On the performance of IoT LPWAN technologies: the case of Sigfox, LoRaWAN and NB-IoT

Giovanni Stanco[1, 2], Alessio Botta[1], Flavio Frattini[2], Ugo Giordano[2], and Giorgio Ventre[1]

[1]*Department of Electrical Engineering and Information Technology, University of Napoli Federico II*, Napoli, Italy
[2]*RisLab – Laboratorio di Ricerca e Innovazione per la Sicurezza S.r.l.*, Napoli, Italy
{*giovanni.stanco, a.botta, giorgio.ventre*}*@unina.it, flavio.frattini@rislab.it, ugogio87@gmail.com*

ICC 2022 - IEEE International Conference on Communications | 978-1-5386-8347-7/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICC45855.2022.9839078

***Abstract*—Internet of Things is a widespread technology that comprises several networking solutions for connecting things to the rest of the Internet. Understanding the characteristics of such solutions is fundamental in order to satisfy the diverse requirements of all the possible applications. The goal of this paper is to empirically evaluate and compare the performance of the most spread technologies for IoT low-power long-range communications. The parameters considered are the energy efficiency, the message losses, and the latency of a message. Obtained results show that up to 2% of messages can be lost when using LoRaWAN, but the latency is always smaller than 7 seconds. NB-IoT shows slightly larger latency values and more delivered messages than LoRaWAN. The messages sent using Sigfox are always correctly delivered, but the communication introduces delays up to 100 seconds. These results are of great importance for all the players in the IoT scenario interested in LPWAN technologies. The results of this study show how different communication technologies provide significantly different performance.**

***Index Terms*—Internet of Things, Sigfox, LoRaWAN, NB-IoT**

I. Introduction

The Internet of Things (IoT) is nowadays deployed in a wide variety of scenarios and applications, with the goal of realizing a network of physical objects connected to the Internet able to perceive and possibly modify the environment they are in. IoT wireless technologies are regarded with great interest by all the players willing to upgrade the currently used wired sensors to exploit the potential benefits provided by wireless sensors: easier deployment, more mobility, and less weight, due to the absence of wires and conductors. The variety of IoT networks is currently wide and several communication technologies with different characteristics and for different purposes are available. Among these technologies, some are suitable for delay-sensitive applications while others provide low throughput communication. Low Power Wide Area Networks (LPWAN) such as Sigfox, LoRaWAN and NarrowBand-IoT (NB-IoT) are gaining attention thanks to their ability to provide long range connectivity for battery-operated devices.

The heterogeneity of the solutions currently available is one of the obstacles to the massive deployment of IoT. IoT devices are in fact produced by different manufacturers, are based on different hardware and operating systems, and use different technologies to communicate. The variety of solutions currently available entails that it is necessary to clearly assess the characteristics of the employed technologies in order to know in advance what we should expect and whether the requirements of the services to be offered are satisfied. This aspect has been studied in the recent literature, but important performance parameters have not been properly considered.

The goal of this paper is to describe the experiments carried out using an IoT development board and multiple communication technologies, also considering different performance parameters. Most works in literature use several development boards and network technologies to propose a solution for different applications, while only a few others are actually focused on the performance of different network technologies. With respect to these works, our study considers for the first time in literature also the time necessary to deliver a message to a final destination. Our tests allow the evaluation of performance parameters that show how the three network technologies tested have different pros and cons, which should be considered when developing and using an IoT application. Obtained results show that LoRaWAN has the smallest average and maximum latency measured (2.4 and 6.8 seconds respectively) but a fraction of 2% of the sent messages is lost in our tests. NB-IoT does not lose messages in multiple repetitions and has the minimum latency value measured (1.6 seconds). However the number of messages sent with a fully charged battery (200) is much smaller than the number of messages sent via LoRaWAN in the same condition (375). Sigfox is the best choice among the three technologies tested in terms of energy efficiency (395 sent messages) and message losses (always 0%), but the latency reaches peak values of 100 seconds that are not measured with the other technologies. This proves even more how important it is to evaluate the performance of the multiple possibilities of IoT.

The rest of the paper is organized as follows. In Section II we review related works. We report some features of the devices, communication technologies and platforms employed in Section III. Section IV describes the tests executed, whose results are reported in Section V. Section VI ends this work.

II. Related works

Several studies about IoT devices and long-range network technologies have been reported in literature. Several groups of researchers have devised solutions for their IoT applications, showing which of the several available devices and communication technologies satisfied their needs. Examples of temperature and environment monitoring stations using different boards (e.g. Arduino, Pycom) and different network technologies (e.g. NB-IoT, LoRaWAN) are reported in [1], [2] and [3]. Ugwuanyi et al. [4] instead present a practical deployment of a typical industry NB-IoT network. These works do not include experimental results to evaluate IoT network performance. Other works focus on the assessment of the performance of communication technologies on different hardware, evaluating several metrics. The authors of [5] use a multichannel LoRa gateway and several LoRa end-nodes to evaluate packet losses, data rates and communication range in an indoor scenario. Prasad et al. [6] compare the performance of LoRa and NB-IoT operating tests in real-world scenarios in terms of signal strength and packet delivery ratio. These two metrics are also considered by Oliviera et al. [7] when comparing their LoRa and Sigfox prototypes. Zhou et al. [8] studied the positioning services provided by Sigfox Geolocation and GPS in terms of precision and power consumption using Pycom boards.

Other works evaluate the performance of real devices in particular scenarios. Wang et al. [9] study LPWAN performance in the case of drones moving at high speeds in the air. They evaluate latency and loss rate of NB-IoT and Sigfox, which are obviously impacted by the high speed of the end-devices and the gateways. Their analysis on the latency is based on the measurement of the arrival of a message to a database server on the Internet in the case of NB-IoT and to the Sigfox gateway in the other case. Perkovic et al. [10] compare Sigfox, NB-IoT and LoRaWAN using three boards based on Arduino and the most popular sensor devices used in a parking context, such as ultrasound and infrared. They classify the sensors according to their power consumption and detection accuracy. Our work focuses on the assessment of performance of widespread LPWAN technologies like previous works have done, but we also evaluate a metric that no previous work has analysed in depth: the latency of a message to its final destination. Perkovic et al. [10] have conducted experiments with the same goal, but they do not include LoRaWAN and do not consider the arrival of a Sigfox message to a final destination. Moreover their work evaluate performance in the case of devices moving at high speed, which is a factor that can easily impact these results. In our tests we do not move our hardware, recreating the situation of sensors employed for monitoring in a fixed position.

III. Description of devices and technologies

A. Pycom devices

The devices employed in our tests are two Pycom development boards, namely two FiPy boards with two Pysense sensor boards. A Pycom FiPy is a programmable board that features WiFi 802.11 b/g/n, Bluetooth (LE and classic), LoRa, Sigfox and dual LTE-M (CAT-M1 and NB-IoT) [11]. It runs on an Espressif ESP32 chipset and is provided with 4 MB of RAM and 8 MB of flash memory. The FiPy board is equipped with a firmware based on Micropython, an optimised implementation of the Python 3 language [12]. Pycom boards support standard Python libraries, Micropython-specific libraries and modules specific to the Pycom devices [13].

B. Sigfox

Sigfox base stations are deployed like a cellular network and are connected to the backend servers using an IP-based network [14]. Sigfox devices broadcast radio messages which are picked by these base stations and conveyed to the Sigfox Core Network [15]. Messages are transmitted over different subcarriers in three consecutive radio frames to multiple base stations, in order to make transmissions robust [16]. Communications exploit sub-GHz ISM band carriers and Ultra Narrow Band (UNB) modulation types. Sigfox offers low duty cycles to conform to the regulations on use of license free spectrum: only 140 12-byte messages per day are allowed over uplink, while 4 8-bytes messages per day are permitted over downlink. Every Sigfox Pycom device needs to be registered on the Sigfox backend, known as **Sigfox Cloud**. The data sent from the devices are received by the Sigfox Cloud, which forwards them using callback services for delivering information to external IoT platforms [17]. The callback used in our tests is a Data Callback and the body of the callback message contains the device ID, the sequence number, the timestamp of the Sigfox event and the data received from the device.

C. LoRaWAN

LoRa® is a physical layer technology that modulates the signals in Sub-GHz ISM band using a proprietary spread spectrum technique [18]. Multiple frames can be exchanged at the same time using different spreading factors [19]. Commercial and industrial partners proposed LoRaWAN, an architecture consisting of layers above the LoRa physical layer. LoRaWAN uses star architecture in which gateways are used to relay messages between end-devices and a central core network. Messages sent by end-devices are received by all the reachable base stations, which are connected to the core network services that handle the packets via backhaul IP connections [20].

We use a FiPy as a node and another FiPy as a **nano-gateway** in the tests with LoRaWAN. The nano-gateway is connected to a Wi-Fi hotspot to reach **The Things Network** (TTN), a decentralized infrastructure that provides a free of charge global LoRaWAN network to build IoT applications [21]. TTN has its own fair access policy: 30 seconds per day of uplink time per device and 10 downlink messages per day. In order to use TTN, we must register our devices to its website, either as a node or as a nano-gateway. An HTTP integration allows to send uplink data to an endpoint and receive downlink data.

D. NB-IoT

NB-IoT is a cellular technology aimed at handling a massive number of connected devices [22]. NB-IoT is based on the LTE protocol and works on the 700, 800 and 900 MHz bandwidths [23]. LTE functionalities are reduced while the functionalities required by IoT applications such as low power

Request

```
POST /api/narrowband/add HTTP/1.1
Host: 1.2.3.4:9898
Content-Type: application/json
Accept: application/json

Message Payload: {"devId":"DEVICE_NAME","data":"MESSAGE_PAYLOAD"}
```

Fig. 1: Example of callback used with NB-IoT.

consumption and spectrum efficiency are enhanced. Since it operates on licensed bands, NB-IoT does not suffer from restrictions on its duty cycle or interference.

The FiPy, equipped with a SIM card, is allowed to reach **Pybytes**, the free cloud-based management platform available for all Pycom development boards. The devices connect to Pybytes over a shared Access Point Name (APN) by default, but it is also possible to opt for a private APN [24]. The APN employed is *pycom.io* while the band employed is Band 20 (uplink band: 832 - 862 MHz, downlink band: 791 - 821 MHz, bandwidth: 30 MHz). The device connects to the broker *mqtt.pybytes.pycom.io* and sends the data as a string or as an integer using a special Pycom function. Pybytes offers a way for interacting with external IoT platforms or services via Web Hooks integrations, that allow to define HTTP callbacks [13]. The platform performs an HTTP request defined by the user every time it receives a message from a registered device. An example of callback is reported in Fig. 1: as shown it is possible to set the remote URL to which the data will be sent, the HTTP method (POST, PUT, etc.), the request format (JSON, custom, etc.) and the body of the message forwarded to the remote destination. In our tests, the remote destination is our backend server and the data exchanged is the message payload received by Pybytes, sent in a JSON format.

IV. Tests

The main goal of this work is to evaluate performance parameters of the IoT technologies described in the previous section. A correct assessment of these parameters is crucial to choose the most suitable communication technology for an application. Some of the main metrics that describe the qualities of an IoT system are the energy efficiency (low consumption also means low cost), the percentage of message losses, and the latency of a message (aiming at the shortest time to deliver a message to its final destination). Another parameter evaluated in our tests is the accuracy of the clock on the device. All the metrics considered are in Table I. Our testbed is made of FiPys connected to a sensing board and communicating with the appropriate gateway or base station. The testbed is represented in Fig. 2. The performance analysis of the devices and the network technologies is carried out by sending a message from the device containing the same information: a timestamp, information on the battery, and data from the temperature, the humidity and the light sensors. The timestamp and the information on the battery are sent because they are fundamental to our analysis, the other data are chosen among the data available from the Pysense sensor board

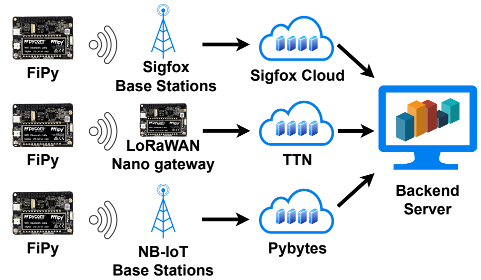


Fig. 2: Testbed employed in our tests.

which allow to maximize the information carried by a Sigfox payload, whose maximum size is limited to 12 bytes. The payload sent over LoRaWAN is identical, while the payload sent over NB-IoT has a different format. The sent messages go through the different IoT networks and platforms and are finally delivered to our backend server, where their payload is saved along with the timestamp of its arrival. The data useful for the performance analysis are retrieved from the device log files and the messages received from our backend. The energy efficiency is evaluated considering the number, reported in the log files, of the messages sent with a fully charged battery. The message losses are calculated by comparing the number of sent messages reported in the log files and the number of received messages at our backend. The latency is evaluated by subtracting the sent timestamp to the timestamp of the arrival at our backend, while we evaluate the accuracy of the clock on the device by considering the mentioned timestamps.

The scripts employed in our tests use functions provided by the Pycom documentation. The executed operations are: awakening of the device, connection to a Wi-Fi network to synchronize to an NTP server, retrieval of the data from the sensors, creation of a socket/connection and sending of the message, deinitialization of the LTE modem and entrance in sleep mode in order to save power.

V. Results

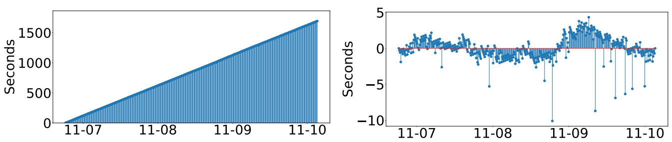
We report the results of our tests in terms of the previously mentioned metrics (see also Table I).

A. Clock accuracy

The first consideration we can make from our results regards the accuracy of the internal clock of the device. In the first tests carried out, the device connects to a NTP server only the first time it wakes up and then sends messages using the Sigfox network. We notice how the three timestamps (timestamp sent, timestamp of the event at the Sigfox Cloud, and timestamp of the arrival of the message at our backend) of the first message are correctly ordered. From the second message onward however, the timestamp sent from the device is subsequent to the other two timestamps. This behaviour is observed with both the devices and it is valid for all the communication technologies. The explanation for this effect is that in our system there are different clocks running at different speeds on different machines: the Pycom device, the

TABLE I: Performance metrics.

Quantity	Methodology
Clock accuracy	Difference of the timestamp at our backend server and the non-updated timestamp sent by the device
Energy efficiency	Number of messages sent with a fully charged battery
Message losses	Comparison of number of sent and received messages
Latency	Difference of the timestamp at our backend server and the updated timestamp sent by the device



(a) Drift. **(b)** Skew.

Fig. 3: Drift and skew using Sigfox.

Sigfox Cloud and our backend server. In the plots of Fig. 3a we can notice how the differences between the timestamps change during the tests. This graph is obtained from one of the first tests using Sigfox on the first device, but we observe the same behaviour with the other device and other network technologies as well. In this plot we report the difference between the timestamps sent and the ones at our backend. This difference is called drift. We can notice that these differences grow according to a linear relation. The slope of the line that interpolates the first and the last points of the plot in Fig. 3a is the ratio of the difference between the drift at the last and at the first moments and the time elapsed between the two moments considered: the slope is then the average rate at which the two clocks diverge. The slope calculated is $5.816 * 10^{-3}$: this means that the difference between the two clocks varies by 5.8 ms every second. If we consider the whole duration of the experiment (around 3 days and 9 hours, that is 291600 seconds), we can calculate that the estimated difference between the two clocks is 1691 seconds, that is close to the value actually measured.

The plot of Fig. 3a shows the average difference of the two speeds, while the plot of Fig. 3b shows the instantaneous difference of the two clocks. This instantaneous difference is called skew and it is calculated by subtracting the value of the ideal line from the actual difference between the two timestamps. This instantaneous variation is caused by different transmission and elaboration times. The plot of Fig. 3b shows that the skew can assume positive or negative values. If we consider that the messages are sent every 12 minutes, we can say that the average misalignment between the two clocks every 12 minutes is 4.2 ± 10.1 seconds, where 4.2 is the average misalignment in 12 minutes $(12*60*5.816*10^-3 = 4.2)$ and 10.1 is the absolute value of the maximum skew measured (as shown in Fig. 3b). **These results show that the two clocks distance themselves of** 5.8 ± 0.01 **milliseconds per second**, where 0.01 is the maximum skew in a second $(10.1/(12*60))$. Drift and skew can be avoided by connecting to a NTP server every time the device wakes up: in this case we do not observe strange trends and the timestamps are correctly ordered.

B. Energy efficiency

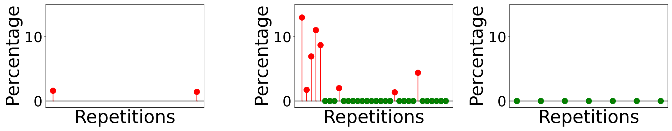
Sigfox and LoRaWAN have similar performance, with Sigfox being the most efficient with around 395 sent messages in different repetitions and LoRaWAN the second most efficient with around 375 messages. Several setbacks are faced when carrying out the same experiment using NB-IoT. In this case, the Pycom device becomes unstable and is not able to exploit completely the battery because it gets stuck during its operations. After numerous repetitions, it becomes evident that the operation that causes the interruption of the normal operations is the connection to the NB-IoT network. The device uses some proprietary functions to connect to the network: one of these is *pybytes.connect()*, which is not always correctly executed. When this instruction is not correctly carried out, the device can behave in two different ways: 1) it hangs when trying to connect to the network, 2) it is restarted by an internal watchdog after 21 minutes of inactivity. However this second possibility does not guarantee that the device resumes its normal operation: it is actually very likely that the *pybytes.connect()* operation will remain unstable until it is manually powered down. Our analysis also highlights that when the device hangs during the connection to the network, the battery is quickly drained. The device has a stable behaviour in only one of the many repetitions performed, and the number of messages sent is 200, which is lower than the number of messages sent using the other two technologies. **The results show that Sigfox has the lowest consumption and that NB-IoT has the worst performance.**

C. Message losses

The message loss ratio is 0% in all of our tests with Sigfox, while we always record a ratio lower than 2% when using LoRaWAN. As stated in Subsection V-B, the device does not show a stable behaviour when using NB-IoT. After several repeated experiments to understand how to make it work better, the performance shows an improvement. The message loss ratio is around 10% in the first tests, but it then decreases to 0% for the 88% of the repetitions. The results in Fig. 4 show that **Sigfox is the best communication option in terms of messages correctly delivered**.

D. Time analysis

We report the CDF of all the latencies measured in all the experiments in Fig. 5 while significant values are reported in Table II. LoRaWAN presents the lowest average value, 90th



(a) LoRaWAN **(b)** NB-IoT **(c)** Sigfox

Fig. 4: Percentage of lost messages.

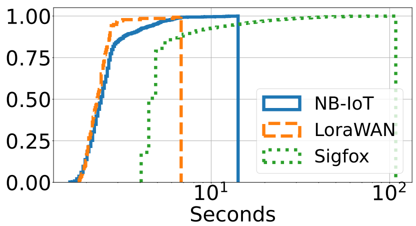


Fig. 5: CDF of the message latencies.

percentile, and maximum value (respectively 2.4, 2.7, and 6.8 seconds). NB-IoT shows the lowest minimum latency (1.6 seconds), while the worst results are obtained when using Sigfox. In this latter case, the 90th percentile is around 8 seconds while the maximum value exceeds 100 seconds, a value that is not measured with other network technologies. If we consider the maximum values measured and the error caused by the clock inaccuracy, we have maximum latencies of $6.79 \pm 0.04s$ for LoRaWAN, $14.15 \pm 0.08s$ for NB-IoT and $108.26 \pm 0.62s$ for Sigfox. These results show that **the values measured using Sigfox are higher than the ones measured with the other two technologies, which have smaller and less variable latencies.**

Another interesting parameter to evaluate is the **duration of the transmission of a message**. The average transmission of a message with Sigfox shows a value of around 11.5 seconds for all the repetitions of our tests. Moreover a peak value is repeated every 47 messages in every repetition. The transmission of a message is way shorter for LoRaWAN and NB-IoT: with the former the average is around 680 microseconds while for the latter the average is around 24 milliseconds. The values retrieved from one of the tests for every technology are reported in Fig. 6.

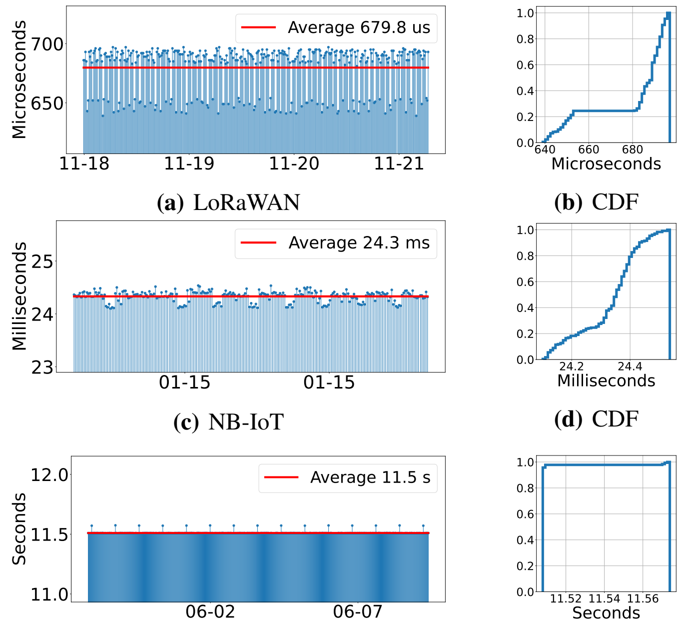
E. Understanding Sigfox latency

It is possible to perform a more detailed analysis of the latency of a message when using Sigfox. In this case we can retrieve the timestamp of the arrival of a message on the Sigfox Cloud, so it is possible to know the time elapsed between the sending of a message and its corresponding Sigfox event and the Sigfox event and the reception of a message on our backend. The results obtained with both the devices, in repeated experiments, in different conditions, are very similar and the outcome of one of our experiments is reported in Fig. 7. The wireless channel between the device and the gateway seems to have a stable behaviour: the majority of the ts_sigfox-ts_sent intervals are indeed included in the (2; 4) seconds intervals, and only in very rare cases there are outliers (Fig. 7a). The communication between the Sigfox Cloud and our backend introduces large latency and variability (Fig. 7c). For the test reported in Fig. 7, the minimum value of the difference between the timestamp at our backend and the timestamp at the Sigfox Cloud is 0.48 seconds, the maximum is 101.26 seconds, while the 90th percentile is 10 seconds. From multiple experiments, it is possible to notice how these outliers do not follow a particular trend and their distribution is random. **The variability introduced between the Sigfox Cloud and the final destination may severely impact communications in critical applications.**

In the Sigfox case, only the time measured between the device and the Sigfox Cloud is subject to the error due to the inaccuracy of the clock, while we can assume that the two clocks on the Sigfox Cloud and our backend server are more accurate. In this case, the maximum is 11 ± 0.06 seconds.

TABLE II: Analysis of the latency for the three technologies.

	Min	Avg	90th perc.	Max
LoRaWAN	1.8 s	2.4 s	2.7 s	6.8 s
NB-IoT	1.6 s	2.7 s	3.7 s	14.1 s
Sigfox	4.1 s	6.3 s	8.0 s	108.3 s

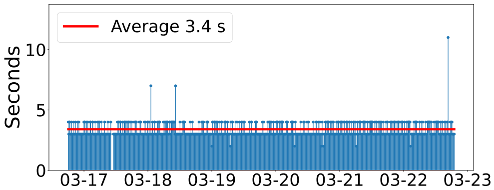


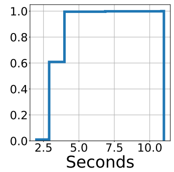
(a) LoRaWAN **(b)** CDF **(c)** NB-IoT **(d)** CDF **(e)** Sigfox **(f)** CDF

Fig. 6: Duration of the transmission of a message.

VI. Conclusion

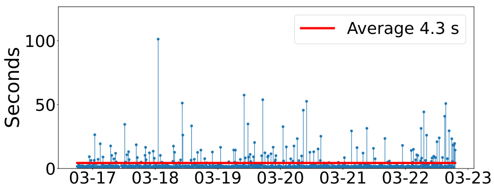
In this work we analyzed the most spread LPWAN technologies currently available for IoT. The existence of several network possibilities entails the need to understand which one can meet the requirements of the diverse possible services. We compared the performance of Sigfox, LoRaWAN and NB-IoT in terms of energy efficiency, number of messages

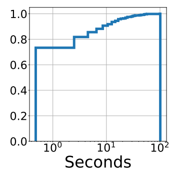




(a) Difference: timestamp Sigfox event - timestamp sent.

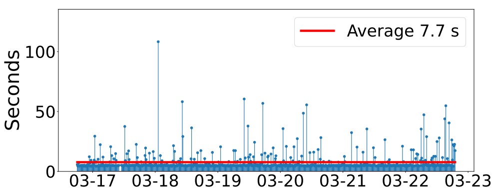
(b) CDF

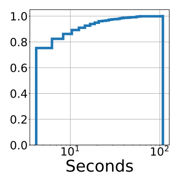




(c) Difference: timestamp server - timestamp Sigfox event.

(d) CDF





(e) Difference: timestamp server - timestamp sent.

(f) CDF

Fig. 7: Latencies in an experiment using Sigfox.

correctly delivered, and latency of a message, using a single development board. We showed that Sigfox correctly delivers every sent message, however our results highlight the delay introduced by the Sigfox Cloud, which could be a not negligible setback. NB-IoT offers increased performance in terms of latency and acceptable results in terms of packet losses. On the other hand, the several issues faced with this technology on this device are definitely not negligible. LoRaWAN is the technology that presents the best results in terms of average and maximum latency.

Based on the results of our experimentations, we can affirm that there seems to be no suitable option for applications for which every message must be correctly delivered in a short time (e.g. a security monitoring application in which alerts are collected to report a danger). Sigfox, with its successful deliveries, would be the ideal solution for an application such as smart meters, in which all the data are necessary for billing purposes, but a small latency is not essential. In other non-critical applications, such as monitoring of lighting and heating systems, the loss of a small fraction of messages would impact the potential benefit, but it would be tolerable.

Acknowledgment

This work was partially supported by MIUR through the "ICT for Health" project, Dipartimento di Eccellenza (2018-2022) "Ingegneria Elettrica e delle Tecnologie dell'Informazione".

References

[1] A. Voisin-Grall, O. O. Malaolu, Y. Zhu, T. Ahmed, S. A. Al-Ahmed, and M. Z. Shakir, "Remote condition monitoring: a prototype based on Pycom development board FiPy and Pysense," in *2019 UK/China Emerging Technologies (UCET)*, 2019, pp. 1–6.

[2] K. Kaewwongsri and K. Silanon, "Design and implement of a weather monitoring station using CoAP on NB-IoT network," in *2020 17th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*, 2020.

[3] M. Herlich and F. von Tüllenburg, "Introduction to narrowband communication," in *Proceedings of the Wireless Congress: Systems and Applications*, 2018.

[4] S. Ugwuanyi, J. Hansawangkit, and J. Irvine, "NB-IoT testbed for industrial Internet of Things," in *2020 International Symposium on Networks, Computers and Communications (ISNCC)*, 2020, pp. 1–6.

[5] M. I. Muzammir, H. Z. Abidin, S. A. C. Abdullah, and F. H. K. Zaman, "Performance analysis of LoRaWAN for indoor application," in *2019 IEEE 9th Symposium on Computer Applications & Industrial Electronics (ISCAIE)*. IEEE, 2019, pp. 156–159.

[6] P. Yogendra Prasad, H. Heera, N. Padmaja, and B. Sony, "A comparative analysis of long range and NB-IoT in terms of quality of connectivity," *Materials Today: Proceedings*, 2021.

[7] L. Oliveira, J. J. Rodrigues, S. A. Kozlov, R. A. Rabêlo, and V. Furtado, "Performance assessment of long-range and Sigfox protocols with mobility support," *International Journal of Communication Systems*, vol. 32, no. 13, p. e3956, 2019.

[8] X. Zhou and E. Flora, "Comparison of performance and power consumption between GPS and Sigfox positioning using Pycom modules," 2018.

[9] S.-Y. Wang, J.-E. Chang, H. Fan, and Y.-H. Sun, "Performance comparisons of NB-IoT, LTE Cat-M1, Sigfox, and LoRa moving at high speeds in the air," in *2020 IEEE Symposium on Computers and Communications (ISCC)*, 2020, pp. 1–6.

[10] T. Perković, P. Šolić, H. Zargariasl, D. Čoko, and J. J. Rodrigues, "Smart parking sensors: state of the art and performance evaluation," *Journal of Cleaner Production*, vol. 262, p. 121181, 2020.

[11] FiPy. Accessed on: Sept. 17, 2021. [Online]. Available: https://pycom.io/product/fipy/

[12] MicroPython. Accessed on: Sept. 22, 2021. [Online]. Available: http://micropython.org/

[13] Pycom documentation. Accessed on: Sept. 22, 2021. [Online]. Available: https://docs.pycom.io/

[14] U. Raza, P. Kulkarni, and M. Sooriyabandara, "Low Power Wide Area Networks: an overview," *IEEE Communications Surveys Tutorials*, vol. 19, no. 2, pp. 855–873, 2017.

[15] Sigfox technology. Accessed on: Sept. 17, 2021. [Online]. Available: https://www.sigfox.com/en/what-sigfox/technology

[16] G. Margelis, R. Piechocki, D. Kaleshi, and P. Thomas, "Low throughput networks for the IoT: Lessons learned from industrial implementations," in *2015 IEEE 2nd World Forum on Internet of Things*, 2015.

[17] Sigfox Cloud integration. Accessed on: Sept. 17, 2021. [Online]. Available: https://build.sigfox.com/backend-callbacks-and-api

[18] G. Ferré and E. P. Simon, "An introduction to Sigfox and LoRa PHY and MAC layers," 2018.

[19] K. Mekki, E. Bajic, F. Chaxel, and F. Meyer, "A comparative study of LPWAN technologies for large-scale IoT deployment," *ICT express*, vol. 5, no. 1, pp. 1–7, 2019.

[20] K. E. Nolan, W. Guibene, and M. Y. Kelly, "An evaluation of Low Power Wide Area Network technologies for the Internet of Things," in *2016 International Wireless Communications and Mobile Computing Conference (IWCMC)*, 09 2016, pp. 439–444.

[21] The Things Network - LoRaWAN documentation. Accessed on: Sept. 22, 2021. [Online]. Available: https://www.thethingsnetwork.org/docs/

[22] K. Nair, A. Abu-Mahfouz, and S. Lefophane, "Analysis of the Narrow Band Internet of Things (NB-IoT) technology," in *2019 Conference on Information Communications Technology and Society*, 2019.

[23] J. Xu, J. Yao, L. Wang, Z. Ming, K. Wu, and L. Chen, "Narrowband Internet of Things: Evolutions, technologies, and open issues," *IEEE Internet of Things Journal*, vol. 5, no. 3, pp. 1449–1462, 2018.

[24] FAQ for Pycom's Vodafone NB-IoT sims and airtime provision. Accessed on: Sept. 17, 2021. [Online]. Available: https://pycom.io/services/cellular-services/faq-for-vodafone-pycoms-nb-iot-services/