

UNA RED DE SENSORES INALÁMBRICA BASADA EN HARDWARE Y SOFTWARE ABIERTO PARA COLABORACIÓN CON APLICACIONES DE ROBÓTICA A MISIONES DE BÚSQUEDA Y RESCATE

F.J. Lara Hurtado, J.J. Fernández Lozano*, R. Vázquez Martín
 Departamento de Ingeniería de Sistemas y Automática. Universidad de Málaga.
 C/ Dr. Ortiz Ramos, s/n, 29071, Málaga.
 jfl@uma.es

Resumen

Las redes de sensores inalámbricas constituyen un complemento en diversas aplicaciones robóticas, como es el caso de las misiones de búsqueda y rescate. En este caso, la capacidad de las redes de sensores inalámbricas para obtener información en la cercanía de la zona de interés, su persistencia, y su capacidad para añadir sensores a vehículos constituyéndolos en nodos móviles, puede aportar a los equipos robóticos información que puede facilitar su misión. No obstante, la integración entre robots y redes de sensores inalámbricas es compleja, y debe prestar especial atención a aspectos como la cobertura, particularmente en el caso de nodos móviles. Este trabajo describe una red de sensores inalámbrica con comunicaciones en malla como alternativa a este problema. La red se basa además en elementos de hardware y software abierto, dando lugar a un sistema de tamaño, peso y coste reducido que facilita contar con un elevado número de nodos. El código y la documentación del sistema presentado en este artículo está disponible en <https://github.com/jfflozano/WSNs>.

Palabras clave: redes de sensores inalámbricas, robots móviles, sistemas ciber-físicos.

1 INTRODUCCIÓN

Las redes de sensores inalámbricos (Wireless Sensor Networks, WSN) permiten obtener datos en aplicaciones en las que es necesario la persistencia o la proximidad al fenómeno observado [1]. Estas características las hacen valiosas para aplicaciones muy diversas, desde la logística a la movilidad [2]. Entre esas aplicaciones se encuentran las misiones de búsqueda y rescate, en las que la combinación de nodos estáticos y móviles puede ser muy útil [3]–[5].

Los robots constituyen un valioso elemento de las operaciones de búsqueda y rescate [6], cuya integración con WSN ha recibido ya atención por

parte de algunos autores [7]. Otro elemento importante en este tipo de misiones son los perros de rescate. Estos perros pueden localizar víctimas en áreas de difícil acceso para los equipos humanos, e incluso para robots, habiéndose propuesto incluso como medio para desplegar sistemas robotizados [8]. Pueden además, y al igual que los robots, portar sensores que complementen la información obtenida de la zona de operaciones, integrándose también en WSN [9].

Estos elementos móviles suponen no obstante un reto en cuanto a su integración con WSN. La efectividad de una WSN depende en gran medida de la cobertura y la conectividad que se consigue con el despliegue de los nodos sensores, así como del tiempo de uso que le permite su consumo energético [10], [11]. La integración de elementos móviles, como robots o perros, supone considerarlos como nodos móviles de una red de sensores. Estas redes pueden adoptar diferentes formas. Por ejemplo, las redes móviles ad hoc (Mobile Ad hoc NETWORKS, MANET) se caracterizan porque cuentan con un conjunto de nodos móviles que se adaptan por sí mismos sin necesidad de configuración adicional, y sin infraestructura fija [5]. En las redes vehiculares ad hoc (Vehicular Ad hoc NETWORKS, VANET), o en general, las redes vehiculares de sensores (Vehicular Sensor Networks, VSN) tanto los vehículos como la infraestructura se consideran nodos de una red dinámica [12]. En general, una red en la que coexisten nodos estáticos y fijos se denomina red de sensores inalámbricos híbrida (Hybrid Wireless Sensor Network, H-WSN) [13]. En todo caso, la existencia de nodos móviles puede hacer difícil garantizar la cobertura [14]. Una alternativa es emplear técnicas que permitan el enrutamiento, como las redes en malla, adaptado además a un tiempo limitado de conexión por la movilidad de los nodos [15], [16].

Este trabajo presenta una red de sensores inalámbrica con comunicaciones en malla desarrollada para facilitar la integración de elementos móviles y estáticos en aplicaciones de búsqueda y rescate. Se basa en el empleo de elementos de hardware y

software abierto, que reducen el coste y facilitan contar con un elevado número de nodos.

La estructura del artículo es la siguiente: tras este apartado de introducción, se describe la arquitectura del sistema. A continuación, se presenta la estructura de los nodos. El siguiente apartado muestra los experimentos realizados para validar el sistema. Por último, se presentan las conclusiones y los trabajos futuros.

2 ARQUITECTURA DEL SISTEMA

2.1 HARDWARE

El protocolo de comunicación usado en la red es ZigBee. Este protocolo se emplea actualmente en diversos campos por su versatilidad y eficiencia energética, y se trata además de un protocolo de comunicación abierto.

Existen muy diversas alternativas comerciales para la implementación de comunicaciones ZigBee. En el caso del sistema descrito en este artículo se han seleccionado módulos RF XBee de Digi. Estos módulos serán los encargados de la creación de nuestra red mediante el protocolo ZigBee.

En el aspecto del hardware se han seleccionado elementos de hardware abierto, como Raspberry Pi y *shield* Arduino para Raspberry Pi de DFRobots. Cuentan además con tamaño reducido, bajo consumo y facilidad de interfaz con diversos componentes hardware, como pueden ser sensores y módulos de radio frecuencia.

Para validar el sistema se han integrado tres tipos distintos de sensores: LDR, LM35 y Groove Sound Sensor.

El sensor LDR es un sensor con capacidad de medir la luminancia del entorno basado en una fotoresistencia que varía la oposición que ejerce frente al paso de corriente dependiendo de la cantidad de luz que este reciba por su cara sensible. Consigue un valor indicativo de la luminancia del entorno siguiendo una función potencial, por lo que es habitual que la relación entre luminancia y resistividad vengan representadas mediante gráficas logarítmicas de modo que aparezcan como una representación lineal.

El sensor LM35 es un sensor capaz de medir variaciones de temperatura en un rango concreto de valores. Consta de circuito integrado capaz de modificar la diferencia de potencial siguiendo una progresión lineal con la temperatura.

El Groove Sound Sensor es un sensor con capacidad para medir los niveles de ruido del entorno. Es uno

de los sensores más utilizados en redes de sensores urbanos. El principal componente de este sensor es un simple micrófono basado en un amplificador LM386 con una salida analógica fácilmente interpretable por Arduino.

La Tabla 1 resume las características generales del hardware empleado. La Figura 1 muestra un nodo del sistema fuera del encapsulado.

Tabla 1: Características generales del hardware.

ELEMENTO	DESCRIPCIÓN
Raspberry Pi Model 3	La Raspberry Pi 3 Modelo B mejora las características de sus predecesoras con un nuevo procesador más rápido integrado para incrementar su velocidad. Cuenta también con funciones de WiFi y Bluetooth BLE para mejorar su funcionalidad y la capacidad de alimentar dispositivos más potentes mediante los puertos USB.
Módulo RF XBee Pro S2	El XBee-PRO OEM es un módulo transceptor RF conforme con IEEE 802.15.4. Funciona en la banda ISM de 2,4 GHz Dispone de una salida de alta potencia y alta sensibilidad de receptor, por lo que puede proporcionar comunicación de largo alcance de línea de visión
Sensor LM35	La serie LM35 se compone de sensores de temperatura de circuito integrado de precisión con una tensión de salida linealmente proporcional a la temperatura en grados centígrados.
Sensor LDR	Célula fotoconductor de sulfuro de cadmio en un encapsulado revestido de epoxi con una respuesta espectral similar a la del ojo humano.
Groove Sound Sensor	Módulo de sensor de sonido basado en el amplificador de potencia LM386 y los micrófonos Electret. El valor de salida se puede ajustar mediante un potenciómetro.
Shield Arduino - Raspberry	Compatible con los modelos Raspberry Pi B+/2B/3B. Añade funcionalidades adicionales al modelo Pi a través de la integración de un Arduino.
Alimentación externa 6000 mAh	Batería externa de 6000 mAh de polímero de litio, con interfaz USB.



Figura 1: Nodo del sistema fuera del encapsulado.

2.2 SOFTWARE

El sistema operativo base del sistema es Raspbian OS. Se basa en la distribución Debian de Linux, optimizado para el hardware de Raspberry Pi. Contiene más de 35.000 paquetes de software precompilado para Raspberry Pi, lo que facilita el desarrollo. El código del sistema se ha desarrollado en Python, junto con Arduino IDE para la gestión de los sensores a través del *shield* Arduino-Raspberry.

Para almacenar la información obtenida gracias a los sensores se emplea MySQL como gestor de bases de datos. Se trata de un gestor sencillo y de software libre.

2.3 COMUNICACIONES

El sistema implementa una red con topología en malla. Para ello, se definen dos tipos de nodo, con una configuración diferente: coordinador, y router. El nodo coordinador deberá realizar las funciones de creación de la red, encargarse de organizar y coordinar los datos transmitidos de forma inalámbrica y de almacenar la información de los sensores de la red en una base de datos. Los nodos router deberán ser capaces de conectarse con la red creada por el nodo coordinador, extraer los datos de los sensores, enrutar los mensajes que circulen por toda la red y transmitir los datos recogidos en el proceso de extracción desde los sensores.

Los nodos coordinador y router comparten el mismo hardware. La diferencia fundamental estriba en la configuración de los módulos XBee. Se establece el firmware apropiado con el protocolo elegido para el uso que se le vaya a dar, puesto que estos

dispositivos pueden funcionar con más de un protocolo de comunicación. En este caso usará ZigBee, por lo que la elección correcta aparece como *ZigBee Coordinator API* para el nodo que hará de coordinador de la red y *ZigBee Router API* para los nodos que harán las veces de routers.

El término API hace referencia al modo en que se comunican los dispositivos XBee entre ellos. Existen dos modos de configuración: API y AT.

El AT es el modo de transmisión de datos más simple, puesto que guarda gran similitud con una transmisión de datos a través de un puerto serie, ya que el dispositivo crea una trama de modo que cuando el dato llega al pin TX del dispositivo este es enviado de forma inalámbrica. La desventaja más destacable de este modo de transmisión es la necesidad de entrar constantemente a la configuración para establecer de nuevo la dirección de destino de los datos

Por otra parte, en el modo API es un microcontrolador externo el encargado de la formación de la trama para el tipo de información que se va a enviar de forma inalámbrica. Este modo de transmisión es especialmente útil para redes grandes y especialmente recomendado para utilizar en redes con topología en malla, como es el caso de este proyecto.

El siguiente parámetro de configuración a cambiar será el PAN ID o identificador de red que, como su propio nombre indica, se encargará de identificar a todos los nodos dentro de la misma red, por lo que todos los dispositivos deberán tener el mismo valor. Los parámetros DH y DL hacen referencia al destinatario de los datos enviados y recibidos por dicho dispositivo y en este caso se han configurado con el valor 0xFFFF para que la comunicación se establezca en modo *broadcast*, que no significa otra cosa que el envío de mensajes se realiza a todos los nodos de la red y no a una dirección concreta. Tanto el XBee que ejerce de coordinador como los que desempeñan el papel de hijos dentro de la red estarán configurados con este parámetro en modo *broadcast* para permitir el enrutamiento de datos, de manera que exista una redundancia en la red y favorezca así la tolerancia a fallos de la red.

Las Tablas 2 y 3 muestran la configuración de los dos tipos de nodo.

La secuencialidad del sistema es la siguiente:

1. El código del Nodo Coordinador va cargado en una placa Raspberry Pi.
2. El código del Coordinador establece la comunicación entre la Raspberry Pi y el *shield*.

3. El *shield* Arduino-Raspberry permite la comunicación directa entre la Raspberry Pi y el módulo RF XBee.
4. Se establecen las comunicaciones entre Coordinador y nodos Router a través de los módulos radio XBee.
5. El *shield* Arduino-Raspberry de los nodos Router se encarga de recibir los mensajes del Coordinador, a través del módulo RF XBee, así como de gestionar la adquisición de datos de los sensores, a través de su procesador Arduino.
6. El código alojado en Raspberry Pi de los nodos Router se encarga de procesar la información traída desde el *shield* Arduino-Raspberry para, volviendo por el mismo camino, llevar dicha información al nodo Coordinador de la red.
7. Por último, el Coordinador recibirá la información de todos los nodos de la red y se encargará de almacenarla en la base de datos.

Tabla 2: Configuración del módulo XBee de los nodos coordinador.

COORDINADOR		
PARÁMETRO	DESCRIPCIÓN	VALOR
PAN ID	Identificador de red	10
SH	Serial Number High	0013A200
SL	Serial Number Low	40C53D49
DH	Destination Address High	0
DL	Destination Address Low	FFFF
NI	Node Identifier	COORD
BD	Baud Rate	9600
AP	API Enable	1

Tabla 3: Configuración del módulo XBee de los nodos router.

ROUTER		
PARÁMETRO	DESCRIPCIÓN	VALOR
PAN ID	Identificador de red	10
SH	Serial Number High	0013A200
SL	Serial Number Low	409EB5AD
DH	Destination Address High	0
DL	Destination Address Low	FFFF
NI	Node Identifier	ROUTER1
BD	Baud Rate	9600
AP	API Enable	1

La Figura 2 muestra el diagrama secuencial de la operación del sistema.

3 VALIDACIÓN DEL SISTEMA

El sistema se ha validado experimentalmente configurando un nodo Coordinador y cuatro nodos Router. Se ha desplegado en laboratorio y se ha verificado mediante la herramienta X-CTU el envío de los paquetes y su enrutado a través de los nodos de la red. La Figura 3 muestra un esquema de la topología adoptada para las pruebas. La leyenda sobre las flechas indica el número de mensajes recibidos respecto del total de mensajes emitidos, para cada par de nodos. Puede verse que se logró una alta proporción de mensajes recibidos, por lo que los resultados obtenidos fueron satisfactorios.

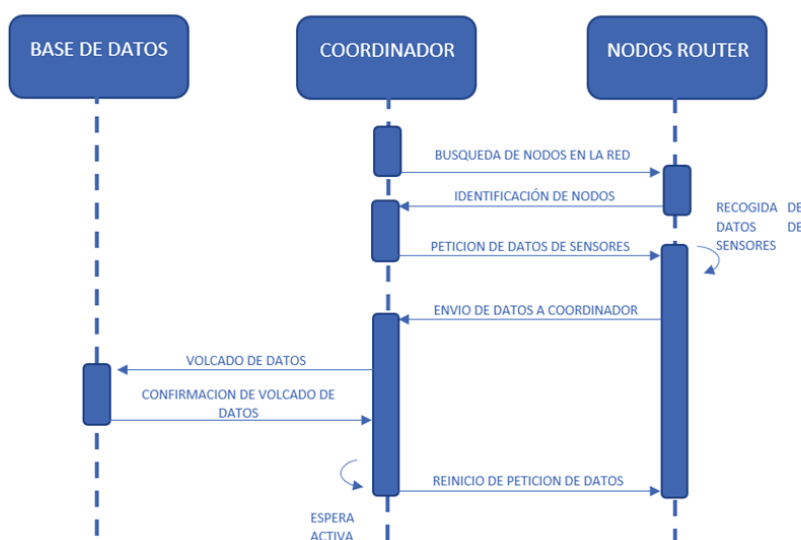


Figura 2: Diagrama secuencial de la operación del sistema.

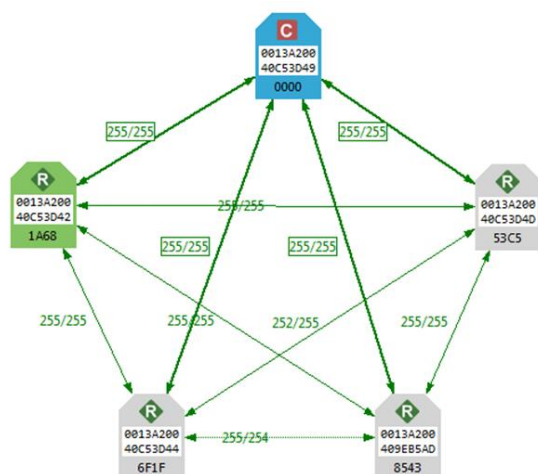


Figura 3: Topología de la red para las pruebas de validación, junto con los mensajes emitidos y recibidos.

4 CONCLUSIONES

Se ha desarrollado una red de sensores inalámbrica basada en elementos de hardware y software abierto. La red cuenta con un único tipo de nodo desde el punto de vista hardware, que puede adoptar el papel de coordinador de la red o de router, mediante la configuración del módulo radio. Los nodos resultantes son de reducido tamaño y peso, y tienen también un coste moderado. Se han realizado pruebas de validación en laboratorio.

Entre las líneas de trabajo futuro está ampliar la validación a pruebas realistas, particularmente en el contexto de aplicaciones robóticas a misiones de búsqueda y rescate. La red desarrollada permite incorporar fácilmente un nodo a cualquier vehículo robótico, ampliando así la información sensorial que puede captarse del entorno de operación. Otras líneas de trabajo son integrar la red con otras basadas en dispositivos comerciales, y ampliar el conjunto de sensores.

El código y la documentación del sistema presentado en este artículo está disponible en <https://github.com/jjflozano/WSNs>.

English summary

A WIRELESS SENSOR NETWORK BASED ON OPEN HARDWARE AND SOFTWARE FOR COLLABORATION WITH ROBOTICS APPLICATIONS IN SEARCH AND RESCUE MISSIONS

Abstract

Wireless sensor networks are a complement to various robotic applications, such as search and rescue missions. In this case, the ability of wireless sensor networks to obtain information close to the area of interest, their persistence, and their ability to add sensors to vehicles by constituting them into mobile nodes, can provide the robotic teams with information that can facilitate their missions. However, the integration between robots and wireless sensor networks is complex. It must pay special attention to aspects such as coverage, particularly in the case of mobile nodes. This paper describes a wireless sensor network with mesh communications as an alternative to this problem. The network is based on elements of open hardware and open software, resulting in a system of reduced size, weight and cost that facilitates increasing the number of available nodes.

Keywords: Wireless sensor networks, mobile robots, cyber-physical systems.

Referencias

- [1] A. Ruano *et al.*, “Wireless Sensors and IoT Platform for Intelligent HVAC Control,” *Appl. Sci.*, vol. 8, no. 3, p. 370, Mar. 2018.
- [2] J. J. Fernández-Lozano, M. Martín-Guzmán, J. Martín-Ávila, and A. García-Cerezo, “A Wireless Sensor Network for Urban Traffic Characterization and Trend Monitoring,” *Sensors (Basel)*, vol. 15, no. 10, pp. 26143–69, Jan. 2015.
- [3] A. Ko, H. Y. K. Lau, and R. P. S. Sham, “Application of Distributed Wireless Sensor Network on Humanitarian Search and Rescue Systems,” *2008 Second Int. Conf. Futur. Gener. Commun. Netw.*, pp. 328–333, 2008.
- [4] J. D. Freeman, V. Omanan, and M. V. Ramesh, “Wireless integrated robots for effective search and guidance of rescue teams,” *8th IEEE IFIP Int. Conf. Wirel. Opt. Commun. Networks, WOCN2011*, pp. 6–10, 2011.
- [5] S. S. Anjum, R. M. Noor, and M. H. Anisi, “Review on MANET Based Communication for Search and Rescue Operations,” *Wirel. Pers. Commun.*, vol. 94, no. 1, pp. 31–52, 2017.
- [6] J. Carlson and R. R. Murphy, “How UGVs physically fail in the field,” *IEEE Trans. Robot.*, vol. 21, no. 3, pp. 423–437, 2005.
- [7] G. Tuna, V. C. Gungor, and K. Gulez, “An

- autonomous wireless sensor network deployment system using mobile robots for human existence detection in case of disasters,” *Ad Hoc Networks*, vol. 13, no. PART A, pp. 54–68, 2014.
- [8] J. Tran, V. Street, V. Street, and D. Ostrom, “A Design for Canine-Delivered Marsupial Robot.”
- [9] J. J. Fernández-Lozano *et al.*, “Integration of a Canine Agent in a Wireless Sensor Network for Information Gathering in Search and Rescue Missions,” in *IEEE/RSJ International Conference on Intelligent and Robotic Systems*, 2018, pp. 5685–5690.
- [10] M. Rout and R. Roy, “Dynamic deployment of randomly deployed mobile sensor nodes in the presence of obstacles,” *Ad Hoc Networks*, vol. 46, pp. 12–22, 2015.
- [11] J. Vales-Alonso, F. J. Parrado-García, P. López-Matencio, J. J. Alcaraz, and F. J. González-Castaño, “On the optimal random deployment of wireless sensor networks in non-homogeneous scenarios,” *Ad Hoc Networks*, vol. 11, no. 3, pp. 846–860, 2013.
- [12] F. Losilla, A.-J. Garcia-Sanchez, F. Garcia-Sanchez, J. Garcia-Haro, and Z. J. Haas, “A Comprehensive Approach to WSN-Based ITS Applications: A Survey,” *Sensors*, vol. 11, no. 11, pp. 10220–10265, Oct. 2011.
- [13] A. Jiménez-González, J. R. Martínez-De Dios, and A. Ollero, “An integrated testbed for cooperative perception with heterogeneous mobile and static sensors.,” *Sensors (Basel)*, vol. 11, no. 12, pp. 11516–43, Jan. 2011.
- [14] S. M. Mohamed, H. S. Hamza, and I. A. Saroit, “Coverage in mobile wireless sensor networks (M-WSN): A survey,” *Computer Communications*, vol. 110. Elsevier, pp. 133–150, 15-Sep-2017.
- [15] F. T. AL-Dhief, N. Sabri, S. Fouad, N. M. A. Latiff, and M. A. A. Albader, “A review of forest fire surveillance technologies: Mobile ad-hoc network routing protocols perspective,” *J. King Saud Univ. - Comput. Inf. Sci.*, 2017.
- [16] T. Darwish, K. Abu Bakar, and A. Hashim, “Green geographical routing in vehicular ad hoc networks: Advances and challenges,” *Comput. Electr. Eng.*, vol. 64, pp. 436–449, 2017.

© 2019 by the authors. Submitted for possible open access publication under the terms and conditions of the Creative Commons Attribution CC BY-NC-SA 4.0 license (<https://creativecommons.org/licenses/by-nc-sa/4.0/deed.es>).

