ISBN: 978-628-95207-4-3. ISSN: 2414-6390. Digital Object Identifier: https://dx.doi.org/10.18687/LACCEI2023.1.1.1172

Towards the prototyping of novel access schemes for IoT networks

Pastor David Chavez Muñoz[1], Julio Cesar Manco Vasquez[2], and Cesar David Cordova Bernuy[3]
[1,2,3]*Pontificia Universidad Catolica del Perú, Perú, dchavez@pucp.edu.pe, manco.jc@pucp.edu.pe, cdcordova@pucp.edu.pe*

Abstract– ***With the increasing number of connected IoT devices for 5G networks, numerous access schemes have been conceived in the literature. Although several cross-layer (PHY & MAC) approaches have been reported in the literature, some aspects regarding their implementation on testbeds are overlooked. In this work, we address the prototyping of low power wide area (LPWA) networks aiming to fill the existing mismatching performance between theoretical and experimental results. In doing so, the main aspects and current challenges are explored with the prototyping of access schemes using reinforcement learning (RL) techniques. The missing implementation aspects omitted during the modelling of these networks are reviewed and addressed by employing software-defined radio (SDR) approaches. Finally, at the light of the exposed implementations, some guidelines and suggestions for the development of this technology are provided.***

Keywords - IoT, LPWA, RL, SDR, prototyping.

Towards the prototyping of novel access schemes for IoT networks

Pastor David Chavez Muñoz[1], Julio Cesar Manco Vasquez[2], and Cesar David Cordova Bernuy[3]
[1,2,3]*Pontificia Universidad Catolica del Perú, Perú, dchavez@pucp.edu.pe, manco.jc@pucp.edu.pe, cdcordova@pucp.edu.pe*

Abstract*– With the increasing number of connected IoT devices for 5G networks, numerous access schemes have been conceived in the literature. Although several cross-layer (PHY & MAC) approaches have been reported in the literature, some aspects regarding their implementation on testbeds are overlooked. In this work, we address the prototyping of low power wide area (LPWA) networks aiming to fill the existing mismatching performance between theoretical and experimental results. In doing so, the main aspects and current challenges are explored with the prototyping of access schemes using reinforcement learning (RL) techniques. The missing implementation aspects omitted during the modelling of these networks are reviewed and addressed by employing software-defined radio (SDR) approaches. Finally, at the light of the exposed implementations, some guidelines and suggestions for the development of this technology are provided.

***Keywords* - IoT, LPWA, RL, SDR, prototyping.**

I. INTRODUCTION

IoT networks promises to support a vast variety of applications where the communication among a large number of devices is required. In this regard, LPWA technology [1] has been conceived to provide long-range communications with low power and low data rate end-devices, being suitable for machine-to-machine (M2M) solutions, such as smart cities and metering applications. LoraWAN and SigFox are some of these technologies currently employed in the monitoring in large scale.

Typically, in these networks a large number of low-power devices send short packets to a base station or gateway to be continuously monitored in different narrowband channels. However, as more devices share the same frequency band, the collisions are inevitable. For that end, the handling of a large number of long-range transmissions plays an important role for the development of this technology. It has motivated the design of novel collision-avoidance methods based on cross-layer designs to avoid degrading the network performance. A LPWA scenario is described in Fig. 1, where collisions among IoT devices and interference from other networks occur degrading the performance of the network.

Digital Object Identifier: (only for full papers, inserted by LACCEI).
ISSN, ISBN: (to be inserted by LACCEI).
DO NOT REMOVE

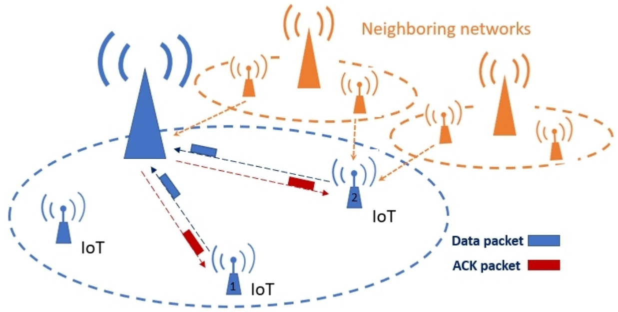


Fig. 1: A LPWA scenario with IoT devices where collisions among other IoT devices and interference from other networks requires to introduce novel access schemes.

RL techniques are probably the most reported approach in the literature to incorporate novel access schemes. In these approaches, the IoT nodes make autonomous decisions related to the channel selection for packet transmissions [2], [3], [4]. These decisions are based on the rewards obtained during the employment of different frequency channels. In doing so, if the frequency channel is free, the IoT node often receives acknowledgment (ACK) packets (i.e. rewards) from the gateway, thus this channel will be used more frequently.

In this work, the prototyping of novel access schemes are addressed. Commonly, these approaches are conceived using simulation models. However, these yield misleading conclusions each time relevant aspects of their implementations on testbeds are omitted. In this regard, prototypes are built to study novel proposals under real-world conditions, and validate the simulation results in true-to-life circumstances. We present the prototyping of a RL approach tailored for IoT networks that allows us to illustrate these issues. In doing so, the main PHY and MAC aspects for this implementation are examined in order to fill the gap between the theory and experimental results. Next, the main challenging for prototyping these networks are exposed and addressed by presenting approaches with the selected SDR tool. Finally, some guidelines are discussed and suggested for the development of this technology.

The rest of the paper is organized as follows. The system model is presented in Section II, while the prototyping of a RL approach is described in Section III. In Section IV, the challenges for prototyping IoT networks as well as the SDR approaches are exposed. Finally, some conclusions are given in Section V.

II. SYSTEM MODEL

We consider a LPWA network composed of a gateway and a large number of end-devices that regularly send short data packets, where K channels ($K > 1$) are available for the transmission of their packets. This network is constituted by two types of IoT devices: *static* devices that operate in one channel in order to communicate with the gateway, and *dynamic* devices that possess the additional advantage of being able to select any of the K available channels to perform their transmissions.

Dynamic devices are able to select a channel frequency by resorting to RL techniques, and we consider, without loss of generality, the implementation of multi-armed bandit (MAB) techniques quite often cited in the literature. In this approach, an IoT device chooses a channel that maximizes the sum of the collected rewards. It is carried out by taking into account the *ACK* packet received after a packet transmission to the gateway. In this way, the channel where more *ACK* packets are received will be often selected since it is available, thus avoiding collisions from the channels occupied by the transmissions of several *static* devices.

For this reason, the incorporation of RL techniques tailored for IoT devices provides significant gains in comparison to those devices that use a fixed channel or randomly choose any channel. This advantage can be observed in Fig. 2, where the introduction of a RL approach brings significant advantages in terms of successful transmission rate[1].

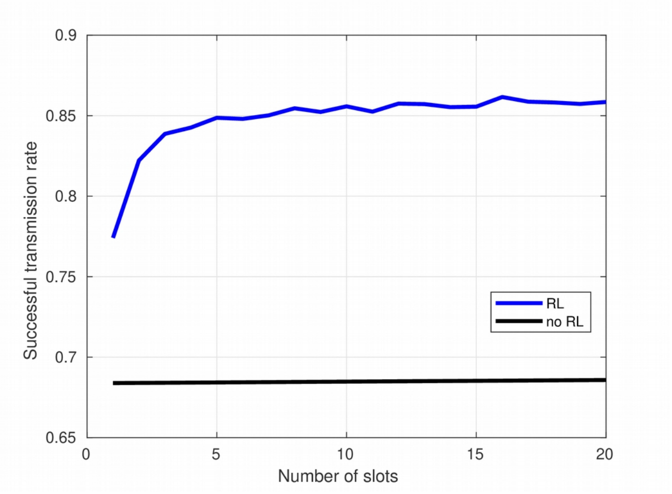


Fig. 2: Successful transmission rate versus number of slots, with the incorporation of a RL technique and without it (non RL).

III. TOWARDS THE PROTOTYPING OF IOT NETWORKS

The design of a novel approach typically employs programming languages at high-level, such as MATLAB, C/ C++, or Python, allowing the researchers to address the targeted design by just providing algorithmic specifications. In addition, tools such as ns-3 [6], or OMNeT++ [7], are also employed since they provide libraries specifically designed for simulations. Furthermore, simulators tailored for IoT networks are developed for these purposes [8].

Prototypes are then built to study its behavior under real-world conditions, and to ensure that there are no technological flaws in the design preventing the product from realization, and to validate the simulation results. To clarify it, we have illustrated in Fig. 3, the process of conceiving new approaches. It is composed of three stages: modeling, the prototyping and emulations, and real-field measurements. Once an approach is evaluated through simulations during the modeling process, it is validated using prototypes. In this stage, the emulations are also useful to recreate some components of an IoT scenario, whenever the complexity is higher for prototyping the whole network. Finally, the approach is evaluated using real-field measurements. It is also worth mentioning that through this process there exists a bidirectional flow between these stages to highlight that missing aspects in one of these stages lead us to review the other stages.

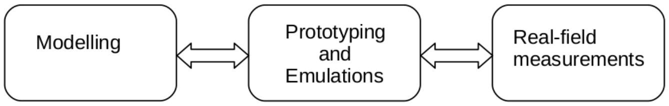


Fig. 3: From ideas conceived in simulation environments to the assessment of prototypes with real measurements.

Next, we describe the main components for prototyping these networks.

A. Prototyping

In Fig. 4, a transceiver is shown as the main component for implementing the *dynamic/static* IoT devices and the gateway. This component allows the communication between IoT devices and the gateway, i.e. the packet and *ACK* transmissions. In addition, a dynamic device incorporates a module introducing a RL approach that will interact with the transceiver. For instance this module will select the channel frequency for packet transmissions.

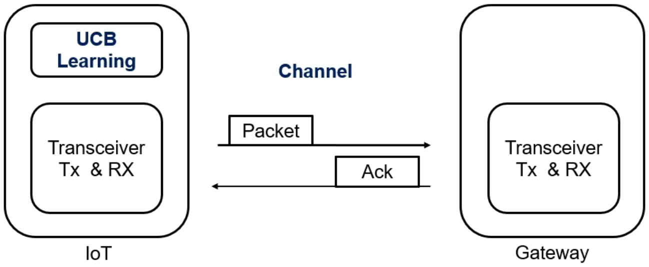


Fig. 4: A dynamic IoT device and gateway composed by a transceiver.

---

[1] In this illustrative example, an upper-confidence bound (UCB) for packet retransmissions is tested, for more details the reader may refer to [5].

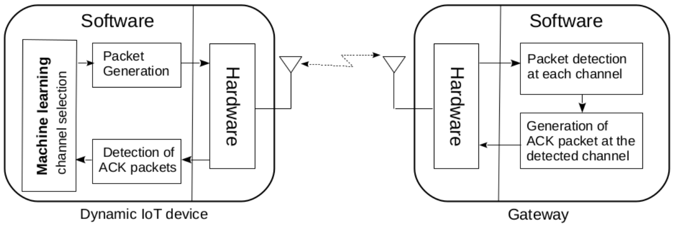


Fig. 5: Hardware and software components of the IoT and gateway.

A SDR implementation of this framework can be further clarified in Fig. 5, where hardware and software components are highlighted. Typically, hardware devices such as USRP [9], PlutoSDR [10], RTL-SDR [11] are employed, while the main efforts are devoted to the software implementation. At the IoT side, three main process are identified. The detection of an *ACK* packet, the RL approach for the channel selection, and finally the generation of a packet for its transmission over the air. In a similar way, at the gateway side, the packet detection is carried out simultaneously at different channels, after which the *ACK* packets are sent to the corresponding IoT devices. Having these processes in mind, the selection of a SDR tool takes place. There exists several tools such as MATLAB/Simulink [12], LabVIEW [13], and GNU Radio [14]. MATLAB along with its high-level language allows a rapid prototyping, however it is often employed for off-line processing, and the implementation of the exposed scenario requires real-time processing as well as scalable designs. In this regard, LabVIEW provides these features with the incorporation of toolboxes. Nevertheless, it requires non-free licenses. On the other hand, GNU Radio is an open-source SDR tool tailored for real-time processing, with a growing community of users providing numerous projects focused on the implementation of wireless communication systems. For this reason, it is the selected SDR tool to meet the requirements of prototyping the aforementioned scenario.

B. *Experimental results*

Preliminary tests to assess the link between the IoT and the gateway are carried out, and some metrics are computed such as symbol error rate and the ratio of loss packets over the total number of packets.

TABLE I

PACKET ERROR RATE

Distance	Loss packets	Transmitted packets	Ratio
0 m	13	783	0.0166
3 m	33	615	0.0537

Table I shows us the packet error rate (PER) obtained for different distances between the IoT and the gateway, where the loss packets can be explained due to the presence of people around. Next, a setup depicted in Fig. 6 is deployed. It is composed of a gateway and two IoT devices which are configured as dynamic devices. These IoT devices are able to choose among 4 frequency channels to transmit their packets to the gateway. The hardware employed is shown in Fig. 7 and it is composed of USRP devices.

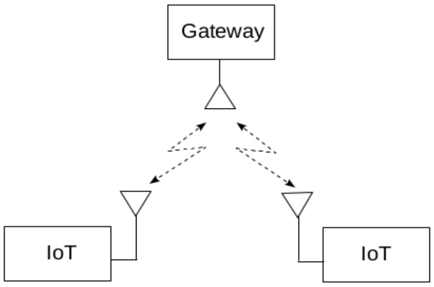


Fig. 6: Deployed network to assess a RL approach

Fig. 7: USRP devices from National Instrument for prototyping.

The learning capability to avoid collisions can be observed in Fig. 8, where two IoT devices select the channels (described by the indices 3 and 7 in the plot) among 4 channels, so as to avoid packet collisions among them. For each peak, two curves are almost superimposed (in blue and red) corresponding to the number of transmitted packets to the gateway and the number of *ACK* packets received from the gateway, since collisions barely occur by transmitting at these frequency channels[2]. Furthermore, a graphical user interface (GUI), is also shown in Fig. 9, where packet delivery ratio (PDR) and the spectrogram of the transmitted/received packets can be depicted for monitoring and debugging purposes.

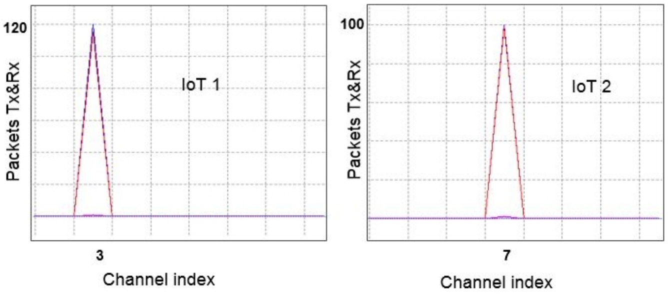


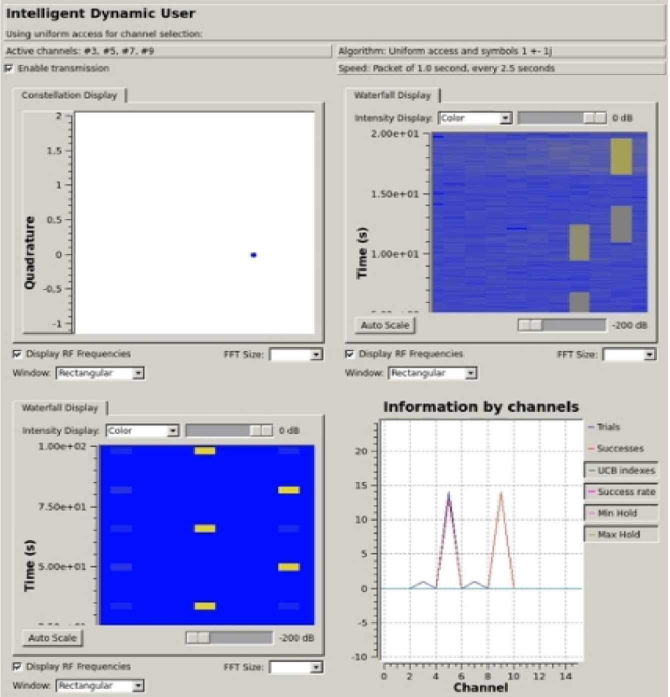
Fig. 8: Experimental setup composed of a gateway and two IoT

IV. CHALLENGES FOR PROTOTYPING IOT NETWORKS

The experimental results confirms the feasibility of introducing a RL technique. Nevertheless, at the light of this implementation and the experimental evaluations, several aspects are omitted, some of them are discussed in the following subsections.

---

[2] For the these experiments, a UCB approach [15] was implemented.



Intelligent Dynamic User

Using uniform access for channel selection:

Active channels: #3, #5, #7, #9

Algorithm: Uniform access and symbols 1 +- 1j

☑ Enable transmission

Speed: Packet of 1.0 second, every 2.5 seconds

Intensity Display: Color

Window: Rectangular

Fig. 9: GUI running at the IoT device. The collected statistics is shown in the box labeled "Information by channels", the spectrograms (in blue) show the transmitted and received packets (in yellow), and the constellation displays box plots the demodulated symbols.

A. PHY LAYER IMPLEMENTATION

The exposed prototype implements a PHY layer employing QPSK symbols and packet transmissions at a carrier frequency of 433.5 MHz, with roughly 0.5 seconds. It meets some specifications in time and frequency following a LoRa-like standard, and showing the feasibility of learning the available channel.

Nevertheless, a performance evaluation still requires a full implementation of the PHY layer. To address this gap, there exists implementations of the LoRa PHY layer [16], [17], [18] that allow us to assess the impact of fading and pathloss on the learning capability. In doing so, it is desirable to have a modular design for the integration of these approaches into a more general framework. More importantly, the reconfigurability of these designs is a relevant feature for real-time assessment. This is the case when setting the channel frequency for packet transmissions in the exposed prototype, as well as other parameters under study. Note that a physical layer can be running all the time to detect and transmit packets, while its reconfiguration will depend on other process such as RL agents, or upper layers.

For that end, the introduction of asynchronous mechanism is required to enable an online configuration. GNU radio, for instance, provides a message-based mechanism to incorporate this change in real time. In this way, it is expected that upper layers or control planes may interact with the PHY layer. It is shown in Fig. 10, where a RL agent interact with the configuration of the SDR board.

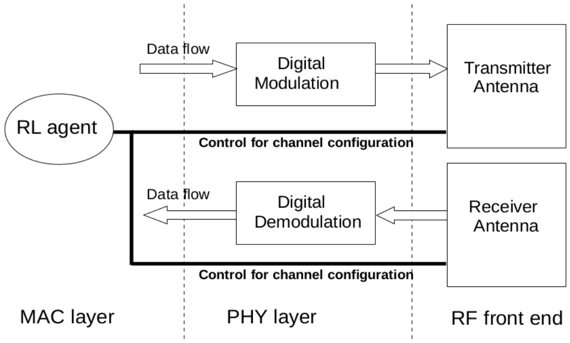


Fig. 10: Reconfiguration of RF front-end employing control signals.

Another solution adopted in the described implementation is the emulation of channel frequency change. It is carried out by generating waveforms at different sub-channel frequencies, as it is shown in Fig. 11. In this approach, the decision of generating a waveform at a given sub-channel frequencies emulates the selection of channel frequency. However, in this design, the aforementioned PHY layer modules can not be easily integrated, thus limiting to some proofs of concept and feasibility of novel approaches.

B. MAC PROTOCOL IMPLEMENTATION

The study of novel access schemes often make assumptions about the MAC protocol such as slotted ALOHA protocol [19] with random back-off time. However, the implementation of a slotted protocol requires to have a synchronization among all devices in the network, which turns out to be costly and may not be feasible.

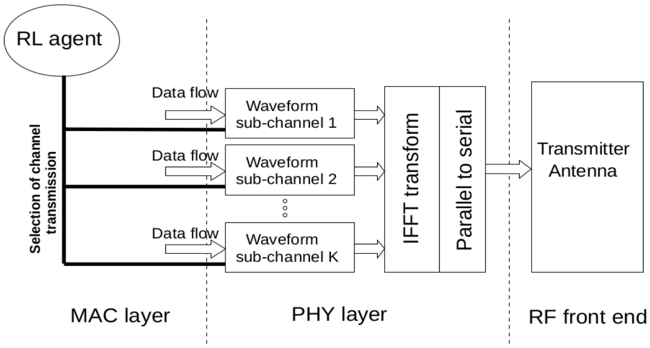


Fig. 11: Transmissions at different sub-channels, while RF front-end remains without change.

On the other hand, the prototype of these schemes assess them under non-slotted time (pure ALOHA) leading to experimental results that depart from those of a simulation model. In addition, SDR architectures where the board is controlled by a host computer introduces latencies. It makes hard to meet the timing of a MAC protocol, and embedded designs containing PHY, MAC, and application (APP) components are appealing, as it is summarized in Fig.12. These designs are more tailored to the needs of IoT endpoints [20], [17].

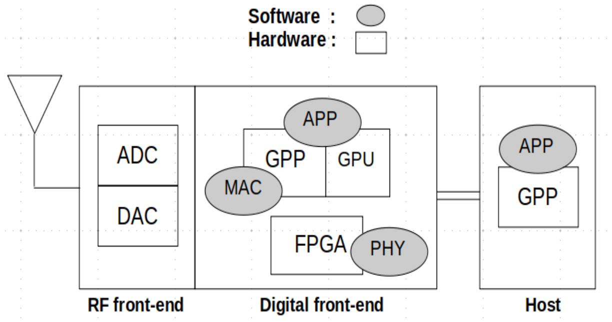


Fig. 12: Embedded implementations of PHY/MAC designs on the same SDR board. SDR hardware composed of a FPGA, general purpose processor (GPP), graphical processing unit (GPU).

In this regard, it is also important to highlight that modular designs for the PHY/MAC layer will ease the implementation of packet retransmission policies with more stringent back-off waiting times, and further work is needed.

C. Scalability

The prototyping is limited by the number of IoT devices, whenever the price for each of them and the deployment is not affordable. Although low-cost devices such as RTL-SDR [11] are nowadays more available at the market, their prototyping capabilities still require to be developed in comparison to other SDR boards [10]. More tailored boards [20], [17] are promising, however, the deployment of a large-scale LoRa networks is still hard to prototype.

One way for addressing this is to resort to emulations. The LoRA traffic received from several IoT devices at a gateway can be emulated employing a single device [21], [22]. In these works, the aim is to generate a waveform to be transmitted over the air, so that it can reproduce similar conditions provided by commercial Lora devices. For instance, in terms of signal-to-noise ratio (SNR) or signal-to-interference ratio (SIR) values. In the exposed implementation, an IoT device is also configured to transmit simultaneously several packets at different channels, and it can be improved by employing the procedures reported in the literature for practical interference scenarios.

D. Real scenarios

For purposes of a performance evaluation, the complexity can be increased significantly, in comparison to a proof-of-concept that simplifies several implementation aspects. For instance, the implementation of the uplink and the downlink (at the gateway side) requires to incorporate modules for the transmission *ACK* packets at different channels, the handling of packet collisions, the identification of IoT users, among other tasks.

The exposed prototyping introduces the implementation of uplink and downlink which is suitable for assessing the feasibility of a RL approach, where collisions and external interference are taken into account. On the other hand, it omits a full implementation of the PHY layer preventing to obtain more accurate performance evaluations. In fact, a recent study [18], shows that there exists a degradation of frame delivery ratio when simultaneous transmissions (uplink & downlink) occur. It is concluded after implementing and testing a LoRa PHY layer using GNU Radio and USRPs, thus the common assumption of orthogonality between these links is no longer valid.

The capture effect on the access protocol also requires to be examined [23], as it may have an impact on collisions, and consequently on the performance of the RL approach. The aforementioned aspects become more evident, when the recreated scenario considers a large number of IoT devices, and the policy for solving these events can impact the performance of the proposal.

Furthermore, the experimental evaluation of the coexistence with other networks introducing interference is also a subject to be addressed with the prototyping of these approaches in order to provide more reliable results.

V. CONCLUSIONS

The prototype of IoT networks aim to fill the gap between the theoretical and experimental results. In the present work, we have exposed several aspects concerning the validation of a RL approach, while highlighting its utility and impact for the development of LPWA networks. The missing aspects of current models are reviewed and the main challenges for prototyping them are discussed. In this regard, the obtained results report significant advantages in terms of successful packet transmissions showing the suitability of introducing RL approaches. The feasibility to emulate large-scale scenarios is also favorable following the different exposed SDR mechanism and guidelines discussed along with approaches found in the literature. Furthermore, more realistic features of PHY/MAC layer can be integrated within the described prototyping.

References

[1] U. Raza, P. Kulkarni, and M. Sooriyabandara, "Low power wide area networks: An overview," IEEE Communications Surveys Tutorials, vol. 19, pp. 855–873, Secondquarter 2017.

[2] C. Moy, A. Nafkha, and M. Naoues, "Reinforcement learning demonstrator for opportunistic spectrum access on real radio signals," in IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN), pp. 283–284, 2015.

[3] R. Bonnefoi, L. Besson, C. Moy, E. Kaufmann, and J. Palicot, "Multi-Armed Bandit Learning in IoT Networks: Learning helps even in non-stationary settings," in 12th EAI Conference on Cognitive Radio Oriented Wireless Network and Communication, CROWNCOM

Proceedings, 2017.

[4] R. Bonnefoi, C. Moy, and J. Palicot, "Improvement of the LP-WAN AMI backhaul's latency thanks to reinforcement learning algorithms, " EURASIP Journal on Wireless Communications and Networking, vol. 2018, no. 1, p. 34, 2018.

[5] R. Bonnefoi, L. Besson, J. Manco-Vasquez, and C. Moy, "Upper-confidence bound for channel selection in LPWA networks with retransmissions," in IEEE Wireless Communications and Networking Conference Workshop (WCNCW), pp. 1–7, 2019.

[6] "Ns3." https://www.nsnam.org/. Accessed: 2021-10.

[7] "OMNeT++." https://omnetpp.org/. Accessed: 2021-10.

[8] D.-T. Ta, K. Khawam, S. Lahoud, C. Adjih, and S. Martin, "LoRa-MAB: A Flexible Simulator for Decentralized Learning Resource Allocation in IoT Networks," in 12th IFIP Wireless and Mobile Networking Conference (WMNC), pp. 55–62, 2019.

[9] "Universal Software Radio Peripheral (USRP)." https://www.ettus.com/. Accessed: 2021-10.

[10] "ADALM-PLUTO." https://www.analog.com/en/design-center/evaluation-hardware-and-software/evaluation-boards-kits/adalm-pluto.html. Accessed: 2021-10.

[11] "RTL-SDR." https://www.rtl-sdr.com/. Accessed: 2021-10.

[12] "Design and prototype software-defined radio (SDR) systems using USRP with MATLAB and Simulink." https://www.mathworks.com/hardware-support/usrp.html. Accessed: 2021-10.

[13] National Instrument. Accessed: Apr. 2, 2022. [Online]. Available: https://www.ni.com/en-us/innovations/wireless/software-defined-radio.html.

[14] "GNU Radio." https://www.gnuradio.org/. Accessed: 2021-10.

[15] P. Auer, N. Cesa-Bianchi, and P. Fischer, "Finite-time Analysis of the Multi-armed Bandit Problem," Machine Learning, vol. 47, no. 2, pp. 235–256, 2002.

[16] J. Tapparel, O. Afisiadis, P. Mayoraz, A. Balatsoukas-Stimming, and A. Burg, "An Open-Source LoRa Physical Layer Prototype on GNU Radio," in IEEE 21st International Workshop on Signal Processing Advances in Wireless Communications (SPAWC), pp. 1–5, 2020.

[17] M. Xhonneux, J. Louveaux, and D. Bol, "Implementing a LoRa Software-Defined Radio on a General-Purpose ULP Microcontroller," in IEEE Workshop on Signal Processing Systems (SiPS), pp. 105–110, 2021.

[18] R. Saroui, A. Guitton, O. Iova, and F. Valois, "Uplink and downlink are not orthogonal in LoRaWAN!," in IEEE 96th Vehicular Technology Conference (VTC2022-Fall), pp. 1–4, 2022.

[19] N. Abramson, "The ALOHA System: Another Alternative for Computer Communications," in Proceedings of the November 17-19, 1970, Fall Joint Computer Conference, AFIPS '70 (Fall), (New York, NY, USA), pp. 281–285, ACM, 1970.

[20] M. Hessar, A. Najafi, V. Iyer, and S. Gollakota, "Demo: TinySDR, A Software-Defined Radio Platform for Internet of Things," in The 25th Annual International Conference on Mobile Computing and Networking, (New York, NY, USA), Association for Computing Machinery, 2019.

[21] A. Lavric, A. I. Petrariu, E. Coca, and V. Popa, "LoRa Traffic Generator Based on Software Defined Radio Technology for LoRa Modulation Orthogonality Analysis: Empirical and Experimental Evaluation," Sensors, vol. 20, no. 15, 2020.

[22] M. Shawki, Z. Ayman, K. Mahmoud, and T. Elshabrawy, "A LoRa Network Emulator Using Software Defined Radio," in IEEE 12th International Conference on Consumer Electronics (ICCE-Berlin), pp. 1–6, 2022.

[23] C. Pham and M. Ehsan, "Dense deployment of lora networks: Expectations and limits of channel activity detection and capture effect for radio channel access," Sensors, vol. 21, no. 3, 2021.