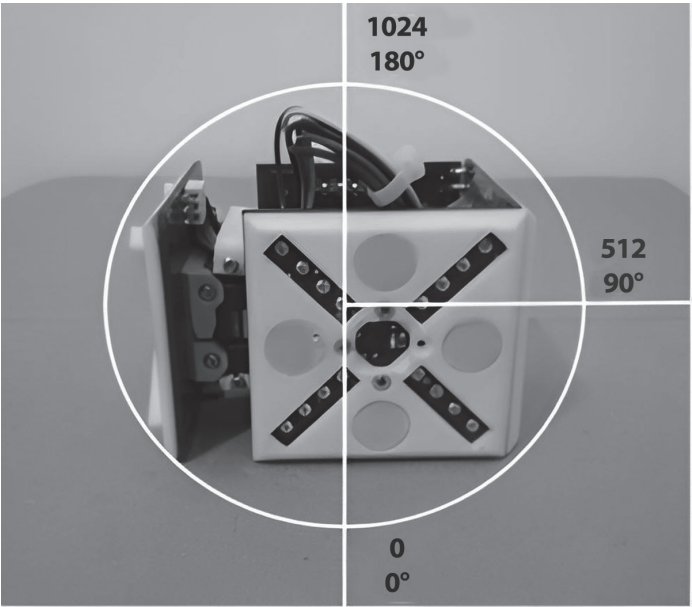
está limitada en el rango que oscila entre π y 3π , ya que valores menores o superiores a este rango comprometen la estructura mecánica del robot [164].

Tabla 4. Segmento de la rutina de control de movimiento

Módulo			1	2	3	4	5	6
Amplitud	1	Valor definido en el rango de movimiento para el motor						
Fase	90		420	380	450	630	550	450
Muestra de la posición angular	1							
Amplitud	1	Valor definido en el rango de movimiento para el motor						
Fase	90		500	420	380	540	630	490
Muestra de la posición angular	45							

Fuente: elaboración propia.

Figura 25. Rangos de movimiento del actuador de un módulo

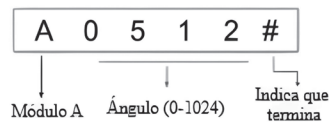


Fuente: tomada de [164].

Algoritmo para la coordinación y control de los movimientos del robot

El computador que muestra la figura 21 tiene las siguientes características: un procesador AMD RYZEN 3300 de cuatro núcleos con 16 GB de memoria RAM, que cumple con los requerimientos mínimos sugeridos para la ejecución del *software* MATLAB [164]. Este se encarga de ejecutar un algoritmo que estima la posición de los módulos mediante un generador sinusoidal; cada parámetro de movimiento está conformado por un empaquetamiento que almacena una cadena de caracteres que describen el nombre del módulo (A, B, C, etc.), el ángulo (en un rango de 0000-1024, como se muestra en la figura 25) y un carácter de terminación de la cadena representado por #, como se ilustra en la figura 26. Los paquetes se envían desde el computador al controlador centralizado del robot y este, a su vez, lo envía a cada módulo; luego, el módulo se posicionará en el ángulo correspondiente. Los parámetros de posición angular se suben a la nube (plataforma ThingSpeak); así, en el caso que el controlador general se desconecte o en el caso que uno o varios módulos del robot fallen, se podrá descargar la posición angular más reciente almacenada para cada módulo.

Figura 26. Paquete del parámetro de movimiento



Fuente: elaboración propia.

En el algoritmo 1 se observa una estructura básica de las órdenes dadas para el control de los movimientos del robot modular EMERGE.

Algoritmo 1. Funcionamiento del controlador

```

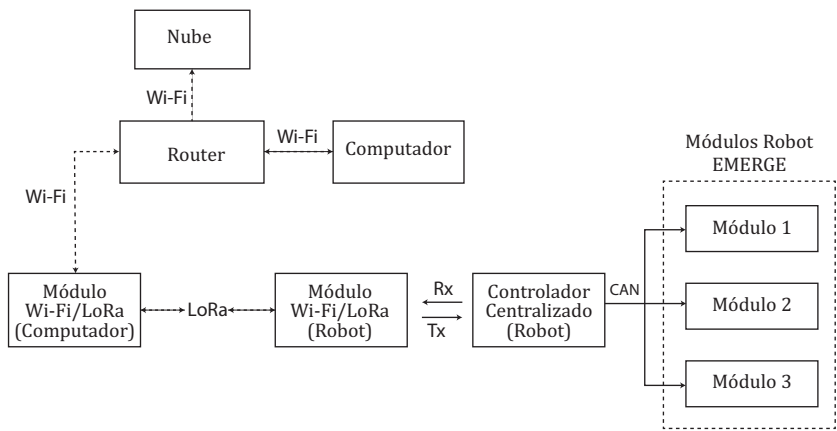
Función: estimar posición angular de los módulos ()
Definir  $A$ ,  $\theta$ ,  $t$ ,  $\varphi$ ,  $\Delta t$ 
 $A \leftarrow$  Amplitud
 $\theta \leftarrow$  Fase, la cual sincroniza los módulos
 $\varphi \leftarrow$  Desplazamiento angular fijo
 $t \leftarrow$  Tiempo
 $\Delta t \leftarrow$  Diferencia de tiempo
Iniciar dispositivo LoRa del robot
Iniciar dispositivo LoRa del router
Si (Datos en la nube  $\neq$  Vacío) entonces
   $[A, \theta, t, \varphi] \leftarrow$  Descargar datos de la nube
  Iniciar radioenlace
  Enviar bit de prueba
  Iniciar comunicación
   $c \leftarrow$  cantidad de módulos
   $M \leftarrow \pi/c$ 
  Para  $i = 0$  hasta  $c$ 
    haga  $I \leftarrow M*i$ 
  Si hay posiciones guardadas en la nube entonces
     $x_c \leftarrow$  Posiciones angulares almacenadas en la nube ( $c$ )
  En otro caso
     $x_c \leftarrow$  Crear arreglo vacío de dimensiones  $c$ 
  Mientras la condición de parada no esté activa haga
    Para  $i = 0$  hasta  $c$  haga
       $x[i] \leftarrow A*\text{sen}(\theta) + \varphi[i]*t$ 
      Enviar posición angular al módulo ( $x_i, i$ )
      Enviar posición angular a la nube ( $x_i, i$ )
       $t \leftarrow t + \Delta t$ 

```

Fuente: elaboración propia.

El controlador centralizado o general está conformado por un microcontrolador PSOC 4 que se encarga de gestionar todas las tareas propias del robot; este se comunica con los controladores locales por medio del protocolo CAN, como se mencionó anteriormente. Los dispositivos Wi-Fi/LoRa permiten comunicar el computador y el controlador centralizado del robot por medio del protocolo LoRa, como se observa en la figura 27. Los parámetros de posición angular se almacenan en la nube. Los datos son importados y exportados al controlador centralizado, lo que se hace en la plataforma ThingSpeak, ya que almacena temporalmente dichos parámetros en un archivo de texto plano en la nube cuando el robot termina la ejecución de un movimiento. La figura 28 muestra un ejemplo de este archivo.

Figura 27. Arquitectura del sistema de comunicación para enviar parámetros de movimiento al robot EMERGE



Fuente: elaboración propia.

Figura 28. Segmento del archivo de texto plano generado en la nube

parametros.txt: Bloc de notas

MÓDULO 1					MÓDULO 2			
Ítem	Amplitud	Fase	Offset	Posición	Angular		Amplitud	
1	1	pi	0	0	1	pi/2	0	0.9129452
2	1	pi	0	0.110882629	1	pi/2	0	0
3	1	pi	0	0.220397743	1	pi/2	0	0
4	1	pi	0	0.327194697	1	pi/2	0	0
5	1	pi	0	0.429956364	1	pi/2	0	0
6	1	pi	0	0.527415386	1	pi/2	0	0
7	1	pi	0	0.618369803	1	pi/2	0	0

Fuente: elaboración propia.

Sistema de comunicación de Internet de las cosas con una red de área extensa de baja potencia

El sistema de comunicación mostrado en la figura 21 y descrito en la figura 22 tiene la función de enviar los movimientos del robot desde el computador hacia el robot de múltiples módulos, además de enviar la información de posición de los módulos del robot EMERGE, cada cierto tiempo, a la nube haciendo uso de la red de internet. Esto se realiza para tener actualizada la posición de los módulos del robot, de manera que si hubiese una desconexión que desenergizara al robot, este pudiera obtener la posición más reciente de sus módulos desde la información de respaldo almacenada en la nube. Enseguida se presentan las ecuaciones fundamentales, los dispositivos y algoritmos que se emplearon para la puesta en marcha y la evaluación del sistema de comunicaciones.

Rata de bit en LoRa

En la modulación LoRa el espectro ensanchado se consigue generando una señal de barrido (del inglés *chirp*) que varía continuamente en frecuencia. Una ventaja de este método es que los desplazamientos de tiempo y frecuencia entre el transmisor y el receptor son equivalentes, lo que aminora en gran medida la complejidad del diseño del receptor. El BW de esta señal de barrido es equivalente al BW del espectro de la señal. La señal de datos deseada es barrida a una velocidad de datos más alta y se modula en la señal de barrido. La relación entre la rata de bits de datos deseada, la rata de símbolo y la rata de chip para la modulación LoRa se expresa de la siguiente forma [166]. La rata de bit modulada es:

$$R_b = SF \frac{1}{\left\lceil \frac{2^{SF}}{BW} \right\rceil} \quad (4)$$

Donde SF es el factor de ensanchamiento y BW es el ancho de banda de la señal modulada en Hz.

El periodo de símbolo es:

$$T_s = \frac{2^{SF}}{BW} \quad (5)$$

Por lo tanto, la rata de símbolo es:

$$R_s = \frac{1}{T_s} = \frac{BW}{2^{SF}} \quad (6)$$

Entonces, la rata de chip es:

$$R_c = R_s \times 2^{SF} \quad (7)$$

La modulación LoRa incluye un esquema de corrección de errores variable que mejora la robustez de la señal transmitida a expensas de la redundancia. Por tanto, la rata de bit nominal de la señal de datos es:

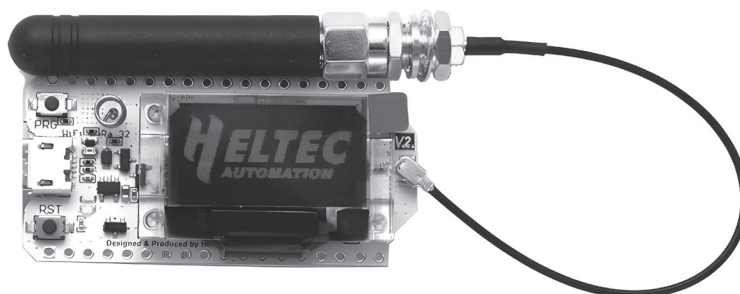
$$R_b = SF \left[\frac{\frac{4}{(4 + CR)}}{\frac{2^{SF}}{BW}} \right] \quad (8)$$

Donde CR es la rata de código [166].

Transceptor de internet de las cosas

El dispositivo seleccionado para transmitir y recibir información en el robot modular y desde el computador y hacia la nube es el Wi-Fi LoRa 32 marca Heltec, mostrado en la figura 29, que cuenta con las tecnologías inalámbricas Wi-Fi, LoRa y Bluetooth [167]. Este dispositivo permite enviar y recibir información mediante cualquiera de los tres protocolos de comunicación de manera individual o con alguna combinación. Para este caso, como ya se indicó, el dispositivo ubicado en el robot EMERGE transmite y recibe por el protocolo LoRa; el otro dispositivo, a su vez, se comunica con el robot por el protocolo LoRa y con el *router* a través del protocolo Wi-Fi para recibir información desde el computador y enviar o recibir información hacia la nube, como se presenta en la figura 21. La modularidad de este dispositivo hace que no sea necesario usar una *gateway*, lo que ahorra costos y espacio.

Figura 29. Dispositivo Wi-Fi LoRa 32 marca Heltec



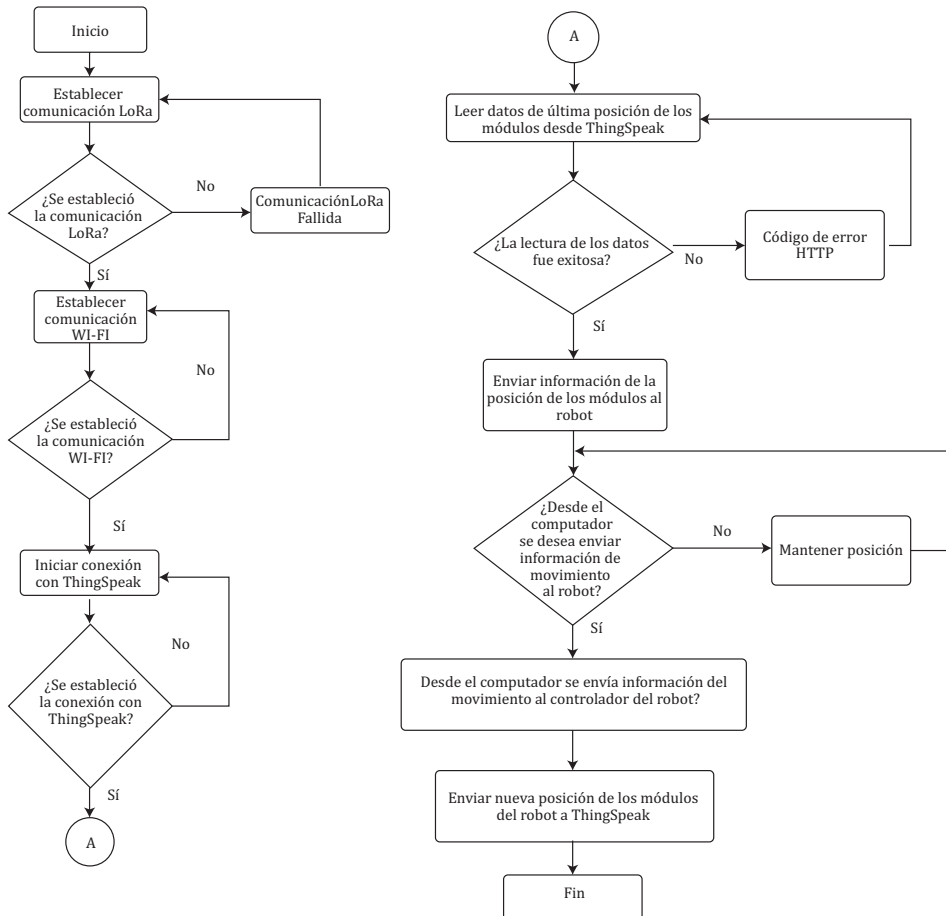
Fuente: tomada de [167].

Desarrollo del programa para el dispositivo que transmite información al robot

El programa diseñado para el dispositivo que envía/recibe la información al/del robot y también se comunica con el *router* para recibir información del computador y enviar o recibir la información de posición del robot hacia o desde la nube (este dispositivo actúa como *gateway*, ya que la información recibida con el protocolo LoRa la convierte al protocolo Wi-Fi para ser transmitida, o viceversa), usa las librerías para la comunicación de las tecnologías LoRa, Wi-Fi y la conexión con la plataforma ThingSpeak, donde se almacena y visualiza la información de la posición de los módulos del robot. Luego, se define la frecuencia de operación, que para este caso es 433 MHz.

El programa se divide en dos partes: la primera parte se ejecuta cuando se enciende el robot; por tanto, se inicializan los protocolos LoRa y Wi-Fi, además de establecerse conexión con la plataforma ThingSpeak. Después, se descarga el último dato de la posición angular de los módulos del robot que se encuentra almacenado en la nube, y se envía al robot. La segunda parte del programa se ejecuta cuando se envía la información desde el computador hacia el robot, indicando los movimientos que este debe realizar, información que solo debe enviarse una vez. Adicionalmente, se remite la información de la posición angular de los módulos del robot cada segundo hacia la plataforma ThingSpeak, donde se almacenan y actualizan estos datos constantemente. En la figura 30 se expone el diagrama de flujo del programa diseñado para este dispositivo.

Figura 30. Diagrama de flujo para programar el dispositivo que transmite información al robot



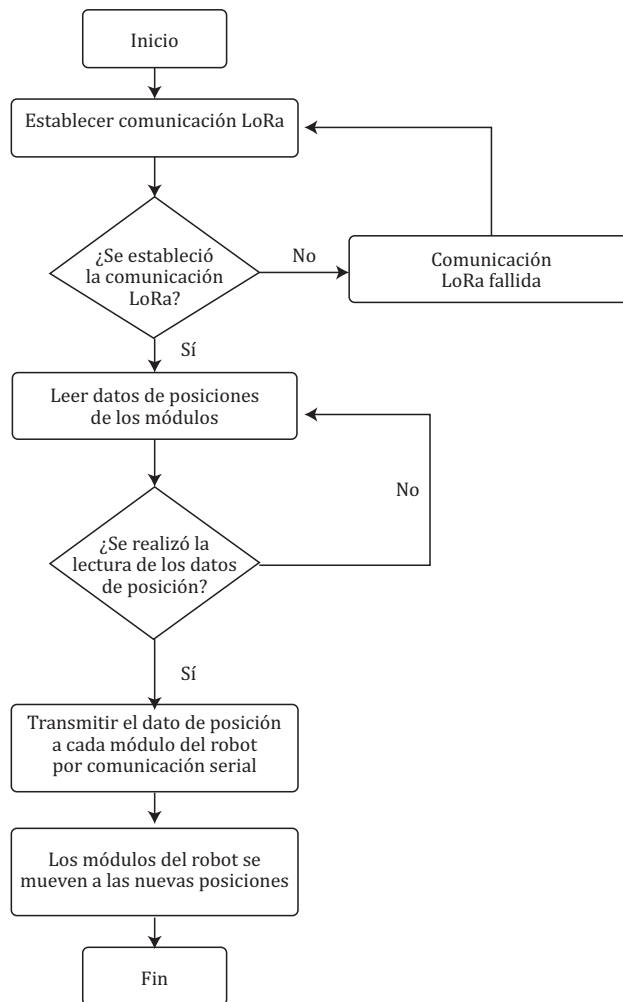
Fuente: elaboración propia.

Desarrollo del programa para el dispositivo que recibe información en el robot

El programa diseñado para el dispositivo que recibe los datos en el robot, al igual que el dispositivo anterior, primero inicializa las librerías necesarias, que para este caso corresponden a las del protocolo LoRa. Posteriormente, se define la banda de frecuencia, que es 433 MHz. Este programa recibe los datos a través del protocolo LoRa y se envían por comunicación serial al módulo controlador del robot, que luego distribuye la información recibida a los restantes módulos a través del protocolo CAN. En la figura 31 se

muestra el diagrama de flujo que representa el programa desarrollado para este dispositivo.

Figura 31. Diagrama de flujo para programar el dispositivo que recibe información en el robot



Fuente: elaboración propia.

Configuración y almacenamiento de la información en la plataforma ThingSpeak

El almacenamiento de la información de la posición angular de cada módulo del robot en la nube se hace en la plataforma ThingSpeak. Para que esta

se sincronice con el dispositivo transceptor Wi-Fi LoRa 32, se crean canales para cada uno de los módulos del robot modular (cuatro en total), y en la configuración de los canales se establecen los campos correspondientes a las variables medidas; así mismo, se determinan las claves de lectura para interactuar con el servidor. LoRaWAN incluye dos niveles de seguridad: uno para la red y otro para la aplicación. La seguridad de la red garantiza la autenticidad del nodo en la red, mientras que la capa de seguridad de la aplicación garantiza que el operador de red no tenga acceso a los datos de la aplicación del usuario final; esto se logra usando el algoritmo de cifrado AES128.

Capítulo 5

Evaluación y análisis de resultados

En esta sección se presentan las pruebas realizadas a los programas que se trataron en el capítulo anterior [168].

En el recuadro de la figura 32 se exponen las evidencias del programa desarrollado para la primera parte del dispositivo que transmite información al robot. Aquí se observa la inicialización correcta de los protocolos LoRa y Wi-Fi en un reducido tiempo (ver la parte izquierda); luego, a la derecha, en la parte inferior del recuadro, aparece la palabra *counter*, seguida del número 512, que corresponde a la última posición del motor de cada módulo descargada desde la plataforma ThingSpeak. El valor de la letra inicial (A, B, C o D) se usa para identificar cada uno de los cuatro módulos del robot modular; el número siguiente indica la posición angular del motor de cada módulo y el # representa el fin de la cadena.

En la figura 33 se ofrece la información de un ejemplo del programa desarrollado para la segunda parte del dispositivo que transmite información al robot. Desde el computador se envía la información (700) para el movimiento del motor del módulo B, que es seleccionada en el recuadro superior. Después, en el texto resaltado se indica que el valor de esta posición es actualizado exitosamente en la plataforma ThingSpeak. En el recuadro inferior se observa que después de doce segundos, la posición del módulo puede visualizarse en la nube porque la versión que se usa de ThingSpeak es gratuita; de lo contrario, la visualización podría materializarse cada segundo.

Figura 32. Inicialización del programa para el dispositivo que transmite información hacia el robot

```

> 0160 74 72 79 20 30 78 34 30 30 38 30 36 62 38 0d 0a try 0x100006b0..
23:15:45 +00:00.402 > 0170 53 65 72 69 61 6c 20 69 6e 69 74 69 61 6c 20 64 Serial initial d
23:15:45 +00:00.587 > 0180 6f 6e 65 0d 0a 79 6f 75 20 63 61 6e 20 73 65 65 one..you can see
> 0190 20 4f 4c 45 44 20 70 72 69 6e 74 65 64 20 4f 4c OLED printed OL
> 01a0 45 44 20 69 6e 69 74 69 61 6c 20 64 6f 6e 65 21 ED initial done!
23:15:45 +00:00.656 > 01b0 0d 0a 4c 6f 52 61 20 49 6e 69 74 69 61 6c 20 73 ..LoRa Initial s
23:15:46 +00:00.960 > 01c0 75 63 63 65 73 73 21 0d 0a 4c 6f 52 61 20 43 6f uccess!..LoRa Co
23:15:46 +00:01.046 > 01d0 6d 70 75 74 61 64 6f 72 0d 0a 41 74 74 65 6d 70 mputador..Attemp
> 01e0 74 69 6e 67 20 74 6f 20 63 6f 6e 6e 65 63 74 20 ting to connect
> 01f0 74 6f 20 53 53 49 44 3a 20 4d 79 20 41 53 55 53 to SSID: My ASUS
23:15:56 +00:11.045 > 0200 0d 0a 2e 2e 0a 43 6f 6e 6e 65 63 74 65 64 0d 0a ....Connected..
23:15:58 +00:13.583 > 0210 43 6f 75 6e 74 65 72 3a 20 35 31 32 0d 0a 43 6f Counter: 512..Co
> 0220 75 6e 74 65 72 3a 20 35 31 32 0d 0a 43 6f 75 6e unter: 512..Coun
> 0230 74 65 72 3a 20 35 31 32 0d 0a 43 6f 75 6e 74 65 ter: 512..Counte
> 0240 72 3a 20 35 31 32 0d 0a 41 30 35 31 32 23 0d 0a r: 512..A0512#..
> 0250 42 30 35 31 32 23 0d 0a 43 30 35 31 32 23 0d 0a B0512#..C0512#..
23:16:02 +00:17.095 > 0260 44 30 35 31 32 23 0d 0a 35 31 32 35 31 32 35 31 D0512#..51251251
23:16:04 +00:19.045 > 0270 32 35 31 32 35 31 32 35 31 32 0d 0a 35 31 32 0d 512512512..512..
> 0280 0a 35 31 32 0d 0a 35 31 32 0d 0a 21 21 20 0a 0d ..512..512..!! ..
23:16:04 +00:19.747 > 0290 0a 43 68 61 6e 6e 65 6c 20 75 70 64 61 74 65 20 ..Channel update
23:16:06 +00:21.094 > 02a0 73 75 63 63 65 73 73 66 75 6c 2e 0d 0a 35 31 32 successful...512
> 02b0 0d 0a 35 31 32 0d 0a 35 31 32 0d 0a 35 31 32 0d ..512..512..512..
23:16:07 +00:22.773 > 02c0 0a 21 21 20 0a 0d 0a 50 72 6f 62 6c 65 6d 20 75 ..!! ...Problem u
> 02d0 70 64 61 74 69 6e 67 20 63 68 61 6e 6e 65 6c 2e pdating channel.
> 02e0 20 48 54 54 50 20 65 72 72 6f 72 20 63 6f 64 65 HTTP error code

```

Fuente: elaboración propia.

Figura 33. Ejemplo de la segunda parte del programa para el dispositivo que transmite información hacia el robot

```

23:17:33 +01:48.195 > 0540 20 2d 33 30 33 0d 0a 36 30 30 36 30 30 0d 0a 35 -303..600600..5
> 0550 31 32 0d 0a 35 31 32 0d 0a 35 31 32 0d 0a 21 21 12..512..512..!!
> 0560 20 0a 0d 0a ...
23:17:33 +01:48.672 < 0000 42 30 37 30 30 23 B0700#
23:17:33 +01:48.807 > 0000 43 68 61 6e 6e 65 6c 20 75 70 64 61 74 65 20 73 Channel update s
23:17:34 +01:49.806 > 0010 75 63 63 65 73 73 66 75 6c 2e 0d 0a 42 30 37 30 uccessful...B070
23:17:36 +01:51.167 > 0020 30 33 0d 0a 36 30 30 36 30 30 0d 0a 35 31 32 0d 0#..600600..512.
> 0030 0a 35 31 32 0d 0a 35 31 32 0d 0a 21 21 20 0a 0d ..512..512..!! ..
23:17:37 +01:52.882 > 0040 0a 50 72 6f 62 6c 65 6d 20 75 70 64 61 74 69 6e ..Problem updatin
> 0050 67 20 63 68 61 6e 6e 65 6c 2e 20 48 54 54 50 20 g channel. HTTP
> 0060 65 72 72 6f 72 20 63 6f 64 65 20 2d 33 30 33 0d error code -303..
23:17:38 +01:53.216 > 0070 0a 36 30 30 0d 0a 35 31 32 0d 0a 35 31 32 0d 0a ..600..512..512..
23:17:40 +01:54.930 > 0080 35 31 32 0d 0a 21 21 20 0a 0d 0a 50 72 6f 62 6c 512..!! ...Probl
> 0090 65 6d 20 75 70 64 61 74 69 6e 67 20 63 68 61 6e em updating chan
> 00a0 6e 65 6c 2e 20 48 54 54 50 20 65 72 72 6f 72 20 nel. HTTP error
23:17:41 +01:56.188 > 00b0 63 6f 64 65 20 2d 33 30 33 0d 0a 36 30 30 0d 0a code -303..600..
> 00c0 35 31 32 0d 0a 35 31 32 0d 0a 35 31 32 0d 0a 21 512..512..512..!
23:17:42 +01:57.797 > 00d0 21 20 0a 0d 0a 50 72 6f 62 6c 65 6d 20 75 70 64 ! ...Problem updat
> 00e0 61 74 69 6e 67 20 63 68 61 6e 6e 65 6c 2e 20 48 ating channel. H
> 00f0 54 54 50 20 65 72 72 6f 72 20 63 6f 64 65 20 2d TTP error code -
23:17:43 +01:58.135 > 0100 33 30 33 0d 0a 35 31 32 36 30 30 0d 0a 35 31 32 303..512600..512
> 0110 0d 0a 35 31 32 0d 0a 35 31 32 0d 0a 21 21 20 0a ..512..512..!! ..
23:17:44 +01:59.846 > 0120 0d 0a 50 72 6f 62 6c 65 6d 20 75 70 64 61 74 69 ..Problem updatin
> 0130 6e 67 20 63 68 61 6e 6e 65 6c 2e 20 48 54 54 50 erg channel. HTTP
> 0140 20 65 72 72 6f 72 20 63 6f 64 65 20 2d 33 30 33 error code -303
23:17:45 +02:00.594 > 0150 0d 0a 37 30 30 36 30 30 0d 0a 37 30 30 0d 0a 35 ..700600..700..5
23:17:47 +02:02.303 > 0160 31 32 0d 0a 35 31 32 0d 0a 21 21 20 0a 0d 0a 50 12..512..!! ...P
> 0170 72 6f 62 6c 65 6d 20 75 70 64 61 74 69 6e 67 20 roblem updating
> 0180 63 68 61 6e 6e 65 6c 2e 20 48 54 54 50 20 65 72 channel. HTTP er
23:17:48 +02:03.054 > 0190 72 6f 72 20 63 6f 64 65 20 2d 33 30 33 0d 0a 36 ror code -303..6
> 01a0 30 30 36 30 30 0d 0a 37 30 30 0d 0a 35 31 32 0d 00600..700..512.
23:17:49 +02:04.761 > 01b0 0a 35 31 32 0d 0a 21 21 20 0a 0d 0a 50 72 6f 62 ..512..!! ...Prob
> 01c0 6c 65 6d 20 75 70 64 61 74 69 6e 67 20 63 68 61 lem updating cha

```

Fuente: elaboración propia.

En el recuadro de la figura 34, correspondiente al programa para el dispositivo que recibe la información en el robot, se revela que la comunicación LoRa se inicializa correctamente, y luego se presentan los datos recibidos de la posición angular para cada módulo.

Figura 34. Ejemplo del programa para el dispositivo que recibe información en el robot

```

> 0160 74 72 79 20 30 78 34 30 30 38 30 36 62 38 0d 0a try 0x400806b8...
23:14:36 +00:00.337 > 0170 53 65 72 69 61 6c 20 69 6e 69 74 69 61 6c 20 64 Serial initial d
23:14:36 +00:00.522 > 0180 6f 6e 65 0d 0a 79 6f 75 20 63 61 6e 20 73 65 65 one..you can see
> 0190 20 4f 4c 45 44 20 70 72 69 6e 74 65 64 20 4f 4c OLED printed OL
> 01a0 45 44 20 69 6e 69 74 69 61 6c 20 64 6f 6e 65 21 ED initial done!
23:14:36 +00:00.590 > 01b0 0d 0a 4c 6f 52 61 20 49 6e 69 74 69 61 6c 20 73 ..LoRa Initial s
23:14:37 +00:00.895 > 01c0 75 63 63 65 73 73 21 0d 0a 4c 6f 52 61 20 52 6f uccess!..LoRa Ro
23:15:35 +00:59.555 > 01d0 62 6f 74 0d 0a ff 0d 0a ff 0d 0a 43 30 35 31 32 bot.....C0512
23:15:58 +01:22.520 > 01e0 23 44 30 35 31 32 23 0d 0a ff 0d 0a ff 0d 0a ff #D0512#.....
23:15:59 +01:22.817 > 01f0 0d 0a 42 30 35 31 32 23 43 30 35 31 32 23 44 30 ..B0512#C0512#D0
23:16:40 +02:03.907 > 0200 35 31 32 23 0d 0a ff 0d 0a 41 30 36 30 30 23 0d 512#....A0600#.
23:18:34 +03:58.369 > 0210 0a ff 0d 0a 42 30 37 30 30 23 0d 0a ff 0d 0a 44 ....B0700#....D
23:19:08 +04:32.492 > 0220 30 34 30 30 23 0d 0a ff 0d 0a 41 30 35 31 32 23 0400#....A0512#
23:20:13 +05:36.892 > 0230 0d 0a ff 0d 0a 42 30 34 31 32 23 0d 0a ff 0d 0a ....B0412#....
23:20:35 +05:58.671 > 0240 44 30 36 30 30 23 0d 0a ff 0d 0a 41 30 33 30 30 D0600#....A0300
23:22:18 +07:42.388 > 0250 23 0d 0a ff 0d 0a 42 30 37 35 30 23 0d 0a ff 0d #....B0750#....
23:22:40 +08:04.507 > 0260 0a 41 30 36 30 30 23 0d 0a ff 0d 0a 42 30 33 30 .A0600#....B030
23:23:27 +08:50.656 > 0270 30 23 0d 0a ff 0d 0a 44 30 33 30 30 23 0d 0a ff 0#....D0300#...
23:23:27 +08:51.029 > 0280 0d 0a 41 30 33 30 30 23 0d 0a ..A0300#..

```

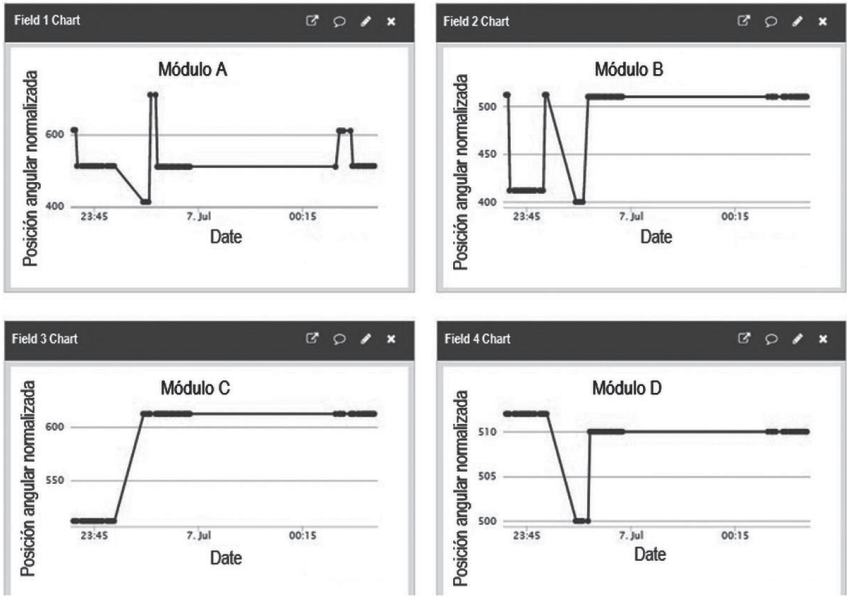
Fuente: elaboración propia.

Posteriormente, en la plataforma ThingSpeak se almacena continuamente la posición angular de cada módulo del robot modular. Después, cuando el robot se inicializa como consecuencia de una desconexión o de un comienzo de actividad, se descarga y se envía hacia el robot la última posición de cada módulo almacenada en la nube, con el propósito de darle continuidad al proceso de movimiento del robot.

La figura 35 es un ejemplo de las diferentes posiciones angulares que se envían desde el robot. Cada campo representa la posición de un módulo del robot, en estos se puede hacer un seguimiento alrededor de cada doce segundos con respecto a los movimientos hechos por cada módulo.

Por último, la figura 36 muestra al controlador del robot recuperando los parámetros de control desde la nube, con lo que termina de efectuar el movimiento indicado (esto en el caso de un reinicio forzado).

Figura 35. Almacenamiento y visualización de la posición angular de los módulos del robot en la plataforma ThingSpeak



Fuente: elaboración propia.

Figura 36. Módulos del robot EMERGE moviéndose a la última posición almacenada y descargada de la nube



Fuente: elaboración propia.

Evaluación y análisis de los parámetros de calidad de servicio del sistema de comunicaciones IoT

Los parámetros de QoS para evaluar el sistema de comunicaciones IoT corresponden a los valores promedio de los siguientes factores: el RSSI o nivel de potencia recibida, el *jitter*, la pérdida de paquetes y el *throughput*. Esta evaluación se ejecuta en un entorno interior a 20 m, 100 m y 150 m de distancia entre los dispositivos transceptores (la limitación en la distancia máxima fue consecuencia de la etapa de confinamiento por la emergencia sanitaria del covid-19), y en el enlace en el que se envía la información hacia el robot de múltiples módulos. El mensaje se intenta enviar por el usuario desde el computador hacia el robot cada ocho segundos, teniendo como limitante el tiempo mínimo entre envío de mensajes para algunos casos. La distancia más corta tiene LOS, mientras que las dos restantes no poseen LOS (en 100 m se tiene un obstáculo y en 150 m, dos obstáculos de aproximadamente 15 cm de ancho cada uno, hechos en hormigón). También se ajustan diferentes valores de SF, BW y CR a una frecuencia de 433 MHz. Para cada configuración se envía un total de 22 paquetes de información para el desplazamiento del robot modular.

Para realizar las mediciones de QoS se adquieren los datos del puerto serie por parte de un computador, con una configuración típica de 8 bits a 9600 bps, usando microcontroladores Arduino Nano. Para esto se utiliza como *sniffer* el *software* IO Ninja, que monitoriza los paquetes de cada uno de los transceptores. Luego de capturar los paquetes, se exportan a archivos de texto con extensión *txt*. Después, los archivos se almacenan en un computador con sistema operativo Linux CentOS 8 para ser analizados por un *script* desarrollado en el *software* Bash. Vale la pena aclarar que LoRa requiere de un tiempo mínimo entre envío de mensajes, según la configuración. Por tanto, mediciones como la del *jitter* se convierten en todo un reto, lo que implicó, en ese caso, restar este tiempo con el fin de equiparar los resultados obtenidos.

En las tablas 5-7 y las figuras 37-46 se indican los parámetros de QoS medidos a las distancias de transceptores de 20 m, 100 m y 150 m. Adicionalmente, se puede advertir la rata de bit teórica hallada a partir de la ecuación (8).

Tabla 5. Parámetros de QoS para una distancia de transceptores de 20 m

BW (kHz)	SF	$\frac{4}{4 + CR}$	Longitud del preámbulo (símbolos)	Throughput (bps)	Rb (bps)	Jitter (s)	Porcentaje de paquetes perdidos	RSSI (dBm)
125	7	$\frac{4/5}{4/8}$	8	3906 2909	5468.7 3417.9	10.27 8.5	13.6 0	-74.15 -75.6
125	8	$\frac{4/5}{4/8}$	8	2147 1360	3125 1953.1	9.31 8.58	9 0	-72.23 -74.9
125	9	$\frac{4/5}{4/8}$	8	1295 907	1757.8 1098.6	8.47 8.63	0 0	-76.11 -69.82
125	10	$\frac{4/5}{4/8}$	8	640 463	976.5 610.3	8.72 9.42	0 4.54	-78.7 -73.65
125	11	$\frac{4/5}{4/8}$	8	324 231	537.1 335.7	9.31 9.74	0 0	-74.42 -73.26
125	12	$\frac{4/5}{4/8}$	8	126 138	292.9 183.1	10.12 10.95	0 0	-71.29 -75.48
250	7	$\frac{4/5}{4/8}$	8	8311 5436	10937.5 6835.9	8.56 8.19	4.54 0	-77.3 -68.3
250	8	$\frac{4/5}{4/8}$	8	4441 2806	6250 3906.2	8.15 8.23	0 0	-73.55 -79.21
250	9	$\frac{4/5}{4/8}$	8	2593 1749	3515.6 2197.2	9.05 8.29	9 0	-75.1 -72.28
250	10	$\frac{4/5}{4/8}$	8	1185 907	1953.1 1220.7	8.36 8.44	0 0	-77.66 -78.42
250	11	$\frac{4/5}{4/8}$	8	707 412	1074.2 671.3	8.57 8.72	0 0	-71.86 -76.46
250	12	$\frac{4/5}{4/8}$	8	358 285	585.9 366.2	8.95 9.43	0 0	-73.74 -72.69

Fuente: elaboración propia.

Tabla 6. Parámetros de QoS para una distancia de transceptores de 100 m

BW (kHz)	SF	$\frac{4}{4 + CR}$	Longitud del preambulo (símbolos)	Throughput (bps)	Rb (bps)	Jitter (s)	Porcentaje de paquetes perdidos	RSSI (dBm)
125	7	4/5 4/8	8	2518 1896	5468.7 3417.9	12.08 11.63	27.2 22.7	-86.66 -84.38
125	8	4/5 4/8	8	1593 985	3125 1953.1	12.15 10.36	27.2 13.63	-89.77 -85.51
125	9	4/5 4/8	8	1059 836	1757.8 1098.6	9.5 9.15	4.54 0	-81.2 -87.49
125	10	4/5 4/8	8	557 326	976.5 610.3	10.28 9.92	9 4.54	-85.46 -86.42
125	11	4/5 4/8	8	315 208	537.1 335.7	9.41 9.29	0 0	-85.84 -87.25
125	12	4/5 4/8	8	85 78	292.9 183.1	10.31 11.52	0 0	-81.36 -86.48
250	7	4/5 4/8	8	7231 4758	10937.5 6835.9	10.05 8.68	13.63 0	-82.54 -79.69
250	8	4/5 4/8	8	4068 2389	6250 3906.2	10.63 8.72	18.18 0	-87.52 -79.3
250	9	4/5 4/8	8	2115 1459	3515.6 2197.2	9.19 9.73	4.54 9	-86.48 -84.36
250	10	4/5 4/8	8	933 872	1953.1 1220.7	8.72 8.91	0 0	-85.61 -81.62
250	11	4/5 4/8	8	782 389	1074.2 671.3	9.37 9.25	4.54 0	-84.93 -79.96
250	12	4/5 4/8	8	297 203	585.9 366.2	9.36 9.85	0 0	-80.41 -82.65

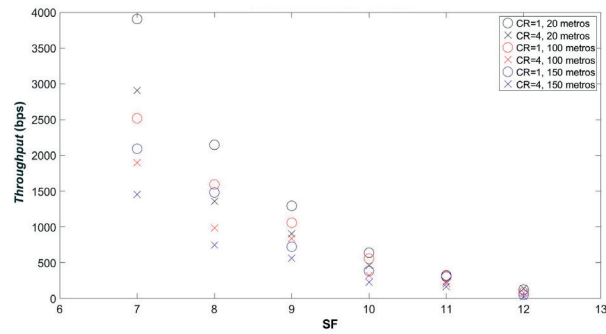
Fuente: elaboración propia.

Tabla 7. Parámetros de QoS para una distancia de transceptores de 150 m

BW (kHz)	SF	$\frac{4}{4 + CR}$	Longitud del preámbulo (símbolos)	Throughput (bps)	Rb (bps)	Jitter (s)	Porcentaje de paquetes perdidos	RSSI (dBm)
125	7	$\frac{4/5}{4/8}$	8	2093 1455	5468.7 3417.9	13.92 10.89	36.36 18.18	-95.62 -93.95
125	8	$\frac{4/5}{4/8}$	8	1482 749	3125 1953.1	12.21 11.73	27.2 22.7	-92.46 -92.84
125	9	$\frac{4/5}{4/8}$	8	724 561	1757.8 1098.6	13.28 10.15	31.81 9	-94.35 -90.62
125	10	$\frac{4/5}{4/8}$	8	386 227	976.5 610.3	9.75 9.53	4.54 0	-89.31 -86.22
125	11	$\frac{4/5}{4/8}$	8	304 164	537.1 335.7	9.94 10.32	0 0	-83.78 -90.63
125	12	$\frac{4/5}{4/8}$	8	53 27	292.9 183.1	10.86 11.67	0 0	-88.36 -97.23
250	7	$\frac{4/5}{4/8}$	8	5316 3645	10937.5 6835.9	11.14 9.58	22.7 9	-91.33 -97.06
250	8	$\frac{4/5}{4/8}$	8	2831 1724	6250 3906.2	12.83 10.74	31.81 18.18	-96.1 -93.27
250	9	$\frac{4/5}{4/8}$	8	1536 1073	3515.6 2197.2	9.22 8.93	4.54 0	-91.11 -89.24
250	10	$\frac{4/5}{4/8}$	8	686 648	1953.1 1220.7	9.85 9.12	9 0	-86.41 -88.01
250	11	$\frac{4/5}{4/8}$	8	497 239	1074.2 671.3	9.62 9.30	4.54 0	-92.53 -91.56
250	12	$\frac{4/5}{4/8}$	8	124 131	585.9 366.2	9.57 9.92	0 0	-96.49 -85.73

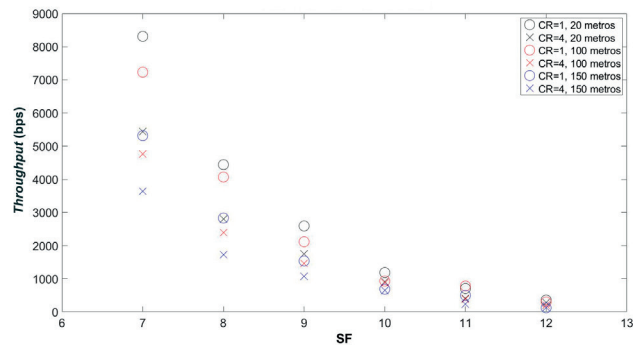
Fuente: elaboración propia.

Figura 37. *Throughput* para BW de 125 kHz



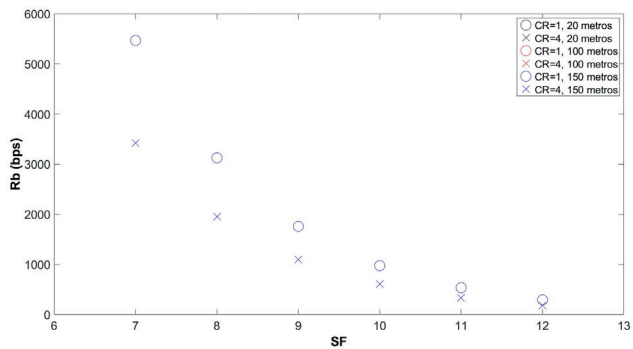
Fuente: elaboración propia.

Figura 38. *Throughput* para BW de 250 kHz



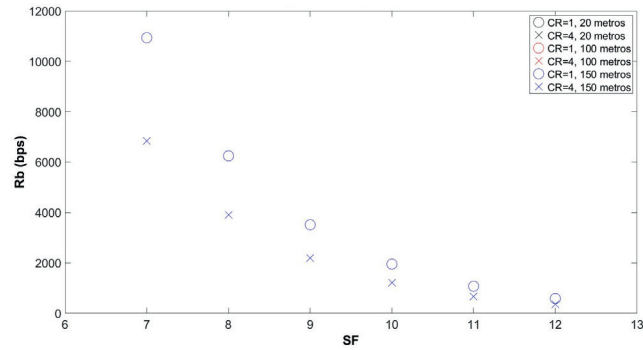
Fuente: elaboración propia.

Figura 39. Rata de bit para BW de 125 kHz



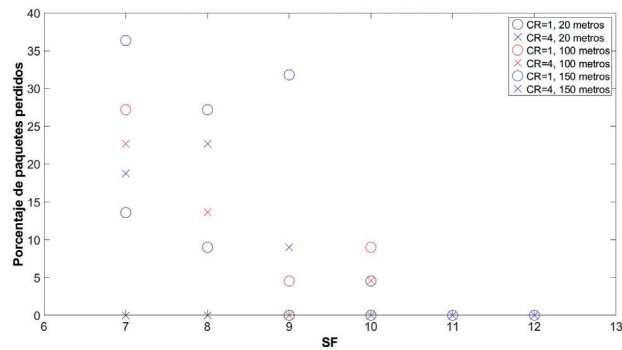
Fuente: elaboración propia.

Figura 40. Rata de bit para BW de 250 kHz



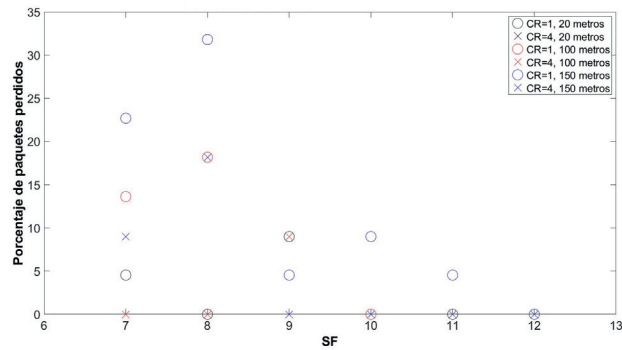
Fuente: elaboración propia.

Figura 41. Porcentaje de paquetes perdidos para BW de 125 kHz



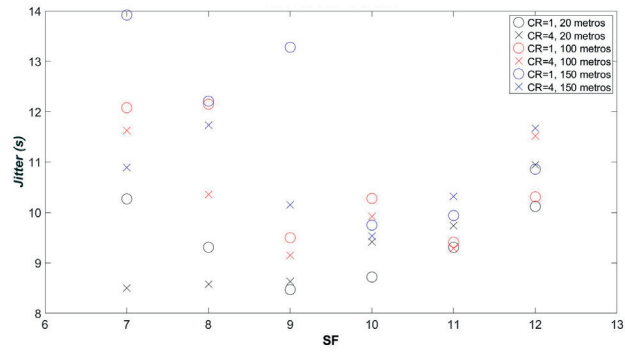
Fuente: elaboración propia.

Figura 42. Porcentaje de paquetes perdidos para BW de 250 kHz



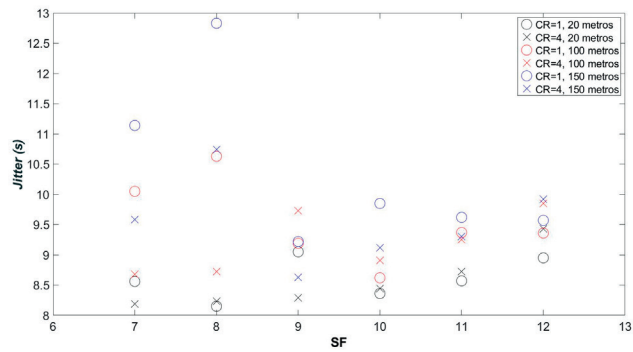
Fuente: elaboración propia.

Figura 43. *Jitter* para BW de 125 kHz



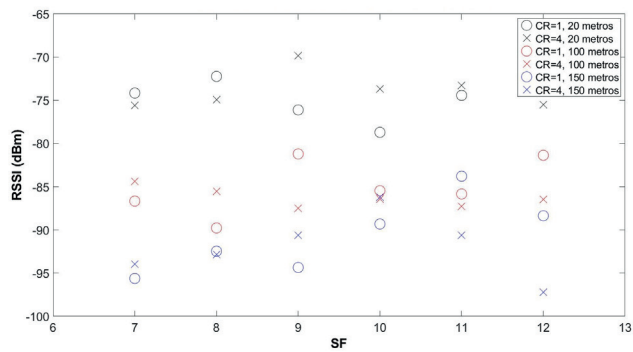
Fuente: elaboración propia.

Figura 44. *Jitter* para BW de 250 kHz



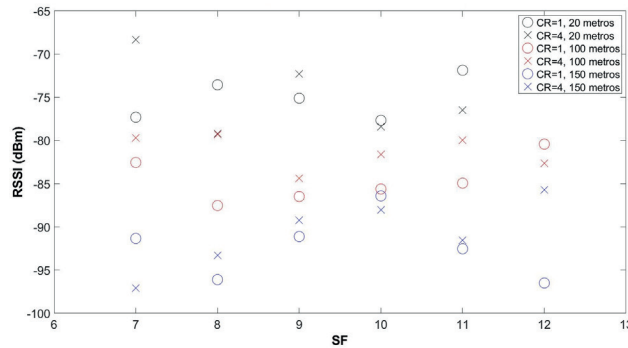
Fuente: elaboración propia.

Figura 45. Nivel de potencia recibida para BW de 125 kHz



Fuente: elaboración propia.

Figura 46. Nivel de potencia recibida para BW de 250 kHz



Fuente: elaboración propia.

En las tablas 5-7 y en las figuras 37 y 38 se advierte que al aumentar la CR de 1 a 4 (lo que corresponde a que cada cuatro bits de información se van a codificar por 5 u 8 bits para ser transmitidos) para una misma distancia, se disminuye el *throughput* como consecuencia de la adición de bits de redundancia, que incrementan la fiabilidad de la transmisión al disminuir la cantidad de errores, pero acortan el número de bits de información transmitidos por unidad de tiempo; es decir, en la trama se aumenta el número de bits de redundancia respecto a los bits de información útil. Adicionalmente, entre mayor es el SF, menor es el *throughput*, puesto que se amplía el número de bits por símbolo, que a su vez incrementa el tamaño del preámbulo y, por ende, recorta el tamaño de la carga útil en la trama. Además, al revisar las figuras 37-40 se nota que el *throughput* medido mantiene una diferencia considerable respecto a la Rb teórica, y se incrementa coherentemente al aumentarse el BW. Contrario a lo anterior, el *throughput* se reduce con el crecimiento de la distancia.

Otro aspecto que se examina en las tablas 5-7 y en las figuras 41 y 42 es que a mayor SF, generalmente se presenta menor pérdida de paquetes, ya que aumenta la sensibilidad en el receptor, lo que redundaría en un aumento en la relación señal a ruido. También se aprecia que a menor distancia y a mayor BW, se tiene una tendencia a la reducción de la pérdida de paquetes.

Así mismo, en las tablas 5-7 y en las figuras 43 y 44 se percibe que para el BW de 250 kHz frecuentemente se reduce el *jitter* respecto al BW de 125 kHz.

Incluso, el *jitter* se incrementa como consecuencia del aumento de la pérdida de paquetes y el crecimiento del tiempo en el aire de la información. En [169] se evidencia que esto último es directamente proporcional al aumento del SF. Además, la ampliación de la distancia entre transceptores genera un aumento en el *jitter*.

Finalmente, en las tablas 5-7 y en las figuras 45 y 46 se constata una disminución en el RSSI a medida que la distancia entre transceptores es mayor. También se contempla una leve mejora en el RSSI al ampliar el SF para una misma distancia.

Conclusiones

- El algoritmo de control desarrollado faculta al robot modular EMERGE para realizar diferentes movimientos, aun si se induce un reinicio forzado al sistema. Normalmente estos movimientos se coordinan para que los módulos del robot resuelvan diferentes tareas que no están predefinidas.
- La robótica modular ha sido ampliamente estudiada, haciendo énfasis en la morfología y los sistemas de control, como se presenta en [95], [96], [113] y [114]; sin embargo, aún queda un gran campo por explorar. Respecto a los sistemas de comunicaciones utilizados, lo más próximo a esta investigación fue el uso de sistemas de control inalámbricos, que permiten a los módulos formar estrategias para moverse en su entorno al enviar mensajes desde el controlador hacia los módulos directamente; no obstante, por ser controladores locales, si se interrumpe el suministro eléctrico por desconexión (probable en la estructura robótica utilizada), los parámetros de control y movimiento del robot se pierden. Lo anterior es subsanado con esta investigación, ya que el usuario podrá ahora mantener los parámetros de control del robot almacenados en la nube usando un sistema de comunicaciones de IoT LPWAN.
- En la actualidad, las aplicaciones robóticas como la de esta investigación requieren de una infraestructura de comunicaciones que no solo garantice que se transmita la información, sino que asegure el almacenamiento y la consulta de la misma en la nube. En este caso, el uso de la tecnología LPWAN permitió obtener la cobertura necesaria a un bajo costo. La elección de la tecnología LPWAN está basada principalmente en tres factores: el económico, la aplicación y el soporte. El factor económico se

refiere al costo de despliegue y mantenimiento de la tecnología; para el caso de esta investigación, se presenta un menor costo en infraestructura para cubrir un área con respecto a tecnologías como Wi-Fi, Bluetooth y ZigBee. La aplicación alude a la necesidad en concreto que se desea suplir, en la que se le puede dar prioridad, como se presenta en la tabla 3 (por ejemplo: a la cobertura, la velocidad o la interferencia), puesto que las tecnologías reportan diferencias para las aplicaciones basadas en IoT. Finalmente, el soporte se relaciona con la facilidad que existe en el desarrollo de la tecnología en la región donde se está aplicando y la legislación existente para el uso de la misma; para el caso de esta investigación, como se expuso en la tabla 3, existe un número superior de proveedores que le dan mayor soporte a la tecnología LoRa en Colombia, respecto a las otras tecnologías analizadas.

- En un sistema de comunicaciones LPWAN con tecnología LoRa, los valores de SF, BW y CR deben ser ajustados según las necesidades de la aplicación. Para este caso, se prioriza la confiabilidad de la información sobre el *throughput*, ya que no es necesario enviar información frecuentemente y a una alta velocidad al robot modular para indicarle los movimientos; sin embargo, sí se requiere que el robot modular envíe a menudo la información de la posición angular de sus módulos a la nube y que la información se pierda lo menos posible. Por esto, se utilizaron dos soluciones: la primera, en la que, si el robot modular solo tenía cobertura de la red LoRa, se recomendaron los siguientes valores: BW igual a 250 kHz, SF entre 9 y 10 (para que el tiempo de envío entre mensajes sea corto) y CR de 4. La segunda, en la que si el robot modular, además de tener cobertura de red LoRa, disponía de acceso a una red Wi-Fi, se ajustaron los siguientes valores: BW de 250 kHz, SF en 12 y CR de 4, ya que en este caso el robot recibe la información por LoRa, y debido a que el transceptor también posee Wi-Fi, se envía la información a la nube a través de esta red.
- En la medición y evaluación de los parámetros de QoS, principal aporte de esta investigación para sistemas de comunicaciones LPWAN, se observó que en los escenarios de mayor distancia entre transceptores y sin LOS

se presentaron mayores pérdidas de paquetes, respecto al escenario con LOS y menor distancia entre transceptores.

- En la actualidad no existe una recomendación que oriente sobre los niveles de QoS que debe tener una red LPWAN. Por tanto, ya que esta tecnología está alcanzando su grado de maduración, se sugiere que organismos como la UIT emitan una recomendación al respecto, ya que esto guiará a los usuarios sobre los niveles de calidad y los servicios que se pueden ofrecer con estas redes. Así mismo, incentivará a los fabricantes a la mejora continua de sus dispositivos.
- El uso de transceptores Wi-Fi LoRa 32, como los empleados en esta investigación, significó un ahorro sustancial en el despliegue de la infraestructura de comunicaciones, debido a que son dispositivos asequibles que, además tienen las tecnologías LoRa y Wi-Fi incorporadas, lo que permitió economizar en un 800 % el precio de adquisición de una *gateway*.

Referencias

- [1] K. Ashton, "That 'Internet of Things' Thing", *RFID J.*, vol. 22, n.º 7, pp. 97-114, 2009.
- [2] J. Gubbi, R. Buyya, S. Marusic y M. Palaniswami, "Internet of Things (IoT): A Vision, Architectural Elements, and Future Directions", *Futur. Gener. Comput. Syst.*, vol. 29, n.º 7, pp. 1645-1660, 2013. <https://doi.org/10.1016/j.future.2013.01.010>
- [3] L. D. Xu, W. He y S. Li, "Internet of Things in Industries: A Survey", *IEEE Trans. Ind. Informatics*, vol. 10, n.º 4, pp. 2233-2243, nov. 2014. <https://doi.org/10.1109/TII.2014.2300753>
- [4] B. Al-Shargabi y O. Sabri, "Internet of Things: An Exploration Study of Opportunities and Challenges", en *2017 International Conference on Engineering MIS (ICEMIS)*, may. 2017, pp. 1-4. <https://doi.org/10.1109/ICEMIS.2017.8273047>
- [5] Y. Chen, "Challenges and Opportunities of Internet of Things", en *17th Asia and South Pacific Design Automation Conference*, 2012, pp. 383-388. <https://doi.org/10.1109/ASPDAC.2012.6164978>
- [6] G. Hu, W. P. Tay y. Wen, "Cloud Robotics: Architecture, Challenges and Applications", *IEEE Netw.*, vol. 26, n.º 3, pp. 21-28, may. 2012. <https://doi.org/10.1109/MNET.2012.6201212>
- [7] P. P. Ray, "Internet of Robotic Things: Concept, Technologies, and Challenges", *IEEE Access*, vol. 4, pp. 9489-9500, 2016. <https://doi.org/10.1109/ACCESS.2017.2647747>
- [8] P. Simoens, M. Dragone y A. Saffiotti, "The Internet of Robotic Things: A Review of Concept, Added Value and Applications", *Int. J. Adv. Robot. Syst.*, vol. 15, n.º 1, p. 1729881418759424, 2018. <https://doi.org/10.1177/1729881418759424>

- [9] A. Chibani, Y. Amirat, S. Mohammed, E. Matson, N. Hagita y M. Barreto, “Ubiquitous Robotics: Recent Challenges and Future Trends”, *Robot. Auton. Syst.*, vol. 61, n.º 11, pp. 1162-1172, 2013. <https://doi.org/10.1016/j.robot.2013.04.003>
- [10] T. Zhang, B. Yuan, T. Meng, Y. Ren, H. Liu y X. Wang, “Ubiquitous Robot: A New Paradigm for Intelligence”, en *Intelligent Data Engineering and Automated Learning*, 2016, pp. 202-211. https://doi.org/10.1007/978-3-319-46257-8_22
- [11] B. Kehoe, S. Patil, P. Abbeel y K. Goldberg, “A Survey of Research on Cloud Robotics and Automation”, *IEEE Trans. Autom. Sci. Eng.*, vol. 12, n.º 2, pp. 398-409, 2015. <https://doi.org/10.1109/TASE.2014.2376492>
- [12] G. Mohanarajah, D. Hunziker, R. D’Andrea y M. Waibel, “Rapyuta: A Cloud Robotics Platform”, *IEEE Trans. Autom. Sci. Eng.*, vol. 12, n.º 2, pp. 481-493, 2015. <https://doi.org/10.1109/TASE.2014.2329556>
- [13] W. Chen, Y. Yaguchi, K. Naruse, Y. Watanobe, K. Nakamura y J. Ogawa, “A Study of Robotic Cooperation in Cloud Robotics: Architecture and Challenges”, *IEEE Access*, vol. 6, pp. 36662-36682, 2018. <https://doi.org/10.1109/ACCESS.2018.2852295>
- [14] M. Chowdhury y M. Maier, “Local and Nonlocal Human-to-Robot Task Allocation in Fiber-Wireless Multi-Robot Networks”, *IEEE Syst. J.*, vol. 12, n.º 3, pp. 2250-2260, 2018. <https://doi.org/10.1109/JSYST.2017.2661282>
- [15] X. Yu y M. A. Hsieh, “Synthesis of a Time-Varying Communication Network by Robot Teams With Information Propagation Guarantees”, *IEEE Robot. Autom. Lett.*, vol. 5, n.º 2, pp. 1413-1420, 2020. <https://doi.org/10.1109/LRA.2020.2967704>
- [16] O. Vermesan, A. Broring, E. Tragos, M. Serrano, D. Bacciu, S. Chessa, C. Gallicchio, A. Micheli, M. Dragone, A. Saffiotti, P. Simoens, F. Cavallo y R. Bahr, “Internet of Robotic Things: Converging Sensing/Actuating, Hypoconnectivity, Artificial Intelligence and IoT Platforms”, en *Cognitive Hyperconnected Digital Transformation : Internet of Things Intelligence Evolution*, 2017, pp. 97-155.
- [17] D. S. Linthicum, “Cloud Computing Changes Data Integration Forever: What’s Needed Right Now”, *IEEE Cloud Comput.*, vol. 4, n.º 3, pp. 50-53, 2017. <https://doi.org/10.1109/MCC.2017.47>

- [18] J. Wan, S. Tang, H. Yan, D. Li, S. Wang y A. V. Vasilakos, "Cloud Robotics: Current Status and Open Issues", *IEEE Access*, vol. 4, pp. 2797-2807, 2016. <https://doi.org/10.1109/ACCESS.2016.2574979>
- [19] O. Saha y P. Dasgupta, "A Comprehensive Survey of Recent Trends in Cloud Robotics Architectures and Applications", *Robotics*, vol. 7, n.º 3, 2018. <https://doi.org/10.3390/robotics7030047>
- [20] C. Razafimandimby, V. Loscri y A. M. Vegni, "Towards Efficient Deployment in Internet of Robotic Things", en *Integration, Interconnection, and Interoperability of IoT Systems*; R. Gravina, C. E. Palau, M. Manso, A. Liotta y G. Fortino, eds. Cham: Springer International Publishing, 2018, pp. 21-37.
- [21] M. Li y Y. T. Kim, "Feasibility Analysis on the Use of Ultrasonic Communications for Body Sensor Networks", *Sensors*, vol. 18, n.º 12, pp. 1-11, 2018. <https://doi.org/10.3390/s18124496>
- [22] A. M. Priester, S. Natarajan y M. O. Culjat, "Robotic Ultrasound Systems in Medicine", *IEEE Trans. Ultrason. Ferroelectr. Freq. Control*, vol. 60, n.º 3, pp. 507-523, 2013. <https://doi.org/10.1109/TUFFC.2013.2593>
- [23] M. Akbari, J. Carriere, T. Meyer, R. Sloboda, S. Husain, N. Usmani y M. Tavakoli, "Robotic Ultrasound Scanning with Real-Time Image-Based Force Adjustment: Quick Response for Enabling Physical Distancing During the COVID- 19 Pandemic", *Front. Robot. AI*, vol. 8, p. 62, 2021. <https://doi.org/10.3389/frobt.2021.645424>
- [24] J. A. Vijaya-Selvi, R. Hemalatha y T. Pasupathi, "An Inspired Underwater Communication Robotic Fish for Aquatic Animals", en *1st International Conference on Innovations in Information and Communication Technology (ICIICT)*, 2019, pp. 1-7. <https://doi.org/10.1109/ICIICT1.2019.8741356>
- [25] R. K. Katzschnmann, J. DelPreto, R. MacCurdy y D. Rus, "Exploration of Underwater Life with an Acoustically Controlled Soft Robotic Fish", *Sci. Robot.*, vol. 3, n.º 16, pp. 1-27, 2018. <https://doi.org/10.1126/sci-robotics.aar3449>
- [26] J. B. Carruthers, "Wireless Infrared Communications", en *Wiley Encyclopedia of Telecommunications*. Wiley, 2003, pp. 1-10.
- [27] F. Arvin, K. Samsudin y A. R. Ramli, "A Short-Range Infrared Communication for Swarm Mobile Robots", en *International Conference*

- on Signal Processing Systems*, 2009, pp. 454-458. <https://doi.org/10.1109/ICSPS.2009.88>
- [28] S. M. Trenkwalder, I. Esnaola, Y. Kaszubowski Lopes, A. Kolling y R. Groß, “SwarmCom: An Infra-red-based Mobile Ad-hoc Network for Severely Constrained Robots”, *Auton. Robots*, vol. 44, n.º 1, pp. 93-114, 2020. <https://doi.org/10.1007/s10514-019-09873-0>
 - [29] A. K. Majumdar, “Chapter 4–Fundamentals of Free-Space Optical Communications Systems, Optical Channels, Characterization, and Network/Access Technology”, en *Optical Wireless Communications for Broadband Global Internet Connectivity*, A. K. Majumdar, ed. Elsevier, 2019, pp. 55-116.
 - [30] J. Derenick, C. Thorne y J. Spletzer, “Hybrid Free-Space Optics/Radio Frequency (FSO/RF) Networks for Mobile Robot Teams BT–Multi-Robot Systems. From Swarms to Intelligent Automata Volume III”, 2005, pp. 263-268. https://doi.org/10.1007/1-4020-3389-3_22
 - [31] D. Ma, J. Wei y Z. Zhuang, “A Novel Optical Signal Detecting and Processing Method for Swarm Robot Vision System”, en *IEEE International Conference on Robotics, Intelligent Systems and Signal Processing*, 2003, vol. 2, pp. 1081-1085. <https://doi.org/10.1109/RISSP.2003.1285740>
 - [32] W. S. Rabinovich, J. Murphy, M. Suite, M. Ferraro, R. Mahon, P. Goetz, K. Hacker, W. Freeman, E. S. Georges, S. Uecke y J. Sender, “Free-space Optical Data Link to a Small Robot Using Modulating Retroreflectors”, en *Free-Space Laser Communications IX*, 2009, vol. 7464, pp. 65-73. <https://doi.org/10.1117/12.828869>
 - [33] Y. F. Fung, M. Dai y M. F. Ercan, “Underwater Short Range Free Space Optical Communication for a Robotic Swarm”, en *2009 4th International Conference on Autonomous Robots and Agents*, 2009, pp. 529-532. <https://doi.org/10.1109/ICARA.2000.4804030>
 - [34] J. L. Murphy, M. S. Ferraro, W. S. Rabinovich, P. G. Goetz, M. R. Suite y S. H. Uecke, “Control of a Small Robot Using a Hybrid Optical Modulating Retro- reflector/RF Link”, en *Laser Radar Technology and Applications XIX; and Atmospheric Propagation XI*, 2014, vol. 9080, p. 90801F. <https://doi.org/10.1117/12.2052942>
 - [35] Bluetooth SIG Proprietary (2016). “Bluetooth specification. Version 5.0”. https://www.bluetooth.org/DocMan/handlers/DownloadDoc.ashx?doc_id=421043

- [36] S. Zeadally, F. Siddiqui y Z. Baig, “25 Years of Bluetooth Technology”, *Futur. Internet*, vol. 11, n.º 9, pp. 1-24, 2019. <https://doi.org/10.3390/fi11090194>
- [37] S. Van Delden y A. Whigham, “A Bluetooth-based Architecture for Android Communication with an Articulated Robot”, en *International Conference on Collaboration Technologies and Systems (CTS)*, 2012, pp. 104-108. <https://doi.org/10.1109/CTS.2012.6261035>
- [38] D. Chakraborty, K. Sharma, R. K. Roy, H. Singh y T. Bezboruah, “Android Application Based Monitoring and Controlling of Movement of a Remotely Controlled Robotic Car Mounted with Various Sensors Via Bluetooth”, en *International Conference on Advances in Electrical, Electronic and Systems Engineering (ICAEES)*, 2016, pp. 170-175. <https://doi.org/10.1109/ICAEES.2016.7888032>
- [39] M. Sampaio, A. L. G. Modesto, E. G. De Almeida-Sobrinho, J. F. Almeida y O. A. Chase, “SELVABOT-1: Embedded Control System Using Bluetooth Technology”, *IEEE Lat. Am. Trans.*, vol. 12, n.º 8, pp. 1404-1409, 2014. <https://doi.org/10.1109/TLA.2014.7014507>
- [40] L. A. Szolga y G. Flueraş, “Robotic Arm Controlled by Android App Through Bluetooth Connection for Organic Solarcell Manipulation”, *{IOP} Conf. Ser. Mater. Sci. Eng.*, vol. 1032, p. 12049, 2021. <https://doi.org/10.1088/1757-899x/1032/1/012049>
- [41] K. Houđa y R. Lakel, “Synchronized Communication in a Set of Autonomous Mobile Robots Using Bluetooth Technology”, *Procedia Comput. Sci.*, vol. 73, pp. 154-161, 2015. <https://doi.org/https://doi.org/10.1016/j.procs.2015.12.061>
- [42] S. Farahani, “Chapter 3–ZigBee and IEEE 802.15.4 Protocol Layers”, en *ZigBee Wireless Networks and Transceivers*, S. Farahani, ed., Burlington: Newnes, 2008, pp. 33-135.
- [43] Zigbee Alliance (2015). “ZigBee Specification”. <https://zigbeealliance.org/wp-content/uploads/2019/11/docs-05-3474-21-0csg-zigbee-specification.pdf>
- [44] C. Tsui, D. Carnegie y Q. W. Pan, “USAR Robot Communication Using ZigBee Technology”, en *Progress in Robotics*, 2009, pp. 380-390.
- [45] A. W. Y. Ko y H. Y. K. Lau, “Intelligent Robot-Assisted Humanitarian Search and Rescue System”, *Int. J. Adv. Robot. Syst.*, vol. 6, n.º 2, p. 12, 2009. <https://doi.org/10.5772/6792>

- [46] T. Brodeur, P. Regis, D. Feil-Seifer y S. Sengupta, "Search and Rescue Operations with Mesh Networked Robots", en *9th IEEE Annual Ubiquitous Computing, Electronics Mobile Communication Conference (UEMCON)*, 2018, pp. 6-12. <https://doi.org/10.1109/UEMCON.2018.8796743>
- [47] Y. Choden, M. Raj, C. Wangchuk, P. Singye y K. Muramatsu, "Remote Controlled Rescue Robot Using ZigBee Communication", en *IEEE 5th International Conference for Convergence in Technology (I2CT)*, 2019, pp. 1-5. <https://doi.org/10.1109/I2CT45611.2019.9033924>
- [48] Q. Liu, Y. Zhang, Y. Wang, B. Li, Y. Gou, J. Xu, D. Sheng y Z. Liu, "Research on Key Technologies of Intelligent Fire Fighting Robot based on ZigBee Network", en *International Conference on Artificial Intelligence and Computer Engineering (ICAICE)*, 2020, pp. 50-53. <https://doi.org/10.1109/ICAICE51518.2020.00016>
- [49] M. T. Rashid, A. A. Ali, R. S. Ali, L. Fortuna, M. Frasca y M. G. Xibilia, "Wireless Underwater Mobile Robot System Based on ZigBee", en *International Conference on Future Communication Networks*, 2012, pp. 117-122. <https://doi.org/10.1109/ICFCN.2012.6206853>
- [50] S. Guo, X. Li y J. Guo, "Study on a Multi-Robot Cooperative Wireless Communication Control System for The Spherical Amphibious Robot", en *IEEE International Conference on Mechatronics and Automation*, 2016, pp. 1143-1148. <https://doi.org/10.1109/ICMA.2016.7558723>
- [51] J. H. Jung, S. Park y S.-L. Kim, "Multi-robot Path Finding with Wireless Multihop Communications", *IEEE Commun. Mag.*, vol. 48, n.º 7, pp. 126-132, 2010. <https://doi.org/10.1109/MCOM.2010.5496889>
- [52] Y. Zhang, X. Shen, Q. Yang, C. Qian y G. Liu, "Infusion Auxiliary Service System Based on ZigBee Wireless Network", *IEEE Access*, vol. 6, pp. 12718-12726, 2018. <https://doi.org/10.1109/ACCESS.2018.2809676>
- [53] G. Song, K. Yin, Y. Zhou y X. Cheng, "A Surveillance Robot with Hopping Capabilities for Home Security", *IEEE Trans. Consum. Electron.*, vol. 55, n.º 4, pp. 2034-2039, 2009. <https://doi.org/10.1109/TCE.2009.5373766>
- [54] A. A. Braidot, C. C. Cifuentes, A. Frizera-Neto, M. Frisoli y A. Santiago, "ZigBee Wearable Sensor Development for Upper Limb Robotics Rehabilitation", *IEEE Lat. Am. Trans.*, vol. 11, n.º 1, pp. 408-413, 2013. <https://doi.org/10.1109/TLA.2013.6502838>

- [55] A. H. Sabry, F. H. Nordin, A. H. Sabry y M. Z. Abidin Ab Kadir, "Fault Detection and Diagnosis of Industrial Robot Based on Power Consumption Modeling", *IEEE Trans. Ind. Electron.*, vol. 67, n.º 9, pp. 7929-7940, 2020. <https://doi.org/10.1109/TIE.2019.2931511>
- [56] Institute of Electrical & Electronics Engineers, "Information Technology - Telecommunications and Information Exchange Between Systems - Local and Metropolitan Area Networks - Specific Requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications - Higher - Speed", 1999. <https://ieeexplore.ieee.org/document/9363693>
- [57] Wi-Fi Alliance. <https://www.wi-fi.org/>
- [58] B. Balaguer, G. Erinc y S. Carpin, "Combining Classification and Regression for WiFi Localization of Heterogeneous Robot Teams in Unknown Environments", en *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012, pp. 3496-3503. <https://doi.org/10.1109/IROS.2012.6385748>
- [59] Y. Sun, M. Liu y M. Q.-H. Meng, "WiFi Signal Strength-Based Robot Indoor Localization", en *2014 IEEE International Conference on Information and Automation (ICIA)*, 2014, pp. 250-256. <https://doi.org/10.1109/ICInfA.2014.6932662>
- [60] T. Wang, L. Zhao, Y. Jia y J. Wang, "WiFi Initial Position Estimate Methods for Autonomous Robots", en *2018 WRC Symposium on Advanced Robotics and Automation (WRC SARA)*, 2018, pp. 165-171. <https://doi.org/10.1109/WRC-SARA.2018.8584212>
- [61] M. Sweatt, A. Ayoade, Q. Han, J. Steele, K. Al-Wahedi y H. Karki, "WiFi Based Communication and Localization of an Autonomous Mobile Robot for Refinery Inspection", en *2015 IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 4490-4495. <https://doi.org/10.1109/ICRA.2015.7139821>
- [62] Y. Gao, H. Chen, Y. Li y Y. Liu, "Autonomous WiFi-relay Control with Mobile Robots", en *2016 IEEE International Conference on Real-time Computing and Robotics (RCAR)*, 2016, pp. 198-203. <https://doi.org/10.1109/RCAR.2016.7784025>
- [63] L. Zhang, Z. Chen, W. Cui, B. Li, C. Chen, Z. Cao y K. Gao, "WiFi-Based Indoor Robot Positioning Using Deep Fuzzy Forests", *IEEE Internet Things J.*, vol. 7, n.º 11, pp. 10773-10781, 2020. <https://doi.org/10.1109/JIOT.2020.2986685>

- [64] H. Zou, C. Chen, M. Li, J. Yang, Y. Zhou, L. Xie y C. Spanos, “Adversarial Learning-Enabled Automatic WiFi Indoor Radio Map Construction and Adaptation With Mobile Robot”, *IEEE Internet Things J.*, vol. 7, n.º 8, pp. 6946-6954, 2020. <https://doi.org/10.1109/JIOT.2020.2979413>
- [65] H.-W. Lee, Z.-C. Fang y J.-Y. Chen, “A Study of WIFI Control Wheeled Robot System with Ultrasonic Obstacle Avoidance”, en *2018 IEEE International Conference on Consumer Electronics-Taiwan (ICCE-TW)*, 2018, pp. 1-2. <https://doi.org/10.1109/ICCE-China.2018.8448635>
- [66] Y. Chen, Z. Cui, H. Zhang y L. Wu, “Design and Research of Visual Pipeline Robot Based on Wi-Fi”, en *2018 IEEE 4th Information Technology and Mechatronics Engineering Conference (ITOEC)*, 2018, pp. 935-939. <https://doi.org/10.1109/ITOEC.2018.8740572>
- [67] G. Zhong, Y. Kobayashi, Y. Hoshino y T. Emaru, “Intuitive Teleoperation of Nonholonomic Mobile Robot with a Manipulator Based on Virtual Reality and WiFi”, en *IET International Conference on Information Science and Control Engineering 2012 (ICISCE 2012)*, 2012, pp. 1-5. <https://doi.org/10.1049/cp.2012.2327>
- [68] K. Shima, Y. Uo y S. Fujita, “Auto Configuration and Management Mechanism for the Robotics Self Extensible WiFi Network”, en *2008 SICE Annual Conference*, 2008, pp. 1648-1652. <https://doi.org/10.1109/SICE.2008.4654927>
- [69] P. B. Jarande, S. P. Murakar, N. S. Vast, N. P. Ubale y S. S. Saraf, “Robotic Vacuum Cleaner Using Arduino with Wifi”, en *2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT)*, 2018, pp. 1513-1517. <https://doi.org/10.1109/ICICCT.2018.8473256>
- [70] J. O. Sallent-Roig y J. Pérez-Romero, *Fundamentos de diseño y gestión de sistemas de comunicaciones móviles celulares*, Barcelona: Universitat Politècnica de Catalunya, 2014.
- [71] J. Velagic, N. Osmic, F. Hodzic y H. Siljak, “Outdoor Navigation of a Mobile Robot Using GPS and GPRS Communication System”, en *Proceedings ELMAR-2011*, 2011, pp. 173-177.
- [72] K. M. Al-Aubidy, M. M. Ali, A. M. Derbas y A. W. Al-Mutairi, “GPRS-based Remote Sensing and Teleoperation of a Mobile Robot”, en *10th International Multi- Conferences on Systems, Signals Devices*, 2013, pp. 1-7. <https://doi.org/10.1109/SSD.2013.6564166>

- [73] G. S. V. Kumar y M. Meenakshi, “Wireless Control of Autonomous Wheeled Robot through GPRS — Design and Real Time Validation”, en *6th International Conference on Emerging Trends in Engineering and Technology*, 2013, pp. 30-34. <https://doi.org/10.1109/ICETET.2013.7>
- [74] F. Zeiger, F. Kempf y K. Schilling, “UMTS One Way Delay Characterization for Mobile Robot Teleoperation”, en *ISR and ROBOTIK*, 2010, pp. 1-8.
- [75] S. Kahar, R. Sulaiman, A. S. Prabuwno, M. F. M. Amran y S. Marjudi, “Data Transferring Technique for Mobile Robot Controller Via Mobile Technology”, en *International Conference on Pattern Analysis and Intelligence Robotics*, 2011, vol. 1, pp. 103-108. <https://doi.org/10.1109/ICPAIR.2011.5976920>
- [76] F. Voigtländer, A. Ramadan, J. Eichinger, C. Lenz, D. Pensky y A. Knoll, “5G for Robotics: Ultra-Low Latency Control of Distributed Robotic Systems”, en *International Symposium on Computer Science and Intelligent Controls*, 2017, pp. 69-72. <https://doi.org/10.1109/ISCSIC.2017.27>
- [77] Y. Y. Li, W. H. Fan, Y. H. Liu y X. P. Cai, “Teleoperation of Robots Via the Mobile Communication Networks”, en *IEEE International Conference on Robotics and Biomimetics*, 2005, pp. 670-675. <https://doi.org/10.1109/ROBIO.2005.246348>
- [78] S. Park, Y. Jung y J. Bae, “InterActive and intuitiVe control interfAce for a Tele- operAted Robot (AVATAR) System”, en *IEEE International Conference on Advanced Intelligent Mechatronics*, 2017, pp. 241-246. <https://doi.org/10.1109/AIM.2017.8014024>
- [79] K. Yaovaja y J. Klunngien, “Teleoperation of an Industrial Robot Using a Non- Standalone 5G Mobile Network”, en *IEEE Eurasia Conference on IOT, Communication and Engineering*, 2019, pp. 320-323. <https://doi.org/10.1109/ECICE47484.2019.8942700>
- [80] A. S. A. Hussaini y M. Sundarambal, “Wireless Control of Humanoid Robot Using 3G”, en *International Conference on Process Automation, Control and Computing*, 2011, pp. 1-6. <https://doi.org/10.1109/PACC.2011.5978956>
- [81] B. Jincun, L. Qi y L. Yanfei, “The Design of the Rescue Robot Long-Distance Control Based on 3G and GPS”, en *International Conference on Intelligent Human- Machine Systems and Cybernetics*, 2009, vol. 2, pp. 170-172. <https://doi.org/10.1109/IHMSC.2009.166>

- [82] S. Mishra, “Robotics Design for Detection of Explosive With 3G Communication Technology”, en *INTERACT*, 2010, pp. 201-205. <https://doi.org/10.1109/INTERACT.2010.5706226>
- [83] A. Fahim y Y. Gadallah, “An Optimized LTE-Based Technique for Drone Base Station Dynamic 3D Placement and Resource Allocation in Delay-Sensitive M2M Networks”, *IEEE Trans. Mob. Comput.*, p. 1, 2021. <https://doi.org/10.1109/TMC.2021.3089329>
- [84] K. Ismail, R. Liu, J. Zheng, C. Yuen, Y. L. Guan y U.-X. Tan, “Mobile Robot Localization Based on Low-Cost LTE and Odometry in GPS-Denied Outdoor Environment”, en *IEEE International Conference on Robotics and Biomimetics*, 2019, pp. 2338-2343. <https://doi.org/10.1109/ROBIO49542.2019.8961750>
- [85] M. Aleksy, F. Dai, N. Enayati, P. Rost y G. Pocovi, “Utilizing 5G in Industrial Robotic Applications”, en *International Conference on Future Internet of Things and Cloud*, 2019, pp. 278-284. <https://doi.org/10.1109/FiCloud.2019.00046>
- [86] S. Avgousti, A. Panayides, A. Jossif, E. Christoforou, P. Vieyres, C. Novales, S. Voskarides y C. Pattichis, “Cardiac Ultrasonography over 4G Wireless Networks Using a Tele-operated Robot”, *Healthc. Technol. Lett.*, vol. 3, n.º 3, pp. 212-217, sep. 2016. <https://doi.org/10.1049/htl.2016.0043>
- [87] D. A. Meshram y D. D. Patil, “5G Enabled Tactile Internet for Tele-Robotic Surgery”, *Procedia Comput. Sci.*, vol. 171, pp. 2618-2625, 2020. <https://doi.org/https://doi.org/10.1016/j.procs.2020.04.284>
- [88] B. S. Chaudhari, M. Zennaro y S. Borkar, “LPWAN Technologies: Emerging Application Characteristics, Requirements, and Design Considerations”, *Futur. Internet*, vol. 12, n.º 3, pp. 1-25, 2020. <https://doi.org/10.3390/fi12030046>
- [89] R. D. Lestari, A. Rusdinar, M. A. Murti, G. Tawaqal y D. Lee, “Design of IoT- Based River Water Monitoring Robot Data Transmission Model Using Low Power Wide Area Network (LPWAN) Communication Technology”, en *IEEE International Conference on Internet of Things and Intelligence System*, 2019, pp. 201-205. <https://doi.org/10.1109/IoTaIS47347.2019.8980377>
- [90] O. Bhatnagar, N. Surendran y M. M. Alam, “Development of an Algorithm for a Target Platform-following Robot Using LoRa Signals”, en *11th International Conference on Computing, Communication*

- and Networking Technologies*, 2020, pp. 1-5. <https://doi.org/10.1109/ICCNT49239.2020.9225679>
- [91] T. Maneekittichote y T. Chanthasopeephan, "Mobile Robot Swarm Navigation and Communication Using LoRaWan", en *6th International Conference on Mechatronics and Robotics Engineering*, 2020, pp. 22-25. <https://doi.org/10.1109/ICMRE49073.2020.9064973>
 - [92] Y. Song, J. Lin, M. Tang y S. Dong, "An Internet of Energy Things Based on Wireless LPWAN", *Engineering*, vol. 3, n.º 4, pp. 460-466, 2017. <https://doi.org/https://doi.org/10.1016/J.ENG.2017.04.011>
 - [93] S. J. Danbatta y A. Varol, "Comparison of Zigbee, Z-Wave, Wi-Fi, and Bluetooth Wireless Technologies Used in Home Automation", en *7th International Symposium on Digital Forensics and Security*, 2019, pp. 1-5. <https://doi.org/10.1109/ISDFS.2019.8757472>
 - [94] V. Sugumaran, *Developments and Trends in Intelligent Technologies and Smart Systems*, IGI Global, 2017.
 - [95] J. González, *Robótica modular y locomoción: aplicaciones a robots ápodas*, Madrid: Universidad Autónoma de Madrid, Departamento de Ingeniería Informática, 2008.
 - [96] A. Brunete, E. Gambao, J. E. Torres y M. Hernando, "A 2 DoF Servomotor-based Module for Pipe Inspection Modular Micro-robots", *IEEE Int. Conf. Intell. Robot. Syst.*, pp. 1329-1334, 2006. <https://doi.org/10.1109/IROS.2006.281898>
 - [97] L. Felipe, L. F. Giraldo y E. Delgado, "Cinemática inversa de un brazo robot utilizando algoritmos genéticos", *Av. en Sist. e Informática*, vol. 3, n.º 1, pp. 29-34, 2006.
 - [98] J. Baca, S. G. M. Hossain, P. Dasgupta, C. A. Nelson y A. Dutta, "ModRED: Hardware Design and Reconfiguration Planning for a High Dexterity Modular Self-reconfigurable Robot for Extra-terrestrial Exploration", *Rob. Auton. Syst.*, vol. 62, n.º 7, pp. 1002-1015, 2014. <https://doi.org/10.1016/j.robot.2013.08.008>
 - [99] A. Lyder, R. F. M. García y K. Stoy, "Mechanical Design of Odin, an Extendable Heterogeneous Deformable Modular Robot", en *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS*, 2008, pp. 883-888. <https://doi.org/10.1109/IROS.2008.4650888>

- [100] J. W. Romanishin, K. Gilpin y D. Rus, “M-blocks: Momentum-driven, Magnetic Modular Robots”, *IEEE Int. Conf. Intell. Robot. Syst.*, pp. 4288-4295, 2013. <https://doi.org/10.1109/IROS.2013.6696971>
- [101] D. Brandt, D. J. Christensen y H. H. Lund, “ATRON Robots: Versatility from Self- Reconfigurable Modules”, *Proc. 2007 IEEE Int. Conf. Mechatronics Autom. ICMA 2007*, pp. 26-32, 2007. <https://doi.org/10.1109/ICMA.2007.4303511>
- [102] J. Neubert y H. Lipson, “Soldercubes: A Self-soldering Self-reconfiguring Modular Robot System”, *Auton. Robots*, vol. 40, n.º 1, pp. 139-158, 2016. <https://doi.org/10.1007/s10514-015-9441-4>
- [103] M. Yim, D. G. Duff y K. D. Roufas, “PolyBot: A Modular Reconfigurable Robot”, en *Proceedings-IEEE International Conference on Robotics and Automation*, 2000, pp. 514-520. <https://doi.org/10.1109/robot.2000.844106>
- [104] J. W. Suh, S. B. Homans y M. Yim, “Telecubes: Mechanical Design of a Module for Self-reconfigurable Robotics”, en *Proceedings-IEEE International Conference on Robotics and Automation*, 2002, pp. 4095-4101. <https://doi.org/10.1109/ROBOT.2002.1014385>
- [105] V. Zykov, P. Williams, N. Lassabe y H. Lipson, “Molecubes Extended: Diversifying Capabilities of Open-Source Modular Robotics”, en *IROS-2008 Self- Reconfigurable Robotics Workshop*, 2008, pp. 22-26.
- [106] A. Castano, A. Behar y P. M. Will, “The Conro Modules for Reconfigurable Robots”, *IEEE/ASME Trans. Mechatronics*, vol. 7, n.º 4, pp. 403-409, 2002. <https://doi.org/10.1109/TMECH.2002.806233>
- [107] B. Salemi, M. Moll y W. M. Shen, “SUPERBOT: A Deployable, Multi-functional, and Modular Self-reconfigurable Robotic System”, en *IEEE International Conference on Intelligent Robots and Systems*, 2006, pp. 3636-3641. <https://doi.org/10.1109/IROS.2006.281719>
- [108] R. Moeckel, C. Jaquier, K. Drapel, E. Dittrich, A. Upegui y A. Ijspeert, “YaMoR and Bluemove — An Autonomous Modular Robot with Bluetooth Interface for Exploring Adaptive Locomotion BT-Climbing and Walking Robots”. Springer, 2006, pp. 685-692.
- [109] K. Gilpin, K. Kotay y D. Rus, “Miche: Modular Shape Formation by Self- dissassembly”, en *Proceedings-IEEE International Conference on Robotics and Automation*, 2007, pp. 2241-2247. <https://doi.org/10.1109/ROBOT.2007.363653>

- [110] C. H. S. S. Reddy, S. Patlolla, A. Agrawal y K. R. Anupama, "HexaMob-A Hybrid Modular Robotic Design for Implementing Biomimetic Structures", *Robotics*, vol. 6, n.º 4, pp. 1-18, 2017. <https://doi.org/10.3390/robotics6040027>
- [111] L. Roberto, *Estudio de las firmas neuronales en los generadores centrales de patrones*, Madrid: Universidad Autónoma de Madrid, Departamento de Ingeniería Informática, 2004.
- [112] A. C. Orgerie y L. Lefèvre, "When Clouds Become Green: The Green Open Cloud Architecture", *Adv. Parallel Comput.*, vol. 19, pp. 228-237, 2010. <https://doi.org/10.3233/978-1-60750-530-3-228>
- [113] H. Martínez, *Estimación de una serie de movimientos utilizando un algoritmo de optimización bio-inspirado para la operación de manera autónoma y on-line de una plataforma multi-robot (caso robot modular)*, Bogotá: Universidad Nacional de Colombia, Departamento de Ingeniería de Sistemas e Industrial, 2018.
- [114] R. Moreno, *Diseño y simulación de un algoritmo para el control de un robot modular tipo cadena*, Bogotá: Universidad Nacional de Colombia, Departamento de Ingeniería Eléctrica y Electrónica, 2010.
- [115] M. Yim, "Locomotion with a Unit-modular Reconfigurable Robot", *Proc JSME Int Conf Adv. Mechatronics*, pp. 283-288, 1994.
- [116] E. Calle, I. Ávila y J. Zambrano, "Diseño e implementación de un robot móvil cuadrúpedo", *Rev. Tecnológica-ESPOL*, vol. 20, n.º 1, pp. 65-72, 2007. <https://doi.org/0257-1749>
- [117] D. Zamorano, *Control locomotor multidireccional de un robot modular mediante circuitos generadores centrales de patrones bioinspirados*, Madrid: Universidad Autónoma de Madrid, Departamento de Ingeniería Informática, 2011.
- [118] H. Espitia y J. Sofrony, "Algoritmo para planear trayectorias de robots móviles, empleando campos potenciales y enjambres de partículas activas brownianas", *Cienc. e Ing. Neogranadina*, vol. 22, n.º 12, pp. 75-96, 2012.
- [119] N. Naik, "Choice of Effective Messaging Protocols for IoT Systems: MQTT, CoAP, AMQP and HTTP", *IEEE Int. Symp. Syst. Eng.*, pp. 1-7, 2017. <https://doi.org/10.1109/SysEng.2017.8088251>

- [120] W. Stallings, *Foundations of Modern Networking: SDN, NFV, QoE, IoT, and Cloud*, Pearson, 2016.
- [121] M. Centenaro, L. Vangelista, A. Zanella y M. Zorzi, “Long-range Communications in Unlicensed Bands: The Rising Stars in the IoT and Smart City scenarios”, *IEEE Wirel. Commun.*, vol. 23, n.º 5, pp. 60-67, 2016. <https://doi.org/10.1109/MWC.2016.7721743>
- [122] A. Montes y L. Camacho, *Telemetría a través de redes de área extensa de baja potencia (LPWA) y en coexistencia con el internet de las cosas (IoT)*, Bogotá: Universidad Distrital Francisco José de Caldas, Especialización en Telecomunicaciones Móviles, 2019.
- [123] K. Mekki, E. Bajic, F. Chaxel y F. Meyer, “A Comparative Study of LPWAN Technologies for Large-scale IoT Deployment”, *ICT Express*, vol. 5, n.º 1, pp. 1-7, 2019. <https://doi.org/10.1016/j.icte.2017.12.005>
- [124] W. Ayoub, M. Mroue, F. Nouvel, A. E. Samhat y J. C. Prevotet, “Towards IP over LPWANs Technologies: LoRaWAN, DASH7, NB-IoT”, *6th Int. Conf. Digit. Information, Networking, Wirel. Commun.*, pp. 43-47, 2018. <https://doi.org/10.1109/DINWC.2018.8356993>
- [125] N. Naik, “LPWAN Technologies for IoT Systems: Choice between Ultra Narrow Band And Spread Spectrum”, en *4th IEEE International Symposium on Systems Engineering*, 2018, pp. 1-8. <https://doi.org/10.1109/SysEng.2018.8544414>
- [126] GSM Association, “Internet of Things (NB-IoT)”. <https://www.gsma.com/iot/mobile-iot-technology-nb-iot/>
- [127] R. S. Sinha, Y. Wei y S. H. Hwang, “A Survey on LPWA Technology: LoRa and NB-IoT”, *ICT Express*, vol. 3, n.º 1, pp. 14-21, 2017. <https://doi.org/10.1016/j.icte.2017.03.004>
- [128] Technical Specification (2016). “3GPP TS 36.104 V13.3.0”. https://www.etsi.org/deliver/etsi_ts/136100_136199/136104/13.03.00_60/ts_136104v130300p.pdf
- [129] H. Paz, *Sistemas de comunicaciones digitales*, Bogotá: Escuela Colombiana de Ingeniería, 2009.
- [130] M. Kanj, V. Savaux y M. Le Guen, “A Tutorial on NB-IoT Physical Layer Design”, *IEEE Commun. Surv. Tutorials*, vol. 22, n.º 4, pp. 2408-2446, 2020. <https://doi.org/10.1109/COMST.2020.3022751>

- [131] M. Lauridsen, H. Nguyen, B. Vejlgaard, I. Z. Kovacs, P. Mogensen y M. Sorensen, "Coverage Comparison of GPRS, NB-IoT, LoRa, and SigFox in a 7800 km² Area", en *IEEE 85th Vehicular Technology Conference*, 2017, pp. 1-5. <https://doi.org/10.1109/VTCSpring.2017.8108182>
- [132] L. Alliance, "LoRaWAN What is it? A Technical Overview of LoRa and LoRaWAN", 2015. <https://www.tuv.com/content-media-files/master-content/services/products/1555-tuv-rheinland-lora-alliance-certification/tuv-rheinland-lora-alliance-certification-overview-lora-and-lorawan-en.pdf>
- [133] O. Yassine, M. Baslam y M. Oukessou, "LPWAN IEEE 802.11ah and LoRaWAN Capacity Simulation Analysis Comparison Using NS-3", en *International Conference on Optimization and Applications*, 2018, pp. 1-4. <https://doi.org/10.1109/ICOA.2018.8370592>
- [134] A. Yegin, T. Kramp, P. Dufour, R. Gupta, R. Soss, O. Hersent, D. Hunt y N. Sornin, "LoRaWAN Protocol: Specifications, Security, and Capabilities", en *LPWAN Technologies for IoT and M2M Applications*, Academic Press, 2020, pp. 37-63.
- [135] T. Nguyen Gia, J. P. Queralt y T. Westerlund, "Exploiting LoRa, Edge, and Fog Computing for Traffic Monitoring in Smart Cities", en *LPWAN Technologies for IoT and M2M Applications*, Academic Press, 2020, pp. 347-371.
- [136] Telensa, "Ultra Narrow Band (UNB) Smart City Network", 2019. https://info.telensa.com/hubfs/Resources%20page%20files/datasheet_telensa_planet_network.pdf
- [137] Telensa, "Telecell : Universal Streetlight Controller", 2019. https://www.essexhighways.org/uploads/docs/datasheet_telensa_planet_telecell.pdf
- [138] Telensa, "Urban IQ for smart cities", 2019. <https://info.telensa.com/data-sheet-urban-iq>
- [139] L. F. Pedraza, C. A. Hernández, K. Galeano, E. Rodríguez e I. Páez, *Ocupación espectral y modelo de radio cognitiva para Bogotá*, Bogotá: Universidad Distrital Francisco José de Caldas, 2016.
- [140] C. Fourtet y B. Ponsard, "An Introduction to Sigfox Radio System", en *LPWAN Technologies for IoT and M2M Applications*, Academic Press, 2020, pp. 103-118.

- [141] Sigfox, “Coverage”. <https://www.sigfox.com/en/coverage>
- [142] GSMA, “Mobile Internet of Things Low Power Wide Area Connectivity”, 2016. <https://www.gsma.com/iot/wp-content/uploads/2016/03/Mobile-IoT-Low-Power-Wide-Area-Connectivity-GSMA-Industry-Paper.pdf>
- [143] A. Belghith y M. S. Obaidat, “Chapter 2–Wireless Sensor Networks Applications To Smart Homes and Cities” en *Smart Cities and Homes*, M. S. Obaidat, P. B. T.-S. C. y H. Nicopolitidis, eds., Boston: Morgan Kaufmann, 2016, pp. 17-40.
- [144] Y. K. Tolcha, H. Nguyen, J. Byun, K. Kwon, J. Han, W. Yoon, N. Lee, H. Kim, N. Pham y D. Kim, “Oliot-OpenCity: Open Standard Interoperable Smart City Platform”, en *IEEE International Smart Cities Conference*, 2018, pp. 1-8. <https://doi.org/10.1109/ISC2.2018.8656763>
- [145] W. Yoon, Y. Lee, H. Chae, S. Seo, S. Heo, N. Lee, K. Kwon y D. Kim, “HERMES: GS1-based Smart City Service Intercommunity”, en *IEEE International Smart Cities Conference*, 2018, pp. 1-8. <https://doi.org/10.1109/ISC2.2018.8656925>
- [146] G. Ceballos y V. Larios, “A Model to Promote Citizen Driven Government in a Smart City: Use Case at GDL Smart City”, en *IEEE 2nd International Smart Cities Conference: Improving the Citizens Quality of Life*, 2016, pp. 1-6. <https://doi.org/10.1109/ISC2.2016.7580873>
- [147] Telensa, “Telensa Deploys Smart City Wireless Control System for 55,000 LED Streetlights in Gloucestershire UK–Telensa”, 2015. <https://www.telensa.com/news/telensa-deploys-smart-city-wireless-control-system-for-55000-led-streetlights-in-gloucestershire-uk/>
- [148] R. Fujdiak, K. Mikhaylov, M. Stusek, P. Masek, I. Ahmad, L. Malina, P. Porambage, M. Voznak, A. Pouttu y P. Mlynek, “Security in Low-power wide-area Networks: State-of-the-art and Development Toward the 5G”, en *LPWAN Technologies for IoT and M2M Applications*, Academic Press, 2020, pp. 373-396.
- [149] J. Jose y T. Sasipraba, “Indoor Air Quality Monitors Using IOT Sensors and LPWAN”, en *International Conference on Trends in Electronics and Informatics*, 2019, pp. 633-637. <https://doi.org/10.1109/icoei.2019.8862647>
- [150] R. Firdaus, M. A. Murti y I. Alinursafa, “Air Quality Monitoring System Based Internet of Things (IoT) Using LPWAN LoRa”, en *IEEE*

- International Conference on Internet of Things and Intelligence System*, 2019, pp. 195-200. <https://doi.org/10.1109/IoTaIS47347.2019.8980437>
- [151] Y. C. Chang, T. W. Huang y N. F. Huang, “A Machine Learning Based Smart Irrigation System with LoRa P2P Networks”, en *20th Asia-Pacific Network Operations and Management Symposium: Management in a Cyber-Physical World*, 2019, pp. 1-4. <https://doi.org/10.23919/APNOMS.2019.8893034>
- [152] J. D. Borrero y A. Zabalo, “An Autonomous Wireless Device for Real-time Monitoring of Water Needs”, *Sensors*, vol. 20, n.º 7, pp. 1-16, 2020. <https://doi.org/10.3390/s20072078>
- [153] T. Janssen, M. Aernouts, R. Berkvens y M. Weyn, “Outdoor Fingerprinting Localization Using Sigfox”, en *9th International Conference on Indoor Positioning and Indoor Navigation*, 2018, pp. 1-6. <https://doi.org/10.1109/IPIN.2018.8533826>
- [154] Gobierno de España, “IoT: el Internet de las cosas en las empresas cordobesas”. <https://sites.google.com/ieslafuentsanta.es/iot/proceso/actividades-del-proyecto>
- [155] B. Vogel, Y. Dong, B. Emruli, P. Davidsson y R. Spalazzese, “What is an Open IoT Platform? Insights from a Systematic Mapping Study”, *Futur. Internet*, vol. 12, n.º 4, pp. 1-19, 2020. <https://doi.org/10.3390/fi12040073>
- [156] UIT, “Definiciones de términos relativos a la calidad de servicio”, *Recomendación UIT-T E.800*, 2008.
- [157] UIT, “Quality of Service Regulation Manual”, Regulatory and Market Environment, 2017.
- [158] ANE, “Resolución 105”, Colombia, 2020. <https://www.ane.gov.co/SitePages/ListaArchivosNormatividad.aspx?p=10>
- [159] A. Faina, R. Moreno, C. Liu, H. Hernández y F. Veenstra, “Easy Modular Embodied Robot Generator”. <https://sites.google.com/view/emergemodular/home?authuser=0>
- [160] C. Liu, J. Liu, R. Moreno, F. Veenstra y A. Faina, “The Impact of Module Morphologies on Modular Robots”, en *18th International Conference on Advanced Robotics*, 2017, pp. 237-243. <https://doi.org/10.1109/ICAR.2017.8023524>

- [161] R. Moreno, C. Liu, A. Faina, H. Hernández y J. Gómez, “The EMeRGE Modular Robot, an Open Platform for Quick Testing of Evolved Robot Morphologies”, en *Genetic and Evolutionary Computation Conference*, 2017, pp. 71-72. <https://doi.org/10.1145/3067695.3075616>
- [162] H. Hernández, “Modular Robot”. https://github.com/HenryHdez/MODULAR_ROBOT
- [163] H. Hernández, “Autonomous Operation of a Modular Robot”. www.youtube.com/watch?v=mN7fpHun60o
- [164] L. F. Pedraza, H. A. Hernández y C. A. Hernández, “Artificial Neural Network Controller for a Modular Robot Using a Software Defined Radio Communication System”, *Electronics*, vol. 9, n.º 10, pp. 1-14, 2020. <https://doi.org/10.3390/electronics9101626>
- [165] H. Hernández, R. Moreno, A. Faina y J. Gómez, “Design of a Bio-inspired Controller to Operate a Modular Robot Autonomously”, en *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2018, vol. 11238, pp. 314-325. https://doi.org/10.1007/978-3-030-03928-8_26
- [166] Semtech, “LoRa Modulation Basics”, 2015. <https://www.fragalprototype.com/wp-content/uploads/2016/08/an1200.22.pdf>
- [167] HELTEC, “WiFi LoRa 32 (V2)”. <https://heltec.org/project/wifi-lora-32/>
- [168] O. F. López y A. A. Andrade, *Desarrollo de un sistema de comunicación IoT para un robot modular*, Bogotá: Universidad Distrital Francisco José de Caldas, Ingeniería en Telecomunicaciones, 2020.
- [169] J. Puentes y C. Rodríguez, *Sistema de monitoreo para un esquema de basuras con base en una tecnología IoT*, Bogotá: Universidad Distrital Francisco José de Caldas, Ingeniería en Telecomunicaciones, 2019.

Sobre los autores

Luis Fernando Pedraza Martínez

Ingeniero electrónico, magíster en Ciencias de la Información y las Comunicaciones de la Universidad Distrital Francisco José de Caldas y doctor en Ingeniería de la Universidad Nacional de Colombia. Profesor titular de la Universidad Distrital Francisco José de Caldas, adscrito al programa de Ingeniería en Telecomunicaciones de la Facultad Tecnológica; investigador sénior del Ministerio de Ciencia, Tecnología e Innovación (MinCiencias) y director del grupo de investigación GIDENUTAS. Sus áreas de interés son las redes inalámbricas, las tecnologías de internet de las cosas, la gestión del espectro radioeléctrico, la ciencia de datos y el *machine learning*. Posee varias publicaciones en el área de las telecomunicaciones en revistas nacionales e internacionales. Correo electrónico: lfpedrazam@udistrital.edu.co

Henry Alberto Hernández Martínez

Ingeniero en control y estudiante del Doctorado en Ingeniería de la Universidad Distrital Francisco José de Caldas, con maestría en Telecomunicaciones de la Universidad Nacional de Colombia. Profesor asociado de tiempo completo ocasional en la Universidad Distrital Francisco José de Caldas e investigador asociado de la Corporación de Investigación Agropecuaria (Agrosavia), en el área de automatización de procesos y maquinaria para uso agrícola. Integrante de los grupos de investigación GIDENUTAS y DIGITI en temas relacionados con sistemas de múltiples robots y sus aplicaciones mediante herramientas de IoRT (internet de las cosas robóticas), aprendizaje automático y análisis de datos. Correo electrónico: hahernandezm@udistrital.edu.co

Ernesto Cadena Muñoz

Ingeniero en telecomunicaciones bilingüe, especialista en Gestión de Proyectos de Ingeniería y en Interventoría de Proyectos, magíster en Telecomunicaciones y doctor en Ingeniería de Sistemas y Computación de la Universidad Nacional de Colombia, con conocimientos y formación integral en gestión de proyectos de telecomunicaciones y sistemas. Ha participado en investigaciones y publicaciones sobre la gestión del espectro de telecomunicaciones en Colombia, detección del ataque de emulación primaria en redes móviles de radio cognitiva, implementación de redes internas de telecomunicaciones bajo el nuevo reglamento técnico colombiano (RITEL), transmisión de contenidos en televisión digital para personas con limitación auditiva e interconexión e interoperabilidad de ANKLA-CEET. Actualmente es investigador del grupo GIDENUTAS y docente titular de vinculación especial de la Universidad Distrital Francisco José de Caldas. Correo electrónico: ecadenam@udistrital.edu.co

Este libro se
terminó de imprimir
en septiembre de 2023
en la Editorial UD,
Bogotá, D. C.,
Colombia.