# Harnessing Communication Heterogeneity: Architectural Design, Analytical Modeling, and Performance Evaluation of an IoT Multi-Interface Gateway

Emanuele Pagliari, Luca Davoli, *Member, IEEE,* and Gianluigi Ferrari, *Senior Member, IEEE.*

***Abstract*—Given the massive deployment of Internet of Things (IoT) applications over the last decade, the need for gateways able to efficiently route information flows across multiple heterogeneous networks has emerged, bringing new challenges. Therefore, the design and implementation of IoT gateways is crucial. In this paper, with reference to the architecture of a prototypical Multi-Interface Gateway (MIG) (based on Commercial-Off-The-Shelf, COTS, devices), we evaluate its performance: (i) analytically, through an innovative Markov chain-based model; (ii) by simulation, with a Python simulator; (iii) experimentally, through the (starting) COTS device-based prototype. In detail, the MIG is equipped with heterogeneous wireless communication interfaces (namely LoRaWAN, BLE, cellular 4G Cat. 4, and IEEE 802.11 Wi-Fi 2.4 GHz) and is applicable to multiple IoT scenarios. The obtained simulation and experimental results show the validity of the proposed analytical model. Further improvements of the proposed framework are eventually discussed.**

***Index Terms*—Internet of Things, Multi-Interface Gateway, Markov Chain, Analytical Modeling, Performance Evaluation, LoRaWAN, BLE, Wi-Fi, Cellular.**

## I. Introduction

AMONG the technologies and paradigms which have recently changed the way of thinking about the Internet, a key role is played by the Internet of Things (IoT), which entails the development of a multitude of nodes equipped with sensors/actuators and different (often wireless) communication interfaces. To this regard, thanks to the large number of contexts and scenarios which IoT can be applied to (e.g., smart cities, Industry 4.0, smart farming), IoT entities are often organized (from a network point of view) as systems of systems.

In general, a reference IoT scenario encompasses heterogeneous communication protocols from short-range (usually with high data transmission rate) to long-range (typically with low data transmission rate) [1], as shown in Fig. 1. Due to this heterogeneity and the need to "federate" IoT networks, one of the main challenges is to allow a transparent (from a high level point of view) interaction between different networks. Therefore, a key element is a *gateway*, whose features and capabilities play a crucial role in enabling IoT applications by efficiently routing data.

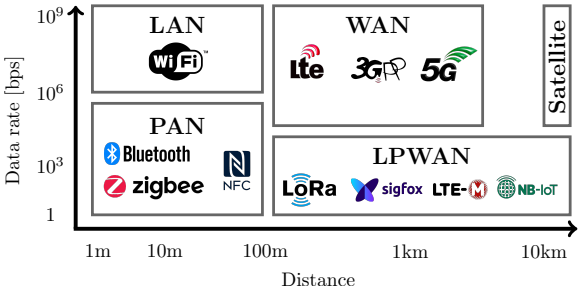



Fig. 1: Communication protocols of interest for heterogeneous IoT scenarios.

Gateways are typically multi-layered (with reference to the ISO/OSI and TCP-IP layered protocol stacks) network entities which support: intelligent information flows' routing; data processing; and queuing mechanisms (this is crucial, as we will see, in the presence of heterogeneous communication interfaces). On the other hand, heterogeneous networks, with a large number of nodes collecting data (e.g., through sensors) from the surrounding environment, exploit proper data processing techniques to extract relevant (non-redundant) information and made its transfer compatible with low data-rate protocols (e.g., Long Range Wide Area Network, LoRaWAN).

In this paper, we first present the architecture of a *Commercial-Off-The-Shelf (COTS) device-based prototypical implementation* of an enhanced scalable and modular Multi-Interface Gateway (MIG) with four heterogeneous communication interfaces, namely: IEEE 802.11 (2.4 GHz) Wi-Fi; Bluetooth Low Energy (BLE); LoRaWAN; and cellular 4G (Cat. 4). Then, in order to accurately predict the experimental performance of our COTS device-based MIG in various scenarios, we derive a *novel Markov chain-based queuing model* of the MIG. Various input distributions for the packet inter-arrival times are considered, in order to evaluate their impact on the system performance. Finally, we develop a *Python-based software simulator* of the MIG in order to verify the analytical performance predicted by our Markov chain-based queuing model, thus paving the way to a comparison between analytical and simulation results, as well as to a deeper analysis of the limitations of the used communication

E. Pagliari, L. Davoli and G. Ferrari are with the Internet of Things (IoT) Lab, Department of Engineering and Architecture, University of Parma, 43124 Parma, Italy, with the National Interuniversity Consortium for Informatics (CINI), 00185 Rome, Italy, and with the National Inter-University Consortium for Telecommunications (CNIT), 43124 Parma, Italy. E-mail: {emanuele.pagliari, luca.davoli, gianluigi.ferrari}@unipr.it

Manuscript received August dd, 2023; revised September dd, 2023.

protocols. Unlike classical approaches, which move from theoretical modeling to experimental validation, in this study we move from a COTS device-based MIG (with a specific architecture) to its analytical and simulation models.

Eventually, the performance of the MIG prototype, equipped with pairwise bidirectional smart data routing mechanisms between the four aforementioned communication interfaces (namely: Wi-Fi, BLE, 4G, and LoRaWAN), is experimentally evaluated and compared with the analytical and simulation performances. The proposed MIG architecture supports internal data pre-processing to make high throughput information flows (from the BLE, Wi-Fi, and cellular interfaces) compatible with the constrained transmission capabilities of LoRaWAN. In particular, proper parsing, adaptation, and conversion rules are used to minimize resource consumption, thus guaranteeing the highest possible (sustainable) throughput. The modular design of the MIG allows (i) to make use of constrained protocols (namely, LoRaWAN) feasible and (ii) to introduce predictive processing mechanisms (e.g., based on Artificial Intelligence, AI, techniques) to be applied to the data flows (e.g., to optimize resource utilization).

The main contributions of this work can be summarized as follows.

- We present the architecture of a COTS device-based prototypical implementation of a modular MIG.
- We evaluate the experimental performance of the MIG with heterogeneous communication interfaces, namely: IEEE 802.11 Wi-Fi, BLE, LoRaWAN, and cellular 4G.
- We derive a novel Markov chain-based queuing model of the MIG, to analytically estimate its performance as a function of the input traffic data distribution.
- We consider different input distributions for the packet inter-arrival times, in order to evaluate their impacts on resource utilization of the MIG.
- We develop a Python simulator to verify the analytical performance predicted by the proposed Markov chain-based queuing model.
- We compare analytical, simulation, and experimental performance results in the presence of pairwise bidirectional smart data routing mechanisms between the communication interfaces.

The rest of the paper is organized as follows. In Section II, we overview existing solutions and comment on related research works available in the literature. Section III introduces and characterizes the internal MIG architecture, with reference to a specific prototypical implementation. In Section IV, we derive a Markov chain-based queuing model to describe the behavior of the MIG. In Section V, we evaluate the performance predicted by the analytical queuing model (in the presence of a representative input data distribution) and compare it with that of a Python-based simulator. In Section VI, we rely on the validated analytical model to analyze the impact of the packet inter-arrival time distribution. In Section VII, this comparison is extended toward experimental performance evaluation of the MIG prototype. Section VIII is dedicated to discussing possible improvements. Finally, in Section IX conclusions are drawn.

## II. Related Works

The heterogeneity of IoT scenarios and applications requires to carefully consider communication aspects, thus looking for potential *trade-offs*. The communication protocols shown in Fig. 1 are strongly heterogeneous in terms of data rate and applicability. In detail, widely adopted wireless communication protocols are: BLE for *short-range* communications [2]; IEEE 802.11 (Wi-Fi) and ZigBee [3] for *medium-range* communications; and NarrowBand-IoT (NB-IoT), cellular (e.g., 4G LTE), LoRaWAN, and Sigfox [4] for *long-range* communications. In order to make these protocols interoperable and manageable by a multi-interface node, their characteristics have to be carefully taken into account [5]. Typically, two entities support interoperability: bridges [6] and gateways [7]. Gateways should be preferred, with respect to bridges, in IoT scenarios, as they support intelligent data routing, queuing policies, and data analysis mechanisms for heterogeneous (in terms of resources and constraints) communication interfaces [8]. Hence, the design and deployment of intelligent gateways plays a crucial role, especially in the case of large heterogeneous networks requiring "intelligent" data pre-processing at the edge.

Even though there are different gateways available on the market (see, for example, [9]), they typically present relevant drawbacks, such as: high costs; limited communication interfaces available on the devices; and, most of all, "closed source" nature. This "closeness" prevents: the introduction of new routing/communication/processing features; the connection of new communication interfaces; and the implementation of complex network scenarios—e.g., with intelligent routing policies based on traffic offloading, as well as enhanced decision mechanisms (e.g., through the adoption of traffic engineering paradigms [10,11]). Therefore, the design and implementation of modular and open gateway architectures, like the one proposed in this work, may lead to a faster deployment of new IoT applications based, for example, on the integration of heterogeneous Wireless Sensor Networks (WSNs).

Focusing on the analytical characterization of a gateway, in [12] a hidden Markov model-based approach for latency-aware and energy-efficient computation offloading in fog computing-like scenarios is proposed. This analytical model is exploited to find the best possible candidate layer which application modules can be executed at. An interesting optimization tool for constrained IoT applications is then proposed. Markov queuing models have also been adopted in [13,14], looking for a trade-off between energy consumption and latency in task assignment in next-generation systems, as well as in [15]–[18].

The research works mentioned in the previous paragraph focus on the use of Markov chain-based models for the characterization and optimization of a few network parameters only for specific IoT system aspects (e.g., energy consumption). Unlike our work, the above works are not applicable to a MIG supporting heterogeneous communication protocols and do not allow to derive a corresponding Markov chain-based model. More in detail, most of the above literature works focus on modeling a single communication protocol according to its specific (low-level) protocol rules. For example, the

analytical model of the LoRaWAN protocol proposed in [19] focuses on LoRaWAN UpLink (UL) latency, collision rate, and throughput, assuming that the packet inter-arrival time has an exponential distribution. An extended model is proposed in [20], where both UL and DownLink (DL) transmissions, together with other relevant features of the LoRaWAN protocol, are taken into account. Unlike [19,20], the goal of our work is not the investigation of the behavior of each protocol *per se*, i.e., without considering the others. Rather, we want to model the behavior of a communication protocol and investigate its limits depending on the constraints imposed by its interaction with other protocols, e.g., when the input information flow comes from another communication interface the MIG is equipped with.

As for the LoRaWAN protocol case, most of the BLE scientific literature models the BLE behavior considering low-layers' protocol details. In [21], an analytical model of BLE advertising channels is proposed for several applications (such as localization or data advertisement in IoT use cases).

Finally, in [22] a high-level characterization of a gateway, to be used for several IoT applications, is proposed. In detail, the authors highlight how the interactions among multiple devices can be modeled in large scale applications, thus optimizing the deployment of gateways to guarantee efficient and adaptive communications in several scenarios. However, in [22] no internal modeling of such gateways is proposed, focusing more on the high layer interactions among multiple gateways, rather than internal packet management between heterogeneous protocols, as presented and discussed in our work.

## III. MIG Architecture

The prototypical IoT-oriented MIG implementation proposed in this paper is based on a Raspberry Pi 4 (RPi4) Single Board Computer (SBC), equipped with an additional Dragino LoRaWAN hat [23] and an external 4G LTE Cat. 4-enabled Huawei E3372 USB dongle connected to one USB port of the RPi4. Owing to this configuration, the MIG may operate with the following connectivities: BLE and Wi-Fi (through the communication interfaces natively provided by the RPi); LoRaWAN (through the external Dragino hat); and cellular (through the Huawei modem). In Fig. 2, we show: (i) the COTS device-based MIG prototype and (b) its corresponding internal architecture. From a software point of view, the MIG architecture is based on two high-layer types of modules, namely: (i) a dedicated software routine for each available communication interface, and (ii) an internal routing module, denoted as Smart Data Broker (SDB), that, jointly operating with an internal Message Queue Telemetry Transport (MQTT) broker, is in charge of handling multiple MQTT topics and is used for temporary internal traffic packets' queuing and management purposes. We now describe in more detail each block.

The main role of the software routine associated with each communication interface equipping the MIG is: (i) to properly handle the tasks which may be required by the corresponding communication protocol (e.g., packet processing, payload constraints' validation, transmission policies' adoption, services' execution, UL and DL operation handling, etc.) and (ii) to optimize, convert, and forward data (e.g., coming from *on-field* end nodes connected to the MIG) toward the right interface-specific MQTT topic.

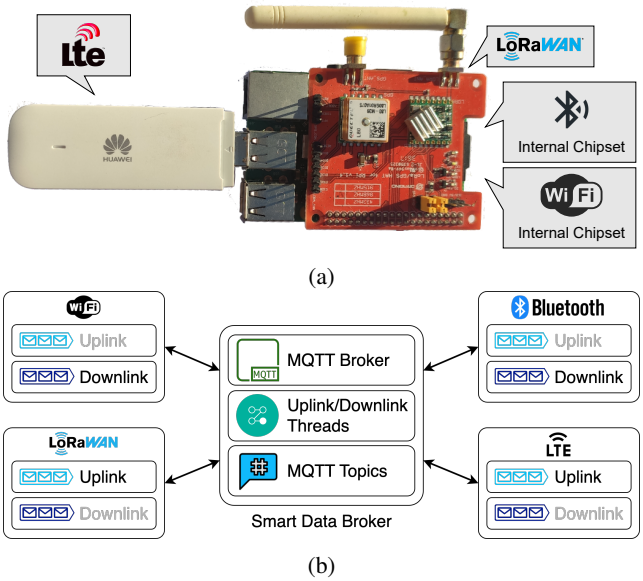



Fig. 2: MIG: (a) COTS device-based prototype and (b) high-layer architecture.

In the following, we consider incoming packets (also denoted as DL packets, from end nodes toward the MIG) only through BLE and Wi-Fi interfaces. This is motivated by the fact that the LoRaWAN interface is practically used only for UL communications[1] and the same applies also to the cellular interface (as IoT nodes may rarely be equipped with a cellular interface to communicate with the MIG). In other words, the envisioned behavior of the MIG is that of collecting data from nearby Wi-Fi or BLE IoT nodes (DL data traffic) and forward this traffic (properly processed) across LoRaWAN or cellular networks (UL data traffic).

Suppose that the data collected from *on-field* end nodes have the illustrative structure shown in Fig. 3 (at the top). Once received by the MIG, these packets will then be processed by the proper DL interface' handler. The selected handler then "appends" a header field, denoted as $ID_{\mathrm{ROUTE}}$, specifying the routing rule which should be applied by the SDB routing system (e.g., forward to the LoRaWAN interface). Furthermore, in the case of a packet coming from Wi-Fi or BLE nodes (as discussed in the previous paragraph), this packet is extended to include the following fields: (i) the source node's MAC address $SRC_{\mathrm{MAC}}$ and (ii) a separator field. The final packet structure will thus possess a general form with a header $H$ and a payload $PKT$.

Looking at the internal routing mechanism, the SDB relies on a standard MQTT broker handling different MQTT topics, each of them managed by a proper UL/DL handler in charge of processing input packets and forwarding them to the right

[1]Class C LoRaWAN nodes [24] can also receive DL traffic from the LoRaWAN's Application Server (AS) through LoRaWAN gateways, but they are typically not used.

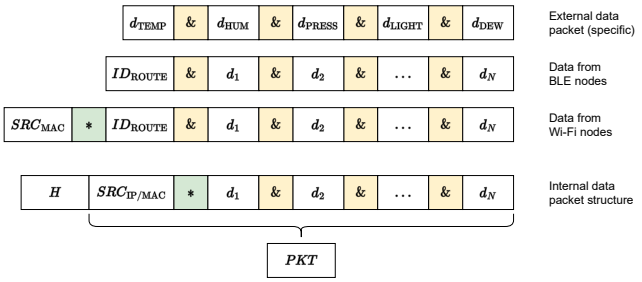



Fig. 3: Data packets from the different communication interfaces adopted in the proposed MIG.

TABLE I: Routing rules available in the proposed IoT MIG.

| Field Value | Routing Destination |
|---|---|
| MAC Address | Specific MAC address of the BLE device |
| IP Address | Specific IP address of the Wi-Fi device |
| 0 | LoRaWAN communication interface |
| 1 | Wi-Fi server's default IP address |
| 2 | BLE server's default MAC address |
| 3 | LoRaWAN communication interface and Wi-Fi server's default IP address |
| 4 | LoRaWAN communication interface and BLE server's default MAC address |
| 5 | Wi-Fi server's default IP address and BLE server's default MAC address |
| 6 | All available communication interfaces, on their default server IP addresses |
| 7 | Server reachable through the Cellular Network |

output interface's UL queue according to a First-In-First-Out (FIFO) policy. Then, the interfaces' servers, being subscribers to the MQTT UL topics of their corresponding communication interfaces, (i) are notified by the SDB with the updated Data Transfer Units (DTUs), leading to aggregated packets, (ii) perform the required actions on the data, and (iii) execute the final UL operation, forwarding data to the right target device through the proper interface. To this end, it should be noted that, from an operational point of view, the MIG creates a new thread each time a message is notified via the proper UL MQTT topics. In the proposed implementation, the data are temporarily stored inside the RPi4's RAM, thus limiting the processing time and increasing the overall performance.

For the sake of completeness, it should be highlighted that each software entity managing its own communication interface also encapsulates (with a proper parsing technique) the retrieved data into DTUs, to make them suitable for constrained protocols. More in detail, IP and MAC addresses are properly "compressed" (to minimize occupation): an IP address is encoded as a single integer and a MAC address is translated into its corresponding HEX value. This approach is especially useful to minimize the dimension of LoRaWAN messages' payloads. Finally, depending on the routing identifier, the resulting DTU is sent to the proper communication interface. An illustrative example of routing rules[2] defined internally in the proposed MIG is shown in Table I.

[2]As a future research direction, additional strategies for injecting routing rules from external entities (e.g., from fog/edge/cloud computing-like systems) could be derived. As mentioned before, only interactions with Wi-Fi and BLE nodes are considered.

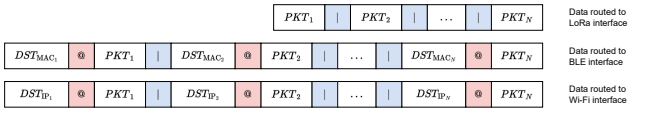



Fig. 4: Data packets format adopted in the proposed MIG to forward the processed information to end targets.

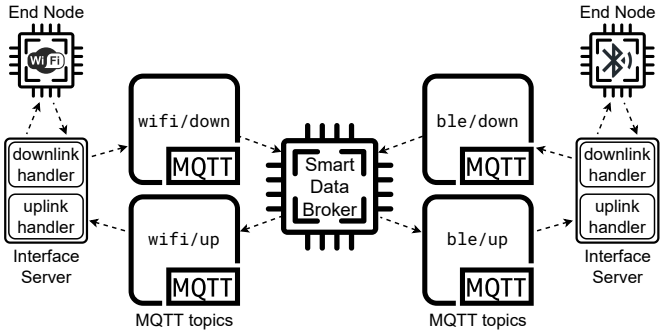



Fig. 5: Interactions among Wi-Fi/BLE end nodes and the proposed MIG.

The proposed DTU structure, shown in Fig. 4, allows processed data, to be transmitted by BLE and Wi-Fi communication interfaces, to be inserted in an output packet with a payload composed by $N$ aggregated payloads $\{PKT_i\}_{i=1}^{N}$ separated by the separator field "|," each one in turn composed by the identifier of the target (either IP or MAC address), separated from the payload by a separator field "@." At the opposite, in the case of data to be transmitted by the LoRaWAN interface, the resulting data packet will be a sequence of processed packets separated by the separator field "|." To this end, in order to further optimize the use of such constrained networks, intelligent (e.g., AI-based) data preprocessing mechanisms could be used, as will be discussed in Section VIII.

In order to better highlight the relation between *on-field* end nodes and the proposed IoT MIG, in Fig. 5 the data flows inside the proposed architecture are shown. The introduction of new communication interfaces would be possible owing to the internal modular architecture of the MIG. In fact, only the specific software routines needed for a new communication interface should be written, abiding by their own constraints and rules, while MQTT broker and SDB would remain unchanged. Moreover, the adoption of publish/subscribe-based routing policies allows the MIG to accept incoming traffic flows even from additional data sources and, possibly, with different input data packets' format—this is not considered in the current paper but represents an interesting research direction. We also remark that routing policies' management could be outsourced and (logically) concentrated on external controllers (e.g., in the cloud) to enhance the infrastructure management—this is out of the scope of this paper and is left as a future development.

Looking at the operational conditions, since the interaction among MQTT topics requires a non-negligible processing time, it can be assumed that the proposed MIG is suitable for non-real-time applications, where data are collected from

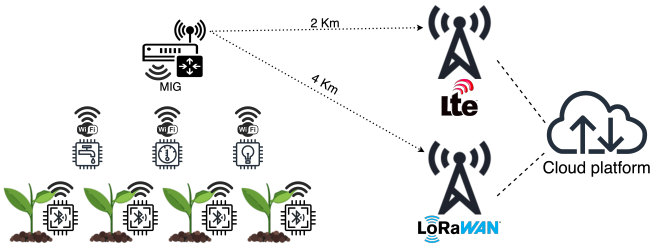



Fig. 6: Smart farming scenario with RPi-based MIG and several ESP32-based Wi-Fi and BLE sensing nodes.

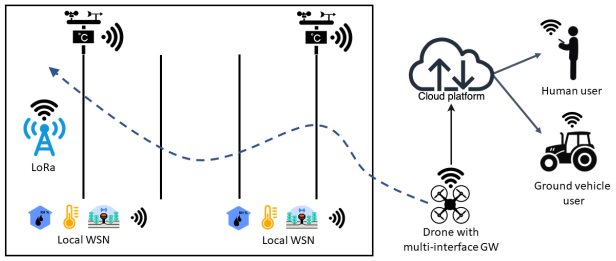



Fig. 7: UAV environment monitoring applications involving the proposed MIG.

different sources (e.g., for environmental monitoring, as well as in non-critical Industrial IoT, IIoT, domains) within a proper amount of time (e.g., at least 1 s). The system performance could be improved by reducing the internal processing time: this represents an interesting research direction.

As anticipated in Section I, the proposed MIG is applicable to several heterogeneous IoT scenarios where data may be collected from different data sources deployed in the environment of interest (e.g., environmental data sensing and remote monitoring), especially in those contexts in which communication protocols may be constrained and/or communication conditions may be challenging.

- A *first* representative application scenario is *smart farming*, in which several IoT sensing/actuating nodes (e.g., based on ESP32 System-on-Chip (SoC) [25] and equipped with Wi-Fi and BLE connectivity, as well as several hardware sensors, such as DHT11 [26]) with short-range communication capabilities are deployed over a large area far away from an Internet access point. The collected data need to reach high-layer processing entities (e.g., cloud platforms, as well as end users, such as farmers) interested on these data, following a *Farm-as-a-Service* (FaaS) approach [27]. Therefore, the use of a MIG, similar to the prototypical implementation proposed in Section III and based on a RPi 4 with additional LoRaWAN and cellular communication interfaces, will be essential to support information collection from the field and forwarding to the Internet [28], as shown in Fig. 6.
- A *second* scenario benefiting from the adoption of the proposed IoT-oriented MIG is *Unmanned Aerial Vehicle (UAV)-based remote monitoring*. As an example, in smart city a large number of short-range WSNs might be deployed to collect data of interest, possibly pre-processing them before forwarding them to high-layer consumers. Then, a UAV equipped with a Wi-Fi-, BLE- and LoRaWAN-enabled MIG can, *first*, gather data (either using short-range or long-range communication protocols) by flying over/near these WSNs and, *then*, forward the collected data to an Internet-connected node using a long-range or cellular communication protocol, as shown in Fig. 7.
- A *third* reference scenario, in the more general context of smart cities, involves Vehicle-to-Everything (V2X) communications [29]. In this case, similarly to scenarios where UAVs should interact with each other, on the ground (i.e., along roads and highways) there may be the need to collect data (e.g., for monitoring purposes) directly from the vehicle and the driver by relying on the MIG. This would involve (i) the interaction of the MIG with the vehicle's *On Board Unit* (OBU) for diagnostic information retrieval, through wireless or wired communication interfaces, as well as (ii) data collection at the MIG from additional equipment and devices installed inside the vehicle's cabin (e.g., driver's smartphone and wearables for monitoring his/her stress levels [30]), through wireless communication interfaces. After preliminary processing performed in the MIG inside the vehicle, the extracted information may be sent to the infrastructure (e.g., via LoRaWAN, provided that the obtained data can fit its payload constraints) as well as to vehicles in the neighborhood (e.g., via vehicular Wi-Fi or BLE, for warning alerting), possibly exploiting Advanced Driver Assistance Systems (ADAS) and targeting cooperative communications which urban and sub-urban mobility can benefit from.

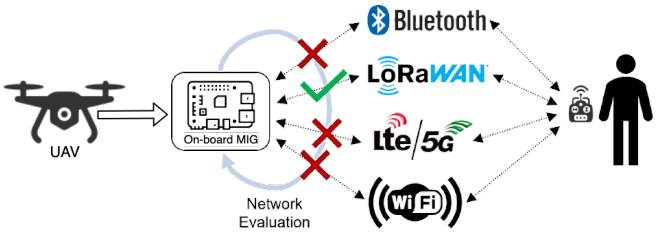



Fig. 8: MIG adopted as network backup solution for UAV application.

Finally, as a general remark, the proposed IoT MIG allows to exploit additional heterogeneous communication interfaces as backup communication interfaces in critical situations or when the communication interface under use fails [31]. This can be the case, as shown in Fig. 8, in contexts involving drones (e.g., for UAV-to-Ground communications and vice-versa), where multiple data streams could be exchanged between flying UAVs and one (or more) ground control centers, as well as among the UAV and its human pilot (to ensure the correct execution of the flight operations).

All the illustrative applications outlined above require the use of heterogeneous communication protocols and the MIG plays a key role in enabling interactions with each other.

## IV. Analytical Queuing Model

In order to investigate how the proposed MIG harnesses communication heterogeneity by properly handling traffic data, in this section we derive a novel Markov chain-based queuing model for the MIG. In Subsection IV-A, the MIG is modeled through an embedded Markov chain with states corresponding to the communication interfaces: the chain is in one state if the corresponding interface is transmitting or receiving. We assume that one interface at a time can be active.[3] The Markov chain transition matrix is associated with input and output flows across different interfaces. In Subsection IV-B, each communication interface's UL queue will be modeled as a G/G/1 queue.

We remark that our Markov chain-based model does not take into account the physical transmission channels associated with the communication interfaces equipping the MIG. In fact, our focus is on internal information flow management and we will assume that the wireless communication channels are error-free. An interesting extension encompasses the analysis of the impact of channel status on system performance (e.g., because of retransmissions). The proposed Markov chain model provides a simple, yet effective, way to predict the performance of the MIG, taking also into account possible limitations of a real system deployment (such as the internal processing time, due to the processing capabilities of the embedded SBC—a RPi, as indicated at the beginning of Section III).

### A. Embedded Markov Chain

The flows of the DTUs (mentioned in Section III) inside the MIG can be characterized through an embedded Markov chain with states corresponding to the MIG's communication interfaces. The transition probability associated with a link between two states depends, in general, on the flow from the input (DL) interface (initial state) to the output (UL) interface (final state). A high-level overview of the state transition diagram is shown in Fig. 9, where $\lambda^{(\mathrm{DL})}_{\mathrm{W}_{(\mathrm{in})}}$, $\lambda^{(\mathrm{DL})}_{\mathrm{C}_{(\mathrm{in})}}$ and $\lambda^{(\mathrm{DL})}_{\mathrm{B}_{(\mathrm{in})}}$ represent the input arrival rates (dimension: [DTU/s]) from the Wi-Fi, cellular and BLE interfaces, respectively, while $\lambda^{(\mathrm{UL})}_{\mathrm{W}_{(\mathrm{out})}}$, $\lambda^{(\mathrm{UL})}_{\mathrm{C}_{(\mathrm{out})}}$, $\lambda^{(\mathrm{UL})}_{\mathrm{L}_{(\mathrm{out})}}$ and $\lambda^{(\mathrm{UL})}_{\mathrm{B}_{(\mathrm{out})}}$ represent the departure rates from Wi-Fi, cellular (dimension: [DTU/s] for both), LoRaWAN and BLE (dimension: [pkt/s] for both) interfaces, respectively.[4] Moreover, $S_{\mathrm{W}}$, $S_{\mathrm{B}}$, $S_{\mathrm{C}}$ and $S_{\mathrm{L}}$ represent the service times (dimension: [s]) of the servers associated with the corresponding interfaces, respectively.

The transition matrix of the Markov chain shown in Fig. 9 can be expressed as follows:

$$\mathbb{P} = [P_{i,j}] = \begin{pmatrix} P_{\mathrm{W,W}} & P_{\mathrm{W,C}} & P_{\mathrm{W,B}} & P_{\mathrm{W,L}} \\ P_{\mathrm{C,W}} & P_{\mathrm{C,C}} & P_{\mathrm{C,B}} & P_{\mathrm{C,L}} \\ P_{\mathrm{B,W}} & P_{\mathrm{B,C}} & P_{\mathrm{B,B}} & P_{\mathrm{B,L}} \end{pmatrix} \quad (1)$$

[3]The extension to the case with multiple active interfaces (e.g., one receiving and one transmitting) is currently under investigation.

[4]We remark that no arrival flow is considered in the LoRaWAN state (i.e., $\lambda^{(\mathrm{DL})}_{\mathrm{L}_{(\mathrm{in})}} = 0$ pkt/s), as the LoRaWAN interface is assumed to support only UL communications (no Class C IoT node is considered). The fact that $\lambda^{(\mathrm{UL})}_{\mathrm{L}_{(\mathrm{out})}}$ and $\lambda^{(\mathrm{UL})}_{\mathrm{B}_{(\mathrm{out})}}$ are expressed in pkt/s, rather than DTU/s, will be clarified in the remainder of the section.

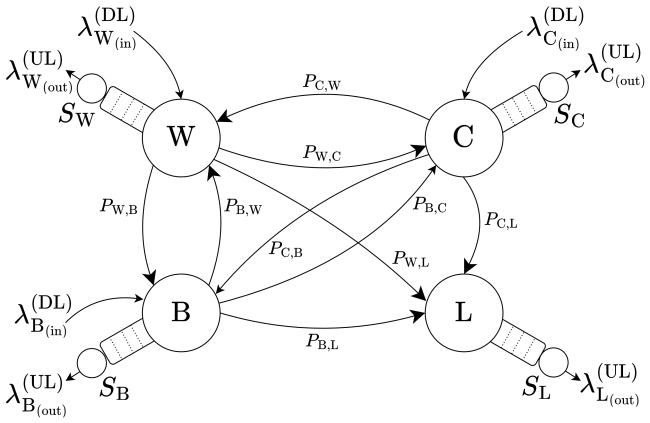



Fig. 9: State transition diagram of the proposed IoT MIG.

where $P_{i,j}$, $i \in \{\mathrm{W,C,B,L}\}$, $j \in \{\mathrm{W,B,C}\}$ represents the transition probability from state $i$ to state $j$ or, in other words, the probability that an information flow has to be transferred from the $i$-th interface (receiving interface) to the $j$-th interface (transmitting interface).

Given that in our Markov chain-based queuing model the transition probabilities depend on (i) the internal (inside the MIG) routing rules and (ii) the input arrival rates at the MIG's communication interfaces (DL flows), the corresponding arrival rates at the output queues at the communication interfaces (UL flows) can be expressed as follows:

$$\begin{aligned} \lambda^{(\mathrm{UL})}_{\mathrm{L}_{(\mathrm{in})}} &= \lambda^{(\mathrm{DL})}_{\mathrm{B}_{(\mathrm{in})}} \cdot P_{\mathrm{B,L}} + \lambda^{(\mathrm{DL})}_{\mathrm{C}_{(\mathrm{in})}} \cdot P_{\mathrm{C,L}} + \lambda^{(\mathrm{DL})}_{\mathrm{W}_{(\mathrm{in})}} \cdot P_{\mathrm{W,L}} \\ \lambda^{(\mathrm{UL})}_{\mathrm{B}_{(\mathrm{in})}} &= \lambda^{(\mathrm{DL})}_{\mathrm{C}_{(\mathrm{in})}} \cdot P_{\mathrm{C,B}} + \lambda^{(\mathrm{DL})}_{\mathrm{W}_{(\mathrm{in})}} \cdot P_{\mathrm{W,B}} \\ \lambda^{(\mathrm{UL})}_{\mathrm{C}_{(\mathrm{in})}} &= \lambda^{(\mathrm{DL})}_{\mathrm{B}_{(\mathrm{in})}} \cdot P_{\mathrm{B,C}} + \lambda^{(\mathrm{DL})}_{\mathrm{W}_{(\mathrm{in})}} \cdot P_{\mathrm{W,C}} \\ \lambda^{(\mathrm{UL})}_{\mathrm{W}_{(\mathrm{in})}} &= \lambda^{(\mathrm{DL})}_{\mathrm{C}_{(\mathrm{in})}} \cdot P_{\mathrm{C,W}} + \lambda^{(\mathrm{DL})}_{\mathrm{B}_{(\mathrm{in})}} \cdot P_{\mathrm{B,W}} \,. \end{aligned} \quad (2)$$

We further remark that, while an output queue is considered at each MIG interface (for transmissions out of the MIG, i.e., UL transmissions), no queues are associated with the links between pairs of states of the state diagram (i.e., between pairs of MIG interfaces), since: (i) packets received from end nodes are immediately processed (no need for input queues at the communication interfaces); and (ii) internal transitions are managed at software level and the corresponding latencies can thus be neglected in the Markov chain-based model (the internal latency is negligible). Therefore, we focus on the output queues associated with the MIG interfaces.

We finally assume that the internal routing between the different MIG interfaces considers direct information flows from one interface to another interface (e.g., an information flow entering from the BLE interface is forwarded to the LoRaWAN interface). This *1-in-to-1-out* assumption on internal routing is meaningful (and non-restrictive) for the following reasons: (i) it reflects a realistic behavior of the MIG for IoT applications, as discussed in Section III; (ii) it keeps the internal Markov chain-based model tractable. The extension to the combination of multiple input information flows (e.g., BLE and Wi-Fi) into a single output information flow (e.g., LoRaWAN) is an

interesting research extension. Taking into account Eq. (2), the *1-in-to-1-out* assumption can be formalized with the following constraints:

$$\begin{aligned} &P_{\mathrm{B,L}}, P_{\mathrm{C,L}}, P_{\mathrm{W,L}}, P_{\mathrm{C,B}}, P_{\mathrm{W,B}}, \\ &\quad P_{\mathrm{B,C}}, P_{\mathrm{W,C}}, P_{\mathrm{C,W}}, P_{\mathrm{B,W}} \in \{0,1\} \\ &\begin{cases} P_{\mathrm{B,L}} + P_{\mathrm{C,L}} + P_{\mathrm{W,L}} = 1 \\ P_{\mathrm{C,B}} + P_{\mathrm{W,B}} = 1 \\ P_{\mathrm{B,C}} + P_{\mathrm{W,C}} = 1 \\ P_{\mathrm{C,W}} + P_{\mathrm{B,W}} = 1\,. \end{cases} \end{aligned} \tag{3}$$

For example, assuming that an information flow from the BLE interface has to be forwarded to the LoRaWAN interface, in Eq. (2) one should set $P_{\mathrm{B,L}} = 1$, $P_{\mathrm{C,L}} = 0$, and $P_{\mathrm{W,L}} = 0$, obtaining:

$$\begin{aligned} \lambda^{(\mathrm{UL})}_{\mathrm{L}_{(\mathrm{in})}} &= \lambda^{(\mathrm{DL})}_{\mathrm{B}_{(\mathrm{in})}} \cdot P_{\mathrm{B,L}} + \lambda^{(\mathrm{DL})}_{\mathrm{C}_{(\mathrm{in})}} \cdot P_{\mathrm{C,L}} + \lambda^{(\mathrm{DL})}_{\mathrm{W}_{(\mathrm{in})}} \cdot P_{\mathrm{W,L}} \\ &= \lambda^{(\mathrm{DL})}_{\mathrm{B}_{(\mathrm{in})}}\,. \end{aligned} \tag{4}$$

Finally, we remark that, even if the proposed MIG's analytical model does not take into account network congestion (especially at transport layer), the extension of the analytical model to encompass the presence of congestion represents an interesting research direction. For example, a preliminary strategy may require to dynamically update the transition probabilities, with a direct impact on the routing rules associated with each communication interface, taking into account the load at the interfaces' servers. As an example, if the LoRaWAN interface server is operating at a value closer to its stability threshold (i.e., $\rho_L \longmapsto 1^-$), the associated LoRaWAN *in-flow* transition probabilities (namely: $P_{\mathrm{B,L}}$, $P_{\mathrm{C,L}}$, $P_{\mathrm{W,L}}$) can be reduced, while the *in-flow* transition probabilities of another interface (e.g., Wi-Fi, namely: $P_{\mathrm{B,W}}$ and $P_{\mathrm{C,W}}$) can be increased, thus lowering $\lambda^{(\mathrm{UL})}_{\mathrm{L}_{(\mathrm{in})}}$ to reduce the load at the LoRaWAN server and, therefore, congestion. Another possible approach to handle network congestion could be based on removing the *1-in-to-1-out* assumption in Eq. (3): part of the information flow entering into a congested interface may be deflected to another interface available to reach the final intended destination. In general, there may be other approaches to handle congestion: however, this analysis goes beyond the scope of the current work and will be the subject of our future research.

### B. G/G/1 Queues for Uplink Interfaces

In the proposed Markov chain model, we assume that each interface UL queue (outgoing traffic) is associated with a G/G/1 queue. This is an analytical queuing model of the MQTT-based system described, from an architectural point of view, in Section III. The G/G/1 queuing model has been chosen due to the generality of the distributions associated with arrival processes and service times. In fact, in the proposed MIG, for each communication interface: the inter-arrival time of DTUs has a general distribution with known parameters; and the service time has a distribution which depends on parameters related to the size of the packet being processed.

In order to accurately model the behavior of the G/G/1 queue at each UL interface, two remarks are expedient: (i) Wi-Fi and cellular G/G/1 queues transmit each DTU without performing any batch operation (on groups of DTUs), whereas (ii) BLE and LoRaWAN G/G/1 queues perform DTUs batching to optimize the throughput. In other words, in BLE and LoRaWAN cases the output packet size is maximized by concatenating together, in a single payload, as many DTUs as allowed by the standards, taking into account the operational settings.[5]

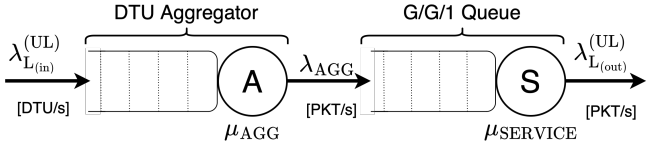



Fig. 10: G/G/1 LoRaWAN queue as a concatenation of DTU Aggregator (modeled as a G/D/1 queue) and Packet Transmitter (modeled as a G/G/1 queue).

*1) LoRaWAN G/G/1 Queue:* LoRaWAN is the most constrained communication interface in the MIG. Its corresponding G/G/1 queue, shown in Fig. 10, can be decomposed into the concatenation of two sub-queues: (a) a DTU Aggregator, receiving DTUs and batching them together in order to create a single LoRaWAN packet, and (b) the LoRaWAN's Packet Transmitter, in charge of processing the packets and transmitting them. In the following, we characterize these two sub-queues.

*a) DTU Aggregator:* The DTU Aggregator can be modeled as a G/D/1 queue, where $\lambda^{(\mathrm{UL})}_{\mathrm{L}_{(\mathrm{in})}}$ is the input arrival rate (dimension: [DTU/s]) of the single DTUs and $\overline{T}_{\mathrm{AGG}}$ is the average service time (dimension: [s]) needed to aggregate a packet.[6] To this end, the G/D/1 queue model has been chosen, as the DTU arrival distribution can, in principle, be general, while the service time is deterministic, as it depends (uniquely) on the number of DTUs flowing into the DTU Aggregator. The behavior of the DTU Aggregator guarantees a trade-off between LoRaWAN packet length maximization—and, consequently, LoRaWAN throughput maximization—and minimization of the waiting time inside the buffer of the DTU Aggregator. In particular, a maximum waiting time, defined as $t_{\max}$ (dimension: [s]), is introduced: after $t_{\max}$, even if the number of DTUs in the buffer is smaller than the maximum (denoted as $n$) allowed in a single LoRaWAN packet payload, the DTUs are aggregated and, then, sent to the Packet Transmitter. As a consequence, this approach allows DTUs inside the DTU Aggregator to incur a limited waiting time, as a trade-off between aggregated packets with small payloads (low throughput and short waiting time) and with large payloads (high throughput and long waiting time).

On the basis of the above assumptions, multiple DTUs will be aggregated together, up to a maximum of $n$ DTUs, *if and only if* the inter-arrival times between consecutive DTUs is shorter than $t_{\max}$. Otherwise, the "incomplete" packet will be

[5]As an example, in the case of LoRaWAN, the average number of DTUs in a single packet depends on the specific Spreading Factor (SF) chosen for the UL transmission [32].

[6]The communication interface subscript used in Subsection IV-A is eliminated for notational simplicity.

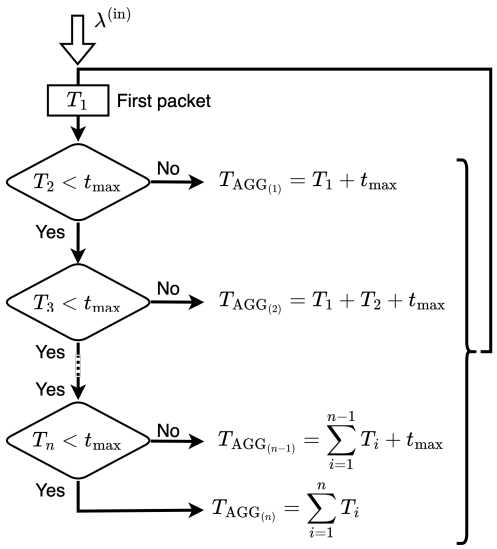



Fig. 11: Flow diagram of the DTU Aggregator module.

sent to the next sub-queue in Fig. 10 (i.e., the Packet Transmitter) *as-is*. For the sake of clarity, the flow diagram detailing the behavior of the DTU Aggregator is shown in Fig. 11 and the meanings of the indicated parameters are the following: $\lambda^{(\text{UL})}_{\text{L}_{(\text{in})}}$ represents the average arrival rate (dimension: [DTU/s]) of the DTUs and, since the DTU Aggregator's model is based on a G/D/1 queue, the arrival rate $\lambda^{(\text{UL})}_{\text{L}_{(\text{in})}}$ can be derived according to Eq. (4) (i.e., based on the *1-in-to-1-out* information flow assumption). Moreover, it is possible to express the arrival rate as a function of the average inter-arrival time as follows:

$$\lambda^{(\text{UL})}_{\text{L}_{(\text{in})}} = \lambda^{(\text{DL})}_{\text{B}_{(\text{in})}} = \frac{1}{\overline{T}} \tag{5}$$

where $\overline{T}$ is the average inter-arrival time (dimension: [s]) between consecutive DTUs and depends on the distribution of the DTU arrival process—this will be further discussed in Section VI.

Owing to the previous discussion, considering (i) the average inter-arrival time $\overline{T}$ between DTUs, (ii) the average DTU size $\overline{L}_{\text{DTU}}$, (iii) the threshold value of the waiting time $t_{\max}$, and (iv) the DTU Aggregator flow diagram shown in Fig. 11, it is possible to evaluate the average LoRaWAN packet aggregation time and the average arrival rate $\lambda_{\text{AGG}}$ (dimension: [pkt/s]) at the input of the LoRaWAN Packet Transmitter. By using the total probability theorem one can write:

$$\mathbb{E}[T_{\text{AGG}}] = \sum_{i=1}^{n} \mathbb{E}[T_{\text{AGG}}|\mathcal{A}_i] \cdot P(\mathcal{A}_i) \tag{6}$$

where:

$$\begin{aligned} \mathcal{A}_1 &\triangleq \left\{T_2 > t_{\max}\right\} \\ \mathcal{A}_i &\triangleq \left\{T_2 < t_{\max}, \ldots, T_i < t_{\max}, T_{i+1} > t_{\max}\right\}, \\ &\quad i = 2, \ldots, n-1 \\ \mathcal{A}_n &\triangleq \left\{T_2 < t_{\max}, \ldots, T_{n-1} < t_{\max}, T_n < t_{\max}\right\} \end{aligned} \tag{7}$$

and

$$\mathbb{E}[T_{\text{AGG}}|\mathcal{A}_i] = \mathbb{E}[T_{\text{AGG}_{(i)}}] \tag{8}$$

where:

$$T_{\text{AGG}_{(i)}} \triangleq \begin{cases} \sum_{j=1}^{i} T_j + t_{\max} & 1 \le i < n \\ \sum_{j=1}^{n} T_j & i = n. \end{cases} \tag{9}$$

Since $\{T_i\}$ are independent and identically distributed, defining $P_{\max} \triangleq P\{T_i > t_{\max}\}$ one can write:

$$P(\mathcal{A}_i) = \begin{cases} (1-P_{\max})^{i-1} \cdot P_{\max} & i = 1, \ldots, n-1 \\ (1-P_{\max})^{n-1} & i = n. \end{cases} \tag{10}$$

From Eq. (6), one obtains:

$$\begin{aligned} \overline{T}_{\text{AGG}} = &\sum_{i=1}^{n-1} \left( \sum_{j=1}^{i} \overline{T}_j + t_{\max} \right) \cdot (1-P_{\max})^{i-1} \cdot P_{\max} \\ &+ \left( \sum_{i=1}^{n} \overline{T}_i \right) \cdot (1-P_{\max})^{n-1}. \end{aligned} \tag{11}$$

Finally, observing that $\overline{T}_i = \overline{T}$, $\forall i \in \{1, \ldots, n\}$, it follows:

$$\begin{aligned} \overline{T}_{\text{AGG}} = &\sum_{i=1}^{n-1} \Big( i \cdot \overline{T} + t_{\max} \Big) \cdot (1-P_{\max})^{i-1} \cdot P_{\max} \\ &+ n \cdot \overline{T} \cdot (1-P_{\max})^{n-1}. \end{aligned} \tag{12}$$

Similarly, one can obtain the average quadratic value of $T_{\text{AGG}}$ as follows:

$$\begin{aligned} \mathbb{E}[T^2_{\text{AGG}}] &= \sum_{i=1}^{n} \mathbb{E}\Big[T^2_{\text{AGG}}|\mathcal{A}_i\Big] \cdot P(\mathcal{A}_i) \\ &= \sum_{i=1}^{n-1} \mathbb{E}\Big[ \big( \sum_{j=1}^{i} T_i + t_{\max} \big)^2 \Big] \\ &\quad \cdot (1-P_{\max})^{i-1} \cdot P_{\max} \\ &\quad + \mathbb{E}\Big[ \big( \sum_{j=1}^{n} T_j \big)^2 \Big] \cdot (1-P_{\max})^{n-1}. \end{aligned} \tag{13}$$

*b) Packet Transmitter:* In order to evaluate the waiting time in the buffer of the LoRaWAN Packet Transmitter, modeled as a G/G/1 queue, one can rely on Kingman's formula [33]:

$$\overline{W}_{\text{q}} = \left( \frac{\rho}{1-\rho} \right) \left( \frac{C_a^2 + C_s^2}{2} \right) \overline{S}_{\text{L}} \tag{14}$$

where: $\rho \triangleq \lambda_{\text{AGG}} \cdot \overline{S}_{\text{L}}$ (adimensional); $\overline{S}_{\text{L}}$ is the average service time of the LoRaWAN server (dimension: [s]); and $C_a$ and $C_s$ are the coefficients of variation of arrival and service times (adimensional), respectively.

The coefficient of variation of arrival times $C_a$ can be expressed as:

$$C_a = \frac{\sigma_{T_{\text{AGG}}}}{\frac{1}{\lambda_{\text{AGG}}}} = \sigma_{T_{\text{AGG}}} \cdot \lambda_{\text{AGG}} \tag{15}$$

where

$$\lambda_{\text{AGG}} = \frac{1}{\overline{T}_{\text{AGG}}} \tag{16}$$

and

$$\sigma^2_{T_{\text{AGG}}} = \mathbb{E}[T^2_{\text{AGG}}] - \mathbb{E}[T_{\text{AGG}}]^2. \tag{17}$$

In order to evaluate the coefficient of variation of the service time, denoted as $C_s$, further analysis on the LoRaWAN protocol behavior and policy rules is required. *First*, the service times' distribution should be related to the LoRaWAN packet size, thus associating each packet configuration (with a specific number of aggregated DTUs) with a proper probability, which depends on the parameters of the DTUs' source generating distribution. As an example, assuming an average DTU size $\overline{L}_{\mathrm{DTU}} = 20$ bytes (as will be detailed in Section VII) and considering a maximum LoRaWAN useful achievable payload size equal to 222 bytes,[7] the maximum number of DTUs possibly inserted into in a single LoRaWAN packet would be equal to 11, which would then correspond to the specific value of the parameter $n$ in the previous derivation (e.g., in Fig. 11).

*Then*, taking into account the DTU size and the LoRaWAN constraints, it is possible to evaluate the service time of the LoRaWAN G/G/1 server. The service time of a packet containing $i \in \{1, \ldots, n\}$ aggregated DTUs, denoted as $S_{\mathrm{L}_i}$, can be expressed as follows:

$$S_{\mathrm{L}_i} = T_{\mathrm{P}_{\mathrm{PROC}-i}} + T_{\mathrm{P}_{\mathrm{AIR}}} + T_{\mathrm{P}_{\mathrm{DUTYCYCLE}}} \tag{18}$$

where: $T_{\mathrm{P}_{\mathrm{PROC}-i}}$ is the internal processing time (dimension: [s]) associated with $i$ DTUs and can be expressed as

$$T_{\mathrm{P}_{\mathrm{PROC}-i}} = i\,\overline{T}_{\mathrm{DTU}_{\mathrm{PROC}}}\,; \tag{19}$$

$T_{\mathrm{P}_{\mathrm{AIR}}}$ is the packet airtime (dimension: [s]) and can be expressed as [34]

$$\begin{aligned} T_{\mathrm{P}_{\mathrm{AIR}}} = {} & (n_{\mathrm{PREAMBLE}} + 4.25)\,\frac{2^{SF}}{BW} \\ & + \left[8 + \max\left(\left\lceil \frac{8PL - 4SF + 28 + 16 - 20H}{4\,(SF - 2DE)} \right\rceil (CR + 4)\,, 0\right)\right] \frac{2^{SF}}{BW}\,; \end{aligned} \tag{20}$$

and $T_{\mathrm{P}_{\mathrm{DUTYCYCLE}}}$ is the LoRaWAN duty cycle time (dimension: [s]) and can be expressed, according to the regional parameters [32], as

$$T_{\mathrm{P}_{\mathrm{DUTYCYCLE}}} = 0.99 \cdot T_{\mathrm{P}_{\mathrm{AIR}}}\,. \tag{21}$$

In the formulas above: $\overline{T}_{\mathrm{DTU}_{\mathrm{PROC}}}$ depends on the specific HW platform which the MIG builds upon; the LoRaWAN preamble size, denoted as $n_{\mathrm{PREAMBLE}}$, is set to 8 byte; $SF = 7$, $BW = 125$, $DE = 0$ (low data rate optimization), $CR = 4$ (coding rate); and $PL$ (LoRa packet payload), which includes a 13 byte LoRaWAN packet header and the aggregated DTUs, can then be expressed (in our model) as $PL = 13 + n \cdot \overline{L}_{\mathrm{DTU}}$. These LoRaWAN-related parameters have been set according to the LoRaWAN regional parameters [32].

According to Eq. (18), the LoRaWAN packet service times $\{S_{\mathrm{L}_i}\}_{i=1}^{n}$, with $n = 11$, depend on the packet dimension. In particular, the service time ranges from $S_{\mathrm{L}_1} \cong 7.2$ s (packet with 1 DTU) up to $S_{\mathrm{L}_n} \cong 37$ s (packet with $n = 11$ DTUs).

The average service time, denoted as $\overline{S}_{\mathrm{L}}$, can be calculated by applying the total probability theorem on the partition $\{\mathcal{A}_i\}_{i=1}^{n}$ in Eq. (7) and the service time defined by Eq. (18), thus obtaining:

$$\begin{aligned} \overline{S}_{\mathrm{L}} &= \sum_{i=1}^{n} \mathbb{E}[S_{\mathrm{L}} | \mathcal{A}_i] \cdot P(\mathcal{A}_i) \\ &\cong \sum_{i=1}^{n} S_{\mathrm{L}_i} \cdot P(\mathcal{A}_i) \\ &= \sum_{i=1}^{n-1} S_{\mathrm{L}_i} (1 - P_{\max})^{i-1} \cdot P_{\max} + S_{\mathrm{L}_n} (1 - P_{\max})^{n-1}\,. \end{aligned} \tag{22}$$

Similarly, one can write

$$\begin{aligned} \mathbb{E}[S_{\mathrm{L}}^2] &= \sum_{i=1}^{n} \mathbb{E}[S_{\mathrm{L}}^2 | \mathcal{A}_i] \cdot P(\mathcal{A}_i) \\ &= \sum_{i=1}^{n} \mathbb{E}\left[\left(\sum_{i=1}^{n-1} S_{\mathrm{L}_i}\right)^2\right] \\ &\quad \cdot (1 - P_{\max})^{i-1} \cdot P_{\max} \\ &\quad + \mathbb{E}\left[S_{\mathrm{L}_n}^2\right] \cdot (1 - P_{\max})^{n-1}\,. \end{aligned} \tag{23}$$

At this point, the variance of the service time, denoted as $\sigma^2_{S_{\mathrm{L}}}$, can be calculated as follows:

$$\sigma^2_{S_{\mathrm{L}}} = \mathbb{E}[S_{\mathrm{L}}^2] - \mathbb{E}[S_{\mathrm{L}}]^2\,. \tag{24}$$

The coefficient of variation of the service time $C_s$ can thus be expressed as

$$C_s = \frac{\sigma_{S_{\mathrm{L}}}}{\overline{S}_{\mathrm{L}}}\,. \tag{25}$$

At this point, one can evaluate the average waiting time in the buffer, denoted as $\overline{W}_{\mathrm{q}}^{(\mathrm{L})}$, according to Eq. (14).

Finally, knowing $\overline{W}_{\mathrm{q}}^{(\mathrm{L})}$ (Eq. (14)), $\overline{S}_{\mathrm{L}}$ (Eq. (22)), and $\overline{T}_{\mathrm{AGG}}$ (Eq. (12)), it is possible to calculate the overall average time (denoted as $\overline{T}_{\mathrm{PKT-SOJ_L}}$) that each DTU is expected to spend at the LoRaWAN communication interface (namely, DTU Aggregator and Packet Transmitter), from the time instant of DTU arrival to the time instant of packet (in which the DTU has been aggregated) departure, as follows:

$$\overline{T}_{\mathrm{PKT-SOJ_L}} = \overline{W}_{\mathrm{q}}^{(\mathrm{L})} + \overline{S}_{\mathrm{L}} + \overline{T}_{\mathrm{AGG}}^{(\mathrm{L})}\,. \tag{26}$$

*2) BLE G/G/1 Queue:* Focusing on the BLE interface, its G/G/1 queue model is similar to the one detailed in Subsection IV-B1a. More specifically, the BLE DTU Aggregator has the same behavior of the LoRaWAN DTU Aggregator, while a proper G/G/1 queue, associated with the BLE Packet Transmitter, must be defined according to the BLE protocol rules.

BLE packets can have a larger dimension (with maximum corresponding to 512 bytes) than LoRaWAN ones. Therefore, the BLE DTU Aggregator is required to aggregate up to $n = 25$ DTUs. Moreover, similarly to the LoRaWAN DTU Aggregator, even for the BLE DTU Aggregator one can introduce the parameter $t_{\max}$, which takes the same value as the one considered in Subsection IV-B1a. The same holds for the other parameters (e.g., $\lambda_{\mathrm{B}_{(\mathrm{in})}}^{(\mathrm{UL})}$), in order to fairly compare the performance of all communication interfaces. Hence, $\lambda_{\mathrm{B}_{(\mathrm{in})}}^{(\mathrm{UL})}$

[7]This value depends on the operating region and the SF chosen for the transmission. In particular, the payload size equal to 222 bytes has been obtained with SF7 and a BandWidth (BW) equal to 125 kHz [24].

can be calculated as in Eq. (5), while $\overline{T}_{\mathrm{AGG}}$ and $\sigma^2_{T_{\mathrm{AGG}}}$ can be evaluated as in Eq. (12) and Eq. (17) (relying on the state diagram in Fig. 11).

The main difference between BLE and LoRaWAN models is related to the service time of the G/G/1 queue modeling the Packet Transmitter. In fact, with the BLE protocol no duty cycle is used, thus resulting in a significantly shorter average service time. However, the BLE protocol requires the MIG (active as *master*) to connect to a BLE *slave* device before being able to communicate with it. Therefore, the BLE model has to take into account this connection time, denoted as $T_{\mathrm{CONN_B}}$ (dimension: [s]). On the basis of our experimental investigation, $\overline{T}_{\mathrm{CONN_B}} \cong 7$ s. The BLE packet service time can thus be calculated as

$$S_{\mathrm{B}_i} = \overline{T}_{\mathrm{P_{PROC-}}i} + \overline{T}_{\mathrm{CONN_B}} \tag{27}$$

where $\overline{T}_{\mathrm{P_{PROC-}}i}$, defined by Eq. (19), is the processing time required to create a packet which aggregates $i \in \{1, \ldots, n\}$ DTUs.

Hence, once all BLE packet service times $\{S_{\mathrm{B}_i}\}_{i=1}^{n}$ are calculated (similarly to the service times $\{S_{\mathrm{L}_i}\}_{i=1}^{n}$ detailed in Subsection IV-B1a for the G/G/1 queue of the LoRaWAN Packet Transmitter), it is possible to evaluate the average waiting time in the buffer of the G/G/1 LoRaWAN queue according to Eq. (14), which still holds for the BLE protocol. Finally, the overall average time spent by the aggregated packet in the BLE interface, denoted as $\overline{T}_{\mathrm{PKT-SOJ_B}}$, can be calculated as follows:

$$\overline{T}_{\mathrm{PKT-SOJ_B}} = \overline{W}_{\mathrm{q}}^{(\mathrm{B})} + \overline{S}_{\mathrm{B}} + \overline{T}_{\mathrm{AGG}}^{(\mathrm{B})}\,. \tag{28}$$

*3) Wi-Fi and Cellular G/G/1 Queues:* Given their similarity, in terms of performance and behavior, these two interfaces can be modeled in the same way. Since Wi-Fi and cellular interfaces have a significantly higher throughput than BLE and LoRaWAN interfaces, a batching operation on the incoming DTUs is not required and, then, the DTU Aggregator is no longer present in their corresponding models. Therefore, both Wi-Fi and cellular interfaces (UL) queues can be modeled as single G/G/1 queues and, furthermore, simplified to G/D/1 queues. In fact, the service time for each DTU can be seen as the sum of a fixed processing time (denoted as $t_{\mathrm{DTU_{PROC}}}$ and equal for both interfaces) and a fixed latency (denoted as $t_{\mathrm{LATENCY_W}}$ and $t_{\mathrm{LATENCY_C}}$ for Wi-Fi and cellular interfaces, respectively): they can thus be modeled as deterministic. In other words, one can write:

$$S_{\mathrm{W}} = t_{\mathrm{DTU_{PROC}}} + t_{\mathrm{LATENCY_W}} \tag{29}$$

$$S_{\mathrm{C}} = t_{\mathrm{DTU_{PROC}}} + t_{\mathrm{LATENCY_C}}\,. \tag{30}$$

Since the service time is deterministic, the coefficient of variation of the service time $C_s$ becomes equal to 0. Therefore, the waiting time in the buffer, given by Eq. (14), reduces, in the Wi-Fi and cellular cases, to

$$\overline{W}_{\mathrm{q}}^{(\mathrm{W})} = \frac{\rho_{\mathrm{W}}}{1-\rho_{\mathrm{W}}} \frac{\left(C_a^{(\mathrm{W})}\right)^2}{2} S_{\mathrm{W}} \tag{31}$$

$$\overline{W}_{\mathrm{q}}^{(\mathrm{C})} = \frac{\rho_{\mathrm{C}}}{1-\rho_{\mathrm{C}}} \frac{\left(C_a^{(\mathrm{C})}\right)^2}{2} S_{\mathrm{C}} \tag{32}$$

where: $S_{\mathrm{W}}$ and $S_{\mathrm{C}}$ are the Wi-Fi and cellular service times (Eq. (29) and Eq. (30), respectively); $\rho_{\mathrm{W}} \triangleq \lambda_{\mathrm{W_{(in)}}}^{(\mathrm{UL})} \cdot \overline{S}_{\mathrm{W}}$ and $\rho_{\mathrm{C}} \triangleq \lambda_{\mathrm{C_{(in)}}}^{(\mathrm{UL})} \cdot \overline{S}_{\mathrm{C}}$ (depending on the interface); $C_a^{(\mathrm{W})} = \sigma_{\mathrm{ARR}}^{(\mathrm{W})} \cdot \lambda_{\mathrm{W_{(in)}}}^{(\mathrm{UL})}$ and $C_a^{(\mathrm{C})} = \sigma_{\mathrm{ARR}}^{(\mathrm{C})} \cdot \lambda_{\mathrm{C_{(in)}}}^{(\mathrm{UL})}$.

Finally, it is possible to obtain the average waiting time $\overline{W}_{\mathrm{q}}$. The overall times spent by a DTU at the Wi-Fi or cellular interfaces can then be expressed as

$$\overline{T}_{\mathrm{DTU-SOJ_W}} = \overline{W}_{\mathrm{q}}^{(\mathrm{W})} + \overline{S}_{\mathrm{W}} \tag{33}$$

$$\overline{T}_{\mathrm{DTU-SOJ_C}} = \overline{W}_{\mathrm{q}}^{(\mathrm{C})} + \overline{S}_{\mathrm{C}} \tag{34}$$

where $\overline{W}_{\mathrm{q}}^{(\mathrm{W})}$ and $\overline{W}_{\mathrm{q}}^{(\mathrm{C})}$ can be computed as in Eq. (31) and Eq. (32), respectively.

As final remark, the main difference between Wi-Fi and cellular Packet Transmitter queues is that, according to experimental measurements, $t_{\mathrm{LATENCY_W}} \gg t_{\mathrm{LATENCY_C}}$. In other words, the cellular interface has a significantly longer sojourn time (due to technological reasons). This aspect, further depending on the specific cellular protocol version (e.g., 4/5G), may introduce a relevant latency in some applications.

## V. Simulation-based Performance Evaluation

In order to validate the analytical queuing model detailed in Subsection IV-B, a Python-based simulator has been developed taking into account all the blocks considered in the analytical model. More in detail, the simulator includes a DTU Generator with DTU inter-arrival time having a uniform distribution $\mathcal{U}[t_{\mathrm{a}}, t_{\mathrm{b}}]$ [35]. The DTU Generator generated the DTUs to be processed by the interface queues. In particular, the following reference values are initially considered: for all interfaces, $t_{\mathrm{a}} = 0$ s, $t_{\max} = 5$ s, $\overline{L}_{\mathrm{DTU}} = 20$ bytes, $\overline{T}_{\mathrm{DTU_{PROC}}} = 20$ ms; for the BLE interface, $\overline{T}_{\mathrm{CONN_B}} = 7$ s; for cellular and Wi-Fi interfaces, $t_{\mathrm{LATENCY_C}} = 50$ ms and $t_{\mathrm{LATENCY_W}} = 10$ ms, respectively.

By generating 1,000 DTUs,[8] we characterize each communication interface according to the following performance metrics: (i) server utilization ratio $\rho$; (ii) average service time $\overline{S}$; (iii) waiting time $\overline{W}_{\mathrm{q}}$ in the buffer of the G/G/1 queue and/or G/D/1 queue (depending on the presence or absence of the DTU Aggregator), and (iv) sojourn time $\overline{T}_{\mathrm{DTU-SOJ}}$ (which includes, in the LoRaWAN and BLE cases, the DTU aggregation time). The selected metrics are relevant for the following reasons.

- The average service time allows to estimate the processing time required by each protocol to serve packets.

[8]Our results show that this number of DTUs is statistically sufficient for performance evaluation, as will be seen by the confidence intervals of the simulation results presented in the following.

- The sojourn time is relevant to understand the overall time spent by the data in the system and, consequently, the latency introduced by the MIG in routing data between heterogeneous communication interfaces.
- The server utilization ratio is expedient to understand the load of the interface server, thus allowing to estimate if an information flow increment can be tolerated. Moreover, the server utilization ratio might be useful for energy consumption optimization purposes (e.g., to maximize battery energy savings with a battery-powered MIG).

As the analytical model in Section IV, the Python-based simulator as well does not take into account the physical characteristics of the wireless transmission channels associated with the communication interfaces of the MIG. In other words, we assume that the corresponding communication channels are ideal. Extending the simulator to take into account the channel status is an interesting research direction.

In order to estimate the accuracy of the simulated performance indicators, we evaluate the Confidence Interval (CI) [36] of each simulation point as follows:

$$\mathrm{CI}_{(95\%)} = z^* \cdot \frac{\sigma}{\sqrt{n_{\mathrm{sim}}}} \tag{35}$$

where: $z^*$ is the $z$ star parameter, set to 1.96 (as defined in [36]) to obtain a $95\%$ confidence interval; $\sigma^2$ is the variance of the analyzed indicator, obtained from the simulator's output; $n_{\mathrm{sim}}$ corresponds to the population number, equal to the number of DTUs processed by the simulator, i.e., $n_{\mathrm{sim}} = 1{,}000$.

### A. Evaluation of the Server Utilization Ratio

In order to analyze the stability conditions of the different communication protocols, we investigate the behavior of the server utilization ratio of each interface as a function of $t_{\mathrm{b}}$ (keeping all other parameters equal to the values set above), with DTUs' generation according to a uniform distribution $\mathcal{U}[0, t_{\mathrm{b}}]$ (i.e., we assume $t_{\mathrm{a}} = 0$). Therefore, the average inter-arrival time $\overline{T}$ can be calculated as follows:

$$\overline{T} = \frac{t_{\mathrm{b}}}{2}\,. \tag{36}$$

In Fig. 12, analytical ($\mathrm{an}$) and simulated ($\mathrm{sim}$) server utilization ratios for the following interfaces are shown: (a) LoRaWAN; (b) BLE; (c) Wi-Fi; and (d) cellular. This allows to directly compare (and validate) the performance predicted by the Markov chain-based analytical model with that predicted by the implemented Python simulator. From Fig. 12(a), it can be observed that $\rho_{\mathrm{L}}^{(\mathrm{sim})} = \rho_{\mathrm{L}}^{(\mathrm{an})} = 1$ for $t_{\mathrm{b}} \simeq 10$ s. Hence, it can be concluded that the LoRaWAN interface cannot support a DTU generation distribution $\mathcal{U}[0, t_{\mathrm{b}}]$ with $t_{\mathrm{b}} \leq t_{\mathrm{b}}^{(\mathrm{min-L})} \simeq 10$ s. In other words, at most one DTU every 10 s can be, on average, handled by the LoRaWAN server.

Regarding Fig. 12(c) and Fig. 12(d), related to Wi-Fi ($\rho_{\mathrm{W}}$) and cellular ($\rho_{\mathrm{C}}$) server utilization ratios, respectively, it can be noted that $\rho_{\mathrm{C}}^{(\mathrm{an})} = \rho_{\mathrm{C}}^{(\mathrm{sim})} = 1$ for $t_{\mathrm{b}} \simeq 0.124$ s, while $\rho_{\mathrm{W}}^{(\mathrm{an})} = \rho_{\mathrm{W}}^{(\mathrm{sim})} = 1$ for $t_{\mathrm{b}} \simeq 0.056$ s. A very good agreement between simulated and analytical performances can be observed.

Finally, looking at Fig. 12(b), it can be concluded that the BLE interface can properly handle incoming DTUs for $t_{\mathrm{b}} > t_{\mathrm{b}}^{(\mathrm{min-B})} \simeq 0.6$ s. Furthermore, from the analytical results one can notice that the BLE interface reaches a peak when $t_{\mathrm{b}} \simeq 5$ s. This corresponds to the value of the parameter $t_{\max}$, defined in Subsection IV-B1a, that maximizes the DTU aggregation process. This analytical result is confirmed also by simulation values and is further explained through an in-deep analysis of the behavior of the DTU Aggregator carried out in Subsection V-B.

### B. Impact of the DTU Aggregator on the Server Utilization Ratio

In order to better understand the impact of the DTU Aggregator on the server utilization ratio $\rho$, we investigate the BLE case, i.e., the behavior of $\rho$ shown in Fig. 12(b) and predicted by our analytical framework. Recall that the inter-arrival time between consecutive DTUs is $\overline{T} \sim \mathcal{U}[0, t_{\mathrm{b}}]$, with average value $\overline{T} = t_{\mathrm{b}}/2$. In Fig. 13, we highlight the analytical behavior of $\rho$, as a function of $t_{\mathrm{b}}$. It is possible to identify the following regions/values:

- $0 < t_{\mathrm{b}} < t_{\mathrm{b}}^{(\mathrm{min-B})} < t_{\max}$: instability region corresponding to $\rho > 1$ (interval (a) in Fig. 13);
- $t_{\mathrm{b}}^{(\mathrm{min-B})} < t_{\mathrm{b}} < t_{\max}$: first stability region, where $\rho$ decreases until reaching a local minimum (region (b) in Fig. 13);
- $t_{\mathrm{b}} \geq t_{\max}$: second stability region, where $\rho$ further decreases, from "sudden" peak around $t_{\mathrm{b}} \simeq t_{\max}$ (region (c) in Fig. 13).

As mentioned above, there is a sudden peak, in the analytical curve, around $t_{\mathrm{b}} = t_{\max}$—we remark that the simulation results, depicted in Fig. 12(b), show a "smoothed" version of this sharp analytical peak. In the following, we motivate this behavior.

*1) Instability Region:* For $0 < t_{\mathrm{b}} < t_{\mathrm{b}}^{(\mathrm{min-B})}$, with $t_{\mathrm{b}}^{(\mathrm{min-B})} \simeq 0.6$ s, it holds that $\rho_{\mathrm{B}}^{(\mathrm{an})} > 1$: the system is unstable, as the server cannot process all the incoming packets. In fact, the inter-arrival time $T$ has always a value smaller than $t_{\max}$, so that the DTU Aggregator always aggregates the largest possible number of DTUs (namely, $n = 25$ for the BLE interface) in each packet. In this region, the average packet generation time $\overline{T}_{\mathrm{AGG}}$ (which can be written as defined by Eq. (11)) is shorter than the average service time $\overline{S}_{\mathrm{B}}$ of the Packet Transmitter, defined according to Eq. (27). For the sake of clarity, this unstable behavior is illustrated in Fig. 14: the DTU Generator generates packets at a rate which cannot be sustained by the Packet Transmitter.

*2) First Stability Region:* This region is associated with values of $t_{\mathrm{b}}$ such that $t_{\mathrm{b}}^{(\mathrm{min-B})} < t_{\mathrm{b}} < t_{\max}$. In this region, each packet derives from the aggregation of the largest possible number (namely, $n$) of DTUs. Unlike the previous instability region, in this case the average packet generation time $\overline{T}_{\mathrm{AGG}}$ is longer than the average service time $\overline{S}_{\mathrm{B}}$ of the Packet Transmitter. This situation is pictured illustratively in Fig. 15: the Packet Transmitter manages to serve efficiently the received packets. The service utilization ratio decreases as

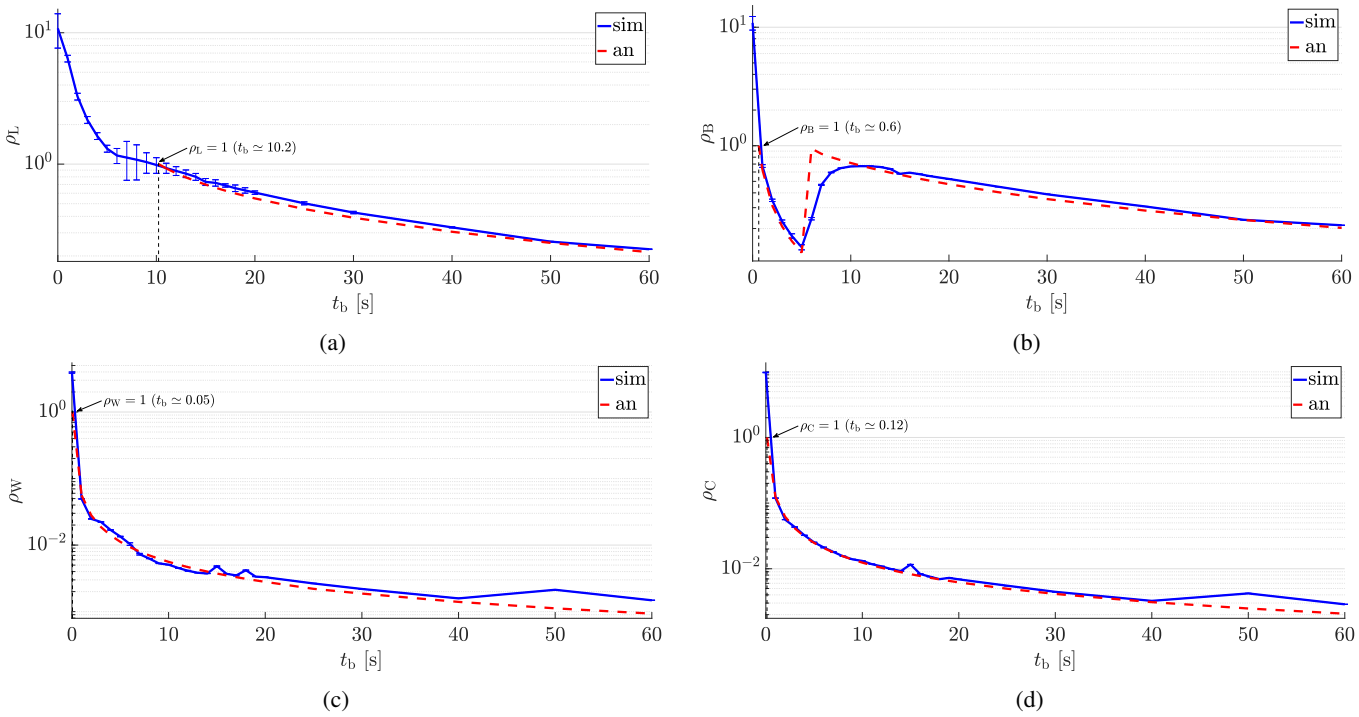



Fig. 12: Analytical (an) and simulated (sim) interface server utilization ratio as a function of $t_{\mathrm{b}}$ (with DTUs' generation distribution $\mathcal{U}[0, t_{\mathrm{b}}]$), for all interfaces: (a) LoRaWAN ($\rho_{\mathrm{L}}$); (b) BLE ($\rho_{\mathrm{B}}$); (c) Wi-Fi ($\rho_{\mathrm{W}}$); and (d) cellular ($\rho_{\mathrm{C}}$).

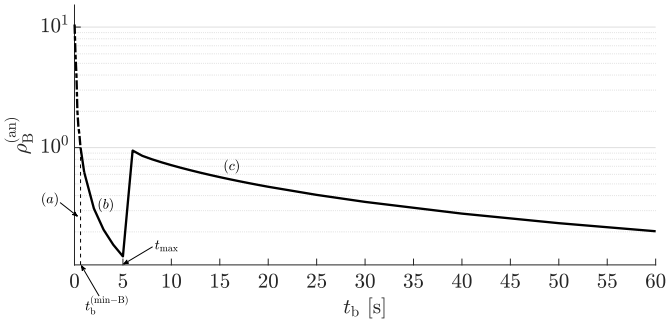



Fig. 13: Behavior of $\rho_{\mathrm{B}}^{(\mathrm{an})}$ in the assumption of fixed DTU inter-arrival time $T = \overline{T} = t_{\mathrm{b}}/2$.

the packet arrival rate reduces and the service time does not change.[9]

*3) Second Stability Region:* As can be observed in Fig. 13, in correspondence to $t_{\mathrm{b}} = t_{\max}$ there is a sharp increase of $\rho_{\mathrm{B}}^{(\mathrm{an})}$, which then decreases for increasing values of $t_{\mathrm{b}}$. This behaviour can be explained as follows. As soon as $t_{\mathrm{b}}$ overcomes $t_{\max}$, it then follows that each generated packet does not necessarily contains the maximum number $n$ of DTUs: for example, if the inter-arrival time between two consecutive DTUs is longer than $t_{\max}$, then a packet is generated. This means that the packet arrival rate at the Packet Transmitter increases. However, as shown in Section IV, the connection time $T_{\mathrm{CONN_B}}$ is the largest component of the service time of a BLE packet: this implies that even if the number of DTUs aggregated in a packet reduces, its service time reduces, proportionally, much less. This explains the sudden increase of the server utilization ratio for $t_{\mathrm{b}} \simeq t_{\max}$. For increasing values of $t_{\mathrm{b}}$, both the packet arrival rate and the number of aggregated DTUs per packet reduce, thus leading to a constant decrease of $\rho_{\mathrm{B}}^{(\mathrm{an})}$. In the limiting case with $t_{\mathrm{b}} \gg t_{\max}$, each packet contains only one DTU, as shown in Fig. 16.

We remark that the simulation-based results shown in Fig. 12(b) approximate the behaviour predicted by our analytical model. There is not the sharpest increase predicted by the analytical model as the number of DTUs considered for the simulation is finite (namely, 1,000). By increasing the number of DTUs, the simulation-based curve would approximate more and more the analytical one.

In order to further investigate the presence of the peak of the server utilization ratio for $t_{\mathrm{b}} \simeq t_{\max}$, in Fig. 17 we evaluate the analytical queuing server utilization ratio $\rho^{(\mathrm{an})}$, as a function of the parameter $t_{\mathrm{b}}$, associated with a uniform DTU generation distribution $\mathcal{U}[0, t_{\mathrm{b}}]$, for various values of $t_{\max}$ (in detail, 5 s, 10 s, and 15 s), considering both LoRaWAN (Fig. 17(a)) and BLE (Fig. 17(b)) interfaces. The obtained results confirm how the server utilization ratios is influenced by the parameter $t_{\max}$ of the DTU Aggregator. This parameter affects the system efficiency by reducing, for small values of $t_{\max}$, the "idle times" between aggregated packets sent to the interface server. For the sake of completeness, this behavior is further investigated in Section VI in the presence of other (non-uniform) input distributions. This will allow to further highlight how the peaks, for both LoRaWAN and BLE

[9]In Section IV, it has been shown that the service time of the BLE Packet Transmitter is associated with the average connection time $\overline{T}_{\mathrm{CONN_B}}$ and with the number of DTUs present in the aggregated packet. Since the number of DTUs in each packet is the largest possible (equal to $n$), then it follows that the service time of each packet is the same.

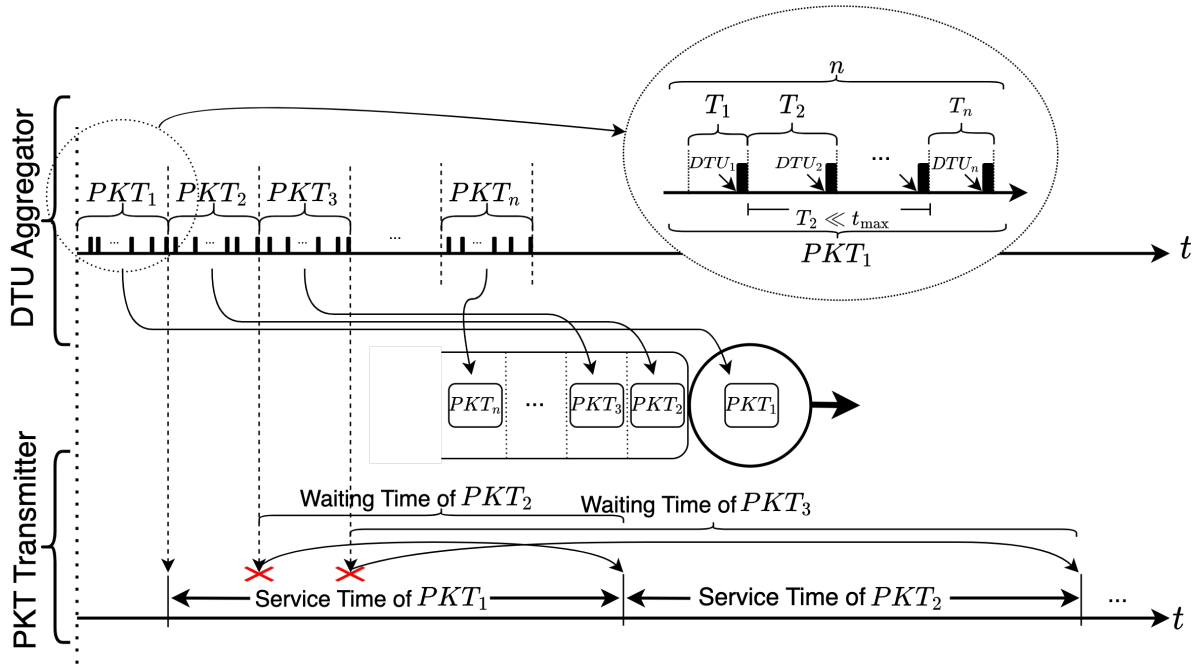



Fig. 14: Illustrative representation of the behavior of DTU Aggregator and Packet Transmitter in the instability region, for $0 < t_{\rm b} < t_{\rm b}^{(\rm min-B)}$, with $t_{\rm b}^{(\rm min-B)} \simeq 0.6$.

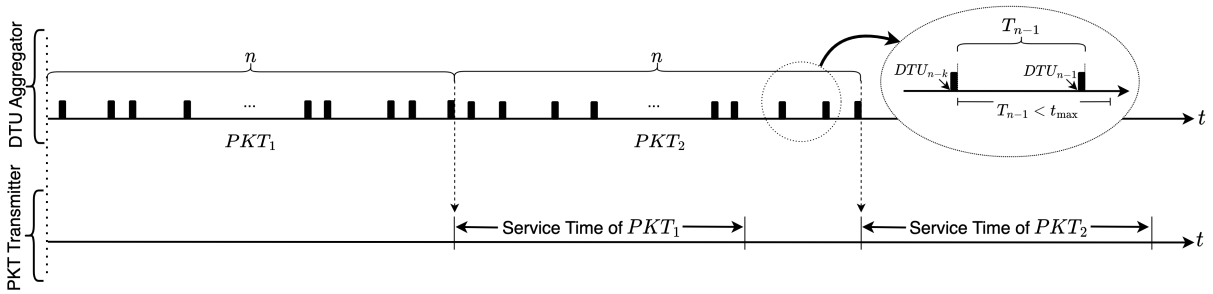



Fig. 15: Illustrative representation of the behavior of DTU Aggregator and Packet Transmitter for $t_{\rm b}^{(\rm min-B)} < t_{\rm b} < t_{\max}$.

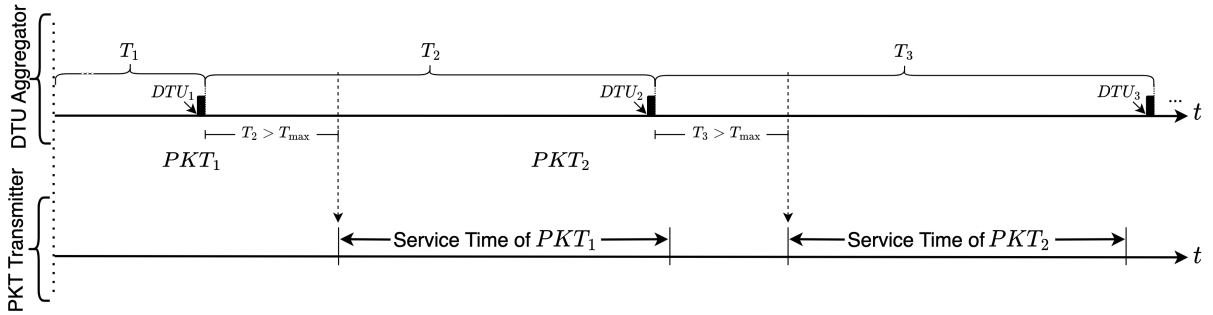
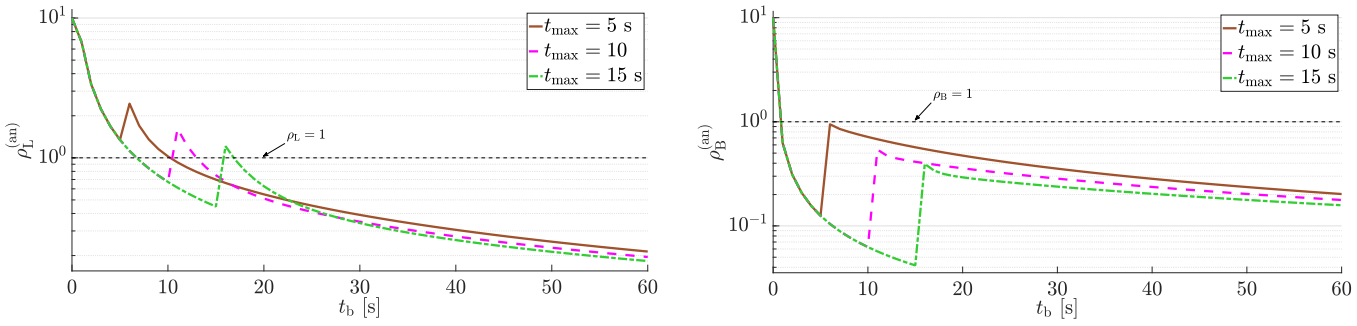



Fig. 16: Illustrative representation of the behavior of DTU Aggregator and Packet Transmitter for $t_{\rm b} \gg t_{\max}$.




(a) (b)

Fig. 17: Analytical server utilization ratio $\rho^{(\mathrm{an})}$ as a function of $t_\mathrm{b}$ (with DTUs' generation distribution $\mathcal{U}[0, t_\mathrm{b}]$), with $t_\mathrm{max}$ set to 5 s, 10 s, and 15 s, for (a) LoRaWAN and (b) BLE interfaces.

interfaces, are due to the DTU Aggregator.

### C. Evaluation of Average Service and Sojourn Times

It is of interest to investigate the behavior of (i) the average service time $\overline{S}$ of each interface's server and (ii) the total average time spent by a DTU at each interface, namely: LoRaWAN ($\overline{T}_{\mathrm{PKT-SOJ_L}}$) and BLE ($\overline{T}_{\mathrm{PKT-SOJ_B}}$), as defined in Eq. (26); and Wi-Fi ($\overline{T}_{\mathrm{DTU-SOJ_W}}$) and cellular ($\overline{T}_{\mathrm{DTU-SOJ_C}}$), as defined in Eq. (33) and Eq. (34), respectively. The obtained results (with the corresponding confidence interval of each simulation point) are shown in Fig. 18. For the sake of clarity, we highlight that, even if these performance metrics have been studied as functions of $t_\mathrm{b} \in [0, 60]$ s, in Fig. 18 the results are shown for $t_\mathrm{b} \in [0, 30]$ s, since for $t_\mathrm{b} \in [30, 60]$ s the performance metrics flatten. In other words, the most interesting behavior is observed for $t_\mathrm{b} \in [0, 30]$ s.

From the results in Fig. 18(a), related to LoRaWAN, it can be noticed that the average service times $\overline{S}_\mathrm{L}^{(\mathrm{an})}$ and $\overline{S}_\mathrm{L}^{(\mathrm{sim})}$ are in very good agreement. In particular, $\overline{S}_\mathrm{L}^{(\mathrm{sim})}$ reaches its saturation value when the DTU aggregation is maximized, i.e., when $t_\mathrm{b} \leq t_\mathrm{max}$. With regard to average analytical ($\overline{T}^{(\mathrm{an})}_{\mathrm{PKT-SOJ_L}}$) and simulated ($\overline{T}^{(\mathrm{sim})}_{\mathrm{PKT-SOJ_L}}$) sojourn times, given the fixed amount of DTUs processed in the simulator (namely, 1,000 as indicated at the beginning of Section IV), it is possible to calculate the average waiting time of a DTU even when the analytical queuing model reaches $\rho_\mathrm{L} = 1$ (for $t_\mathrm{b} \simeq 10$ s), thus confirming the overall behavior of the system predicted by the Markov chain-based model proposed in Section IV.

Similar considerations can be carried out for Fig. 18(b), referring to the BLE interface. It can be observed that both $\overline{S}_\mathrm{B}^{(\mathrm{an})}$ and $\overline{S}_\mathrm{B}^{(\mathrm{sim})}$ reach their maximum possible values when $t_\mathrm{b} \leq t_\mathrm{max}$ (i.e., when the BLE aggregated packet size is maximized), while $\overline{T}^{(\mathrm{an})}_{\mathrm{PKT-SOJ_B}}$ grows rapidly for $t_\mathrm{b} \simeq 0.6$ s, as confirmed by $\overline{T}^{(\mathrm{sim})}_{\mathrm{DTU-SOJ_B}}$. Furthermore, it can be observed both analytical and simulation sojourn times have a peak at $t_\mathrm{b} \simeq t_\mathrm{max} = 0.5$ s, as already detailed.

Finally, Fig. 18(c) and Fig. 18(d), referring to Wi-Fi and cellular interfaces' analytical and simulated service and sojourn times, respectively, confirm how analytical and simulation results are in very good agreement—note that the range of the $y$-axis is limited[10] and $\overline{T}_{\mathrm{DTU-SOJ}}$ grows rapidly at $t_\mathrm{b} \simeq 0$ s (as observed in Fig. 12(c) and Fig. 12(d)).

### D. Impact of SF on LoRaWAN DTUs' Aggregation

With regard to the LoRaWAN interface, it is of interest to investigate the average service time and the number of DTUs that can be aggregated within a single LoRAWAN packet as functions of the SF. The obtained results, both analytical and simulation-based, are shown in Fig. 19. It can be observed that (as suggested by the LoRaWAN specifications [24]) increasing the SF (i) limits the amount of DTUs possibly being aggregated and (ii) significantly increases the average service time needed to process packets with aggregated DTUs, thus significantly increasing $\overline{S}$.

### E. Final Considerations

For the sake of readability and analysis, and to ease a performance comparison between the Markov chain-based model and the implemented Python simulator, in Table II the main performance indicators investigated before, in the case of uniform distribution $\mathcal{U}[t_\mathrm{a}, t_\mathrm{b}]$ of generated DTUs, with $t_\mathrm{a} = 0$ s and $t_\mathrm{b} = 15$ s, are summarized. These results confirm that the most constrained communication interface is the LoRaWAN one, followed by the BLE interface and, then, by the cellular, with the Wi-Fi interface being the best performing.

Finally, through the developed simulator it is possible to evaluate the average data rate, denoted as $\overline{\psi}$, achieved by each communication interface equipping the MIG itself. More in detail, the simulated average data rates (over 1,000 generated DTUs) are:

- $\overline{\psi}_\mathrm{L} = 28.657$ bps for the LoRaWAN interface;
- $\overline{\psi}_\mathrm{B} = 46.725$ bps for the BLE interface;
- $\overline{\psi}_\mathrm{C} = 2{,}139.8$ bps for the cellular interface;
- $\overline{\psi}_\mathrm{W} = 5{,}479.32$ bps for the Wi-Fi interface.

It can be concluded that the LoRaWAN interface has a data rate $\overline{\psi}_\mathrm{L}$ close to the value predicted by the protocol guidelines with the same configuration (namely, 48 bps [24]).

[10]We remark that the seemingly irregular behavior of the curves of both the Wi-Fi and cellular simulated service and sojourn times is due to the reduced range of the $y$-axis.

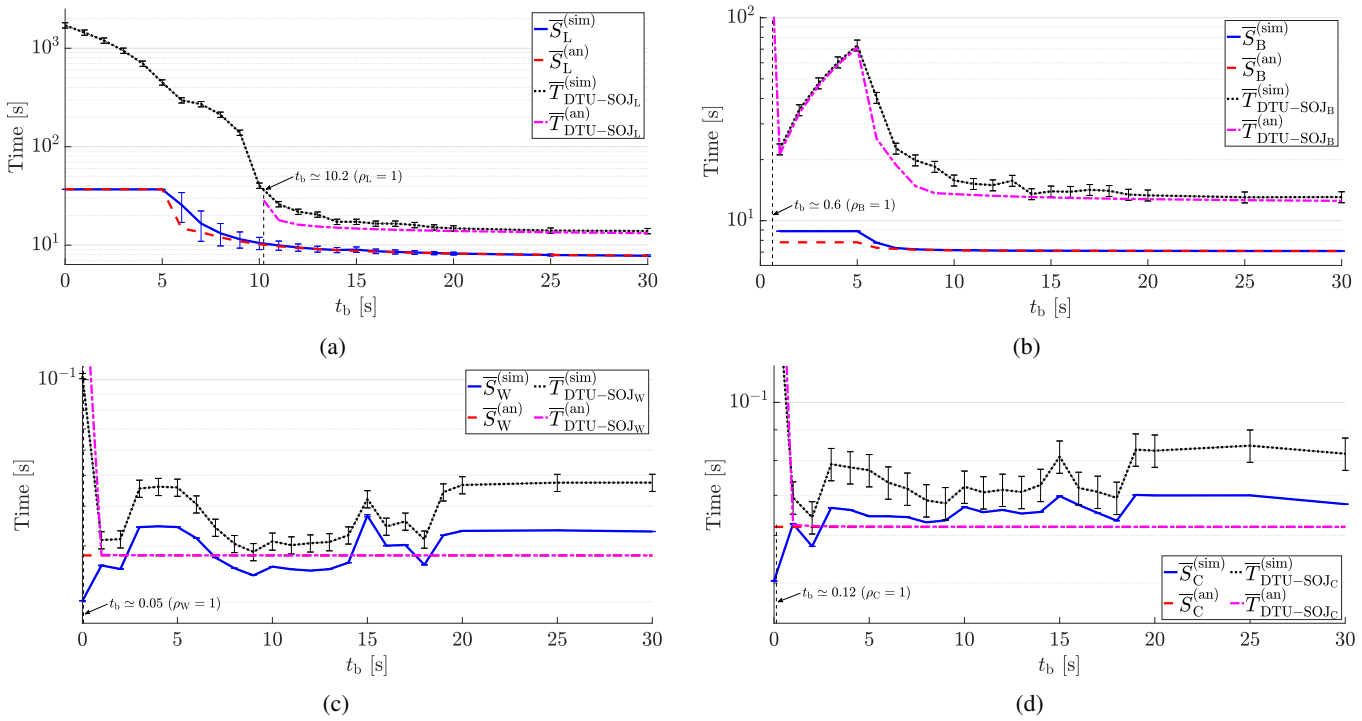



Fig. 18: Direct comparison between analytical queuing average service/sojourn times ($\overline{S}^{(an)}/\overline{T}^{(an)}_{PKT-SOJ}$) and simulation-based average service/sojourn times ($\overline{S}^{(sim)}/\overline{T}^{(sim)}_{PKT-SOJ}$), as functions of $t_b$ (with DTUs' generation distribution $\mathcal{U}[0, t_b]$), at the considered interfaces: (a) LoRaWAN ($\overline{S}_L$ and $\overline{T}_{PKT-SOJ_L}$), (b) BLE ($\overline{S}_B$ and $\overline{T}_{PKT-SOJ_B}$), (c) Wi-Fi ($\overline{S}_W$ and $\overline{T}_{DTU-SOJ_W}$), and (d) cellular ($\overline{S}_C$ and $\overline{T}_{DTU-SOJ_C}$).

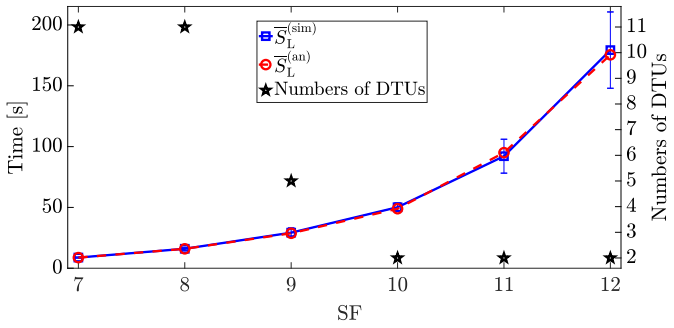



Fig. 19: LoRaWAN analytical ($\overline{S}_L^{(an)}$) and simulated ($\overline{S}_L^{(sim)}$) average service time and number of DTUs per aggregated packet as a function of the SFs allowed by the LoRaWAN protocol.

The other interfaces may be limited by the system's specific implementation, which reduces the useful payload processed by the MIG and introduces a high overhead, thus significantly lowering the achievable data rates. This is especially true for Wi-Fi and cellular communication interfaces. Moreover, the cellular technology is also affected by a higher average latency introduced by the network topology. These constraints are further investigated in Section VII, where the experimental performance evaluation of the MIG prototype introduced at the beginning of Section III is carried out. Possible additional improvements to overcome limitations of the proposed MIG implementation will be discussed in Section VIII.

TABLE II: Comparison among analytical and simulation-based values with $T \sim \mathcal{U}[0, 15]$.

| Term | Analytical Value | Simulation Value |
|---|---|---|
| $\rho_W$ | 0.0037 | 0.0032 |
| $\rho_C$ | 0.0083 | 0.0081 |
| $\rho_B$ | 0.57 | 0.60 |
| $\rho_L$ | 0.70 | 0.75 |
| $\overline{T}_{AGG_L}$ | 12.50 s | 11.85 s |
| $\overline{T}_{AGG_B}$ | 12.50 s | 11.87 s |
| $\overline{W}_q^{(L)}$ | 0.19 s | 1.62 s |
| $\overline{W}_q^{(B)}$ | 0.068 s | 0.18 s |
| $\overline{W}_q^{(W)}$ | 0.000058 s | 0.0025 s |
| $\overline{W}_q^{(C)}$ | 0.00029 s | 0.0038 s |
| $\overline{S}_W$ | 0.028 s | 0.025 s |
| $\overline{S}_C$ | 0.062 s | 0.062 s |
| $\overline{S}_B$ | 7.11 s | 7.12 s |
| $\overline{S}_L$ | 8.73 s | 8.88 s |
| $\overline{T}_{PKT-SOJ_L}$ | 21.42 s | 22.35 s |
| $\overline{T}_{PKT-SOJ_B}$ | 19.68 s | 19.16 s |
| $\overline{T}_{DTU-SOJ_W}$ | 0.028 s | 0.028 s |
| $\overline{T}_{DTU-SOJ_C}$ | 0.062 s | 0.065 s |

## VI. Impact of DTU Inter-Arrival Time Distribution

### A. Inter-Arrival Time Distributions

While in Section IV and Section V the performance of the MIG has been investigated (analytically and with simulations, respectively) in the presence of DTU inter-arrival time $T \in \mathcal{U}[t_a, t_b]$ (with $t_a = 0$ s and several values for $t_b$),

it is of interest to investigate also the impact of non-uniform DTU inter-arrival time distributions. According to Eq. (5), in the LoRaWAN and BLE cases the average input arrival rate at the DTU Aggregator depends on the average inter-arrival time $\overline{T}$ between DTUs. In the following, we investigate the impact of three non-uniform DTU inter-arrival time distributions on system performance: (i) truncated Gaussian, (ii) exponential, and (iii) gamma.

Since, in Section V, the accuracy of our analytical model has been validated by our simulator, in the following we investigate the impact of the input distribution only analytically. More specifically, our goal is to investigate the impact of the input distribution on the server utilization ratio. In the LoRaWAN and BLE cases, we will further evaluate the impact of the following DTU Aggregator.

For the sake of comparison between the aforementioned non-uniform distributions, their corresponding parameters are summarized in the following. In order to make a fair comparison, we will assume that all distributions have the same average value. In order to make a fair comparison with the previously considered uniform distribution, in all cases the average value is set to $\mu = (t_\mathrm{a} + t_\mathrm{b})/2$.

*1) Truncated Gaussian Distribution:* The truncated Gaussian distribution can be derived from an initial Gaussian distribution [37]. The corresponding Probability Density Function (PDF) is expressed as follows:

$$\phi\left(\mu, \sigma^2; x\right) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2} \tag{37}$$

where $\mu = (t_\mathrm{b}+t_\mathrm{a})/2$ and $\sigma^2$ is the variance. By truncating the normal distribution $\mathcal{N}(\mu, \sigma^2)$ over the interval $[t_\mathrm{a}, t_\mathrm{b}]$, the PDF of the corresponding truncated Gaussian distribution becomes

$$\psi(\mu, \sigma^2, t_\mathrm{a}, t_\mathrm{b}; x) = \begin{cases} 0 & \text{if } x \le t_\mathrm{a} \\ \frac{\phi(\mu,\sigma^2;x)}{\Phi(\mu,\sigma^2;t_\mathrm{b}) - \Phi(\mu,\sigma^2;t_\mathrm{a})} & \text{if } t_\mathrm{a} < x < t_\mathrm{b} \\ 0 & \text{if } x \ge t_\mathrm{b}\,. \end{cases} \tag{38}$$

Since the truncated Gaussian distribution is still symmetrical with respect to $\mu$, its mean value is still equal to $\mu = (t_\mathrm{b} + t_\mathrm{a})/2$.

*2) Exponential Distribution (Poisson Arrivals):* The exponential distribution characterizes the inter-arrival time of a Poisson DTU arrival process [37]. The corresponding PDF can be expressed as follows:

$$\theta(\mu; x) = \begin{cases} \frac{1}{\mu} e^{-\frac{x}{\mu}} & \text{if } x \ge 0 \\ 0 & \text{if } x < 0 \end{cases} \tag{39}$$

where $\mu$ is the average value.

*3) Gamma Distribution:* The gamma distribution is a 2-parameter distribution [37] with the following (general) PDF:

$$\gamma(\alpha, \beta; x) = \begin{cases} \frac{\beta^\alpha x^{\alpha-1} e^{-\beta x}}{\Gamma(\alpha)} & \text{if } x > 0 \\ 0 & \text{otherwise} \end{cases} \tag{40}$$

where: $\alpha > 0$ and $\beta > 0$ are shape parameters; and $\Gamma(\alpha)$ is the Gamma function defined as

$$\Gamma(\alpha) \triangleq \int_0^\infty x^{\alpha-1} e^{-x}\, \mathrm{d}x \qquad \text{for } \alpha > 0\,. \tag{41}$$

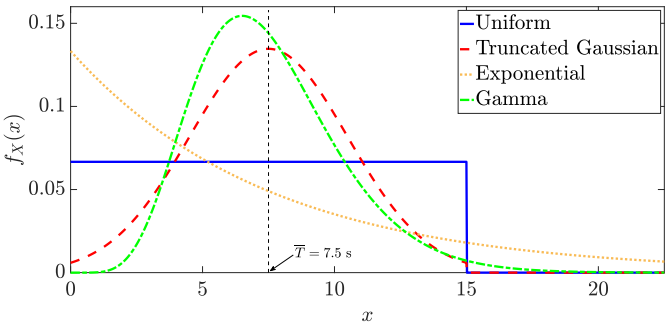



Fig. 20: Comparison plot of the PDFs of the uniform, truncated Gaussian, exponential and gamma distributions, with $t_\mathrm{a} = 0$ s, $t_\mathrm{b} = 15$ s, and $\mu = 7.5$ s.

The average value of the gamma distribution is $\alpha/\beta$: for comparison fairness, we impose $\mu = \alpha/\beta$. In the case with $\beta = 1$, it follows that

$$\alpha = \mu \cdot \beta = \frac{(t_\mathrm{a} + t_\mathrm{b})}{2}\,. \tag{42}$$

*4) Distributions Comparison and Motivation:* In Fig. 20, a graphical representation of the PDFs of the four considered DTU inter-arrival time distribution, with $t_\mathrm{a} = 0$ s and $t_\mathrm{b} = 15$ s, is shown. As indicated above, in all cases the average value is $\mu = (t_\mathrm{a} + t_\mathrm{b})/2 = 7.5$ s. The choice of the aforementioned distributions can be motivated as follows.

The *uniform distribution* is the reference distribution adopted in our analytical, simulation, and experimental performance analysis, since it is suitable to represent the DTU inter-arrival time in several IoT applications, especially where multiple sensors are involved and randomly (over a reference sampling interval) transmit data to the MIG.

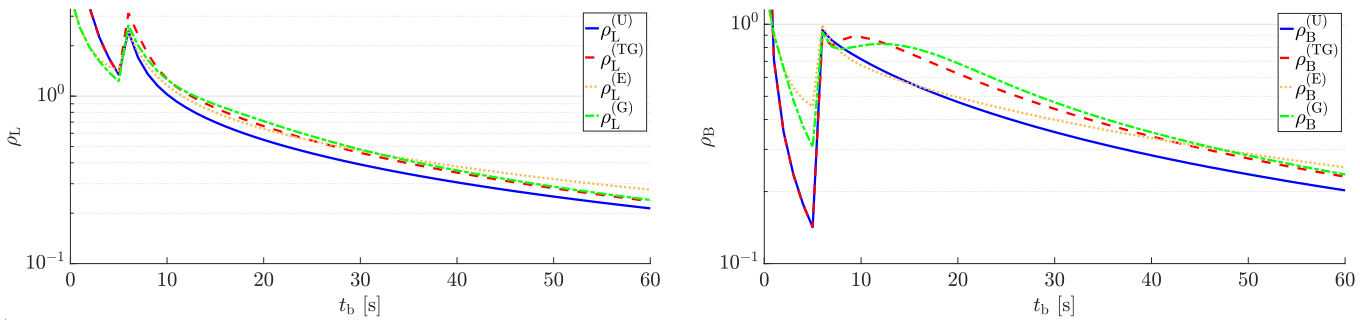
The *truncated Gaussian distribution* has been chosen to characterize an application where the DTU inter-arrival time concentrates more, with respect to the uniform distribution, around its mean value. The truncated Gaussian distribution is more suitable to represent multiple sensors transmitting data to the MIG with a pre-defined polling interval corresponding to the average value. However, due to possible problems (e.g., synchronization errors, packet delays or collisions), some sensors might transmit their data a bit earlier or later than the pre-defined update instant, making the DTU inter-arrival time distribution similar to a truncated Gaussian PDF.

The *exponential distribution* has been taken into account because it is commonly used in queuing theory to model an arrival process. It is meaningful for IoT applications where a "memoryless" data transmission strategy might be adopted by IoT nodes interacting with the MIG.

Finally, the *gamma distribution* has been taken into account as it can be interpreted as an intermediate distribution between Gaussian and exponential distributions.

### B. Impact on the Server Utilization Ratio

We now investigate the impact of the inter-arrival time distributions presented in Subsection VI-A on the server utilization ratio $\rho$. This analysis aims at evaluating the stability




(a) (b)

Fig. 21: Analytical (a) LoRaWAN ($\rho_{\mathrm{L}}$) and (b) BLE ($\rho_{\mathrm{B}}$) server utilization ratio as a function of $t_{\mathrm{b}}$, with $t_{\max} = 5$ s, for different distribution, namely, uniform distribution ($\rho_{\mathrm{L}}^{(\mathrm{U})}$ and $\rho_{\mathrm{B}}^{(\mathrm{U})}$, respectively), truncated Gaussian distribution ($\rho_{\mathrm{L}}^{(\mathrm{TG})}$ and $\rho_{\mathrm{B}}^{(\mathrm{TG})}$, respectively), exponential distribution ($\rho_{\mathrm{L}}^{(\mathrm{E})}$ and $\rho_{\mathrm{B}}^{(\mathrm{E})}$, respectively), and gamma distribution ($\rho_{\mathrm{L}}^{(\mathrm{G})}$ and $\rho_{\mathrm{B}}^{(\mathrm{G})}$, respectively).

of the MIG as a function of the average DTU inter-arrival time $\overline{T}$. In all cases, $t_{\mathrm{a}} = 0$ s, $t_{\mathrm{b}} = 15$ s, $\overline{T} = t_{\mathrm{b}}/2 = 7.5$ s. In the LoRaWAN and BLE cases, $t_{\max}$ is set to 5 s.

In Fig. 21(a), we focus on the LoRaWAN interface. When $t_{\mathrm{b}} < t_{\max}$, the LoRaWAN server utilization ratio with uniform distribution (denoted as $\rho_{\mathrm{L}}^{(\mathrm{U})}$) and with the truncated Gaussian distribution (denoted as $\rho_{\mathrm{L}}^{(\mathrm{TG})}$) have almost the same behavior. At the opposite, when $t_{\mathrm{b}} > t_{\max}$, the truncated Gaussian distribution slightly increases the server utilization ratio. A similar behavior is experienced with both exponential (denoted as $\rho_{\mathrm{L}}^{(\mathrm{E})}$) and gamma (denoted as $\rho_{\mathrm{L}}^{(\mathrm{G})}$) distributions, with $\rho_{\mathrm{L}}^{(\mathrm{E})}$ being higher for high values of $t_{\mathrm{b}}$, while obtaining a higher $\rho_{\mathrm{L}}^{(\mathrm{G})}$ for $t_{\max} < t_{\mathrm{b}} < 30$ s.

In Fig. 21(b), we focus on the BLE interface. The general trends of the BLE server utilization $\rho_{\mathrm{B}}$ (based on the various distributions) is similar to that shown for the LoRaWAN interface in Fig. 21(a). However, $\rho_{\mathrm{B}}$ is more affected by the chosen input distribution. More in detail, when $t_{\mathrm{b}} < t_{\max}$, both the truncated Gaussian and the uniform distribution return the same behavior (in terms of $\rho_{\mathrm{B}}^{(\mathrm{U})}$ and $\rho_{\mathrm{B}}^{(\mathrm{TG})}$), while for $t_{\mathrm{b}} > t_{\max}$, $\rho_{\mathrm{B}}^{(\mathrm{TG})}$ reaches higher values. The difference between gamma and exponential distributions is also accentuated (especially when $t_{\mathrm{b}} < 40$ s), with largest difference for $t_{\mathrm{b}} \simeq 16$ s.

Recalling the presence of the DTU Aggregator for both LoRaWAN and BLE interfaces, it can be observed that: (i) different distributions, even with the same average value, impact on the server utilization of the corresponding Packet Transmitter; and (ii) the uniform distribution offers the best performance, guaranteeing the lowest server utilization ratio for both LoRaWAN and BLE interfaces.

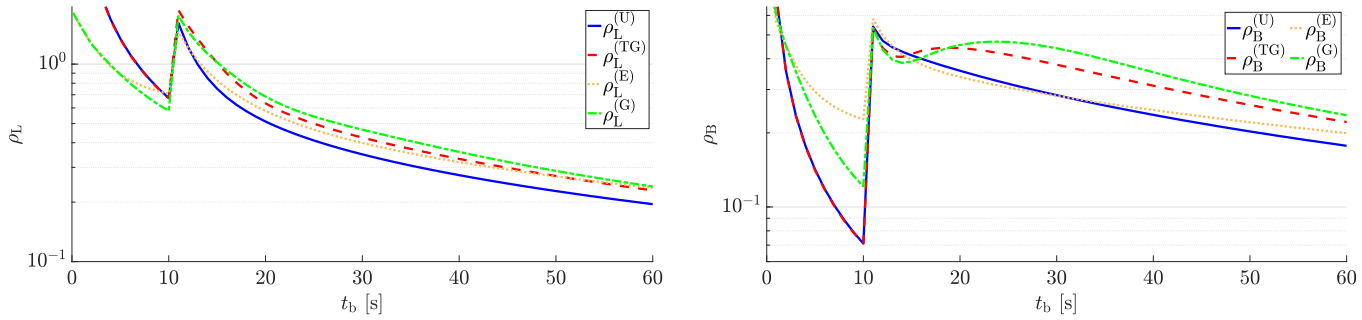
In Section V, we have investigated the presence of a peak for both LoRaWAN and BLE server utilization ratios (due to the DTU Aggregator) with a uniform distribution for various values of $t_{\max}$. In Fig. 22 and Fig. 23, a similar analysis is carried out with the considered non-uniform distributions and two values of $t_{\max}$, namely 10 s and 15 s. As expected, increasing the value of $t_{\max}$ shifts the peak (in correspondence to $t_{\mathrm{b}} \simeq t_{\max}$) to the right and the server utilization ratio decreases for both LoRaWAN and BLE interfaces (regardless of the adopted distribution). However, as discussed in Subsection V-B, a higher value of $t_{\max}$ can increase the packet length (e.g., aggregating more DTUs), but can also slightly increase the waiting time of the DTUs inside the system, thus significantly increasing the latency of the system as well.

Finally, since Wi-Fi and cellular interfaces do not need a DTU Aggregator (with their corresponding servers directly processing the incoming DTUs without any aggregation) and have deterministic service times (much lower than those of LoRaWAN and BLE interfaces), it turns out (as initially expected) that the use of different DTU inter-arrival time distributions has no impact on the server utilization ratios (denoted as $\rho_{\mathrm{W}}$ and $\rho_{\mathrm{C}}$, respectively). This is confirmed, as shown in Fig. 24, by a complete overlap of $\rho_{\mathrm{W}}$ and $\rho_{\mathrm{C}}$ for all considered input distributions (with the same average value). It can thus be concluded that, unlike LoRaWAN and BLE, specifying the DTU inter-arrival time distribution has no impact on high-throughput communication interfaces.

## VII. Experimental Evaluation and Analysis

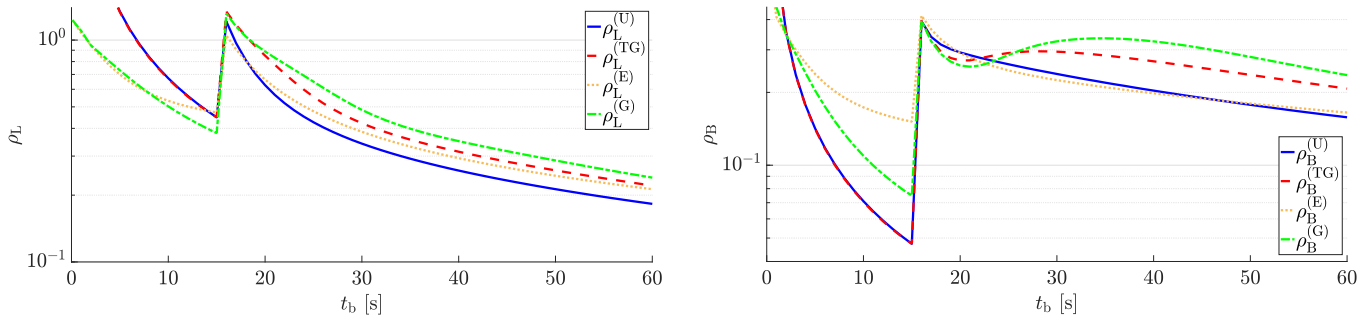
The topology of the overall experimental testbed is shown in Fig. 25(a): the MIG (see Fig. 2a) is connected to 10 IoT nodes (5 BLE and 5 Wi-Fi), based on the ESP32 SoC HW platforms and equipped with either a DHT11 humidity sensor or a temperature sensor [26]. In Fig. 25(b), a node with a DHT11 sensor is shown. The MIG is connected to the closest cellular base station (of the Italian network provider WindTre, identified by Mobile Country Code, MCC = 222 and Mobile Network Code, MNC = 88) through the 4G dongle, and to a LoRaWAN gateway connected to the TTN network, as depicted in Fig. 25(a).

As highlighted in Section III, the starting point of the framework proposed in our work is a COTS device-based implementation of the MIG. While the analytical Markov chain-based queuing model and the Python-based simulator allow to investigate several performance metrics, the experimental testbed allows to investigate a limited number of metrics. From a practical point of view, our prototypical COTS device-based MIG does not allow to evaluate service times, sojourn times, and server utilization ratios of the various interfaces. In order to fairly compare experimental results with analytical and simulation ones, we evaluate the time needed to process a specific



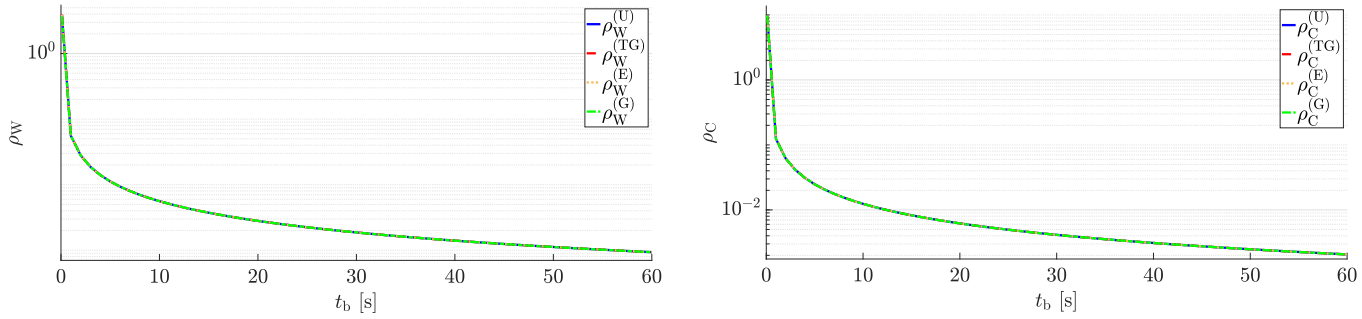

(a) (b)

Fig. 22: Analytical (a) LoRaWAN ($\rho_{\mathrm{L}}$) and (b) BLE ($\rho_{\mathrm{B}}$) server utilization ratio as a function of $t_{\mathrm{b}}$, with $t_{\max} = 10$ s, for different distribution, namely, uniform distribution ($\rho_{\mathrm{L}}^{(\mathrm{U})}$ and $\rho_{\mathrm{B}}^{(\mathrm{U})}$, respectively), truncated Gaussian distribution ($\rho_{\mathrm{L}}^{(\mathrm{TG})}$ and $\rho_{\mathrm{B}}^{(\mathrm{TG})}$, respectively), exponential distribution ($\rho_{\mathrm{L}}^{(\mathrm{E})}$ and $\rho_{\mathrm{B}}^{(\mathrm{E})}$, respectively), and gamma distribution ($\rho_{\mathrm{L}}^{(\mathrm{G})}$ and $\rho_{\mathrm{B}}^{(\mathrm{G})}$, respectively).




(a) (b)

Fig. 23: Analytical (a) LoRaWAN ($\rho_{\mathrm{L}}$) and (b) BLE ($\rho_{\mathrm{B}}$) server utilization ratio as a function of $t_{\mathrm{b}}$, with $t_{\max} = 15$ s, for different distribution, namely, uniform distribution ($\rho_{\mathrm{L}}^{(\mathrm{U})}$ and $\rho_{\mathrm{B}}^{(\mathrm{U})}$, respectively), truncated Gaussian distribution ($\rho_{\mathrm{L}}^{(\mathrm{TG})}$ and $\rho_{\mathrm{B}}^{(\mathrm{TG})}$, respectively), exponential distribution ($\rho_{\mathrm{L}}^{(\mathrm{E})}$ and $\rho_{\mathrm{B}}^{(\mathrm{E})}$, respectively), and gamma distribution ($\rho_{\mathrm{L}}^{(\mathrm{G})}$ and $\rho_{\mathrm{B}}^{(\mathrm{G})}$, respectively).

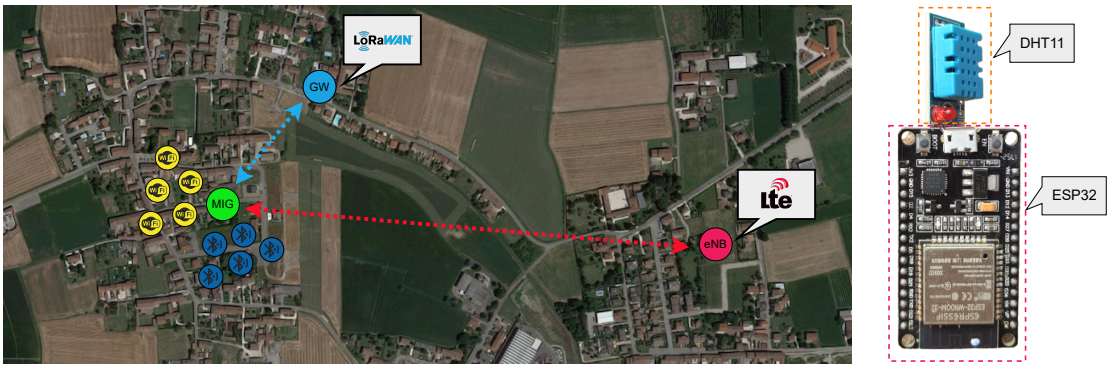



(a) (b)

Fig. 24: Analytical (a) Wi-Fi ($\rho_{\mathrm{W}}$) and (b) cellular ($\rho_{\mathrm{C}}$) server utilization ratio as a function of $t_{\mathrm{b}}$ for different distribution, namely, uniform distribution ($\rho_{\mathrm{W}}^{(\mathrm{U})}$ and $\rho_{\mathrm{C}}^{(\mathrm{U})}$, respectively), truncated Gaussian distribution ($\rho_{\mathrm{W}}^{(\mathrm{TG})}$ and $\rho_{\mathrm{C}}^{(\mathrm{TG})}$, respectively), exponential distribution ($\rho_{\mathrm{W}}^{(\mathrm{E})}$ and $\rho_{\mathrm{C}}^{(\mathrm{E})}$, respectively), and gamma distribution ($\rho_{\mathrm{W}}^{(\mathrm{G})}$ and $\rho_{\mathrm{C}}^{(\mathrm{G})}$, respectively).

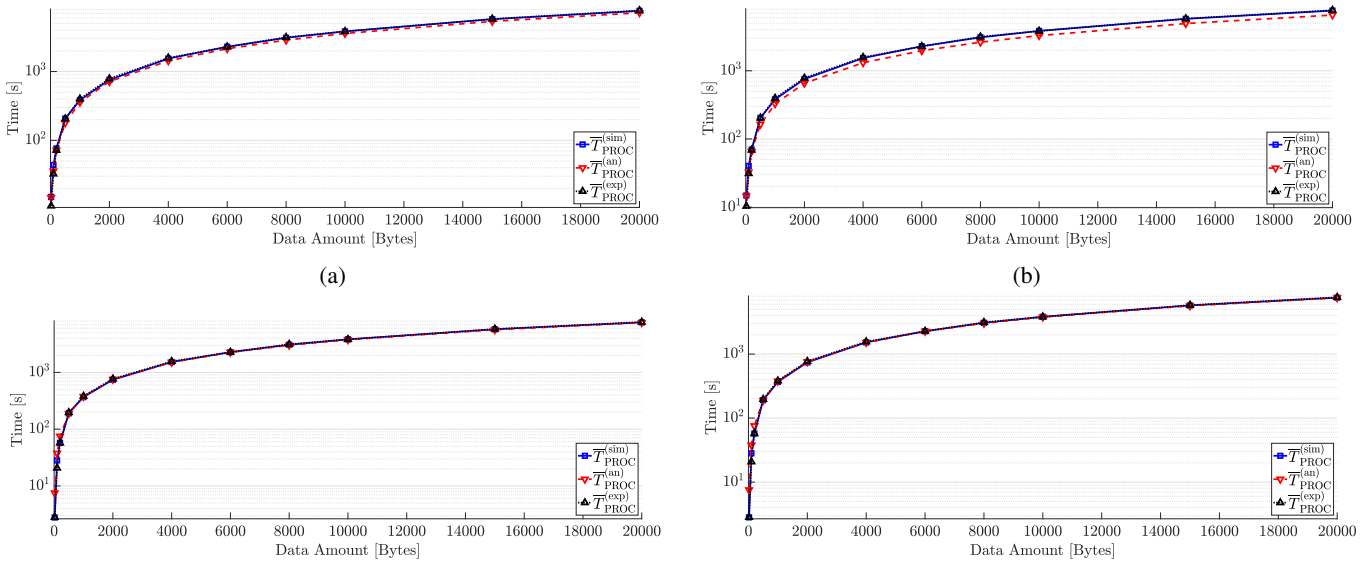
amount of data—namely, a specific number of DTUs—at the various interfaces. To this end, the performance comparison is carried out by setting a common uniform distribution $\mathcal{U}[0, 15]$ for the DTU inter-arrival time $T$ at any interface. This choice allows to compare the time needed by each interface to process a similar amount of DTUs.

In the following, the *experimental processing time* (denoted as $\overline{T}_{\mathrm{PROC}}^{(\mathrm{exp})}$) and the *simulator processing time* (denoted as $\overline{T}_{\mathrm{PROC}}^{(\mathrm{sim})}$) have been obtained by measuring the difference between the time instant of system initialization and the time instant corresponding to processing completion of the last DTU. The *analytical processing time* (denoted as $\overline{T}_{\mathrm{PROC}}^{(\mathrm{an})}$) for Wi-Fi and cellular interfaces has been calculated by multiplying the corresponding average sojourn time ($\overline{T}_{\mathrm{DTU-SOJ_W}}$ and $\overline{T}_{\mathrm{DTU-SOJ_C}}$, respectively) of a DTU by the number of DTUs needed to send the defined amount of information through the specific interface. In fact, this allows to obtain an acceptable




(a) (b)

Fig. 25: (a) Real IoT testbed built for the experimental performance evaluation of the proposed MIG, involving: a RPi4-based MIG implementation; a cellular base station; a LoRaWAN gateway; and 10 IoT nodes (5 BLE and 5 Wi-Fi). As shown in (b), each IoT node is based on the ESP32 SoC and equipped with a DHT11 humidity and temperature sensor. For the sake of readability, the IoT nodes are shown out of scale (with respect to the real distances among the MIG and both LTE and LoRaWAN equipments).




(a) (b) (c) (d)

Fig. 26: Comparison among analytical ($\overline{T}^{(an)}_{PROC}$), simulated ($\overline{T}^{(sim)}_{PROC}$), and experimental ($\overline{T}^{(exp)}_{PROC}$) performance results obtained on the different communication interfaces equipping the MIG, namely (a) LoRaWAN, (b) BLE, (c) Wi-Fi, and (d) cellular interfaces.

estimation of the overall processing time needed to send the chosen amount of data over the designated communication interface.

The analytical, simulated, and experimental performance results for LoRaWAN, BLE, Wi-Fi, and cellular interfaces are shown in Fig. 26(a), Fig. 26(b), Fig. 26(c), and Fig. 26(d), respectively. As can be seen from the results in Fig. 26, the trends obtained through analytical queuing model, simulator, and experimental testbed are very similar, with $\overline{T}^{(exp)}_{PROC}$ slightly higher than both $\overline{T}^{(sim)}_{PROC}$ and $\overline{T}^{(an)}_{PROC}$. This is mainly due to the additional processing times introduced by the current MIG implementation based on COTS components, which may partially degrade the performance with respect to that predicted by both Markov chain-based model and Python simulator.

For LoRaWAN and BLE interfaces (whose analytical, simulated, and experimental results are shown in Fig. 26(a) and Fig. 26(b), respectively), $\overline{T}^{(an)}_{PROC}$ can be expressed as follows:

$$\overline{T}^{(an)}_{PROC} = \Upsilon^{(AGG)}_{DTU} \cdot \Pi_{PKT-SOJ} \quad (43)$$

where: $\Upsilon^{(AGG)}_{DTU} \triangleq N_{DTU}/\overline{n}_{DTU-PKT}$ is the number of aggregated DTUs generated by the DTU Aggregator of LoRaWAN or BLE interfaces, as $N_{DTU}$ corresponds to the amount of DTUs to be sent and $\overline{n}_{DTU-PKT}$ is the average amount of DTUs inside a single aggregated LoRaWAN or BLE packet at the output of the DTU Aggregator; $\Pi_{PKT-SOJ} \triangleq \overline{T}_{PKT-SOJ}/2$ corresponds to the average time spent into the queuing system, where $\overline{T}_{PKT-SOJ}$ represents

the overall time spent by the aggregated DTUs inside LoRaWAN or BLE interfaces, respectively. In detail, $\Pi_{\mathrm{PKT-SOJ}}$ is obtained by dividing $\overline{T}_{\mathrm{PKT-SOJ}}$ by 2 to take into account the overlapping of aggregating and processing activities ot the DTU Aggregator. In fact, while a packet is aggregated by the DTU Aggregator, the Packet Transmitter is processing another packet previously aggregated.

Looking at the results shown in Fig. 26(a) and Fig. 26(b), it can be seen that $\overline{T}^{(\mathrm{an})}_{\mathrm{PROC}}$ (for both LoRaWAN and BLE) is slightly underestimated if compared to both simulation-based and experimental values. This might be due to the approximation of the $\overline{T}_{\mathrm{PKT-SOJ}}$ introduced in Eq. (43). However, the maximum approximation error is lower than $10\%$ for $\overline{T}^{(\mathrm{an})}_{\mathrm{PROC_L}}$ and lower than $15\%$ for $\overline{T}^{(\mathrm{an})}_{\mathrm{PROC_B}}$, respectively, with the relative error decreasing for larger amounts of aggregated DTUs. Instead, simulated and experimental values seem to be aligned (within a maximum $5\%$ difference), thus confirming the overall accuracy of the simulator when compared to the experimental setup.

The obtained experimental results prove that the overall performance of the proposed MIG prototype can be well estimated using both the Markov chain-based model presented in Section IV and the Python-based simulator discussed in Section V.

## VIII. Improvements and Future Works

Given the simulation and experimental performance results shown in Section V, Section VI and Section VII, together with the modularity of the proposed MIG architecture, further improvements might be considered in order to improve its performance. To this end, possible ideas are presented and discussed in the following.

### A. Enhanced Queuing Mechanisms and Packet Overhead Optimization

The use of the proposed SDB (detailed in Section III) for DTUs' queuing purposes partially limits the performance of high throughput interfaces (such as Wi-Fi and cellular). To this end, the implementation of a faster and reliable queuing solution (e.g., based on Zenoh protocol [38]) may further improve both reliability and performance of the MIG, making it applicable to more complex and critical scenarios.

Moreover, the current version of the MIG is not optimized for massive data transfer (through Wi-Fi and cellular interfaces). Hence, an enhanced implementation of the MIG with a more efficient UDP socket creation and management might improve the IP-based protocols' performance. This would significantly reduce the packet overhead currently affecting the prototype and, in turn, improve the overall performance of both cellular and Wi-Fi communication interfaces.

### B. Exploiting BLE Connection Optimization and Advertising

Focusing on the BLE communication interface, as discussed in Subsection IV-B2 the BLE capabilities seem to be mainly constrained by the BLE connection time required by external BLE end nodes to establish a communication link with the MIG. To this end, it might be possible to reduce idle times by scheduling and optimizing the connection phase, allowing also each external BLE node to keep its connection alive and to exchange multiple BLE packets with the MIG. This approach would drastically increase the maximum throughput achievable by the BLE interface handler of the MIG, allowing to approach the theoretical upper bound of the BLE application level data rate.

Furthermore, a different BLE interaction scheme might be adopted among *on-field* end nodes and the MIG, in particular for specific time-constrained applications—e.g., those requiring only mono-directional communication. To this end, one could exploit BLE advertising channels, where BLE end nodes could act as independent GATT servers, broadcasting new information (e.g., collected through their *on-field* sensors) through the Packet Data Units (PDUs), which are then advertised with a small interval on each available BLE advertising channel. The MIG would act as a passive BLE scanner, sensing for the available PDUs in the air and processing them according to the rules detailed in Table I. As a consequence, this would require the implementation of a new interface entity in the architecture shown in Fig. 2(b), having to (i) passively scan and detect BLE devices advertising PDUs and (ii) forward the PDUs to the SDB in the proper way. Given the experimentally observed BLE connection time $\overline{T}^{(\mathrm{exp})}_{\mathrm{CONN_B}} \simeq 7$ s, it is clear that this alternative approach would allow to handle a larger number of end nodes and to obtain a higher throughput on the BLE communication interface (despite the limited 27 byte advertisement packet size) [39].

### C. AI-based Interface Throughput Optimization

In order to optimize data transmission with a constrained protocol (e.g., LoRaWAN), the design and deployment of an *on-board* "data analysis block," in charge of deciding if an information is relevant (and needs to be forwarded to high-layer entities) or irrelevant (and can thus be discarded, avoiding to unnecessarily fill the MIG's queues), might be beneficial. More in detail, this data analysis block may be performed by an AI-based entity assigning proper weights (corresponding to the priorities in the system's queue) to data, thus leading to priority-based data transmission (in which irrelevant data will be assigned small weights). This approach could, in principle, increase the performance of the MIG's constrained communication interfaces, prioritizing only relevant data transfers, eventually exploiting innovative AI technologies able to improve and optimize data transfer [40]. Moreover, this can open the way to other next-generation paradigms to be exploited in conjunction with the functionalities proper of the proposed MIG (e.g., serverless and quantum computing).

## IX. Conclusions

In this paper, the architecture of an enhanced modular MIG, equipped with four heterogeneous communication interfaces (namely, LoRaWAN, BLE, LTE, and Wi-Fi), has been presented, with reference to a specific prototypical COTS device-based implementation. A novel Markov chain-based queuing model, expedient to characterize the MIG behavior

and evaluate its performance, has been proposed. A Python-based software simulator has been implemented to further validate the Markov chain-based queuing model's predicted analytical performance. Finally, the experimental performance of the COTS device-based MIG prototype has been directly compared to analytical and simulation-based performances, showing a very good agreement. Our results highlight the modularity and scalability of the proposed MIG architecture, allowing the integration of heterogeneous communication interfaces for several IoT applications and scenarios. Possible MIG improvements have also been discussed.

## ACKNOWLEDGMENT

This work received funding from the European Union's Horizon 2020 research and innovation program ECSEL Joint Undertaking (JU) under grant agreements: No. 876019, ADACORSA project - "Airborne Data Collection on Resilient System Architectures;" No. 876038, InSecTT project - "Intelligent Secure Trustable Things;" No. 876487, NextPerception project - "Next Generation Smart Perception Sensors and Distributed Intelligence for Proactive Human Monitoring in Health, Wellbeing, and Automotive Systems." It has also received funding from the European Union's Horizon Europe research and innovation program Key Digital Technology (KDT) JU under grant agreement No. 101097267, OPEVA project - "OPtimization of Electric Vehicle Autonomy." Finally, we acknowledge also partial support from the Agritech project - "National Research Centre for Agricultural Technologies," project code CN00000022, funded under the National Recovery and Resilience Plan (NRRP), Mission 4 Component 2 Investment 1.4 - Call for tender no. 3138 of 16/12/2021 of Italian Ministry of University and Research funded by the European Union – NextGenerationEU, Concession Decree no. 1032 of 17/06/2022 adopted by the Italian Ministry of University and Research. The ECSEL/KDT JUs received support from the European Union's Horizon 2020/Horizon Europe research and innovation programme and the nations involved in the mentioned projects. The work reflects only the authors' views; the European Commission is not responsible for any use that may be made of the information it contains.

## REFERENCES

[1] L. Davoli *et al.*, "From Micro to Macro IoT: Challenges and Solutions in the Integration of IEEE 802.15.4/802.11 and Sub-GHz Technologies," *IEEE Internet of Things Journal*, vol. 5, no. 2, pp. 784–793, April 2018, doi:10.1109/JIOT.2017.2747900.

[2] J. Ding *et al.*, "IoT Connectivity Technologies and Applications: A Survey," *IEEE Access*, vol. 8, pp. 67 646–67 673, 2020, doi:10.1109/ACCESS.2020.2985932.

[3] M. R. Palattella *et al.*, "Internet of Things in the 5G Era: Enablers, Architecture, and Business Models," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 3, pp. 510–527, 2016, doi:10.1109/JSAC.2016.2525418.

[4] K. Mekki *et al.*, "Overview of Cellular LPWAN Technologies for IoT Deployment: Sigfox, LoRaWAN, and NB-IoT," in *2018 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, Athens, Greece, 2018, pp. 197–202, doi:10.1109/PERCOMW.2018.8480255.

[5] C. Chen, M. Lin, and C. Liu, "Edge Computing Gateway of the Industrial Internet of Things Using Multiple Collaborative Microcontrollers," *IEEE Network*, vol. 32, no. 1, pp. 24–32, 2018, doi:10.1109/MNET.2018.1700146.

[6] A. Ansari *et al.*, "Multi-interface wireless adapter and network bridge," US Patent US8 649 386B2, 09, 2007.

[7] B. da Silva Campos *et al.*, "Design and Construction of Wireless Sensor Network Gateway with IPv4/IPv6 Support," in *2011 IEEE International Conference on Communications (ICC)*, Kyoto, Japan, 2011, pp. 1–5, doi:10.1109/icc.2011.5962848.

[8] R. Ranjan *et al.*, "City Data Fusion: Sensor Data Fusion in the Internet of Things," *Int. J. Distrib. Syst. Technol.*, vol. 7, no. 1, pp. 15–36, 1 2016, doi:10.4018/IJDST.2016010102.

[9] B. Oniga, A. Munteanu, and V. Dadarlat, "Open-source multi-protocol gateway for Internet of Things," in *2018 17th RoEduNet Conference: Networking in Education and Research (RoEduNet)*, Cluj-Napoca, Romania, 2018, pp. 1–6, doi:10.1109/ROEDUNET.2018.8514136.

[10] S. Salsano *et al.*, "PMSR–Poor Man's Segment Routing, a Minimalistic Approach to Segment Routing and a Traffic Engineering Use Case," in *2016 IEEE/IFIP Network Operations and Management Symposium (NOMS)*, Istanbul, Turkey, Apr 2016, pp. 598–604, doi:10.1109/NOMS.2016.7502864.

[11] L. Davoli *et al.*, "Traffic Engineering with Segment Routing: SDN-based Architectural Design and Open Source Implementation," in *2015 Fourth European Workshop on Software Defined Networks (EWSDN)*. Bilbao, Spain: IEEE, Sep 2015, pp. 111–112, doi:10.1109/EWSDN.2015.73.

[12] F. Jazayeri, A. Shahidinejad, and M. Ghobaei-Arani, "A Latency-aware and Energy-efficient Computation Offloading in Mobile Fog Computing: A Hidden Markov Model-based Approach," *The Journal of Supercomputing*, vol. 77, no. 5, pp. 4887–4916, Oct 2021, doi:10.1007/s11227-020-03476-8.

[13] H. Cui *et al.*, "Cloud Service Reliability Modelling and Optimal Task Scheduling," *IET Communications*, vol. 11, no. 2, pp. 161–167, Jan 2017, doi:10.1049/iet-com.2016.0417.

[14] W. Zhou *et al.*, "Markov Approximation for Task Offloading and Computation Scaling in Mobile Edge Computing," *Mobile Information Systems*, vol. 2019, pp. 1–12, Jan 2019, doi:10.1049/iet-com.2016.0417.

[15] S. Sefati and N. J. Navimipour, "A QoS-Aware Service Composition Mechanism in the Internet of Things Using a Hidden-Markov-Model-Based Optimization Algorithm," *IEEE Internet of Things Journal*, vol. 8, no. 20, pp. 15 620–15 627, 2021, doi:10.1109/JIOT.2021.3074499.

[16] B. Liu *et al.*, "Congestion-Optimal WiFi Offloading with User Mobility Management in Smart Communications," *Wireless Communications and Mobile Computing*, vol. 2018, pp. 1–15, Aug 2018, doi:10.1049/iet-com.2016.0417.

[17] A. Samir and C. Pahl, "DLA: Detecting and Localizing Anomalies in Containerized Microservice Architectures Using Markov Models," in *2019 7th International Conference on Future Internet of Things and Cloud (FiCloud)*, Istanbul, Turkey, 2019, pp. 205–213, doi:10.1109/FiCloud.2019.00036.

[18] A. K. Sangaiah *et al.*, "Enforcing Position-Based Confidentiality With Machine Learning Paradigm Through Mobile Edge Computing in Real-Time Industrial Informatics," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 7, pp. 4189–4196, 2019, doi:10.1109/TII.2019.2898174.

[19] R. B. Sørensen, D. M. Kim, J. J. Nielsen, and P. Popovski, "Analysis of Latency and MAC-layer Performance for Class A LoRaWAN," *IEEE Wireless Communications Letters*, vol. 6, no. 5, pp. 566–569, 2017, doi:10.1109/LWC.2017.2724700.

[20] M. Capuzzo, D. Magrin, and A. Zanella, "Mathematical Modeling of LoRaWAN Performance with Bi-directional Traffic," in *2018 IEEE Global Communications Conference (GLOBECOM)*, Abu Dhabi, UAE, 2018, pp. 206–212, doi:10.1109/GLOCOM.2018.8647351.

[21] A. Nikoukar *et al.*, "Empirical Analysis and Modeling of Bluetooth Low Energy (BLE) Advertisement Channels," in *2018 17th Annual Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net)*, Capri, Italy, 2018, pp. 1–6, 10.23919/MedHocNet.2018.8407089.

[22] A. Altamimi and R. Ramadan, "Towards Internet of Things Modeling: A Gateway Approach," *Complex Adaptive Systems Modeling*, vol. 4, 11 2016, doi:10.1186/s40294-016-0038-3.

[23] "LoRa GPS HAT for Raspberry Pi," Accessed on June 20, 2023. [Online]. Available: https://www.dragino.com/products/lora/item/106-lora-gps-hat.html

[24] "LoRa and LoRaWAN Specifications," Accessed on June 20, 2023. [Online]. Available: https://lora-developers.semtech.com/documentation/tech-papers-and-guides/lora-and-lorawan/

[25] "Espressif ESP32 System-on-Chip (SoC)," Accessed on June 20, 2023. [Online]. Available: https://www.espressif.com/en/products/socs

[26] "DHT11 Humidity & Temperature Sensor," Accessed on June 20, 2023. [Online]. Available: https://www.mouser.com/datasheet/2/758/DHT11-Technical-Data-Sheet-Translated-Version-1143054.pdf

[27] G. Codeluppi, L. Davoli, and G. Ferrari, "Forecasting Air Temperature on Edge Devices with Embedded AI," *Sensors*, vol. 21, no. 12, 2021, doi:10.3390/s21123973.
[28] G. Codeluppi *et al.*, "LoRaFarM: A LoRaWAN-Based Smart Farming Modular IoT Architecture," *Sensors*, vol. 20, no. 7, 2020, doi:10.3390/s20072028.
[29] O. Vermesan *et al.*, "Automotive Intelligence Embedded in Electric Connected Autonomous and Shared Vehicles Technology for Sustainable Green Mobility," *Frontiers in Future Transportation*, vol. 2, 2021, doi:10.3389/ffutr.2021.688482.
[30] L. Davoli *et al.*, "On Driver Behavior Recognition for Increased Safety: A Roadmap," *Safety*, vol. 6, no. 4, pp. 1–33, 12 2020, doi:10.3390/safety6040055.
[31] L. Davoli, E. Pagliari, and G. Ferrari, "Hybrid LoRa-IEEE 802.11s Opportunistic Mesh Networking for Flexible UAV Swarming," *Drones*, vol. 5, no. 2, 2021, doi:10.3390/drones5020026.
[32] "LoRaWAN 1.0.3 Regional Parameters," Accessed on June 20, 2023. [Online]. Available: https://lora-alliance.org/wp-content/uploads/2020/11/lorawan_regional_parameters_v1.0.3reva_0.pdf
[33] K. J. F. C., "The single server queue in heavy traffic," *Mathematical Proceedings of the Cambridge Philosophical Society*, vol. 57, no. 4, pp. 902–904, 1961, doi:10.1017/S0305004100036094.
[34] "SX1272/3/6/7/8 LoRa Modem Design Guide," Semtech Corporation, 2013, accessed on June 20, 2023. [Online]. Available: https://www.openhacks.com/uploadsproductos/loradesignguide_std.pdf
[35] S. M. Ross, *Introduction to Probability Models*. Academic Press, 2014.
[36] D. Freedman, R. Pisani, and R. Purves, *Statistics: Fourth International Student Edition*. W. W. Norton & Company, 2007.
[37] G. C. R. Montgomery Douglas C, *Applied Statistics and probability for Engineers*. Wiley, 2011.
[38] Eclipse Foundation, "Zenoh: Zero Overhead Pub/Sub, Store/Query and Compute," Accessed on June 20, 2023. [Online]. Available: https://zenoh.io/
[39] L. Davoli, M. Moreni, and G. Ferrari, *A Sink-oriented Routing Protocol for Blue Light Link-based Mesh Network*, ser. Telecommunications. Institution of Engineering and Technology, 2022, pp. 21–31, doi:10.1049/PBTE101E_ch2.
[40] S. S. Gill *et al.*, "AI for Next Generation Computing: Emerging Trends and Future Directions," *Internet of Things*, vol. 19, p. 100514, 2022, doi:10.1016/j.iot.2022.100514.

**Emanuele Pagliari** is a Ph.D. student at the Internet of Things (IoT) Laboratory, Department of Engineering and Architecture, University of Parma, Italy. He obtained his Dr. Ing. degree in communication engineering at the Department of Information Engineering of the same university in 2020. His research interests focus on IoT, wireless communication systems, UAVs and heterogeneous networking.

**Luca Davoli** (GS'15, M'17) is a non-tenured Assistant Professor at the Internet of Things (IoT) Laboratory, Department of Engineering and Architecture, University of Parma, Italy. He obtained his Dr. Ing. degree in computer engineering and his Ph.D. in Information Technologies at the Department of Information Engineering of the same university, in 2013 and 2017, respectively. He is a Research Scientist at things2i ltd., a spin-off of the University of Parma dedicated to IoT and smart systems. His research interests focus on IoT, Pervasive Computing, Big Stream and Software-Defined Networking.

**Gianluigi Ferrari** (S'96–M'98–SM'12) received the Laurea (*summa cum laude*) and Ph.D. degrees in electrical engineering from the University of Parma, Parma, Italy, in 1998 and 2002, respectively. Since 2002, he has been with the University of Parma, where he is currently a Full Professor of telecommunications and also the coordinator of the Internet of Things (IoT) Laboratory, Department of Engineering and Architecture. He is co-founder and President of things2i ltd., a spin-off of the University of Parma dedicated to IoT and smart systems. His current research interests include signal processing, advanced communication and networking, IoT and smart systems.