**Diseño e Implementación de un Sistema de Comunicación Inalámbrica para Aplicaciones de Robótica de Enjambre – Fase II**

**Design and Implementation of a Wireless Communication System for Swarm Robotics Applications – phase II**

Hee Chan Kim (kim17040@uvg.edu.gt)

Departamento de Ingeniería Electrónica, Mecatrónica y Biomédica, Facultad de Ingeniería, Universidad Del Valle de Guatemala

**Resumen**

La robótica de enjambre ha surgido como una propuesta para resolver problemas de optimización que han ido aumentando su nivel de complejidad con el avance de la tecnología. Sin embargo, con la visión de implementar esta tecnología de manera física a diferencia de una simulación, se enfrenta a un reto que es intrínseco de una implementación física, la comunicación entre robots. Por este motivo, el enfoque principal del proyecto es proponer un sistema de comunicación descentralizada que cumplan con las características de la robótica de enjambre: auto-organización, escalabilidad y robustez. Una investigación y evaluación de las topologías y tecnologías actuales nos dirigieron a construir una red entorno a una topología descentralizada y utilizar Zigbee como tecnología de comunicación inalámbrica. Con las características de la robótica de enjambre en mente desarrollamos servicios para Zigbee capaces de mitigar o eliminar los problemas relacionados a las redes y robótica de enjambre. Validamos el funcionamiento de los servicios armando una red de 5 dispositivos y probando cada servicio 2000 iteraciones. Además, medimos el desempeño de la red para tener una métrica en el futuro sobre el funcionamiento de la red en aplicaciones de robótica de enjambre.

**Palabras clave:** Zigbee, robótica de enjambre, topología descentralizada.

**Abstract**

Swarm robotics has emerged as a proposal to solve optimization problems that have increased in complexity with the advancement of technology. However, with the vision of implementing this technology in a physical way as opposed to a simulation, it faces a challenge that is intrinsic to a physical implementation, the communication between robots. For this reason, the main focus of the project is to propose a decentralized communication system that meets the characteristics of swarm robotics: self-organization, scalability and robustness. An investigation and evaluation of current topologies and technologies led us to build a network around a decentralized topology and use Zigbee as a wireless communication technology. With the characteristics of swarm robotics in mind, we developed services for Zigbee capable of mitigating or eliminating problems related to networks and swarm robotics. We validate the operation of the services by setting up a network of 5 devices and testing each service for 2000 iterations. In addition, we measure the network performance to have a future metric on network performance in swarm robotics applications.

**Keywords:** Zigbee, swarm robotics, decentralized topology

**Introducción**

En este trabajo se presenta una alternativa a la topología seleccionada en un trabajo de graduación anterior, que era una topología de estrella. Una de las motivaciones para presentar una alternativa es que el comportamiento colectivo del enjambre se encontraba en peligro por cualquier falla del router. Con la alternativa propuesta se pretende tener una mayor fiabilidad de los algoritmos de inteligencia de enjambre al brindar una red más robusta. Para alcanzar esta meta se establecieron los objetivos enfocados a seleccionar y diseñar la red alterna.

Maximizar ganancias o minimizar pérdidas ha sido siempre el énfasis de resolver los problemas ingenieriles y para diversas ramas de la ciencia los problemas de optimización han ido aumentando su nivel de complejidad con el avance de la tecnología. Por este motivo, surgió la robótica de enjambre que es la coordinación de sistemas de agentes que interactúan entre si y con el entorno acorde a una inteligencia de enjambre. Los distintos algoritmos que se están desarrollando en la inteligencia de enjambre como el algoritmo de optimización por enjambre de partículas (PSO, por sus siglas en inglés), algoritmo de optimización por colonia de hormigas (ACO, por sus siglas en inglés), entre otros, han servido como herramientas para resolver problemas estocásticos a un nivel de simulación. Sin embargo, con la visión de implementar esta tecnología de manera física como respuesta a resolver tareas complejas con un diseño económico se enfrenta a un reto que es intrínseco de una implementación física, la comunicación entre robots. A diferencia de una simulación donde se itera una función o un algoritmo que es equiparable a la comunicación entre robots, en la realidad se necesita traducir las iteraciones a interacciones entre robots y es esta traducción al mundo físico que requiere de un sistema de comunicación entre los robots para llevar a cabo los algoritmos a fin de encontrar la solución óptima.

Por este motivo, el enfoque principal del proyecto es proponer sistemas de comunicación descentralizada que cumplan con las características de la robótica de enjambre: auto-organización, escalabilidad y robustez (Cianci & Raemy & Pugh & Martinoli, 2007). Dadas estas tres características de la robótica de enjambre se proponen sistemas de comunicación alternativos al presentado en la fase I, red WiFi con topología de estrella, ya que este sistema no cumple con la robustez. Una red WiFi con topología de estrella requiere de un router para que todos los hosts que serían los robots pudieran comunicarse, por ende, si llega a fallar el router se cae toda la red y los robots ya no pueden comunicarse. Entonces la propuesta de una red descentralizada busca solucionar este problema al brindar una comunicación entre pares en donde si falla un robot, simplemente dejan de comunicarse con él.

Finalmente, al proponer sistemas de comunicación descentralizada se tiene la intención de construir distintas opciones que implementan distintos protocolos en cada capa del modelo de referencia OSI con el fin de utilizar la red que mejor se adecúe a un algoritmo de inteligencia de enjambre.

Como metodología, primero se tiene comparar las características que describen una topología para todas las topologías propuestas y utilizar las características de robótica de enjambre como criterio para seleccionar la topología más adecuada. Se realiza la comparación para las distintas tecnologías de comunicación con las mismas características para tener la más adecuada. Luego, como se busca tener una aplicación real, se seleccionaron y se compraron los transceptores necesarios para construir una red física.

Con la topología, tecnología y transceptores definidos se prosiguió a construir la red. Se desarrollaron las funciones necesarias para la comunicación entre el microcontrolador y el transceptor. También, se desarrollaron los servicios necesarios para resolver los problemas inherentes a las redes de comunicación. Finalmente, se realizaron experimentos para medir el rendimiento de la red y brindar una estructura completa del sistema de comunicación para futuras investigaciones.

**Materiales y métodos**

Debido a que el enfoque de este trabajo se enfoca en proponer una topología alternativa para resolver los problemas de falla de punto único se partió de una investigación para la topología, las tecnologías y los módulos disponibles en el mercado. Luego se realizó el desarrollo de la librería y las pruebas necesarias para validar su funcionamiento.

**Elección de topología, tecnología y hardware**

Para la elección de la topología y tecnología se utilizaron las características de robótica de enjambre como criterios. Se utilizaron estos criterios para validar que conceptual y físicamente la red construida esté orientada hacia aplicaciones de robótica de enjambre. Las características de robótica de enjambre son: auto-organización, escalabilidad y robustez (Cianci & Raemy & Pugh & Martinoli, 2007).

La auto-organización en el sistema de comunicación debe de cumplir con los principios de retroalimentación positiva, retroalimentación negativa, aleatoriedad y múltiples interacciones. La escalabilidad radica en la capacidad que tiene el sistema para crecer sin que su rendimiento se vea afectado. La robustez es importante en la robótica de enjambre porque el comportamiento de estos algoritmos radica en la multitud y si llegara a fallar algún aspecto en la red de tal manera que se pierde la naturaleza de enjambre desaparece la inteligencia de enjambre.

La forma en que se organizó y se tomó una decisión fue a partir de un trade study.

*Eleccióon de la topología*

Se identificaron ocho aspectos cualitativos que describen una topología y se clasificaron en cada una de las tres características de robótica de enjambre.

* Alcance
* Tolerante a colisiones
* Fiabilidad
* Complejidad
* Flexibilidad
* Velocidad
* Seguridad
* Enrutamiento

Las topologías propuestas abarcan tanto centralizadas como descentralizadas para discriminar los aspectos descritos en base a los criterios propuestos de robótica de enjambre. Las cinco topologías propuestas son las siguientes:

* Bus
* Anillo
* Estrella
* Árbol
* Malla

Se realizó un estudio de compensación en el cual se midió cada propuesta en los ocho aspectos que se describieron anteriormente y se utilizaron (Mousa & Mohammed, 2019), (Jassam & Guirguis), (Soparia & Bhatt, 2014), (Shrestha & Xin, 2007) y (Chong & Arun, 2021) para calificar cada topología. Para cada aspecto se calificaron las topologías de mejor a peor siendo 1 la peor topología y 5 la mejor topología. Finalmente, a cada aspecto se le asignó un peso bajo el criterio de las características de robótica de enjambre donde un peso igual a 10 es el peso base.

*Elección de la tecnología*

Luego de tener la topología seleccionada se prosiguió a deliberar entre las tecnologías de comunicación inalámbrica. Se seleccionaron las cinco opciones que pueden implementarse en una topología descentralizada y los aspectos que se utilizaron para describir las redes son más puntuales, debido a que son variables cuantitativas en su mayoría y se obtuvieron de las páginas oficiales de cada tecnología (DigiKey, 2021), (ESPRESSIF,2016), (Zigbee, 2021), (Horyachyy, 2017), (ZWAVE, 2020), (Lin & Talty & Tonguz, 2015) y (ANT, 2021).

**Tabla 1.** Tecnologías para la topología descentralizada.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Tecnología** | **Nodos** | **Consumo [mA]** | **Costo [$]** | **Alcance [m]** | **Velocidad de transmisión [Mbps]** | **Topología** |
| Zigbee | >65000 | 29 | 3.92 | 10-100 | 0.250 | Malla, estrella |
| BLE | >65000 | 18 | 2.38 | 50 | 1.000 | P2p |
| ESP NOW | 20 | 200 | 4.20 | 100 | 1.000 | P2p, estrella |
| ANT | >65000 | 16 | 3.93 | <30 | 1.000 | P2p, estrella, árbol, malla |
| Z-wave | 232 | 23 | 9.84 | 30 | 0.100 | Estrella, malla |

Al igual que con la topología se realizó un estudio de compensación para determinar la tecnología implementada en la red. Se compararon las cinco tecnologías en los aspectos descritos en la tabla 1 y se enumeraron de 1 a 5, siendo 1 la peor tecnología y 5 la mejor tecnología. Además, se utilizó el criterio de las características de robótica de enjambre para darle un peso a cada aspecto y, nuevamente, el peso base es igual a 10.

*Elección del transceptor*

Se seleccionaron dos opciones por su fácil implementación y bajo costo relativo a los demás módulos de la misma tecnología. El primero es Xbee Module ZB series S2C que tiene un consumo de 40 mA mientras está operando y menos de 1 µA cuando está en modo dormido y un alcance de 40 metros en lugares cerrados. El segundo es el módulo MRF24J40MA desarrollado por Microchip que tiene un consumo de 23 mA mientras está operando y 2 µA mientras está dormido y un alcance de 38 metros en lugares cerrados. Ambos módulos presentan parámetros parecidos, sin embargo, la principal diferencia radica en los precios ya que el primero tiene un precio de $22.95 y el segundo de $7.40. Por otra parte, Xbee es un módulo del que se encuentra mayor documentación e información para implementar y programar que MRF24J40MA.

**Diseño de la red**

Primero se especificaron los supuestos de la red para su diseño. Luego, se describen las funciones básicas que se tuvieron que desarrollar para la construcción de los servicios en el lenguaje de programación C++. Finalmente, se describen los cuatro servicios que se desarrollaron: servicio de asociación, servicio de sincronización de tiempo, algoritmo de nodos conectados, y servicio de elección de coordinador.

Supuestos

* Todos los canales que utiliza Zigbee en la frecuencia de 2.4 GHz están libres.
* El broadcast ejecutado por el coordinador es recibido por todos los nodos.
* Todos los nodos son dispositivos de funcionalidad completa (FFD).
* Todos los nodos tienen el transceptor encendido.

**Medición de la red**

En la red que se construyó entorno a ZigBee es necesario que se determinen métricas que describan la red física. Como se ha presentado este trabajo, la transición de una simulación a la realidad trae problemas inherentes a la topología y tecnología que se seleccionaron. Por lo tanto, las métricas que se determinaron muestran el desempeño de la red construida y servirá para futuras investigaciones en la línea de robótica de enjambre para que se tenga un contexto real de la capacidad de la comunicación del enjambre.

En la tesis de Kui Liu (Liu, 2009), se encuentra una metodología para evaluar ZigBee y tener una referencia real de su desempeño. En el trabajo se encuentran cinco métricas estudiadas:

* Tasa de rendimiento efectiva
* Tamaño de la red
* Seguridad
* Consumo energético
* Alcance y retardo en paquetes

Sin embargo, para la medición de la red propuesta se enfocarán únicamente en la tasa de rendimiento efectiva, consumo energético y alcance y retardo de paquetes. No se enfocó en la seguridad y el tamaño de la red ya que tanto (Liu, 2009) y el presente trabajo tienen estas métricas definidas por el protocolo ZigBee.

ZigBee ofrece ocho niveles de seguridad para asegurar la integridad y confidencialidad de los datos. También, el tamaño de la red para Zigbee dicta como límite un poco más de sesenta y cinco mil nodos en la misma red.

Para la evaluación de las otras métricas se tuvieron estas metodologías.

*Metodología para la tasa de rendimiento efectiva*

Acorde a la documentación de Zigbee (Zigbee,2021), la tasa de transferencia es de 250 kbps para la frecuencia de 2.4 GHz. Sin embargo, debido a los encabezados que se agregan en cada capa la tasa de la carga útil disminuye. Es por ello, que el experimento para evaluar la tasa de rendimiento efectiva consiste en determinar cuántos mensajes de la carga útil llegan desde el emisor al receptor en un segundo.

El emisor envía inmediatamente un mensaje cuando recibe una confirmación del receptor, que ocurre cuando el mensaje llega completo. Cuando el receptor recibe un paquete del emisor, este revisa si el tamaño y el contenido son los correctos o no. Si no es correcto, el paquete es descartado y no se toma en cuenta en el cálculo de la tasa efectiva.

Se construyó un programa para el emisor y otro para el receptor. El emisor envía una carga útil para los ciento dieciséis bytes. Los demás bytes de la trama en Zigbee son utilizados para los encabezados. Este paquete lo recibe el receptor y compara la carga útil para confirmar que el paquete llegó completo.

*Metodología para el consumo energético*

El consumo energético es importante para las aplicaciones de robótica de enjambre debido a que los robots no estarán estáticos todo el tiempo. Por lo tanto, necesitaran una fuente de energía que pueden cargar como una batería 18650 o bien cables suficientemente largos para los robots, pero esta solución no es eficiente.

Entonces la medición del consumo energético se basa en determinar la corriente demandada por el microcontrolador para comunicarse con el transceptor y enviar y recibir paquetes. Se implementó el circuito de la imagen 1 en el cual se mide el diferencial de potencial en la resistencia de 10 Ω y mediante la ecuación:

Se despeja para la corriente (I) y se calcula con el diferencial de potencial medido y la resistencia de 10 Ω.

Diagram, schematic

Description automatically generated

**Imagen 1.** Circuito para medir el consumo energético.

*Metodología para el alcance y retardo en paquetes*

Finalmente, se empleó una prueba similar a (Liu, 2009) para el alcance y retardo de paquetes siendo la única diferencia que se empleó la antena del PCB y no se sustituyó como en la tesis de Kui Liu.

Se calculó el tiempo de ida y vuelta, RTT, para determinar el retardo de paquetes. Además, las medidas se realizaron a distintas distancias y se contaron los paquetes exitosos y los fallidos para establecer la tasa de paquetes perdidos. De esta manera, se puede establecer una tolerancia máxima de paquetes perdidos para definir la distancia máxima a la que pueden estar los robots.

Para el cálculo de RTT se consideran los siguientes tiempos:

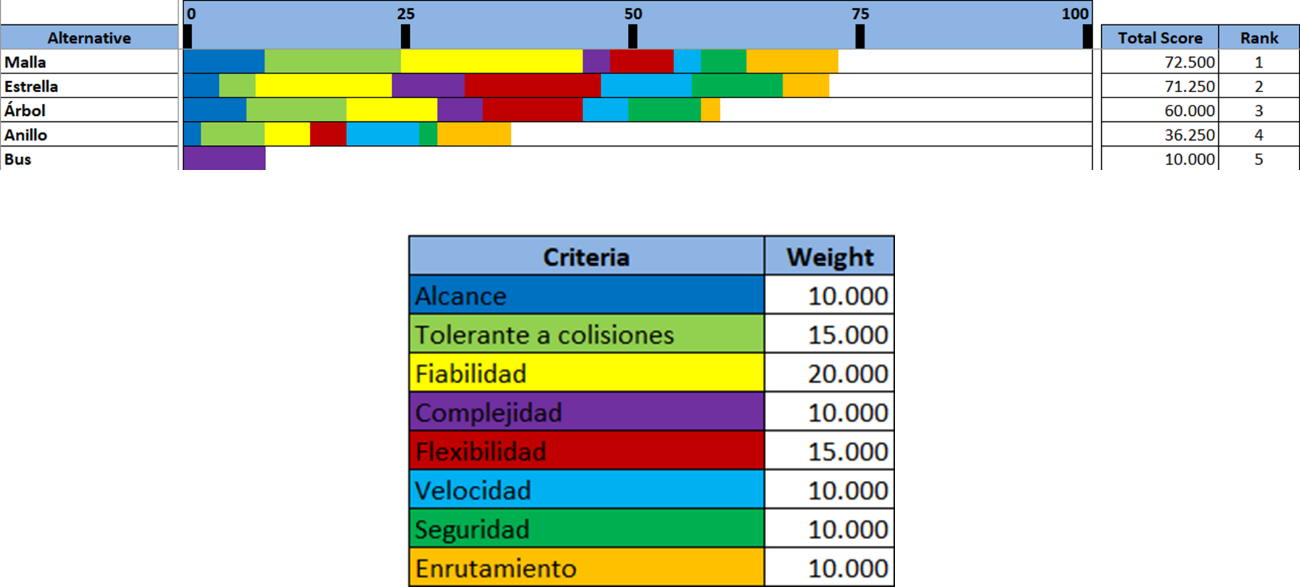
* Tiempo de procesamiento de la capa superior a la inferior en el emisor y de la inferior a la superior en el receptor.
* El tiempo de cola para poder enviar el mensaje. También se toma en cuenta el tiempo que le toma al nodo ganar el acceso al medio.
* Tiempo de transmisión a una velocidad de 250 kbps.
* Tiempo de propagación en el aire.

El proceso consiste en el envío de un mensaje y que el receptor lo procese para leerlo y comparar si es el mensaje correcto. Una vez confirmado el mensaje, se responde con una palabra de confirmación para terminar una vuelta. El tiempo de ida y vuelta comienza desde que se envió el primer paquete hasta que se recibe la palabra de confirmación. En el caso que se pierda el paquete en la transmisión, existe un temporizador que al llegar a cero se reenvía el mensaje; el tiempo de la retransmisión también se considera en el tiempo de ida y vuelta y el paquete que nunca llegó se considera como un paquete perdido. Se realizaron dos mil corridas para calcular el tiempo de ida y vuelta a distintas distancias.

**Resultados y discusión**

**Elección de topología**

La topología que tuvo un mejor resultado para las aplicaciones de robótica de enjambre es la topología de malla como se observa en la imagen 2.



**Imagen 2.** Trade study de la topología.

Se asignó a la fiabilidad un mayor peso bajo el criterio de robustez porque el funcionamiento correcto de los algoritmos de enjambre implica una comunicación segura entre los robots del enjambre. Si llegara a caerse la red, el enjambre completo dejaría de funcionar.

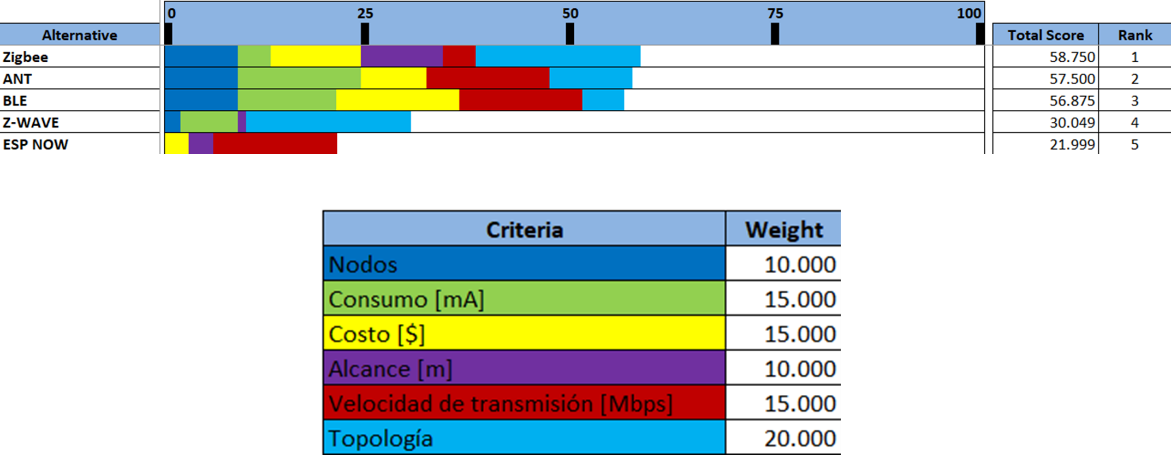
En segundo lugar, se les asigno un mayor peso a los aspectos tolerante a colisiones y flexibilidad; el primero bajo los criterios de robustez y escalabilidad, mientras que el segundo bajo los criterios de escalabilidad y auto-organización. Una red debe ser tolerante a colisiones ya que la robótica de enjambre implica tener un grupo de robots que puede escalar en gran número de robots y de no ser tolerante a colisiones se pierde el comportamiento colectivo del enjambre. La robótica de enjambre debe ser flexible ya que para que los robots cumplan con sus tareas deben de posicionarse de manera dinámica y este proceso requerirá de varios algoritmos, protocolos y servicios de la red. Por lo tanto, una mayor flexibilidad en la red brindará una mayor libertad al enjambre.

En tercer lugar, está el enrutamiento bajo los criterios de auto-organización y escalabilidad. Cualquier mejora en la transmisión de los paquetes significa que los servicios de la red fluyen con mayor facilidad y mejora la comunicación para redes con varios nodos. En consecuencia, mejora el comportamiento colectivo.

Finalmente, el alcance, la complejidad, la velocidad y la seguridad el menor peso porque si bien afectan la red en general y tendrán un impacto para la robótica de enjambre, no son suficientemente significativos como los demás aspectos.

**Elección de la tecnología**

La tecnología que tuvo un mejor resultado para aplicaciones de robótica de enjambre fue ZigBee como se observa en la siguiente imagen.



**Imagen 3.** Trade study de la tecnología.

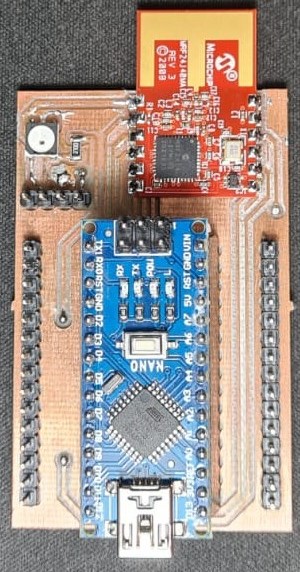
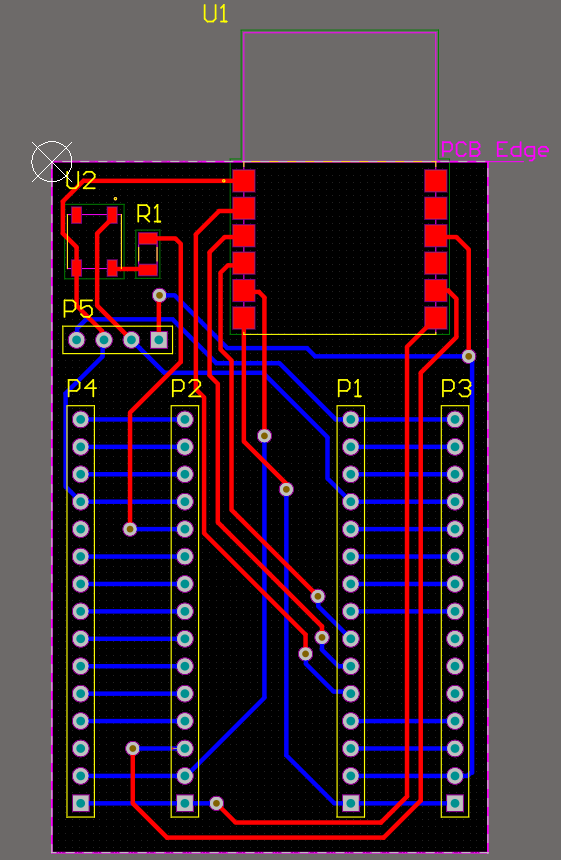
Se asignó al aspecto de topología la mayor importancia en el estudio de compensación debido a que la topología que se seleccionó era una topología en malla. En consecuencia, la tecnología que se busca debe de ser compatible con esta topología. Luego, se les asignó un peso poco menor al consumo, al costo y a la velocidad de transmisión. Debido a que las aplicaciones están orientadas a robótica de enjambre, algunos algoritmos requieren de un funcionamiento autónomo y por ello se tomó en cuenta el consumo de la tecnología. También, al ser un grupo de robots el costo puede incrementar y una tecnología de bajo costo puede reducir los gastos del hardware que se necesita. Bajo el criterio de las características de robótica de enjambre una mayor velocidad de transmisión significa un mejor comportamiento del enjambre. Finalmente, como últimos aspectos se tienen el número de nodos y el alcance. El primero, ya que todas las tecnologías permiten conectar varios nodos a la red, y el segundo, porque entre los objetivos que se plantearon está diseñar e implementar el sistema de comunicación en la mesa de pruebas de UVG, todas las tecnologías cumplen con la restricción de alcance para las dimensiones de la mesa.

**Elección del transceptor**

El dispositivo que se seleccionó fue el MRF24J40MA desarrollado por Microchip por su bajo costo y baja carga de procesamiento del microcontrolador a utilizar. El módulo de Microchip tiene integrado la capa física y la capa MAC por lo que se reduce el costo computacional por parte del MCU. Entre las características a destacar de este módulo están (Microchip, 2016):

* Detección de energía. Es útil ya que se podrá medir el nivel de energía en el canal y servirá para determinar si el canal está ocupado. Sin embargo, no puede determinar si esa energía corresponde a otro dispositivo 802.15.4 o no.
* Escucha de señal portadora (CS, por sus siglas en inglés). Esto consiste en demodular la señal recibida para determinar si es compatible con el estándar. El canal se considera ocupado si la señal es compatible con 802.15.4.
* Evaluación de canal libre (CCA, por sus siglas en inglés) y mecanismo CSMA-CA. Este mecanismo hace que la MAC le pida a la capa física que evalúe el canal para ver si está libre. El CCA es parte del manejo de la capa física.
* Retransmisión automática de datos.
* Respuesta ACK automática.
* Transmisión independiente o con baliza.
* Encriptación y desencriptación para la capa MAC y capas superiores.

Adicionalmente, se desarrolló un shield para el Arduino Nano con el fin de realizar las pruebas del módulo y mediciones de la red. El programa para desarrollar la placa fue Altium Designer. El shield cuenta con un led RGB, el módulo MRF24J40MA y los pines machos para conectarse al Arduino Nano.

**(a)**Placa física **(b)**Placa diseñada

**Imagen 4.** Shield para el MRF24J40MA

**Diseño de la red**

*Servicio de asociación*

Uno de los servicios necesarios en la red es la de asociación automática. Ya que en el caso que el coordinador se pierda, alguien más puede tomar su lugar y en el peor de los casos comenzar una nueva red. Entonces se desarrolló un algoritmo para que usuarios que no pertenecen a una red hagan un barrido entre los canales disponibles según el protocolo y se asocien de manera automática para que no sea necesario programarles su dirección y la red a la que pertenecen.

La solicitud ocurre en la capa de Red y manda la solicitud a través de la capa inferior, MAC, para poder hacer un broadcast y solicitar la asociación a cualquier red existente dentro de su rango. En el instante que el coordinador de la red recibe la solicitud automáticamente responde con un ack y transfiere la solicitud a la capa superior, Red, quien selecciona una dirección disponible dentro de la red para asignársela al nuevo miembro de la red. Mientras tanto, el nuevo usuario, al recibir el ack, reinicia el contador para expirar el servicio y se prepara para recibir el mensaje con la información de la nueva red. Cuando el coordinador envía la información a la dirección del usuario, este le responde con la palabra “ok” para confirmar la recepción y asociación del nuevo usuario. De parte del usuario, modifica su información en la capa de Red y manda una confirmación a la capa superior para terminar el servicio de asociación.

Se empleó un temporizador como mecanismo para determinar cuándo se acaba el servicio. Cuando ocurre la interrupción del temporizador se para el servicio y la asociación a la red fue fallida. Es importante agregar mecanismos como estos para que el MCU no quede atrapado dentro de un ciclo infinito.

El desempeño del servicio se midió realizando 2000 iteraciones del servicio y los resultados se presenta a continuación.

**Tabla 2.** Comparación de éxitos para el servicio de asociación.

|  |  |  |
| --- | --- | --- |
| **Distancia [m]** | **Iteraciones con éxito** | **Porcentaje de éxito [%]** |
| 2 | 1999 | 99.95 |
| 6 | 1981 | 99.05 |
| 10 | 1964 | 98.20 |

En la tabla 2 se muestra el resultado de asociar un nodo a la red creada por un coordinador. Se clasifica como éxito cuando el nodo después de modificar su dirección y su PAN manda el mensaje de confirmación al coordinador y éste actualiza la información de los nodos conectados.

*Servicio de sincronización*

El algoritmo se desarrolló con el concepto de los espacios de tiempo. El coordinador es el nodo encargado de mandar el mensaje por medio de un broadcast a todos los miembros de la red en todos los canales disponibles para que activen la bandera de sincronización e inicien el temporizador y puedan enviar sus mensajes cuando el tiempo llegue al desfase particular respecto el momento en que se recibe el mensaje del coordinador. Una vez enviado el mensaje, se baja la bandera, se reinicia el temporizador y queda a la espera nuevamente del mensaje del coordinador para iniciar el temporizador nuevamente. Este algoritmo otorga robustez a la red ya que garantiza que cada miembro de la red envíe su mensaje en un tiempo determinado y se eviten las colisiones.

El desempeño del servicio se midió realizando 2000 iteraciones del servicio y los resultados se presenta a continuación.

**Tabla 3.** Comparación de éxitos para el servicio de sincronización

|  |  |  |
| --- | --- | --- |
| **Distancia [m]** | **Iteraciones con éxito** | **Porcentaje de éxito [%]** |
| 2 | 2000 | 100.00 |
| 6 | 1972 | 98.60 |
| 10 | 1959 | 97.95 |

En la tabla 3 se muestra el resultado de sincronizar el mensaje de cada nodo asociado a la red. La red con la que se realizaron las pruebas contiene un coordinador y dos nodos. Se clasifica como éxito cuando los mensajes de los nodos llegan en el orden correcto y dentro del tiempo esperado. Si llega un mensaje del segundo nodo conectado primero, el servicio falló. Si los mensajes no llegan en el tiempo esperado para dos nodos conectados, el servicio falló.

*Algoritmo de nodos conectados*

El coordinador envía un mensaje a los nodos uno por uno esperando recibir un ACK como respuesta en un tiempo definido. En el caso que ocurre la interrupción del temporizador y el nodo nunca respondió con el ACK, el coordinador llega a la conclusión que el nodo está desconectado y actualiza la red. Cuando el coordinador termina de revisar todos los nodos conectados procede a enviar un paquete por medio de broadcast para que todos actualicen la información de la red.

El desempeño del servicio se midió realizando 2000 iteraciones del servicio y los resultados se presenta a continuación.

**Tabla 4.** Comparación de éxitos para el algoritmo de nodos conectados

|  |  |  |
| --- | --- | --- |
| **Distancia [m]** | **Iteraciones con éxito** | **Porcentaje de éxito [%]** |
| 2 | 1999 | 99.95 |
| 6 | 1996 | 99.80 |
| 10 | 1998 | 99.90 |

En la tabla 4 se muestra el resultado del proceso del coordinador al revisar si ningún nodo de la red se ha desconectado y actualizar la información de la red en todos los nodos. La red con la que se realizaron las pruebas contiene un coordinador y dos nodos. Se clasifica como éxito cuando todos los nodos responden al coordinador para informarle que siguen conectados. Cuando un nodo no llega a responder en el tiempo fijado el coordinador lo interpreta como si el nodo se ha desconectado y un fallo al servicio.

*Servicio de elección de coordinador*

En el servicio de elección de coordinador se incluye un algoritmo para el coordinador y un algoritmo para los nodos. El algoritmo para el coordinador es trivial ya que únicamente debe de enviar un PING a todos los nodos en un tiempo T. Mientras que el algoritmo para los nodos es un poco más extenso.

Cada nodo espera recibir un PING del coordinador para estar al tanto que el coordinador sigue conectado. Este mensaje debe de llegar antes de un tiempo 2T. Cuando transcurre el tiempo 2T y no recibió ningún PING, el nodo inicia el servicio de elección de coordinador. El nodo cambia a candidato, activa un temporizador para caducar el servicio y envía un mensaje a toda la red para informar que el coordinador se desconectó y que ahora él es un candidato. Cada nodo recibe el mensaje y están enterados de la situación, pero no responden. Cuando transcurre el tiempo 2T para cada uno de los nodos, revisan si alguien ya se postuló como candidato y confirman que se desconectó el coordinador enviando su voto al candidato que comenzó el servicio. Además, activan sus temporizadores esperando una confirmación del nuevo coordinador.

En el caso que existan dos nodos que se postularon como candidatos, ambos enviaron sus solicitudes a la red. Sin embargo, cada uno de los nodos votará únicamente por uno; por el que envió la solicitud de primero. Cuando el candidato recibe suficientes votos para confirmar su elección, cambia su configuración de candidato a coordinador y envía un broadcast a toda la red informando que hay un nuevo coordinador.

Si ocurre la interrupción del temporizador del candidato y no llegó a los suficientes votos, el candidato cambia su estado a un nodo más y espera el siguiente PING del coordinador. Si ocurre la interrupción del temporizador para los nodos que votaron y el candidato no tomó el puesto del coordinador, los nodos esperan el siguiente PING del coordinador anterior.

El desempeño del servicio se midió realizando 100 iteraciones del servicio y los resultados se presenta a continuación.

**Tabla 5**. Resultados del servicio de elección de coordinador.

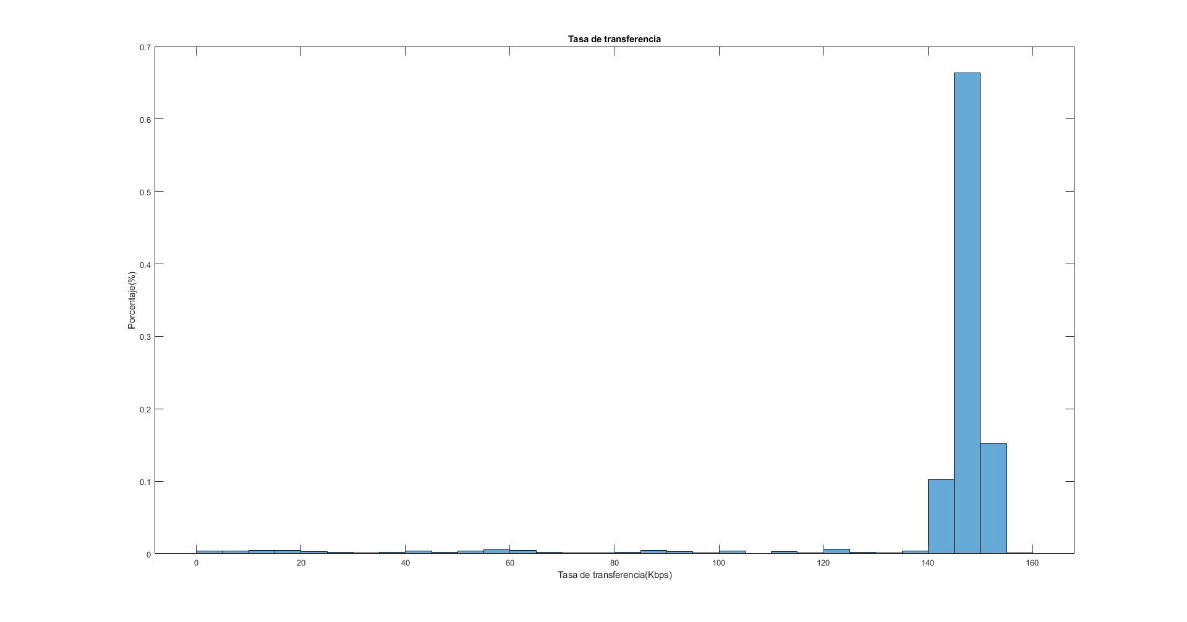
|  |  |  |
| --- | --- | --- |
| **Iteraciones** | **Iteraciones con éxito** | **Porcentaje de éxito [%]** |
| 100 | 88 | 88.00 |

En la tabla 5 se muestra el resultado de hacer fallar al coordinador y que la red seleccione un nuevo coordinador cien veces. La red que se construyó consistió en un coordinador y cuatro nodos conectados. Cuando todos los nodos se conectaron a la red y se dejó correr un tiempo un código de ejemplo en la red se prosiguió a desconectar el coordinador. Los nodos al no recibir el latido del coordinador inicializan el servicio de nuevo coordinador. Un nodo comienza antes que todos y solicita una votación para auto seleccionarse como coordinador, los demás nodos votan y se lleva a cabo la transición al nuevo coordinador. Cuando un nodo no logra hacer la transición al nuevo coordinador comenzará a intentar iniciar el servicio de nuevo coordinador, pero como no existen nodos en la red previa no se llevará a cabo. Luego de tres intentos asume que ya hay una red la cual no está conectado, por lo que ejecuta el servicio de asociación para unirse a la nueva red.

Se toma como acierto cuando un nodo logra tomar el lugar de coordinador y todos los nodos hacen la transición al nuevo coordinador ya sea porque cada uno logró hacer la transición cuando el servicio se ejecutó o porque se asociaron luego de que no estaban asociados a ninguna red.

**Rendimiento de la red**

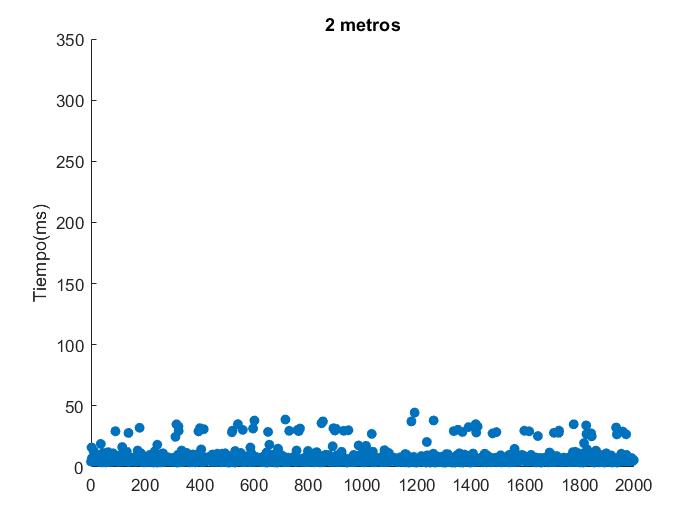
*Tasa de rendimiento efectiva*



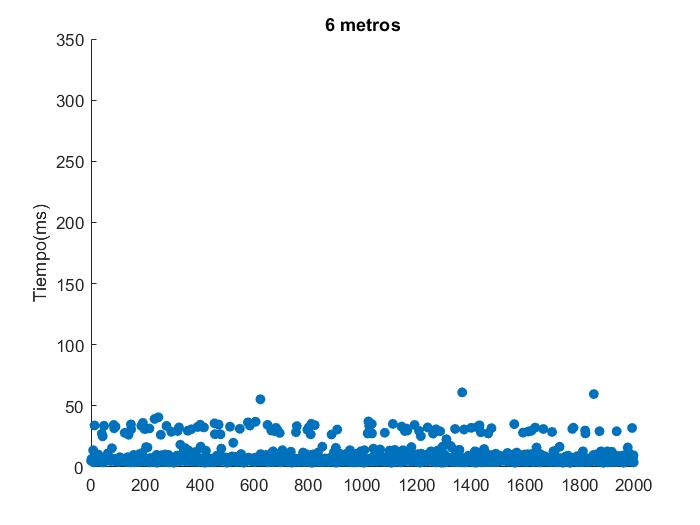
**Imagen 5.** Histograma para la tasa de rendimiento efectiva.

Como se observa en la imagen 5 la velocidad con mayor porcentaje está entre 140 y 160 kbps que se encuentra muy por debajo de los 250 kbps de la teoría. Adicionalmente, esta velocidad equivale al 91.75% con una dispersión baja. La tasa de transferencia efectiva promedio es igual a 141.20 kbps que para aplicaciones de robótica de enjambre es suficiente.

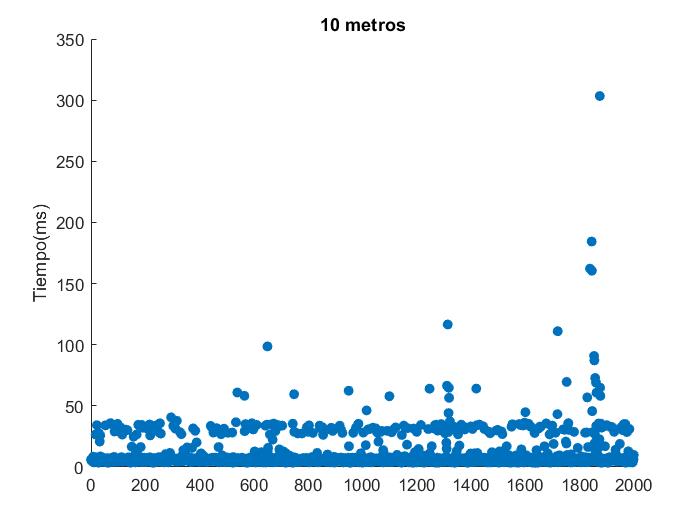
*Alcance y retardo*



**Imagen 6.** Diagrama dispersión con una distancia de 2 metros.



**Imagen 7.** Diagrama dispersión con una distancia de 6 metros.



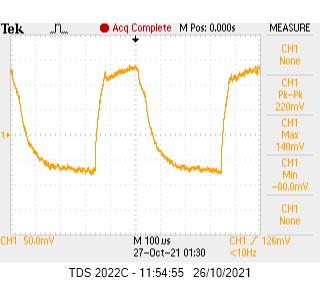
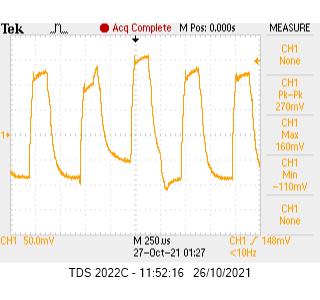
**Imagen 8.** Diagrama dispersión con una distancia de 10 metros.

Adicionalmente, se tiene una mayor pérdida de paquetes conforme la distancia incrementa como se muestra en el siguiente cuadro. Cabe mencionar que para condiciones de escenarios distintos: espacios abiertos, interferencia controlada, distinta antena, entre otros., el desempeño puede mejorar o empeorar.

**Tabla 6.** Comparación de distancias en un espacio cerrado

|  |  |  |
| --- | --- | --- |
| **Distancia [m]** | **RTT promedo [ms]** | **Pérdida de paquetes [%]** |
| 2 | 6.8067 | 2.85 |
| 6 | 7.5057 | 5.05 |
| 10 | 9.7466 | 13.50 |

*Consumo energético*

**(a)**Recepción **(b)**Transmisión

**Imagen 9.** Medición del voltaje pico-pico del transceptor

Las mediciones en el osciloscopio fueron realizadas con un acople AC para analizar únicamente el cambio en el voltaje causado por el transceptor. Utilizando la ecuación se despeja para la corriente y se calcula utilizando la resistencia de 10 Ω y el voltaje pico a pico. El consumo de corriente para la transmisión es 27 mA y el consumo de corriente para la recepción es 22 mA para la recepción. La corriente consumida por el transceptor es consistente con la teoría de Zigbee que indica un consumo de 29 mA. Ya que el transceptor es de bajo consumo tiene la capacidad de ser utilizado en aplicaciones con baterías como fuente de energía.

**Conclusiones**

Con la investigación y los trades studies empleados se encontró la red más viable para aplicaciones de robótica de enjambre. Para la topología, la topología descentralizada presenta una ventaja respecto topologías centralizadas debido a que evitan fallas de punto único y la tecnología de Zigbee es la mejor opción por su bajo costo, bajo consumo y optimización para esta topología.

En el desarrollo de la red y asegurar el cumplimiento de las características de enjambre se emplearon los servicios construidos. Para que la red sea fiable en su funcionamiento es necesario los servicios desarrollados para resolver las posibles fallas en la red como: asociación a una nueva red, sincronización de mensajes y elección de nuevo coordinador.

Con los resultados de las mediciones para el desempeño de la red se tiene que la tasa de transferencia del transceptor es menor a la teórica, sin embargo, para aplicaciones que requieren enviar menos de 100 paquetes por segundo como la robótica de enjambre es suficientemente rápido. Además, el retardo de los paquetes y la tasa de transferencia para espacios cerrados cumplen para aplicaciones que requieren comunicación sencilla como la robótica de enjambre.

**Agradecimiento**

MBAE Pablo Mazariegos por asesorarme y resolverme mis dudas.

Al Dr. Luis Alberto Rivera Estrada por ayudarme a resolver mis dudas.

**Bibliografía**

C. M. Cianci, X. Raemy, J. Pugh y A. Martinoli, (2007) Communication in a Swarm of Miniature Robots: The e-Puck as an Educational Tool for Swarm Robotics, Computer Science, vol. 4433, pág. 115.

CISCO. (2006) Troubleshooting Ethernet Collisions, https://www.cisco.com/c/en/us/support/docs/interfaces-modules/ port - adapters / 12768 - eth - collisions .html, Accessed: 2021-09-14.

AVI Networks. (2021) Single Point Of Failure, https://avinetworks.com/glossary/single-point-of-failure/, Accessed: 2021-09-14.

A. H. Mousa, N. T. Mohammed y E. A. Mohammed. (2019) EFCNT: An evaluation framework for computer’s network topologies. doi: 10.1063/1.5123126.

Israa Faisal Jassam and Shawkat K. Guirguis. (2020) Comparison of network topology types, https://gs.alexu.edu.eg/upload/Students/1404/1404703\_2020-2021\_Fall/1404-3-160\_assignment\_1.pdf, Accessed: 2021-09-14.

J. Soparia y N. Bhatt (2014) A Survey on Comparative Study of Wireless Sensor NetworkTopologies, International Journal of Computer Applications, vol. 87.doi: 10.5120/15175-3255.

A. Shrestha y L. Xin (2017) A Performance Comparison of Different Topologies for WirelessSensor Networks, IEEE Conference on Technologies for Homeland Security.doi: 10.1109/ths.2007.370059.

P.-S. Chong y K. C. Arun. (2021) Performance evaluation of wireless and wired network topologies, Journal of Applied Technology and Innovation, vol 5, no. 3.

DigiKey. (2021) Módulos y módems de transceptores RF, https://www.digikey.com/,Accessed: 2021-09-15.

ESPRESSIF. (2016) ESP-NOW, https://www.espressif.com/en/products/software/esp-now/resources, Accessed: 2021-09-15.

Connectivity standart alliance. (2021) Zigbee FAQ, https://zigbeealliance.org/zigbeefaq/,Accessed: 2021-09-15.

O. Horyachyy. (2017) Comparison of Wireless Communication Technologies used in a Smart Home: Analysis of wireless sensor node based on Arduino in home automation scenario, Thesis for the degree of Master, Blekinge Institute of Technology, Karlskrona,Sweden.

ZWAVE. (2020) Learn about Z-Wave, https://www.z-wave.com/learn, Accessed: 2021-09-15.

J.-r. Lin, T. Talty y O. Tonguz. (2015) On the Potential of Bluetooth Low Energy Technology in Vehicular Applications, IEEE Communications Magazine, vol. 53, págs. 267-275. doi: 10.1109/MCOM.2015.7010544.

ANT (2021) ANT BASICS, https://www.thisisant.com/developer/ant/ant-basics,Accessed: 2021-09-15.

J. P. Dignani. (2011) Análisis del protocolo Zigbee, Tesis de maestría, Universidad Nacional de La Plata, La Plata, Argentina.