

Received June 11, 2020, accepted June 21, 2020, date of publication June 25, 2020, date of current version July 6, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3004745

Building Decentralized Fog Computing-Based Smart Parking Systems: From Deterministic Propagation Modeling to Practical Deployment

MIKEL CELAYA-ECHARRI¹, (Graduate Student Member, IEEE), IVÁN FROIZ-MÍGUEZ^{2,3},
LEYRE AZPILICUETA¹, (Senior Member, IEEE),
PAULA FRAGA-LAMAS^{2,3}, (Senior Member, IEEE), PEIO LOPEZ-ITURRI^{4,5},
FRANCISCO FALCONE^{4,5}, (Senior Member, IEEE),
AND TIAGO M. FERNÁNDEZ-CARAMÉS^{2,3}, (Senior Member, IEEE)

¹School of Engineering and Sciences, Tecnológico de Monterrey, Monterrey 64849, Mexico

²Department of Computer Engineering, Universidade da Coruña, 15071 A Coruña, Spain

³CITIC Research Center, Universidade da Coruña, 15071 A Coruña, Spain

⁴Department of Electrical, Electronic and Communication Engineering, Public University of Navarre, 31006 Pamplona, Spain

⁵Institute of Smart Cities, Public University of Navarre, 31006 Pamplona, Spain

Corresponding author: Leyre Azpilicueta (leyre.azpilicueta@tec.mx)

This work was supported in part by the School of Engineering and Sciences, Tecnológico de Monterrey, in part by the Xunta de Galicia under Grant ED431G2019/01, in part by the Agencia Estatal de Investigación of Spain under Grant TEC2016-75067-C4-1-R, Grant RED2018-102668-T, and Grant PID2019-104958RB-C42, in part by the European Regional Development Fund (ERDF) funds of the European Union (EU) (AEI/FEDER, UE), and in part by the Ministerio de Ciencia, Innovación y Universidades, Gobierno de España (MCI-U/AEI/FEDER,UE) under Grant RTI2018-095499-B-C31.

ABSTRACT The traditional process of finding a vacant parking slot is often inefficient: it increases driving time, traffic congestion, fuel consumption and exhaust emissions. To address such problems, smart parking systems have been proposed to help drivers to find available parking slots faster using latest sensing and communications technologies. However, the deployment of the communications infrastructure of a smart parking is not straightforward due to multiple factors that may affect wireless propagation. Moreover, a smart parking system needs to provide not only accurate information on available spots, but also fast responses while guaranteeing the system availability even in the case of lacking connectivity. This article describes the development of a decentralized low-latency smart parking system: from its conception, design and theoretical simulation, to its empirical validation. Thus, this work first characterizes a real-world scenario and proposes a fog computing and Internet of Things (IoT) based communications architecture to provide smart parking services. Next, a thorough analysis on the wireless channel properties is carried out by means of an in-house developed deterministic 3D-Ray Launching (3D-RL) tool. The obtained results are validated through a real-world measurement campaign and then the communications architecture is implemented by using ZigBee sensor nodes. The implemented architecture also makes use of Bluetooth Low Energy beacons, an Android app, a decentralized database and fog computing gateways, whose performance is evaluated in terms of response latency and processing rate. Results show that the proposed system is able to deliver information to the drivers fast, with no need for relying on remote servers. As a consequence, the presented development methodology and communications evaluation tool can be useful for future smart parking developers, which can determine the optimal locations of the wireless transceivers during the simulation stage and then deploy a system that can provide fast responses and decentralized services.

INDEX TERMS Smart parking, fog computing, ZigBee, BLE, IoT, wireless channel, 3D-Ray Launching, IPFS.

I. INTRODUCTION

The Internet of Things (IoT) enables the development of wireless sensor networks (WSNs) for making transportation

The associate editor coordinating the review of this manuscript and approving it for publication was Ding Xu¹.

safer and more efficient within the context of intelligent transportation systems (ITSs) and smart cities. There is currently intense research devoted to finding effective solutions for reducing traffic congestion, fuel consumption and greenhouse gas emissions while increasing drivers' and

citizens' security [1]. In the context of traffic-congested cities, the improvement of traffic management and urban mobility becomes necessary, so smart parking (SP) systems can be a really useful tool in this regard [2]. Essentially, SP systems collect information about parking availability and, by using some sort of platform, send real-time parking information to potential or subscribed drivers. Thus, travel time for commuters, urban traffic congestion and air pollution are reduced. In addition, pricing and parking reservation can be managed by SP systems [3].

In order to obtain real-time parking availability data in a city, fixed and mobile sensing systems can be deployed [4]. Different kinds of sensors can be installed for the detection of parked vehicles, either in parking space infrastructure (e.g., infrared sensors, ultrasonic sensors, optical sensors) or in vehicles (e.g., laser detectors, sonars or Radio Frequency Identification (RFID) tags) [5]. It is also worth noting that many works on SP systems are focused on sensing technologies, architectural development or on specific mobile apps. However, the performance of such smart management systems depends largely on the sensor capabilities and on real-time wireless data communications. Therefore, although aspects like security should be carefully considered [6], the location of the deployed sensors is essential, since it determines the connectivity and, ultimately, the performance of the SP system. In order to achieve an optimized deployment of WSNs within parking environments, radio channel propagation studies are necessary. Moreover, considering the advent of 5th generation (5G) communications and IoT WSNs, where high-density deployments of wireless transceivers are expected, especially in the smart city context, extensive radio frequency (RF) planning will be required.

Regardless of the selected technology, wireless communication protocols are commonly used to transmit information from the sensor nodes to the platform in charge of the data management. Short-range technologies like Bluetooth, WiFi or ZigBee are often used to implement WSNs for small-scale urban parking areas. In such systems, sensors exchange data through a local network whose data are collected and sent to a gateway. Such a gateway can use a long-range communications system to send the information to a remote platform, where it is stored, analyzed and processed with the aim of providing valuable information to parking managers, to drivers and to the general public.

For large-scale deployments, centralized architectures based on WiFi and cellular networks are commonly used to provide parking information to users [2]. The popularity and ubiquity of smartphones that can connect to existing mobile networks make centralized architectures an attractive solution, since drivers can receive parking information directly on their smartphones via an application. Nowadays, many SP implementations and architectures proposed in the literature are based on IoT technologies, whose collected information is stored in a remote server on a cloud to provide SP services by means of specific mobile apps [7]–[9]. However,

such traditional server-based architectures have two common limitations:

- All data are usually stored in a cloud server that may be down during certain periods of time due to maintenance, hardware/software problems, cyberattacks or congestion derived from excessive incoming sensor traffic.
- There is usually a long physical distance between the sensor nodes and the cloud, so responses are in general not fast when providing information to the drivers or when rapid decision times are required.

To tackle the mentioned problems, different IoT paradigms have recently been proposed to offload cloud computing capabilities and then distribute computational tasks and reduce response latency [10]. Fog computing is one of such paradigms: it offloads the cloud by moving part of the computational power and storage resources to the network edge, to devices located close to the sensor nodes that are able to respond fast to node requests [11], [12]. However, in order to provide a proper quality of service and reduce deployment cost, fog computing systems need to perform a thorough wireless channel characterization, analysis and optimization during their design stage.

This article includes four main contributions aimed at creating a decentralized cost-effective fog computing-based SP system. First, in order to establish the basics, it presents a detailed review of the state of the art of SP wireless channel characterization, the most relevant fog computing-based SP systems and the main communication technologies used in SP. Second, it presents a novel development methodology for a successful and cost-effective SP deployment. Third, as part of such methodology the article thoroughly explains each of the phases that include the design, theoretical simulation, validation, implementation and empirical validation of a fog SP system that is low cost and scalable in terms of protocols and technologies. Finally, performance tests of the decentralized fog computing approach and the proposed decentralized database are presented in Section VI to guarantee proper operation under low-latency conditions.

The rest of this paper is organized as follows. Section II reviews the state of the art on previous fog computing-based SP systems and on the most relevant aspects that impact their development, like wireless channel modeling or the used communications technologies. Section III presents the design of the proposed system, while Section IV validates it by means of an in-house 3D-RL tool. Section V details the system implementation, which is evaluated in terms of response latency in Section VI. Finally, Section VII is devoted to the conclusions.

II. STATE OF THE ART

A. SMART PARKING WIRELESS CHANNEL CHARACTERIZATION

Previous literature has detailed different approaches to analyze communications links for Vehicle-to-everything (V2X) scenarios [13]. In the specific case of parking lots, park-

ing buildings and parking applications, some studies have been presented. Stochastic models have been proposed to study predictable propagation characteristics when applied to different vehicular environments, including outdoor parking lot scenarios. For instance, some of the existing results provide models for short range communications, with maximum transceiver distances of up to 15 m, focusing mainly in Line-of-Sight (LoS) conditions [14]. Wireless propagation characterizations have also been performed for parking buildings by means of upper bound-lower bound models for the 1.8 GHz band [15]. Moreover, a previous work has performed wireless channel characterization in the 433 MHz band in an outdoor parking and whose measurement results were compared with a free space path-loss model and a two-ray path loss model [16]. The obtained empirical results provide a characterization of a specific radial location within the parking.

The impact of obstructions in vehicle-to-vehicle (V2V) communications links for parking lot scenarios has also been analyzed in [17] for Dedicated-short-range-communication (DSRC) within the 5.9 GHz band. In such a work, the Received Signal Strength Indicator (RSSI) values are correlated with packet error rate in order to estimate the quality of service.

In addition, measurement results have been obtained for path-loss estimation and root mean-square delay spread in parking buildings, considering same floor and floor to floor V2V links in the 5 GHz frequency band [18]. Underground tunnel-like parking scenarios have been analyzed in terms of the determination of quasi-static channel conditions in the 5.3 GHz band and for different Multiple-Input Multiple Output (MIMO) configurations related to V2V communication links [19]. Channel impulse response models have been proposed, based on different measurement sets considering motion and motionless conditions, at a center frequency of 5.12 GHz [20]. More recently, in [21] the authors present an empirical based model for indoor and outdoor parking environments.

The previously mentioned works show that, although several theoretical models exist, radio propagation is usually estimated by using empirical propagation models that are based on measurements obtained from specific environments and provide more accurate RF signal propagation estimations when applied to the same specific kind of environments. However, these empirical approaches exhibit disadvantages such as low reusability, low scalability and high cost, in addition to being time-intensive. In this context, this article proposes the use of an in-house deterministic propagation model based on a 3D-Ray Launching (3D-RL) algorithm for RF propagation analysis within SP environments.

B. FOG COMPUTING-BASED SMART PARKING SYSTEMS

Since fog computing was coined by Cisco in 2012 [11], only a few authors proposed SP implementations that are actually based on it [22]–[26]. For instance, in [22] the authors present a fog computing-based SP architecture and suggest different algorithms to optimize parking request allocation so as to

reduce parking cost, fuel consumption and gas emissions. The system makes use of fog gateways that are deployed throughout different parking lots and send parking recommendations to the existing vehicles. Such recommendations are based on multiple factors, like the costs related to waiting, to walking or to keep on driving to find a spot. Another fog computing-based SP system is described in [23]. There the authors show through simulations how fog computing can reduce lag and network usage in comparison to traditional cloud-based deployments.

A similar approach is detailed in [24], but, instead of fog gateways, a relatively powerful computer is used to provide advanced edge computing services (machine learning processing), presenting an architecture that is similar to the ones used with cloudlets [27]. In the mentioned article, the authors focused on improving vehicle position accuracy on the parking lot, which, in the selected experimental scenario, reached 99.1% thanks to the use of machine learning techniques that processed the Received Signal Strength Indicator (RSSI) from Bluetooth Low Energy (BLE) beacons deployed in a parking lot.

Finally, it is worth mentioning the work in [25] and [26]: the former paper describes a theoretical design for an IoT-enabled fog computing-based SP, while the latter proposes a low-cost smart parking system based on Arduino nodes that was devised having the Nigerian market in mind.

As a summary, Table 1 shows the features of the most relevant characteristics of the previously mentioned fog computing based smart parking systems and compares them with the proposed work. As it can be observed in Table 1, in contrast to the proposed work, most of the compared solutions have not been validated in real environments and none modeled the practical scenario with the objective of optimizing the communications coverage or the existing throughput.

C. OTHER USES OF FOG COMPUTING FOR PARKING SYSTEMS

The fog computing paradigm has also been proposed recently by diverse researchers in order to harness its benefits for different parking applications. An example of such proposals is described in [28], where the authors present a vehicular fog computing solution for parking reservation auctions that combines SP features and parked vehicle assistance. The proposed system involves the use of parked vehicles (which act as static network infrastructure) to improve connectivity in areas where roadside units are not available or have poor coverage.

Another interesting application is detailed in [29], which proposes a scheme that combines crowdsourced information and data from fog computing nodes in order to indicate parking availability. Due to the vast amount of information that can be gathered through SP systems, in [30] the authors propose an analytics system based on Hadoop MapReduce. Such a system runs on a cluster of commodity computers that the authors denote as ‘fog computing node’, but whose power seems to be more like a cloudlet.

TABLE 1. Most relevant fog computing based smart parking systems.

Reference	Location	Applications	Communications Interface	Parking Availability and Positioning Sensors	Scenario Propagation Modeling	Empirical (Non-Simulated) Validation	Notes
[22]	Simulations based on a real mall in Xuzhou (China)	Real-time parking slot information	n/a (simulated)	n/a (simulated)	No	No	Smart parking algorithms are evaluated through simulations.
[23]	n/a (simulated scenario)	Real-time parking slot information	WiFi (simulated)	Video cameras (simulated)	No	No	Fog node latency and network usage are estimated with iFogSim simulations.
[24]	Campus parking lot	Real-time parking slot information and vehicle positioning	BLE (iBeacon)	GPS, smartphone sensors (e.g., magnetometer)	No	Yes	System aimed at achieving good positioning accuracy with the help of edge computing.
[25]	n/a (theoretical article)	Real-time parking slot information and reservations	WAVE, ZigBee, Bluetooth, WiFi	RFID, Ultrasound sensors	No	No	Theoretical design article.
[26]	Nigeria (not deployed, just tested)	Real-time parking slot information	GSM	PIR and ultrasound sensors	No	No (only tested, not validated in a real scenario)	Low-cost smart parking system conceived for the Nigerian market.
Our proposal	Campus of Tecnológico de Monterrey (Mexico)	Real-time parking slot information and reservation	Bluetooth Low Energy, ZigBee	Ultrasound sensors	Yes	Yes	Low-cost decentralized proposal including design, implementation, theoretical simulation and empirical validation.

Finally, it is worth mentioning the work presented in [31], which, based on the concepts of fog computing and roadside cloud, proposes a theoretical shared parking model and an association algorithm for finding the optimal parking slot.

D. SMART PARKING COMMUNICATIONS TECHNOLOGIES

In the past years, a plethora of SP developments has been proposed for simplifying and speeding up the parking process [2], [33], [50]–[53]. SP solutions can make use of different sensors to detect parking occupancy, thus embedding magnetometers, light sensors, optical sensors, inductive loop detectors, passive or active infrared sensors, piezoelectric sensors, ultrasound sensors, acoustic sensors or cameras. The most promising solutions use a combination of different sensing devices. The authors of [54] provide a literature review on SP sensors and their main related technologies. Another relevant review is presented in [52], where the authors provide insights about the main communication networks and the most promising SP applications within a smart city context.

Once the information is collected from the environment through parking sensors, it is sent over a radio channel for further processing or storage by making use of communication protocols. Table 2 summarizes the main characteristics of the most popular communications technologies used for SP, indicating their frequency band, usual maximum range, data rate, topology, battery life, power efficiency, scalability, latency, cost and examples of their use in academic developments and commercial solutions. Table 2 includes technologies that can be classified into two main categories: short-range wireless networks and long-range Low-Power Wide-Area Network (LPWAN) technologies. For instance, wireless technologies such as Bluetooth, WiFi and ZigBee have been used for providing short-distance communications.

Most of the short-range wireless developments use a mesh topology to extend their range [52]. However, their energy use and development cost when deploying a large number of devices make them unsuitable for large scale deployments [52], [55]. For such cases, LPWAN technologies are adequate to overcome the limitations of scalability. However, short-range wireless protocols usually provide less latency than LPWAN technologies like SigFox or LoRaWAN when transferring the collected data from sensor nodes to the central server. In addition, it is worth noting that SP applications that use LPWAN protocols are currently limited mainly due to the lack of standardization and limited commercial hardware availability. Nonetheless, as these conditions evolve and the number of vehicles increases (and with them the complexity of the deployments), it is expected that the number of solutions including LPWAN technologies will increase and, in fact, for large-scale SP deployments with low latency requirements, NB-IoT and LTE-M are currently among the best options at the cost of paying fees to telephone companies. Future 5G deployments are also expected to provide communication options, with specific V2X capabilities under consideration.

The most promising academic SP solutions in terms of cost-effectiveness and low-power detection are hybrid or multi-technology solutions. A recent example is described in [33], where vehicle presence detectors switch between different technologies depending on the context requirements: one alternative uses LoRa with a battery while the other option consists of a solar cell powered UHF RFID device. Energy consumption is also measured (together with transmission time) but for ZigBee, Bluetooth and WiFi in [56]. In such paper the authors conclude that ZigBee is the best choice in terms of scalability in a mesh network and sensor lifetime.

TABLE 2. Main characteristics of the most relevant communications technologies for smart parkings.

Technology	Standard	Frequency Band	Maximum Range	Data rate	Topology	Battery life	Power efficiency	Scalability	Latency	Cost	Examples of Relevant Academic Developments & Commercial Solutions
RFID	-	HF 3–30 MHz (13.56 MHz), LF 30–300 KHz (125 KHz), UHF 30 MHz–3 GHz	from cm (LF) to tens of meters (UHF)	<640 kbit/s	-	Long durability	-	-	-	Low	[25], [32]–[36]
NB-IoT	3GPP	LTE in-band, guard-band (700-900 MHz)	<15 km	<250 kbit/s	Star	10 years	Very high	Yes	<10 s	High	[37], [38]
LTE-M	3GPP	LTE in-band, guard-band (700-900 MHz)	100 km (LTE)	<1 Mbit/s	Star	10+ years	Medium	Yes	10-15 ms	High	[39]
LoRa, LoRaWAN	LoRa Alliance	2.4 GHz	2-5 km (urban), 15 km (suburban), 45 km (rural)	0.25–50 kbit/s	Star on star	8-10 years	Very high	Yes	1-2 s	Low	[33]
SigFox	Sigfox	868-902 MHz	3-10 km (urban), 30-50 km (rural)	100 kbit/s	Star	8-10 years (140 12-byte messages per day)	Very high	Yes	1-30 ms	Medium	[40]
Weightless W/N/P	Weightless SIG	License-exempt sub-GHz	up to 15 Km	1 Kbit/s – 10 Mbit/s (W), 30– 100 Kbit/s (N), 200 bit/s – 100 Kbit/s (P)	Star	<10 years	Very high	Yes, 10-byte messages(W/P) and up to 20-byte (N)	Low	Low	[41], [42]
Ingenu	Ingenu, IEEE 802.15.4k	2.4 GHz	15 km (urban), 80 km (rural)	624 Kbit/s (DL), 156 Kbit/s (UL)	Star, tree	>10 years	Very high	Yes	10-20 s	Medium	[43]
Wi-Fi	IEEE 802.11b/g/n/ac	2.4–5 GHz	<250 m	11 Mbit/s (b), 54 Mbit/s (g), 1 Gbit/s(n/ac)	Point to hub	Days, up to 1 year (AA battery)	Medium	Limited	<50 ms	Low	[44]
Bluetooth, BLE	IEEE 802.15.1	2.4 GHz	50-100 m	<24 Mbit/s (v. 4)	Point-to-point, star, mesh	Years on coin cell battery	Very high	Limited	3 ms	Low	[45]
ZigBee	ZigBee Alliance	868-915 MHz, 2.4 GHz	<100 m	20–250 kbit/s	Star, mesh, cluster tree	From months to years	Very high	Yes (up to 65,536 nodes)	15 ms	Low	[36], [46]–[49]

Finally, it must be noted that, despite the existence of the previously mentioned works, most of the academic developments available in the literature are either outdated or present prototypes in their early stages. In the case of commercial solutions, manufacturers guarantee years of battery lifetime, but most of their hardware devices are relatively power-hungry and, in some occasions, rather expensive, considering the features they provide.

III. DESIGN OF THE SYSTEM

A. METHODOLOGY

This work proposes a novel development methodology to build and deploy SP systems successfully and in a cost-effective manner. The essential steps of the methodology are presented in Figure 1, which consist of the following main stages:

- 1) Design. In this first stage the parking lot is characterized considering aspects like the communications distance to be covered or the existing obstacles. Then, the scenario characteristics are used together with the rest of the desired requirements (e.g., response latency, bit-rate, supported services) for defining the SP communications architecture.
- 2) Validation. Before carrying out an expensive deployment that depends heavily on the deployment environment characteristics, the scenario is first modelled accurately in 3D and then simulations are carried out by using an advanced 3D-RL tool. The results of such a tool are then analyzed and the different

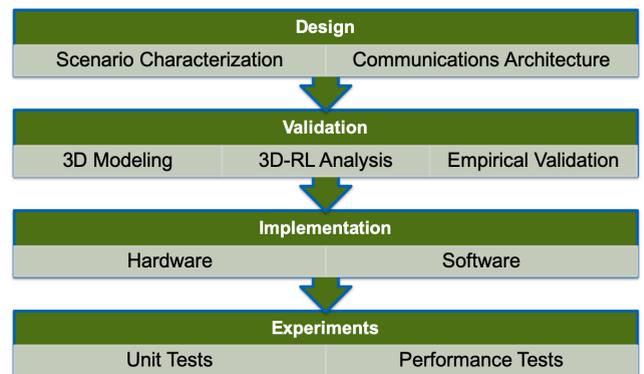


FIGURE 1. Smart parking development methodology.

- 3) Implementation. Once the designed system is validated, a prototype testbed is implemented with the required hardware and software components.
- 4) Experiments. In this stage the different components of the developed system are first tested individually and then jointly in order to guarantee proper operation. In addition, the performance of the system is evaluated under different conditions of computational load and traffic.

B. SCENARIO CHARACTERIZATION

In order to illustrate the proposed methodology with a practical real-world example and get insights into the main RF channel propagation phenomena encountered in a typical outdoor parking scenario, an outdoor urban parking scenario located at Tecnológico de Monterrey, Monterrey Campus, Mexico was selected. Figure 2a shows an aerial view of the scenario, where numerous obstacles and vegetation can be observed together with a high concentration of vehicles. Such an amount of vehicles derives into a high density of metallic elements, which leads to a complex channel propagation environment that needs to be fully understood before the deployment of the wireless communications system.

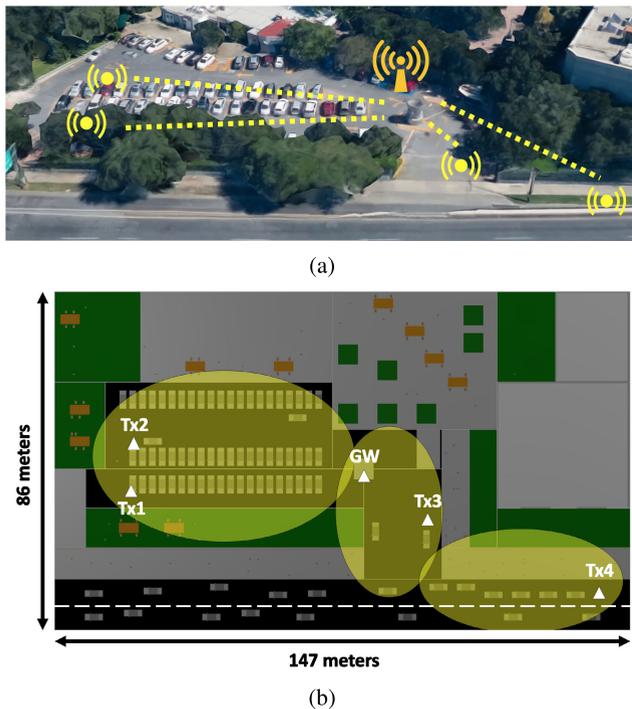


FIGURE 2. Real (a) and 2D (b) view of the considered SP scenario.

In addition, Figure 2b shows a 2D representation of the scenario, where the areas where full coverage is required are highlighted with yellow ellipses, while localization of sensor nodes (Tx) and a gateway (GW) are represented with white triangles. As it can be observed in Figure 2b, the scenario dimensions have a length of 147 m, a width of 86 m, and a height of 20 m. In the selected SP scenario, the main objectives include identifying available parking spots, monitoring traffic conditions within the vicinity of the parking spots and developing an application to provide real-time parking information to the university community.

C. COMMUNICATIONS ARCHITECTURE

In order to fulfill the requirements of the SP application defined in the previous subsection, this article proposes to make use of the SP communications architecture depicted in Figure 3. At the bottom of the image is the IoT Node

Layer, which essentially includes the parking sensors, but where additional sensors and actuators may be also included. Since the system parking sensors may be deployed throughout large areas, the architecture considers that the IoT nodes may collaborate among them through a mesh network to exchange information and commands with the upper layers. At the bottom of the architecture are also the vehicles that drive through the parking lot, either because they are looking for a spot to park or because they are leaving it.

Both the circulating vehicles and the parking sensors communicate with the fog computing nodes of the fog layer. Each fog computing node essentially consists of a communications interface, a beaconing interface and a control subsystem. The communications transceiver is used for exchanging data with the deployed parking sensor nodes. The beaconing interface sends beacons to the circulating vehicles in order to broadcast relevant information periodically without requiring to establish a connection (thus, being usually faster than connection-oriented communications). Regarding the control subsystem, it manages the previously mentioned interfaces and provides fog services to the parking sensors. In addition, the control subsystem is able to communicate with the cloud through the gateway layer, which in the proposed architecture only performs routing tasks. It is important to note that fog computing nodes can collaborate among them so as to carry out more computationally complex tasks and to share the sensor data collected from physically scattered parking areas.

Finally, at the top of the architecture is the cloud, which provides storage and processing power to the rest of the architecture. Thus, it acts as a back-end and, often, as a front-end, providing remote users with access to the stored data and to the management interface.

IV. VALIDATION

A. 3D-RL TECHNIQUE

Before deploying the necessary hardware infrastructure to implement the previously described communications architecture, radio planning analysis tasks can be performed in order to assist optimal node configuration and network level layout. For such a goal, 3D-RL is a useful tool for communications system designers, since it helps to understand and predict propagation channel behavior. Specifically, for this work, an in-house deterministic 3D-RL algorithm was developed to gain insight into the relevant channel propagation characteristics for RF analysis and characterization in complex SP scenarios. Since deterministic models are based on full-wave techniques or approximations of the resolution of Maxwell's equations, they provide more accurate results than those obtained by theoretical or empirical methods. The drawback of such deterministic approaches is that higher computational time is required for simulations. Nonetheless, the proposed 3D-RL algorithm has been previously tested within ITS environments for V2V and Vehicle-to-Infrastructure (V2I) communication links [57], achieving a good trade-off between accuracy and computational time.

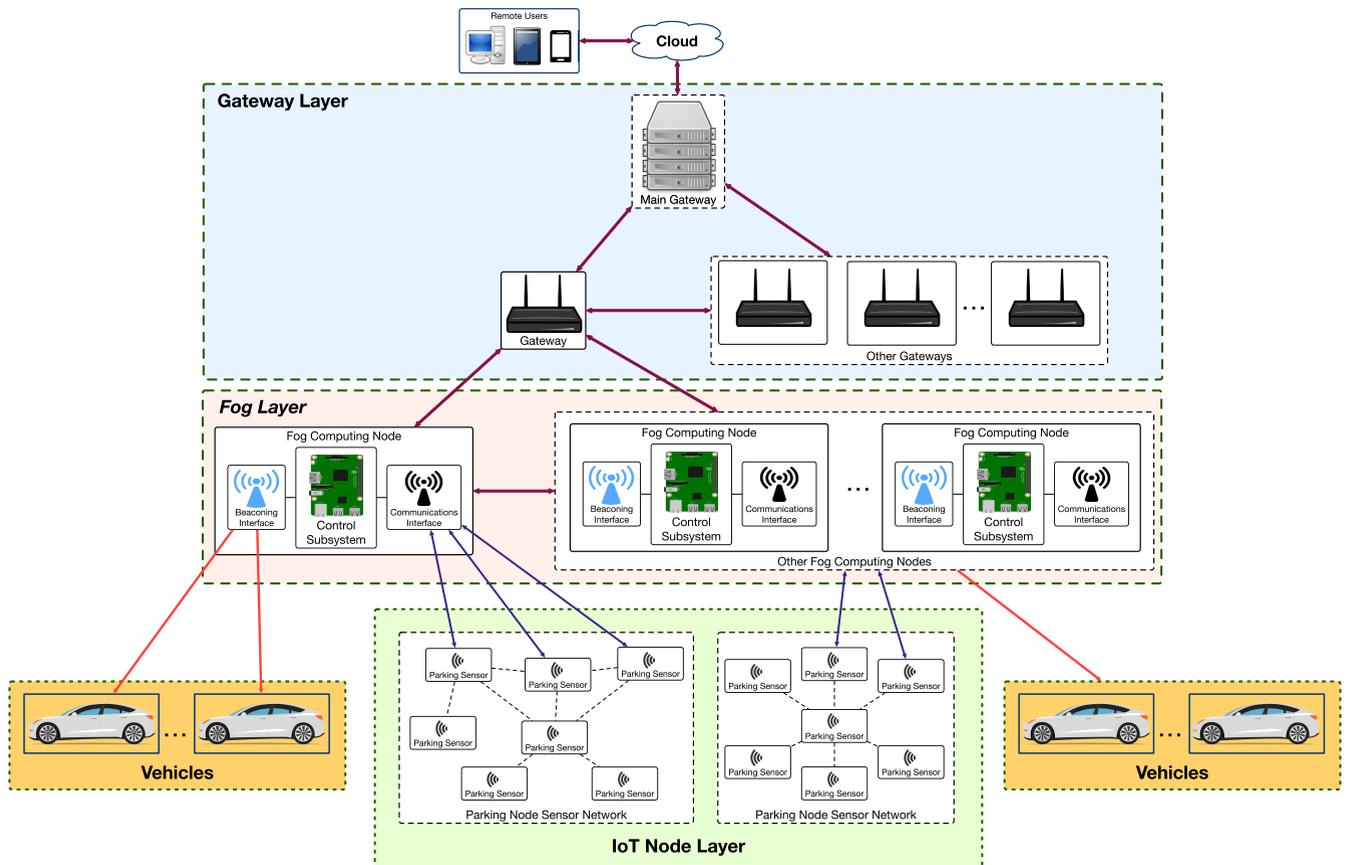


FIGURE 3. Proposed smart parking communications architecture.

The employed 3D-RL technique is based on geometrical optics (GO) and the uniform theory of diffraction (UTD). A combination of optic and electromagnetic theories is used to classify a ray with a spatial point on the wavefront of the radiated wave. The spherical coordinate system is used to launch rays at an elevation angle θ and an azimuth angle ϕ . Hence, the 3D-RL strategy can be described as a set of rays launched in every direction from the transmitter, emulating the electromagnetic propagation behavior of real waves. For this simulation methodology, the complete volume of the scenario is designed and a full 3D scenario is created, considering all the elements present in the real environment under analysis.

For achieving realistic and accurate simulation results, the frequency of operation, the number of multipath reflections, antenna parameters, as well as antennas' radiation patterns and angular and spatial resolution must be considered as input parameters in the algorithm. Multiple scatterers or obstacles such as vehicles, people, buildings, or trees can produce significant signal blockage in this challenging environment, where the different signal paths give rise to multiple attenuated, delayed and phase-shifted echoes of the transmitted signal arriving at the receiver. One of the main contributions of the proposed 3D-RL technique is the precise modeling of the electromagnetic propagation

phenomena (reflection, transmission, diffraction) considering all the obstacles within the selected scenario, as well as their material properties (relative permittivity and conductivity). Thus, when a ray hits an obstacle, it produces reflected and refracted rays with new angles that can be derived from Snell's law. A new family of diffracted rays is generated when a ray hits a wedge. The interested reader can find further details on the proposed technique in [58], where the optimal input parameters are also analyzed.

B. SCENARIO MODELING AND 3D-RL SETUP

After designing the SP system, a 3D simulation model can be implemented. Thus, Figure 4 shows the rendered view of the scenario described in Section III-B.

With such a 3D model, thanks to 3D-RL simulations, the coverage region for the different transceiver locations can be analyzed to fulfill the proposed requirements, thus allowing for optimizing the network performance, for decreasing energy consumption and, eventually, for reducing implementation costs.

In the analyzed scenario, simulations were initially performed positioning the transmitters at the locations indicated in Figure 2b. Note that such locations may be conditioned when direct communications with the gateway are needed.

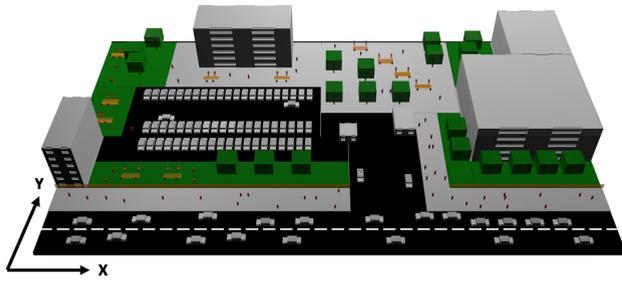


FIGURE 4. 3D model of the scenario.

For this work, among the different technologies analyzed in Section II-D, ZigBee was chosen because of its reliability, low energy consumption and cost, in addition to its hardware integration possibilities. Therefore, the 3D-RL tool was adjusted to consider the following main ZigBee characteristics: a 2.4 GHz operation frequency, 250 kbps of transmission rate, 4 dBm of transmission power and -97 dBm sensitivity. Transmitter and receiver omnidirectional antennas were selected, with linear vertical polarization and a gain of 2 dBi (a 2.4 GHz dipole antenna from LS Research).

The design of the scenario considered potential scatterers (e.g., buildings, vehicles, trees, people) and their frequency dispersive material properties at the frequency under analysis. A summary of the simulation parameters is given in Table 3. Such parameters were obtained after a convergence analysis performed for the algorithm to obtain a trade-off between accuracy and simulation time [58]. The material properties of the different obstacles were considered by indicating their conductivity and relative permittivity for the frequency under analysis and are shown in Table 4 [59]. Trees are considered in the model with two different parts, the trunk and the foliage, which can be modeled in the algorithm with different dimensions and geometries depending of the considered vegetation. These two differentiated parts of the tree were

TABLE 3. Simulation parameters.

Parameter	Value
Operation frequency	2.4 GHz
Transmitter power	4 dBm
Horizontal plane angle resolution ($\Delta\phi$)	1°
Vertical plane angle resolution ($\Delta\theta$)	1°
Reflections	6
Cuboid resolution	1 m × 1 m × 1 m
Diffraction	Yes

TABLE 4. Material properties in the ray-launching simulation.

Material	Relative Permittivity (ϵ_r)	Conductivity (S/m)
Air	1	0
Glass	6.06	10^{-12}
Concrete	5.66	0.142
Metal	4.5	$4 * 10^7$
Rubber	2.61	0
Tree	[60]	[60]
People	[61]	[61]

considered in the geometrical model as homogeneous, considering their different material properties. The top portion of the tree exhibits high variability depending on the season. In addition, the humidity of the wood of the trunk of the trees strongly varies depending on the weather. This has led us to consider average conditions for the material properties of the foliage and the trunk of the trees. For that purpose, the values obtained in [60] for the material properties of the wood and the foliage were used.

C. 3D-RL ANALYSIS

In the RL approach, the full volume of the scenario is divided into 3D cuboids that store the propagation parameters during simulation. With these results, the received power is calculated as the sum of the incident electric vector fields received by each cuboid during a time interval. Thus, the received power for the full volume of the scenario can be obtained as a function of location.

The location of the nodes (GW and Tx_i , $i = 1, \dots, 4$) has been obtained by performing a site surveying analysis, considering both system level requirements on wireless channel connectivity and specific requirements for equipment deployment. Site surveying techniques have been analyzed in the literature, with particular emphasis for mobile communication systems and more recently, for wireless sensor networks [62], [63]. Depending on the application, static location (i.e., deterministic network topology, with a moderate number of nodes or with precise knowledge of the coverage area topology in which nodes are located at specific locations) or dynamic location (i.e., random network topology requirements with large scale deployments in areas with limited topological information, with no prior location and the application of statistical based modeling and subsequent optimization) techniques can be employed. In the case of a SP, due to the number of deployed sensors and the scenario configuration, it is appropriate to follow a static approach, since node location can be selected in order to optimize coverage requirements as well as to consider physical deployment constraints.

The parking scenario under analysis has a unique vehicle entrance, in which an access control post is located. Vehicles must access through this point to either park, following a circular one directional path along the parking premises, or drop off travelers in the access control and then follow the roundabout to exit. The access control point has been selected as the GW location because it has all the infrastructure requirements in terms of connectivity, equipment protection and electrical mains. Moreover, the access control point, because of its inherent surveillance requirements, has a direct field of view of over 200° within the parking scenario, providing LoS coverage for a large percentage of the parking area.

Regarding the node transmitters, four different locations were chosen in order to consider both potential worst-case operating conditions and intensive communication requirements. Thus, the transmitter nodes termed as Tx 1 and Tx 2

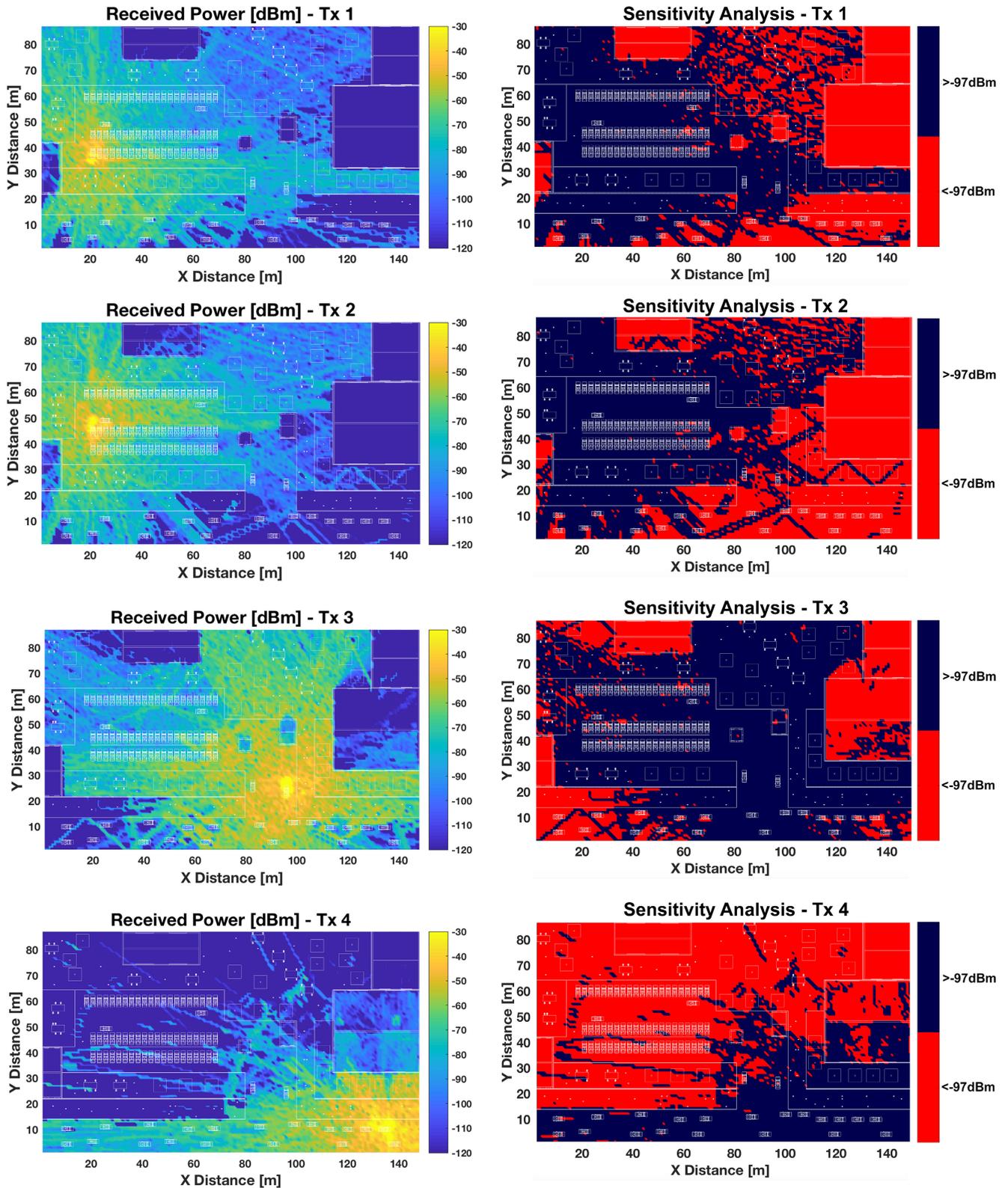


FIGURE 5. Two-dimensional planes of Received Power [dBm] (left) and Sensitivity (right) for a 10 cm height for $Tx_i, i = 1, \dots, 4$.

consider LoS and Non-Line-of-Sight (NLoS) operating conditions respectively for maximum link distances to the GW. Tx 3 corresponds to the parking entrance, a LoS location in

which higher data exchanges are expected due vehicle and user dynamics. An additional node Tx 4 is considered to be in NLoS conditions next to the entrance to the parking, in order

to provide information to the drivers prior to their access to the parking premises. Moreover, for all the considered locations of the sensor nodes $Tx_i, i = 1, \dots, 4$, infrastructure requirements (e.g., the possibility of having access to the electrical mains or to install nodes on the floor) have also been taken into account, analyzing the existence of pre-existing elements, such as lamp posts or indication posts.

For the selected urban smart parking environment, the transmitter locations (shown in Figure 2b) were simulated to assess radio wave propagation characteristics and, specifically, direct communication with the GW. Thus, first, the electromagnetic radio wave propagation impact for the selected transmitter positions was assessed. Figure 5, on its left side, shows the received power XY-planes for transmitters $Tx_i, i = 1, \dots, 4$, which were located in each area of interest at the same height as the receiver antenna (the surface plane was placed on the road). As it can be observed, the influence of the multiple obstacles present in the scenario (e.g., vehicles, buildings, vegetation, people) has a significant impact on signal propagation, existing significant multipath interference in the scatterer zone.

To gain insight into the system performance, sensitivity fulfillment planes (according to the ZigBee characteristics) are presented on the right of Figure 5 for transmitters $Tx_i, i = 1, \dots, 4$. Such figures show that sensitivity fulfillment is highly dependent on the transmitter location and on the morphology and topology of the considered scenario. The received power variations (and hence sensitivity fulfillment) can be strongly mitigated by changing the morphology of the wireless network (i.e., by adding sensor nodes), thus obtaining an adequate received power level for all the transceiver locations in the areas of interest.

In order to consider the worst possible scenario for the proposed SP system, the network composed by the critical transmitters of each area of interest and the GW were analyzed. The received power results are presented in Figure 6, which allows for verifying the desired full coverage and the direct communication with the GW.

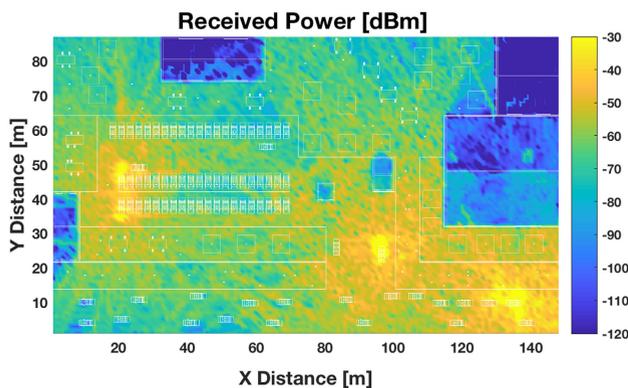


FIGURE 6. Two-dimensional plane of Received Power [dBm] for height cut-plane of 10 cm for the complete network.

The time domain results shown in Figure 7 demonstrate the strong presence of multipath propagation in this type of

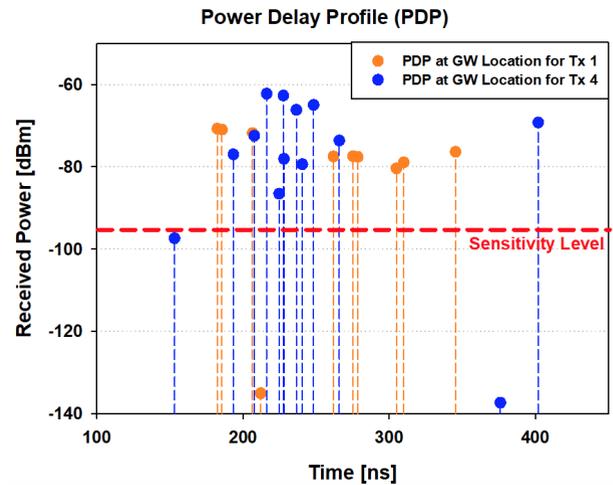


FIGURE 7. Power Delay Profile at the GW when Tx 1 and Tx 4 are transmitting.

complex urban environment. Specifically, the power delay profile for the GW location, when two of the most distant transmitters of the network are emitting (Tx 1 and Tx 4), is shown. A red line has been included to clearly mark the sensitivity level of the ZigBee nodes (-97 dBm). The figure shows that most of the multipath components are above the sensitivity level in both cases, which implies that they can directly communicate with the GW by using single-hop directional links. Thus, there is no need to use multi-hop communications for this specific SP scenario.

The obtained simulation results and the estimated values of the received power enable wireless system performance analysis. Thus, the signal to interference noise ratio (SINR) was calculated for the Tx 3 transmitter case and is shown in Figure 8. The SINR was obtained by considering Tx 3 as the transmitter of interest and the other three transmitters (Tx 1, Tx 2, Tx 4) as interference sources that operate in the same frequency band. The thermal noise power was also considered as $N_0 = KTB$, where K is the Boltzmann's constant, T is the temperature in $^{\circ}K$ and B is the ZigBee channel bandwidth (2 MHz). The SINR was obtained for the full volume of the considered scenario, giving valuable information about the areas and exact locations where the placement of a parking sensor node will be more adequate in terms of received signal quality while maintaining the optimal wireless power transmission of the system. In addition, it can be observed in Figure 8 that the areas where interfering devices were placed are the ones with lower SINR. The GW was placed optimally, where the SINR was high.

The obtained SINR values can be used to calculate the bit error probability (E_b/N_0) for a given modulation scheme. In this case, the ZigBee modulation scheme Offset-Quadrature-Phase-Shift-Keying (OQPSK) was used and a maximum likelihood receiver was assumed. The E_b/N_0 values were estimated for the complete volume of the scenario. Thus, Figure 9 represents the bi-dimensional plane of E_b/N_0 for the same height at which Tx 3 is placed. To gain insight

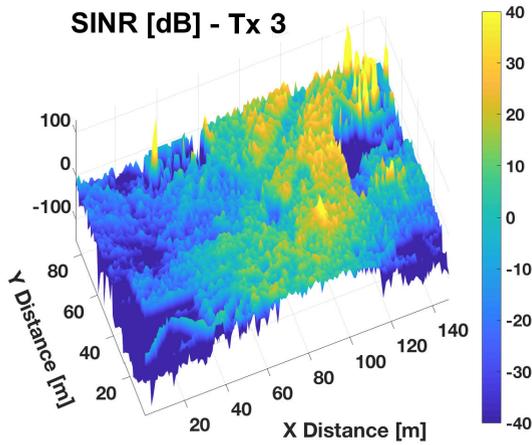
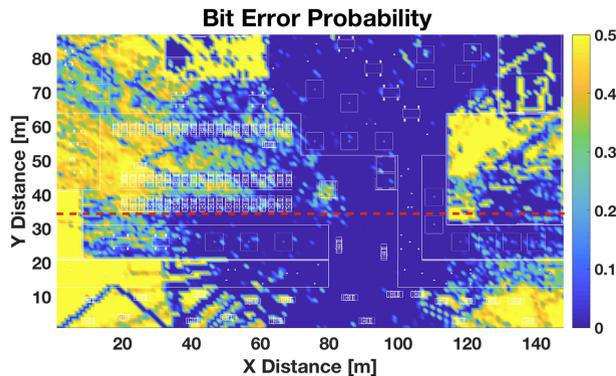
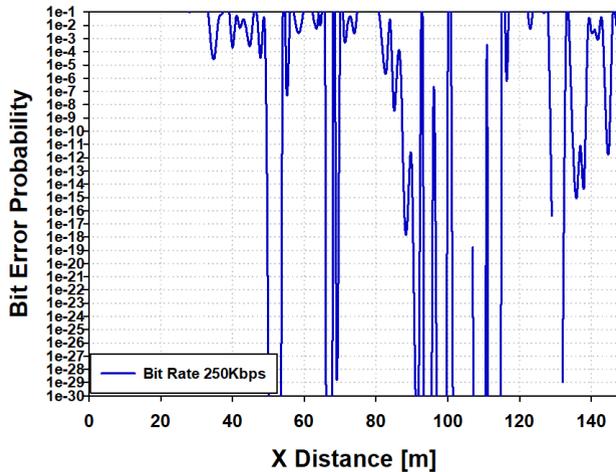


FIGURE 8. Spatial distribution of the SINR in the SP scenario when Tx 3 is transmitting.



(a)



(b)

FIGURE 9. (a) Bi-dimensional plane of E_b/N_0 when Tx 3 is transmitting for an OQPSK modulation scheme. (b) Radial distribution lines of Bit Error Probability along the red dash line for 250 kbps bit rate.

into these results, the linear distribution of E_b/N_0 is depicted along a red dash line shown in Figure 9a, and is presented in Figure 9b. It can be observed that quality degradation strongly depends on the transceiver distance, with large variations over small distances.

As a conclusion, it can be stated that the results demonstrate the importance of performing complete radio wave propagation analysis before the deployment of IoT wireless systems in complex urban scenarios with large density of vehicles and other scatterers affecting signal propagation. Hence, the proposed deterministic approach can be useful for predicting potential communications issues in advance, before deploying the smart parking system. For instance, the presented 3D-RL tool allows designers to readjust the transceiver locations to extend coverage, prevent communication failures and to optimize the performance of their wireless networks.

D. EMPIRICAL VALIDATION

In order to verify the accuracy of the obtained simulation results, an empirical measurement campaign was carried out in the considered scenario. The measurements were performed to assess the communication links between the transceivers and the GW for all the areas of interest. The CC2530 ZigBee development kit from Texas Instruments, was used for such a purpose, as well as an FSH18 spectrum analyzer from Rohde&Schwarz. The different transmitter locations and the GW position are shown in Figure 10. As it can be observed in the Figure, the measurements are classified as Line-of-Sight (LoS) and Non-Line-of-Sight (NLoS) for the parking lot area, and LoS and NLoS along the road.

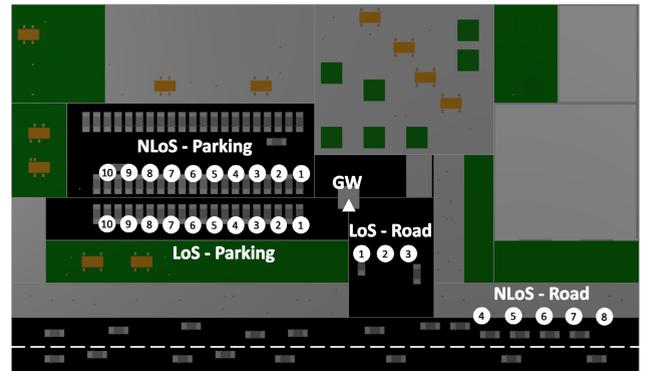


FIGURE 10. Aerial view of measurement campaign. The considered transmitter locations are represented with white points while the GW is depicted with a white triangle.

Figure 11 shows a comparison between the received power results obtained through simulations and during the measurement campaign. The results reach a good agreement, with a mean error of 0.5 dB and a standard deviation of 0.81 dB for LoS, and a mean error of 1.28 dB and a standard deviation of 1.36 dB for NLoS.

In addition, the bit error rate was also measured with the CC2530 kit. To compare such results with the simulated ones, the maximum packet length for ZigBee was considered (i.e., 133 bytes). Figures 12 to 14 compare the empirical and simulated bit error probability for different measurement locations with the radial distribution lines obtained by simulation for the same spatial points. Two different simulations

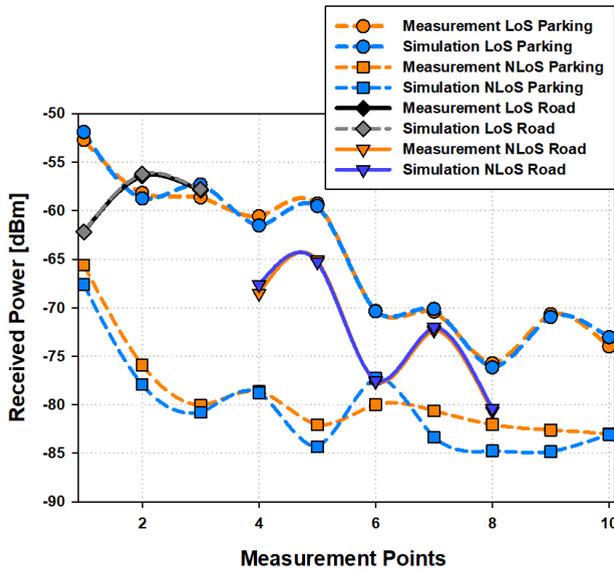


FIGURE 11. Simulation and measurements received power comparison. LoS in the parking, NLoS in the parking and LoS (Points 1-3) and NLoS (Points 4-8) along the road. (Measurement points are represented in Figure 10).

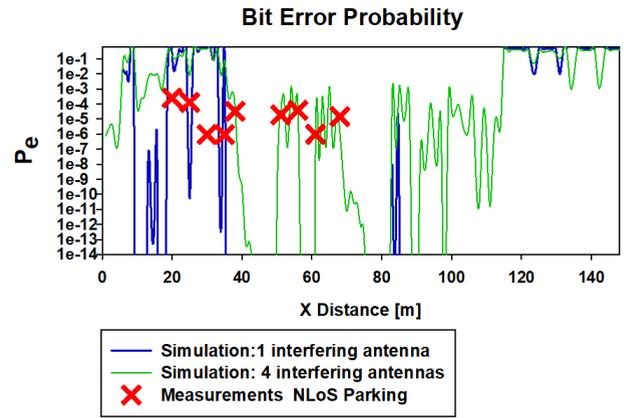


FIGURE 13. Simulation and measurement bit error probability comparison for NLoS in the parking area.

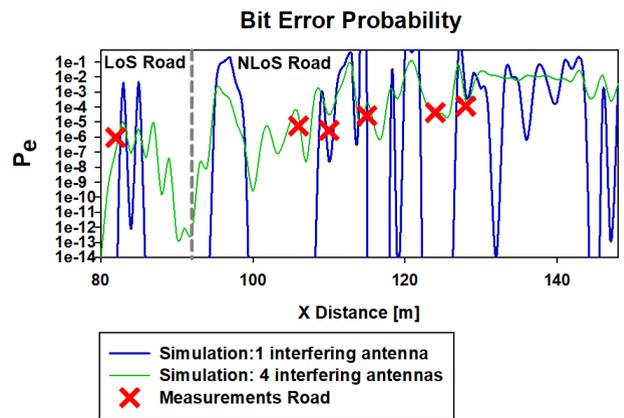


FIGURE 14. Simulation and measurement bit error probability comparison for LoS and NLoS in the road.

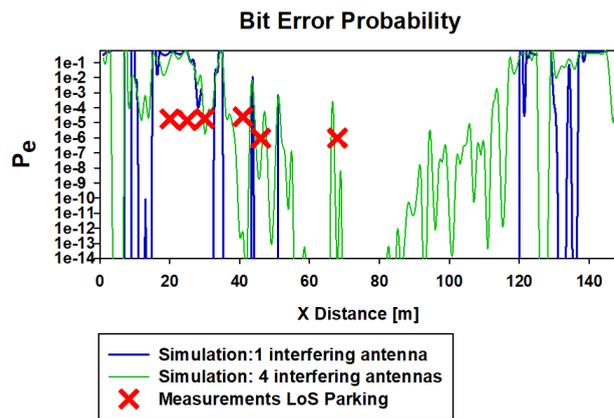


FIGURE 12. Simulation and measurement bit error probability comparison for LoS in the parking area.

were considered: the first with one interfering antenna and the second one with four interfering antennas. As it can be observed in Figures 12 to 14, the simulations with four interfering antennas show a better match with the measurement values, which highlights the importance of considering all the interference sources in the scenario.

V. IMPLEMENTATION

Once the 3D-RL analysis of the previous section confirmed the viability of the proposed SP architecture, the implementation stage was undertaken. The next subsections describe such an implementation, first detailing the implemented communications architecture and then its main hardware and software components.

A. IMPLEMENTED ARCHITECTURE

The designed architecture was implemented as it is illustrated in Figure 15. Specifically, the layers of the architecture were implemented as follows:

- IoT Node Layer:
 - Each smart parking node is composed of a GY US42 ultrasound sensor and an Xbee 3 module. The ultrasound sensor is used to detect vehicles due to its accuracy and low cost [2]. Specifically, the selected GY US42 sensor is a rangefinder sensor module powered at 5 V with a measuring range between 20 cm and 720 cm, a resolution of 1 cm, a 15 Hz response frequency and a working current of 9 mA (at 5 V). Regarding the Xbee 3, it is a 2.4 GHz module from Digi [64] that can operate as a ZigBee, IEEE 802.15.4, DigiMesh or BLE transceiver. A picture of a disassembled smart parking node is shown (without its battery) in Figure 16a, while the schematic in Figure 16b shows how such components are connected.
 - For the experiments described in this article, the Xbee modules were programmed with the ZigBee 3.0 firmware. The ZigBee modules on the sensor nodes acted as end-devices (i.e., they simply

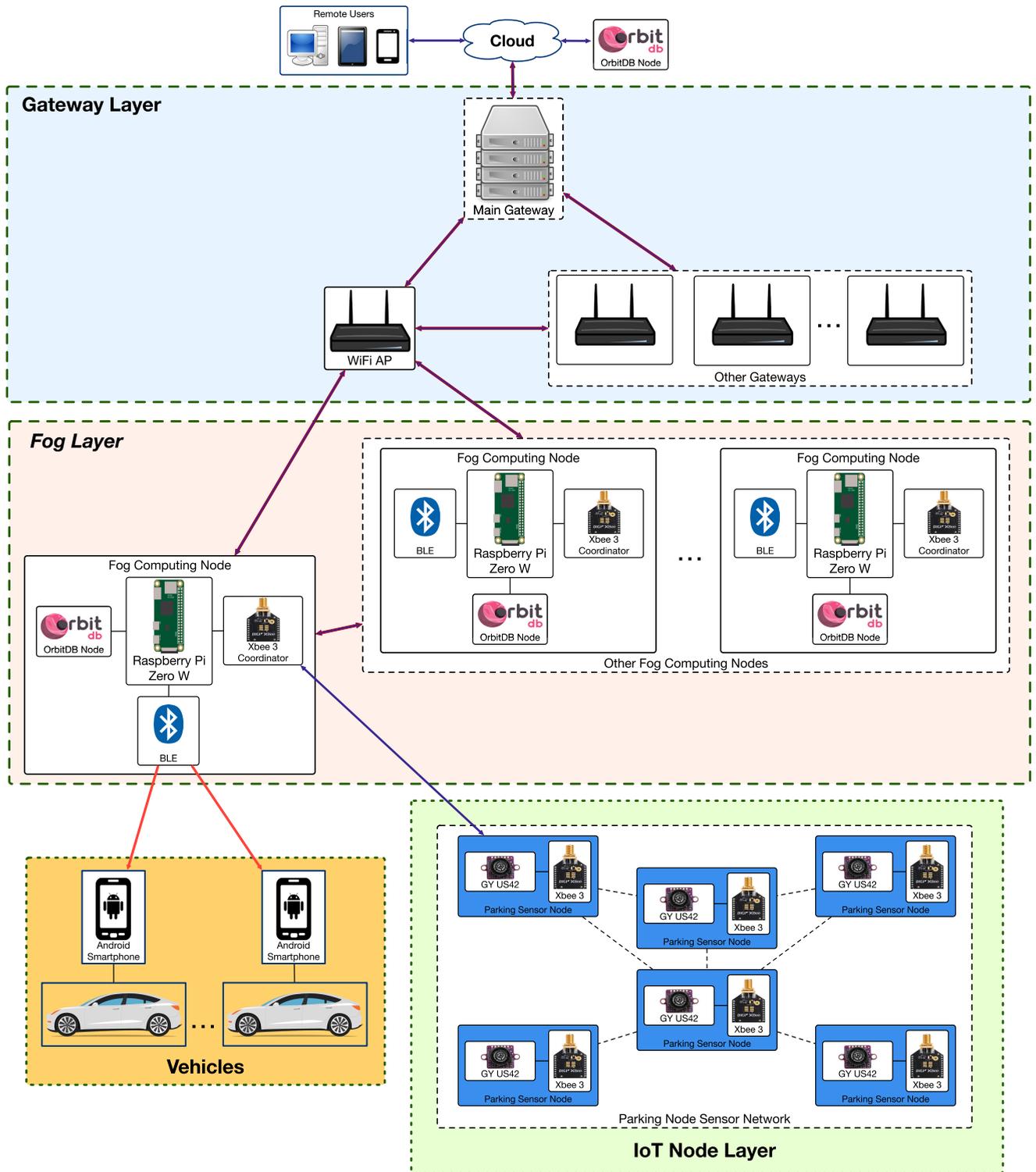
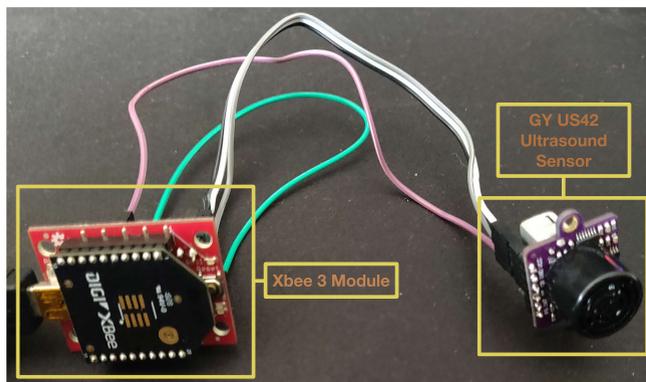


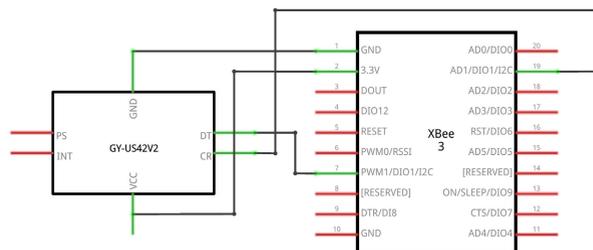
FIGURE 15. Implemented architecture for the Smart Parking System.

- collect sensor data and receive remote commands) or routers (they can also route the communications from other ZigBee devices).
- Vehicles. Each vehicle of the system makes use of a mobile app whose functionality is described later in Section V-C.

- The fog layer consists of fog computing nodes whose control subsystem can run on a Single-Board Computer (SBC) like Raspberry Pi, Orange Pi or Beagle Bone, which provide a good tradeoff between cost, size, energy efficiency, development speed and computational power. Among the multiple existing SBCs,



(a)



(b)

FIGURE 16. Disassembled components of the parking sensor node (a) and schematic (b).

the Raspberry Pi Zero W was selected. Such an SBC contains a 1 GHz single-core CPU, 512 of RAM and provides WiFi (IEEE 802.11 b/g/n) and Bluetooth (Bluetooth 4.1 and BLE) connectivity. The Raspberry is connected to an Xbee 3 module that acts as ZigBee coordinator, so it manages the IoT communications information exchanged between the IoT Node Layer and the Fog Layer. In addition, every fog computing node runs an OrbitDB database that is synchronized among the different fog nodes (this is later detailed in Section V-D).

- The GW layer is made of the necessary infrastructure to connect the fog nodes with the cloud. Since the Raspberry Pi Zero W has no Ethernet connector, all the communications are performed through WiFi.
- The cloud also runs an OrbitDB node that is synchronized periodically with the decentralized databases of the fog computing nodes.

B. BEACONING SYSTEM

To transmit information from the fog nodes to the drivers, each node emits BLE beacons that are decoded by an Android app used by such drivers (the Android application is described later in Section V-C). Among the different available beacon frame formats (e.g., Eddystone, AltBeacon), iBeacon [65] was selected due to their wide support and implementation simplicity. Specifically, the emitted beacons preserve most of the iBeacon frame format, but modify certain fields to code relevant information on the SP application. Specifically, the Universally Unique IDentifier (UUID) field of each iBeacon frame is used as illustrated in Figure 17 to code data related to three different types of beacons:

- Type 1 beacon: it indicates the status of the monitored parking spots. Specifically, each fog node monitors a predefined group of parking spots that is called a section. The UUID field of this type of beacons is structured as follows:
 - Type (one nibble): it indicates the type of beacon. For Type 1 beacons this nibble is always set to 0 × 1.

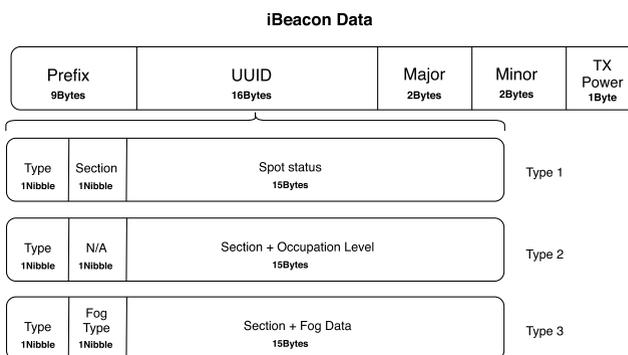


FIGURE 17. Internal structure of the three types of BLE beacons.

- Section (one nibble): it indicates the parking lot section associated with the fog node that generates the beacon. With one nibble (i.e., 4 bits), the system allows for monitoring a maximum of 16 sections, which is more than enough for most parking lots.
- Spot Status (15 bytes): these bytes inform on the individual status of each parking spot located in a specific section. The spot status is equal to ‘1’ when the parking spot is occupied and to ‘0’ when it is not. With this coding, the system can address up to 120 spots per section, so it can manage a total of up to 1,920 spots for the 16 available sections.
- Type 2 beacons: this type of beacon is used to indicate the overall occupation of all the sections. This information is useful when a driver arrives at the parking, since it provides a global vision of the occupation of all the sections so that the user can go directly to a specific section with available parking spots. The internal sub-fields of the UUID field of this beacon are structured as follows:
 - Type (one nibble): it is set to 0 × 2.
 - N/A (one nibble): it is currently not used.
 - Section + Occupation level (15 Bytes): these bytes represent the occupation of all sections except for the section of the node that generates the beacon

(i.e., the fog node informs with this type of beacon on the occupancy level of the other sections, since it already informs on its monitored section through Type 1 beacons). For each of the 15 bytes, the first nibble indicates the section while the second one codifies the section occupation. Since the second nibble has only 4 bits to encode the occupancy of the 120 parking spots of a section, they are coded as follows:

- * 0×0 - there are no spots available.
- * 0×1 - there is one spot available.
- * $0 \times 2, 0 \times 03, \dots, 0xA$ - there are two, three...ten spots available.
- * $0xB$ - there are between 10 and 20 spots available.
- * $0xC$ - there are more than 20 spots available.

In addition, the second nibble is set to $0xF$ when it is used as a delimiter to indicate the system that there is no more information on the occupancy of the sections (this is used when the 16 sections are not managed by the system).

- Type 3 beacon: this is a generic beacon used to provide relevant information from the decentralized database. It consists of the following sub-fields of its UUID field:
 - Type (one nibble): it is always set to 0×3 .
 - Fog type (one nibble): it defines the type of data that is embedded into the beacon.
 - Section + Fog Data (15 bytes): these bytes carry the provided data, which can be related to a specific section, to all the sections or to the whole system. For instance, these data can be used to indicate the average amount of time that a driver would have to wait for an empty spot (this information may be estimated by applying a probabilistic prediction algorithm).

The usual operation of the beaconing system of a fog node can be divided into two stages. In the first stage the ZigBee interface of the fog node receives the information from the associated parking sensor nodes and updates their status in the database. During the second stage the different types of beacons are generated periodically: the collected information is first read from the database, then coded according to the previously described beacon frame formats and, finally, the beacons are broadcast. Type 1 and 2 beacons are sent more frequently than Type 3 beacons, since their information is updated more frequently. In this regard, it is important to point out that the information broadcast by Type 1 and 2 beacons does not need data persistence if storing historical data is not necessary. In such a case, a subscription/publication messaging strategy could be more efficient, especially when the broadcast information needs to be updated fast. In contrast, Type 3 beacons are not updated frequently, but the information they use requires persistence in order to generate the involved statistics. The experiments detailed later in Section VI analyze the performance of the two previously mentioned data storage approaches.

C. SMARTPHONE APP

Drivers receive the information of the proposed SP system through a smartphone application. Such an application provides users with a practical and efficient way to park the vehicle through the coded information embedded into the BLE beacons broadcast by the fog nodes. The user only needs to enable Bluetooth on his/her smartphone to use all the app features. Figure 18 shows a screenshot of the main screen of the developed SP app. As it can be observed, such a screen, at the top, provides information on all the sections of the parking lot, so as to identify quickly the sections where there are available spots. Below the global information, the data on the current section of the parking lot is shown, where it is indicated in green the parking spot identifiers of the available spots. Both the global and local information are updated automatically as new beacons are received. To estimate the driver current section, the app uses the Received Signal Strength Indicator (RSSI) and MAC of the received BLE beacons.

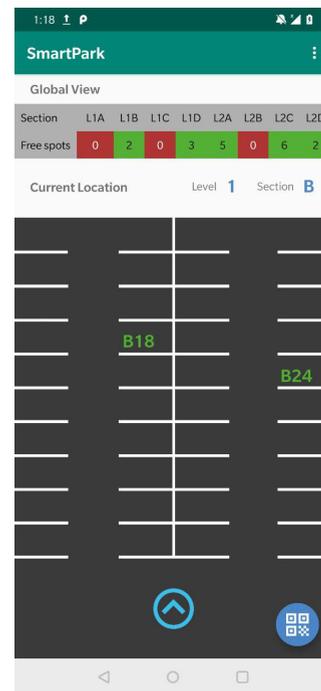


FIGURE 18. Main screen of the SP app.

At the bottom, on the right of Figure 18, a QR icon allows for accessing an additional feature: after parking the vehicle, the driver can scan a QR code that is associated to the parking spot to remember where he/she has parked. This information can be used to guide the driver back to the parked vehicle, but it can also be used by the SP app to confirm that the driver has actually parked and left the vehicle.

The sequence diagram on Figure 19 illustrates the overall operation of the implemented system and how the SP app works. The process starts when the sensor nodes read the status of the monitored parking spots and send a message to the associated ZigBee coordinator, which is run on a fog node.

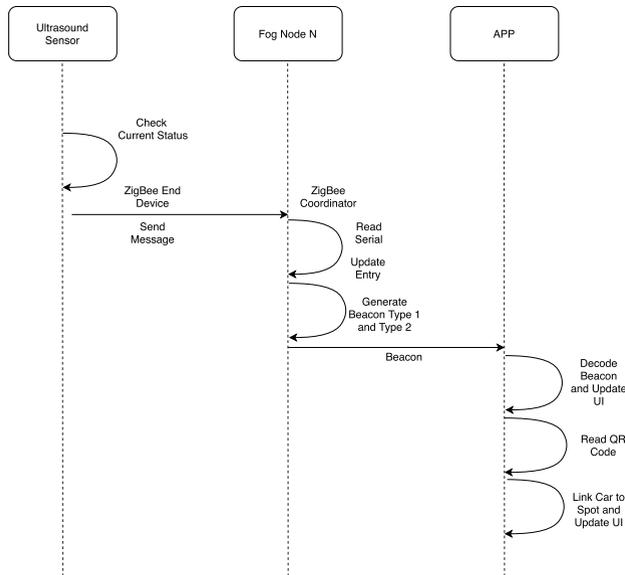


FIGURE 19. Sequence diagram on the system and app main steps.

In such a fog node, the status of each spot is updated and, periodically, Type 1 and Type 2 beacons are generated. Such beacons are then transmitted sequentially, adding a short delay between their transmission to ease their reception and processing by the mobile app. When the app receives the beacons from a section, it decodes them and updates in its User Interface (UI) the status of all the parking spots in such a section (in Figure 18, in the ‘Current Location’ area of the UI) and the status of all the available spots in all sections (in Figure 18, in the ‘Global View’ area). As it was previously mentioned, it is also possible to register in the app the specific spot where the vehicle was parked by scanning the QR code of such a spot.

D. DECENTRALIZED STORAGE

The data collected from the parking sensors is stored on OrbitDB, which is a decentralized database that runs on Inter-Planetary File System (IPFS) [66]. IPFS implements a peer-to-peer protocol that allows for sharing content-addressed hyperlinks. Underneath, IPFS uses a variant of Merkle Direct-Acyclic Graphs (DAGs) that ease file identification and deduplication. OrbitDB uses IPFS to implement a serverless database that is distributed among multiple peers and that are able to synchronize the data content.

OrbitDB allows for creating five different types of databases depending on their use (i.e., eventlog, feed, key-value, docstore and counter). For the experiments described in this article, a eventlog database was used to keep track of the historical parking use values in an immutable way, while a docstore database was deployed to store the information on the characteristics on the diverse parking sections and spots.

Once OrbitDB is running in a network of peers, a node can fetch content from another node. Such an operation is performed automatically by OrbitDB to keep all the node

data synchronized. For other operations like deleting or writing data, a node needs to send specific requests using the OrbitDB API.

VI. EXPERIMENTS

In order to evaluate the performance of the system in terms of response latency, several tests were performed. It should be noted that the obtained latency values were conditioned by the used hardware (e.g., Raspberry Pi Zero for the fog nodes), the implemented software and the selected wireless communications protocols. Considering such restrictions, the following subsections describe experiments that estimate the performance of the system when reading and writing data through the decentralized storage and when using the low level publish-subscribe mechanism.

A. OrbitDB WRITING AND REPLICATION OPERATIONS

This first set of tests measured fog node performance when writing data on a fog node and when reading such data from a remote fog node. Thus, Figure 20 shows two curves:

- The ‘sender’ line represents the number of queries per second that the developed fog node can perform when writing information from the parking sensor nodes on its OrbitDB instance.
- The ‘replication’ line indicates the number of queries per second that a remote fog node can perform when reading data from the ‘sender’ node.

For the sake of clarity, Figure 20 shows the average number of queries per second during 10s intervals and for a total of 450 queries. As it can be observed, the average rate for reading and writing data on OrbitDB fluctuates between 1 and 3 queries per second, which is not really high, but usual for a Raspberry Pi Zero like the one used by each fog node. In addition, it can be observed that, as it can be expected, the rate is on average higher and more stable for the ‘sender’, since its operations are performed locally (i.e., no wireless communications are involved).

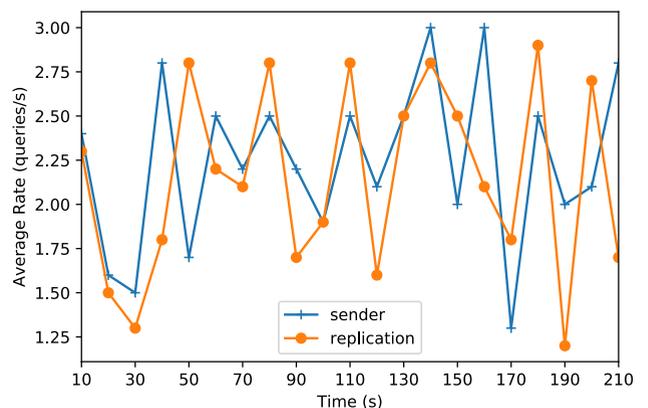


FIGURE 20. Fog node decentralized storage reading and writing performance.

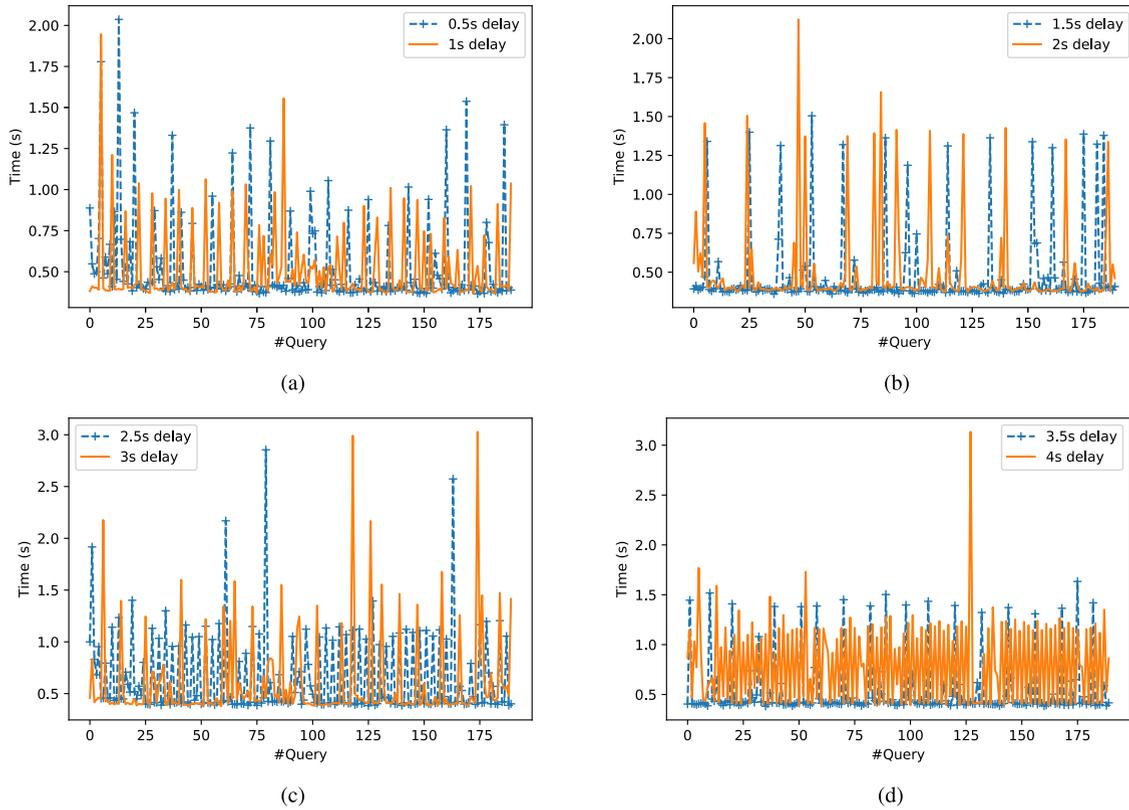


FIGURE 21. Fog node decentralized storage insertion performance with 0.5 s to 4 s delay between queries.

B. OrbitDB INSERTION OPERATIONS

The performance of the decentralized storage was evaluated by carrying out insertion operations on the eventlog database in the same way that fog nodes perform them. For this set of experiments, the Go IPFS REST API was used, since preliminary experiments showed fairly superior performance and concurrence than the JavaScript version of the API when executed on the fog node hardware.

Figure 21 show the database insertion latency for a total of 190 queries when, in order to simulate different load scenarios, such insertions were delayed between 0.5 s and 4 s with respect to each other.

In addition, Figure 22 shows, for the sake of clarity, the most relevant statistics on the measured latencies: their average, variance, standard deviation and high median. This latter was added to reduce the influence of outliers, then providing a more realistic estimation of the perceived latency. Nonetheless, for the case of a 4 s delay, the values shown in Figure 22 show the influence of certain specific outliers, which can be clearly observed in Figure 21d.

Table 5 shows, for each delay between queries, the percentage of queries that were processed before certain amount of time went by (between 0.5 s and 4 s). It can be observed that the vast majority of queries are handled in less than 1.5 s. In addition, it can be concluded that, except for a delay of 0.5 s, more than 95% of the queries were processed before the next query arrived to the fog node. For instance, for a delay

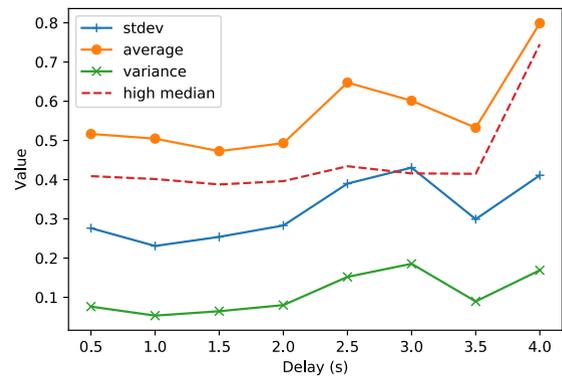


FIGURE 22. OrbitDB latency statistics for different delays.

of 1.5 s, 99.47% of the queries required less than 1.5 s to be processed.

C. PubSub LATENCY

In addition, response times were also measured when making use of the internal communication library of IPFS (PubSub) instead of the OrbitDB API. This case allows for estimating the performance of the fog nodes when no persistence is required.

For the sake of fairness, the results were performed exactly like for the IPFS REST API experiments, but instead of using insertion queries (that would be persisted), publications were made to a PubSub topic. Thus, Figure 23 shows relevant

TABLE 5. Percentage of queries performed in less than 0.5 s to 4 s.

Delay \ Latency	<= 0.5 s	<= 1 s	<= 1.5 s	<= 2 s	<= 2.5 s	<= 3 s	<= 3.5 s	<= 4 s
0.5 s	79.47%	93.68%	98.42%	99.47%	100%	100%	100%	100%
1 s	73.68%	95.26%	98.95%	100%	100%	100%	100%	100%
1.5 s	86.84%	92.63%	99.47%	100%	100%	100%	100%	100%
2 s	84.21%	93.16%	98.42%	99.47%	100%	100%	100%	100%
2.5 s	62.63%	77.89%	97.89%	98.42%	98.95%	100%	100%	100%
3 s	72.63%	86.84%	95.26%	97.89%	98.95%	99.47%	100%	100%
3.5 s	81.05%	90%	98.42%	100%	100%	100%	100%	100%
4 s	44.21%	59.47%	97.89%	99.47%	99.47%	99.47%	100%	100%

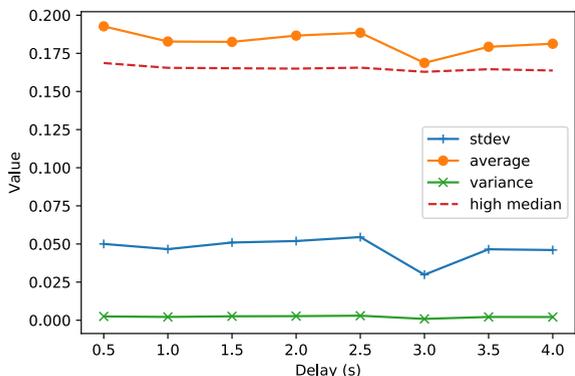


FIGURE 23. PubSub statistics for different data publication delays.

statistics on the obtained latencies, while Figure 24 shows the latencies for different delays of PubSub publications. As it can be observed, latencies are clearly smaller than for the OrbitDB insertions, so less computational load and resources are involved. In fact, the measurements show that, for all the analyzed delays, every message was processed before the next message was published. In addition, the statistics show almost no fluctuations, which indicates that the added data publication delays barely affect the performance of the system. Therefore, it can be concluded that, when no persistence is necessary (like when broadcasting Type 1 and 2 beacons), it is more efficient to make use of PubSub communications instead of involving the whole decentralized database. Nonetheless, both with the OrbitDB API and PubSub the SP information is delivered really fast to the drivers.

D. ANALYSIS OF THE TOTAL LATENCY

The total latency of the proposed system can be modeled as indicated in Equation (1) and includes the time required by the nodes for updating the state of the parking spots ($t_{data_collection}$) and the time needed to notify users about each update ($t_{user_notification}$).

$$t_{total} = t_{data_collection} + t_{user_notification} \tag{1}$$

The first term of Equation (1) ($t_{data_collection}$) can be divided into the three latencies indicated in Equation (2): the latency required by a parking sensor to determine the status of the monitored spot (t_{sensor}); the latency to transmit such a status to the sensor network coordinator ($t_{status_transmission}$); and the latency of updating on the fog node the received status (t_{fog_update}).

$$t_{data_collection} = t_{sensor} + t_{status_transmission} + t_{fog_update} \tag{2}$$

The second term of Equation (1) ($t_{user_notification}$) consists of three latencies indicated in Equation (3): the time needed by a fog node to generate the different types of beacons (t_{beacon_gen}); the beacon advertisement interval (t_{adv}); and the time required by the app to decode the beacon and update the smartphone User Interface (UI) ($t_{updateUI}$).

$$t_{data_transmission} = t_{beacon_gen} + t_{adv} + t_{updateUI} \tag{3}$$

It is important to note that two different types of beacons can be generated and their generation time is slightly different. Type 1 beacons can be generated faster than Type 2 beacons because they only include information that is collected by a local fog node. In contrast, Type 2 beacons, which include information that involve all nodes, require additional time for performing data replication tasks among fog nodes.

The latencies of Equations (2) and (3) were measured and analyzed. The following are the main results and conclusions of such an analysis:

- t_{sensor} : this latency is obtained by taking four consecutive measurements from the ultrasound parking sensor (with a delay of 20 ms between measurements, since the sensor operational frequency is 50 Hz). The performed experiments showed a constant latency of roughly 215 ms.
- $t_{status_transmission}$: To measure this latency, tests were performed with 600 messages that were sent sequentially to the ZigBee coordinator. The average latency was 97 ms, observing no performance degradation over time, just occasional peak values. Such a short latency is essentially related to the low size of the update message, which only requires a single character per sport to indicate its status ([F]ree, [O]ccupied, [U]ndefined).
- t_{fog_update} : it was previously analyzed in detail in Section VI-B, but it can be indicated that the average latency obtained under regular operation was 408 ms.
- t_{beacon_gen} : as it was previously mentioned, latency varies depending on the type of beacon that is generated:
 - $t_{beacon1_gen}$: it depends on the number of spots per section.
 - $t_{beacon2_gen}$: in this case, the required latency is affected by the number of sensors (the larger the number of sensors, the longer it takes to collect their information and build the beacon) and the number of operative sections (i.e., fog nodes).

Tests were performed to determine the beacon generation latencies when measured under different load

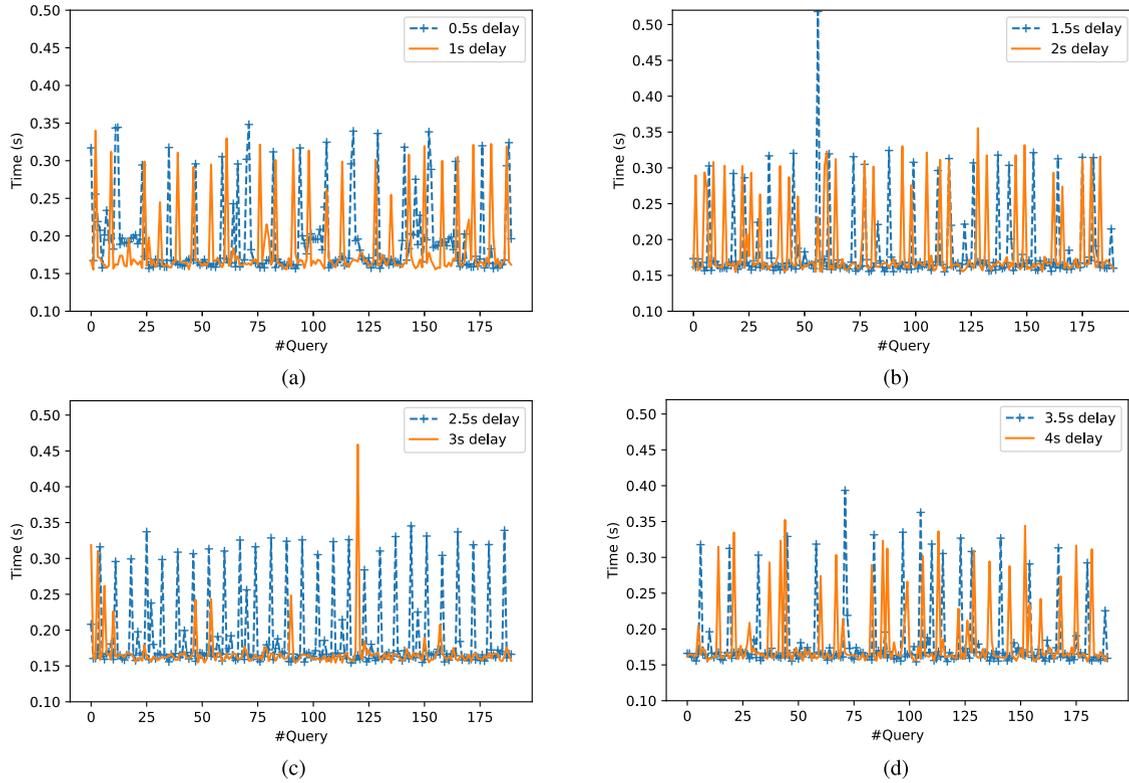


FIGURE 24. PubSub latencies for different data publication delays.

conditions (i.e., for a different number of sections and spots per section) and when generating Type 1 and Type 2 beacons. To estimate the beacon generation latency when using a different number of fog nodes, they were emulated on a device with an ARM1176 CPU, 512 MB of RAM and an Ethernet interface.

The obtained values are presented in Table 6. As it can be observed in the Table, $t_{\text{beacon1_gen}}$ barely changes when doubling the number of monitored parking spots per section. This is expected, since the generation of Type 1 beacons only requires to collect the available information on parking spots from the internal database of a fog node. For instance, the results for 120 parking spots per section is similar for the different number of fog nodes (between 400 and 500 ms). In contrast, the latency related to $t_{\text{beacon2_gen}}$ increases with the number of fog nodes and monitored sensors due to the performed data replication tasks. Nonetheless, for the case

of most computational load (16 section and 120 parking spots), the system only requires 1163 ms.

- t_{adv} , t_{updateUI} : the performed measurements showed almost constant values for these two latencies: 1280 ms for t_{adv} and 31 ms for t_{updateUI} .

E. COMPARISON OF THE SYSTEM PERFORMANCE WITH THE STATE OF THE ART

In order to determine the efficiency of the proposed system, the latencies measured in the previous sections can be compared with those obtained by other state-of-the-art systems. Among all the systems reviewed in Section II-B and compared in Table 1, the only solution that provides experimental results related to latencies is the one detailed in [23]. Nonetheless, it should be noted that, when comparing latency results, there are significant differences between the system described in [23] and the one proposed in this article:

- The results provided in [23] were obtained for simulations, while in this work measurements were performed on a real system.
- The parking sensors used in [23] are simulated video cameras. Thus, image processing is used to detect parking spot status, while the system proposed in this article is based on ultrasound sensors. As it is indicated in [23], a successful sensor-based SP solution is more dependent on the location where sensors are placed, but it is less process-intensive, so it requires less powerful hardware. Therefore, the sensors of the system presented in [23]

TABLE 6. Latencies for different number of fog nodes and parking sensors.

Fog Nodes/Sensors	$t_{\text{beacon1_gen}}$ (ms)	$t_{\text{beacon2_gen}}$ (ms)
5/60	377	667
5/120	484	735
10/60	368	669
10/120	475	974
16/60	350	987
16/120	481	1163

TABLE 7. Latencies for different fog nodes and sensors configurations compared with [23].

Number of Sensors	$t_{data_collection} + t_{beacon1_gen}$ (ms)	$t_{data_collection} + t_{beacon2_gen}$ (ms)	Fog Latency per Sensor in [23] (ms)	Total Fog Latency in [23] (ms)
16	970	1249	7.85	125.6
20	993	1300	8.23	164.6
24	1024	1319	8.59	206.16
28	1021	1346	8.95	250.6
32	1050	1318	9.3	297.6
40	1040	1350	10.02	400.8
44	1023	1332	10.3	453.2
48	1028	1344	10.73	515.04

TABLE 8. Total data traffic and comparison with [23].

Number of Sensors	Fog Network Usage (Bytes)	Type-1 Beacon Traffic (Bytes per beacon)	Type-2 Beacon Traffic (Bytes per beacon)	Total Traffic (Network + Type-1 Beacon + Type-2 Beacon) (Bytes)	Fog Network Usage in [23] (Bytes)	Traffic Reduction
16	6,048	706	2,105	8,859	3,198,400	99.72%
20	7,560	1,180	2,914	11,654	3,998,000	99.71%
24	9,072	1,752	3,293	14,117	4,797,600	99.70%
28	10,584	2,421	3,672	16,677	5,597,200	99.70%
32	12,096	3,187	4,051	19,334	6,396,800	99.70%
40	15,120	4,051	4,817	23,988	7,996,000	99.70%
44	16,632	5,013	5,204	26,849	8,795,600	99.69%
48	18,144	6,073	5,592	29,809	9,595,200	99.69%

require more power consumption than the solution presented in this paper.

- Besides being less computing-intensive, the system proposed in this article generates less network traffic, which is beneficial for the scalability of the system. In fact, no significant latency degradation was observed in the proposed system, even when the maximum number of sections and sensors supported by the system transmitted information. In contrast, in the experiments shown in [23], latency degradation can be observed when increasing the number of video cameras that operate simultaneously in the system.
- To connect the sensors with the fog nodes, the work in [23] assumes that WiFi interfaces are used, but the presented results only perform local simulations. On the contrary, this work makes use of real ZigBee interfaces.
- The system presented in [23] was devised to show real-time parking information on a display panel for each individual section. In contrast, the solution detailed in this paper, thanks to the proposed decentralized architecture that allows fog nodes to collaborate among them, provides information for both every individual section and for the whole parking lot. In addition, such an information is transmitted directly to the users through the developed mobile app.

In addition, to perform a fair comparison with the system described in [23], it is necessary to create a similar experimental setup. Since in [23] the users receive parking information through LEDs placed on each fog node, the latencies to compare should only consider the times required for collecting parking sensor data, for processing them and for updating the spot status on a fog node. Such latencies, for the proposed

system, are shown in Table 7 as the values of $t_{data_collection} + t_{beacon1_gen}$ or $t_{beacon2_gen}$. Thus, the measured time considers, besides sensor data collection and fog node update, the time for generating the two possible beacons to be transmitted.

Table 7 also shows the results given in [23]. Unfortunately, such results only consider simulated latencies for collecting pictures from the virtually deployed cameras: no image acquisition, picture processing or fog node update times are measured, so latencies are clearly lower than the results provided for the system evaluated in this article. Note that, if only status data transmission would be compared, it would be equivalent to comparing the performance of emulated WiFi and real ZigBee interfaces, whose results would not be useful for extracting conclusions that could be used by future SP developers.

Nonetheless, it can be useful to compare the data traffic generated by the proposed system and the one required by the system described in [23]. Such results are provided in Table 8, where three kinds of data traffic are distinguished: the traffic related to the transmission of the two types of BLE beacons and to the sensor data collection by the fog network. This latter traffic includes the one generated by the ZigBee network and the one involved in updating the data from a sensor in OrbitDB. There is also a column that indicates the total traffic, which is the sum of the three previously mentioned data traffics (i.e., it includes the fog network traffic and the transmission of two beacons, one Type-1 beacon and one Type-2 beacon).

Although, as it can be observed in Table 8, the traffic generated by the proposed system increases with the number of deployed sensors, such a traffic is several orders of magnitude smaller than the one required by the system detailed in [23].

In fact, the proposed system allows for removing roughly 99.7% of the traffic generated by the SP system described in [23], thus decreasing significantly fog network traffic and power consumption needs.

VII. CONCLUSIONS

This paper detailed the complete development of a decentralized fog computing-based SP system. Such a development followed a methodology that includes its initial design, its theoretical simulation and its empirical validation. Specifically, for this article, a parking lot located in Monterrey (Mexico) was analyzed and modeled in 3D in order to evaluate through a 3D-RL tool its wireless channel properties. The results obtained by the 3D-RL tool were later validated through a ZigBee measurement campaign, allowing for obtaining the optimal sensor node and GW locations, and concluding that the proposed technologies and architecture are adequate for implementing the SP system. Finally, the proposed system was built by making use of parking sensor nodes based on ZigBee transceivers and ultrasound sensors, while fog computing nodes were created based on a Raspberry Pi Zero and ZigBee/BLE interfaces. Such fog nodes also provided decentralized storage, whose performance was evaluated. The results show that the proposed system is able to deliver information to the driver smartphone in less than 4 s by using lightweight and scalable protocols and with no need for relying on remote servers. Therefore, the article demonstrates the feasibility of the proposed SP development methodology and the usefulness of the used 3D-RL tool. Moreover, this article provides the necessary details on the steps involved in the development of a practical decentralized low-latency system, thus providing useful guidelines for future SP developers.

ACKNOWLEDGMENT

The authors would like to thank Dr. Cesar Vargas-Rosales and Dr. Mahdi Zareei at Tecnológico de Monterrey, for helpful discussion in the preliminary stage of this work.

REFERENCES

- [1] T. M. Fernández-Caramés and P. Fraga-Lamas, "A review on the application of blockchain to the next generation of cybersecure industry 4.0 smart factories," *IEEE Access*, vol. 7, pp. 45201–45218, 2019.
- [2] T. Lin, H. Rivano, and F. Le Mouél, "A survey of smart parking solutions," *IEEE Trans. Intell. Transport. Syst.*, vol. 18, no. 12, pp. 3229–3253, Dec. 2017.
- [3] A. O. Kotb, Y.-C. Shen, X. Zhu, and Y. Huang, "iParker—A new smart car-parking system based on dynamic resource allocation and pricing," *IEEE Trans. Intell. Transport. Syst.*, vol. 17, no. 19, pp. 2637–2647, Sep. 2016.
- [4] C. Roman, R. Liao, P. Ball, S. Ou, and M. de Heaver, "Detecting on-street parking spaces in smart cities: Performance evaluation of fixed and mobile sensing systems," *IEEE Trans. Intell. Transport. Syst.*, vol. 19, no. 7, pp. 2234–2245, Jul. 2018.
- [5] A. Bagula, L. Castelli, and M. Zennaro, "On the design of smart parking networks in the smart cities: An optimal sensor placement model," *Sensors*, vol. 15, no. 7, pp. 15443–15467, Jun. 2015.
- [6] T. M. Fernández-Caramés, P. Fraga-Lamas, M. Suárez-Albela, and L. Castedo, "A methodology for evaluating security in commercial RFID systems," in *Radio Frequency Identification*. Rijeka, Croatia: InTech, 2017.
- [7] Z. Ji, I. Ganchev, M. O'Droma, L. Zhao, and X. Zhang, "A cloud-based car parking middleware for IoT-based smart cities: Design and implementation," *Sensors*, vol. 14, no. 12, pp. 22372–22393, Nov. 2014.
- [8] J. Lanza, L. Sánchez, V. Gutiérrez, J. Galache, J. Santana, P. Sotres, and L. Muñoz, "Smart city services over a future Internet platform based on Internet of Things and cloud: The smart parking case," *Energies*, vol. 9, no. 9, p. 719, Sep. 2016.
- [9] M. Garcia, P. Rose, R. Sung, and S. El-Tawab, "Secure smart parking at James Madison University via the cloud environment (SPACE)," in *Proc. IEEE Syst. Inf. Eng. Design Symp. (SIEDS)*, Apr. 2016, pp. 271–276.
- [10] K. Dolui and S. K. Datta, "Comparison of edge computing implementations: Fog computing, cloudlet and mobile edge computing," in *Proc. Global Internet Things Summit (GIoTS)*, Jun. 2017, pp. 1–6.
- [11] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog computing and its role in the Internet of Things," in *Proc. 1st Ed. MCC Workshop Mobile Cloud Comput. (MCC)*, Helsinki, Finland, Aug. 2012, pp. 13–16.
- [12] T. Fernández-Caramés, P. Fraga-Lamas, M. Suárez-Albela, and M. Díaz-Bouza, "A fog computing based cyber-physical system for the automation of pipe-related tasks in the industry 4.0 shipyard," *Sensors*, vol. 18, no. 6, p. 1961, Jun. 2018.
- [13] W. Viriyasitavat, M. Boban, H.-M. Tsai, and A. Vasilakos, "Vehicular communications: Survey and challenges of channel and propagation models," *IEEE Veh. Technol. Mag.*, vol. 10, no. 2, pp. 55–66, Jun. 2015.
- [14] A. Domazetovic, L. J. Greenstein, N. B. Mandayam, and I. Seskar, "Propagation models for short-range wireless channels with predictable path geometries," *IEEE Trans. Commun.*, vol. 53, no. 7, pp. 1123–1126, Jul. 2005.
- [15] S. Phaiboon, "Propagation path loss models for parking buildings," in *Proc. 5th Int. Conf. Inf. Commun. Signal Process.*, 2005, pp. 1348–1351.
- [16] K. Phaeua, C. Phongcharoenpanich, D. Torrungrueng, and J. Chinrungrueng, "Short-distance and near-ground signal measurements in a car park of wireless sensor network system at 433 MHz," in *Proc. 5th Int. Conf. Electr. Eng./Electron., Comput., Telecommun. Inf. Technol.*, vol. 1, May 2008, pp. 241–244.
- [17] R. Meireles, M. Boban, P. Steenkiste, O. Tonguz, and J. Barros, "Experimental study on the impact of vehicular obstructions in VANETs," in *Proc. IEEE Veh. Netw. Conf.*, Dec. 2010, pp. 338–345.
- [18] R. Sun, D. W. Matolak, and P. Liu, "Parking garage channel characteristics at 5 GHz for V2 V applications," in *Proc. IEEE 78th Veh. Technol. Conf. (VTC Fall)*, Sep. 2013, pp. 1–5.
- [19] R. He, O. Renaudin, V.-M. Kolmonen, K. Haneda, Z. Zhong, B. Ai, and C. Oestges, "Characterization of quasi-stationarity regions for vehicle-to-vehicle radio channels," *IEEE Trans. Antennas Propag.*, vol. 63, no. 5, pp. 2237–2251, May 2015.
- [20] R. Sun, D. W. Matolak, and P. Liu, "5-GHz V2 V channel characteristics for parking garages," *IEEE Trans. Veh. Technol.*, vol. 66, no. 5, pp. 3538–3547, May 2017.
- [21] T. O. Olasupo, C. E. Otero, L. D. Otero, K. O. Olasupo, and I. Kostanic, "Path loss models for low-power, low-data rate sensor nodes for smart car parking systems," *IEEE Trans. Intell. Transport. Syst.*, vol. 19, no. 6, pp. 1774–1783, Jun. 2018.
- [22] C. Tang, X. Wei, C. Zhu, W. Chen, and J. J. P. C. Rodrigues, "Towards smart parking based on fog computing," *IEEE Access*, vol. 6, pp. 70172–70185, 2018.
- [23] K. S. Awaisi, A. Abbas, M. Zareei, H. A. Khattak, M. U. S. Khan, M. Ali, I. Ud Din, and S. Shah, "Towards a fog enabled efficient car parking architecture," *IEEE Access*, vol. 7, pp. 159100–159111, 2019.
- [24] C. Lee, S. Park, T. Yang, and S.-H. Lee, "Smart parking with fine-grained localization and user status sensing based on edge computing," in *Proc. IEEE 90th Veh. Technol. Conf. (VTC-Fall)*, Sep. 2019, pp. 1–5.
- [25] O. Abdulkader, A. M. Bamhdi, V. Thayananthan, K. Jambi, and M. Alrasheedi, "A novel and secure smart parking management system (SPMS) based on integration of WSN, RFID, and IoT," in *Proc. 15th Learn. Technol. Conf. (LT)*, Feb. 2018, pp. 102–106.
- [26] E. C. Anderson, K. C. Okafor, O. Nkwachukwu, and D. O. Dike, "Real time car parking system: A novel taxonomy for integrated vehicular computing," in *Proc. Int. Conf. Comput. Netw. Informat. (ICCN)*, Oct. 2017, pp. 1–9.
- [27] T. Fernández-Caramés, P. Fraga-Lamas, M. Suárez-Albela, and M. Vilar-Montesinos, "A fog computing and cloudlet based augmented reality system for the industry 4.0 shipyard," *Sensors*, vol. 18, no. 6, p. 1798, Jun. 2018.
- [28] Y. Zhang, C.-Y. Wang, and H.-Y. Wei, "Parking reservation auction for parked vehicle assistance in vehicular fog computing," *IEEE Trans. Veh. Technol.*, vol. 68, no. 4, pp. 3126–3139, Apr. 2019.

- [29] C. Zhu, A. Mehrabi, Y. Xiao, and Y. Wen, "CrowdParking: Crowdsourcing based parking navigation in autonomous driving era," in *Proc. Int. Conf. Electromagn. Adv. Appl. (ICEAA)*, Sep. 2019, pp. 1401–1405.
- [30] S. Nguyen, Z. Salcic, and X. Zhang, "Big data processing in Fog-Smart parking case study," in *Proc. IEEE Intl Conf Parallel Distrib. Process. with Appl., Ubiquitous Comput. Commun., Big Data Cloud Comput., Social Comput. Netw., Sustain. Comput. Commun. (ISPA/IUCC/BDCloud/SocialCom/SustainCom)*, Dec. 2018, pp. 127–134.
- [31] O. T. T. Kim, N. Dang Tri, V. D. Nguyen, N. H. Tran, and C. S. Hong, "A shared parking model in vehicular network using fog and cloud environment," in *Proc. 17th Asia-Pacific Netw. Oper. Manage. Symp. (APNOMS)*, Aug. 2015, pp. 321–326.
- [32] M. Mohandes, M. Deriche, M. T. Abuelma'atti, and N. Tasadduq, "Preference-based smart parking system in a university campus," *IET Intell. Transp. Syst.*, vol. 13, no. 2, pp. 417–423, Feb. 2019.
- [33] P. Solic, R. Colella, L. Catarinucci, T. Perkovic, and L. Patrono, "Proof of presence: Novel vehicle detection system," *IEEE Wireless Commun.*, vol. 26, no. 6, pp. 44–49, Dec. 2019.
- [34] R. E. Barone, G. Tesoriere, T. Giuffrè, M. A. Morgano, and S. M. Siniscalchi, "Architecture for parking management in smart cities," *IET Intell. Transp. Syst.*, vol. 8, no. 5, pp. 445–452, Aug. 2014.
- [35] A. Shahzad, J.-Y. Choi, N. Xiong, Y.-G. Kim, and M. Lee, "Centralized connectivity for multiwireless edge computing and cellular platform: A smart vehicle parking system," *Wireless Commun. Mobile Comput.*, vol. 2018, pp. 1–23, 2018.
- [36] L. Mainetti, L. Palano, L. Patrono, M. L. Stefanizzi, and R. Vergallo, "Integration of RFID and WSN technologies in a smart parking system," in *Proc. 22nd Int. Conf. Softw., Telecommun. Comput. Netw. (SoftCOM)*, Sep. 2014, pp. 104–110.
- [37] Y.-C.-P. Chang, S. Chen, T.-J. Wang, and Y. Lee, "Fog computing node system software architecture and potential applications for NB-IoT industry," in *Proc. Int. Comput. Symp. (ICS)*, Dec. 2016, pp. 727–730.
- [38] *Huawei Smart Parking*. Accessed: Apr. 13, 2020. [Online]. Available: <http://www.huawei.com/minisite/iot/en/smart-parking.html>
- [39] M. Lauridsen, L. C. Gimenez, I. Rodriguez, T. B. Sorensen, and P. Mogensen, "From LTE to 5G for connected mobility," *IEEE Commun. Mag.*, vol. 55, no. 3, pp. 156–162, Mar. 2017.
- [40] *Wireless Smart Parking SigFox Sensor, IoT Parking Sensor RC1*. Accessed: Apr. 13, 2020. [Online]. Available: <https://partners.sigfox.com/products/iot-parking-sensor-rc1>
- [41] *NWAVE. The Future of Smart Parking*. Accessed: Apr. 13, 2020. [Online]. Available: <https://www.nwave.io/parking-technology/>
- [42] *Parkright Park App*. Accessed: Apr. 13, 2020. [Online]. Available: <https://www.westminster.gov.uk/parkright>
- [43] *InGenu Smarter City Program, Smart Parking*. Accessed: Apr. 13, 2020. [Online]. Available: <https://www.ingenu.com/solutions/smarter-city-program/>
- [44] *Area Verda*. Accessed: Apr. 13, 2020. [Online]. Available: <http://www.areaverda.cat>
- [45] M.-F. Tsai, Y. C. Kiong, and A. Sinn, "Smart service relying on Internet of Things technology in parking systems," *J. Supercomput.*, vol. 74, no. 9, pp. 4315–4338, Sep. 2018.
- [46] L. Lambrinos and A. Dosis, "DisAssist: An Internet of Things and mobile communications platform for disabled parking space management," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2013, pp. 2810–2815.
- [47] R. Vishnubhotla, P. S. Rao, A. Ladha, S. Kadiyala, A. Narmada, B. Ronanki, and S. Illapakurthi, "ZigBee based multi-level parking vacancy monitoring system," in *Proc. IEEE Int. Conf. Electro/Inf. Technol.*, May 2010, pp. 1–4.
- [48] C. Shiyao, W. Ming, L. Chen, and R. Na, "The research and implement of the intelligent parking reservation management system based on ZigBee technology," in *Proc. 6th Int. Conf. Measuring Technol. Mechatronics Autom.*, Jan. 2014, pp. 741–744.
- [49] A. Hilmani, A. Maizata, and L. Hassouni, "Designing and managing a smart parking system using wireless sensor networks," *J. Sensor Actuator Netw.*, vol. 7, no. 2, p. 24, Jun. 2018.
- [50] T. N. Pham, M.-F. Tsai, D. B. Nguyen, C.-R. Dow, and D.-J. Deng, "A cloud-based smart-parking system based on Internet-of-Things technologies," *IEEE Access*, vol. 3, pp. 1581–1591, 2015.
- [51] K. Hassoune, W. Dachry, F. Moutaouakkil, and H. Medromi, "Smart parking systems: A survey," in *Proc. 11th Int. Conf. Intell. Syst., Theories Appl. (SITA)*, Oct. 2016, pp. 1–6.
- [52] F. Al-Turjman and A. Malekloo, "Smart parking in IoT-enabled cities: A survey," *Sustain. Cities Soc.*, vol. 49, Aug. 2019, Art. no. 101608.
- [53] A. O. Kotb, Y.-C. Shen, and Y. Huang, "Smart parking guidance, monitoring and reservations: A review," *IEEE Intell. Transport. Syst. Mag.*, vol. 9, no. 2, pp. 6–16, 2017.
- [54] V. Paidi, H. Fleyeh, J. Håkansson, and R. G. Nyberg, "Smart parking sensors, technologies and applications for open parking lots: A review," *IET Intell. Transp. Syst.*, vol. 12, no. 8, pp. 735–741, Oct. 2018.
- [55] U. Raza, P. Kulkarni, and M. Sooriyabandara, "Low power wide area networks: An overview," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 2, pp. 855–873, 2nd Quart., 2017.
- [56] A. Asaduzzaman, K. K. Chidella, and M. F. Mridha, "A time and energy efficient parking system using Zigbee communication protocol," in *Proc. SoutheastCon*, Apr. 2015, pp. 1–5.
- [57] L. Azpillicueta, C. Vargas-Rosales, and F. Falcone, "Deterministic propagation prediction in vehicular environments," *IEEE Veh. Technol. Mag.*, vol. 11, no. 3, pp. 29–37, Sep. 2016.
- [58] L. Azpillicueta, M. Rawat, K. Rawat, F. Ghannouchi, and F. Falcone, "Convergence analysis in deterministic 3D ray launching radio channel estimation in complex environments," *Appl. Comput. Electromagn. Soc. J.*, vol. 29, no. 4, pp. 256–271, 2014.
- [59] C. A. Balanis, *Advanced Engineering Electromagnetics*. Hoboken, NJ, USA: Wiley, 1989.
- [60] V. K. Vyacheslav, *Handbook of Dielectric and Thermal Properties of Materials at Microwave Frequencies*. Norwood, MA, USA: Artech House, 2012.
- [61] D. A. Sánchez-Hernández, *High Frequency Electromagnetic Dosimetry*. Norwood, MA, USA: Artech House, 2009.
- [62] N. Nasri, S. Mnasri, and T. Val, "3D node deployment strategies prediction in wireless sensors network," *Int. J. Electron.*, vol. 107, no. 5, pp. 808–838, May 2020.
- [63] J. Mao, X. Jiang, and X. Zhang, "Analysis of node deployment in wireless sensor networks in warehouse environment monitoring systems," *EURASIP J. Wireless Commun. Netw.*, vol. 2019, no. 1, p. 288, Dec. 2019.
- [64] *Digi's Official Web Page for the Xbee 3 Module*. Accessed: Apr. 11, 2020. [Online]. Available: <https://bit.ly/39XeZCN>
- [65] *Ibeacon Official Web Page*. Accessed: Apr. 13, 2020. [Online]. Available: <https://developer.apple.com/ibeacon/>
- [66] *IPFS Official Web Page*. Accessed: Apr. 11, 2020. [Online]. Available: <https://ipfs.io>



MIKEL CELAYA-ECHARRI (Graduate Student Member, IEEE) received the Computer Science Engineering Degree and the master's degree in project management from the Public University of Navarre (UPNA), Pamplona, in 2011 and 2015, respectively. He is currently pursuing the Ph.D. degree in engineering of science with the Tecnológico de Monterrey, Mexico. He has worked in different research projects at Tafco Metawireless S.L., a telecommunications company placed at Navarre, Spain. He has been a Visiting Assistant at the Networks and Telecommunications Research Group, Tecnológico de Monterrey, from 2015 to 2017. His research interests include wireless sensor networks, radio propagation, dosimetric analysis, project management, and computer science.



IVÁN FROIZ-MÍGUEZ received the M.Sc. degree in computer engineering from the University of A Coruña (UDC), in 2016. From 2013 to 2019, he worked as DevOps and technical support engineer for companies, such as Inditex, Sergas, and Euskaltel. Since 2019, he has been a part of the Group of Electronic Technology and Communications (GTEC), Department of Computer Engineering, UDC, where he is attending the Ph.D. program. His current research interests include

Industry 4.0, wireless technologies, the Internet of Things (IoT), deep/machine learning, fog and edge computing, cybersecurity, distributed ledger technology (DLT), and blockchain.



LEYRE AZPILICUETA (Senior Member, IEEE) received the Telecommunications Engineering Degree, the master's degree in communications, and the Ph.D. degree in telecommunication technologies from the Public University of Navarre (UPNa), Spain, in 2009, 2011, and 2015, respectively. In 2010, she worked at the Research and Development Department, Osés RFID, as a Radio Engineer. She is currently working as an Associate Professor and a Researcher at Tecnológico de

Monterrey, Monterrey Campus, Mexico. She has over 150 contributions in relevant journals and conference publications. Her research interests include radio propagation, mobile radio systems, wireless sensor networks, ray tracing, and channel modeling. She was a recipient of the IEEE Antennas and Propagation Society Doctoral Research Award 2014, the Young Professors and Researchers Santander Universities 2014 Mobility Award, the ECSA 2014 Best Paper Award, the IISA 2015 Best Paper Award, the Best Ph.D. in 2016, awarded by the Colegio Oficial de Ingenieros de Telecomunicación, the N2Women: Rising Stars in Computer Networking and Communications 2018 Award, the ISSI 2019 Best Paper Award, and the Junior Research Raj Mittra Travel Grant 2020.



PAULA FRAGA-LAMAS (Senior Member, IEEE) received the M.Sc. degree in computer engineering from the University of A Coruña (UDC), in 2009, and the M.Sc. and Ph.D. degrees in the joint program Mobile Network Information and Communication Technologies from five Spanish universities: University of the Basque Country, University of Cantabria, University of Zaragoza, University of Oviedo, and University of A Coruña, in 2011 and 2017, respectively. She holds the

M.B.A. and master's studies in business innovation management (Jean Monnet Chair in European Industrial Economics, UDC), Corporate Social Responsibility (CSR) and social innovation (Inditex-UDC Chair of Sustainability). Since 2009, she has been with the Group of Electronic Technology and Communications (GTEC), Department of Computer Engineering (UDC). She has over 60 contributions in indexed international journals, conferences, and book chapters, and holds four patents. Her current research interests include mission-critical scenarios, Industry 4.0, the Internet of Things (IoT), cyber-physical systems (CPS), augmented/mixed reality (AR/MR), fog and edge computing, blockchain and distributed ledger technology (DLT), and cybersecurity. She has also been participating in over 30 research projects funded by the regional and national government as well as research and development contracts with private companies. She is actively involved in many professional and editorial activities, acting as a reviewer, an advisory board member, a topic/guest editor of top-rank journals, and a TPC member of international conferences.



PEIO LOPEZ-ITURRI received the Telecommunications Engineering Degree, the master's degree in communications, and the Ph.D. degree in communication engineering from the Public University of Navarre (UPNA), Pamplona, in 2011, 2012, and 2017, respectively. He has worked in ten different public and privately funded research projects. In 2019, he partly worked as a Researcher at Tafco Metawireless. He has over 150 contributions in indexed interna-

tional journals, book chapters, and conference contributions. He is affiliated with the Institute for Smart Cities (ISC), UPNA. His research interests include radio propagation, wireless sensor networks, electromagnetic dosimetry, modeling of radio interference sources, mobile radio systems, wireless power transfer, the IoT networks and devices, 5G communication systems, and EMI/EMC. He received the 2018 Best Spanish Ph.D. thesis in Smart Cities in CAEPIA 2018 (third prize), sponsored by the Spanish network on research for Smart Cities CI-RTI and Sensors (ISSN 1424-8220). He received the ECSA 2014 Best Paper Award, the IISA 2015 Best Paper Award, and ISSI 2019 Best Paper Award.



FRANCISCO FALCONE (Senior Member, IEEE) received the degree in telecommunication engineering and the Ph.D. degree in communication engineering from the Universidad Pública de Navarra (UPNA), Spain, in 1999 and 2005, respectively. From February 1999 to April 2000, he was a Microwave Commissioning Engineer at Siemens-Italtel, deploying microwave access systems. From May 2000 to December 2008, he was a Radio Access Engineer at Telefónica Móviles, perform-

ing radio network planning and optimization tasks in mobile network deployment. In January 2009 to May 2009, as a Co-Founding Member, he was the Director of Tafco Metawireless, a spin-off company from UPNA. In parallel, he was an Assistant Lecturer with the Electrical and Electronic Engineering Department, UPNA, from February 2003 to May 2009, where he becomes an Associate Professor with the EE Department, in June 2009, and the Department Head, from January 2012 to July 2018. From January 2018 to May 2018, he was a Visiting Professor with the Kuwait College of Science and Technology, Kuwait. He is also affiliated with the Institute for Smart Cities (ISC), UPNA, which hosts around 140 researchers. He is currently acting as the Head of the ICT Section. His research interest includes computational electromagnetics applied to the analysis of complex electromagnetic scenarios, with a focus on the analysis, design, and implementation of heterogeneous wireless networks to enable context-aware environments. He has over 500 contributions in indexed international journals, book chapters, and conference contributions. He received the CST 2003 and CST 2005 Best Paper Award, the Ph.D. Award from the Colegio Oficial de Ingenieros de Telecomunicación (COIT), in 2006, the Doctoral Award from UPNA, in 2010, the First Juan Gomez Peñalver Research Award from the Royal Academy of Engineering of Spain, in 2010, the XII Talgo Innovation Award 2012, the IEEE 2014 Best Paper Award, in 2014, the ECSA-3 Best Paper Award, in 2016, and the ECSA-4 Best Paper Award, in 2017.



TIAGO M. FERNÁNDEZ-CARAMÉS (Senior Member, IEEE) received the M.Sc. and Ph.D. degrees in computer science from the University of A Coruña (UDC), Spain, in 2005 and 2011, respectively.

Since 2016, he has been working as an Assistant Professor of electronic technology at UDC. Since 2005, he has been working with the Department of Computer Engineering, UDC, through different Predoctoral Scholarships, from 2005 to 2009, and as an Interim Professor, from 2007 to 2016. His current research interests include the IoT/IIoT systems, RFID, wireless sensor networks, augmented reality, embedded systems, and blockchain, as well as the different technologies involved in the Industry 4.0 paradigm. In such fields, he has contributed to 40 papers for conferences, to 35 articles for JCR-indexed journals, and to two book chapters. Due to his expertise in the previously mentioned fields, he has acted as a peer reviewer and a guest editor for different top-rank journals, and as a project reviewer for national research bodies from Austria, Croatia, Latvia or Argentina.

...