Bluetooth Mesh Technology for the Joint Monitoring of Indoor Environments and Mobile Device Localization: A Performance Study

Published Version:
Montecchiari L., Trotta A., Bononi L., Di Felice M. (2022). Bluetooth Mesh Technology for the Joint Monitoring of Indoor Environments and Mobile Device Localization: A Performance Study. NY : IEEE [10.1109/CCNC49033.2022.9700518].

This version is available at: https://hdl.handle.net/11585/905846 since: 2022-11-22

Published:

DOI: http://doi.org/10.1109/CCNC49033.2022.9700518

Bluetooth Mesh Technology for the Joint Monitoring of Indoor Environments and Mobile Device Localization: A Performance Study

Leonardo Montecchiari*, Angelo Trotta*, Luciano Bononi*, Marco Di Felice*†,
* Department of Computer Science and Engineering, University of Bologna, Italy
† Advanced Research Center on Electronic Systems "Ercole De Castro", University of Bologna, Italy
Emails: {leonardo.montecchiari, angelo.trotta5, luciano.bononi, marco.difelice3}@unibo.it

***Abstract*—Bluetooth Mesh is a recent SIG standard enabling the deployment of multi-hop Wireless Sensor Networks (WSNs) over Bluetooth Low Energy (BLE) communication links. The standard introduces many novel and interesting features in the Internet of Things (IoT) domain, such as the seamless integration among sensors and mobile and wearable devices, and the support for a wide range of different IoT application profiles. At the same time, fine-grained assessments of the performance are still needed to understand the potential of the technology. In this paper, we investigate the usage of Bluetooth Mesh solutions for the joint monitoring of indoor spaces and humans. Through the deployment of a test-bed, we evaluate the performance of Bluetooth Mesh WSNs under varying traffic loads and network sizes. In addition, by exploiting the short-range, multi-hop communications, we propose a procedure for the indoor localization of mobile devices and evaluate its accuracy. The results demonstrate that the technology supports reasonable delivery ratio under high traffic loads, however the network and localization performance sharply decreases when increasing the number of hops between the source and destination nodes.**

***Index Terms*—Internet of Things, Bluetooth Mesh Networking, Performance Evaluation, Localization**

I. Introduction

Predictive maintenance, Structural Health Monitoring (SHM), precision irrigation are just few examples of applications of the Internet of Things (IoT) enabled by the capability to gather and process context data produced by a sensing infrastructure [1]. The latter can be deployed by using a plethora of Machine to Machine (M2M) communication technologies with different characteristics in terms of bandwidth and range [2]. On the one side, LP-WAN technologies [3] [4] such as LoRa and SigFox have become extremely popular for the coverage of large-scale environments, thanks to their low-power and long-range operations. On the other hand, WPAN technologies such as Zigbee and 6LoWPAN [5] can support the deployment of multi-hop Wireless Sensor Networks (WSNs) in restricted areas, such as single-floor or multi-floor building scenarios.

This paper investigates the performance of the recent Bluetooth Low Energy (BLE) Mesh standard (shortened to BLE Mesh in the following), promoted by the Bluetooth Special Interest Group (SIG), for the deployment of WPAN WSNs [6]. BLE Mesh can not be considered a novel M2M technology, rather a networking solution, since it enables multi-hop communication and routing on top of the legacy BLE stack. This choice introduces several advantages compared to other WPAN solutions. First of all, thanks to proxy mechanisms, legacy BLE devices such as smartphones and wearables can be integrated with the mesh in a seamless way and enabled to exchange data to or from the sensors [7]. In addition, BLE advertisements could be exploited to support device localization and location-aware networking [8]. Finally, the BLE Mesh stack envisages a Model Layer concerning the implementation of nodes' behaviors, messages and states; as a result, several IoT application profiles have been mapped to the standard, and new ones can be defined by programmers by extending the generic models. At the same time, some recent papers raised concerns about the performance of multi-hop BLE Mesh networks, and hence about their capability to support IoT monitoring systems with specific Quality of Service (QoS) requirements. Studies like [9] [10] warn about the number of parameters to tune, which may create complex trade-offs e.g. between the energy efficiency and the end-to-end delay, and about the interference issues on the ISM bands. Similarly, multi-hop deployments can suffer from broadcast problems caused by the controlled flooding mechanism, which simplifies the implementation of the routing strategy, but at the cost of affecting the scalability and throughput [11].

In this paper, we investigate the usage of BLE Mesh technology for the indoor monitoring of spaces and humans. This may be the case for instance of an industrial environment, where there is the need of monitoring the state-of-health of critical equipment (e.g. temperature of a tank) as well as of tracking the access of workers to restricted areas, for security or safety reasons. More specifically, our study aims at providing insights on two research issues related to BLE Mesh networking, i.e.: (*i*) how the technology is able to scale in terms of network size, density and traffic loads and consequently (*ii*) which classes of IoT applications and services can be supported, considering their traffic profiles and QoS requirements. In order to address the key questions above, we consider a three-stage research approach. First, we detail the design of an IoT-based monitoring system based on BLE Mesh; the proposed architecture includes a layered software stack, running on an Edge Processing Unit (EPU), that allows remote device control, device localization, data management and visualization. Second, we describe the system implementation in an indoor, single-floor test-bed composed of BLE Mesh nodes equipped with the ESP-BLE-MESH framework by Espressif [12], and evaluated the system performance under varying traffic loads and network sizes. Third, we discuss the suitability of BLE Mesh for indoor navigation services; to this aim, we propose a network procedure to enable device localization at the EPU device based on the Received Signal Strength (RSS) and trilateration techniques. The results of our study demonstrate that BLE Mesh can support also highly-demanding IoT applications in really small-case topologies; however, the PDR decreases quickly when increasing the number of hops to more than three units. Similar considerations can be drawn also for the real-time localization of mobile devices, which is accurate but only when the target node is close to the sink device. Vice versa, the positioning error increases sharply as a consequence of the poor network performance.

The rest of the paper is structured as follows. Section II reviews the main concepts of the BLE Mesh standard and the enhancements and performance studies proposed in the literature. Section III illustrates the proposed BLE Mesh-based architecture for space and human monitoring. Section IV describes the test-bed implementation and presents the performance results. Section V draws the conclusions and discusses some future works.

II. Related Works

A. BLE Mesh: Overview

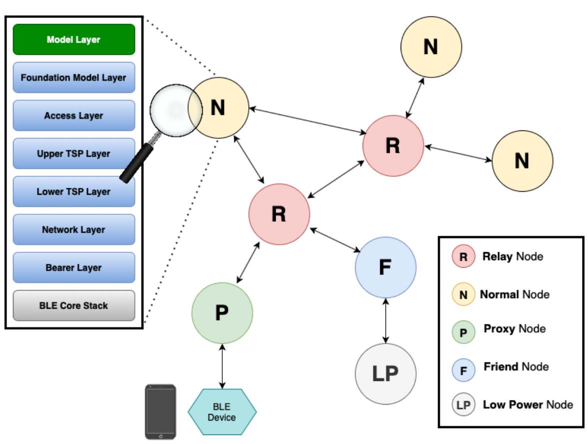


Fig. 1. BLE Mesh topology and layered stack architecture.

BLE Mesh is a networking standard firstly released by the Bluetooth Special Interest Group (SIG) in 2017 [6]. The specifications describe the network architecture and the protocol stack needed to deploy multi-hop WSNs among BLE-based devices, as depicted in Figure 1 (right side). Following the terminology adopted in [6], a BLE Mesh is composed of wireless nodes which can be configured in order to cover different logical roles, i.e.: (*i*) *Relay* nodes, which act as routers by receiving and re-transmitting mesh messages on behalf of other nodes; (*ii*) *Proxy* nodes, which enable the communication towards BLE legacy (non-Mesh) devices, such as wearables or smartphones; (*iii*) *Low Power* nodes, which are battery constrained devices, and hence rarely in active state; (*iv*) *Friend* nodes, which buffer the received messages on behalf of the *Low Power* nodes, and deliver them once the *Low Power* nodes wake-up. In addition, the BLE Mesh network can be initialized by means of a *Provisioner* node, which is in charge of registering new devices and of sharing the security keys as well as of triggering the authentication process. It is important to point out that BLE Mesh is not a M2M communication standard, rather a networking standard that relies on the BLE technology for data transmission over the wireless medium. More in detail, the underlying BLE stack is configured in *connection-less* mode, i.e. all data transmissions are broadcast on the advertisement channels. However, to support the networking operations, the BLE Mesh specifications introduce the layered stack shown in Figure 1 (left side), which stays on top of the traditional BLE Stack. Here, the Model Layer is concerned with the implementation of Models, i.e. default applications with well defined states and sequences of message exchanges: a wide range of Models (e.g. the ON/OFF Model for lightning applications) have been defined by the Bluetooth SIG developers for many existing IoT scenarios. At the other end, the Bearer Layer defines how the mesh messages are handled by the underlying BLE communications system. Two different Bearers are described by the specifications, respectively the ADV Bearer for nodes using non-connectable advertising (e.g. BLE Mesh devices) and the GATT Bearer for the data exchange between Proxy Nodes and legacy BLE devices. The intermediate layers handle different issues of mesh networking, such as data encryption, message segmentation and filtering; a detailed illustration of the capabilities of each layer, as well as of the security mechanisms defined by the standard, can be found in [6].

Finally, we focus on two key mechanisms enabling the message forwarding within the BLE Mesh network, respectively the messaging system model and the forwarding scheme. The messaging system is based on a publish/subscribe paradigm: for each new transmission, the sender node issues a publish action on a specific address/topic. Virtual and group topic/addresses are also allowed: as a result of a publish action on a virtual/group address, all devices having subscribed to that address will receive the message and will consume it. The forwarding scheme is based on a managed flooding mechanism, i.e. all the Relay nodes receiving a message will re-transmit it. However, some basic mechanisms such as the usage of a message cache or of a Time-To-Live (TTL) are employed to limit the number of re-transmissions and hence to mitigate the broadcast storm problem. Still, the scalability on large-scale scenarios is questionable and has been investigated by some recent research works reviewed in the Section below.

B. BLE Mesh: Evaluation studies and Enhancements

The BLE Mesh standard has gained a considerable attention from both academic and industrial research. The existing literature can be classified into: (*i*) proposals of applications or support tools; (*ii*) enhancements to the standard; (*iii*) performance evaluation studies.

Regarding the first category, [13] proposes to apply the BLE Mesh technology to smart homes and more specifically to pervasive door control systems in multi-rooms and multi-floors scenarios. Similarly, there are many proposals for libraries and tools easing the provisioning and remote management of the BLE Mesh network. In [7], the setup of the mesh topology can be done through an Android application. Similarly, the authors of [14] investigate the application of Software Defined Networking (SDN) to control the operations of a BLE Mesh network; the separation between the control and the data plane is justified by the usage of different channels for advertisements and data messages. Through a proof-of-concept testbed, the authors demonstrate that the SDN framework is able to react to congestion situations by self-tuning the parameters of the affected nodes. Security mechanisms and vulnerabilities of BLE Mesh are discussed in [15]: here, a machine-learning based Intrusion Detection System (IDS) is proposed, based on a newly collected training data-set.

Regarding the enhancement to the standards, in [16] the authors address the problem of energy overhead for Friend nodes which must be continuously listening to the medium, and propose a novel asynchronous scanning mechanism. The way to achieve real-time communications over multi-hop BLE networks is explored in [17]; however, the proposed solution employs a connection-oriented, TDMA approach that is not compatible with the standard.

Regarding the performance studies, most of them raised concerns about the managed control flooding mechanism and the optimal tuning of the network parameters, which may incur in the complex trade-offs pointed out in [9]. The performances of BLE Mesh networks are investigated in [10] via simulations; more specifically, the authors investigate the impact of randomization techniques to mitigate broadcast storm problems, and demonstrate the trade-off between the end-to-end delay and the throughput. In addition, they study the performance under increasing network sizes, and show that the WLAN interference in the ISM bands can significantly affect the end-to-end delay and delivery ratio. BLE Mesh test-beds have been described in [18] and [11]. The first study aims to analyze the statistical closeness between the controllable BLE Mesh network parameters (e.g. transmit power and source rate) via network inference techniques. In [11], the authors evaluate the performance of indoor BLE Mesh networks based on the CSRMesh implementation; the packet delivery ratio is used as the main performance indicator and the results confirm the adverse impact of flooding mechanisms in highly dense WSNs. The most similar work to our is [19], where an evaluation of the BLE Mesh technology is conducted within an indoor (office) scenario, also in presence of mobile nodes, and under different traffic loads; the results demonstrate that the technology fails in supporting IoT monitoring applications with high data generation frequency.

III. System Design

Without loss of generality, we consider a multi-room, single-floor indoor scenario like the one depicted in Figure 2. The goal of this study is to investigate the design and deployment of BLE Mesh networking solutions with twofold functionalities: (i) *scenario monitoring*, i.e. the WSN is able to collect environmental data from BLE sensors spread out all over the scenario; (ii) *human monitoring*, i.e. the WSN enables room-level indoor localization of BLE devices moving within the same building. A possible use-case is represented by an industrial environment, where there is a need of monitoring the state-of-health of critical equipment (e.g. the temperature of a tank) as well as of tracking the access of workers to the restricted area, for security or safety reasons. For this purpose, in Section IV, we assess the characteristics of IoT monitoring applications that can be supported by the BLE Mesh, by evaluating the system performance under different traffic loads and node density.

The proposed system includes many hardware/software components. On the hardware side, we installed $N + 1$ BLE devices, $n_0, \ldots, n_N$, forming a connected multi-hop WSN; n_0 is used as data collector, also called Sink node in the following, and is directly connected to an Edge Processing Unit (EPU) through a Serial cable. All the BLE devices, except for the Sink, are equipped with multiple sensors and report their sensing data to the Sink, as further detailed in Section III-A. On the EPU, we deployed the software stack for data processing and analytics. More specifically, the *Parser Module* is in charge of filtering the messages received from the BLE Mesh network via the Sink node. The useful data are extracted and sent to the *Monitoring Module*, which includes two separate software components: (*i*) the *Network Monitor Module* is in charge of computing the network metrics of the BLE Mesh network as well as of extracting the application-specific features from the sensor data (e.g. the average of temperature values); (*ii*) the *Human Monitor Module* is in charge of detecting the presence of mobile devices, and of estimating their current positions according to the procedure detailed in Section III-B. Finally, the *Data Module* allows to store network and application-related metrics and position data in a database, and to visualize them through a Web Dashboard powered by external tools. In the following, we detail the operations of BLE Mesh networking, and of the proposed BLE-based localization technique.

A. BLE Mesh Networking

Let us consider a system setup where all the BLE Mesh devices are configured as *Router* nodes. We do not analyze energy efficiency issues in the paper, hence all the nodes are assumed to be powered by current. Also, all the N nodes (Sink excluded) support the Sensor Model and act as Servers, i.e. they hold a state related to the current sensing values, while the Sink node acts as Client. The data collection process works as follows. We assume that the IoT monitoring system must collect measurements from each sensor at periodic intervals. Let T_p be the interval among consecutive sensing actions.

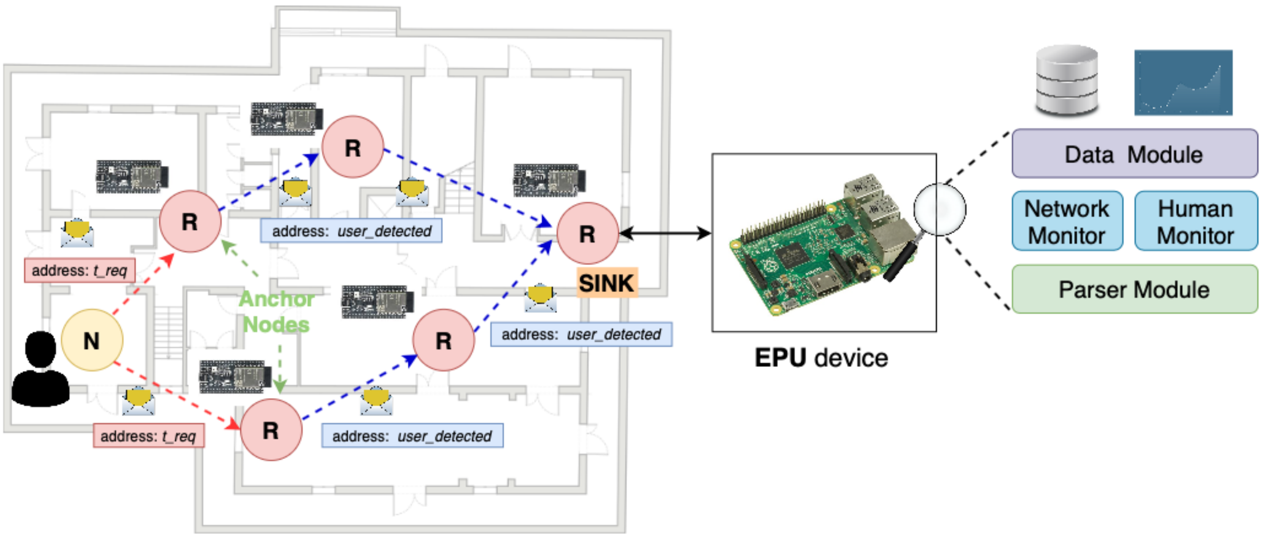


Fig. 2. The BLE Mesh-based monitoring system with the layered software suite installed on the EPU device (on the right) and the sequence messages involved in the localization procedure (on the left).

Every T_p seconds, the EPU triggers the Sink node, which in turns publishes a new `GET` request on the `measure` group address (to which the other N Servers subscribed), and expects to receive N measurements from the other devices. The received messages are then transferred to the EPU via the Serial connection, and here processed for the computation of the network metrics and for the sensor data analytics. The setting of T_p is clearly application-dependant and strongly affects the system performance, as further investigated in Section IV.

B. BLE Mesh Localization

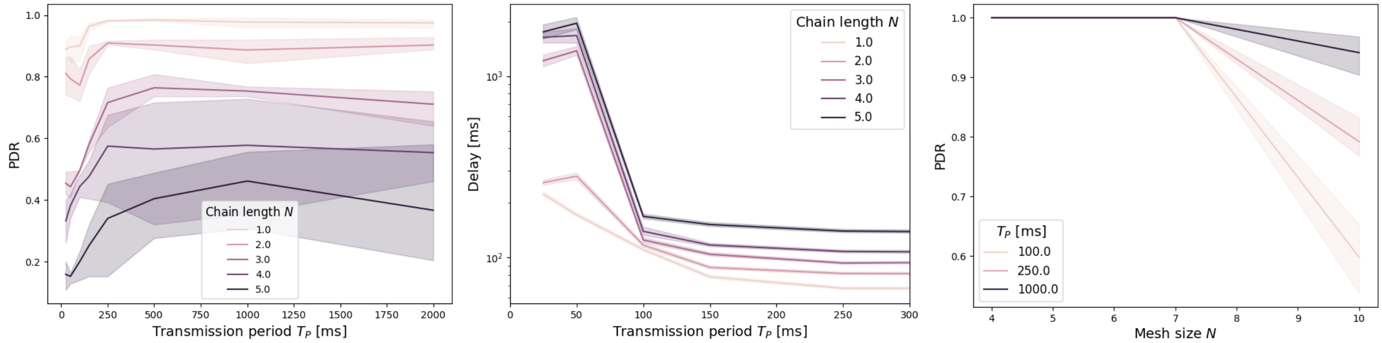
The localization procedure through the BLE Mesh network is illustrated in the left part of Figure 2. We assume that each agent moving within the environment (being a person or a robot) is equipped with a mobile device provided with BLE connectivity and provisioned to operate over the BLE Mesh as a Normal node. Let j indicate its unique unicast address. We omit further details regarding the characteristics of the mobile device. Rather, we focus on its interaction with the nodes composing the BLE Mesh network. More specifically, we set-up the system so that: (*i*) all the $N+1$ Mesh nodes subscribed to the `loc_req` group; (*ii*) the Sink node subscribed to the `user_detected` group. The mobile device periodically advertises its presence, by publishing a message on the `loc_req` channel, with TTL set to 1, every $T_{presence}$ seconds. Let k_j be the unique sequence number of the advertising message sent by the mobile device j. The BLE Mesh nodes located in the transmitting range of the mobile device will detect the message; however, they will not forward it because of the TTL restriction. We denote them as Anchor nodes in Figure 2; since their positions are assumed static and known, the current position of the mobile device can be inferred via trilateration techniques further detailed in Section IV. Let $S(j, k_j)$ be the list of Anchor nodes for mobile device j sending the advertisement with sequence number k_j, where $|S(j, k_j)| \leq N + 1$. Each node $s \in S(j, k_j)$ reports the event to the Sink node by publishing a message on the `user_detected` topic, by adding the following information to the payload: $< ad_s, P(j, k_j), t_c >$ where ad_s is its unicast address, $R(j, k_j)$ is the Received Signal Strength (RSS) of the received message from the mobile node j, and t_c is the current time-stamp. The message is filtered by the Parsing Module of the EPA, and the payload is processed by the Human Monitor Module, which is in charge of estimating P_j, i.e. the current position of mobile node j. We consider two possible formats for P_j, with different spatial granularity levels, i.e.: (*i*) P_j is the room id, hence the localization procedure aims at detecting the current room where the mobile node is located, or (*i*) P_j is a continuous value representing the 2D relative coordinates with respect to a reference point (e.g. the top left angle of the building).

IV. Implementation and Performance Evaluation

In this Section we describe the implementation of the BLE Mesh network and we evaluate the performance over multiple configurations.

We rely on the Espressif ESP32[1] devices (*wroom* and *wrover* versions) to setup the BLE Mesh nodes. Espressif provides an implementation of the BLE Mesh stack that can be deployed and customized via the provided Espressif IoT Development Framework[2]. For the experiments, we uploaded two different profiles on the BLE Mesh devices: (*i*) the Sensor module profile, enabling the device to send sensory data and to exchange control messages, as well as the usage of different topics depending on the experiment, and (*ii*) the Relay module profile enabling the routing and message forwarding capabilities. We

[1] *https://www.espressif.com/en/products/socs/esp32*
[2] *https://github.com/espressif/esp-idf* - Used version v4.3



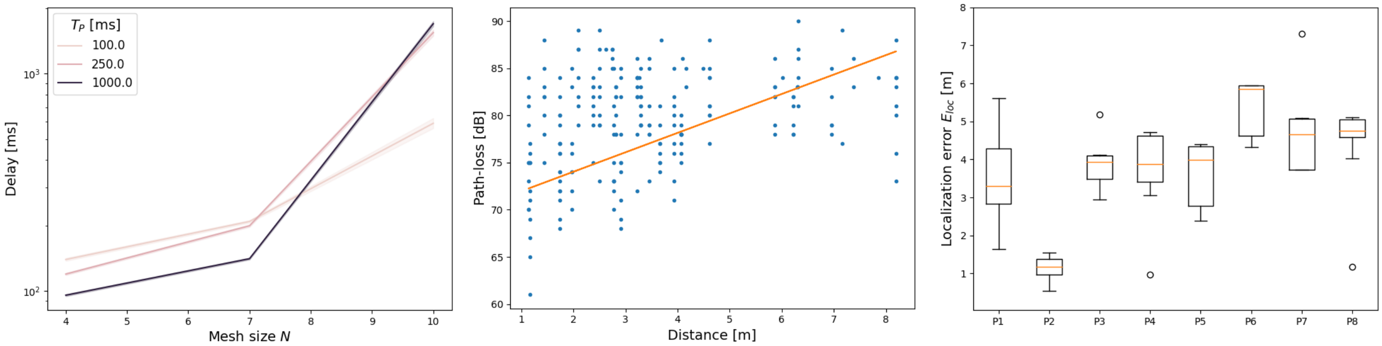
(a) (b) (c)

Fig. 3. The PDR and the end-to-end delay varying T_P and N in the chain topology scenario are shown in Figures 3(a) and 3(b), respectively. The PDR varying T_P and N in the mesh topology scenario is shown in Figure 3(c)

set the transmission power $P_{\text{tx}} = -6\text{dBm}$ and used the default configuration with no packet retransmissions and no message acknowledgements. As depicted in Figure 2, the Sink node n_0 is connected directly to the EPU device. In our deployment, the Sink is constituted by a Raspberry Pi 3B+ executing Nodejs v16.4.0 scripts that implement the software modules (i.e. the Parser Module, the Data Module, the Network Monitor, and the Human Monitor) previously described.

First, we assess the network performance through the metrics computed by the Network Monitor Module, i.e. the Packet Delivery Ratio (PDR) and the end-to-end delay. The former states the reliability of the network, while the latter indicates the crossing speed of data packets through a multi-hop connection. Two different BLE Mesh network deployments have been considered: a chain topology and a multi-path mesh topology. In the chain scenario, the $N+1$ nodes are placed at equal distance in such a way that device n_i is connected with its predecessor and its successor only, n_{i-1} and n_{i+1} respectively, with $1 \leq i < N$, while node n_N is the only sensor that is subscribed to the `measure` topic. In addition, we deployed the Relay profile module on all the devices, the Server Sensor profile only on n_N and the Client Sensor profile on n_0. Despite its simplicity, the topology permits us to analyze the multi-hop capabilities of the BLE Mesh network under self-interference and external interference conditions caused by WLAN devices operating on the same bands and on the same area. The evaluation results related to the chain topology are depicted in Figures 3(a) and 3(b). We varied the length of the chain N (denoted by different curves) and the transmission period T_P (denoted by different values on the x-axis). As expected, the chain length impacts negatively both on the PDR (Figure 3(a)) and on the delay (Figure 3(b)). For $N > 1$, the data packets experience a non-zero probability of packet loss, leading to a PDR value lower than 40% for $N = 5$. When increasing the number of hops, the variance of results becomes relevant due to the unpredictable fluctuations of the channel conditions. Similarly, the end-to-end delay increases accordingly with the chain length N and the related number of forwarding actions. The analysis in terms of the application-dependent variable T_P (on the x-axis) allows us to determine the maximum workloads of the BLE Mesh network. However, we can notice that the T_P parameter impacts the system performance only under high loads (low T_P), i.e. when many packet losses may be experienced due self-interference issues or buffer overflow events at the intermediate nodes. Figure 3(b) shows that the delay is higher than 1 second for $N = 5$ and $T_P \leq 50$ ms.
The second deployment consists of a generic multi-path mesh topology, where the BLE devices are placed randomly within the scenario. Also in this case we analyzed the network metrics by varying the mesh size (N) and the transmission period (T_P). However, differently from the previous experiment, all the devices feature both the Relay and the Server Sensor profiles, while the Sink node features the Client Sensor profile only. All the nodes but the sink subscribe to the `measure` topic. Figures 3(c) and 4(a) show respectively the PDR and delay metrics. Regarding the PDR index, we can notice a significant performance degradation with a mesh of 11 nodes ($N = 10$). It is easy to notice that the T_P parameter becomes effective only when $N \geq 8$, suggesting that the network size, and mainly the flooding mechanism, impacts the system performance more than the network load. Similar considerations can be drawn from the end-to-end delay results shown in Figure 4(a).

Finally, we analyze the performance of the Human Monitor Module. Figure 5 shows the planimetry of the indoor environment used for our localization tests, i.e. a floor hosting the research laboratory at the Department of Computer Science and Engineering of the University of Bologna. We consider $N = 12$ nodes (denoted as red squares and green diamond in the Figure) and $P = 8$ localization points (the blue dots) where the experiments are executed. The mobile user j is provided with a provisioned BLE Mesh node, and walks through predefined fixed points ($P_1, \ldots P_8$ in Figure 5) where the current position is estimated and the localization error is computed. As described in Section III-B, the mobile device sends a message on the `loc_req` topic with $\text{TTL} = 1$, and every BLE Mesh node that receives the message acts as an Anchor node. Then, it computes the RSS value, and forwards



(a) (b) (c)

Fig. 4. The end-to-end delay varying T_P and N in the mesh topology scenario is shown in Figure 4(a). The measurements for the path-loss calibration are shown in Figure 4(b). The localization error for the different reference points $P_0, \dots, P_8$ is shown in Figure 4(c).

it to the Sink node. The Human Monitor Module implements the *Least-Squares Trilateration Algorithm* to estimate P_j from the received RSS measures. It is worth remarking that the aim of this work is not to evaluate the localization algorithm itself, rather to evaluate the ability of the BLE Mesh to support it. Interested readers can refer to [20] for other indoor localization algorithms that could be deployed on top of our testbed. The Trilateration Algorithm estimates the device location assuming the knowledge of distances from known anchor points. In our case, the distance d between the mobile device and the Anchor node n_i is computed from the RSS value by using the log-normal path-loss model: $d = d_0 \cdot 10^{\frac{P_{\text{tx}} - RSSI - PL_0}{10 \cdot \alpha}}$, where Without loss of generality, we consider a multi-room, single-floor indoor scenario like the one depicted$d_0 = 1$ is the close-in reference distance at which the path-loss PL_0 has been measured and α is the path-loss exponent. The values of PL_0 and α have been empirically calibrated as shown in Figure 4(b). The accuracy of the localization process is depicted in Figure 4(c); the points on the x-axis ($P_0, \dots, P_8$) indicate the reference measuring locations depicted in the map of Figure 5. The y-axis shows the localization error E_{loc}, in meters, as the difference between the estimated position computed by the Human Monitor Module and the ground truth. We can notice significant differences in the E_{loc} values between the locations close to the Sink node (e.g. P_2) and the others (e.g. P_6, P_7, P_8). At location P_2, $E_{\text{loc}} \approx 1$m due to the presence of many Anchor nodes and the proximity to the Sink. Vice versa, locations P_6, P_7, P_8 experience higher E_{loc} values because they are placed on the borders of the mesh network. From further investigations, we realized that the poor localization accuracy is again a consequence of the poor network performance; indeed, in Figure 3(a) we show that the PDR sharply decreases with the number of hops. As a result, even if the presence of a mobile node is detected by some Anchor nodes, only few messages belonging to group `user_detected` are reaching the Sink node, hence negatively affecting the calculations of the trilateration algorithm. The average localization error is around 4m, hence this system can be considered suitable only for low-granularity spatial requirements. Indeed, we implemented a room localization module that exploits the knowledge of the floor planimetry for the estimation. Figure 6 depicts the per-room accuracy of the localization process, while the red line indicates the average accuracy within the target building. Similarly to the previous experiment, the accuracy is highly dependant on the user position with respect to the Sink; the accuracy exceeds 80% for rooms close to it, with the exception of room R_3 which is actually a very small corridor. On the other side, the accuracy drops drastically for rooms covered by few Anchor nodes (since the trilateration algorithm lacks of inputs), and for rooms far away from the Sink due to the poor network performance.

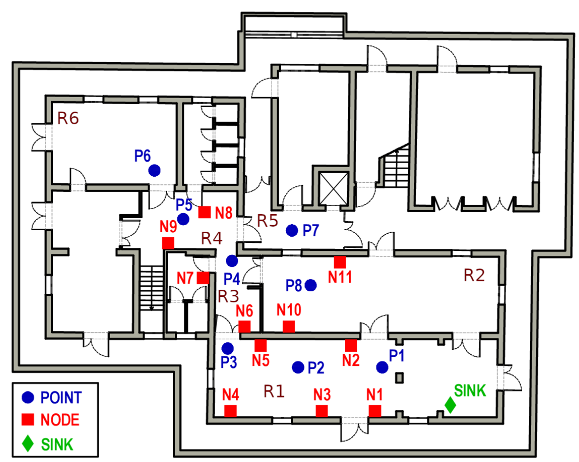


Fig. 5. Scenario map used for the localization deployment where $N = 11$. The red squares and the green diamond represent the devices belonging to the BLE Mesh; the blue points indicate the locations where the localization tests have been carried out.

Based on these results, we can conclude that the proposed BLE Mesh network is not capable of supporting IoT applications with strict delivery requirements, or where the data aggregation/fusion algorithm is computed on the Sink node. Also, high transmission frequency impacts harmfully the system performance, but still less than the network size.

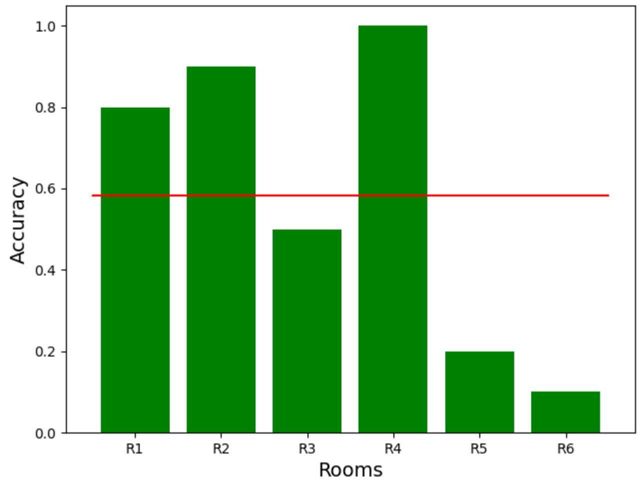


Fig. 6. The room detection accuracy at different rooms of the scenario. The red line indicates the average accuracy within the building.

V. Conclusions and Future Works

In this paper, we described the design, implementation and performance evaluation of an indoor monitoring system based on the BLE Mesh standard. The monitoring system is composed of BLE nodes with multiple sensors (accelerometers, temperature, humidity, etc) in order to support different IoT applications and is currently installed at the Department of Computer Science and Engineering of the University of Bologna. Through such test-bed, we evaluated the network performance in terms of delivery ratio and end-to-end delay for different traffic loads and network sizes. In addition, we investigated the possibility to use the BLE Mesh network to localize mobile devices, and we implemented a network strategy for this purpose. The results demonstrate that the network and localization performance of the BLE Mesh network is harmfully affected by the hop distance more than from the traffic loads generated by the source node. The packet delivery ratio sharply decreases even with three hops and this may clearly impact the operations of many IoT monitoring applications. To tackle the issue, some practical solutions might be to increase the number of sink nodes (hence reducing the average length of the routes) and/or to offload some computational tasks to the edge devices, clearly when such choice does not significantly impact the energy efficiency: in our case, the localization procedure may be implemented by the Anchor nodes rather than by the EPU, by adopting a gossiping procedure among them. Based on the challenges raised by this preliminary study, we are planning to extend the work in many research directions, like for instance: implementing and testing additional localization techniques, analyzing the performance of the BLE Mesh network in presence of mobile, autonomous nodes such as ground rovers or micro Unmanned Aerial Vehicles (UAVs), investigating the joint utilization of multiple wireless technologies on multi-stack nodes like the ESP32 devices in order to increase the delivery ratio on bottleneck links.

References

[1] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of things: A survey on enabling technologies, protocols, and applications," *IEEE Communications Surveys Tutorials*, vol. 17, no. 4, pp. 2347–2376, 2015.

[2] F. Montori, L. Bedogni, M. Di Felice, and L. Bononi, "Machine-to-machine wireless communication technologies for the internet of things: Taxonomy, comparison and open issues," *Pervasive and Mobile Computing*, vol. 50, pp. 56–81, 2018.

[3] L. Sciullo, A. Trotta, and M. D. Felice, "Design and performance evaluation of a lora-based mobile emergency management system (locate)," *Ad Hoc Networks*, vol. 96, p. 101993, 2020.

[4] U. Raza, P. Kulkarni, and M. Sooriyabandara, "Low power wide area networks: An overview," *IEEE Communications Surveys Tutorials*, vol. 19, no. 2, pp. 855–873, 2017.

[5] E. Toscano and L. Lo Bello, "Comparative assessments of ieee 802.15.4/zigbee and 6lowpan for low-power industrial wsns in realistic scenarios," in *2012 9th IEEE International Workshop on Factory Communication Systems*, 2012, pp. 115–124.

[6] "Bluetooth sig. mesh profile specification version 1.0.1, revision date: 2019-01-21."

[7] A. Lacava, G. Nero, P. Locatelli, F. Cuomo, and T. Melodia, "Demo Abstract: BE-Mesh: Bluetooth Low Energy Mesh Networking," *INFOCOM 2019 - IEEE Conference on Computer Communications Workshops, INFOCOM WKSHPS 2019*, pp. 989–990, 2019.

[8] M. Fazio, A. Celesti, and M. Villari, "Improving proximity detection of mesh beacons at the edge for indoor and outdoor navigation," in *IEEE 25th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD)*, 2020, pp. 1–6.

[9] Hernández-Solana, ángela and Pérez-Díaz-De-Cerio, David and García-Lozano, Mario and Bardají, Antonio Valdovinos and Valenzuela, José-Luis, "Bluetooth mesh analysis, issues, and challenges," *IEEE Access*, vol. 8, pp. 53 784–53 800, 2020.

[10] R. Rondon, A. Mahmood, S. Grimaldi, and M. Gidlund, "Understanding the Performance of Bluetooth Mesh: Reliability, Delay, and Scalability Analysis," *IEEE Internet of Things Journal*, vol. 7, pp. 2089–2101, 2020.

[11] P. Zenker, S. Krug, M. Binhack, and J. Seitz, "Evaluation of BLE Mesh capabilities: A case study based on CSRMesh," *International Conference on Ubiquitous and Future Networks, ICUFN*, vol. 2016-Augus, pp. 790–795, 2016.

[12] "Esp-ble-mesh framework." [Online]. Available: https://www.espressif.com/en/products/sdks/esp-idf/esp-ble-mesh

[13] Q. T. Tran, D. D. Tran, D. Doan, and M. S. Nguyen, "An approach of ble mesh network for smart home application," in *2020 International Conference on Advanced Computing and Applications (ACOMP)*, 2020, pp. 170–174.

[14] Y. Murillo, A. Chiumento, B. Reynders, and S. Pollin, "Sdn on ble: Controlling resource constrained mesh networks," in *IEEE International Conference on Communications (ICC)*, 2019, pp. 1–7.

[15] A. Lacava, E. Giacomini, F. D'Alterio, and F. Cuomo, "Intrusion detection system for bluetooth mesh networks: Data gathering and experimental evaluations," in *2021 IEEE International Conference on Pervasive Computing and Communications Workshops and other Affiliated Events (PerCom Workshops)*, 2021, pp. 661–666.

[16] D. Perez-Diaz-De-Cerio, J. L. Valenzuela, M. Garcia-Lozano, A. Hernandez-Solana, and A. Valdovinos, "BMADS: BLE Mesh Asynchronous Dynamic Scanning," *IEEE Internet of Things Journal*, vol. 8, no. 4, pp. 2558–2573, 2021.

[17] L. Leonardi, G. Patti, and L. Lo Bello, "Multi-hop real-time communications over bluetooth low energy industrial wireless mesh networks," *IEEE Access*, vol. 6, pp. 26 505–26 519, 2018.

[18] A. Chiumento, B. Reynders, Y. Murillo, and S. Pollin, "Building a connected BLE mesh: A network inference study," *2018 IEEE Wireless Communications and Networking Conference Workshops, WCNCW 2018*, pp. 296–301, 2018.

[19] E. D. Leon and M. Nabi, "An Experimental Performance Evaluation of Bluetooth Mesh Technology for Monitoring Applications," *IEEE Wireless Communications and Networking Conference, WCNC*, vol. 2020-May, 2020.

[20] S. Traini, L. Sciullo, A. Trotta, and M. Di Felice, "Practical indoor localization via smartphone sensor data fusion techniques: A performance study," in *2019 16th IEEE Annual Consumer Communications Networking Conference (CCNC)*, 2019, pp. 1–7.