Improving LoRaWAN Performance by Randomizing Network Access for Data and On-Air Activation

Konstantin Mikhaylov[1], Pavel Masek[2], Tuomo Hanninen[1], Martin Stusek[2], and Jiri Hosek[2]

[1]Centre for Wireless Communications, University of Oulu, Oulu, Finland
[2]Department of Telecommunications, FEEC, Brno University of Technology, Brno, Czech Republic
Contact author's e-mail: konstantin.mikhaylov@oulu.fi

***Abstract*—During the past years, the LoRaWAN technology took a prominent position among the wireless connectivity solutions for the Internet of Things (IoT) and has attracted substantial attention. The LoRaWAN technology has become popular for collecting the data from the sensors, which traditionally have periodic communication patterns. Meanwhile, recent studies have shown that the LoRaWAN procedures, such as the over-the-air activation, may also compromise the uniformity of data traffic distribution. Therefore, in this study, we investigate how the communication pattern of LoRaWAN devices during activation and data communication affects the overall network performance with respect to speed of activation and overall packet delivery probability. We show that the periodic communication patterns, widely employed by commercial LoRaWAN devices today, are less efficient than the patterns featuring random delays between the packets. Also, we find that introduction of a random delay between uplink data packets helps randomizing the channel access and enables network performance boost both for the application data transfer and during the activation. Finally, we show that implementation of the suggested communication patterns modification is feasible for the state-of-the-art LoRaWAN transceivers with no hardware modifications.**

***Index Terms*—LoRa, LPWAN, Traffic, Pattern, Optimize, MAC**

I. Introduction

The introduction of Low Power Wide Area Network (LPWAN) technologies enabling low-energy and low-cost radio-based connectivity for versatile machine applications to transfer of the so-called "small data" has drastically changed the landscape of the radio access technologies (RATs) for the Internet of Things (IoT). Among several LPWAN RATs available on the market, the Long Range Wide Area Network (LoRaWAN) is the most widespread technology operating in the license-free frequency bands [1] today. The support by LoRaWAN of private deployments, which companies or even individuals can carry, made this technology especially popular and generated a specific business niche.

Significant efforts have been invested over the past years to understand better and boost the performance of LoRaWAN devices and networks. Specifically, several studies focused on developing better models, simulation tools and carry experimental studies to characterize the performance of LoRaWAN networks in the different environments [2]–[6]. Also, quite much work has been devoted to optimizing the configuration of the communication parameters used by the devices in LoRaWAN networks, as discussed, e.g., in [7]–[11].

However, one aspect which has gotten quite limited attention so far is the effect of the data traffic on the performance of LoRaWAN networks. Specifically, it is often implied that since the different machine-devices operating in LoRaWAN are typically not synchronized, they generate the data and access the radio channel independently and uniformly in time. However, as this is discussed and demonstrated, e.g., in [12] the real-life LoRaWAN devices often send their data periodically. Notably, in our recent study [13] we have shown that the features of the LoRaWAN over-the-air activation (OTAA) procedure operating under duty cycle (DC) restrictions may also contribute to creating the non-uniform distribution of data packet transmissions even for the devices with uniformly distributed starting times. In this paper, we address this gap and (i) investigate how the communication pattern of a device affects the network performance, (ii) suggest modifications to the communication patterns to improve the performance. The key contributions of this study are:

- We discuss in detail the LoRaWAN mechanisms affecting the communication patterns and deliver MATLAB-based simulation models accurately modelling them.
- We investigate how the communication patterns (and, specifically, the introduction of random delays at different phases) affect the performance of LoRaWAN for data packet delivery and device activation robustness.
- We suggest the modifications to the data patterns and discuss their implementation on the LoRaWAN transceivers.

The paper is organized as follows. In Section II we recap the basic mechanisms of the LoRaWAN technology. Section III introduces the baseline and the proposed modifications. Section IV details the simulation models and our modelled scenario. The selected numeric results illustrating the performance of LoRaWAN are presented and discussed in Section V. Finally, Section VI concludes the paper, summarizes the results and pinpoints the potential directions for further studies.

II. Technical Background: LoRaWAN Operation

This section briefly discusses the LoRaWAN procedures defined in the LoRaWAN specification, relevant for this study. First, we briefly discuss the LoRa physical (PHY) layer and the methods for accessing the channel and transmitting a packet. Then we revisit the duty cycle handling and OTAA procedure.

A. Media Access, Architecture and Basic Procedures

The primary physical layer (PHY) of LoRaWAN is based on the LoRa modulation. This modulation is a variant of a

frequency chirp spread spectrum modulation, in which the instantaneous frequency is linearly increased and then wrapped to the minimum frequency when reaching the maximum frequency of the occupied band [14]. The spreading factor (SF) is the parameter of LoRa modulation, which specifies the number of chips per symbol [15]. By setting a higher value for SF, the on-air time of a frame is increased. This boosts the energy consumption and reduces network throughput but increases the maximum communication range. The transmissions with various SFs are quasi-orthogonal and can be sometimes demodulated simultaneously [15]. The SF can be allocated to an end device (ED), i.e., a LoRaWAN terminal, either statically or dynamically adjusted by the network through adaptive data rate (ADR) procedure [16]. The on-air time for LoRaWAN packets can be calculated using the following equations:

$$T_{\text{symbol}}(SF) = \frac{2^{SF}}{BW} \quad [\text{s}], \tag{1}$$

$$T_{\text{preamble}}(SF) = (L_{\text{preamble}} + 4.25) \cdot T_{\text{symbol}}(SF) \quad [\text{s}], \tag{2}$$

$$L_{\text{payload}} = 8 + \left\lceil \frac{(8B - 4SF + 28 + 16CRC + 20H)}{4(SF - 2DE)} \right\rceil \cdot (CR + 4) \quad [-], \tag{3}$$

$$T_{\text{payload}}(SF) = L_{\text{payload}} \cdot T_{\text{symbol}}(SF) \quad [\text{s}], \tag{4}$$

$$ToA(SF) = T_{\text{preamble}}(SF) + T_{\text{payload}}(SF) \quad [\text{s}], \tag{5}$$

where BW is the bandwidth in Hz, B is the payload (i.e., PHY_payload) in bytes, DE is 1 for SF11/SF12 for 125 kHz BW and 0 otherwise, CR denotes coding rate, CRC signalizes the presence of PHY layer cyclic redundancy code (1 for uplink (UL), 0 for downlink (DL)), H marks the presence of explicit header (for LoRaWAN $H = 1$), L_{preamble} and L_{payload} are the lengths of the preamble and the payload in symbols, respectively. For LoRaWAN operating in the EU, according to [17], [18], $CR = 1$ and $L_{\text{preamble}} = 8$.

The LoRaWAN networks are intended primarily to transfer data from EDs to a LoRaWAN gateway (GW) in UL. The GWs further transfer the data to a network server (NS). The NS manages the network, cashes all the messages, and may forward them further to application servers. The LoRaWAN specification defines several ED classes, but only class A is obligatory and the default one for all EDs.

The class A ED can initiate its UL transmission at any moment, as depicted in Fig. 1. Specifically, the ED randomly picks one frequency channel served by the network and sends its data. Following each UL transmission (after the delays given by LoRaWAN configuration parameters RECEIVE_DELAY1 and RECEIVE_DELAY2, which we for conciseness denote T_{RX1} and T_{RX2}, respectively), the ED opens up to two receive windows (marked as RX1 and RX2, respectively). The network can use either of them to reach the ED in DL with an application data or a command. At this point, it is worth noting that RX1 is opened in the same frequency channel that has been used for UL, while RX2 is allocated in a dedicated frequency channel, which is often

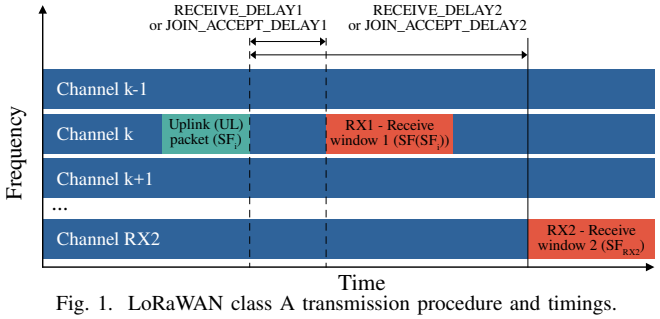

Fig. 1. LoRaWAN class A transmission procedure and timings.

TABLE I
EU LoRaWAN Frequency Bands and Limitations [19], [20].

Subband [20]	Frequency band, MHz	max DC, %	max power, mW
h1.3	863-865	0.1	25
h1.4	865-868	1	25
h1.5	868-868.6	1	25
h1.6	868.7-869.2	0.1	25
h1.7	869.4-869.65	10	500
h1.9	869.7-870	1	25

configured: (i) not to overlap with the channels employed for UL data transfers and (ii) feature high DC limit. Similarly, the SF which a network needs to use during RX1 depends (mostly often equals) on the SF used by the ED in UL, while the SF of RX2 is fixed (typically set to the maximum possible).

B. Frequency Regulations and Duty-Cycle

The LoRaWAN media access does not employ any listen-before-talk (LBT). Thus the regulations on the frequency use (i.e., [20] in EU) restrict the relative time an ED can transmit on a specific frequency subband by specifying the maximum DC threshold. Specifically, the recommended (in [19]) frequency channels for LoRaWAN operation in EU and the respective DC restrictions are summarized in Table I. Note that LoRaWAN regional parameters document [18] prescribes each network to support the three default LoRaWAN channels (centred at 868.1 MHz, 868.3 MHz, and 868.5 MHz). Notably, these also have to be utilized for activation, as discussed in the following section. Also, due to its high DC and maximum transmit power allowance, a channel in subband h1.7 is often reserved in LoRaWAN networks for RX2 [17], [19].

Even though the latest versions (i.e., 1.1 and 1.04) of the LoRaWAN specification do not specify how DC should be accounted for, the earlier versions (e.g., 1.0) explicitly require an ED not to transmit on a channel in the same subband for

$$T_{\text{backoff}} = \frac{ToA}{DC} - ToA \quad [\text{s}] \tag{6}$$

after it has used the subband to send a packet with on-air time ToA, where DC is the maximum DC for this subband. The key benefit of this method is its easy implementation.

C. Activation Procedures

The activation procedure serves for introducing an ED to the network and providing the ED with all the credentials (namely, the device address - DevAddr, the three network session keys (i.e., NwkSEncKey/SNwkSIntKey/FNwkSIntKey), and the application session key (AppSKey)) allowing a device

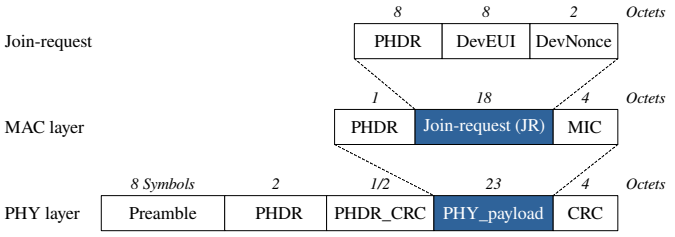

Fig. 2. Format of LoRaWAN Join-request frame.

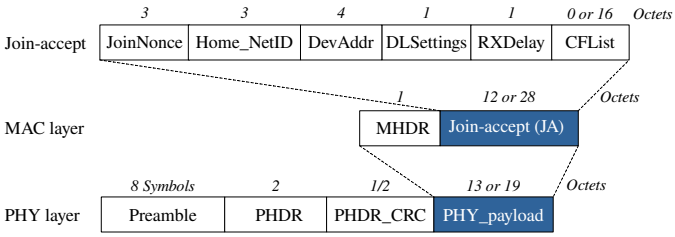

Fig. 3. Format of LoRaWAN Join-accept frame.

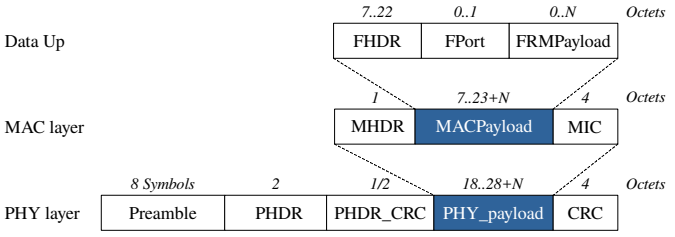

Fig. 4. Format of LoRaWAN uplink data frame.

to operate in the network. The LoRaWAN specification [17], defines two activation procedure options.

The activation by personalisation (ABP) suggests all keys and credentials be provided offline, e.g., written to the memory of the ED. An ABP-activated device can start communication right away after power-up. However, the main downside of ABP is the lack of possibility to change the keys, even if these get compromised, throughout the device's lifetime, except through a manual reprogramming [21].

The OTAA implies generating a new set of keys for every new communication session. It is thus considered to provide a higher level of security [21] than ABP. For this reason, OTAA is the recommended activation method in many real-life LoRaWAN networks. The OTAA procedure composes the exchange of two messages, named Join-request (JR) and Join-accept (JA). The format of these two messages and that of a generic UL data packet are depicted in Figs. 2-4. Note that as per LoRaWAN specification, the DL packets do not include CRC at PHY layer. This is also worth noting that the JA packet may also include optional channel list (CFList field) containing the list of UL channels supported by the network. The size of the packet payload for these three types of packets is given by: $B_{\mathrm{JR}} = 27$, $B_{\mathrm{JA(noCFList)}} = 13$, $B_{\mathrm{JA(CFList)}} = 29$ and

$$B_{\mathrm{Data}} = 16 + sign(B_{\mathrm{App}}) + B_{\mathrm{App}} \quad [\mathrm{byte}], \qquad (7)$$

where B_{App} is the application-layer payload in bytes (note that the maximum payload varies depending on the SF used). The

TABLE II
ON-AIR AND BACKOFF TIME FOR JA AND JR.

Parameter	Time, s					
	SF7	SF8	SF9	SF10	SF11	SF12
On-air time,s						
JR	0.06	0.11	0.21	0.37	0.82	1.48
JA no CFL	0.04	0.08	0.14	0.29	0.58	1.16
JA CFL	0.07	0.12	0.23	0.41	0.82	1.64
Backoff time, s (DC=0.1%)						
JR	61.67	113.15	205.82	370.69	823.30	1482.75
JA no CFL	46.29	82.35	164.70	288.48	576.96	1153.92
JA CFL	66.75	123.27	226.08	411.24	904.31	1644.95
Backoff time, s (DC=1%)						
JR	6.17	11.32	20.58	37.07	82.33	148.28
JA no CFL	4.59	8.16	16.32	28.59	57.18	114.35
JA CFL	6.62	12.22	22.40	40.75	89.62	163.01
Backoff time, s (DC=10%)						
JR	0.62	1.13	2.06	3.71	8.23	14.83
JA no CFL	0.42	0.74	1.48	2.60	5.20	10.40
JA CFL	0.60	1.11	2.04	3.71	8.15	14.82

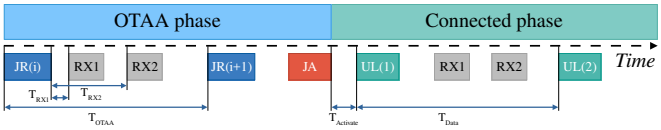

Fig. 5. Illustration and notations for the communication pattern.

$sign()$ function takes the value of 0 in case $B_{\mathrm{App}} = 0$ and 1 if $B_{\mathrm{App}} > 0$ and accounts for the presence of the frame port (FPort) field. Note that (7) describes the case when the optional frame field (i.e., FOpts) is not included in the packet. Also, in the case of application UL data packets, the time from the end of a UL transmission to RX1 and RX2 is defined by parameters $RECEIVE\_DELAY1$ and $RECEIVE\_DELAY2$. We imply that these parameters take the values recommended by LoRaWAN specification, which are 1 s and 2 s, respectively. During the OTAA procedure, instead of these two parameters, the delays $JOIN\_ACCEPT\_DELAY1$ and $JOIN\_ACCEPT\_DELAY2$, having recommended values of 5 s and 6 s, respectively, are utilised. The on-air time for JR and JA packets for different SFs are listed in Table II.

III. Communication Patterns: Baseline Scenario and Suggested Modifications

The generalized communication pattern of a LoRaWAN ED and the key notations are presented in Fig. 5, and the corresponding state diagram is depicted in Fig. 6. Specifically, the communication pattern is defined by the three main parameters. The first one is T_{OTAA}, which denotes the time period between two consecutive JRs. The second one, $T_{activate}$, denotes the time period between the end of JA reception and the first UL packet carrying data. Finally, the T_{data} denotes the time period between the two consecutive uplink data packets.

A. Baseline Scenario

As the baseline scenario we use the conventional communication pattern of LoRaWAN sensors of today, which implies periodic transmission of packets both during the OTAA and data communication, and thus T_{OTAA}, $T_{activate}$, and T_{data}

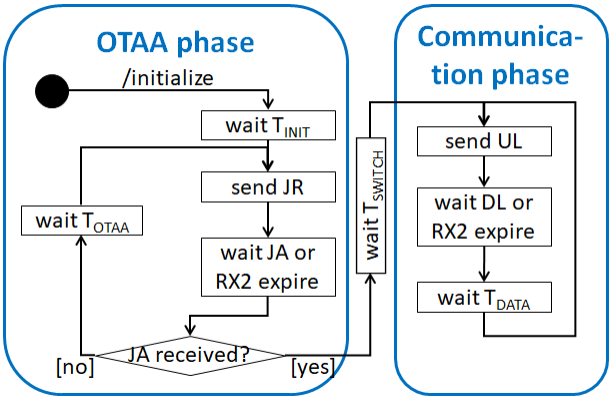

Fig. 6. Generalized state diagram of ED's operation.

take constant values, i.e., $T_x = c_x$, where c_x denotes the constant.

B. Suggested Communication Pattern Modifications

Contrary to the considered baseline scenario, we propose making T_{OTAA}, $T_{activate}$, and T_{data} a combination of two components: a constant and a random one. Thus either of these parameters can be expressed as $T_x = C_x + R_x rand(0..1)$, where C_x denotes the constant component, R_x - the random component multiplier, and rand(0..1) denotes a uniformly-distributed random variable between 0 and 1.

Note that we see an added value and propose introducing the random component into all three parameters (i.e., T_{OTAA}, $T_{activate}$, and T_{data}). However, to better understand how the three considered communication pattern parameters affect the resulting performance, we also investigate the cases when the random component is introduced in just one of them.

IV. Validation Methodology and Simulator

To study the effect of the communication patterns on the LoRaWAN network performance with respect to the two parameters (i) the connection time, (ii) the overall packet delivery ratio (PDR, i.e., the share of all packets sent by EDs, which the GW has received), we have instrumented a special MATLAB simulation model (available from [22]).

The script accurately models the LoRaWAN OTAA procedure including the respective time delays, and the data communication following it. The model simulates the operation of a single LoRaWAN GW, and a number of EDs deployed around it. At the beginning of the simulation, all EDs are not activated. Therefore, the EDs start by transmitting the JR. Note that the transmissions of the different EDs are uniformly distributed in time (i.e., in Fig. 6 $T_{INIT} = rand(0..1)R_{INIT}$). Once the ED receives the JA from the GW, it stops sending the JRs and starts broadcasting the application UL packets. Note that the simulator accounts for the DC restrictions - if the packet cannot be sent (by ED or GW) due to DC, this packet is discarded. We consider that the GW is capable of full-duplex operation and thus can transmit and receive at the same time, though the simulator accounts for the uplink-downlink interference in the same frequency channel.

TABLE III
Key parameters for simulations.

Parameter	Value	Unit
$RECEIVE\_DELAY1$	1	s
$RECEIVE\_DELAY2$	2	s
$JOIN\_ACCEPT\_DELAY1$ (T_{JD1})	5	s
$JOIN\_ACCEPT\_DELAY2$ (T_{JD2})	6	s
DR_{JA} and DR_{UL}	0	-
$RX1DRoffset$	0	-
UL and RX1 number of channels	3	-
UL and RX1 subband	h1.5	-
UL and RX1 subband DC	1	%
RX2 subband	h1.7	-
RX2 DC	10	%
t	160,200,240	s
$c_{OTAA}, c_{activate}, c_{data}$	t	s
$C_{OTAA}, C_{activate}, C_{data}$	t	s
$R_{OTAA}, R_{activate}, R_{data}, R_{INIT}$	t	s
N_{ED}	2,4,8,16,32,64,128	-
T_{sim}	4	hours
iterations	100	-
$B_{\mathrm{JA}} = B_{\mathrm{JA(CFList)}}$	29	byte
B_{App}	9	byte

For tractability, we consider that all EDs use the same SF and can reach the GW. Furthermore, we imply that all packets overlapping in time, frequency and SF are lost. More details about the simulation model and results of its validation and calibration are available in [13], and the key simulation parameters are summarized in Table III. Note that for our simulations, we have implied the use of the three obligatory default LoRaWAN uplink channels as per EU region specification, and the use of h1.7 band, featuring the maximum possible DC, for RX2. During each iteration, we have simulated the operation of the network for 4-hour period, and 100 iterations were run with each set of parameters (i.e., the N_{ED} and t in Table III). For each iteration, we have logged the number of transmitted and received packets for each ED, as well as the timestamp when each ED has been activated (i.e., received the JA) in the network.

V. Numeric Results

First, we have assessed the time required for device activation. Figures 7 and 8 illustrate the time since the start of the simulation up to the moment when 50% and 100% of devices are activated in the network, respectively. The results are presented as an error-bar chart, with the mean and the standard deviation (over 100 iterations) for different number of EDs and different communication pattern options marked. Specifically, the baseline denotes strictly periodic OTAA and data traffic, the "random delay for all frames" refers to the pattern with random components present in T_{OTAA}, $T_{activate}$, and T_{data}. The three other curves imply the presence of a random component in only one of the elements, i.e., random delay between OTAA JR - T_{OTAA}, random delay after activation (i.e., between JA reception and the first data packet) - $T_{activate}$, and a random delay between data packets - T_{data}.

From our results, it can be seen that the time required for activating 50% EDs was about the same for all the communication patterns and, not surprisingly, increased with the increase of N_{ED}. However, from Fig. 8 one can see that the time needed for all devices joining the network varied

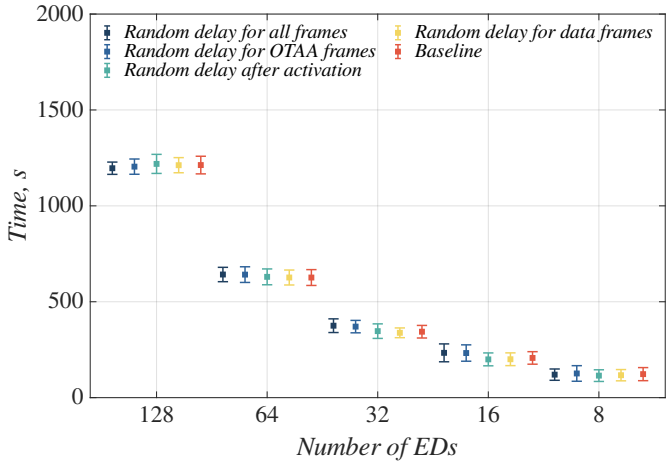

Fig. 7. Time for activating 50 % EDs for different number of EDs and communication patterns (t=200 s; mean and standard deviation for 100 iterations).

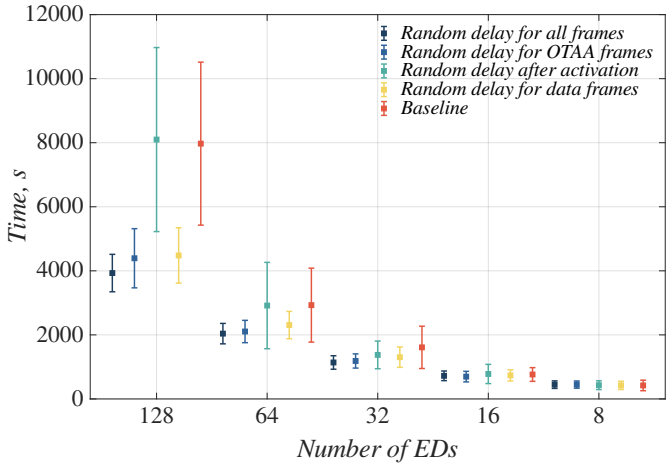

Fig. 8. Time for activating all EDs for different number of EDs and communication patterns (t=200 s; mean and standard deviation for 100 iterations).

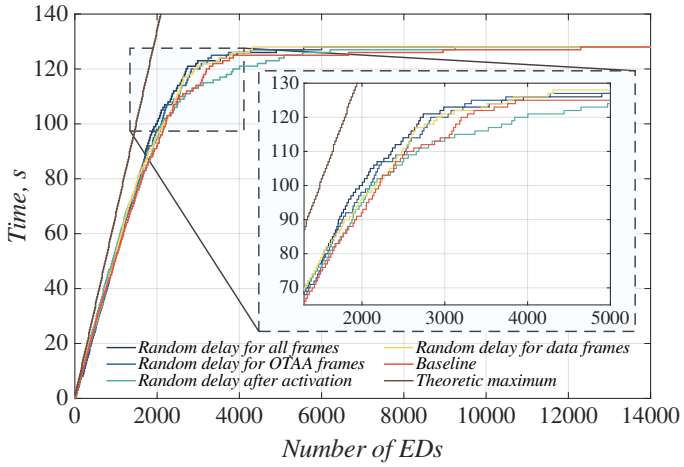

Fig. 9. Number of activated EDs (t=160 s; N_{ED}=160 s; illustration for a single iteration).

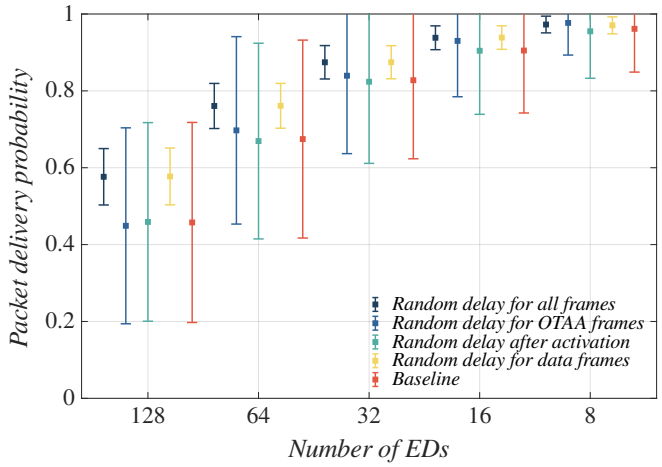

Fig. 10. Probability of data delivery after all devices have activated in the network (t=160 s; mean and standard deviation over 100 iterations).

depending on the communication patterns. Specifically, for the decently dense network, the proposed modification – i.e., introducing the random component at all phases of operation - enabled the minimum mean activation time and the minimum standard deviation. The introduction of a random delay during only one component - either T_{OTAA} or T_{data} - demonstrated a slightly (12% for $N_{ED} = 128$) worse performance. Both the baseline scenario and introduction of a random component in $T_{activate}$ featured the worst performance - the time of network building was twice lower than that for the other proposed communication pattern modifications. Note that an increase of the performance during activation due to the introduction of a random delay element to T_{data} may look counter-intuitive. However, one should note that the data transfers by the activated devices are carried in parallel and in the same frequency channels, which are used by non-activated devices for OTAA. Therefore, "randomization" of the data transfers results in a more uniform distribution of the collisions suffered by EDs during the OTAA, thus boosting their association pace.

Similar results can be observed from Fig. 9, which demonstrates the increase of the number of activated EDs in time for one illustrative iteration for different communication patterns. At the initial phase, the pace of growth is rather linear and does not differ much for the various patterns. However, once approximately half of the EDs are connected to the network, the growth becomes sublinear, and the pace of growth for the baseline scenario starts to lag behind. Note that Fig. 9 also depicts the "theoretical maximum" curve. Specifically, for the configurations listed in III, due to the DC limitations, a GW may accept nodes at minimum every $16.45/(1+0.1) = 14.95$ (16.45 is the ToA of JA; refer to Table II and note RX1 and RX2 DC). Since some of the JR and JA packets in RX1 are lost due to collisions and delays between the end of the back-off and the reception of a JR, the observed results are below the theoretical ones. Also, this can be seen that there is still space for improvement - implying "ideal" scheduling (which requires some form of synchronization between the EDs) all EDs can be admitted to the network in about 1900 s. This

is almost twice lower than the average time required for the proposed scheme and more than four times lower than the average performance of the baseline scenario.

Finally, Fig. 10 illustrates the PDR after the activation is done by all EDs. This can be clearly seen that for a decently dense network, the introduction of a random delay in-between the data packets allows boosting the average performance (e.g., for ED=128 from 45 % to 58 %) and reducing the fluctuations among the different EDs, thus making their performance with respect to the PDR more uniform. Meanwhile, we observed no improvement from an introduction of a random delay to T_{OTAA} and $T_{activate}$ as compared to the baseline scenario.

VI. Conclusions

The LPWANs and, especially, LoRaWAN networks are a critical element of machine connectivity for contemporary IoT applications. However, despite being in deployment and under research for a relatively long time, they still feature some unknown points and possibilities for performance improvement. One of these, which we have focused on in this study, is the communication pattern and its effect on performance of an individual device and the network as whole.

Specifically, the significant number of the real-life LoRaWAN devices today, especially the sensors, feature periodic data patterns both for their application-layer data and when activating in the network. However, as we have demonstrated, periodic communication patterns are ineffective during activation and data transfer. Meanwhile, an introduction of a random delay component in the communication patterns, as proposed in this paper, allows improving both the pace of EDs' association with the network (e.g., twice for a network composed of 128 EDs) and the average probability of packet delivery (by 13 %). Notably, our results show that even though an introduction of a random element to all phases of operation: (i) in OTAA, (ii) between OTAA and UL transfer, and (iii) between application UL packets gives the best results, an introduction of a random delay just between the data packets is often sufficient and provides a notable performance boost.

Also, we would like to highlight that the proposed communication pattern modifications are feasible, comply with LoRaWAN protocol, and can be implemented on real-life transceivers. To give a practical example, consider the RN2483 LoRaWAN transceiver by Microchip [23], which is often used as a component of LoRaWAN EDs. The transceiver automatizes some of the procedures (e.g., the OTAA); however, the transmission of UL data packets is fully controlled by the application processor (through *mac tx* command). Therefore, the processor can implement the suggested communication pattern modification and introduce a random delay between *mac tx* instances with no hardware change.

In our follow-up studies, we aim to practically implement the proposed modifications to study the performance of the proposed mechanisms in-field. Also, we plan to continue detailing the simulation models by introducing more realistic propagation and collision models and studying the effects of multiple GWs presence, as well as develop an analytic model.

VII. Acknowledgment

This study has been supported by the Academy of Finland projects: MRAT-SafeDrone (341111), RoboMesh (336060), FireMan (348008) and 6Genesis Flagship (318927).

References

[1] Statista, "Number of LPWAN Connections by Technology Worldwide From 2017 to 2023," 2021. [Online]. Available: https://www.statista.com/statistics/880822/lpwan-ic-market-share-by-technology/

[2] G. Di Renzone, S. Parrino, G. Peruzzi, A. Pozzebon, and D. Bertoni, "LoRaWAN Underground to Aboveground Data Transmission Performances for Different Soil Compositions," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–13, 2021.

[3] D. Magrin, M. Capuzzo, A. Zanella, L. Vangelista, and M. Zorzi, "Performance Analysis of LoRaWAN in Industrial Scenarios," *IEEE Trans. Ind. Inform.*, vol. 17, no. 9, pp. 6241–6250, 2021.

[4] M. Stusek *et al.*, "LPWAN Coverage Assessment Planning without Explicit Knowledge of Base Station Locations," *IEEE IoT J.*, pp. 1–1, 2021.

[5] R. Marini, K. Mikhaylov, G. Pasolini, and C. Buratti, "LoRaWANSim: A Flexible Simulator for LoRaWAN Networks," *Sensors*, vol. 21, no. 3, 2021. [Online]. Available: https://www.mdpi.com/1424-8220/21/3/695

[6] O. Elijah *et al.*, "Effect of Weather Condition on LoRa IoT Communication Technology in a Tropical Region: Malaysia," *IEEE Access*, vol. 9, pp. 72 835–72 843, 2021.

[7] H. Fawaz, K. Khawam, S. Lahoud, S. Martin, and M. E. Helou, "Cooperation for Spreading Factor Assignment in a Multioperator LoRaWAN Deployment," *IEEE IoT J.*, vol. 8, no. 7, pp. 5544–5557, 2021.

[8] R. Kufakunesu, G. P. Hancke, and A. M. Abu-Mahfouz, "A Survey on Adaptive Data Rate Optimization in LoRaWAN: Recent Solutions and Major Challenges," *Sensors*, vol. 20, no. 18, 2020. [Online]. Available: https://www.mdpi.com/1424-8220/20/18/5044

[9] G. G. M. de Jesus, R. D. Souza, C. Montez, and A. Hoeller, "LoRaWAN Adaptive Data Rate With Flexible Link Margin," *IEEE IoT J.*, vol. 8, no. 7, pp. 6053–6061, 2021.

[10] Y. A. Al-Gumaei, N. Aslam, X. Chen, M. Raza, and R. I. Ansari, "Optimising Power Allocation in LoRaWAN IoT Applications," *IEEE IoT J.*, pp. 1–1, 2021.

[11] F. S. Silva *et al.*, "A Survey on Long-Range Wide-Area Network Technology Optimizations," *IEEE Access*, vol. 9, pp. 106 079–106 106, 2021.

[12] R. Yasmin, K. Mikhaylov, and A. Pouttu, "LoRaWAN for Smart Campus: Deployment and Long-Term Operation Analysis," *Sensors*, vol. 20, no. 23, 2020. [Online]. Available: https://www.mdpi.com/1424-8220/20/23/6721

[13] K. Mikhaylov, "On the Uplink Traffic Distribution in Time for Duty-Cycle Constrained LoRaWAN Networks," in *Proc. 13th Int. Congress Ultra Modern Telecommun. Control Syst.*, 2021, pp. 16–21.

[14] S. Kim, L. Heonkook, and S. Jeon, "An Adaptive Spreading Factor Selection Scheme for a Single Channel LoRa Modem," *Sensors*, vol. 20, no. 4, 2020. [Online]. Available: https://www.mdpi.com/1424-8220/20/4/1008

[15] D. Croce, M. Gucciardo, S. Mangione, G. Santaromita, and I. Tinnirello, "Impact of LoRa Imperfect Orthogonality: Analysis of Link-level Performance," *IEEE Commun. Let.*, vol. 22, no. 4, pp. 796–799, 2018.

[16] R. Kufakunesu, G. Hancke, A. Abu-Mahfouz, and M. Adnan, "A Survey on Adaptive Data Rate Optimization in LoRaWAN: Recent Solutions and Major Challenges," *Sensors*, vol. 20, no. 19, 2020.

[17] N. Sornin and A. Yegin, Eds., "LoRaWAN 1.1 Specification," 2017.

[18] ——, "LoRaWAN 1.1 Regional Parameters," 2017.

[19] Semtech, "TN1300.01 - How to Qualify a LoRaWAN Device in Europe," 2018.

[20] CEPT ECC, "ERC Recommendation 70-03," 2021.

[21] LoRaWAN Alliance, "LoRaWAN Security: Frequently Asked Questions," 2016.

[22] K. Mikhaylov, "GitHub Repository: DOI:10.5281/zenodo.6034636," 2022. [Online]. Available: *https://github.com/kvmikhayl/LoRaWAN_traffic_pattern*

[23] Microchip, "RN2483 LoRa Technology Module Command Reference User's Guide," 2015.