

Proceeding Paper

A Bluetooth 5 Opportunistic Edge Computing System for Vehicular Scenarios [†]

Ángel Niebla-Montero ^{1,2,*} , Iván Froiz-Míguez ^{1,2}  and Paula Fraga-Lamas ^{1,2,*}
and Tiago M. Fernández-Caramés ^{1,2} 

¹ Department of Computer Engineering, Faculty of Computer Science, Universidade da Coruña, 15071 A Coruña, Spain

² Centro de Investigación CITIC, Universidade da Coruña, 15071 A Coruña, Spain

* Correspondence: angel.niebla@udc.es (Á.N.-M.); paula.fraga@udc.es (P.F.-L.);
Tel.: +34-981167000 (ext. 6051) (P.F.-L.)

[†] Presented at the 9th International Electronic Conference on Sensors and Applications, 1–15 November 2022;
Available online: <https://ecsa-9.sciforum.net/>.

Abstract: The limitations of many IoT devices in terms of storage, computing power and energy consumption require them to be connected to other devices when performing computationally intensive tasks, as happens with IoT systems based on edge computing architectures. However, the lack of wireless connectivity in the places where IoT nodes are deployed or through which they move is still a problem. One of the solutions to mitigate this problem involves using opportunistic networks, which provide connectivity and processing resources efficiently while reducing the communications traffic with remote clouds. Thus, opportunistic networks are helpful in situations when wireless communication coverage is not available, as occurs in certain rural areas, during natural disasters, in wars or when other factors cause network disruptions, as well as in other IoT scenarios in which the cloud becomes saturated (for example, due to an excessive amount of concurrent communications or when denial-of-service (DoS) attacks occur). This article presents the design and initial validation of a novel opportunistic edge computing (OEC) system based on Bluetooth 5 and the use of low-cost single-board computers (SBCs). After describing the proposed OEC system, experimental results are presented for a real opportunistic vehicular IoT scenario. Specifically, the latency and packet loss are measured thanks to the use of an experimental testbed made of two separate IoT networks (each conformed by an IoT node and an OEC gateway): one located in a remote office and another one inside a moving vehicle, which was driven at different vehicular speeds. The obtained results show average latencies ranging from 716 to 955 ms with packet losses between 7% and 27%. As a result, the developed system is useful for providing opportunistic services to moving IoT nodes with relatively low latency requirements.

Keywords: IoT; Bluetooth 5; opportunistic edge computing; edge computing; vehicular scenarios



Citation: Niebla-Montero, Á.; Froiz-Míguez, I.; Fraga-Lamas, P.; Fernández-Caramés, T.M. A Bluetooth 5 Opportunistic Edge Computing System for Vehicular Scenarios. *Eng. Proc.* **2022**, *27*, 66. <https://doi.org/10.3390/ecsa-9-13214>

Academic Editor: Stefano Mariani

Published: 1 November 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

According to some forecasts [1], 75 billion Internet of Things (IoT) devices will be operational by 2025. Many of these IoT devices will be resource constrained, with limited storage, computing power and power consumption, thus forcing them to rely on other remote devices to perform compute-intensive tasks. In addition, smart IoT devices can be anywhere and may need to be interconnected, but, in some areas, there is not always wireless communications coverage. To address the previous issues, opportunistic communications systems can be beneficial, as they allow for collaboration among IoT devices so that they can share resources and services when they are available [2]. Furthermore, opportunistic collaboration becomes advisable in cloud computing environments, which are not intrinsically energy efficient or secure and have limitations when it comes to large-scale IoT deployments [3].

Part of the previously mentioned constraints has been addressed by the adoption of novel communications architectures, such as those based on edge computing [4], which, when combined with the notion of opportunistic systems, give rise to the opportunistic edge computing (OEC) paradigm. This paper presents a communications architecture that makes use of vehicular Bluetooth 5-based IoT nodes while leveraging the OEC paradigm. Such a solution is evaluated in terms of latency and packet loss.

2. State of the Art: Opportunist Mobile Edge Computing Systems

A significant amount of computing power can be provided by SBCs and other similar IoT devices thanks to recent technical advancements, enabling them to serve as endpoints or gateways in both edge computing systems and new fog computing architectures [5]. Different OEC systems have already been implemented for fields related to mobility [6] and autonomous vehicles [7]. However, the majority of these IoT systems still have one crucial requirement: in order to use certain remote cloud services, they must have access to the Internet.

Some authors have studied the use of opportunistic solutions in scenarios with mobile nodes that require edge computing services. For instance, in [6] the authors proposed the use of proximal mobile edge servers. Specifically, in their paper, the authors evaluate and compare the performance of mobile devices when running activity detection applications that involve mobile devices, edge servers and cloud computing servers. For such a purpose, an Android-based mobile application was developed, and, for mobile edge servers, the AllJoyn framework [8] was used for device discovery, peer-to-peer (P2P) network creation and data download.

Another interesting development is presented in [7], in which the authors propose the deployment of a UAV-based fog computing network. The described system makes use of the advantages of applying the concepts of fog computing as well as the mobility of UAVs, which enables the fog network to be placed where it is required.

In contrast to the previously analyzed works, the architecture proposed in this article provides the following features together:

- The usage of mobile fog computing enables the provision of edge computing services, thus reducing the time response to IoT nodes via low-cost gateways dispersed throughout large environments to provide opportunistic communications.
- The use of Bluetooth 5 enables mobile data exchanges with a greater range and less energy consumption than other technologies, such as WiFi, GSM or 4G/5G. In addition, Bluetooth 5 does not impose any mobility or transmission constraints, unlike other technologies such as LoRa, which is limited in terms of transmission speed and packets per time unit.
- The proposed architecture has been designed to be simultaneously decentralized, distributed, scalable and modular.
- The designed architecture also incorporates additional features such as resource sharing, data routing and IoT node discovery.

3. Design and Implementation of the System

3.1. Communications Architecture

The proposed OEC IoT architecture is shown in Figure 1, which consists of three layers:

- The IoT Network Layer. This layer consists of the several IoT networks that make up the system (IoT networks A and B in Figure 1), whose devices can exchange data with the higher layers. As a result, IoT nodes utilize sensors and actuators to monitor and interact with their surroundings. Such nodes can be either static (i.e., the ones represented in IoT Network A in Figure 1) or mobile (i.e., the node that belongs to IoT Network B in Figure 1). In any case, the nodes exchange information with the opportunistic layer, which provides them with multiple edge computing services. What distinguishes the IoT Network Layer from others present in generic IoT architectures is the fact that communications with the upper layer are not always

available, so they need to be carried out in an opportunistic way (i.e., IoT devices need to discover when the upper layer devices are available to make use of its services).

- The OEC Smart Gateway layer. This layer is made up of fog computing gateways that may supply services opportunistically to the IoT Network Layer. Such services are characterized by having lower latency than the ones provided by a remote cloud on the internet due to the physical proximity of the gateway to the IoT nodes. Moreover, in the proposed OEC architecture, the data from the IoT devices are stored in a distributed hash table (DHT) network shared by the gateways.
- The Cloud layer. This layer is in charge of providing services that the OEC smart gateways cannot deliver, such as compute-intensive processing or large-scale data storage.

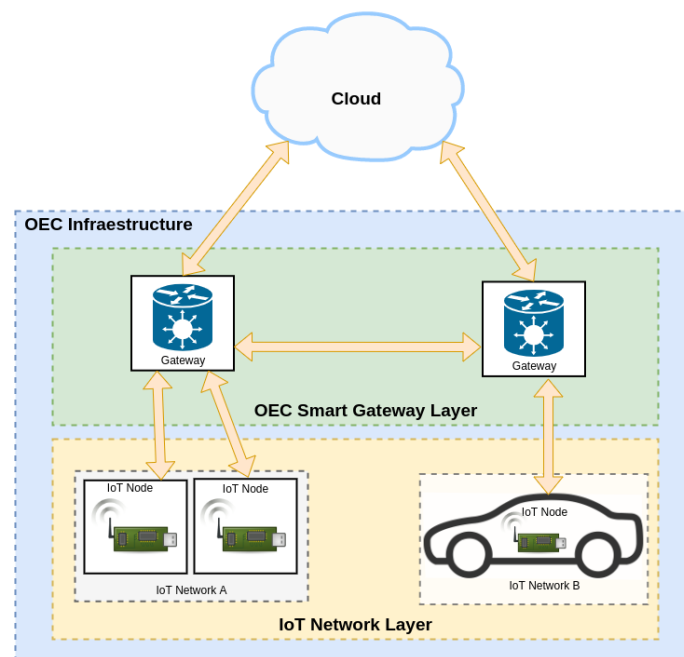


Figure 1. Designed OEC communications architecture.

3.2. Implementation

The implementation of the proposed communications architecture relies on Bluetooth 5 for the communications between the OEC IoT nodes and the smart gateways. The communications between the gateways and the cloud are provided through WiFi/4G networks. In addition, the three layers of the architecture provide different smart services, which are the most important for opportunistic communications, such as the ones delivered by the OEC Smart Gateway layer: peer discovery, peer routing, data routing and resource sharing. Furthermore, the cloud layer provides a routing service for communicating the different networks whose gateways are not directly connected.

The most critical part of implementing OEC systems is related to the OEC Smart Gateway Layer. For this purpose, libp2p [9] was chosen, which, thanks to its modularity, can be easily adapted to various OEC architectures. Libp2p started as part of the IPFS project [10] and currently has multiple implementations (e.g., in Go, JavaScript, Rust, Python and C++) that provide flexible solutions for packet transport, security, P2P routing and content discovery. Since libp2p is transport agnostic, it allows OEC developers to select the most appropriate transport protocol for each application. For the development presented in this paper, libp2p is the basis for the main implemented features:

- Peer discovery and peer routing. Peer discovery allows for discovering peer addresses thanks to the knowledge provided by the other peers. The peer routing subsystem exposes an interface to identify the peers to which the message should be routed. The protocol used was Kademlia DHT [11], which is based on DHTs.

- Data routing. Data routing uses mechanisms to forward messages to peers if the receiving peer is not within range of the sending peer. Kademlia DHT was also used for implementing this feature.

4. Experiments

4.1. Experimental Setup

To evaluate the performance of the proposed system, an experimental testbed was built. This testbed consisted of two IoT nodes based on an SBC (Raspberry Pi 3B) with a Nordic nRF52840 development kit connected through the serial port to add Bluetooth Mesh support. Nodes were flashed with the developed software and then provisioned with the Nordic nRF Mesh app for Android. Moreover, two OEC gateways were built with a Raspberry Pi 3B+ and a Raspberry Pi Zero. Similar to the OEC IoT nodes, the gateways used two Nordic nRF52840 devkits to provide Bluetooth Mesh connectivity (all Bluetooth devkits were provisioned on the same network to allow them to intercommunicate).

With such a testbed, tests were performed to measure the latency and packet losses of the developed OEC system when sending data from a static IoT node located indoors (in an office) to a mobile node located inside a car. Specifically, the latency was measured as the time it took the system to perform the following steps:

- The data were first sent by using Bluetooth 5 (BLE) from the static IoT node to the smart OEC gateway located locally in the office. The gateway and the static IoT node were separated by roughly one meter and had a direct line of sight (LoS). Both devices were connected through WiFi and used an optic fiber router to connect the Internet to the cloud.
- Then, the indoor gateway received the data from the static IoT node through its serial port (to which a Bluetooth 5 development kit was attached) and uploaded them to the DHT network.
- Next, the OEC gateway that operated outdoors in the same opportunistic network in which the mobile node was located, collected the data sent through the DHT network. The outdoor gateway was located on a straight road (shown in Figure 2) and at a height of approximately 1.80 m. To make the communication between the indoor and outdoor gateways possible, it was necessary for the cloud to act as a relay. For the tests performed in this article, the cloud was located in a remote server whose ping was about 50 ms when using the fiber optic router of the office and roughly 150 ms for the 4G smartphone carried in the car.
- Finally, the outdoor gateway sent the message to the mobile IoT node through Bluetooth 5 (using BLE). The mobile node was carried inside a car as shown in Figure 3. The car was driven at different constant speeds between points A and B indicated in Figure 2 (the itinerary started at point A, reached point B and returned to point A). It must be noted that, before beginning this set of experiments, it was verified that 100% of the packets were received at the vehicular IoT node when it was static in point A inside the vehicle. After this verification, tests were performed at four vehicular speeds (at 30, 50, 70 and 90 km/h) while evaluating the packet reception loss and end-to-end latency.

The source code for replicating all of the performed experiments is available online [12], thus allowing any researcher to use and extend the developed OEC system.



Figure 2. Testing scenario.



Figure 3. IoT node inside the vehicle in the test scenario.

4.2. Obtained Results

Figure 4a shows the obtained latency results through time (for consecutive packets) for the different evaluated vehicular speeds. As it can be observed, latency remains almost constant, independently of the speed. The lowest average latency was 716 ms for 30 km/h, while the highest one was 955 ms (for 70 km/h). For 50 km/h and 90 km/h, the average latencies were 893 and 786 ms, respectively.

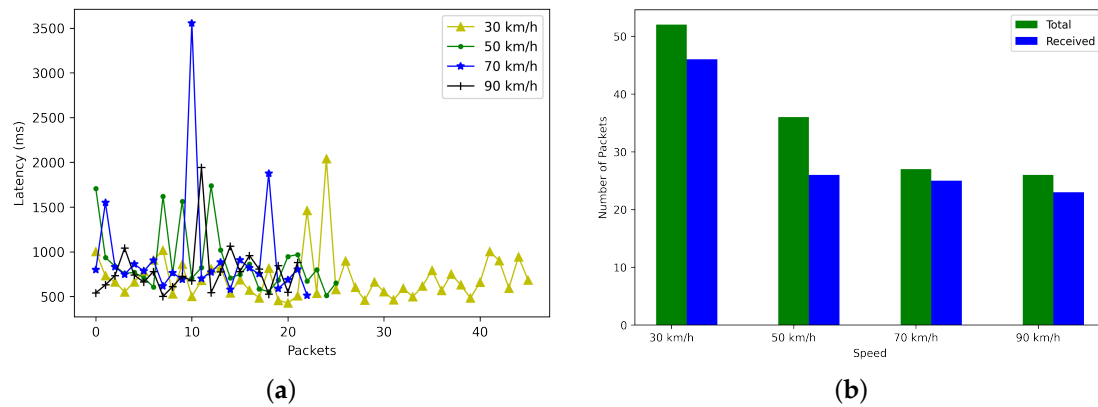


Figure 4. Obtained results. (a) End-to-end latency at different speeds. (b) Total packets sent versus packets received.

Regarding the packet loss results, Figure 4b shows that, for each vehicular speed, the total number of packets sent by the indoor IoT node (in green) and the ones actually received by the opportunistic mobile IoT node (in blue). As can be observed, the slower the vehicular speed, the higher the number of packets that could be sent and received, since it took more time for the vehicle to drive through the established itinerary. Moreover, it can be seen that the speed at which there were the least lost packages was 70 km/h (7%). In the case of 30 and 90 km/h, 11% of the total transmitted packages were lost. In the case of 50 km/h, 27% of the packets were lost, which is related to a temporary communications failure that occurred during the experiments (note that this problem cannot be observed in Figure 4a, since it only shows the latencies for the received packets). Despite this failure, the results for 50 km/h are shown to illustrate the fact that non-opportunistic technologies such as WiFi and 4G can impact an OEC system's performance.

The analysis of the previous figures allows us to conclude that the tested speeds have a limited impact on latency and on the number of lost packets. As a consequence, the proposed opportunistic system is suitable for mobile IoT scenarios similar to the one tested and those that require a relatively low number of lost packets and under one-second latency.

5. Conclusions

This article presented the design, implementation and preliminary evaluation of a novel OEC system based on Bluetooth 5 nodes for mobile IoT scenarios. After describing the communications architecture, the test results were presented to demonstrate its feasibility. These results show that it is possible to provide opportunistic communications between a static and a mobile IoT node carried in a vehicle at speeds between 30 and 90 km/h, providing under one-second end-to-end latency values (between 716 and 955 ms) and losing a relatively small number of packets (between 7% and 27%). These preliminary findings suggest that the proposed opportunistic network is a valid solution for implementing opportunistic vehicular IoT applications for scenarios where wireless communications are not continuously available.

Author Contributions: Conceptualization, T.M.F.-C.; methodology, P.F.-L. and T.M.F.-C.; investigation, Á.N.-M., I.F.-M., P.F.-L. and T.M.F.-C.; writing—original draft preparation, Á.N.-M., I.F.-M. and T.M.F.-C.; writing—review and editing, Á.N.-M., I.F.-M., P.F.-L. and T.M.F.-C.; supervision, T.M.F.-C.; project administration, T.M.F.-C.; funding acquisition, T.M.F.-C. All authors read and agreed to the published version of the manuscript.

Funding: This work has been supported by grants ED431C 2020/15 and ED431G 2019/01 (to support the Centro de Investigación de Galicia "CITIC") funded by Xunta de Galicia and European Regional Development Funds through ERDF Galicia 2014–2020; and by grant PID2020-118857RA-I00 (ORBALLO) funded by MCIN/AEI/10.13039/501100011033.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. HIS. Internet of Things (IoT) Connected Devices Installed Base Worldwide from 2015 to 2025. Available online: <https://bit.ly/3n6aQDO> (accessed on 13 September 2022). (In Billions)
2. Gubbi, J.; Buyya, R.; Marusic, S.; Palaniswami, M. Internet of Things (IoT): A vision, architectural elements, and future directions. *Future Gener. Comput. Syst.* **2013**, *29*, 1645–1660. [[CrossRef](#)]
3. Kshetri, N. Can Blockchain Strengthen the Internet of Things? *IT Prof.* **2017**, *19*, 68–72. [[CrossRef](#)]
4. Fraga-Lamas, P.; Lopes, S.I.; Fernández-Caramés, T.M. Green IoT and Edge AI as Key Technological Enablers for a Sustainable Digital Transition towards a Smart Circular Economy: An Industry 5.0 Use Case. *Sensors* **2021**, *21*, 5745. [[CrossRef](#)] [[PubMed](#)]
5. Froiz-Míguez, I.; Lopez-Iturri, P.; Fraga-Lamas, P.; Celaya-Echarri, M.; Blanco-Novoa, Ó.; Azpilicueta, L.; Falcone, F.; Fernández-Caramés, T.M. Design, Implementation, and Empirical Validation of an IoT Smart Irrigation System for Fog Computing Applications Based on LoRa and LoRaWAN Sensor Nodes. *Sensors* **2020**, *20*, 6865. [[CrossRef](#)] [[PubMed](#)]
6. Habib ur Rehman, M.; Batool, A.; Salah, K. The Rise of Proximal Mobile Edge Servers. *IT Prof.* **2019**, *21*, 26–32. [[CrossRef](#)]
7. Mohamed, N.; Al-Jaroodi, J.; Jawhar, I.; Noura, H.; Mahmoud, S. UAVFog: A UAV-based fog computing for Internet of Things. In Proceedings of the 2017 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computed, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCOM/IOP/SCI), San Francisco, CA, USA, 4–8 August 2017; pp. 1–8.
8. Alliance, A. Alljoyn Framework, Linux Found. Collaborative Projects. Available online: <https://openconnectivity.org/technology/reference-implementation/alljoyn/> (accessed on 13 September 2022).
9. Guidi, B.; Michienzi, A.; Ricci, L. A libP2P Implementation of the Bitcoin Block Exchange Protocol. In Proceedings of the 2nd International Workshop on Distributed Infrastructure for Common Good (DICG'21), Virtual, 6–10 December 2021 ; pp. 1–4.
10. IPFS. The Interplanetary File System (IPFS). Available online: <https://ipfs.io/> (accessed on 13 September 2022).
11. Maymounkov, P.; Mazières, D. Kademlia: A Peer-to-Peer Information System Based on the XOR Metric. In *Peer-to-Peer Systems. IPTPS 2002. Lecture Notes in Computer Science*; Druschel, P., Kaashoek, F., Rowstron, A., Eds.; Springer: Berlin/Heidelberg, Germany, 2002; Volume 2429.
12. Official Gitlab Repository of the ORBALLO Project. Available online: https://gitlab.com/orballo_project/opportunistic-ble-mesh (accessed on 13 September 2022).