Low-power Wide-Area Networks: A Comparative Analysis Between LoRaWAN and NB-IoT

Chiara Buratti*, **Konstantin Mikhaylov**†, **Riccardo Marini***, **Roberto Verdone***

*Università di Bologna, †University of Oulu

Abstract: *Low Power Wide Area Networks are becoming the most important enabler for the Internet of Things (IoT) connectivity. Application domains like smart cities, smart agriculture, intelligent logistics and transportation, require communication technologies that combine long transmission ranges, energy efficiency and low infrastructure costs. Recent and future trends make LoRaWAN and 4G/NB-IoT the main drivers of IoT business. In this contribution we briefly discuss the main features of the two technologies, analyzing some important Key Performance Indicators. The presented results have been obtained analytically, via simulations and experiments developed at the University of Bologna via testbeds, that are currently under use to verify different IoT applications and demonstrate their feasibility.*

1 Introduction

The proliferation of embedded systems, wireless technologies, and Internet protocols have enabled the Internet of Things (IoT), to bridge the gap between the virtual and physical world by enabling the monitoring and control of the physical world by data processing systems. A large variety of communication technologies has gradually emerged, reflecting a large diversity of application domains and requirements. Some of these technologies are prevalent in a specific application domain, such as Bluetooth Low Energy in Personal Area Networks [1], and Zigbee in Home Automation systems [2]. Others, like Wi-Fi Low Power, Low Power Wide Area Networks (LPWAN) [3], and cellular communications, such as the 3GPP Long Term Evolution for Machines (LTE-M) and Narrowband IoT (NB-IoT), have a much broader scope. In addition, this landscape is constantly and rapidly evolving, with new technologies being regularly proposed, and with existing ones proliferating into new application domains. In this Chapter, we focus on LoRa and NB-IoT, presenting their main features and characteristics, and some examples of achievable results via analyzing a set of selected generic Key Performance Indicators (KPIs). The Chapter is organized as follows: Section 2 deals with the LoRa technology, while Section 3 reports NB-IoT technology. Finally, Section 4 compares the two solutions and reports drawn conclusions.

2 LoRaWAN Technology: main features and characteristics

2.1 LoRaWAN Technology

2.1.1 Overview

LoRa is a Physical Layer developed by Cycleo (a French company) later acquired by Semtech, on top of which LoRaWAN specifies the link and network layer procedures. The first target of LoRa is to allow very low power operations to ensure with a single battery a long lifetime to the devices - of more than ten years. It also allows long communication ranges (2-5 km in urban areas and up to 15 km in suburban areas) [20, 18]. The downside is low data rates, some tens of bit per second in the most robust options. However, LoRa can offer certain flexibility and can reach a data rate up to 50 kbit/s [4, 5]. LoRa physical layer is based on Chirp Spread Spectrum (CSS) modulation. Using a bandwidth exceeding the necessary one to transmit the data, LoRa performs spectrum spreading, which brings robustness against some characteristics of the channel (e.g., interference, frequency selectivity, Doppler effect). One original characteristic of LoRa is that information is carried by a cyclic shift in the chirp (position modulation).

The transmitter generates chirp signals by varying their frequency over time and keeping phase between adjacent symbols constant. The signal frequency band is usually set to 125, 250 or 500 kHz in the Industrial Scientific Medical (ISM) bands of 863-870 MHz for Europe or 902-928 MHz for US[21]. However, there also exist some narrower bands (7.8 to 62.5 kHz) in the 166 and 433 MHz bands. Finally, a new version at 2.4 GHz has recently emerged. The main characteristics of LoRa's modulation depend on several parameters:

- The Spreading Factor (SF): it is related to the duration of a symbol. For the higher spreading factors, more chips are combined in a single symbol, making the transmission longer (thus reducing the data rate), but increasing its energy thus, potentially, allowing longer communication range. LoRa employs six quasi-orthogonal SFs (numbered 7 to 12). Consequently, up to six frames can be exchanged in the network at the same time over the same frequency channel, as long as each one is configured with unique SF.

- Forward Error Correction (FEC) techniques, and, specifically, Hamming code, are also used to increase receiver sensitivity. The Code Rate (CR) index specifies the number of additional bits added to a LoRa frame. LoRa offers CR = 0, 1, 2, 3 and 4, where CR = 0 means no encoding and the effective coding rate is 4/(4+CR), ranging from 1 (no coding) to 1/2.

- The output of the encoder passes through the Whitening block (optional). Whitening induces randomness, in order to make sure that there are no long chains of 0's and 1's in the payload. An interleaving block is then implemented to avoid bursts of errors. The interleaver uses a diagonal placing method to scramble each codeword.

A packet contains a preamble (for the detection and synchronization purpose), possibly a header (depends on operation mode) and the payload, with a maximum size between 51 bytes and 222 bytes, depending on the SF. The raw on-air data rate varies according to the SF and the bandwidth, and it ranges between 22 bit/s (BW = 7.8 kHz and SF = 12) to 27 kbit/s (BW = 500 kHz and SF = 7). The SF 6 offers another option with a rate of 50 kbit/s. Frequency hopping is exploited at each transmission in order to mitigate external interference. The choice of the bandwidth, the SF and the CR impact the Time-on-Air. An increase in this time will consequently increase the duration of the period the radio has to be off, which imposed by the frequency use regulation. Although few information bits are transmitted per packet, the packet duration can be long, more than one second for large SF and small bandwidth. To decode a packet, first, a receiver has to detect the preamble consisting of successive up-chirps (typically 4 or 6) and two down-chirps (the up-chirp reversed in time). This allows the synchronization and the detection of the beginning of the frame. The decoding consists of multiplying each symbol by a down-chirp. The resulting signal is a sine wave with a fixed frequency, given by the shift. The Fourier transform then exhibits a peak, easy to detect, that allows recovering the bit sequence. Besides, the capture effect allows receiving the target packet even under the interference of a signal with the same SF, given that the interfering signal is weaker than the target one.

LoRaWAN networks are based on single-hop transmissions, leading to a star-of-stars topology. Devices transmit their packets directly to Gateways that relay messages to a central Network Server, through another network (Cellular, Wi-Fi or Ethernet for instance). Bi-directional communications are allowed too. LoRaWAN defines three classes of devices (A, B and C):

- Class A devices, aiming low cost and long life devices, use pure ALOHA to access the channel in the uplink. A Class-A device is always in sleep mode unless it has something to transmit. After transmission, the device listens during two window periods, defined by duration, offset time and a data rate. Feedback can only happen after a successful uplink transmission. The second window can increase robustness in the downlink, and it is disabled when the end-device receives downlink traffic in the first window.

- Class B devices are designed to support additional downlink traffic, at the price of higher energy consumption. A Class-B device synchronizes its internal clock using beacons emitted by the gateway. This process is called a “beacon lock”. After synchronization, the device negotiates its ping-interval. The LoRa Server is then able to schedule downlink transmissions on each ping-interval. By doing so, additional downlink traffic can also be supported without relying on prior successful uplink transmissions.

- Class C devices are always listening to the channel except when they are transmitting.

Class A is intended for End-Devices (EDs). The other classes must remain compatible with Class A. The three classes can coexist in the same network and devices can switch from one class to another. However, there is no specific message defined by LoRaWAN to inform the gateway about the class of a device and, hence, this must be handled by the application.

An essential parameter in LPWANs and networks operating in unlicensed bands is the maximum allowed duty-cycle. It corresponds to the percentage of time during which an

Table 1: LoRa Key Parameters Values

Parameter	Value	Comment
Bit Rate	22 bit/s – 50 kbit/s	Depending on SFr
Frequency Bands	[69, 433, 868] MHz (Europe) 915 MHz (North America)	2.4 GHz version available
Bandwidth	[125, 250, 500] kHz	7.8 - 62.5 kHz Bandwidths available in the 433 MHz band
Topology	Stars of stars	
Link budget	155 dB – 170 dB	Depending on SF
TX Range	Up to 15 km	Few km in urban area
Consumptions (TX)	18 mA at 10 dBm 84 mA at 20 dBm	

end-device can occupy a channel and equals 1% in EU 868 for end-devices. The channel selection is pseudo-random and happens at each transmission.

The most critical parameters are summarized in Table 1.

2.1.2 Protocol Operation

LoRaWAN networks allow EDs to individually use any of the possible combinations of data rate and transmitted power. This is referred to as Adaptive Data Rate (ADR) and is designed in order to increase the battery life of the ED while maximizing the network capacity [8]. In order to determine the optimal data rate, the network needs to take some measurements: this is achieved by estimating the link budget between the ED and the gateway looking at uplink messages. For example, an ED very close to the gateway should transmit with the highest possible data rate (i.e., the lowest possible value of SF), in order to reduce as much as possible the Time on Air (ToA) of the transmitted packet, allowing the ED to reduce its energy consumption whilst also reducing the probability of collisions with other nodes. The algorithm operating on the network server is designed by the developer while the one working on the node is specified by LoRa Alliance [8].

EDs are in charge of deciding if ADR should be used or not. If it is activated (ADR bit in the frame header set to 1), the network server will control the transmission parameters of the ED through ad-hoc commands. When the ADR bit in the downlink packet is set to 1, the server informs the ED that it will send ADR commands; differently, the node will not receive any indication because the server is not able to estimate the best data rate to be used; this happens when the radio channel varies too fast. Besides this, the node should periodically check if the network still receives its uplink frames when ADR is enabled.

Figure 1 shows the algorithm implemented on the ED. Each time the uplink frame counter is incremented, the same happens to the ADR_ACK_CNT counter (except for repeated transmissions that do not increase the counter). After ADR_ACK_LIMIT messages (by default 64) without any downlink response, the device sends a request to the network, which must respond within the next ADR_ACK_DELAY frames (by default 32) with a downlink frame. If no reply is received, the ED must try to reconnect to the network by first setting the transmitted power to its default one and then possibly switching to the next lower data rate (which will provides longer transmission range). The device must lower its data rate every time the ADR_ACK_DELAY expires and, once it reaches the lowest data rate, it must re-enable all the default uplink frequency channels.

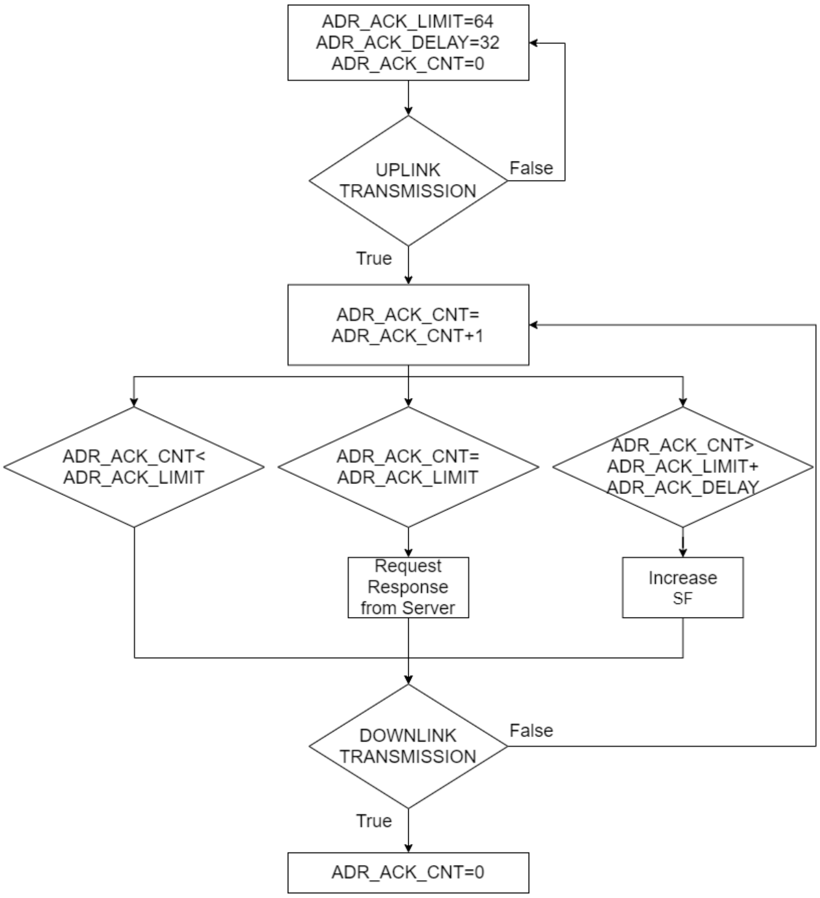


Figure 1: ED ADR

The ADR algorithm working on the network server exploits data regarding the uplink transmissions processed by the network server in order to define the optimal data rate to be used. One of the most widespread implementations, used by The Things Network or ChirpStack, to mention some of the most famous LoRa Server architectures, is based on Semtech's recommended algorithm [17]. It is shown in Figure 2.

Once the network server detects that the ED is sending a packet with the ADR bit set, it starts collecting Signal-to-Noise Ratio (SNR) measurements of the received signal. It keeps recordings, typically, of the 20 most recent transmissions from each ED. After receiving 20 samples, it computes the SNR margin: $SNR_{margin} = SNR_{max} - SNR_{req} - M$, where SNR_{max} is the maximum SNR among the collected data, M is set as 10 dB by default (it can be controlled by the server administrator), and SNR_{req} is the minimum SNR required to demodulate the received signal correctly, and it varies according to the data rate (or the Spreading Factor) as shown in Table 2.

Data Rate	Spreading Factor	SNR_{req}[dB]
0	12	-20
1	11	-17.5
2	10	-15
3	9	-12.5
4	8	-10
5	7	-7.5

Table 2: SNR_{req} values for different SF with BW=125 kHz.

From SNR_{margin}, N_{step} is computed as: $N_{step} = \left\lfloor \frac{SNR_{margin}}{3} \right\rfloor$.
Then an iterative process starts:

- If $N_{step} < 0$, the transmitted power is increase in each step by 3 dB until the maximum one (according to the regional regulations) is reached; N_{step} is increased by 1;
- If $N_{step} > 0$, first the algorithm tries to increase the data rate in each step until it reaches the maximum one (DR=5) or, if this is not possible anymore, the transmitted power is decreased by 3 dB until it reaches the minimum one; N_{step} is decremented by 1.

The algorithm stops when $N_{step} = 0$ and the server generates a specific packet with the new transmission parameters and sends it to the selected ED; the changes introduced will reflect on the next uplink message if the packet is correctly received.

2.2 LoRa KPI

We concentrate on the following KPIs: Reliability, Network Throughput and the End-to-End (E2E) Delay, and we provide some example of numerical results achieved via simulations and experiments. Simulations have been carried out on Matlab. A squared area with one gateway in the centre has been considered for the sake of simplicity; the square size has been chosen such that nodes using SF7 have a 90% probability of being connected to the gateway, while nodes using SF12 have a connection probability of 100%. In the simulator the path loss is modeled as follows: $Loss = k_0 + k_1 log_{10}(d) + s$, with $k_0 = 10 log_{10} \left(\frac{4\pi}{\lambda}\right)^2 \quad and \quad k_1 = 10\beta$, where β is the propagation coefficient and λ is the wavelength, d is the distance between transmitter and receiver. s represents random channel fluctuations, described as a Gaussian r.v. in dB, zero mean and standard deviation σ. The parameters used during simulation are reported in Table 3.

Concerning the experimental platform, Idesio Rigers Board 1.0, a multi-sensor platform specifically designed for smart city applications, has been used. It is equipped with the microchip RN2483 radio transceiver, fully certified 433/868 MHz LoRa module and it supports LoRaWAN Class A devices. Fifteen devices were programmed to work at the same time, divided into three clusters of 5 devices using respectively SF7, SF10 and SF12, sending packets every 60 s.

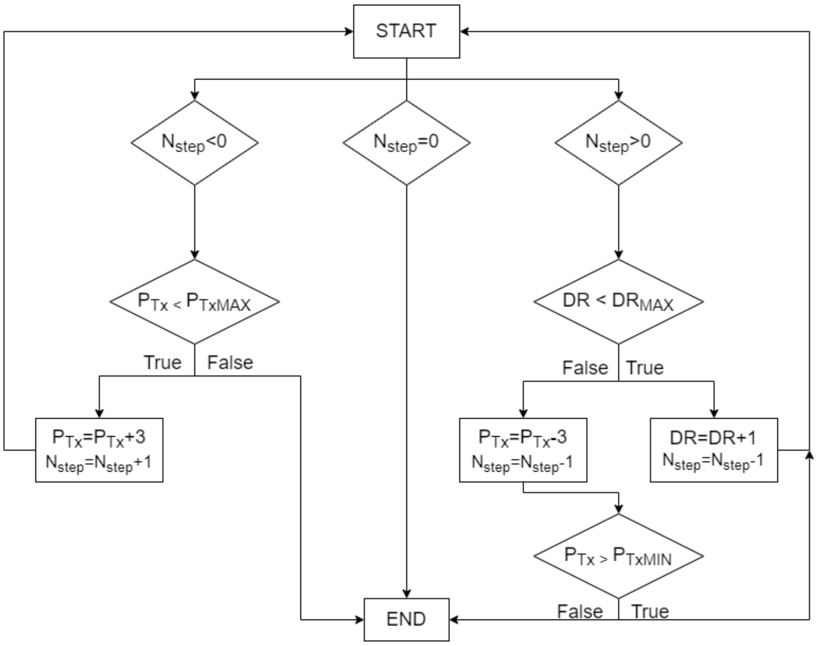


Figure 2: Network Server ADR Algorithm

Packet Periodicity T	60 s	Area Side	6500 m
Packet Size B	16 Bytes	Confirmed Message	Enabled
Preamble Length	8 Symbols	Header	Enabled
f	868,5 MHz	BW	125 kHz
CR	4/5	PTx	13.5 dBm
β	3	σ	3

Table 3: Simulation Parameters

2.2.1 Reliability

Reliability is mainly affected by the use of ALOHA protocol by Class A devices. It is well known that the performance of ALOHA is poor in terms of packet delivery success rate and, further, it does not support a network load increase, due to the interference increase. Hence, as long as the network load is low enough (taking into account the number of SFs, frequency channels, devices), reliability should be enforced. Figure 3 shows the Packet Error Rate (PER) as a function of the number of the nodes in the network, both considering simulations (curve) and experimental results (square points). When computing the PER in the simulations, we account for both, connectivity issues (i.e., probability that an ED is not connected to the gateway) and collisions (i.e., probability that more than one ED transmit at the same time, using the same channel and SF). In the experiments, three measurements sessions of 20 minutes each have been carried out and at the end statistics regarding the PER, obtained by counting the number of packets received/lost, have been computed. In the figure, the different curves are related specific values of SF (in this case, we fixed the same value for all nodes in the network), and to the case of ADR. As for the experiments, no ADR has been considered. As can be seen, the optimum value of SF to be set varies with the number of nodes: when few nodes are present, the PER is mainly limited by connectivity - therefore SF=10 is the best solution. When the traffic load increases, SF=7 becomes better, because it allows keeping under control collisions due to the smaller ToA. Besides this, introducing ADR drastically improves the performance, since it reduces both connectivity and collisions issues managing the SF used by the device, making it able to reach the gateway in almost all cases and distributing different SFs among all the nodes.

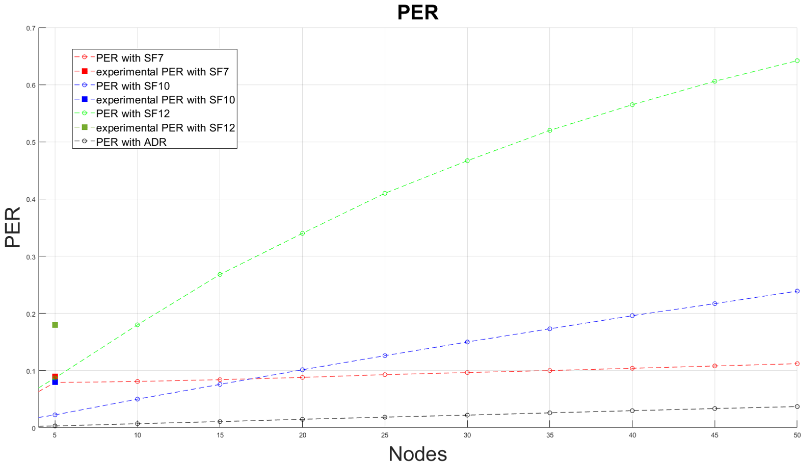


Figure 3: PER versus the number of nodes in the network.

2.2.2 Network Throughput

The network throughput is defined as the number of bits per second correctly received at the gateway, given by $S = \frac{B \cdot N \cdot (1-PER)}{T} \quad [bit/s]$, where N is the number of nodes, T is the period of time between two successive generated packets, and B is the packet size (see Table 3 for parameters settings). The network throughput is depicted in Figure 4 as a function of the number of nodes, and demonstrates a trend similar to that of the PER.

2.2.3 End-to-End Delay

We define the End-to-End (E2E) Delay as the interval of time between the generation of the packet at the network server to be sent in the downlink to a given node, and the instant when the network server receives a reply from the node. Tests and simulations to derive the average End-to-End Delay have been carried out, accounting for the fact that this delay strongly depends on the operating class used by the ED.

In the case of Class C devices, that is assuming the node are always on, the E2E Delay

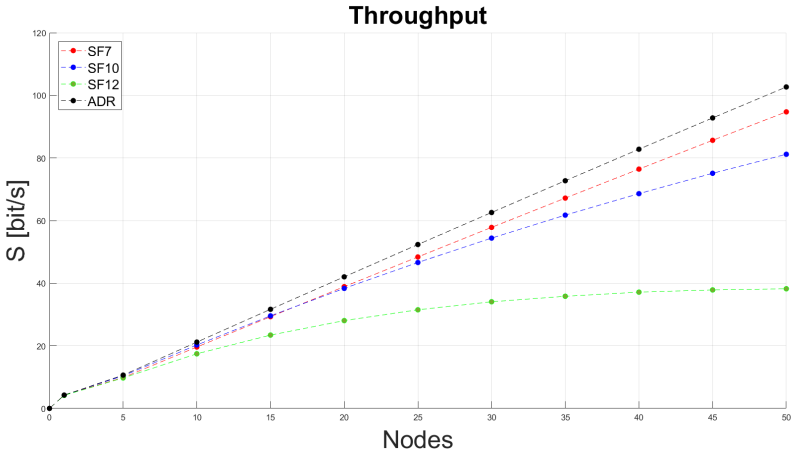


Figure 4: Throughput

is given by:

$$E2E_{Delay} = \tau_{NS-GW} + ToA_{DL} + T^{(node)}_{proc} + ToA_{UL} + \tau_{GW-NS} + T^{NS}_{proc} \quad [s] \qquad (1)$$

where $\tau_{GW-NS} = \tau_{NS-GW}$ is the propagation time from gateway to the network server and viceversa; ToA_{DL} and ToA_{UL} are the ToA of the packets transmitted in downlink and uplink, respectively (they depend on the SF set); $T^{(node)}_{proc}$ is the processing time at the ED and T^{NS}_{proc} is the processing time at the network server.

Since in class A the receive window is opened only after an uplink message, the server is able to send a downlink message (which contains the request) only after the correct transmission of the ED. This means that if for some reason an uplink packet is lost, the network server will not know that the receive window of the device is opened and therefore it will not send the downlink message, waiting for the next uplink. Therefore, in the case of Class A devices the packet to be sent in downlink remains at the network server for a certain amount of time, denoted as $T_{wait-NS}$, which depends on the frequency with which the ED generates packets in uplink and on the probability that this packet is sent with success. Therefore, since in our simulations (and also experiments) EDs generate an uplink packet every T, the E2E Delay for the case of Class A is given by:

$$\begin{aligned} E2E_{Delay} \quad = \quad & T_{wait-NS} + T_{RXwind} + ToA_{UL} + ToA_{DL} + T^{(node)}_{proc} + ToA_{UL} + \\ & \tau_{GW-NS} + T^{NS}_{proc} \quad [s] \end{aligned} \qquad (2)$$

where $T_{wait-NS} = \frac{T}{2} + T \cdot PER$ is the average waiting time of the packet at the network server, given that nodes generate packets in uplink every T and these packets have a probability of being correctly received given by $(1 - PER)$. T_{RXwind} is the interval between the end of uplink transmission and the beginning of the first receive window opened by the ED. The other terms are the same considered for Class C. Results are provided in Figure 5 for Class A and in Table 4 for Class C. Note that in the case of Class

C the network server can send downlink messages almost at any time; therefore there is no variation due to the number of nodes, because delay does not depend on the PER. This deeply reflects the user perception of the system and it appears clearly that Class A LoRaWAN is not thought for real-time application.

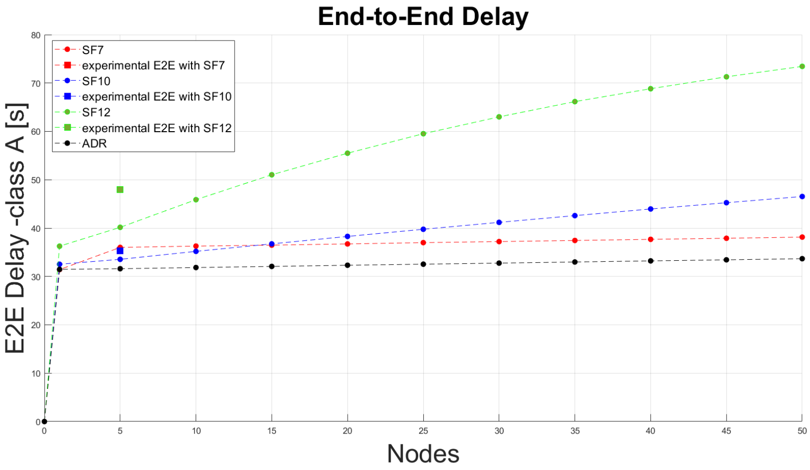


Figure 5: End-to-end delay class A

SF7	SF 10	SF12	ADR
0,38 s	0,9 s	2,79 s	0,38 s

Table 4: End-to-End Delay class C

3 NB-IoT Technology: main features and characteristics

3.1 NB-IoT technology

3.1.1 Overview

NB-IoT is designed to achieve efficient communication in the cellular IoT framework and reach a longer battery life for a massive distribution of nodes. Three key elements characterize it: low cost, a large number of connections per cell and robust coverage, with very good penetration in underground and indoor environments [6]. NB-IoT is introduced in Release 13 (Rel. 13) of 3GPP, emerging as an alternative solution to the LPWA technologies already present on the market (e.g., LoRaWAN). NB-IoT leverages on the LTE standard and numerology, but it is designed for ultra-low-cost Machine Type Communications (MTC), supporting a massive number of devices per cell. From LTE it takes the synchronization, radio access, resources definition and assignment. The standard

Table 5: NB-IoT Key Parameters Values

Parameter	Value	Comment
Bit Rate	up to 253.6 kbit/s	UE capabilities and netw. config.
Frequency Bands	various in [400,2700] MHz	TDD only in band 2490-2690 MHz
Bandwidth	180 kHz	200 kHz in re-farmed GSM
Topology	star of stars	similar to LTE
Link budget	up to 164 dB	netw. config.
TX Range	up to 35 km	expected to reach 100 km [16]
Consumption	TX: 230 mA at 23 dBm RX: 61 mA	[15]

allows modifications to regular LTE by enhancing the link budget and reducing the energy consumption, complexity and costs to a minimum.

While the other cellular systems for MTC are based on existing radio access technologies, NB-IoT can either operate in a stand-alone mode, within the guard bands of LTE carriers or within LTE carriers. It supports a nominal system bandwidth of 180 kHz (equal to the one of an LTE Physical Resource Block (PRB)) in both uplink and downlink. The (narrowband) channel spacing is 15 kHz as in LTE, but it can be decreased to 3.75 kHz in uplink communications [7]. Traditionally, in Rel. 13 and Rel. 14 the NB-IoT was limited to frequency-division duplexing (FDD) operation implying the use of different frequency bands for uplink and downlink transmissions. However, in Rel. 15 (2019) a new option - the time-division duplexing (TDD) - has been introduced allowing to use the same frequency band both for uplink and downlink.

As in LTE, NB-IoT eNBs (enhanced Node-B) employ Orthogonal Frequency Division Multiple Access (OFDMA) in the downlink, and the User Equipments (UEs, the term used in LTE to denote an end node or a user terminal) use Single Carrier Frequency Division Multiple Access (SC-FDMA) in the uplink. However, the modulation schemes are limited to Binary Phase Shift Keying (BPSK) and Quadrature Phase Shift Keying (QPSK) to reduce complexity and ensure a better link budget. A single process Hybrid Automatic Repeat Request (HARQ) is expected in both uplink and downlink by default (this requirement was relaxed in Rel. 14), and the half-duplex operation is allowed. NB-IoT UEs (cat NB1/NB2) implement power control in the uplink, in order to keep low power and consumption where possible.

The expected Coverage Enhancement (CE) is mainly achieved by allowing repetitions (i.e., temporal diversity [19]). The signalling for control information and data is repeated a number of times in different uplink and downlink channels. Each replica has a different coding, and multiple replicas can be combined at the receiver to increase the reception probability.

NB-IoT also introduces a UE categorization in several classes of devices, based on measured power levels. It allows an energy-efficient operation, though keeping an ultra-low device complexity. To further reduce costs, the device searches for only one synchronization sequence and can use a low sampling rate (e.g., 240 kHz) to establish primary time and frequency synchronization to the network. Also, the maximum transport block size is 680 bits/1000 bits in downlink and uplink in Rel. 13 (in Rel. 14 both were increased to 2536 bits) and a single transmit-receive antenna can guarantee the performance objectives of NB-IoT.

Techniques like Power Saving Mode (PSM) and extended Discontinuous Reception (eDRX) are used to increase the battery life for cellular IoT devices. Energy consumption critically depends on the device behaviour when it is not on an active session: these idle time intervals for cellular networks are used to monitor paging and perform mobility measurements. For this reason, PSM and eDRX support a reduced energy consumption by extending the periodicity of paging occasions or requiring no monitoring at all.

3.1.2 Protocol operation

To get a better understanding of the NB-IoT technology, we detail the operation of a UE operating in an FDD-based network, using different frequency resources for uplink and downlink [19]. In case of uplink, the resource grid is composed of multiple subcarrier frequencies with a step (the so-called frequency separation - Δf) of either 3.75 kHz or 15 kHz, and time slots with a duration of 0.5 ms and 2 ms in case of Δf=15 kHz and Δf=3.75 kHz, respectively. On top of this, NB-IoT introduces the notation of a resource unit (RU), denoting a combination of a specific number of consecutive subcarriers (i.e., 1,3,6 or 12) and a number of time-domain slots. The RU represents the minimum element, which can be allocated to a UE for an uplink data transmission. In the case of downlink, the frequency separation is fixed at 15 kHz, and the concept of PRBs is used. A PRB spans 12 subcarriers over 7 OFDM symbols, and a pair of PRBs is the smallest schedulable unit, which is referred to as a single subframe (thus having the total duration of one millisecond). Ten subframes compose a single frame (of 10 ms), and 1024 frames make a hyperframe (10.24 s).

Once powered up, a UE typically starts the procedure of cell search, which is the procedure by which the UE acquires time and frequency synchronization with a cell and identifies it. For this, the UE enables the receiver and searches first for the narrowband primary synchronization signals (NPSS) which are sent by eNB in every 5th subframe of each frame. Then it proceeds with detecting the narrowband secondary synchronization signals (NSSS) which encode the physical cell identity (PCID) and are sent in 9th subframe of each even frame (the transmission of a complete NSSS sequence takes 4 subframes and thus NSSS are repeated every 80 ms). Once finished, the NB-IoT UE proceeds with acquiring the Master Information Block (MIB-NB, [10]) which has a fixed schedule with a periodicity of 640 ms composed of 8 data blocks, each repeated eight times. The elements of MIB (sent in the so-called Narrowband Physical Broadcast Channel - NPBCH) are transmitted in subframe 0 of every single frame. Once decoded, MIB provides the UE with relevant information about the network deployment mode, timings and the scheduling of the first system information (SI) block (SIB1-NB).

The SIB1-NB uses a fixed schedule with a periodicity of 2560 ms in subframe 4 of every other frame in 16 continuous frames [10]. The starting frame of the SIB1-NB depends on the PCID and is derived by the UE from NSSS, while the configuration for repetitions is specified in MIB-NB. The SIB1-NB provides the UE with the information needed to evaluate whether it is allowed to connect the cell and carries the scheduling information of the other SI blocks. To the "required" SI for UE belong MIB-NB, SIB1-NB, SIB2-NB(radio resource configuration), SIBs 3-5-NB(neighbouring cell-related and cell re-selection information), and SIB22-NB(radio resource configuration on non-anchor carriers). The SI messages are sent on Narrow Band Downlink Shared Channel (NPDSCH)[11].

Once possessing all the required SI, the UE may try to establish the connection to the network. For this, it has to execute the special random access (RA) procedure to gain access to a radio channel. Specifically, the UE waits for a scheduled (scheduling specified in SIB2-NB and is periodic with a period ranging between 40 ms and 2.56 s [10]) RA channel (RACH) window, randomly selects one of the preambles (number of which depends on the number of available carriers from 12 to 48) and transmits it. A preamble is sent using single-tone transmission employing frequency hopping between symbol groups. Three different NPRACH preamble formats are currently defined (formats 0 and 1 introduced in Rel. 13 and format 2 added in Rel. 15), featuring the different trade-offs between the on-air time and maximum communication range, which can, potentially, reach 120 km [12]. The basic NPRACH repetition unit consists of four symbol groups for Formats 0 and 1, or 6 symbol groups for Formats 2, with a special relationship between tone frequencies within a repetition unit. Note, that up to three periodic NPRACH windows can be configured in a cell, each associated with a CE level and characterized by the different number of preamble repetitions (ranging from 1 to 128 [10]). The selection of NPRACH to use is made by the device based on its estimation of the radio signal received power (RSRP) from eNB, the network configurations, and the number of previous unsuccessful RA attempts. The NPRACH transmission is sometimes referred to as Message 1 (Msg1) since this is the first message in RA procedure.

After a RACH window, the eNB delivers the scheduling for RA response (RAR or Msg2) in NPDCCH during the Type 2 common search space[11]. The RAR itself is sent in the narrowband physical downlink shared channel (NPDSCH) and allocates the resources and specifies the MCS and the number of repetitions for the next uplink transmission - the radio resource control (RRC) connection request (Msg3) - for the RA preambles it has received. The Msg3 are sent by all the UE which have used the specific RA preamble carrying the unique data identifying the device, i.e., the UE Contention Resolution Identity and which is used to detect the possible collisions. In case of successful Msg3 reception, the eNB replies them with Msg4, i.e., the RRC connection setup, which is also sent in NPDSCH and scheduled through NPDCCH.

Note, that a similar procedure has to be repeated each time an unconnected UE requires to access the radio resources to transmit or receive the data. Note, however, that NB-IoT also supports the contention-free channel access procedure initiated by the eNB, which implies that an eNB dictates a UE it wants to get connected the random access preamble, which no other UE are allowed to use thus ensuring collision avoidance. Following the discussed above RA procedure, the UE and eNB continue exchanging the data sent in NPDSCH in downlink and NPUSCH in uplink having each data transmission scheduled through NPDCCH (the respective procedures are discussed in more details in the following subsection).

Importantly, similarly to LTE, the closing of an active RRC session is handled by the network (i.e., the MME and eNB) and done based on the inactivity timer. In addition to this, the release assistance indication (RAI) procedure has been introduced in Rels. 13 and 14, which allows a UE to signalize the network that the UE has no other data to send and ask for connection release.

Table 6: Uplink and downlink TBS configurations

TBS size (bits) in NPDSCH								
I_{TBS}	Number of subframes (1 ms long)							
0	1	2	3	4	5	6	8	10
1	16	32	56	88	120	152	208	256
2	24	56	88	144	176	208	256	344
3	32	72	144	176	208	256	328	424
4	56	120	208	256	328	408	552	680
5	72	144	224	328	424	504	680	872
6	88	176	256	392	504	600	808	**1032**
7	104	224	328	472	584	**680**	**968**	1224
8	120	256	392	536	680	808	1096	**1352**
9	136	296	456	616	776	936	1256	1544
10	144	328	504	680	872	**1032**	1384	1736
11	*176*	*376*	*584*	*776*	*1000*	*1192*	*1608*	*2024*
12	*208*	*440*	*680*	***904***	*1128*	*1352*	*1800*	*2280*
13	*224*	*488*	*744*	*1032*	*1256*	*1544*	*2024*	*2536*

TBS size (bits) in NPUSCH								
I_{TBS}	N_RU-Number of resource units							
0	1	2	3	4	5	6	8	10
1	16	32	56	88	120	152	208	256
2	24	56	88	144	176	208	256	344
3	32	72	144	176	208	256	328	424
4	56	120	208	256	328	408	552	680
5	72	144	224	328	424	504	680	872
6	88	176	256	392	504	600	808	**1000**
7	104	224	328	472	584	**712**	**1000**	1224
8	120	256	392	536	680	808	1096	**1384**
9	136	296	456	616	776	936	1256	1544
10	144	328	504	680	872	**1000**	1384	1736
11	*176*	*376*	*584*	*776*	*1000*	*1192*	*1608*	*2024*
12	*208*	*440*	*680*	***1000***	*1128*	*1352*	*1800*	*2280*
13	*224*	*488*	*744*	*1032*	*1256*	*1544*	*2024*	*2536*

3.2 NB-IoT KPI

Note, that unless stated otherwise in what follows we imply the NB-IoT operation using frame structure type 1, i.e., the FDD mode. In what follows, we start by discussing the performance of the NB-IoT physical layer and then present the results taking into account the link-layer procedures. Importantly, the results presented below illustrate the NB-IoT performance in different deployment modes and various network and UE configurations, differing with respect to multi-tone support, uplink frequency separation, etc.

3.2.1 Physical layer performance

The peak data rate of NB-IoT at the physical layer is defined by the configurations of the NPDSCH and NPUSCH illustrated in Table 6 .To give an example, for Rel. 13 the 680 bits of downlink data can be sent fastest within three one-millisecond-long subframes, resulting in peak throughput of 226.6 kbit/s. The Rel. 14 has introduced new TBS options (devices implementing these are referred to as class NB2 in contrast to NB1, which denote to devices operating based on Rel. 13), allowing for slightly higher data rates, which can reach 2536bits/10ms=253.6 kbit/s. Even though the TBS allocation tables for NPDSCH and NPUSCH are rather similar, for NPUSCH TBS is allocated in terms of the resource units, duration of which depends on the number of subcarriers and the subcarriers spacing as discussed in [9].

For frame structure type 1 implying FDD operation and NPUSCH format 1, which is used to transfer user data in uplink, in case of 3.75 kHz subcarrier spacing (Δf=3.75 kHz) only single tone transmissions are supported and the maximum I_{TBS} equals 10. Given that the duration of a single resource unit (RU) for Δf=3.75kHz equals 2ms*16slots = 32 ms, the maximum uplink physical layer data rate for Rel. 13 equals 1000 bit/(6*32ms) = 5.208 kbit/s and for Rel. 14 is 1736 bit/(10*32ms) =5.425 kbit/s. The respective values in Table 6 are highlighted with yellow and lime. In the case of 15 kHz subcarrier spacing (Δf=15 kHz) an eNB may assign to the 1,3,6 or 12 sequential tones resulting in the durations of a single resource unit becoming equal to 8, 4, 2, or 1 ms, respectively. Therefore, the maximum uplink throughput (the respective TBS configuration in Table 6 is highlighted with turquoise) for Rel. 13 is 1000 bit/(6*8ms)=20.833 kbit/s, 1000 bit/(4*4ms)=62.5 kbit/s, 1000 bit/(4*2ms)=125 kbit and 1000 bit/(4*1ms)=250 kbit/s for single, 3, 6, and 12 tone transmissions, respectively. In case of Rel. 14 the numbers (TBS configuration highlighted with magenta) are 1736 bits/(10*8ms)=21.275 kbit/s, 2536 bits/(10*4ms)=63.4 kbit/s, 2536 bits/(10*2ms)=126.8 kbit and 2536 bits/(10*1ms)= 253.6 kbit/s for single, 3, 6, and 12 tone transmissions, respectively.

Note, that all the calculations above imply that the minimum number of repetitions configured in the network is one and that the condition of the radio channel between a UE and an eNB is sufficiently good. Otherwise, e.g., for the UE located close to the cell edge or experiencing hard radio signal propagation conditions (e.g., a sensor device in the basement of a building) – a UE may be instructed by the eNB (within the Downlink Control Information (DCI) packet assigning the uplink/downlink resources) to repeat the transmission multiple times. The possible options for downlink (NPDSCH) and uplink (NPUSCH) are listed in Table 7. If repetitions are used, the maximum physical layer data rate decreases proportionally to the increase of the number of repetitions.

However, the discussion above does not account for the protocol-layer features and procedures, which affect directly the throughput experienced by the applications.

3.2.2 Performance of NB-IoT medium access protocol

Note, that for the following discussion, we imply that the RRC session between the UE and an eNB has been already established. First, we consider the uplink transmission scenario, which is illustrated with the respective timings in Fig. 6.

Before sending the actual data in the uplink, a UE has to receive in NPDCCH the DCI of format N0, which carries the information about the resources allocated for NPUSCH,

Table 7: Repetitions in NPDSCH and NPUSCH

Repetitions in NPDSCH																
I_{Rep}	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
N_{Rep}	1	2	4	8	16	32	64	128	192	256	384	512	768	1024	1536	2048
Repetitions in NPUSCH																
I_{Rep}	0	1	2	3	4	5	6	7								
N_{Rep}	1	2	4	8	16	32	64	128								

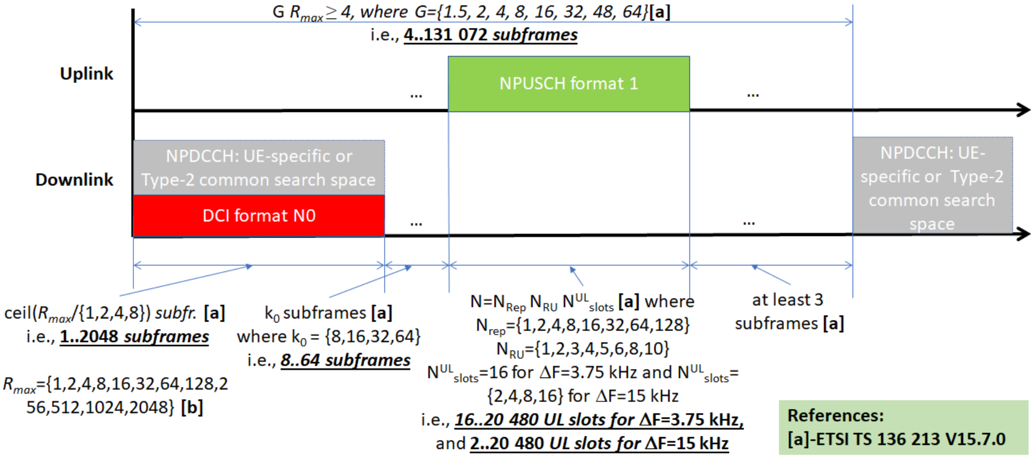
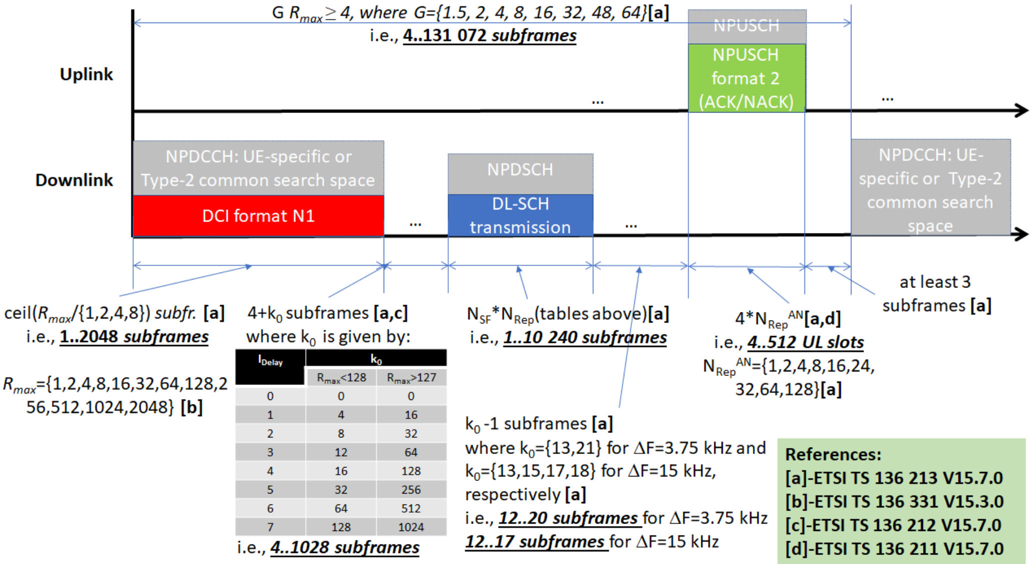


Figure 6: Uplink transmission in NB-IoT and the respective timings

the MSC and the number of repetitions to be used, as well as contains a flag playing a role of a negative acknowledgement and indicating the need of repeating the previous NPUSCH transmission. Depending on the channel conditions the DCI message itself may be repeated multiple (up to 2048) times. Note, that the time windows when an eNB may send a DCI packet, which are referred to in the protocol as "search spaces" are limited and happen periodically, with period depending on the configuration of the network. The minimum period for search space is four subframes (remind, that duration of one subframe equals to 1 ms), but since this leaves not many resources for actual data transfers, in practice the value of the period can be much bigger. After receiving the DCI N0 and before commencing the NPUSCH transmission the UE has to wait for 8 to 64 ms, as specified by the eNB. The NPUSCH transmission itself has been discussed in the previous subsection, and its duration depends on the number of repetitions, the TBS, the number of subcarriers and subcarrier spacing. Following the NPUSCH transmission, the UE waits for an NPDCCH search space, in which the eNB may request (by sending another DCI N0) the repetition of NPUSCH in case it was not received or wants to proceed with the transmission of new data. Note, that the protocol prescribes to have at least a 3-subframe gap between the end of NPUSCH transmission and the start of the next DCI message to this device.

Considering all the implications discussed above, to every NB-IoT UE's uplink transmission, there is an associated signalling overhead of at least 11 subframes for NPDCCH and guard time intervals. Given this, the practical uplink throughput drops up to twice. Specifically, for 12-tone transmission Rel. 13 and Rel. 14 NB-IoT UE can achieve the throughput of 62.5 kbit/s and 115,27 kbit/s, respectively. For single tone and frequency separation of Δf=15 kHz the throughput peaks at 16.67 kbit/s and 18.87 kbit/s for Rel. 13 and Rel. 14, respectively. Finally, for Δf=3.75 kHz due to long uplink resource unit duration the maximum possible throughput does not change significantly, staying at 4.9 kbit/s and 5.22 kbit/s for Rel. 13 and Rel. 14, respectively.

When this comes to the latency, the minimum one in case of uplink data equals the actual on-air time and can be as small as one subframe duration, i.e., 0.5 ms (up to 224-bit TBS with no repetitions in case of a multitone). However, this implies that the eNB has to know exactly when the UE will have data to be transmitted and provide resources to such transmission in advance. This situation is hardly realistic unless the UE traffic is strictly periodic.



I_{Delay}	k_0	
	R_{max}<128	R_{max}>127
0	0	0
1	4	16
2	8	32
3	12	64
4	16	128
5	32	256
6	64	512
7	128	1024

Figure 7: Downlink transmission in NB-IoT and the respective timings

The phases of NB-IoT downlink transmission and their respective durations are illustrated in Fig. 7. Similarly to uplink transmission, the downlink transmission starts with eNB sending a DCI message within the search space time window. Note, that when arranging a data transfer in the downlink, the format of the DCI message differs from the one used for scheduling an uplink. Specifically, the DCI format N1 message, in addition to the information on the resources and MCS to be used for transmitting the downlink, includes the scheduling information for the uplink acknowledgement (ACK) message to be transmitted by the UE following the downlink. As one can see from Fig. 7, the minimum time gap from the end of DCI message to start of downlink data depends both on the number of downlink repetitions and a scheduling delay and ranges from 4 to 1028 downlink subframe length (equal to 1 ms). The time gap between the downlink transmission and the following ACK transmission depends on both the scheduling and the frequency separation used for uplink resource grid, with the minimum duration of 12 downlink subframe lengths. The ACK is sent in NPUSCH using the special uplink frame type (i.e., the NPUSCH format 2). The size of the frame is fixed, and the number of repetitions is one of the network-specific configuration parameters. Similarly to the uplink, the protocol prescribes to have at least a 3-subframe gap between the end of NPUSCH transmission and the start of the next DCI message.

Considering all these, the maximum feasible downlink throughput for inband deployment in Rel. 13 for the case of uplink frequency separation of Δf=3.75 kHz is 21.25 kbit/s, and for Δf=15 kHz is 26.15 kbit/s. For Rel. 14, the respective numbers are 45.68 kbit/s and 54.25 kbit/s, respectively. For standalone deployment for Rel. 13 and uplink frequency separation of Δf=3.75 kHz the maximum downlink throughput is 21.93 kbit/s and for Δf=15 kHz is 27.2 kbit/s. For Rel. 14 these numbers increase to 66.74 kbit/s and 79.25 kbit/s, respectively.

The minimum latency for downlink transmission is defined by the duration of the DCI, the gap between DCI and NPDSCH transmission and the duration of NPDSCH. As this can be seen from Fig. 7, the cumulative duration of these three phases is six subframes – i.e., 6 ms.

Note that the calculations above do not account for the signalling overhead due to the scheduled RACH windows in the uplink, or transmission of synchronization signals and SI in the downlink. These may introduce additional delays, thus reducing the throughput.

3.3 Important mechanisms

Since its initial introduction in Rel. 13 NB-IoT technology has significantly evolved, having a set of new (often optional) functionalities introduced. Since these modifications have the potential to affect the KPIs, in what follows we briefly discuss some of them.

3.3.1 Two HARQ processes

In Rel. 13 the NB-IoT UE were restricted to have only a single HARQ process both with respect to uplink and downlink. As a result, a UE had to wait for the previous block to be acknowledged before sending/receiving the next one. This, due to the signalling overhead and various gaps, has drastically limited the maximum throughput. This limitation has been relaxed in Rel. 14 introducing for the NB2 devices the optional support of two HARQ processes [13, 12]. In essence, the support of the second HARQ process allows a device to send or receive the second block of data even before the first one is acknowledged, thus increasing the maximum data rate. Depending on the configuration, this can bring up to 50% improvement for the throughput.

3.3.2 Early data transmission (EDT)

The need of using RA procedure and establishing an RRC to send the data in uplink brings with it substantial overheads for both the data delivery time and the energy consumption, which become especially notable in case if the amount of data is small. To address this issue, the EDT mechanism has been introduced as a part of Rel. 15. This mechanism allows to integrate up to 1000 bits of data [13] into Msg3 and have them acknowledged in the following message, without even establishing the connection. Note, that for EDT special RACH windows are defined, different from the ones used for conventional RRC connection establishment.

3.3.3 Other mechanisms

Among other notable mechanisms affecting the NB-IoT performance are the enablement of unacknowledged mode RLC (Rel. 15) [13], the allocation of the NPUSCH resources for periodic buffer status report (BSR) transmission for connected UE (Rel. 15) [9, 13], introduction of the RLC unacknowledged mode (UM) [14] (Rel. 15), introduction of optional additional SIB1-NB transmissions to facilitate acquisition of SI needed to connect to the network (Rel. 15), etc.

4 Comparing the two technologies

Both LoRaWAN and NB-IoT have enormous potential for the development of many different IoT applications. Smart cities and precision agriculture are among the domains that can benefit more from the adoption of these technologies. Indeed, in most cases, the IoT applications from such domains do not demand high throughput or low latency; their requirements are compatible with the performance offered by either of the two technologies.

However, the comparison between LoRaWAN and NB-IoT technologies, and the identification of their actual strengths and limitations, must take into account regulatory issues and business models, besides technical aspects.

From the regulatory viewpoint, there is a clear difference between the two technologies. NB-IoT can be deployed over existing 4G systems. Only Mobile Network Operators (MNOs) who have a 4G license can offer NB-IOT services. This is both an advantage and a drawback. The good side is that for MNOs, deploying the network is just a technical and investment issue. In many countries all over the world, they have already deployed NB-IoT plug-ins, and there is no other issue in exploiting it from the user viewpoint. On the opposite, LoRaWAN operates on a license-exempt ISM band which is regulated differently from country to country. In Europe, the document providing guidelines for the use of LoRaWAN (and other) technologies is CEPT Recommendation number 70 03. Different national authorities interpret it in various ways. To date, in Italy, it is still not possible to operate a LoRaWAN network on the 868 MHz band, based on current limitations posed by the Ministry for Economic Development. In the rest of Europe, the same frequency band is used by many operators delivering IoT services since some time now. Assuming that this will be solved soon also in Italy, from the user viewpoint this frequency band poses constraints: nodes can not go beyond the one per cent duty cycle boundary. This means that users have to ensure that their devices do not generate data too frequently. In most applications, this is not an issue, but potentially such limitation brings complexity on the shoulders of the user.

The business model behind the two technologies is totally different. As mentioned, NB-IoT services can only be offered by MNOs. As long as they deploy the network, it is publicly available (upon payment of a subscriber fee). On the opposite, anyone in principle could offer LoRaWAN coverage; private deployments may be useful for particular applications (especially in remote locations, which are not attractive to MNOs). LoRaWAN networks might be available for free in some areas, as it happens, e.g., with the Things Network - a community of open source LoRaWAN gateway owners. Large LoRaWAN networks are deployed in Italy by some companies for smart city applications. As long as they will be allowed to operate commercially, they will offer subscription-based services in large cities like Milano and others.

Finally, the technical side. Numerous differences characterise the two technologies.

- Latencies. The two systems offer comparable performance in the uplink, with latencies of up to about two seconds. In the downlink, however (e.g. for sending commands to actuators), the two options are quite different. NB-IoT has smaller latency than in the uplink, while the LoRaWAN protocol requires transmission on the uplink first, to piggyback packets in the downlink acknowledgements; therefore, downlink latency can be very large, depending on the uplink transmission rate of the device.
- Throughput. As mentioned above, NB-IoT can offer throughput in the order of some (or tens of) kbit/s. On the opposite, a LoRaWAN device has maximum throughout severely limited by the duty cycle constraint and typically close to few tens of bit/s.
- Security and identification. LoRaWAN with ABP is insecure. On the contrary, NB-IoT has advanced security protocols in place.
- Roaming. Sub-GHz ISM bands (normally used by MNOs for NB-IOT wide coverage) are not uniform around the globe, which complicates LoRaWAN trans-ocean roaming. NB-IoT terminals supporting multiple bands can handle this. Recently, the intra-continental roaming solutions for LoRaWAN (allowing to roam between networks deployed in the same bands) have been delivered. However, their widespread adoption is still underway.
- Energy consumption. In addition to the overall consumption, the potential problem for NB-IoT is high peak consumption and the need for a lot of energy during the initial connection with the network. This may make it hard to enable energy-harvesting powered NB-IoT devices. Overall, for non-frequent transmission of small amounts of data, battery duration of a LoRaWAN-powered sensor system can be one order of magnitude larger than for NB-IoT.
- IP support. NB-IoT supports IP, and many off-the-shelf transceivers implement IP-based protocols like TCP/UDP, FTP, HTTP, CoAP, MQTT. This enables seamless integration between NB-IoT and the Internet; whist LoRaWAN requires some form of adaptation layer (most often based on the NS) in between.
- Handover. LoRaWAN networks do not implement any sort of handover mechanism. NB-IoT has to handle it, though this requires additional signalling.

In conclusion, the two technologies differ in many aspects, and both have strengths and weaknesses. Depending on the specific application, the best solution can be identified based on the above considerations. While this is true for all countries of Europe, Italy still suffers from the lack of a vision: while NB-IoT is available, LoRaWAN networks can not be operated commercially yet. This is affecting the development of the digital agenda of the country.

References

[1] C. Gomez, J. Oller, and J. Paradells. *Overview and evaluation of Bluetooth Low Energy: An emerging low-power wireless technology.* Sensors, vol. 12, no. 9, pp. 11 734–11 753, 2012.

[2] M. Siekkinen, M. Hiienkari, J. K. Nurminen, and J. Nieminen. *How Low Energy is bluetooth low energy? Comparative measurements with Zigbee/802.15.4.* Wireless Communications and Networking Conference Workshops (WCNCW), 2012 IEEE. IEEE, 2012, pp. 232–237.

[3] M. Research. *The need for low cost, high reach, wide area connectivity for the internet of things: A mobile network operator's perspective.* White paper, 2014.

[4] F. Adelantado, X. Vilajosana, P. Tuset-Peiro, B. Martinez, J. Melia-Segui and T. Watteyne. *Understanding the Limits of LoRaWAN.* in IEEE Communications Magazine, vol. 55, no. 9, pp. 34-40, Sept. 2017.

[5] Claire Goursaud, Jean-Marie Gorce. *Dedicated networks for IoT: PHY / MAC state of the art and challenges.* EAI endorsed transactions on Internet of Things, 2015.

[6] Olof Liberg, Marten Sundberg, Eric Wang Johan, Bergman Joachim Sachs. *Cellular Internet of Things. Technologies, Standards, and Performance.* Elsevier, September 2017.

[7] 3GPP TS 36.300 Group Radio Access Network. *Evolved Universal Terrestrial Radio Access (E-UTRA) and Evolved Universal Terrestrial Radio Access Network (E-UTRAN). Overall description.* 3GPP, December 2018.

[8] LoRa Alliance. *LoRaWAN Specification (V1.0.2)*, 2016.

[9] ETSI TS 136 211 V15.7.0

[10] ETSI TS 136 331 V15.3.0

[11] 3GPP TS 36.523-3 V13.1.0

[12] Olof Liberg, Marten Sundberg, Eric Wang Johan, Bergman Joachim Sachs. *Cellular Internet of Things. Technologies, Standards, and Performance.* Elsevier, December 2019.

[13] GSMA, NB-IoT Deployment Guide to Basic Feature set Requirement, June 2019.

[14] ETSI TR 121 915 V15.0.0

[15] Quectel, BC95 Hardware Design, https://www.quectel.com

[16] A. Weissberger, NB-IoT, https://techblog.comsoc.org/category/nb-iot/

[17] Semtech Corporation *LoRaWAN – simple rate adaptation recommended algorithm* Revision 1.0, October 2016

[18] Gianni Pasolini, Chiara Buratti, Luca Feltrin, Flavio Zabini, Roberto Verdone, Oreste Andrisano, Cristina De Castro, *Smart City Pilot Project Using LoRa* European Wireless 2018; 24th European Wireless Conference, Catania, Italy, 2018, pp. 1-6.

[19] L. Feltrin, M. Condoluci, T. Mahmoodi, M. Dohler and R. Verdone, *NB-IoT: Performance Estimation and Optimal Configuration* European Wireless 2018; 24th European Wireless Conference, Catania, Italy, 2018, pp. 1-6.

[20] L. Feltrin, C. Buratti, E. Vinciarelli, R. De Bonis and R. Verdone, *LoRaWAN: Evaluation of Link- and System-Level Performance* IEEE Internet of Things Journal, vol. 5, no. 3, pp. 2249-2258, June 2018.

[21] LoRa Alliance *LoRaWAN 1.1 Regional Parameters*, 2017