

# Evaluación de protocolos de redes inalámbricas de computadoras para aplicaciones del Internet de las Cosas

Alejandra Flores Buruel, Arnoldo Díaz-Ramírez, Jesús E. Miranda-Vega,  
Verónica Quintero Rosas, Claudia Martínez Castillo, Jaime Olvera González

Departamento de Sistemas y Computación, Tecnológico Nacional de México/ Instituto Tecnológico de Mexicali, Av. Tecnológico s/n, Col. Elías Calles, Mexicali BC, 21376

## Resumen

El paradigma del Internet de las Cosas (IoT, por sus siglas en inglés) hace posible la monitorización de entornos, personas y cosas de manera ubicua y eficiente. Gracias a los avances tecnológicos en hardware y software, así como la madurez de nuevos modelos y aplicaciones de las ciencias computacionales, como la Computación en la Nube, la Inteligencia Artificial, los Sistemas Ciber-Físicos y la Industria 4.0, las posibilidades de aplicación del IoT han crecido exponencialmente.

Las tecnologías de comunicación inalámbrica son un componente fundamental de las aplicaciones IoT. Gracias a ellas es posible enviar a servidores en el borde (*Edge Computing*) a la Nube, las enormes cantidades de datos que son recolectados por los sensores de las aplicaciones IoT, para ser procesados, tomar las decisiones pertinentes y ajustar el comportamiento del sistema. Existen una buena cantidad de tecnologías de comunicación inalámbricas para ser utilizadas en aplicaciones IoT, por lo que elegir la más conveniente puede ser una tarea complicada y que consume mucho tiempo.

En este artículo se presentan los resultados de una evaluación cualitativa de algunas de las más importantes tecnologías de redes inalámbricas; a saber, WiFi, Bluetooth, ZigBee, LoRa y 6LoWPAN. Se presentan brevemente las principales características de estas tecnologías, su disponibilidad de dispositivos y software y detalles de su implementación. Adicionalmente, se presentan y discuten los resultados de diversos experimentos en los que evaluaron los parámetros de desempeño Jitter y Tasa de Paquetes Perdidos, tanto en escenarios sin y con interferencia, con distancias entre nodos de 2.5, 5, 10, 20 y 3° metros. Los resultados obtenidos permitirán a los diseñadores de aplicaciones IoT elegir la tecnología de comunicación inalámbrica más conveniente, con base en las necesidades específicas de su aplicación.

## Abstract

The Internet of Things (IoT) paradigm makes it possible to monitor environments, people, and things ubiquitously and efficiently. Thanks to technological advances in hardware and software, as well as the maturity of new models and applications of computational sciences, such as Cloud Computing, Artificial Intelligence, Cyber-Physical Systems, and Industry 4.0, the possibilities of application of the IoT, have grown exponentially.

Wireless communication technologies are a fundamental component of IoT applications. Thanks to them, it is possible to send to servers on edge (*Edge Computing*) to the Cloud the enormous amounts of data collected by the IoT applications' sensors to be processed, make the appropriate decisions, and adjust the system's behavior. Many wireless communication technologies are used in IoT applications, so choosing the most suitable one can be complicated and time-consuming.

This article presents the results of a qualitative and quantitative evaluation of some of the essential wireless network technologies, namely WiFi, Bluetooth, ZigBee, LoRa, and 6LoWPAN. The main characteristics of these technologies, their availability of devices and software, and details of their implementation are briefly presented. Additionally, the results of various experiments in which the performance parameters Jitter and Lost Packet Rate were evaluated, both in scenarios without and with interference, with distances between nodes of 2.5, 5, 10, 20, and 3 meters, are presented and discussed. The results will allow IoT application designers to choose the most convenient wireless communication technology based on the specific needs of their application.

**Palabras Clave:** Internet de las cosas, Computación en la nube, WiFi, Bluetooth, ZigBee, LoRa y 6LoWPAN

**Keywords:** Internet of things, Cloud Computing, WiFi, Bluetooth, ZigBee, LoRa y 6LoWPAN

## 1. INTRODUCCIÓN

La Industria 4.0 (I40) pretende, a través de tecnologías como sistemas ciber-físicos (CPS – *cyber- physical systems*) y el Internet de las cosas (IoT - *Internet of Things*), crear una producción inteligente, descentralizada y autónoma con sistemas trabajando en tiempo real, con la finalidad de brindar al cliente un servicio lo más personalizado posible [5].

Las aplicaciones IoT cumplen con la tarea de la recopilación y evaluación integral de datos de diferentes fuentes (equipos y sistemas de producción, así como sistemas de gestión), en el contexto de la I40, se convertirán en estándares para soportar la toma de decisiones en tiempo real [8]. Por lo tanto, definir los estándares y protocolos de comunicación más eficientes, se vuelve algo fundamental al momento de querer construir un sistema con estas características. Sin embargo, no existen muchos estudios sobre su desempeño, soporte, facilidad de instalación o el soporte de bibliotecas para el desarrollo de aplicaciones, entre otros parámetros [6].

El desarrollo de este trabajo busca presentar el resultado de un estudio cualitativo y experimental de algunos de los protocolos de red de computadoras más importantes de la actualidad que pueden ser utilizados para el desarrollo de aplicaciones IoT. Los protocolos evaluados son WiFi, Bluetooth, ZigBee y LoRa. El resto del documento está organizado de la siguiente manera. En la sección 2 se presentan brevemente las principales características de los protocolos evaluados, así como de otros que pueden ser utilizados para aplicaciones IoT y de la I40. En la Sección 3 se discuten los resultados de las evaluaciones cualitativas de los experimentos, mientras que en la sección 4 los resultados de las evaluaciones cuantitativas. Finalmente, en la sección 5 se presentan las conclusiones y trabajo futuro.

## 2. PROTOCOLOS DE REDES PARA IOT

En esta sección se presentan brevemente algunas de las principales características de los protocolos WiFi, Bluetooth, ZigBee y LoRa, que fueron utilizados en las evaluaciones de desempeño. Adicionalmente, se presentan brevemente las características de los protocolos 6LoWPAN y WirelessHART, que también pueden ser utilizados para aplicaciones IoT e I40.

### 2.1 Wi-Fi

Según el artículo "A Review of Wireless Technology Usage for Mobile Robot Controller" (publicado en ICSEM 2012), Wi-Fi o WLAN (redes de área local inalámbricas) es una familia de redes inalámbrica con base en una serie de especificaciones del Instituto de Ingenieros Eléctricos y Electrónicos (IEEE) denominada 802.11. Wi-Fi utiliza radiofrecuencia sin licencia, principalmente en la banda de 2,4 GHz. El Wi-Fi estaba destinado a ser utilizado para dispositivos móviles y redes de área local, pero ahora se utiliza a menudo para el acceso a Internet en exteriores [4]. Esta tecnología de red inalámbrica se creó con el objetivo de extender las prestaciones de las redes cableada Ethernet, pero con la ventaja de no requerir cables. La alianza WiFi adoptó el protocolo IEEE 802.11 y comenzó a comercializarlo bajo el nombre de WiFi. La versión 1 (802.11 b) de WiFi ofrecía un ancho de banda de hasta 11 Mbps, mientras que la versión WiFi 6 (802.11 ax) puede ofrecer hasta 9,600 Mbps. Se espera que la versión 7 de WiFi (802.11 be) obtenga un ancho de banda de hasta 40,000 Mbps. WiFi utiliza el protocolo CSMA/CA para acceder al medio de comunicación inalámbrica. Una de las principales características de este protocolo de acceso al medio es que utiliza contención; esto es, los nodos que desean transmitir deben "competir" con el resto de los nodos para acceder al canal. Para minimizar este efecto, WiFi ha introducido prestaciones de Calidad de Servicio (QoS – *Quality of Service*) para permitir que aplicaciones de mayor prioridad tengan preferencia en el acceso al medio.

## 2.2 Bluetooth

Bluetooth es una norma industrial enfocada a la implementación de redes inalámbricas de área personal y permite la transmisión de voz y otros datos entre diferentes tipos de dispositivos mediante un enlace por radiofrecuencia en la banda ISM de los 2,4 GHz. [1]. Ejemplos de redes de área personal son las implementadas a través del uso de audífonos, teléfonos y relojes inteligentes, ratones (*mouse*) y teclados de computadoras, tabletas, entre otros. Sin embargo, el bajo costo y consumo de energía de esta tecnología la ha hecho atractiva para aplicaciones del IoT. A diferencia de WiFi, los dispositivos Bluetooth pueden operar con baterías, lo que representa un atractivo adicional para aplicaciones IoT.

Para que los nodos Bluetooth transmitan información a través del medio inalámbrico, se utiliza la técnica denominada FHSS (*Frequency Hopping Spread Spectrum*), que divide el espectro en 79 canales de transmisión. Al enviar información, se utiliza alguno de estos canales para transmitir un número limitado de paquetes de información, para posteriormente continuar la transmisión usando otro canal, siguiendo con el proceso de cambio de canales de manera aleatoria. Con esta estrategia se busca reducir la probabilidad de colisiones. La versión 5 de Bluetooth es capaz de transmitir a una tasa de 50 Mbps.

Para acceder al medio, ZigBee utiliza el protocolo CSMA/CA, aunque puede ser configurado para utilizar un protocolo sin colisión, en el que se le asigna a cada nodo una ventana de tiempo para transmitir. ZigBee es un protocolo de bajo consumo de energía, cuyos nodos pueden ser alimentados de energía utilizando baterías.

## 2.3 ZigBee

Zigbee es un estándar de comunicaciones inalámbricas diseñado por la Zigbee Alliance con base en el estándar IEEE 802.5.4. Es un conjunto de protocolos de comunicación para el intercambio de datos inalámbricos entre dispositivos de bajo costo, bajo consumo y baja tasa de datos [9]. Esta tecnología ha sido concebida para permitir la implementación de redes sin infraestructura o redes *ad hoc*; esto es, que no requieren de algún dispositivo central o punto de acceso que se encargue de formar y organizar a la red. Una de las características más interesantes de ZigBee es que, a diferencia de los protocolos WiFi y Bluetooth, es que es capaz de utilizar a nodos intermedios, que funcionan como encaminadores o *routers*, para re-transmitir la información hacia algún nodo fuera del alcance del radiotransmisor del nodo emisor.

## 2.4 LoRa

LoRa (Long Range) es una tecnología inalámbrica que utiliza modulación en radiofrecuencia patentado por Semtech, usa una tecnología que se denomina Chirp Spread Spectrum (o CSS) [3]. El objetivo de LoRa es el de proporcionar una tecnología de bajo costo y consumo de energía, que sea capaz de transmitir información entre nodos separados por largas distancias. El estándar LoRa define los protocolos de la capa física de la red, en donde CSS es utilizado para la transmisión física de los bits. LoRaWAN es uno de los diversos protocolos que pueden ser utilizados con LoRa para definir los protocolos de las capas superiores: enlace y sub-capas de acceso al medio. LoRaWAN define ALOHA y *Escucha Antes de Hablar* (LBT – *Listen Before Talk*) como protocolos de acceso al medio.

## 2.5 6LoWPAN

6LoWPAN es un conjunto de estándares definidos por Internet Engineering Task Force (IETF), que construye y sostiene todo el trabajo y los estándares básicos de la arquitectura de Internet. Los estándares de 6LoWPAN hacen posible la utilización eficiente e ingeniosa de IPv6 sobre baja potencia, la baja velocidad de redes

inalámbricas en dispositivos integrados simples a través de una capa de adaptación y la optimización de protocolos relacionados [2].

## 2.6 WirelessHART

WirelessHART es un estándar (62591) global aprobado por la IEC que especifica una tecnología de malla interoperable autoorganizada en el cual los dispositivos de la red inalámbrica mitigan obstáculos en el ambiente del proceso, posee excelentes habilidades para el control y monitoreo, con una gran confiabilidad [7]. Una de las características sobresalientes de WirelessHART es que utiliza protocolos deterministas para el acceso al medio, lo que convierte a esta tecnología como idónea para aplicaciones I40.

## 3. EVALUACIÓN CUALITATIVA DEL DESEMPEÑO DE LOS PROTOCOLOS

En este proyecto, un conjunto de herramientas fue desarrollado para evaluar *jitter* y tasa de paquetes perdidos dentro de una conexión entre dos dispositivos. Se creó una aplicación cliente-servidor para cada uno de los protocolos escogidos a evaluar. Todas las aplicaciones fueron creadas con el lenguaje C, bajo el ambiente de Linux. Para algunos protocolos sólo se necesitaron algunas bibliotecas o *librerías* especializadas, mientras que, en otros, como en LoRa, fue necesaria una integración.

Después, se realizaron pruebas con las herramientas para medir la calidad de los distintos protocolos de comunicación de redes inalámbricas, con base en el *jitter* (segundos) y la tasa de paquetes perdidos (porcentaje) calculados en función del tamaño del buffer (2, 4, 6, 8, 10, 12, 14 y 16 bytes). Las pruebas se realizaron con y sin interferencia, a distintas distancias entre los dispositivos: 2.5m, 5m, 10m, 15m, 20m, 25m y 30m.

### 3.1 WiFi

Para el desarrollo de esta aplicación se utilizaron sockets. En la programación en C, un socket es un "enchufe" con el cual se crea una conexión con otro ordenador en la que se permite intercambiar datos dentro de una red. Se creó una aplicación cliente-servidor, la cual, con la ayuda de la biblioteca `<sys/socket.h>`, que es parte de la biblioteca GNU de C, se utilizaron sockets de tipo UDP (Datagramas) para crear conexiones entre dos nodos.

- Servidor: es el programa que permanece pasivo en la espera de que algún cliente solicite conexión. Puede o no devolver datos.
- Cliente: es el programa que solicita la conexión para enviar o solicitar datos al servidor.

Es una tecnología flexible y muy fácil de utilizar debido a la gran cantidad de documentación que existe al respecto, no fue necesario hacer instalaciones especiales, debido a que la biblioteca ya se encuentra entre las esenciales del lenguaje C, fue diseñada en un inicio para UNIX, pero en la actualidad existen versiones para una gran variedad de plataformas y sistemas operativos. La Fig. 1 muestra la arquitectura utilizada en los experimentos para evaluar WiFi. Es importante señalar que los nodos transmiten información directamente entre ellos y que el punto de acceso o modem WiFi se utiliza para la creación de la red. En el caso de los experimentos WiFi sin interferencia, se utilizó tan solo un punto de acceso (modem) para la creación de la red. En el caso de los experimentos con interferencia, en la evaluación de todos los protocolos se utilizó un punto de acceso WiFi para crear una red adicional, en la que un par de nodos se enviaban paquetes de manera masiva.

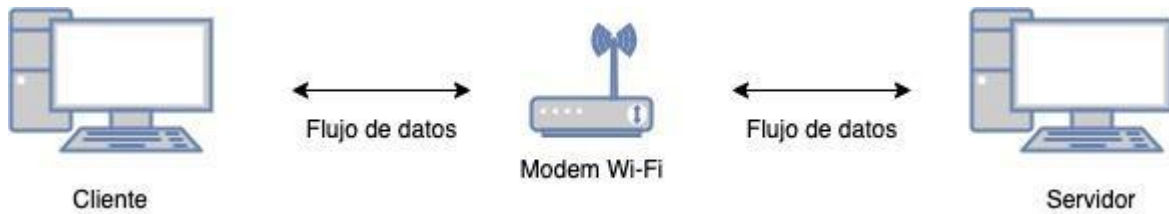


Figura 1. Descripción gráfica de las conexiones en el experimento Wi-Fi.

### 3.2 Bluetooth

Para evaluar el protocolo de comunicación Bluetooth, se realizó una aplicación cliente-servidor que utiliza la biblioteca BlueZ, una potente pila de comunicaciones Bluetooth con amplias APIs para el lenguaje C, que permite al usuario aprovechar los recursos Bluetooth locales. BlueZ proporciona soporte para las capas y protocolos centrales de Bluetooth.

Para establecer y usar conexiones, Bluetooth utiliza RFCOMM sockets, tecnología que se reduce a las mismas técnicas de programación de sockets mencionada anteriormente para la programación de TCP/IP. La única diferencia es que las estructuras de direccionamiento de los sockets son diferentes, y se utilizan diferentes funciones para el ordenamiento de bytes.

BlueZ es flexible y eficiente. Las bibliotecas y utilidades del *kernel* de BlueZ funcionan perfectamente en muchas arquitecturas compatibles con Linux. El soporte para BlueZ se puede encontrar en muchas distribuciones de Linux y, en general, es compatible con cualquier sistema Linux del mercado. La Fig. 2 muestra la arquitectura de los experimentos para la evaluación de Bluetooth.



Figura 2. Descripción gráfica de las conexiones en el experimento Bluetooth.

### 3.3 ZigBee

Para desarrollar aplicación en lenguaje C que mide la calidad de un enlace de red ZigBee, se utilizaron dispositivos ZigBee (Digi XBee®), los cuales, a través de un puerto serial/USB y con la ayuda de la biblioteca de código abierto `<xbee.h>`, especialmente creada para la intercomunicación con dispositivos xbee, se comunican con los programas cliente y servidor de la aplicación. La biblioteca `<xbee.h>`, resulta muy fácil de utilizar debido a la gran cantidad de ejemplos con los que cuenta en el repositorio de GitHub en el que se encuentra publicada. La tecnología sigue la misma lógica de programación que hemos venido viendo hasta el momento en las demás aplicaciones, de igual manera, lo que cambia son las funciones de direccionamiento y configuración de los dispositivos.

Además, para que funcionen adecuadamente, los dispositivos se deben configurar según la función a ejecutar, para esto se utiliza XCTU, una aplicación multiplataforma gratuita diseñada para que los desarrolladores

puedan interactuar con los módulos RF de Digi a través de una interfaz gráfica. Incluye herramientas que facilitan la instalación, configuración y prueba de los módulos XBee® RF. Utilizar la aplicación es bastante sencillo gracias a que la página oficial ofrece soporte y documentación para Windows y Linux en general.

Un dato muy importante que se debe tomar en cuenta al querer hacer una aplicación donde la información fluya de cliente a servidor y de servidor a cliente indistintamente, es que un dispositivo con el modo API desactivado no será capaz de recibir los datos correctamente, es por lo que, en esta aplicación, el dispositivo cliente también se configura como API. La Fig. 3 muestra la arquitectura utilizada para la evaluación del desempeño de ZigBee.



Figura 3. Descripción gráfica de las conexiones en el experimento ZigBee.

### 3.4 LoRa

La aplicación para evaluar al protocolo LoRa se compone por tres programas, dos de ellos: Sender y Receiver, fueron desarrollados para trabajar en placas Arduino UNO en conjunto con un módulo Ra-01 LoRa desarrollado por la compañía Ai-Thinker. Para poder programar el módulo en Arduino se utilizó la biblioteca: `#include <LoRa.h>`.

El tercer programa se desarrolló en el lenguaje C, el cual se comunica con la placa arduino para recibir los datos calculados por los programas que corren en las placas, todo esto a través de un puerto serial con la ayuda de la biblioteca `<termios.h>`. La Fig. 4 ejemplifica el diseño de la red para los experimentos con LoRa.

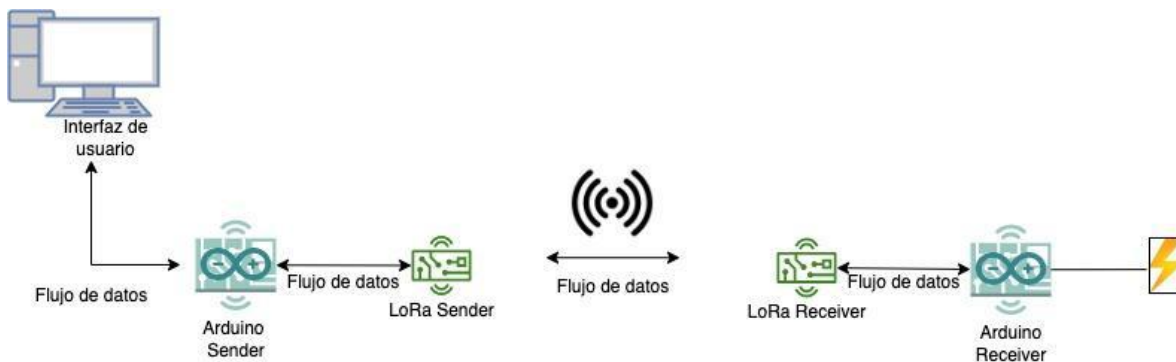


Figura 4. Descripción gráfica de las conexiones en el experimento LoRa.

#### 4. EVALUACIÓN CUANTITATIVA DEL DESEMPEÑO DE LOS PROTOCOL

Los resultados arrojados por las pruebas se vaciaron en gráficas para poder observar el comportamiento de los datos *jitter* y tasa de paquetes perdidos. El *jitter* es una medida del tiempo medio que toman los paquetes para llegar al nodo destino y de regreso al nodo origen. La tasa de paquetes perdidos mide el número de paquetes recibidos en función del número de paquetes enviados.

##### 4.1 Wi-Fi

##### Jitter

En general, como puede observarse en la Fig. 5, el *jitter* se mantuvo en cero segundos durante todo el experimento, tanto sin interferencia como con interferencia. Los dispositivos estaban separados por 2.5. Los resultados de los experimentos con una distancia entre nodos de 10 y 20 metros mostraron un comportamiento similar. Puede observarse que esta tecnología es bastante robusta a señales ruidosas.

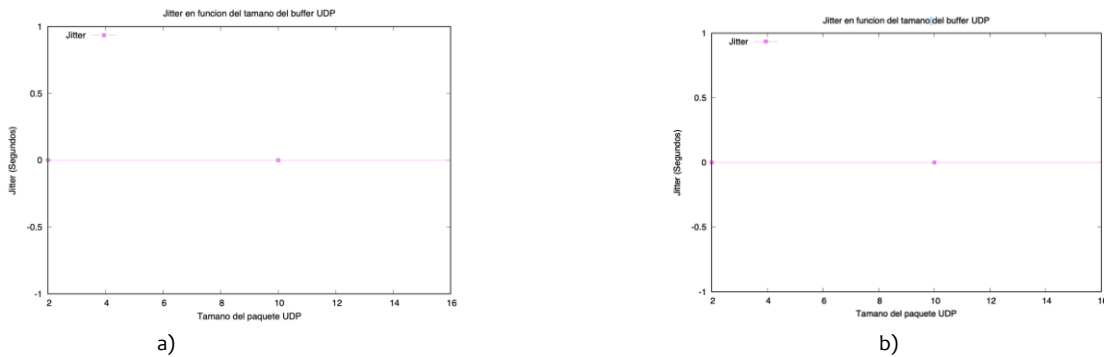


Figura 5. Jitter WiFi con 2.5 metros sin interferencia (a) y con interferencia (b)

##### Gráfica Tasa de paquetes perdidos

Como se muestra en la Fig. 6, la tasa de paquetes perdidos se incrementa significativamente con una distancia entre nodos de 20 metros en presencia de interferencia, mientras que para menores distancias (2.5 y 10 metros) la tasa en los experimentos sin y con interferencia fueron muy similares. Puede observarse que tanto la distancia como la interferencia influyen en el valor de este parámetro.

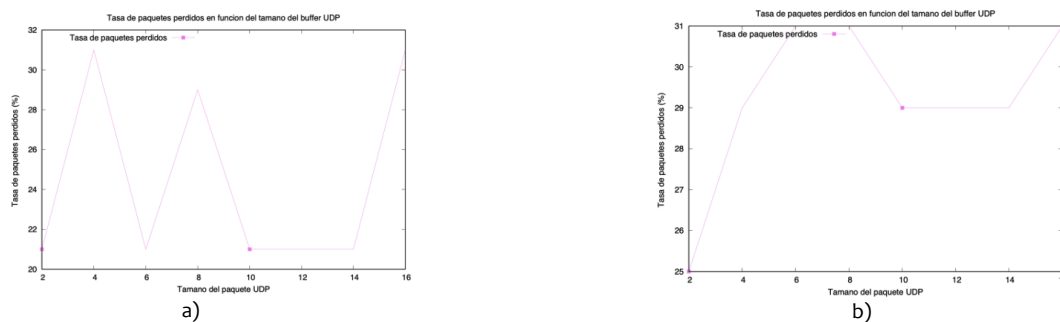


Figura 6. Tasa paquetes perdidos WiFi con 20 metros sin interferencia (a) y con interferencia (b)

## 4.2 Bluetooth

### Jitter

Los resultados del desempeño de Bluetooth en los experimentos se muestran en la Fig. 7. Las pruebas dieron el mismo resultado desde los 2.5m hasta los 10m en todas las medidas de buffer, con el jitter en 0 segundos. El resultado arrojó un pequeño cambio a los 15m sin interferencia, en el jitter en el tamaño de buffer 2, pero continuó en 0 el resto de las pruebas. En las pruebas de 30m, el jitter mostró un resultado más variado, pero, aun así, sigue siendo menos de 1 segundo.

Decir que tarda menos de 1 segundo puede hacer pensar que el sistema de envío de datos es muy rápido, pero de cierto modo también puede ser poco confiable para la programación de aplicaciones de tiempo real estricto, debido a las variaciones de microsegundos que suceden de manera aleatoria.

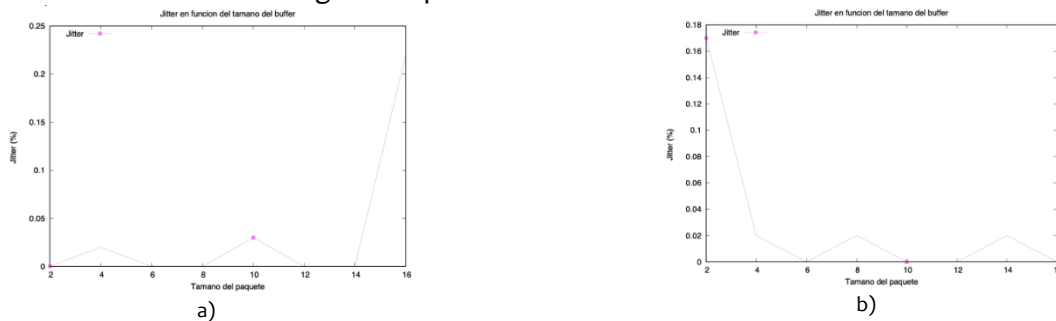


Figura 7. Jitter Bluetooth con 20 metros sin interferencia (a) y con interferencia (b)

### Tasa de paquetes perdidos

La tasa de paquetes perdidos se mantuvo en 0% durante todas las pruebas, a excepción de un par de ocasiones que subió al 90% y al 100% en las pruebas de 30 metros con interferencia. Por lo que se puede decir que Bluetooth es bastante confiable respecto al envío de información completa, debido a su tecnología multi-saltos (*multi hop*), en la que la transmisión se hace cambiando (saltando) de canales aleatoriamente. En la Fig. 8 se muestran los resultados de los experimentos con una separación de 30 metros entre nodos.



Figura 8. Tasa de paquetes perdidos Bluetooth 20 metros sin interferencia (a) y con interferencia (b)

## 4.3 ZigBee

### Jitter

Los resultados del jitter fueron bastante estables durante todas las pruebas, desde los 2.5 hasta los 30 metros con y sin interferencia, con un promedio entre 0 y 0.12 segundos de velocidad. Se esperaría que ZigBee funcione



en una aplicación del internet de las cosas, más por su estabilidad que por su velocidad. Los resultados del desempeño de ZigBee en función del Jitter, con una separación de 2.5 entre nodos, se muestran en la Fig. 9.

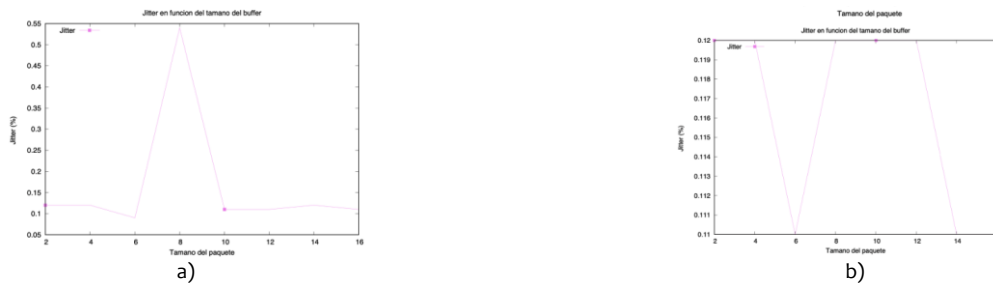


Figura 9. Jitter ZigBee con 2.5 metros sin interferencia (a) y con interferencia (b)

### Tasa de paquetes perdidos

La Fig. 10 muestra como la tasa de paquetes perdidos se mantuvo en 0 % durante todas las pruebas, con y sin interferencia. Por lo que se puede decir que ZigBee es bastante confiable respecto al envío de información completa, independientemente de distancias e interferencias.



Figura 10. Tasa de paquetes perdidos ZigBee con 30 metros sin interferencia (a) y con interferencia (b)

### 4.4 LoRa

Durante las pruebas de desarrollo, los dispositivos se comportaron como se esperaba, recibiendo y enviando datos de uno a otro, *jitter* y tasa de paquetes perdidos resultaron en 0 en las pruebas de 1/2 a 1 metro. Los resultados comenzaron a cambiar cuando se decidió aumentar la distancia entre los dispositivos; la comunicación se perdió a unos escasos centímetros más de distancia, se tendrán que repetir las pruebas con nuevos dispositivos para asegurar o descartar que los utilizados en el proyecto no funcionaban correctamente.

## 5. CONCLUSIONES Y TRABAJO FUTURO

Queda claro que, para realizar aplicaciones del Internet de las Cosas, algo muy importante por tener en consideración es la comunicación, así como la potencia requerida para transmitir y recibir datos. Por lo tanto, la elección de la tecnología de red correcta es fundamental. Elementos importantes por considerar son la distancia entre el emisor y el receptor, la naturaleza de los obstáculos, el ruido ambiental y las regulaciones gubernamentales definidas para cada tecnología. Se debe elegir un protocolo de red inalámbrica determinado que se ajuste a las necesidades del proyecto. Este estudio demuestra que existen muchas tecnologías inalámbricas para aplicaciones IoT, y que cada una tiene sus propias especificaciones, perjuicios y beneficios. Sin embargo, es importante recalcar que las tecnologías más modernas aún se encuentran en etapa de mejora y que no siempre será lo más sencillo trabajar con ellas; hacen falta más estudios dedicados a la configuración y su aplicación en ambientes de experimentación.

Con base en los resultados de los experimentos llevados a cabo, se encontró que las tecnologías utilizadas mostraron un muy buen desempeño y robustez independientemente de la distancia y el ruido. Sin embargo, si hay diferencias cualitativas a considerar. WiFi, ZigBee y Bluetooth son tecnologías que pueden considerarse más maduras y de las que hay más opciones de hardware, software y documentación. En el caso del resto de las tecnologías evaluadas, esto es, LoRa y 6LoPan, se encontraron más limitaciones de opciones de hardware, software y soporte técnico. Por otra parte, las principales desventajas de WiFi son que se trata de una tecnología con infraestructura; esto es, se requiere la presencia de un punto de acceso para la creación y operación de la red. Adicionalmente, en muchos casos debe tenerse acceso a una fuente permanente de suministro de energía eléctrica. En el caso de Bluetooth, la principal desventaja es la distancia máxima alcanzada entre nodos, sin el uso de repetidores. Por tal motivo, se considera que actualmente ZigBee es una muy buena opción para implementar aplicaciones IoT robustas.

En un trabajo futuro, esta investigación se extenderá para revisar las tecnologías de comunicación que por falta de tiempo y poca disponibilidad de dispositivos no se revisaron más a fondo. Sería un gran aporte a la comunidad un documento que especifique cómo configurar y usar las tecnologías existentes para que cualquier investigador o desarrollador lo tome como base y no tenga que empezar su proyecto desde cero.

## REFERENCIAS

- [1] Eured. Bluetooth. 04 2021.
- [2] Khalid Fakieh. A survey on 6lowpan its future research challenges. 01 2014.
- [3] Juha, Konstantin Mikhaylov, Marko Pettissalo, Janne Janhunen, and Jari Linatti. Performance of a low-power wide-area network based on lora technology: Doppler robustness, scalability, and coverage. 2017.
- [4] Saliyah Kahar, Riza Sulaiman, Anton Satria Prabuwo, Nahdatul Akma Ahmad, Mohammad Ashri, and Abu Hassan. A review of wireless technology usage for mobile robot controller. IPCSIT vol. 34 (2012), 04 2012.
- [5] William MacDougall. Industrie 4.0: Smart manufacturing for the future. macdougall2014industrie, 2014.
- [6] Claudia Martínez. Evaluación de protocolos de redes de computadoras para aplicaciones de la industria 4.0. Instituto Tecnológico Nacional de México, 2021.
- [7] Leonardo Pérez Martínez. Consideraciones de diseño e implementación de un sistema de monitoreo inalámbrico en campo con tecnología wireleshart. Instituto Politécnico Nacional, 09 2016.
- [8] Michael, Markus Lorenz, Philipp Gerbert, Manuela Waldner, Jan Justus, Pascal Engel, and Michael Harnisch. Industry 4.0: The future of productivity and growth in manufacturing industries. Boston Consulting Group, 2015.
- [9] Manuel Menchaca Paz. Integración de zigbee/6lowpan en una red de sensores inalámbrica. Universidad de Cantabria, 07 2012.

Correo autor: [floresb.alejandra@gmail.com](mailto:floresb.alejandra@gmail.com)