Towards an Application-Based Methodology for IoT Network Technology Performance Evaluation

Samir Si-Mohammed[1,2], Thomas Begin[1], Isabelle Guérin Lassous[1], and Pascale Vicat-Blanc[2]

[1]Univ Lyon, ENSL, UCBL, CNRS, Inria, LIP, F-69342, LYON Cedex 07, France
[2]Stackeo, Lyon, France
[1]{name.surname}@ens-lyon.fr
[2]firstname@stackeo.io

***Abstract*—Internet of Things (IoT) covers a large range of applications such as telemetry, remote control, video-surveillance, and monitoring. Those applications rely on a communication technology to exchange data between the different elements of the IoT system. The IoT communication technologies are multiple and come with different features in terms of range, throughput, latency, scalability, energy, etc. It is complex, yet critical, for an IoT architect to select the most adequate technology for a targeted application. In this paper, we propose a methodology to estimate the performance of an IoT communication technology for typical IoT applications. The proposed methodology is based on four steps that lead to a systematic evaluation. We illustrate the methodology with two IoT communication technologies (Wi-Fi and LoRaWAN) on a set of typical applications using the NS-3 simulator.**

***Index Terms*—Internet Of Things, Applications, Methodology, Performance Evaluation, Simulation.**

I. Introduction

Internet of Things is commonly described as the connection to the Internet of a whole range of end-devices that directly interact with the physical world. These connections enable the transport of real-time industrial machine data or the execution of remote commands to improve operational efficiency. The connected IoT devices have often less memory, less processing power and less bandwidth than traditional Internet devices. The Internet of Things forms a complex eco-system for exchanging a large spectrum of information. The data can flow (i) from sensors to data center servers through gateways, where collected data are processed to enable informed decisions (ii) from the cloud or edge servers down to actuators or more generally end-devices to transmit information or commands, etc. IoT has recently seen a tremendous expansion in terms of the range of services it can support such as tracking, telemetry, condition monitoring, alerting, security in many domains like precision agriculture, smart health, predictive maintenance, industry 4.0, etc.

To handle the various traffic of such diverse IoT applications and contexts, connectivity technologies have evolved. The recent years have seen an important growth in the number of available IoT network technologies, such as Sigfox [1], LoRaWAN [2], NB-IoT [3], LTE-M [4] for long range low power communications or BLE [5], Zigbee [6] and Wi-Fi [7] for short-medium range communications. Each technology has its own specifications, protocols, advantages and drawbacks. Moreover, they are in constant evolution. For example the 5G [8] cellular technology getting outstanding attention for its usage in IoT, will be challenged by 6G [9] in the future. The heterogeneity and dynamics of these technologies make users and architects increasingly confused, which tends to be counterproductive and slowing down the IoT adoption. The choice of an IoT network and its configuration for a given application can be, indeed, very challenging. But making the right choice is critical, as under- or over-sizing is not optional when cost and performance are highly constrained. The decision often consists in finding the best trade-off between cost, range, bandwidth and energy consumption for a given deployment and application.

One way of making the appropriate choice is to explore the technologies via IoT surveys. A recent list of them is provided in [10]. These papers offer an overview about the existing IoT technologies and the differences between them, but they can be either incomplete or become quickly outdated, due to the continuous evolution of the technologies. There is a constant need for a fresh and up-to-date view in such a prolific domain.

In addition, compared to other communities like imagery with its benchmarks, the IoT community lacks a reproducible and scalable approach to assess the performance of an IoT network technology for a given scenario. This often leads to misunderstanding between stakeholders. We believe that providing a structured and systematic approach to evaluate a network technology in different applications would help the users to succeed in their IoT journey. This led us to propose a methodology to estimate the performance of an IoT network technology in well defined application contexts.

Although many studies are focused on the evaluation of IoT network technologies, there is a relative paucity for developing a reproducible and robust methodology. In most articles, the considered scenarios for evaluation are either dedicated to the used communication technology or to a targeted application. For instance, in [11], the authors analyze the performance of LoRaWAN on four use-cases using the LoRaSim simulator with a modified MAC protocol. The authors of [12] use simulation to compare the reliability of LoRaWAN, Sigfox and NB-IoT for the very specific Smart Water Grid Scenario. Another example is [13] wherein the authors compare the network performance of Wi-Fi and Cellular technologies in terms of throughput and latency in metro areas.

The remainder of the paper is organized as follows: Section II gives a description of the proposed methodology, with its goals, guiding principles and key components. The evaluation process and the results are detailed in Section III. Finally, a discussion and a conclusion are developed in Section IV.

II. Methodology

Our application-based IoT network performance evaluation methodology aims at systematically analyzing the behavior of an IoT application's traffic within an IoT network on a set of well defined IoT-relevant metrics. The methodology should allow: (i) users and researchers to rapidly evaluate the applicability of a technology to an IoT scenario (ii) network technology or service providers to present the effective key performance indicators (KPI) of their product/service in a standardized framework to accelerate comparison, qualification and adoption. This methodology can be beneficial to both the research and the industrial communities.

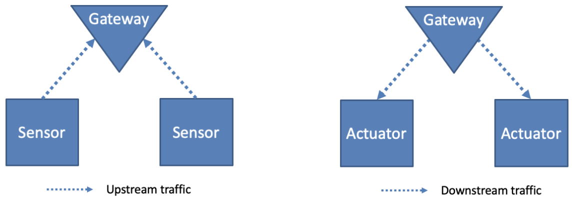
The methodology consists in four components: (i) a set of IoT traffic types, (ii) a set of metrics to evaluate the performance of an IoT network technology, (iii) an evaluation tool and (iv) a framework to specify a scenario with a) a selected network technology, b) low-level configuration parameters, c) application topology definition, d) workload parameters to conduct the systematic evaluation.

A. *Traffic types*

We propose to classify the IoT traffic types by (i) their direction: upstream (from end-devices to gateways or the cloud) or downstream (from the cloud or gateways to end-devices) and (ii) their profile: periodic or stochastic (sporadic, bursty...). The periodic traffic corresponds to a fixed data rate, while the stochastic traffic has a variable rate. Although some applications have bidirectional traffic, we observe that a majority of IoT applications have unidirectional flows.

1) IoT traffic type 1: IoT traffic type 1 corresponds to a periodic upstream traffic. One popular example is telemetry, where measurements are periodically collected from sensors and transmitted upstream for analysis purposes. Meteorology, precision agriculture, smart building, tracking applications exhibit such traffic pattern. The traffic period may differ depending on the application from few seconds up to one message per day. This traffic can be latency-sensitive. Figure 1a shows the typical star topology of this workload.

2) IoT traffic type 2: Periodic downstream traffic is represented by this type. Notifications is one application of it, which is used when a solution provider wants to transmit regularly some information to all the end-devices simultaneously. This traffic can be also latency-sensitive. The topology is shown in Figure 1b.



(a) Traffic types 1 and 3.

(b) Traffic types 2 and 4.

Figure 1: Examples of traffic types.

Traffic type	Traffic profile	Traffic direction	Examples
1	Periodic	Upstream	Telemetry, Remote control...
2	Periodic	Downstream	Notifications, Alert...
3	Stochastic	Upstream	Video-Surveillance, Cloud gaming...
4	Stochastic	Downstream	Webcast, Virtual Reality...
5	NA	Upstream	Overload
6	NA	Downstream	Overload

Table I: Traffic types characteristics.

3) IoT traffic type 3: Unlike the previous types, IoT traffic type 3 represents stochastic traffic, in an upstream direction. Videos taken by cameras placed at specific positions and transmitted for real-time image analysis in video-surveillance applications is an example of IoT traffic type 3. Video-surveillance is largely used today to avoid intrusions, thefts, and other kinds of physical threats.

4) IoT traffic type 4: This type represents stochastic and downstream traffic. An example of such a traffic is webcasting, where a bursty traffic, for instance video-streaming, is cast from a gateway to different end-devices. It can be used in various places like houses with smart TV, or between individuals on a game streaming. Virtual reality and augmented reality traffic also belong to this class. In such case, the traffic is latency-sensitive.

5) IoT traffic types 5 and 6: To test the limits of the technology, we propose two not realistic types, which correspond to extreme conditions for a network. In this pattern, the radio medium is used at its maximum, with all sending end-devices (end-devices if upstream (type 5) or gateway if downstream (type 6)) having always a packet to send. We believe it helps to characterise the technology in worst conditions and lead to interesting results on the limits of a technology.

Similar applications can be grouped according to this classification. Table I gives a synthesis with some examples of well-known applications.

B. *IoT-relevant metrics*

We define now IoT relevant metrics as network technology performance parameters: (i) throughput, (ii) latency, (iii) success rate, (iv) energy consumption metrics and (v) scalability index.

Throughput is a practical measure of effective packet delivery, and it represents the overall speed of the network. Latency, which is the time that a packet takes to transit from its source to its destination, is also important when packets must be delivered in real-time, especially in critical operations like rescue using drones. The success rate which is the ratio of the packets successfully received from all the sent packets, does matter when the good reception of packets is mandatory and reliable data delivery SLAs are imposed. The energy metric is highly important in the IoT industry where end-devices have often limited power supply (battery). We call scalability index the maximum number of devices that can be managed by a single gateway without deteriorating the performance in terms of throughput, latency, success rate, energy, etc. This metric is important to limit the cost of a deployment and maximize the device density. It can be derived from the preceding ones, depending on the application needs. For instance, for a latency-sensitive application, we will focus on the latency to determine the scalability index.

C. *Evaluation tool*

The performance evaluation can be realized through different tools: experimentation, analytical modelling or simulation. Experimentation offers the advantage of reliability and precision, since it is done with real hardware, but it has expensive deployment costs. Analytical models typically offer scalability and reproducibility but are limited and not necessarily accurate. Finally, simulation is a trade-off between experimentation and analytical models. It enables to analyze a network technology by running its protocols in silico at scale without deploying hardware in real world and it provides reproducibility. However, the results may be hard to interpret and confidence intervals will be needed for validation.

Depending on their requirements, users can choose between one or several of these evaluation tools to apply our methodology. For high precision in the results, experimentation will be prioritized, while for scalability analysis, one will select analytical modelling or simulation.

D. *Instantiating a scenario*

In this last part, the user defines the scenario implementing the application with low-level parameters. These parameters are (1) the number of nodes and their positions, (2) the number of gateways, (3) the network technology defined by its PHY and MAC layers, the radio channel, the frequency and bandwidth, the transport protocol (UDP/TCP, MQTT, etc.) and (4) the traffic workload (packet size and period). Depending on the chosen evaluation tool, some parameters like the radio channel properties can be defined by the user (in case of analytical models and simulation) or be fixed by the evaluation environment (in case of experiments). This specification is critical to ensure results reproducibility.

The next step is to execute the scenario before gathering and analyzing the results. We show in the next section, how the evaluation can be done for two different network technologies: Wi-Fi for short-medium range applications and LoRaWAN for long range IoT applications.

III. Examples of application

For illustrating the methodology, we evaluate various applications using Wi-Fi or LoRaWan. These latter network technologies are two among the most adopted unlicensed technologies and address multiple IoT applications [14]. We use the simulator NS-3, which is well documented, popular and can simulate all the network layers.

Application	Traffic type	Packet size	Load freq	Mean load
Telemetry	1	1024 bytes	1 packet/sec	NA
Notifications	2	1024 bytes	1 packet/sec	NA
Video-surveillance	3	1024 bytes	NA	2 Mbps
Webcast	4	1024 bytes	NA	2 Mbps
Overload	5	1500 bytes	NA	500 Mbps

Table II: Wi-Fi application traffic specifications.

Propagation Delay Model	Constant Speed Propagation Delay
Propagation Loss Model	Log Distance Propagation Loss
Packet aggregation	Disabled
Frequency	5 GHz
MCS	9
Number of gateways	1
Number of end-devices	[1-22]
Distance end-devices - gateway	1 m
Bandwidth	80 MHz
Guard Interval	802.11ac: 400 ns 802.11ax: 800 ns
Channel number	42
Spatial streams	1
Buffer queue size (packets)	500
Transport protocol	UDP

Table III: Configuration of Wi-Fi simulations.

A. *Example on Wi-Fi*

The whole Wi-Fi stack is implemented in a detailed way in the official release of NS-3.32, and it is regularly updated with the last amendments, in particular: 802.11ac and 802.11ax (Wi-Fi 6). We consider 5 applications which are listed together with their traffic types specifications in Table II.

Table III presents the low-level instantiation parameters of our simulations, which are common to all the applications.

The results of the Wi-Fi evaluation are detailed below:

1) Telemetry: For this application the end-devices send a packet to the gateway every second. We observed that for this scenario and for both amendments, the number of end-devices does not affect the latency, which never exceeds 0.08 milliseconds (not shown in this paper). This is because with 1 packet per second, the radio medium is far from being overloaded. For the same reason, the success rate is close to 100% (Some transmission errors may be observed since the radio channel is not perfect). The scalability index can therefore attain hundreds of end-devices.

2) Notifications: The gateway sends one packet per second to each end-device. The same remarks as before can also be made for this application: latency never exceeds 0.08 milliseconds and the success rate is close to 100%. The same remark as before can be make regarding the scalability index.

3) Video-surveillance: We consider here that the traffic consists of video frames having a resolution of (1280 x 720), with 30 frames per second, compressed with H.264 technique. These specifications correspond to an application data rate of

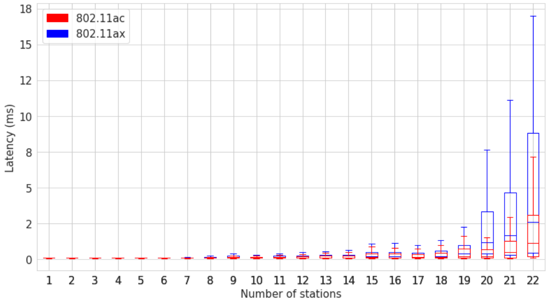


Figure 2: Latency for Video-surveillance in Wi-Fi.

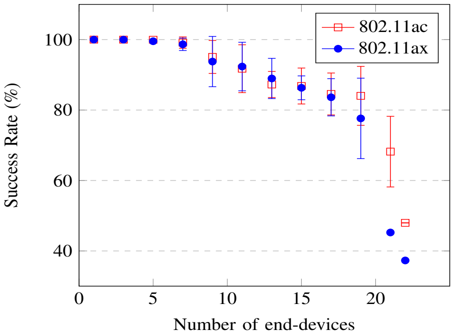


Figure 3: Success Rate for Video-surveillance in Wi-Fi.

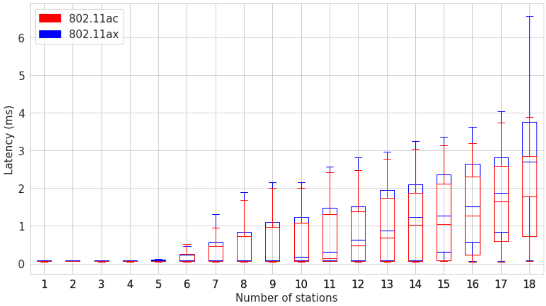


Figure 4: Latency for Webcast in Wi-Fi.

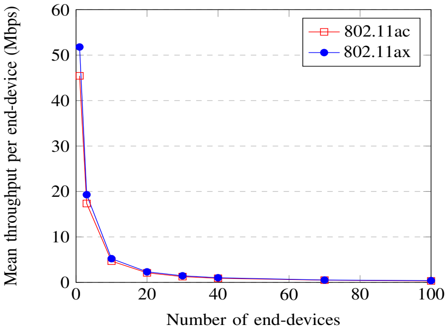


Figure 5: Mean throughput per end-device for Overload in Wi-Fi.

2 Mbps [15], with a packet size of 1024 bytes. The latency is displayed in Figure 2 as boxplots, where we observe that below 20 end-devices the latency is low, and it slowly increases until around 22 end-devices. These values are in line with the success rate graph in Figure 3 where we see that for 22 end-devices the success rate goes below 50% in a drastic way. Thus, the scalability index is equal to 20 end-devices for this application.

4) Webcast: The application data rate here is also fixed to 2 Mbps, with a packet size of 1024 bytes, on a downstream direction. The only measured metric is the latency, because only transmission errors may deteriorate the success rate (No collision may occur since there is one sender). We see in Figure 4 that the latency is increasing until 18 end-devices for 802.11ac and 802.11ax. We also observed that for approximately 20 end-devices in both amendments, the latency grows up in a fast way reaching almost hundreds of milliseconds (not shown in this paper). This is because starting from these values, there is an overloading in the gateway where the queue becomes full. Thus, the approximate value of the scalability index for this application is 20.

5) Overload: As mentioned before, this application aims at testing the limits of a technology. The data rate for each end-device is set at 500 Mbps, above the possible physical data rate for our simulation configuration. The packet size has been set to 1500 bytes. We only test an upstream traffic because it is much more difficult to predict the behaviour of the technology when there are several senders.

Figure 5 shows the evolution of the mean throughput obtained by each end-device, for different numbers of end-devices in the network. As expected, for both amendments, the throughput decreases fast, until reaching less than 1 Mbps at about 20 end-devices. In Figure 6, the medians of latency are getting higher as the number of devices increases, which is also due to the frames collisions. However, they stagnate at around 2.5 ms, which can be considered as a reasonable latency for most Wi-Fi applications [16].

Finally, we show the evolution of the success rate in Figure 7. It decreases fast until attaining 50% at 10 end-devices. Therefore, we can consider that the scalability index for the Overload application is 10 end-devices.

Discussion: We can conclude from the presented results that, except for the downstream applications and when the queue in the gateway becomes full (like for Webcast), the bottleneck in Wi-Fi networks can generally be observed in the success rate and the throughput for the considered scenarios, while the latency remains somehow insensitive to higher density.

B. Example on LoRaWAN

We now test the methodology on a long-range technology, which is LoRaWAN. Even though LoRaWAN stack is not implemented in the official release of NS-3, a link to a public repository [17] is provided in the official NSNAM website[1]. This technology is limited by some inherent restrictions. For instance, the maximum size of packets depends on the spreading factor. Also, there is a 1% duty cycle on European

[1] https://apps.nsnam.org/app/lorawan/

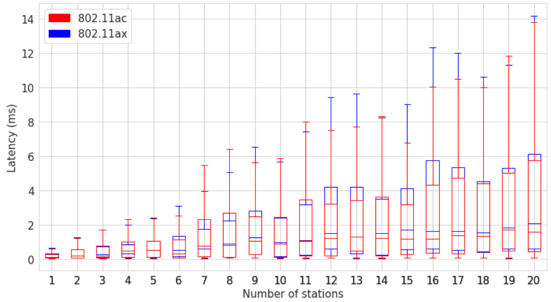


Figure 6: Latency for Overload in Wi-Fi.

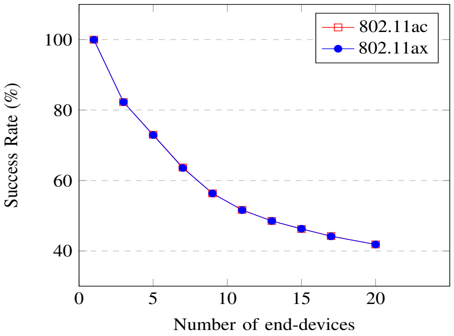


Figure 7: Success rate for Overload in Wi-Fi.

bands which prohibits a end-device from sending a packet before 100 times the time that the last sent packet took on air. There is no listen-before-talk mechanism at the MAC level of LoRaWAN, which implies that there is no waiting time before sending a packet, but this comes at the expense of more collisions. Therefore, the time that packet takes from a source to a destination (latency) is already known, depending of the spreading factor (SF) and the size of the packet, following a specific formula detailed in [2]. Figure 8 represents the evolution of this time depending on the SF and the payload size. For this reason we only focus on the success rate and the throughput in what follows.

As the 1% duty cycle does not allow stochastic traffic, traffic types 3 and 4 cannot be tested. The type 2 cannot be tested neither because NS-3 implements only the Class A devices, thus, the only possible traffic to simulate is upstream [18]. However, this is not really restrictive, because even though communication in LoRaWAN can be bi-directional, in practice, uplink communication from end-devices to the network server is strongly favoured [19]. For these reasons, we test only applications from traffic types 1 and 5 for which we detail the traffic specifications in Table IV, and the low-level instantiation parameters in Table V.

We simulate scenarios of these two types and we present the associated results:

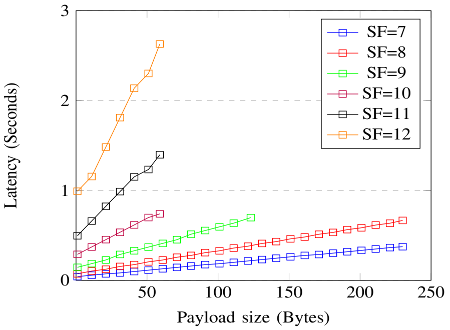


Figure 8: Latency according to packet size in LoRaWAN.

Application	Traffic type	Packet size	Load frequency
Smart metering	1	23 bytes	1 packet / hour
Overload	5	[53-230] bytes depending on SF	1 packet / [30-260]s depending on SF

Table IV: LoRaWAN application traffic specifications.

1) Smart metering: We consider that the end-devices send one packet per hour, with a payload size of 23 bytes. We test different configurations by changing the SF for each one, and we measure the success rate, which is displayed in Figure 9. We notice that the SF strongly affects the performance of the technology in this application: For the first three SFs (from 7 to 9) the number of devices can attain tens of thousands of end-devices (until 40,000) while maintaining more than half of the transmissions successful (it attains 70,000 for SF=7), while for the other SFs it barely attains 10,000. This is due to the latency which increases with the spreading factor, and so does the probability of collision, which decreases the success rate. Thus, depending on SF, the scalability index can attain between 10,000 and 70,000 end-devices.

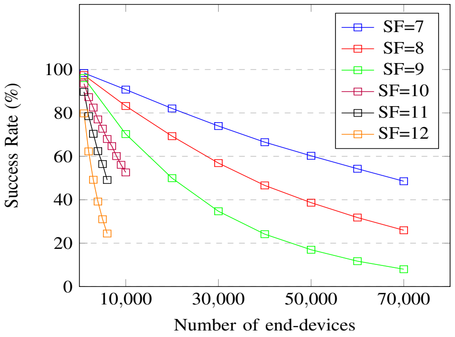


Figure 9: Success rate for Smart metering in LoRaWAN.

2) Overload: We overload the network by tuning different parameters that act on LoRaWAN, which are the SF, varying it from 7 to 12, the payload size, by pushing it to its maximum according to the SF (since the values are fixed in advance), and finally the packet load frequency: We put it at the highest possible value, taking the 1% duty cycle into consideration. For example, a 230 bytes packet does not take less than 0.3 seconds, thus the end-device will not be able to send another packet for 30 seconds. Therefore, for SF = 7, the load frequency is set at 1 packet each 30 seconds.

Figure 10 shows the evolution of the mean throughput per end-device. We see that it is not decreasing as fast as for Wi-Fi, because the 1% duty cycle restricts the amount of load

Propagation delay model	Constant speed propagation delay
Propagation loss model	Log distance propagation loss
Frequency	868 MHz
Bandwidth	125 KHz
Coding Rate	4/5
Cyclic redundancy check	Enabled
Low Data rate optimization	Disabled
Implicit header mode	Disabled
Preamble symbols	8
Number of gateways	1
Number of end-devices	[1-70,000]
Distance end-devices - gateway	[100-6000] m

Table V: Configuration of LoRaWAN simulations.

that each node, be it in saturation, can bring to the network. The packet success rate is shown in Figure 11, where we observe that for all the values of SF the curve follows the same behaviour, and attain 50% at practically 100 end-devices. We should note that the number of packets is not the same for each SF, since the load frequency changes. The scalability index is therefore equal to 100 for the Overload scenario in LoRaWAN.

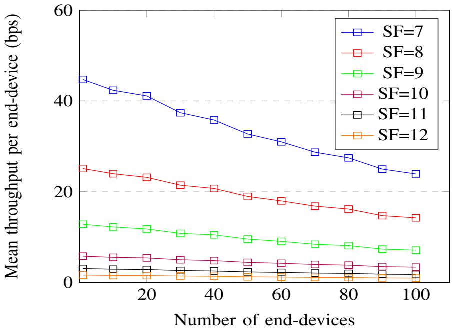


Figure 10: Mean throughput per end-device for Overload in LoRaWAN.

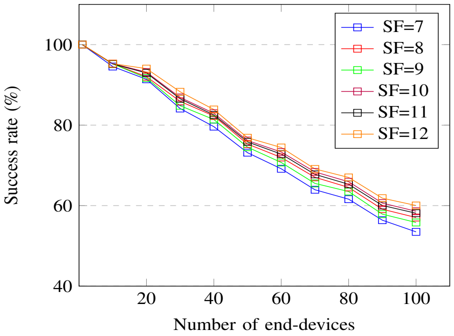


Figure 11: Success rate for Overload in LoRaWAN.

Discussion: The presented results of the instantiation of Smart metering and Overload scenarios show that the 1% duty cycle typically slows down the collapse of performance in terms of throughput obtained by each device and success rate, since one end-device can at most overload the network at 1%.

IV. Conclusion

In this paper, we laid out a versatile methodology to evaluate the suitability of an IoT network technology for a given application. The performance data used by the methodology can originate from a simulator, an experimental testbed, or an analytical model. We illustrate the versatility of our methodology using several examples of IoT applications, two widespread network technologies, and the simulator NS-3. In particular, we explored the maximum number of end-devices that can be satisfactorily handled in each example. For the sake of reproducibility, we made the code repositories for our simulations available in [20] and [21].

As future works, we intend to enhance our methodology in two ways. First, we will include energy-related metrics. This will likely require implementing and integrating existing energy models in a simulator as most existing wireless network simulators currently do not possess this feature. Second, we plan to compare IoT network technologies on the same application contexte using our methodology. For instance, given an application and a testing environment, a clear comparison between the performance of LoRaWAN and NB-IoT may turn to be of strong interest to IoT network architects in the choice of the right technology for their needs.

References

[1] J. C. Zuniga and B. Ponsard, “Sigfox System Description,” *IETF*, 2016.
[2] Semtech Corporation, “SX1272/3/6/7/8 LoRa Modem Design Guide, AN1200.13,” 2013.
[3] R. S. Sinha, Y. Wei, and S. H. Hwang, “A survey on LPWA technology: LoRa and NB-IoT,” *ICT Express*, 2017.
[4] B. Foubert and N. Mitton, “Long-range wireless radio technologies: A survey,” *Future Internet*, vol. 12, no. 1, 2020.
[5] C. Gomez et al., “Overview and evaluation of bluetooth low energy: An emerging low-power wireless technology,” *Sensors*, 2012.
[6] S. C. Ergen, “ZigBee/IEEE 802.15. 4 Summary,” *UC Berkeley*, 2004.
[7] B. M. Gast, *Wireless Networks: The Definitive Guide*, 2002, no. April.
[8] J. G. Andrews et al., “What will 5G be?” *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 6, pp. 1065–1082, 2014.
[9] K. B. Letaief, et al., “The roadmap to 6g: Ai empowered wireless networks,” *IEEE Communications Magazine*, 2019.
[10] W. Kassab and K. A. Darabkh, “A–Z survey of Internet of Things: Architectures, protocols, applications, recent advances, future directions and recommendations,” *Journal of Network and Computer Applications*, no. September 2019.
[11] M. O. Farooq and D. Pesch, “Analyzing LoRa: A use case perspective,” *IEEE World Forum on Internet of Things*, no. February, 2018.
[12] Y. Lalle et al., “A Comparative Study of LoRaWAN, SigFox, and NB-IoT for Smart Water Grid,” *Global Information Infrastructure and Networking Symposium*, 2019.
[13] J. Sommers and P. Barford, “Cell vs. wifi: On the performance of metro area mobile connections,” in *Proceedings of the 2012 Internet Measurement Conference*. Association for Computing Machinery.
[14] L. Alliance and I. W. Group, “Wi-Fi & LoRaWAN ® Deployment Synergies,” no. September, 2019.
[15] W. Paper, “Understanding IP Surveillance Camera Bandwidth,” 2017.
[16] C. Pei, et al., “Wifi can be the weakest link of round trip network latency in the wild,” in *IEEE INFOCOM*.
[17] D. Magrin, M. Centenaro, and L. Vangelista, “Performance evaluation of LoRa networks in a smart city scenario,” *IEEE ICC*, 2017.
[18] P. S. Cheong et al., “Comparison of lorawan classes and their power consumption,” in *IEEE Symposium on Communications and Vehicular Technology, SCVT*, 2017.
[19] A. Ferran, et al., “Understanding the Limits of LoRaWAN,” *IEEE Communications Magazine*, 2017.
[20] “Wi-fi,” https://gitlab.inria.fr/ssimoham/wifi-simulations-ns3.
[21] “Lorawan,” https://gitlab.inria.fr/ssimoham/lora-simulations-ns3.