Future Generation Computer Systems 87 (2018) 242–250

Contents lists available at ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

Comparison of WSN and IoT approaches for a real-time monitoring system of meal distribution trolleys: A case study[☆]

Pedro Morillo [a], Juan M. Orduña [b,*], Marcos Fernández [a], Inmaculada García-Pereira [a]

[a] IRTIC-Universidad de Valencia, Catedrático José Beltrán, 2, 46980-Paterna (Valencia), Spain
[b] Departamento de Informática-Universidad de Valencia, Avda. Universidad, s/n, 46100-Burjassot (Valencia), Spain

HIGHLIGHTS

- WSN and IoT implementations of real-time monitoring system of meal trolleys.
- Performance comparison of both implementations on real facilities.
- IoT implementation yields much lower latencies.
- The latency in the IoT implementation does not depend on the location of trolleys.
- IoT implementation yields better battery life expectancy.

ARTICLE INFO

Article history:
Received 17 July 2017
Received in revised form 22 November 2017
Accepted 16 January 2018
Available online 31 January 2018

Keywords:
Internet of Things
Wireless sensor networks
Hospital food distribution
Food tracking

ABSTRACT

International regulations determine that food in hospitals and elderly homes must be served at given temperature ranges. However, the real-time surveillance of the meal distribution trolleys along all the institutions facilities, guaranteeing conformity to rules from the instant when all the meals are put in the distribution trolley until they are delivered to the patients, is still a challenge.

In this paper, we present a comparison of two approaches based on Wireless Sensor Networks (WSN) and Internet of Things (IoT) technologies for implementing a Real-Time Monitoring System of Meal Distribution Trolleys in a hospital. The performance evaluation results show that the IoT implementation yields much lower latencies than the WSN implementation, and the latency does not depend on the location of the trolley or the building size. Also, the IoT approach yields slightly better battery life expectancy than the WSN approach. Thus, the IoT approach seems the best option for real-time surveillance systems installed in meal distribution trolleys.

1. Introduction

Hospitals play a prominent role in national healthcare systems, and the medical care provided to patients in these facilities are expected to reach the highest quality standard. A nutritious diet is essential for patient treatment and recovery, and therefore food must be safe and served at times that are convenient and appropriate in this kind of facilities [1]. In particular, food-borne pathogens can multiply if food is not maintained at an appropriate temperature or if there are delays between food preparation and distribution in healthcare settings like hospitals or elderly homes [2–4]. Moreover, the patient's perception of the hospital quality is closely related to the quality of the meals distribution system [5,6].

International regulations, like Hazard Analysis of Critical Control Points (HACCP) [7,8], determine that food is at risk in all areas where it is stored, prepared, transported and served, and therefore good food hygiene conditions are necessary. Among them, an important point is to ensure that hot food served is kept above 63 °C and cold food below 5 °C [9]. These regulations have led to a re-definition of technical specifications in the public tenders of western countries hospitals [10] and health institutions [9], as well as a redesign of hospital meal carts [11] and the development of new products [12,13]. However the real-time surveillance of the meal distribution trolleys along all the institutions facilities, guaranteeing conformity to HACCP rules from the instant when all the meals are put in the distribution trolley until they are delivered to the patients, it is still a challenge.

[☆] This work has been supported by Spanish MINECO and EU ERDF programs under grants TIN2015-66972-C5-5-R, TIN2016-81840-REDT, and TIN2016-81850-REDC.

[*] Corresponding author.
E-mail addresses: Pedro.Morillo@uv.es (P. Morillo), juan.orduna@uv.es (J.M. Orduña), Marcos.Fernandez@uv.es (M. Fernández).

https://doi.org/10.1016/j.future.2018.01.032
0167-739X/

In this paper, we present a comparison of two approaches, Wireless Sensor Networks (WSN) and Internet of Things (IoT), for implementing a Real-Time Monitoring System of Meal Distribution Trolleys. The WSN approach consists of *FM-Foodmote*, a low-cost surveillance system based on a WSN for converting non-HACCP compliant distribution trolleys into compliant ones, in such a way that the meal carts inside each trolley are not only real-time monitored, but it also yields the trolley location within the hospital facilities. The problem specification requires that the FM-Foodmote system does not use the hospital Wifi network, it is a self-powered system, and it does not require to screw and/or drill mounting holes in the walls for any network device. These requirements are common in medical environments [14]. The information is displayed on a PC connected to the sensor network. The nodes of the sensor network are based on MEMSIC TelosB Mote Platform [15], and the network should cover the entire facilities. A transmitting mote should be coupled to the distribution trolley, in order to monitor the temperatures of cold and hot meals. The transmitting mote reports the temperatures, while the network reports the location of the trolley, depending on the network nodes receiving the transmitter signal. The IoT approach consists of *Trolleytrack*, a different solution for the same surveillance problem, which is based on self-powered IoT tags and a smartphone, included in the distribution trolley, which is connected to the gateway through 3G network. The main technical contributions of this work (in order of importance) are the following ones:

- We provide a general solution based on IoT for monitoring meal distribution trolleys, which had never been achieved.
- We compare the actual performance of two solutions based on different technologies in a real scenario.
- We provide the reader with real measurements of what can be expected (in terms of battery life expectancy and latency) from implementations carried out in buildings with a similar layout.

The performance evaluation results on real installations show that the WSN approach yields much higher latencies than the IoT approach and a short-life duration of the sensors batteries, thus requiring even a one-week periodical maintenance of the WSN for batteries replacement. A trade-off between latency and battery life expectancy should be achieved in order to reduce maintenance costs. However, the latencies yielded by the IoT approach are much lower and they exclusively depend on Internet. The battery life expectancy of the IoT approach is slightly higher, requiring less frequent maintenance tasks. These results validate the IoT approach as the best option for designing real-time surveillance systems of the meal distribution trolleys. Thus, this work will impact in hospitals everyday life by achieving much more cost-effective and low maintenance monitoring systems for guaranteeing the hospital quality standard.

The rest of the paper is organized as follows: Section 2 introduces the related work. Section 3 describes in detail the FM-Foodmote and the Trolleytrack architectures and components. Next, Section 4 shows the performance evaluation of the FM-Foodmote system on real facilities. Finally, Section 5 presents some concluding remarks and future work to be done.

2. Related work

Wireless Sensor Networks (WSNs) are distributed systems composed by a network of tiny, battery powered sensor nodes with limited on-board processing, storage and radio capabilities [16]. Wireless Sensor Networks (WSNs) have been deployed for a wide range of applications [17], including environment monitoring, smart buildings, medical care, industrial and military applications. In a typical WSN deployment scenario, nodes sense and send their reports toward a processing center using a wireless interconnection network [18]. Most of these networks are based on the IEEE 802.15.4 (ZigBee) standard [19], although the IEEE 802.15.1 (Bluetooth) [20] standard has been used in some cases.

In particular, WSNs have been used in different medical environments and/or applications. They have been proposed for personal health monitoring, being developed and systems for monitoring activities of daily living, fall and movement detection, location tracking, and medication intake monitoring [21–23].

Also, WSNs have been proposed for patient and/or equipment tracking in hospitals. In this field, there are proposals based on RFID Tags [24–26]. However, the short-range of these tags often involves the need for other input devices connected to the computer system to read the monitored information. Additionally, a WSNs based on RFID device can yield location information, but no temperature information. Therefore, the tracking of meal distribution trolleys cannot rely on these devices.

On other hand, the Internet of Things (IoT), denoted also as the Internet of Everything (IoE) or the Industrial Internet (II), is a more recent technology paradigm consisting of a global network of machines and devices capable of interacting with each other [27]. IoT is a more global concept than WSN, and it involves a technology framework where a decentralized network of interconnected objects, all equipped with ‘Intelligent’ decision-making and data-gathering capabilities, exchange information [28]. This technology framework is shaping the evolution of monitoring environments. In this sense, a recent paper presents a two-step framework for designing WSN or IoT systems in manufacturing environments [29]. IoT systems have also been applied to many aspects of healthcare [30]. Moreover, IoT systems are gaining importance as a new technology that allows more efficient and effective hospital management [31].

Although WSNs can be considered as initial IoT-based healthcare research efforts, the ongoing trend is to shift away from registered standards and adopt IP-based sensor networks [32]. The IoT has given rise to many medical applications such as remote health monitoring [33], chronic diseases, and elderly care, including compliance with treatment and medication at home and by healthcare providers [34]. In this field of medical applications, the use of smartphones as the driver of the IoT system has become usual, due to the popularity, wereability, and acceptance of these devices in all environments [35]. We have also explored the feasibility of this approach for the particular problem of meal distribution trolleys. Therefore, in the next sections we show both a WSN and an IoT approach for the implementation of a real-time monitoring System of Meal Distribution Trolleys.

3. WSN and IoT implementations

In this section, we describe the implementations of a low cost, real-time surveillance and monitoring system for the meal distribution trolleys of Hospital General de Vic (Barcelona, Spain). Concretely, we implemented the monitoring system in section of the hospital floors shown in Fig. 1. This hospital section consists of a ground floor and four floors, all of them of rectangular shape.

The upper four floors contain hospital rooms, while the ground floor contains spaces assigned to different hospital services. The upper four floors follow the same layout, with the rooms located at both sides of a central corridor. Fig. 2 illustrate this layout, with the lifts and the stairs are located in one of the rectangle extremes. All the floors are logically divided into sectors. We have denoted a sector as the space unit covered by the range of the WSN motes devoted to detect the location of the trolleys (10–12 m as an average). Thus, Fig. 2 shows that in this case the complete hospital layout can be fully covered with 8 motes, although the number of sectors can be adjusted to fit the building size and/or

Fig. 1. Hospital section where the monitoring system was installed.

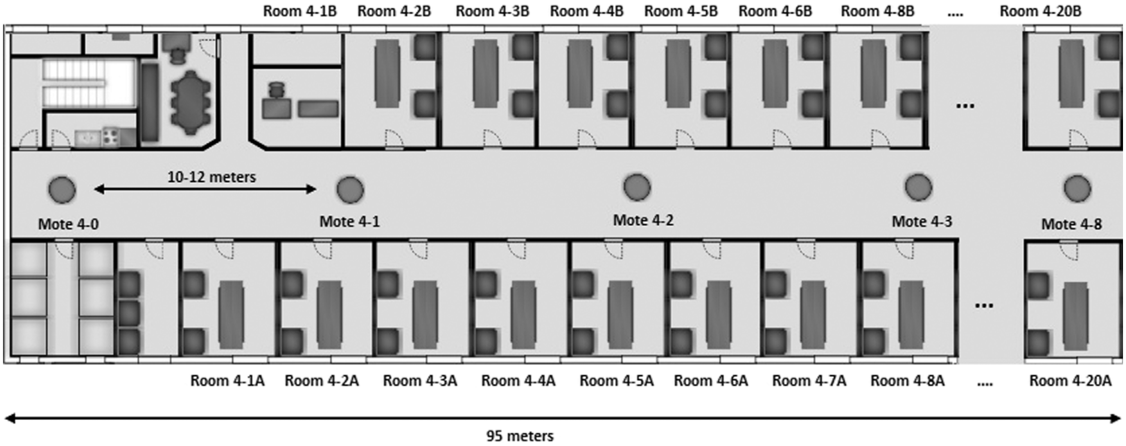


Fig. 2. Hospital layout and distribution of WSN motes.

geometry. It is worth mention that, like in many public tenders of western countries hospitals, the technical specifications of the project highlighted the constraints for the WSN system of not using the hospital Wifi network, being an autonomous, self-powered system, and not requiring to screw and/or drill mounting holes in the hospital walls for any network device.

The ground floor contains the hospital kitchens, where the meals are cooked and packed in meal carts which in turn are stored in the distribution trolleys. Fig. 3 shows an example of the real trolleys used in this implementation. Each cart has two different areas, one for cold and the other one for hot meals, and the temperature on both sides must be monitored, since they must not exceed a temperature of 12 °C for the cold meals and they must keep the temperature above 65 °C for the hot meals. The hospital used Burlodge NovaFlex and B-Pod meal distribution trolleys [36], both of them with air convection feature to keep the temperatures of the meals along the delivery process, and also with a regeneration system which can be connected to the electric power to regenerate the hot and cold section temperatures inside the trolley while they are empty.

The carts in the trolleys should be distributed to each of the patients rooms, and the surveillance system must monitor and register at every moment the location (which floor and floor sector) and the time elapsed from the instant the full trolley left the hospital kitchen, as well as the temperature of the cold and hot section of the trolley until the delivery on each sector. Due to security issues, the main specification constraints are that the surveillance system must not use the hospital Wifi network (in order to avoid interferences with Wifi users), it is a self-powered system, and it should not require to screw and/or drill mounting holes in the walls for any network device.

Fig. 3. One of the distribution trolleys which must be monitored.

3.1. WSN implementation

We implemented a real-time surveillance system, denoted as *FM-Foodmote*, based on WSNs. This system was installed and evaluated in Hospital General de Vic (Barcelona, Spain). In particular, we used the IEEE 802.15.4 compliant MEMSIC TelosB TPR2420 Mote [15], whose image is shown in Fig. 4(a). This mote includes an integrated onboard antenna and integrated temperature sensor, among others devices. TPR2420 is powered by two AA batteries. Using this platform, we built three different kinds of motes, Foodmote N01, Foodmote N02 and Foodmote N00 Gateway. The

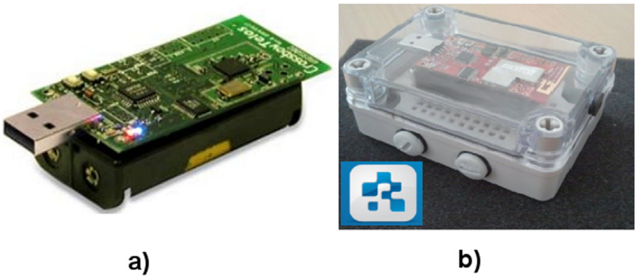


Fig. 4. (a) Telos TPR2420 mote (b) Foodmote N02.

N01 mote included two temperature sensors (ADC Single Delta-Sigma 2KSPS 24-Bit, and Carel NTC Plastic NTC015HP00) , one for the cold and one for the hot section of the trolleys. These motes were mounted each one in one meal distribution trolley, so they moved around the hospital. The N02 motes exclusively were used to transmit the data sent by the former one. These motes were located each one on a given (and known) hospital sector, and their purpose was to locate the position of the N02 mote (and therefore each meal distribution trolley) in real time, as well as to transmit the temperature data collected by N02 motes toward the gateway mote N00. Finally, the mote N00 was directly connected to the PC controlling the system (and therefore this mote was powered by the USB port), and its purpose was to collect the temperature and location from all the trolleys and deliver that information to the PC.

Fig. 4(b) shows an example of the N02 mote, composed of a TPR2420, and external antenna and the batteries. In order to fulfill the constraint of not drilling or screwing the walls, we dropped N02 motes on the existing suspended ceiling in the hospital corridors. On each floor, we used one N02 mote for each of the 7 floor sectors (leaving around 15 m between motes), plus an additional N02 mote for the stairwell, for a total number of 33 N02 motes (8 motes by 4 room floors, plus the mote in the stairwell at the ground floor). The PC and the N00 were located in the hospital basement, close to the stairwell.

The software must show (in the PC monitor display) at every moment the temperature and current state of all N01 motes (each one corresponding to a distribution trolley). The user can see the temperatures list of a given N01 mote, selecting a given time period. The software show real time information, and it can be executed from any Internet browser like Chrome, Firefox or Internet Explorer. The software defines several working hours for lunch and time hours. These working periods will put the N01 Foodmotes in a RUN state. The rest of the time the trolley will be idle. In this case, the software mark the trolley as sleeping. In a RUN period, the N01 mote can be either in a regeneration state (connected to electric power to regenerate the cold and hot temperatures) or in a service state (moving around the floors to deliver the meals). Fig. 5 shows a snapshot of the computer display, where the list of temperatures registered for a given trolley and time period can be seen.

Fig. 6 shows a different snapshot of the PC display, where the software shows the general system status. In this case, a scheme of the building is shown (in this case only 2 sectors per floor appear) including the location of different trolleys with a symbol for each of them. The symbols change from stopped to moving according to the movement of the trolley. Also, each symbol includes four temperatures: real-time hot and cold temperatures in the trolley, and the temperatures (hot and cold) at which last meals have been delivered. Also, Fig. 6 shows on the left the current state for a given cart and a list of both temperatures for different trolleys at a given instant.

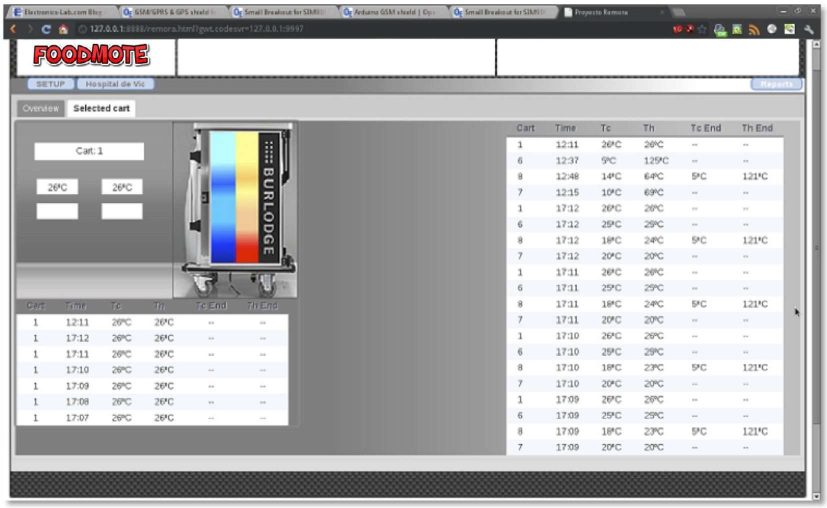
In order to collect the information shown in Figs. 5 and 6, the movement of N01 motes around the hospital corridors is detected by the N02 motes located on each sector. When a N02 mote detects the signal from an N01 mote, it must transmit the temperature data received from the N01, as well as its identification number, toward the N00 mote and the PC through the WSN formed by the rest of N02 motes. That id number allow the software to know indicates in which sector the trolley is located.

Several routing algorithms can be used to transmit the information received from N01 motes to the N00 gateway. For evaluation purposes, we have implemented three of them: flooding routing [37,38], routing based on static neighbors [39], and the Collection Tree Protocol (CTP) [40]. The first algorithm consists of each mote sending each message received to all of its neighbor motes. In order to avoid the unlimited message propagation of messages, we have uniquely identified each mote, and the algorithm does not re-send a message which was previously sent. Also, the motes remove any message which exceeds a maximum hop distance, like the Internet Protocol. Although this algorithm is quite simple and easy to implement, it requires a high power consumption, since it re-sends a lot of messages. The routing based on static neighbors is based on static routing tables. In this routing algorithm, each node always transmits the received messages to the same neighbor node, according to its routing table. The main drawback of this routing technique is that the final destination become unreachable when any of the nodes fails. Therefore, we implemented routing tables with two alternative destination nodes. Finally, the CTP is a dynamic routing algorithm where nodes form a tree structure. The tree root, denoted as coordinator node or base station in some occasions, is established as the closest node to the computer collecting the information acquired by the WSN. In this way, the transmission of the information is carried out from the sensors (motes) to the base station. CTP does not use predetermined addresses, providing the required flexibility when facing network inconsistencies and/or variations in the network size and density. In our implementation, the mote N00 acted as the tree root. CTP implicitly selects the next hop in the path to the root node, and it allows the network reconfiguration in a few seconds after a node fails.

Since the power consumption is a critical issue in WSNs when motes cannot be powered (see for example [41]), we also used the Low Power Listening (LPL) function available in TelosB motes. LPL is a low power communications mechanism included in the B-MAC access control protocol [42]. B-MAC is a medium access control protocol based on carrier detection which provides a flexible interface for ultra-low power operations, yielding effective collision-avoidance mechanisms and a high channel utilization. The reason for implementing this LPL is that the greatest power consumption of the motes occurs when the radio is operating (reaching 17.4 mA for transmitting and 18.9 mA for receiving), being only 1 μA when the radio module is off. When the LPL function is active, a mote switches on its radio only the time needed for detecting the carrier signal in the communication channel. If it detects the carrier signal, then if keeps the radio on while receiving a packet. The *sleep_interval* parameter defines the time interval during which the radio sleeps before awakening to detect the signal carrier. We have added the LPL function when implementing CTP routing.

3.2. IoT implementation

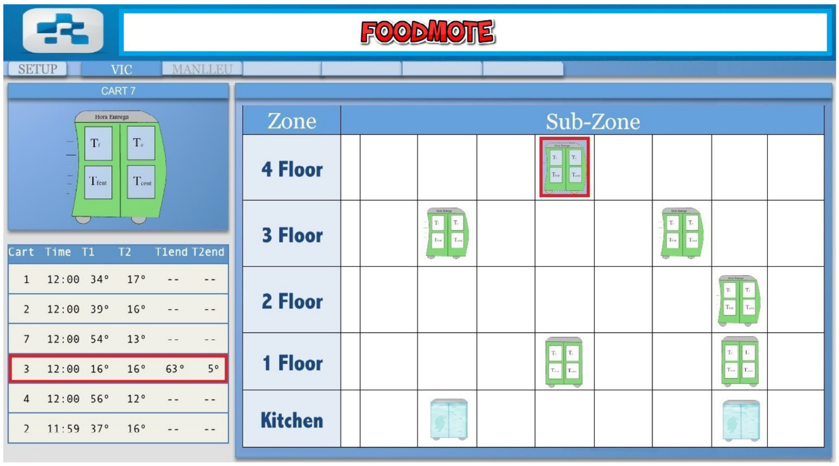
We also implemented the real-time surveillance system based on IoT, and we denoted it as *Trolleytrack*. Trolleytrack is not a simple substitution of the elements making up the *FM-Foodmote* system, but and ad-hoc system with a totally different approach. In particular, the system has been designed using the Texas Instruments BLE SensorTag CC2650 device, usually known as SensorTag [43]. This element is one of the most common elements for the fast prototyping of low-power IoT systems. It is based



Cart	Time	Tc	Th	Tc End	Th End
1	12:11	26°C	26°C	--	--
1	17:12	26°C	26°C	--	--
1	17:11	26°C	26°C	--	--
1	17:10	26°C	26°C	--	--
1	17:09	26°C	26°C	--	--
1	17:08	26°C	26°C	--	--
1	17:07	26°C	26°C	--	--

Cart	Time	Tc	Th	Tc End	Th End
1	12:11	26°C	26°C	--	--
6	12:37	5°C	125°C	--	--
8	12:48	14°C	64°C	5°C	121°C
7	12:15	10°C	69°C	--	--
1	17:12	26°C	26°C	--	--
6	17:12	25°C	25°C	--	--
8	17:12	18°C	24°C	5°C	121°C
7	17:12	20°C	20°C	--	--
1	17:11	26°C	26°C	--	--
6	17:11	25°C	25°C	--	--
8	17:11	18°C	24°C	5°C	121°C
7	17:11	20°C	20°C	--	--
1	17:10	26°C	26°C	--	--
6	17:10	25°C	25°C	--	--
8	17:10	18°C	23°C	5°C	121°C
7	17:10	20°C	20°C	--	--
1	17:09	26°C	26°C	--	--
6	17:09	25°C	25°C	--	--
8	17:09	18°C	23°C	5°C	121°C
7	17:09	20°C	20°C	--	--

Fig. 5. Snapshot of the software displaying real-time information.



Cart	Time	T1	T2	T1end	T2end
1	12:00	34°	17°	--	--
2	12:00	39°	16°	--	--
7	12:00	54°	13°	--	--
3	12:00	16°	16°	63°	5°
4	12:00	56°	12°	--	--
2	11:59	37°	16°	--	--

Fig. 6. Snapshot of the software displaying a general view of the system.

on the ARM® Cortex® M3 32-bit controller, with which it manages the information coming from multiple on-board sensors, like infrared-based temperature sensor, accelerometer, humidity sensor, compass, barometric pressure sensor, gyroscope and microphone. Also, it includes two wireless communications interfaces based on IEEE802.15.4 and Bluetooth Low Power Energy (BLE). Sensortags can be powered by either one 3 V CR032 coin cell battery or a "battery pack" of two AAA batteries connected to special connectors present in the board for that purpose.

We have used Contiki [44] as the operating system for configuring and programming the SensorTag devices. This is a very light operating system specifically designed for embedded systems with very small memory size. Fig. 7 shows a node based on the Texas Instruments BLE SensorTag CC2650.

We have used SensorTag nodes for reading the temperature in the trolleys (we have denoted these nodes as IOT1 nodes) and also for detecting the trolleys position within the hospital (we have denoted the location nodes as IOT2). IOT1 nodes are directly connected to the trolley power source. Unlike Telos B motes, which required additional ADC circuitry, the SensorTag nodes include 4 ADC inputs which can directly read the values yielded by the two Carel NTC Plastic NTC015HP00 passive temperature sensors. IOT2 nodes are the equivalent nodes to N02 motes. In these nodes, we have modified the fabric firmware (version 1.5) to prevent the system from both leaving the "discovery" mode and lighting up the LEDs. This configuration allows a relatively steady and low power consumption (around 0.4 mA), since the only purpose of these nodes is to periodically announce their identifier every second to any device which has its BLE interface active.

The deployment and distribution of SensorTag nodes has been identical to the one for the WSN implementation. Fig. 8 shows an scheme of the IoT implementation. The identifier of each IOT2 node unambiguously the location of the trolley in a given sector. We have developed an application for Android 5.0 (or higher) operating system which can be downloaded to a smartphone included in the trolley itself, or it can be executed on the smartphone of the hospital staff in charge of delivering meals to patients. This applications shows every 60 s the meal temperatures, as well as the location of the trolleys within the hospital (depending on the IOT2 node identifier received). These three values (hot area temperature, cold area temperature and trolley position) are sent to the cloud via the 3G/4G smartphone interface. It must be noted that in this case the routing is provided by the Internet (Internet Protocol), and there is no need to implement routing algorithms from each node to the smartphone, which directly receives the signal from the corresponding IOT2 node. We have used the IBM Watson IoT Platform [45] as the software-based hub on the cloud where we build, collect, process and visualize the smartphone

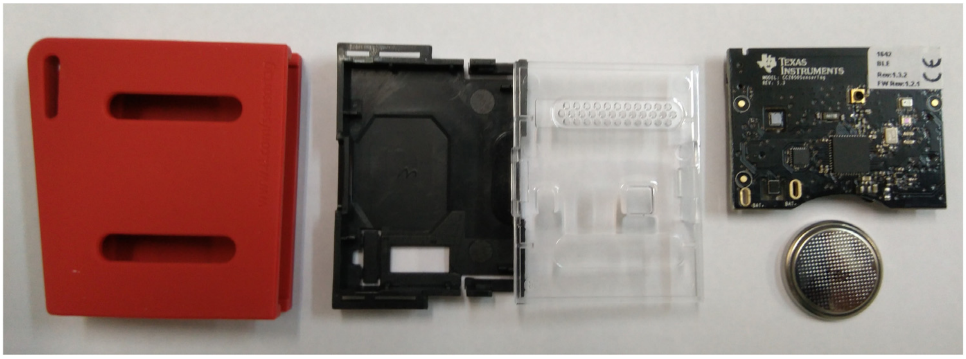


Fig. 7. A node based on the SensorTag CC2650 device.

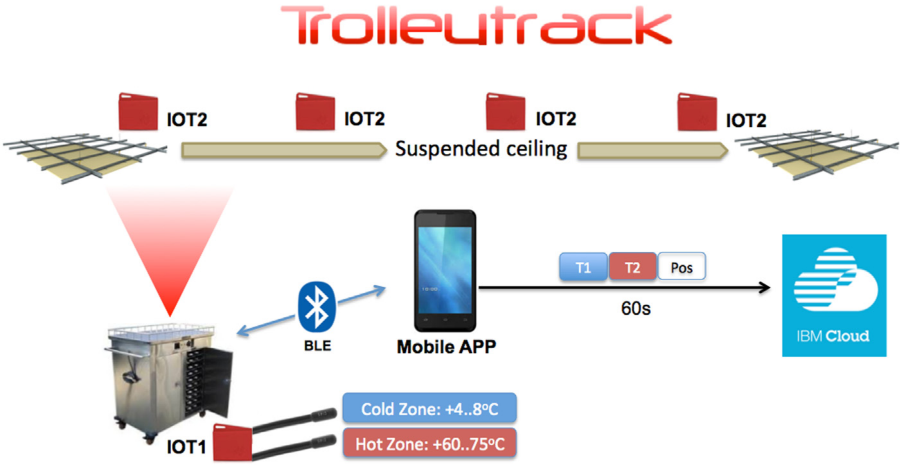


Fig. 8. Scheme of the IoT implementation.

readings. In case the 3G/4G connection is lost (interferences with medical equipment, provider service interruption, location inside lifts and/or basements, etc.) data are temporarily stored in the smartphone with a timestamp, and they are sent to the cloud once the connection is restored.

In order to collect the data to be sent, the smartphone accesses via the Bluetooth interface to the temperatures registered by the IOT1 device. The smartphone permanently remains paired with the Bluetooth IOT device, and the IOT1 device transmit the temperatures every 60 s. However, the smartphone is never paired with the IOT2 devices, since they transmit their identifier every 1000 ms in the "discovery" mode. The smartphone simply stores the higher IOT2 identifier from the ones whose signal is receiving. It must be noted that the sending of data from the smartphone to the cloud and the sending of the temperature values from the IOT1 devices to the smartphone are asynchronously performed.

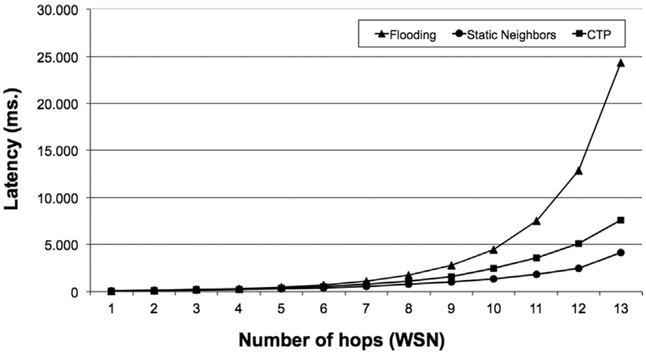


Fig. 9. Latencies achieved with the three routing algorithms.

4. Performance evaluation

In this section, we present the performance evaluation of the two implementations described above. The purpose is to find the real performance and power requirements (in terms of battery life) of each implementation.

4.1. *WSN implementation*

First, we have measured the average packet latency (the time required by the network to transport packets from the receiving N02 mote to mote N00) and the packet reception rate (PRR, the opposite of packet loss rate) achieved by the system when using each of the considered routing techniques (flooding, static neighbors, or CTP), as done in other proposals measuring real WSN deployments [46]. Fig. 9 shows the latencies achieved for different of hops, from the closest one to the most distant mote. This figure shows that significant differences appear for distances longer than 7 hops, and these differences quadratically increase with the number of hops. The shorter latencies are yielded by the static routing algorithm, followed by the CTP algorithm.

Table 1 shows the average latency and the standard deviation for all the latencies shown in each of the plots in Fig. 9. Table 1 shows that the lower latency is achieved by the static routing, followed by the CTP and the flooding algorithms. Since the static routing algorithm is not feasible for real implementations because

Table 1
Average latencies and standard deviation yielded by each routing technique.

	Latency (ms.)	Std. Dev. (ms.)
Flooding	3672.18	2622.18
Static	1010.74	618.42
CTP	1758.92	1221.71

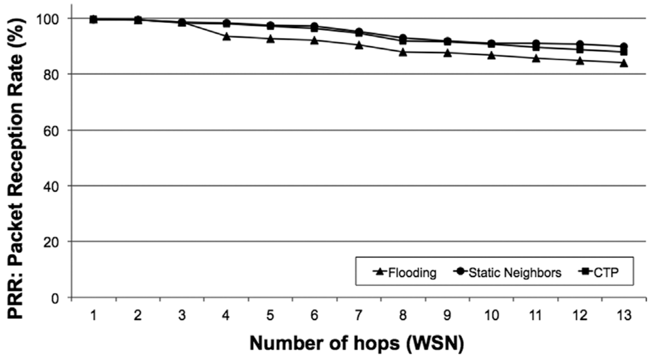


Fig. 10. Packet reception rates achieved by the routing algorithms.

of its zero failure tolerance, the best option seems to be the CTP routing algorithm. Table 1 also shows that all the standard deviation values are very high, reaching 72% of the average latency for the case of flooding. This is due to the fact that the trolleys must sweep all the floor sectors, appearing big differences in the number of hops of the path and therefore in the latencies required for different messages.

Fig. 10 shows the packet reception rates achieved by each routing technique. This figure shows that there are no significant differences between static routing and CTP. However, the flooding algorithm yields slightly lower rates for paths including four hops or more. These results confirm that CTP routing is the best option to provide certain level of fault tolerance and a good level of packet reception rate, at the cost of yielding relatively high latencies for paths longer than 7 hops.

On other hand, it must be noted that power consumption (in terms of battery life expectancy) is a crucial issue that should be addressed in WSNs, in order to avoid expensive maintenance costs for networks in production. Thus, we have measured the batteries life duration for the considered routing techniques. In particular, we measured the average battery life expectancy with alkaline batteries Duracell PreCharged AA 2500mAh (BUN0052A), as well as lithium batteries Energizer Ultimate Lithium L91 3000 mAh. Although the results are not shown here for the sake of brevity, the CTP algorithm yielded a longer battery life expectancy than the flooding algorithm. However, the main result was that the longest battery life did not exceed 10 working days, requiring high maintenance costs (battery replacement operation in all the motes each 7–10 days). In order to face this problem of infeasible maintenance costs, we tested the LPL mode available in TelosB motes. Table 2 shows the batteries life expectancy, measured in working days, for different values of the *sleep_interval* parameter (measured in milliseconds) of TelosB motes. It must be noted that although the system is switched off daily from 14:30 to 17:30 and from 20:20 to 11:00 h, the tests of battery life expectancy were performed keeping the system switched on 24 h a day, in order to test the system in the worst case.

Table 2 shows that the LPL mode adds a huge and effective power saving, extending the battery life expectancy one order of magnitude. However, since an increase in the *sleep_interval* means an increase in the packet latencies, we selected the shortest interval that resulted in long enough periods between batteries replacement. That *sleep_interval* was 512 ms.

Table 2
Average battery life expectancy (days) for CTP routing and LPL mode.

	Alkaline	Lithium
CTP	7 ± 1	10 ± 1
Sleep 256	19	43
Sleep 512	62	143
Sleep 104	135	282
Sleep 2048	204	521

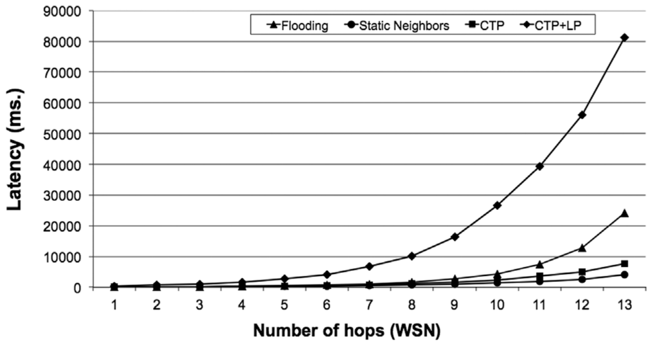


Fig. 11. Latencies achieved with the three routing algorithms and the CTP+LPL mode.

Next, Fig. 11 shows the same latencies shown in Fig. 9, but now including the latencies yielded by the joint use of CTP routing and LPL modes, using the value of 512 ms for the *sleep_interval* parameter. This figure shows that, as it could be expected, the use of the LPL mode adds huge latencies, and these latencies also increase in a quadratic manner with the number of hops in the path. The reason for this behavior is that when motes are in LPL mode, they behave asynchronously, so in each hope the transmitting mote may have to wait until the next mote in the path wakes up, in order to effectively transmit the packet. The average latency value for all the hops are 3672 ms for flooding algorithm, 1010 ms for static routing, 1759 ms for CTP routing, and 18.112 ms for the CTP+LPL option. The huge latency increase in the latter option comes from the huge standard deviation of that routing option, depending on the number of hops in the path.

Finally, Fig. 12 shows the PRR for the CTP routing when the LPL mode, compared to the different routing techniques when LPL is now used. As it could also be expected, the asynchronous behavior of the motes hugely reduces the packet reception rate (increases the packet drop rate), reaching a minimum value of around 35% for the worst case of 13 hops. Nevertheless, these low values can be compensated by increasing the frequency used by the N01 motes to send the packets containing the temperature values. Since these motes are connected to the power supply available in the trolley and they can be charged each time the trolleys are in regeneration mode, increasing the packet frequency does not affect their battery life expectancy.

4.2. IoT implementation

We have measured the latency and power consumption (in terms of battery life expectancy) yielded by the IoT implementation. Regarding latency, we have measured the average latency from the instant when the smartphone transmits the three data (hot and cold temperatures as well as the position) until these data are stored in the cloud and they can be visualized. We computed the average latency value for a complete day of operation (24 h), sending data every 60 s, for a total amount of 1440 packets sent. The result is that the average latency is 426 ms, with an standard deviation of 195 ms. Since in this implementation the routing

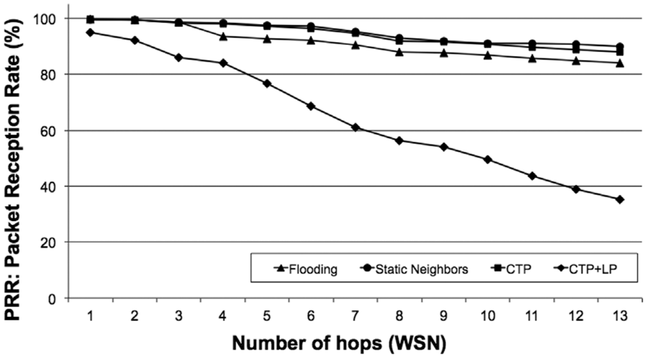


Fig. 12. Packet reception rates achieved by the routing algorithms and the CTP+LPL mode.

Table 3
Average battery life expectancy (days) in IoT implementation.

Interval	2 × L91	CR032
1000 ms	161	23
100 ms	28	4.5

algorithm is provided by Internet, no different routing alternatives can be tested. Comparing this latency value to the ones yielded by the WSN implementation (Fig. 11), we can state that it is less than half of the lowest average latency achieved by the WSN implementation (1010 ms for static routing), and more important, it does not depend on the location of the trolley.

Regarding the power consumption, we measured the average battery life expectancy with Energizer one 3V CR032 Lithium battery and also using a pack of 2x Energizer Ultimate Lithium L91 3000 mAh. Table 3 shows the results for two different identifier announcement intervals.

Table 3 shows that when the IOT2 nodes are powered by a single 3V CR032 lithium battery, the life expectancy is very short. However, if these nodes are powered by the L91 battery pack then the life expectancy spans one order of magnitude. Comparing these values to the ones in Table 2, we can see that the IoT implementation with a 1000 ms interval yields a higher battery life expectancy than the WSN implementation the option of 512 ms interval.

These results validate the IoT implementation as a very efficient alternative to the WSN approach for real-time monitoring of meal distribution trolleys, since the installation and maintenance costs are lower in the case of the IoT implementation and the actual performance (in terms of both latency and battery life expectancy) are better in the case of the IoT approach.

5. Conclusions and future work

In this paper, we have present a comparison of two approaches, Wireless Sensor Networks (WSN) and Internet of Things (IoT), for implementing a Real-Time Monitoring System of Meal Distribution Trolleys in a hospital. The performance evaluation results show that the IoT implementation yields much lower latencies than the WSN implementation, and the latency does not depend on the location of the trolley or the building size. Regarding battery life expectancy, the IoT approach yields slightly better expectancy than the WSN approach. Thus the IoT approach seems the best option for real-time surveillance of the meal distribution trolleys.

As a future work, we plan to improve the functionality and the performance of the IoT-based monitoring system. In this sense, we are dealing with two different challenges involving the implementation of our IoT-based monitoring system in more complex buildings. In particular, we are adapting the IoT monitoring system for being deployed in multiple-building hospitals (like Hospital Universitario La Fé at Valencia, Spain, or the Royal Edinburgh Hospital, UK), where the traceability of the meals path from the kitchen to the patients may include some buildings separated from each other. Also, we are improving the functionality of the App executed on the smartphone, adding an alarm system to warn the user if any of the maximum/minimum threshold temperatures are reached within the trolley cold/hot compartments, respectively. This alarm system will include a software module based on predictive analytics which will estimate the remaining time before reaching these temperature thresholds.

References

[1] B.M. Lund, S.J. O'Brien, Microbiological safety of food in hospitals and other healthcare settings, J. Hosp. Infect. 73 (2) (2009) 109–120. http://dx.doi.org/10.1016/j.jhin.2009.05.017.

[2] H. Réglier-Poupet, C. Parain, R. Beauvais, P. Descamps, H. Gillet, J.L. Peron, P. Berche, A. Ferroni, Evaluation of the quality of hospital food from the kitchen to the patient, J. Hosp. Infect. 59 (2) (2005) 131–137. http://dx.doi.org/10.1016/j.jhin.2004.07.023.

[3] M. Rodriguez, A. Valero, E. Carrasco, F. Prez-Rodrguez, G. Posada, G. Zurera, Hygienic conditions and microbiological status of chilled ready-to-eat products served in Southern Spanish hospitals, Food Control 22 (6) (2011) 874–882. http://dx.doi.org/10.1016/j.foodcont.2010.11.015.

[4] P. Rosset, M. Cornu, V. Nol, E. Morelli, G. Poumeyrol, Time-temperature profiles of chilled ready-to-eat foods in school catering and probabilistic analysis of listeria monocytogenes growth, Int. J. Food Microbiol. 96 (1) (2004) 49–59. http://dx.doi.org/10.1016/j.ijfoodmicro.2004.03.008. http://www.sciencedirect.com/science/article/pii/S0168160504001576.

[5] Z. Stanga, Y. Zurflh, M. Roselli, A. Sterchi, B. Tanner, G. Knecht, Hospital food: a survey of patients perceptions, Clin. Nutr. 22 (3) (2003) 241–246. http://dx.doi.org/10.1016/S0261-5614(02)00205-4.

[6] M. Troutner, M. Gregoire, L. Lafferty, M. Stone, Analysis of temperature of patient meals, J. Foodserv. Manage. Educ. 6 (2) (2012) 1–5.

[7] Hazard analysis and critical control point (HACCP) system and guidelines for its application, 1997. URL http://www.fao.org/docrep/005/y1579e/y1579e03.htm.

[8] Haccp, 2017. URL https://www.food.gov.uk/business-industry/food-hygiene/haccp.

[9] Food hygiene guidelines, 2015, p. 7. URL http://www.eastcheshire.nhs.uk/About-The-Trust/policies/F.

[10] Technical specifications for the tender of the food service of melilla regional hospital, Spanish Ministry of Health and Social Services, 2015, p. 15. URL http://www.ingesa.msssi.gob.es/ciudadanos/licitaciones.do?id=PA+2%2F2015M&metodo=consultaSinForm.

[11] B. Das, J. Wimpee, B. Das, Ergonomics evaluation and redesign of a hospital meal cart, Appl. Ergon. 33 (4) (2002) 309–318. http://dx.doi.org/10.1016/S0003-6870(02)00018-2.

[12] Electrolux meal distribution system, 2017. URL http://tools.professional.electrolux.com/Mirror/Doc/BR/BR_BR-9JEEF_1_5_1_1_9JEEFU.pdf.

[13] Socamel compactserv V2 Trolley, 2017. URL http://www.socameluk.co.uk/no/distribution-of-meal-trays.html.

[14] S.D. Baker, J. Knudsen, D.M. Ahmadi, Security and safety for medical devices and hospitals, Biomed. Instrum. Technol. 47 (3) (2013) 208–211. http://dx.doi.org/10.2345/0899-8205-47.3.208.

[15] Telosb motes, 2017. URL http://www.memsic.com/wireless-sensor-networks/..

[16] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, A survey on sensor networks, Comm. Mag. 40 (8) (2002) 102–114. http://dx.doi.org/10.1109/MCOM.2002.1024422.

[17] I.F. Akyildiz, T. Melodia, K.R. Chowdhury, A survey on wireless multimedia sensor networks, Comput. Netw. 51 (4) (2007) 921–960.

[18] C.S. Raghavendra, K.M. Sivalingam, T. Znati (Eds.), Wireless Sensor Networks, Kluwer Academic Publishers, Norwell, MA, USA, 2004.

[19] D. Gislason, Zigbee Wireless Networking, pap/onl ed., Newnes, Newton, MA, USA, 2008.

[20] B. SIG, Core Specification 5.0, Bluetooth SIG, 2017. https://www.bluetooth.org/en-us/specification/adopted-specifications/.

[21] H. Alemdar, C. Ersoy, Wireless sensor networks for healthcare: A survey, Comput. Netw. 54 (15) (2010) 2688–2710. http://dx.doi.org/10.1016/j.comnet.2010.05.003.

[22] D. He, The ZigBee Wireless Sensor Network in medical care applications, in: 2010 International Conference on Computer, Mechatronics, Control and Electronic Engineering, Vol. 1, 2010, pp. 497–500. doi:10.1109/CMCE.2010.5610435.

[23] M. Fezari, R. Rasras, I.M.M.E. Emary, Ambulatory health monitoring system using wireless sensors node, Procedia Comput. Sci. 65 (2015) 86–94. http://dx.doi.org/10.1016/j.procs.2015.09.082. http://www.sciencedirect.com/science/article/pii/S1877050915029129.

[24] R.S. Sangwan, R.G. Qiu, D. Jessen, Using RFID tags for tracking patients, charts and medical equipment within an integrated health delivery network, in: Proceedings. 2005 IEEE Networking, Sensing and Control, 2005, 2005, pp. 1070–1074. doi:10.1109/ICNSC.2005.1461346.

[25] A.M. Wicks, J.K. Visich, S. Li, Radio frequency identification applications in hospital environments, Hosp. Top. 84 (3) (2006) 3–9. http://dx.doi.org/10.3200/HTPS.84.3.3-9. arXiv:http://dx.doi.org/10.3200/HTPS.84.3.3-9.

[26] P. Najera, J. Lopez, R. Roman, Real-time location and inpatient care systems based on passive {RFID}, J. Netw. Comput. Appl. 34 (3) (2011) 980–989. http://dx.doi.org/10.1016/j.jnca.2010.04.011. {RFID} Technology, Systems, and Applications. http://www.sciencedirect.com/science/article/pii/S1084804510000858.

[27] J. Gubbi, R. Buyya, S. Marusic, M. Palaniswami, Internet of Things (IoT): A vision, architectural elements, and future directions, Future Gener. Comput. Syst. 29 (7) (2013) 1645–1660. https://doi.org/10.1016/j.future.2013.01.010. http://www.sciencedirect.com/science/article/pii/S0167739X13000241.

[28] A. Rayes, S. Salam (Eds.), Internet of Things From Hype to Reality. The Road to Digitization, Springer International Publishing, 2017.

[29] W. Li, S. Kara, Methodology for monitoring manufacturing environment by using Wireless Sensor Networks (WSN) and the Internet of Things (IoT), Procedia CIRP 61 (Supplement C) (2017) 323–328. https://doi.org/10.1016/j.procir.2016.11.182. The 24th CIRP Conference on Life Cycle Engineering. http://www.sciencedirect.com/science/article/pii/S2212827116313427.

[30] D.V. Dimitrov, Medical internet of things and big data in healthcare, Healthc. Inform. Res. 22 (3) (2016) 156–163. http://doi.org/10.4258/hir.2016.22.3.156.

[31] C. Thompson, As Healthcare Costs Rise and Patients Demand Better Care, Hospitals Turn to New Technologies, Tech. Rep., Business Insider Inc., 2016. http://www.businessinsider.com/how-hospitals-are-using-iot-2016-10.

[32] S.M.R. Islam, D. Kwak, M.H. Kabir, M. Hossain, K.S. Kwak, The Internet of Things for health care: A comprehensive survey, IEEE Access 3 (2015) 678–708. http://dx.doi.org/10.1109/ACCESS.2015.2437951.

[33] A. Dohr, R. Modre-Opsrian, M. Drobics, D. Hayn, G. Schreier, The Internet of Things for ambient assisted living, in: 2010 Seventh International Conference on Information Technology: New Generations, 2010, pp. 804–809. doi:10.1109/ITNG.2010.104.

[34] I. Laranjo, J. Macedo, A. Santos, Internet of Things for medication control: E-health architecture and service implementation, Int. J. Reliab. Qual. E-Healthc. 2 (3) (2013) 1–15. http://dx.doi.org/10.4018/ijrqeh.2013070101.

[35] A.S.M. Mosa, I. Yoo, L. Sheets, A systematic review of healthcare applications for smartphones, BMC Med. Inform. Decis. Mak. 12 (1) (2012) 67. http://dx.doi.org/10.1186/1472-6947-12-67.

[36] Burlodge Equipments, 2017. URL http://www.burlodge.com/burlodge-equipment/.

[37] J.N. Al-Karaki, A.E. Kamal, Routing techniques in wireless sensor networks: a survey, IEEE Wirel. Commun. 11 (6) (2004) 6–28. http://dx.doi.org/10.1109/MWC.2004.1368893.

[38] S.-Y. Ni, Y.-C. Tseng, Y.-S. Chen, J.-P. Sheu, The broadcast storm problem in a mobile ad hoc network, in: Proceedings of the 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking, MobiCom '99, ACM, New York, NY, USA, 1999, pp. 151–162. http://dx.doi.org/10.1145/313451.313525. http://doi.acm.org/10.1145/313451.313525.

[39] L. Li, J.Y. Halpern, Minimum-energy mobile wireless networks revisited, in: ICC 2001, IEEE International Conference on Communications, Conference Record (Cat. No.01CH37240), Vol. 1, 2001, pp. 278–283. doi:10.1109/ICC.2001.936317.

[40] O. Gnawali, R. Fonseca, K. Jamieson, D. Moss, P. Levis, Collection tree protocol, in: Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems, SenSys '09, ACM, New York, NY, USA, 2009, pp. 1–14. http://dx.doi.org/10.1145/1644038.1644040. http://doi.acm.org/10.1145/1644038.1644040.

[41] F.G. Montoya, J. Gmez, A. Cama, A. Zapata-Sierra, F. Martnez, J.L.D.L. Cruz, F. Manzano-Agugliaro, A monitoring system for intensive agriculture based on mesh networks and the android system, Comput. Electron. Agric. 99 (2013) 14–20. https://doi.org/10.1016/j.compag.2013.08.028.

[42] J. Polastre, J. Hill, D. Culler, Versatile low power media access for wireless sensor networks, in: Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems, SenSys '04, ACM, New York, NY, USA, 2004, pp. 95–107. http://dx.doi.org/10.1145/1031495.1031508. http://doi.acm.org/10.1145/1031495.1031508.

[43] SimpleLink^{TM} Bluetooth low energy/Multi-standard SensorTag, Texas Instrument, Inc., 2017. URL http://www.ti.com/tool/cc2650stk.

[44] A. Dunkels, B. Gronvall, T. Voigt, Contiki — a lightweight and flexible operating system for tiny networked sensors, in: Proceedings of the 29th Annual IEEE International Conference on Local Computer Networks, IEEE Computer Society, Washington, DC, USA, 2004, pp. 455–462. http://dx.doi.org/10.1109/LCN.2004.38.

[45] R. Nelson, IBM Watson takes to the road, EE Eval. Eng. 55 (7) (2016) 32–33. https://www.evaluationengineering.com/ibm-watson-takes-road.

[46] D. Pediaditakis, A. Boulis, H.N. Pham, From simulation to real deployments in WSN and back, in: 2007 IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks, Vol. 00, 2007, pp. 1–6. doi.ieeecomputersociety.org/10.1109/WOWMOM.2007.4351800.

Pedro Morillo received the M.S. and Ph.D. degrees in computer engineering from the University of Valencia, Spain, with a dissertation on "Improving the Performance in Distributed Virtual Environments". Currently, he is an associate professor in the Department of Informatics, University of Valencia. In this department, he belongs to the Networking and Virtual Environments (GREV) group, where he focuses on the design and development of network architectures for distributed virtual environments. Furthermore, his research interests include distributed virtual environment systems, load balancing, metaheuristics, and cluster computing. He also served as a visiting scientist at Iowa State University, Ames, Iowa, in 2004 and the University of Louisiana, Lafayette, in 2006.

Juan M. Orduña received the M.S. in computer engineering from the Technical University of Valencia, Spain, in 1990. He worked at Telefónica de España, Manpel Electrónica, S.A. and at the Technical University of Valencia as a computer engineeer. He received the Ph.D. in computer engineering from the University of Valencia in 1998. His research has been developed inside the ACCA team (http://www.acca-group.info). Currently, he is a lecturer profesor in the Department of Informatics, at the University of Valencia, SPAIN, where he leads the GREV research group (http://grev.uv.es). He is member of the HiPEAC network of excellence (http://www.hipeac.net/), and his research is currently supported by the Spanish MEC and the European Commission through several projects. Dr. Orduna served as a member of the Program Committee in different conferences and workshops (ICPP, Europar, VR, ICPADS, etc.) as well as a reviewer for scholarly journals like Parallel Computing, IEEE TPDS, JNCA, or FGCS. He has published papers about his research in a number of high impact international journals and conferences.

Marcos Fernández received the M.Sc. degree in physics and the Ph.D. degree in computer science from the University of Valencia. His Ph.D. degree has been oriented to the generation of traffic for driving simulators. He is a lecturer professor in the Department of Informatics at the University of Valencia, Spain. He is also the coordinator of the ARTEC Graphics Group at the Robotics Institute. Currently, his research addresses highly immersive virtual environments. Currently, he is involved in the development of a immersive visualization facility call VISIONARC.

Inmaculada García-Pereira received the degrees in Audiovisual Communication (2008) and Computer engineering (2015) from the University of Valencia, Spain. She is part of the research team of the University Research Institute on Robotics and Information and Communication Technologies (IRTIC) at the University of Valencia. In this institute she belongs to the ARTEC group, dedicated to 3D interactive graphics, virtual reality, augmented reality and civil simulation. Currently, she is developing her doctorate in the field of indoor real-time locating systems and its application to augmented reality.