Narrowband IoT Device Energy Consumption Characterization and Optimizations

Galini Tsoukaneri
University of Edinburgh
g.tsoukaneri@sms.ed.ac.uk

Franscisco Garcia
Keysight Technologies
frankie_garcia@keysight.com

Mahesh K. Marina
University of Edinburgh
mahesh@ed.ac.uk

Abstract

Narrowband IoT (NB-IoT) is a new and attractive low power wide area cellular technology for low-capability and low-cost IoT devices, that is starting to see real-world deployments. NB-IoT devices are expected to operate unattended, potentially in inaccessible and signal-challenged locations, for at least 10 years on a single battery charge, making the NB-IoT device energy consumption significantly important. Despite the importance of their function, the communication protocols have largely been copied from older generations of cellular networks to preserve interoperability, without considering their specific characteristics and needs. In this paper, we perform a detailed energy consumption analysis for NB-IoT devices, that we use as a basis to develop an energy consumption model for realistic energy consumption assessment. Finally, we take the insights from our analysis and propose optimizations to significantly reduce the energy consumption of NB-IoT devices in different traffic conditions. These optimizations are also complementary to current 3GPP optimizations towards the 10-year battery goal, and assess their performance.

1 Introduction

Narrowband IoT (NB-IoT) [42] is a new cellular network technology, which is intended to cater to large numbers of stationary, low-power IoT devices deployed over a wide area [23], such as sensors and meters, placed in signal-challenged locations (e.g. basements, bridge foundations etc.) Although other proposed technologies (e.g., LoRa, SigFox) target similar devices, the licensed nature of NB-IoT that allows for limited interference, the open standards it implements and the improved coverage [43] make NB-IoT technology a major candidate for future IoT support, with several network operators already deploying commercial NB-IoT networks.

Due to their nature, physical access to these devices is likely to be difficult, which necessitates that they have the ability to operate for 10 or more years on a single battery charge [43]. However, these devices are expected to cost less than $5 [23], which precludes the usage of high capacity batteries. As such, it is vitally important that NB-IoT network protocols are designed with device energy efficiency as a central goal.

In this paper we show that, despite recent enhancements (e.g. extended Discontinuous Reception (eDRX), Power Saving Mode (PSM)), the current NB-IoT protocols are not efficient enough to achieve the desired battery life goal. Furthermore, we argue that many of the inefficiencies of the current protocols stem from the fact that they have been directly inherited by the LTE and 5G designs, which traditionally focused on Human-Type-Communication (HTC) devices (phones, tablets), without much concern for energy efficiency, as they can be recharged often. Additionally, they have been optimized to cater to HTC traffic patterns and use cases (long connections uniformly spread over time, mainly downlink traffic). In contrast, NB-IoT devices typically exhibit very short connections in frequent, periodic intervals [21], and existing procedures can incur a disproportionate energy overhead in relation to the actual data communicated.

This paper makes three key contributions:

1. We perform a thorough experimental measurement of the power consumption of each individual operation that a NB-IoT device performs under normal use (e.g., Random Access, Attach, encryption), using three different commercial NB-IoT devices. These operation-specific measurements offer significantly greater insight than prior works that only measure the total power consumption [34], or just the data exchange energy cost [35, 51], as they allow us to unearth potentially inefficient areas of the NB-IoT protocols, and can guide us towards effective optimizations. They also show a large deviation from the energy consumption assumptions published by 3GPP [3] as well as other works [15], which can greatly affect studies that rely on them (e.g., [34]). *To the best of our knowledge, this is the first work for NB-IoT devices that measures each operation in isolation.*
2. Building on the above characterization, we present an

International Conference on Embedded Wireless Systems and Networks (EWSN) 2020
17–19 February, Lyon, France
ISBN: 978-0-9949886-4-5

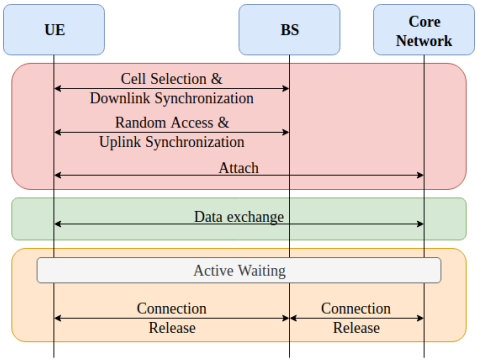

Figure 1. Communication process in NB-IoT.

NB-IoT energy consumption model, which we use to simulate the battery life of a device under realistic traffic conditions [21] and factoring in collisions and signal degradation. This gives us an estimate of the battery capacity requirements to meet the 10 year goal for different coverage scenarios, and we find that it is far from realizable with current practices and previously proposed approaches, given a $5 cost constraint. We also use our model to compare against prior work that do not take all operations into account, and assess the overall energy consumption difference. We show that operations ignored by prior works (e.g., [15]) can have a significant energy cost, and it is imperative they are considered and optimized to lower the device energy consumption.

3. We propose two novel mechanisms that exploit the stationary and periodic nature of NB-IoT devices, and substantially reduce device energy consumption (up to 37.8%), while at the same time free up network resources. Furthermore, we discuss a set of best practices under the existing protocols, that device vendors and network operators should consider in order to maximize battery life.

The rest of the paper is organized as follows. Section 2 provides the essential background. Section 3 presents detailed energy consumption measurements with three commercial NB-IoT devices. In section 4 we present our energy consumption model and a simulation-based study to assess the battery requirements of NB-IoT devices. Section 5 presents our energy optimization mechanisms along with best practices, and evaluates their energy performance against baseline approaches. Section 6 discusses the related work. We conclude in section 7.

2 Background

2.1 Communication Process

In NB-IoT (like in LTE), whenever a device wishes to transmit new data and does not have an active connection with the network, it needs to follow the connection establishment procedure (Fig. 1(a)). This procedure is made up from the the *Random Access (RA)* and the *Attach* processes.

To establish a connection, the device begins to scan the air interface for a suitable network to connect to, and synchronizes with it in the downlink direction (i.e. from the network to the device). Afterwards, the *Random Access (RA)* process is followed to gain synchronization in the uplink (i.e. from the device to the network) and request resources for transmission, which begins with the preamble transmission by the device. For each identified preamble, the *Base Station (BS)* replies with a *Random Access Response (RAR)* message, that contains an uplink grant for the next transmission, and the device-specific *Timing Advance (TA)* value to control subsequent uplink transmissions (i.e. how much time in advance the device needs to start transmitting, so that the BS receives the signal at the correct time). Next, the BS resolves possible collisions that might have happened at the preamble transmission stage, and selects the devices that are allowed to proceed with the connection establishment process. The devices then need to follow the *Attach* process, to establish a connection with the core network, generate a security context, and set up data and signalling radio bearers (DRBs and SRBs, respectively). At the end of the Attach process the device can exchange data at will.

To increase the probability of signal reception in signal-challenged locations, NB-IoT specifies three different coverage levels, *normal*, *robust* and *extreme* that use different number of repetitions (up to 128 and 2048 repetitions for the uplink and downlink respectively), which can either be decoded separately, or they can be combined to further increase the reception probability.

After the data transmission, devices remain connected to the network for a specific amount of time (called herein as *active waiting*). This feature was initially designed for HTC devices, in order to avoid having to repeat the connection establishment process when the devices exchange data within short periods of time. The duration of the active waiting is defined by the *inactivity timer (IT)* [8], which is network operator specific (10-50 seconds in most commercial networks).

Upon expiration of the IT, the network releases the connection, and the device switches to an idle (*light sleep*) state, with lower energy consumption, requiring the device to repeat the connection establishment process when it wishes to transmit data again. If data arrives while the device is in the idle state, the network uses the paging procedure to notify the device. The frequency that the device checks for paging messages is defined by its *Discontinuous Reception (DRX)* cycle[1] [11], whose length is negotiated between the device and the BS.

Additionally, the *Power Saving Mode (PSM)* [11] feature can be used, that allows devices to enter a *deep sleep* state, and operate with power consumption close to power-off. The PSM has longer cycles than DRX, but the device is not reachable by the network during a PSM cycle. At the end of it, the device is required to follow the *Tracking Area Update (TAU)* process, to inform the network about its presence, and its short availability to receive network-originated data.

[1]Here we refer to both DRX and extended DRX (eDRX), as their functionality is the same. Also note that the DRX cycle can also be used during the active waiting period (referred to as *connected mode DRX (C-DRX)*, to further reduce the energy consumption.

2.2 3GPP Optimizations for NB-IoT Device Energy Savings

To better accommodate battery constrained NB-IoT devices, 3GPP has specified both *Control Plane (CP)* and *User Plane (UP)* optimizations [11]. In the CP optimization, the devices encapsulate application data in control messages and transmit them over their SRBs, avoiding the need to setup a new DRB at each connection, essentially skipping the Attach process. Although the support for the CP optimization is mandatory, the use of the SRBs limits the size of the data that can be encapsulated, and as such, it can only be used for small data transmissions. Additionally, the QoS of the data transmission is upper bounded by the QoS that can be supported by the SRBs, which might be unsatisfactory to the device application. Finally, the transmission of application data in control messages does not allow for encryption of the transmitted data. Therefore, to accommodate larger packet transmissions, better QoS and secure transmissions, 3GPP also defines UP optimization according to which a device's pre-established connection can be suspended and resumed with fewer control messages [10]. Essentially, the network retains control information about the pre-established connection, which can be used when the device wishes to transmit new data, without the need to setup new SRBs. However, a new DRB still needs to be set up for the transmission of the application data, as this is not retained from the device's previous connection.

2.3 Security Framework in Cellular Networks

The security framework used in NB-IoT is inherited from 4G and 5G networks [14], and provides processes for mutual authentication between the device and the network, and establishment of the device's *security context (SC)*, to be used in subsequent communications [14] for data integrity and confidentiality. The SC is derived from a unique, user-specific root key K that is stored only on the device, and the core network entity responsible for user authentication. As there exists a one-to-one mapping between the device's globally unique *International Mobile Subscriber Identity (IMSI)*, and its key K, the robustness of the security framework relies on the assumption that K will not be disclosed to or stolen by unauthorized parties.

Each time a device establishes a new connection (Sec. 2.1), the network checks for its SC if any (i.e. the device has previously connected, and its SC has not expired). If none exists, the network requests the device's IMSI to identify it, and retrieve its K key. Next, the device and the network mutually authenticate each other, and setup a hierarchy of keys and encryption algorithms for subsequent communications. The network can request a device re-authentication as often as it wishes [14], even if the device is already connected to the network. In some cases, the network is even obliged to delete the device's SC, and request a re-authentication at the device's next connection (e.g. during the TAU process).

2.4 Identity Privacy Mechanism

The IMSI is an important identifier that allows a device to access the network when no other information is available. As the IMSI is transmitted unencrypted, it is susceptible to spoofing. Having gained access to the IMSI, a malicious actor can launch a range of attacks, such as user-targeted DoS attacks [26, 28, 29], and leaking of user location history [18, 24, 46]. Therefore, 3GPP recently proposed the *Identity Privacy Mechanism (IPM)* [14] to protect against leaking of the IMSI to malicious eavesdroppers.

Specifically, the IPM is based on a public-key infrastructure (PKI), where the IMSI is sent encrypted, using a combination of public key generation with Diffie-Hellman and symmetric encryption. To transmit its IMSI, a device first generates a pair of ephemeral private-public keys using the home network's public key, which is permanently stored on the device, similarly to the root key K, and any algorithm mutually supported by both the network and the device (e.g. RSA [27]). The device then uses the Diffie-Hellman algorithm to generate a second ephemeral key using the home network's public key and its previously generated ephemeral private key. This second ephemeral key is then used to encrypt the IMSI with a symmetric algorithm, such as AES. To avoid replay attacks, 3GPP specifies that new keys must be generated for each IMSI transmission.

3 Device Energy Consumption Measurements

3.1 Experimental Setup

Initially, we measure the energy consumption of three popular NB-IoT development kits (GPy from Pycom (device A) [40], BC95 from Quectel (device B) [41] and SARA-N2 from Sodaq (device C) [48]), henceforth referred to as devices A, B and C. Note that the devices B and C require a separate Micro-Controller Unit (MCU) to operate. For this, we used an Arduino Uno board [16], as, at the time of writing, it is the one with the lowest power consumption among the various Arduino MCUs. Device A was used with the MCU unit of its vendor [39]. In order to get accurate power consumption measurements for the various operations, we switched off all LEDs at boot time (on all devices), and any WiFi/Bluetooth features if available.

To measure the energy consumption of the different network processes (RA, Attach etc.) we used the E7515A UXM Wireless Test Set [31] by Keysight Technologies, which implements a fully compliant NB-IoT base station (BS). In terms of the NB-IoT configuration, the UXM box was set to use an in-band deployment over a 10MHz channel, with 15KHz subcarrier spacing and QPSK/BPSK modulation in the downlink/uplink respectively. We also assumed a normal coverage level. For operations outwith the actual communication (e.g., message generation, key generation, encryption etc.), we used a Power Monitor (FTA22J) [38] from Monsoon Solutions. The devices were powered through the Power Monitor, using a USB cable which was configured to only supply power.

In these experiments we measure the energy consumption while the devices are in any of the three performance states: (i) light sleep, (ii) deep sleep and (iii) working. The light and deep sleep states correspond to the idle and PSM states of 3GPP [12] respectively (Sec. 2.1), and reflect the states when the device has limited or almost no energy consumption. The working state is the state during which the device generates data and communicates with the network. For this state, we

separately measure the *(a)* RA process, *(b)* Attach process (for cases when CP optimization is not used), *(c)* exchange of application data (including any required scheduling requests, reception of control data for ACK, encryption/decryption) (for cases when CP optimization is not used), *(d)* IMSI encryption (Sec. 2.4)), and *(e)* active waiting with C-DRX of 10 and 30-second IT. For these measurements, we assume a NB-IoT specific traffic pattern, where devices send 200 bytes of application data and receive a 140 bytes acknowledgement every 5 minutes. We consider a broad range of traffic patterns in subsequent sections.

Strong encryption mechanisms can be very expensive for NB-IoT devices in terms of energy consumption, so the choice of security procedures can make a significant difference on their battery life. For symmetric encryption we measured the power consumption of the EEA1/EIA1, EEA2/EIA2 and EEA3/EIA3 algorithms which are recommended by 3GPP [7], and are based on the SNOW3G, AES, and ZUC algorithms respectively. For the IMSI encryption, we follow the approach of [26] and use the Diffie-Hellman key exchange algorithm implementing a Curve25519 elliptic curve. To generate asymmetric keys, we used public implementations of the popular RSA and El-Gamal algorithms [1, 2] with 1024-bit keys for both. Each data point in our results is the average of 10 different runs.

3.2 Results

We first measure the average power consumption during the three operational states (Fig. 2), assuming a complete connection cycle (i.e. connection establishment, data exchange, active waiting, light sleep and deep sleep). Overall, all three devices have similar power usage in the working state, with major differences occurring in the deep sleep state. As expected, the energy consumption in the working state can be orders of magnitude greater than the other two states. Perhaps surprisingly, although the light sleep state is more energy-efficient than the working state, it still uses 3 orders of magnitude more power than the deep sleep state. This indicates that in cases where latency for network originated data is not critical, there would be substantial gains if the devices switched to the deep sleep state earlier (Sec. 5).

We then examine the energy consumption of the operations related to the security framework. 3GPP allows for alternatives for both symmetric and asymmetric encryption [14], and as such, we examine the energy consumption of the three recommended symmetric encryption algorithms (Fig. 3), and two popular asymmetric encryption algorithms (El-Gamal, RSA) (Fig. 4) in isolation (Sec. 2). For reference, we also measured the energy required for the Diffie-Hellman protocol. We observe that among the symmetric algorithms, EEA2/EIA2 is the most energy efficient, followed closely by EEA3/EIA3. When comparing the asymmetric algorithms, El-Gamal is more efficient for decryption and key generation but is significantly more expensive for encryption compared to RSA. As only the key generation part of these algorithms (Sec. 2.4) is used, El-Gamal emerges as the most efficient choice.

We further examine the energy consumption of the various operations in the working state (Fig. 5) to get insight on potential areas for improvement. For this experiment,

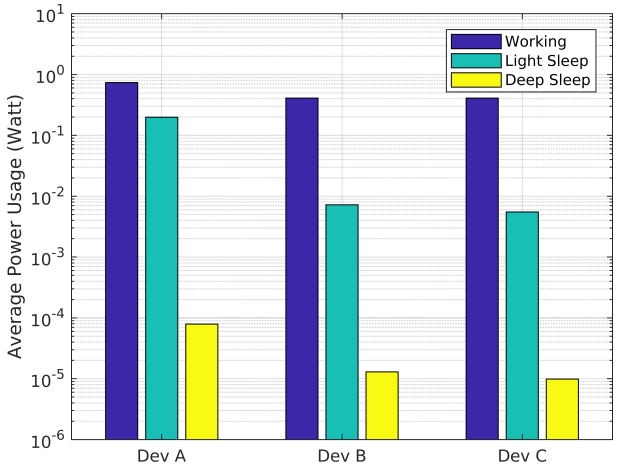

Figure 2. Power consumption (in Watts) with different devices in different operational states.

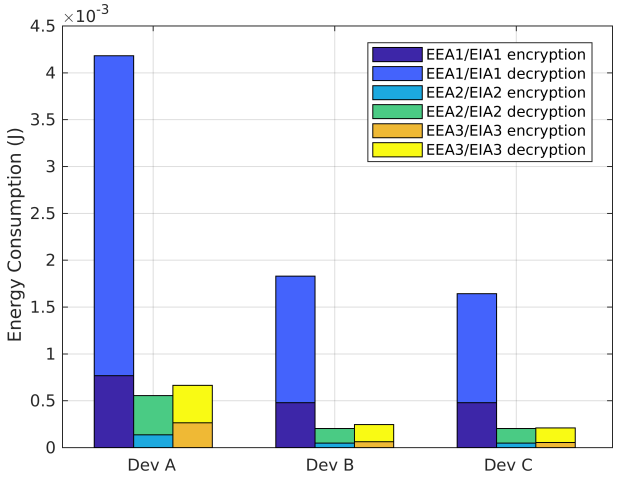

Figure 3. Energy consumption of symmetric encryption algorithms.

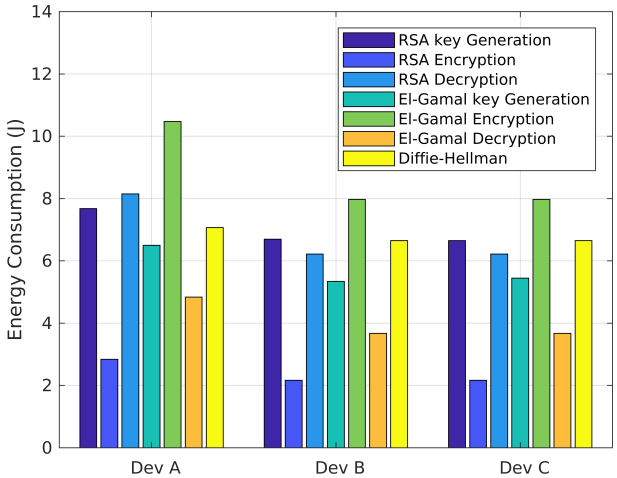

Figure 4. Energy consumption of asymmetric encryption algorithms that can be used for the IMSI encryption.

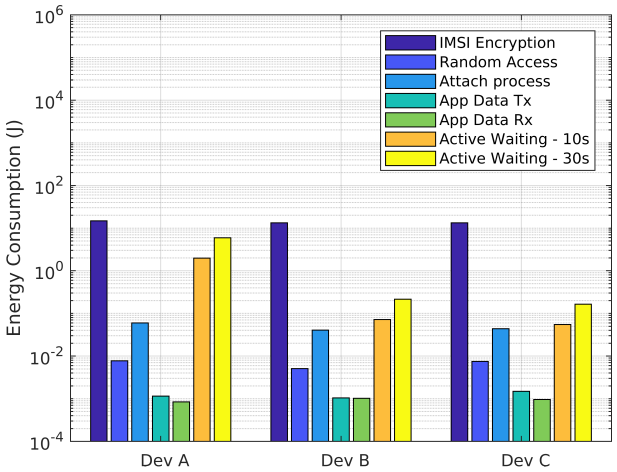

Figure 5. Energy consumption for various operations in the working state. For active waiting, we show the energy consumption with different values for IT: 10 **and** 30 **seconds.**

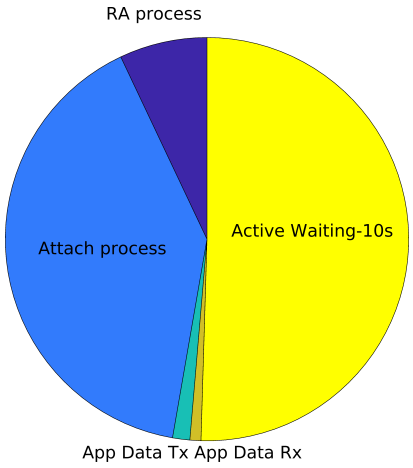

Figure 6. Pie chart showing proportion of time spent for different operations in the working state (excluding IMSI encryption).

we used the EEA2/EIA2 algorithm for symmetric encryption and integrity protection and the El-Gamal algorithm for the IMSI encryption (Sec. 2.3), as these yielded the lowest energy cost. Notably, our results show that the energy consumption values on real devices are considerably greater compared to the values considered by 3GPP for NB-IoT [3], based on which the 10 year life goal was set.

When looking at each operation individually, we see that the energy consumption for the actual data transmission and reception is orders of magnitude lower than that for operations like the RA, Attach and Active Waiting. This is also reflected in the pie chart (Fig. 6) showing the proportion of time taken for different operations in the working state with a $10s$ IT. Please not that the time proportion is the same for all examined devices. In this experiment we have excluded the IMSI encryption as it is very expensive for NB-IoT devices and can dominate the time spent in the working state. Crucially, these results show that a holistic and fine-grained view of NB-IoT device energy consumption characterization leads to significantly different battery life estimates compared to prior works [34, 15]. In section 5, we account for the significance of the various operations towards optimizing the overall device energy consumption.

4 Battery Life Estimation Analysis

In this section we build on the energy consumption measurements from the previous section towards a realistic battery life estimation of NB-IoT devices. To this end, we first model the energy consumption for a connection cycle. Then, we use our energy model and realistic traffic patterns [32] to simulate the overall energy consumption. This allows us to take into account factors such as collisions during the RA process, and gives us a realistic estimate of the energy consumption outside lab conditions.

4.1 Energy Consumption Model

We define P as the time period between two successive instances that a device wished to transmit data. Then, the total energy consumption for a single period P is $E_p = E_w + E_l + E_d$, where E_w, E_l and E_d is the energy spent in the working, light sleep and deep sleep states, respectively.

The energy consumption in the working state E_w is equal to $E_w = E_{RA} + E_{At} + E_{app} + E_{AW}$. Here E_{RA} is the energy consumption for the completion of the RA process. This can in turn be defined as: $E_{RA} = \varepsilon_{RA} * \overline{R}$, where ε_{RA} represents the energy consumption of an RA process without collisions and $\overline{R}$ means the average number of connection attempts required to complete the RA process. E_{At} is the average energy consumption required for completion of an Attach process (when the CP optimization is not used), which is specified to be at most 15 seconds [11]. E_{app} is the total energy consumed for the transmission and reception of application data, including the data generation, scheduling requests, generation and appendage of the required headers, and transmission/reception including all repetitions. Finally, E_{AW} is the energy consumed in active waiting.

E_l is equal to $E_l = t_l U_l$, where U_l is the average power usage in the light sleep state, and t_l is the time spent in the light sleep state during a period P. Similarly, E_d equals $E_d = (P - t_l - t_w) * U_d + E_{TAU}$, where U_d is the average power usage in the deep sleep, and t_w is the time spent in the working state during a period P. E_{TAU} is the energy consumed during the TAU process that is executed at the end of the PSM cycle. It can be defined as $E_{TAU} = E_{RA} + 2\varepsilon_{Tx} + \varepsilon_{Rx}$ where ε_{Tx} and ε_{Rx} is the energy consumption required for the transmission and reception of the messages for the TAU. The E_{RA} is also included, as devices must first go through the RA process.

Given the above, the overall energy consumption of a device during the course of a day is the total of the energy consumption of all periods in that day, and the energy consumption of the IMSI encryption. Please note that the latter depends on the deletion frequency of the SC.

4.2 Simulation Setup

In these experiments we use a custom simulator written in Matlab, which follows the current 3GPP specifications in detail, and implements the same procedures to the ones of

Variable	Value
UE Tx power on NPUSCH	Based on 16.2.1.1.1 [9])
Num of subcarriers	12
UE Tx power on NPRACH	Based on 16.3.1 [9]
Time spent in NPUSCH	0.93 ms
Time spent in NPDCCH	0.22 ms
C-DRX cycle	2.56 s
I-DRX cycle	5.24 s
PSM Duration	$dev\_IAT/5$ or random PSM
Subcarrier spacing (KHz)	15
Coverage Levels	Normal / Robust / Extreme
NPDCCH repetitions	1 / 1024 / 2048
NPDSCH repetitions	1 / 1024 / 2048
NPUSCH repetitions	2 / 64 / 128
NPRACH repetitions	2
NPDCCH periodicity	$T_{NPDCCH} = R_{MAX} * G$ [9] $R_{MAX} = 1$ $G = 32$
RACH periodicity	40 ms [13]

Table 1. Simulation configuration parameters

real deployments. For any optimizations, we altered the current procedures to incur the minimum changes possible. Our simulations are based on our energy model (Sec. 4.1), and follow realistic traffic patterns for a NB-IoT cell in a dense urban environment [21, 45]. Specifically, we simulate 50000 devices uniformly distributed in a 500m radius cell, which is considered a typical-sized urban cell. We assume that 80% of the devices are periodic with variable inter-arrival times (IATs) (i.e., application periodicities), ranging from 5 minutes to 24 hours, while the remaining 20% devices are event-driven. In each connection, devices transmit a single application packet of 200 bytes, and receive an application ACK of 140 bytes (based on [21]). We also assume that devices employ the CP optimization (Sec. 2), unless explicitly stated otherwise. In terms of network configuration our parameters are summarized in table 1.

In all of our experiments we used the most energy-efficient configuration, to set up a lower limit for the required battery capacity. We use the EEA3/EIA3 for symmetric encryption, and the El-Gamal for asymmetric key generation (Sec. 3). We assume a 10-seconds IT for the active waiting, during which devices apply C-DRX (Sec. 2) with 2.56-seconds cycle. After the expiration of the IT, devices switch to the light sleep state (idle) for 2 minutes and apply I-DRX. We also assume that the PSM feature is enabled. For periodic devices, we assume that the PSM duration is $1/5$ of their IAT, for all IAT values with the exception of 5-min (for which PSM feature is not applicable). For event-driven devices, we assume a randomly generated PSM duration of up to 120 minutes. For all devices, we setup a new security context (SC) on their initial network attach using an encrypted IMSI (Sec. 2.3) and assume that is never deleted. Note that we do not include possible SC renewals, that would require the generation of asymmetric keys for the IMSI encryption, as this feature is currently optional for NB-IoT. Including that feature would only add to the battery drainage so our analysis should be seen to provide an upper limit on the battery life.

4.3 Results

Fig. 7 shows the estimated energy consumption for various most common IoT IATs [21, 45], and different coverage levels. Our results deviate from the battery cost or lifetime targeted by 3GPP. If we consider the most efficient device *C*, our results also show that for 5-minute IAT, the minimum capacity needed is $\approx$ 88 Wh, while a 30-min IAT would require a battery of $\approx$ 30 Wh capacity. These higher battery capacity requirements come at the expense of increased cost, and far exceed the overall cost goal of $5 per device that was set by 3GPP [4] – at the time of writing we found the cost of the cheapest batteries with $\approx$ 90Wh and $\approx$ 30Wh capacities to be $\approx$$45 and $\approx$$30, respectively. To meet the specified cost and battery lifetime requirements, only devices with much shorter IATs can be accommodated (3-h or higher IAT for the above example).

Overall, shorter IATs require frequent connections that consume significantly more energy as the devices need to frequently establish a connection (Sec. 2.1) and get in the working state, which has a significant energy cost (Sec. 3). On the other hand, as the IAT increases, the devices spend increasingly more time in the more efficient deep sleep state (Fig. 8) with a significantly lower energy consumption. When in deep sleep state, a device mainly consumes energy to perform the TAU process at the end of each PSM cycle. As such it is imperative that this is taken into account in the energy consumption model. In Fig. 9, we see that the model of [15] significantly underestimates the energy consumption of the device, for example leading to an underestimation of energy consumption by around 45% for 24-h IAT and normal coverage level. *These results indicate that optimizing the TAU process would have the greatest impact in the battery life of devices with long IATs. Conversely, for devices with shorter IATs, minimizing the time spent in the working state and switching faster to the more efficient light sleep state would bring a noticeable difference.*

5 Energy Optimization Mechanisms

Our device measurements (Sec. 3) show that the active waiting, IMSI encryption and Attach/RA processes are most energy consuming operations in the working state. Therefore, in this section we propose a set of novel mechanisms to optimize their energy consumption to increase the battery life expectancy. Furthermore, we present a guideline of best practices for device manufacturers and network operators, and quantitatively measure their impact on the overall energy consumption. For fair comparison, we assume a baseline that corresponds to current and unoptimized 4G/5G procedures. Please note that we only consider the device *C* and normal coverage level for better clarity, however, the same trends can be observed for the other coverage levels.

5.1 Reducing RA Connections and Security Context Renewal

As the RA process and IMSI encryption are two of the most energy consuming operations, reducing their frequency would bring significant benefit to the battery life expectancy

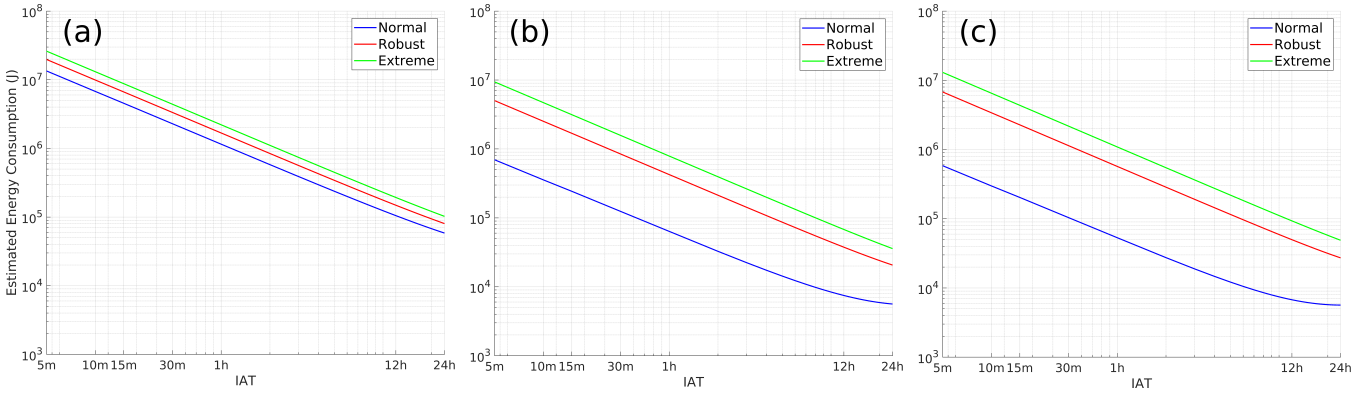

Figure 7. Energy consumption in 10 years for different IATs and coverage levels for devices *A*, *B* and *C*. Please not that log-scale is used in both axes.

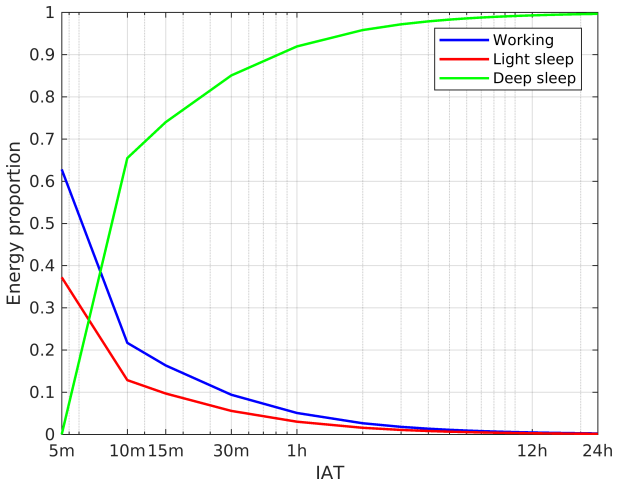

Figure 8. Proportion of energy spent on each state as a function of the IAT.

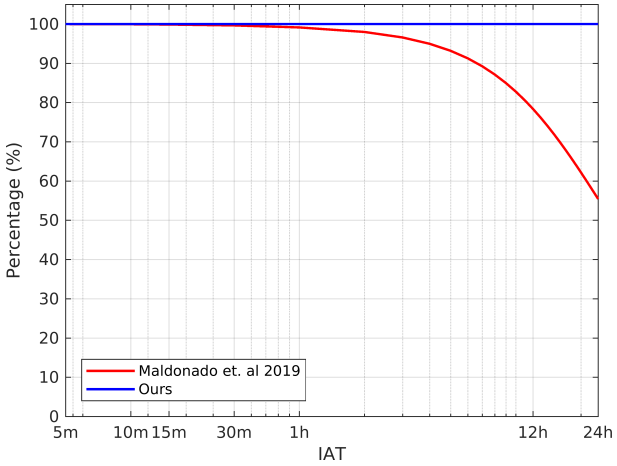

Figure 9. Difference (%) of the energy consumption estimated by the model from [15] for different IATs relative to our model assuming normal coverage level and the use of CP optimization. As the IAT increases, the energy for the TAU process dominates total energy consumption and increases the estimation error with the model from [15].

of NB-IoT devices. Towards this end, we propose a mechanism that exploits the periodicity of IoT devices, to reduce the frequency of these operations, without limiting functionality nor breaking backwards compatibility for event-driven devices. Since the majority of NB-IoT devices are expected to fall into that category [25], this would have a major impact in practice.

First, we propose reducing the number of RA processes by eliminating the TAU process at the end of a PSM cycle. The TAU process is mainly used to refresh the Tracking Area of the device, and to indicate availability to receive data from the network. While the first point is necessary in mobile, non-periodic devices as the network needs to be aware of their location, that is not the case for stationary and periodic IoT devices, as there is communication with the network at fixed and predictable intervals. Therefore, by estimating their IAT, the network can assume that their Tracking Area remains the same as long as a transmission is not missed for more than n consecutive periods. To notify the device about network-originated data, we propose the use the paging procedure. As the PSM cycle is agreed between the device and the network, its exact ending time is known to the network in advance with millisecond accuracy, and can thus be used to page the device efficiently. Furthermore, paging the device is preferable in terms of energy consumption, as the RA process will need to be followed only when network-originated data really exist.

Second, we propose using the estimated IATs to decide when to delete the device's SC. Currently, the SC will be deleted if it has not been used for a (relatively short) period of time. As such, devices with long IATs may be forced to perform IMSI encryption (if used) before every transmission. A naive approach would be to simply increase the deletion threshold to a much larger value, but that is sub-optimal as the network would be forced to retain the SCs of periodic devices with short IATs for longer than necessary. However, similar to the Tracking Area refresh, we can assume that a device is still operational as long as it does not miss more than n consecutive periods. A similar approach can also be adopted for event-driven devices. While we cannot use estimated IAT in this case, these devices will perform the TAU process at the end of each PSM cycle. Therefore, we can

use the length of the PSM cycle, and only delete the SC if the device misses a TAU for more than n consecutive PSM cycles.

5.1.1 Mechanism

Our mechanism exploits the lack of mobility of NB-IoT devices, and reduces the number of costly and unnecessary procedures, thus decreasing their energy consumption. At the same time, it is able to accurately estimate when a SC needs to be deleted to prevent stalling contexts from being stored indefinitely, without increasing the energy consumption of the devices.

In our mechanism, devices need to inform the network whether they are periodic or not, but are not required to provide their actual IAT value. At their initial connection devices register with the network using the existing RA and Attach processes without any modifications. During the Attach process, the network questions the device regarding its capabilities using the *Capability Enquiry* message. The device replies with the *Capabilities Enquiry Response* message, which includes a new field indicating whether the device is periodic or not, which will prompt the network to estimate its IAT using either its future transmissions for periodic devices, or the agreed PSM cycle for non-periodic devices. The new field can either be included in one of the message's existing extensions to retain backwards compatibility, or can be added in its main body. On the device side, this can be accomplished with the use of AT commands which can be easily set by the application designer, or added with a firmware update. The remainder of the Attach process remains unchanged.

The network estimates the IATs as the running average of the last y connections. The value of y is defined by the network operator based on the storage capabilities and the number of devices being served by the network. Specifically, the network estimates the IAT of the i_{th} device at the n_{th} connection as $s_i = \frac{1}{y} * (t_{n-y} + t_{n-y+1} + ... + t_n)$. Based on the estimated IATs the network can determine an approximate time of the device's next transmission/TAU, and adjust its SC deletion time as: $t_{deletion} = t_{now} + s_i * (m + offset)$ where m is the maximum number of missed periods allowed and is determined by the network, t_{now} is the current time, and $offset$ is the IAT percentage that can be tolerated as time offset. For this mechanism we consider that a value of $m = 0.5$ is adequate.

5.1.2 Results

Fig. 10 shows the energy gains per period P (section 4.1) of the device C. We can see that the proposed optimization significantly benefits devices with large IATs, resulting in energy gains of up to 37.8% per period P. As these devices spend most of their lives in the PSM state, the TAU process in these cases adds an unnecessary energy cost that can be avoided without impacting their operation.

5.2 Elimination of Random Access Process

The RA process is repeated every time the device connects to the network, and considering the significant energy cost of a single RA process, this accounts for a large part of the overall battery consumption. The RA process serves two purposes: (i) receiving the TA information (Sec. 2.1) in order to synchronize in the uplink, and (ii) receiving an uplink grant for the following transmissions.

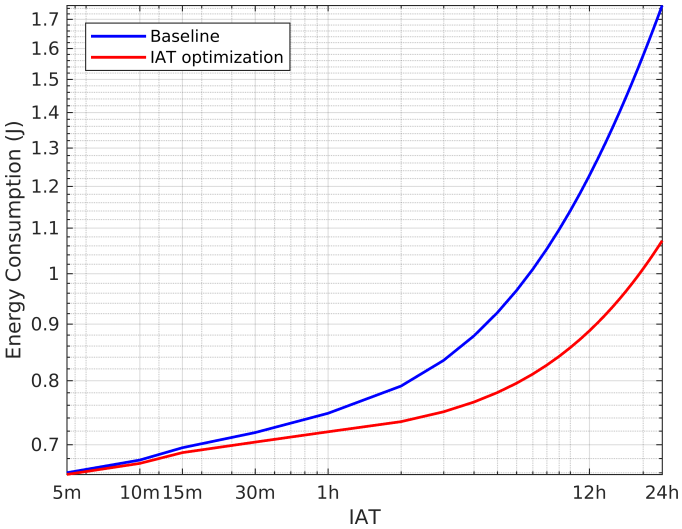

Figure 10. Energy gains per period P for different IATs, with and without our optimization based on IAT estimation.

For highly mobile devices, getting a new TA at each connection is necessary as it is likely to have changed from their last connection. As NB-IoT devices are mainly stationary, their TA does not significantly change between subsequent connections, and changes in their environment (e.g. a passing car) do not have a significant impact on the signal propagation delay, thus not affect their TA value. Therefore, we propose a mechanism that omits the RA process for stationary devices, and provisions an uplink grant for their next transmission. We do note however, that if synchronization is lost for any reason, it can still be detected and corrected using the existing TA update message. Further, in contrast to the two-step RA process currently discussed by 3GPP for Release 17 onward [5, 6] which aims to minimize the cost of an RA process, our mechanism can eliminate RA processes altogether (apart from the initial one), by exploiting the periodic nature of most IoT devices to pre-schedule appropriate grants in advance.

Our mechanism works as follows. On its very first connection, a stationary NB-IoT device performs the existing RA process without modifications to receive its corresponding TA value. During the Attach process, the device informs the BS whether it is stationary or not, using the Capability Response message. Similarly to our first optimization, this can be done either by introducing a new field in the existing message, or by using one of its extensions. After their application data exchange, stationary devices retain their TA value to use it again the next time they wish to transmit data. The BS also retains the devices' bearers (DRBs and SRBs).

In order to completely eliminate the RA process, stationary devices that re-use their TA values, need to have an uplink grant for their next transmission. This can be scheduled by the network based on the estimated IAT (Sec.5.1.1),

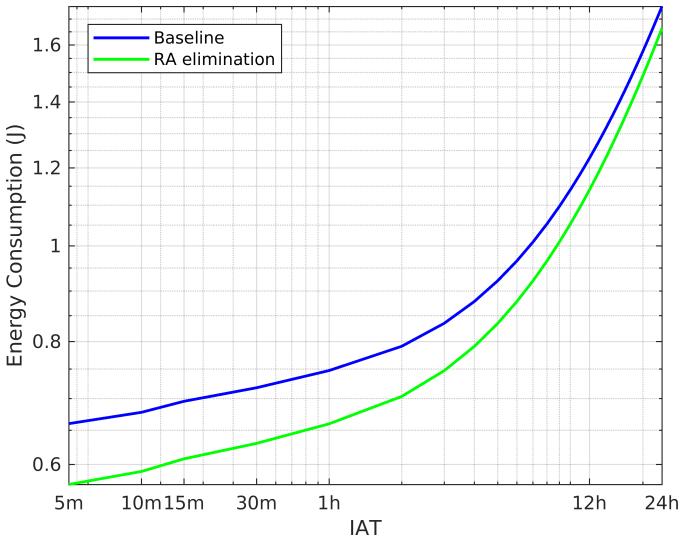

Figure 11. Reduction in device energy consumption over a 10-year period due to RA elimination for different IATs.

which is used to calculate the correct *System Frame Number (SFN)* and *Hyper Frame Number (HFN)* in the future that the device will wake up and pre-schedule sufficient resources then, adding a small number of frames to cover clock drifts at the device side. The network informs the device about its future scheduling of the Connection Release procedure. As the elimination of the RA process removes extra delays (e.g. due to collisions), such future resource scheduling is straight-forward and accurate. If however the device misses its scheduled resources, the existing RA process can be followed to request new resources. However, as the bearers exist, the Attach process can still be skipped.

An important feature of the proposed mechanism is that although the devices do not follow the RA and Attach processes, the data can still be securely exchanged. Specifically, as the scheduling of resources is device-specific, the BS is aware of which device transmitted in what resources. This means that the device can reuse its previously established security context to encrypt its data, and the network will still be able to decrypt them. If required, the network can request the renewal of the SC, which can be applied either to the current transmission (i.e. the first transmission is rejected and the device needs to repeat it after the renewal of the SC), or to the device's next transmission.

To assess the performance of this optimization we examine the reduction in energy consumption over a 10-year period (Fig. 11). We can see that eliminating the RA process further reduces the energy consumption compared to the current procedure where no optimization is applied, regardless of the device's IAT. The greatest gains are observed when short IATs are used, due to the increased number of connections they have to do throughout their lives, with a maximum gain of 13.6%. Smaller, but still important energy gains are also observed for longer IATs, with a 10.3% gain for a device with a 24-h IAT.

5.3 Best Practices for Energy Reduction

5.3.1 Inactivity Timer & Active Waiting

Active waiting requires the second greatest amount of energy after the IMSI encryption. Since this feature was introduced to limit the number of frequent connection establishments, it may not be required in NB-IoT where applications are expected to transmit all of their data in one go. Although using the C-DRX can decrease the energy consumption, forcing the devices to remain connected to the network for a period of time after their transmission unnecessarily wastes energy. Our results showed that even the use of the shortest C-DRX cycle (0.32 seconds) can decrease the energy consumption by 22% and 38.2% for a 10-sec and 50-sec inactivity timer respectively. Further gains can be achieved as the C-DRX cycle increases.

3GPP has recognized the benefits of releasing a connection early and introduced the *Release Assistance Indicator (RAI)* feature [8] as part of a series of optimizations in Releases 14 and 15. The RAI is a new field included in control messages to indicate that the device wishes to terminate its connection, so that the connection release procedure can be triggered immediately, reaching an energy gain of 98%.

It is important to note that this novel feature is not supported by older NB-IoT devices, and in fact our experiments showed that two of the tested devices did not include it. Although this feature can be added with a software update, existing applications also need to be updated in order to use it. Furthermore, there is currently no support to allow the application provider to inform the network whether outstanding data exists [37], to avoid paging the devices shortly after they released a connection. However, the energy benefits are significant, and we believe that it is important that this feature is enabled in both older and new NB-IoT devices.

5.3.2 Attach process

Similarly, the Attach process is costly, and needs to be repeated at each network connection, incurring a significant impact on the battery life of the devices, especially for devices with short IATs. Therefore, 3GPP defines the *Control Plane (CP)* and *User Plane (UP)* optimizations [11] (Sec. 2.2) to help reduce the energy consumption of NB-IoT devices. Although these optimizations have some drawbacks, we believe that they can contribute to the reduction of the energy consumption, and therefore, we estimate their energy gains if they are implemented (Fig. 12) using our simulator. Our experiments show that both approaches can decrease the energy consumption, especially for devices with short IATs. Greater energy gains can be achieved when the CP optimization is used, provided that the data size and QoS requirements can be met by the SRBs. Specifically, the UP optimization decreases the energy consumption by 2.9% while the CP optimization results in an energy gain of 5.2% when a 5-min IAT is used. For a 24-h IAT, the energy gain drops to 1.2% and 2.2% for the UP and CP optimizations, respectively.

5.3.3 Security Deletion Frequency

Although our proposed IAT optimization mechanism simplifies the decision of when to delete a SC to prevent stale contexts from being stored indefinitely and without incurring extra energy consumption, it is a new proposal waiting

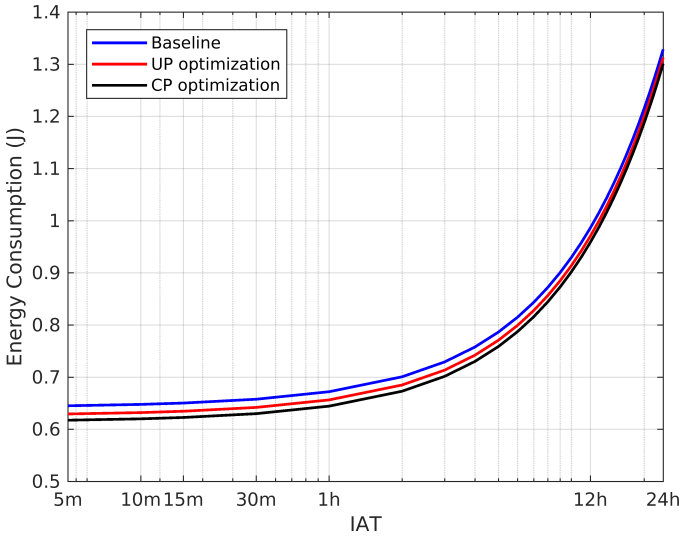

Figure 12. Energy consumption per period P with and without the UP and CP optimizations for different IATs.

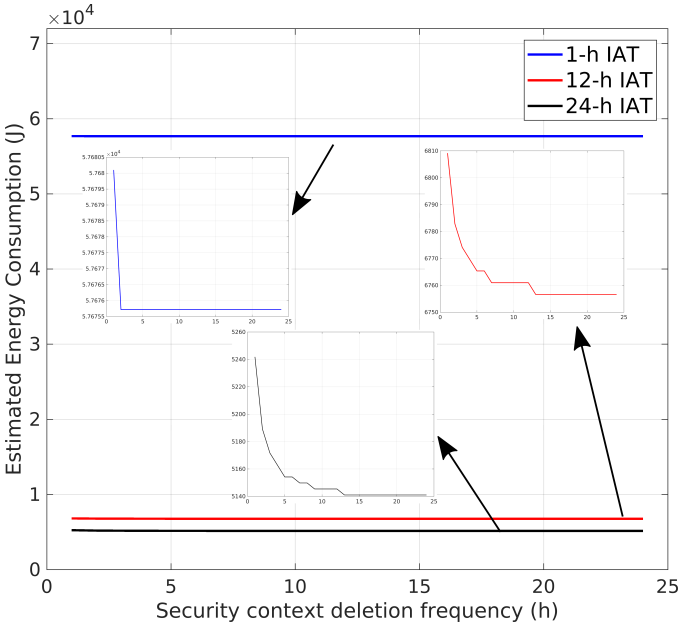

Figure 13. Energy consumption for 3 different IATs when the SC deletion frequency ranges from 1 to 24 hours. Devices with short IATs will seldom be required to renew their SC, while larger IATs introduce a notable energy cost. We have enlarged the individual results for better clarity

to be deployed, and operators must make a decision on when to delete a security context. Therefore, here we provide an evaluation of the energy consumption with 1-h, 12-h and 24-h IATs, and a SC deletion frequency ranging from 1 to 24 hours, considering device C (Fig. 13). Please note that the energy consumption includes all operations within a period P (Sec. 4.2).

We can observe a significant difference between the 1-h IAT and the other two, which is explained by the fact that for the 12-h and 24-h IAT the device spends the majority of its life in PSM. However, a device with 1-h IAT will seldom be required to renew its SC, as usually the SC deletion times in commercial networks are larger than 1 hour. This is not the case for the 12-h and 24-h IATs that will be required to renew their SCs more frequently.

5.4 Ablation Study

Finally, we compare the energy consumption per period P for different IATs, using each of the aforementioned optimizations (either proposed by us, or by 3GPP), and their combinations (Fig. 14). Here, we use a 10-second IT, with 5.12-second C-DRX cycles. As the IAT decreases, the energy consumption of the RA and Attach processes becomes increasingly significant as they need to be repeated often. Therefore, for short IATs, the 3GPP proposed UP/CP optimizations and our RA elimination based optimization provide the greatest gain. As the IAT increases, however, the energy consumption of the device starts being dominated by the deep sleep state, which significantly reduces the effectiveness of these optimizations. Instead, our IAT optimization that targets the PSM/security renewal provides the most reduction in energy consumption (up to 46.14%). As these various optimizations are mutually complementary, they can be combined to yield the greatest gains across the entire range of IATs.

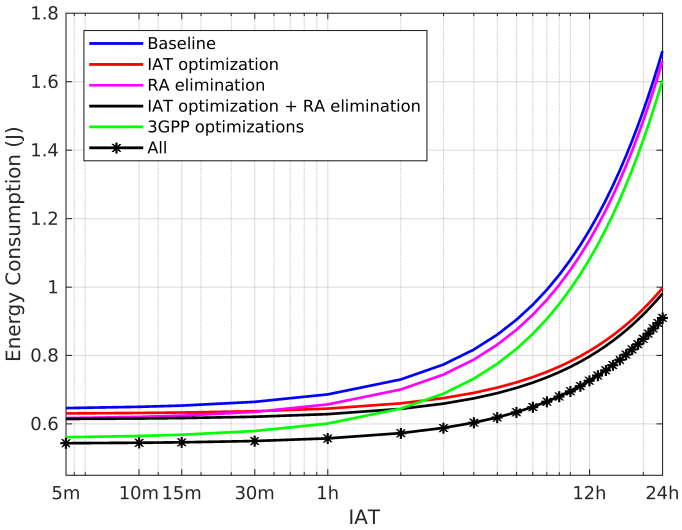

Figure 14. Ablation study of the estimated energy consumption for different optimizations for device energy savings and IATs.

6 Related Work

Device energy consumption has been a concern that influenced the design of cellular networks, even more so since the beginning of the IoT era (e.g., [30]). The DRX feature can significantly contribute to the energy savings, and as such a large number of works (e.g., [20, 33]) attempted to optimize the DRX parameters of IoT devices, in order to increase their sleep period, and thus decrease the energy consumption. However, the energy consumed during a DRX cycle is orders of magnitude lower than that of the RA or Attach processes, and thus optimizing the currently used protocols is equally important. Other works (e.g., [19, 50, 44]) use the DRX settings, as well as information (e.g., the device capabilities, the battery level), as a basis to efficiently schedule data transmissions in order to achieve low power usage. Although these approaches can synchronize when the device checks for network-originated data with the times it needs to send its own data, they do not optimize the currently used procedures that incur a significant energy cost (Sec. 3).

Due to the large number of messages exchanged (either control or data), several works (e.g., [36, 37]) attempted to optimize the transmission parameters to decrease the energy consumption of IoT devices while in the working state. While such changes can have a positive impact on the battery life, the major inefficiencies stem from the fact that the protocols used were initially designed for HTC devices, that present significantly different traffic characteristics. As such, they are ill-suited for NB-IoT devices.

Until recently, works on energy consumption did not focus on a specific cellular network technology, and thus the individual characteristics of the devices were not taken into account. However, as such characteristics (e.g. periodic, stationary devices) can make important difference, or pose strict constraints (low-capabilities, cost limits), several recent works have focused specifically on the NB-IoT technology. The works in [37, 22, 47, 49, 17] discuss the NB-IoT technology in terms of energy consumption, analyzing the different modes of operation and their associated energy cost. However, they only measure the transmission and reception operations, which are the least expensive operations in terms of energy consumption (Sec. 3). As such, they do not give enough insight into which areas require improvement.

Similar to us, some recent works [34, 15] attempt to model and experimentally measure the energy consumption of NB-IoT devices. However, they differ to our work in terms of scope and granularity of the measurements. Both [34, 15] focus on measuring the energy consumption of data transmission/reception as a function of different network configurations (data rates, etc.) In contrast, our work additionally measures the energy that the devices need to spend for operations that facilitate these data exchanges (encryption, communication with the core network, active waiting etc.), which gives a better estimate of the overall energy consumption of the device. Also crucially, [34, 15] are limited to modeling the energy consumption and do not assess the impact and necessity of each individual operation, nor do they propose improvements. Instead, we examine which processes contribute the most to the energy consumption, and propose appropriate protocol optimizations and best practices aimed at lowering the corresponding components of device energy consumption.

7 Conclusions

In this paper, we presented a detailed measurement-based characterization of the energy consumption in real NB-IoT devices, and identified expensive network operations that can be the target for optimizations. Further, we presented an energy consumption model which we used in combination with our measurements in a simulation study to estimate the battery requirements of NB-IoT devices to achieve a battery life of 10 years. Our analysis showed that the current 3GPP specifications for NB-IoT that are largely inherited from 4G/LTE and 5G are not efficient in NB-IoT if used unchanged. We then proposed two optimization mechanisms that exploit the traffic characteristics of NB-IoT devices to reduce the energy cost of unnecessary operations, thereby substantially decreasing the overall energy consumption.

8 References

[1] Crypto++ Library 7.0 - Free C++ Class Library of Cryptographic Schemes.

[2] Python Cryptography Toolkit.

[3] 3GPP. NB-LTE - Battery lifetime evaluation. RP 151393, 3rd Generation Partnership Project (3GPP), Sep 2015.

[4] 3GPP. Cellular system support for ultra-low complexity and low throughput Internet of Things (CIoT). TR 45.820, 3rd Generation Partnership Project (3GPP), Aug 2016.

[5] 3GPP. 2-step RACH procedure consideration. TDoc R1-1700792, 3rd Generation Partnership Project (3GPP), 2017. For Discussion.

[6] 3GPP. Discussions on 2 Steps RACH Procedure. TDoc R1-1700668, 3rd Generation Partnership Project (3GPP), 2017. For Discussion.

[7] 3GPP. 3GPP System Architecture Evolution (SAE); Security architecture. TS 33.401, 3rd Generation Partnership Project (3GPP), Jul 2018. Rel. 15.

[8] 3GPP. General Packet Radio Service (GPRS) enhancements for Evolved Universal Terrestrial Radio Access Network (E-UTRAN) access. TS 23.401, 3rd Generation Partnership Project (3GPP), Jul 2018. Rel. 15.

[9] 3GPP. LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); Physical layer procedures. TS 36.213, 3rd Generation Partnership Project (3GPP), Oct 2018. Rel. 15.

[10] 3GPP. LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); Radio Resource Control (RRC). TS 36.331, 3rd Generation Partnership Project (3GPP), Oct 2018. Rel. 15.

[11] 3GPP. Non-Access-Stratum (NAS) protocol for Evolved Packet System (EPS). TS 24.301, 3rd Generation Partnership Project (3GPP), Jun 2018. Rel. 15.

[12] 3GPP. Radio Resource Control (RRC); Protocol specification. TS 25.331, 3rd Generation Partnership Project (3GPP), Jul 2018. Rel. 15.

[13] 3GPP. Radio transmission and reception; Part 3: Radio Resource Management (RRM) conformance testing. TS 36.521, 3rd Generation Partnership Project (3GPP), Oct 2018. Rel. 15.

[14] 3GPP. Security architecture and procedures for 5G System. TS 33.501, 3rd Generation Partnership Project (3GPP), Jul 2018. Rel. 15.

[15] P. Andres-Maldonado, M. Lauridsen, P. Ameigeiras, and J. M. Lopez-Soler. Analytical Modeling and Experimental Validation of NB-IoT Device Energy Consumption. *IEEE Internet of Things Journal*, pages 1–1, Mar 2019.

[16] Arduino. Arduino Uno Rev3.

[17] H. Bello, J. Xin, Y. Wei, and M. Chen. Energy-Delay Evaluation and Optimization for NB-IoT PSM with Periodic Uplink Reporting. *IEEE Access*, PP, 12 2018.

[18] R. Borgaonkar and S. Udar. Understanding IMSI privacy. In *Black-Hat*, Aug 2014.

[19] H. Chang and M. Tsai. Optimistic DRX for Machine-Type Communications in LTE-A Network. *IEEE Access*, 6, Jan 2018.

[20] K. Davaslioglu, C. C. Coskun, and E. Ayanoglu. Energy-Efficient Resource Allocation for Fractional Frequency Reuse in Heterogeneous Networks. *IEEE Transactions on Wireless Communications*, 14(10):5484–5497, Oct 2015.

[21] Ericsson. Cellular networks for massive IoT: Enabling low power wide area applications white paper. Technical report, Ericsson, Stocholm, Mar 2016. Accessed: 2018-11-5.

[22] J. Finnegan. An Analysis of the Energy Consumption of LPWA-based IoT Devices. In *IEEE International Symposium on Networks, Computers and Communications (ISNCC)*, Jun 2018.

[23] Huawei. NB-IoT - Enabling New Business Opportunities. White paper, Huawei, 2015.

[24] S. R. Hussain, O. Chowdhury, S. Mehnaz, and E. Bertino. LTEInspector: A Systematic Approach for Adversarial Testing of 4G LTE. In *Network and Distributed System Security Symposium*, 2018.

[25] J. Jermyn, R. P. Jover, I. Murynets, M. Istomin, and S. Stolfo. Scalability of Machine to Machine systems and the Internet of Things on LTE mobile networks. In *2015 IEEE 16th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, pages 1–9, Jun 2015.

[26] E. C. Jiménez, P. K. Nakarmi, M. Näslund, and K. Norrman. Subscription identifier privacy in 5G systems. In *2017 International Conference on Selected Topics in Mobile and Wireless Networking (MoWNeT)*, May 2017.

[27] J. Jonsson and B. Kaliski. Public-Key Cryptography Standards (PKCS): RSA Cryptography Specifications Version 2.1, 2003.

[28] R. P. Jover. LTE security, protocol exploits and location tracking experimentation with low-cost software radio. *CoRR*, 2016.

[29] R. P. Jover. Some key challenges in securing 5G wireless networks. *FCC*, 2017.

[30] N. Kaur and S. K. Sood. An Energy-Efficient Architecture for the Internet of Things (IoT). *IEEE Systems Journal*, Jun 2017.

[31] Keysight Technologies. E7515A UXM Wireless Test Set.

[32] S. Landstrom, J. Bergstrom, E. Westerberg, and D. Hammarwall. NB-IoT: A Sustainable Technology for Connecting Billions of Devices. Review, Ericsson, 2016.

[33] M. Lauridsen, G. Berardinelli, F. M. L. Tavares, F. Frederiksen, and P. Mogensen. Sleep Modes for Enhanced Battery Life of 5G Mobile Terminals. In *2016 IEEE 83rd Vehicular Technology Conference (VTC Spring)*, pages 1–6, May 2016.

[34] M. Lauridsen, R. Krigslund, M. Rohr, and G. Madueno. An Empirical NB-IoT Power Consumption Model for Battery Lifetime Estimation. In *2018 IEEE 87th Vehicular Technology Conference (VTC Spring)*, Jun 2018.

[35] M. Lichtman, R. P. Jover, M. Labib, R. Rao, V. Marojevic, and J. H. Reed. LTE/LTE-A jamming, spoofing, and sniffing: threat assessment and mitigation. *IEEE Communications Magazine*, Apr 2016.

[36] P. A. Maldonado, P. Ameigeiras, J. P. Garzon, J. J. R. Munoz, and J. M. L. Soler. Optimized LTE Data Transmission Procedures for IoT: Device Side Energy Consumption Analysis. *CoRR*, 2017.

[37] P. A. Maldonado, P. Ameigeiras, J. Prados-Garzon, J. Navarro-Ortiz, and J. M. Lopez-Soler. Narrowband IoT Data Transmission Procedures for Massive Machine-Type Communications. *IEEE Network*, 31(6):8–15, Nov 2017.

[38] Monsoon Solutions. High Voltage Power Monitor.

[39] Pycom. Expansion Board 2.0.

[40] Pycom. GPy - Triple-network WiFi, Bluetooth and LTE-M dev board.

[41] Quectel. Quectel LTE BC95 NB-IoT Module.

[42] J. Schlienz and D. Raddino. Narrowband Internet of Things White Paper. Technical report, Rohde & Schwarz, 2016.

[43] J. Schlienz and D. Raddino. 3GPP Low Power Wide Area Technologies. Technical report, GSMA, 2017.

[44] A. Sehati and M. Ghaderi. Online Energy Management in IoT Applications. In *IEEE International Conference on Computer Communications (INFOCOM)*, 04 2018.

[45] M. Z. Shafiq, L. Ji, A. X. Liu, J. Pang, and J. Wang. Large-Scale Measurement and Characterization of Cellular Machine-to-Machine Traffic. *IEEE/ACM Transactions on Networking*, 21(6):1960–1973, Dec 2013.

[46] A. Shaik, R. Borgaonkar, N. Asokan, V. Niemi, and J. P. Seifert. Practical attacks against privacy and availability in 4G/LTE mobile communication systems. *CoRR*, 2015.

[47] R. S. Sharan, Y. Wei, and S. H. Hwang. A survey on LPWA technology: LoRa and NB-IoT. *ICT Express*, Mar 2017.

[48] Sodaq. The first NB-IoT shield for Arduino: supported by T-Mobile.

[49] M. E. Soussi, P. Zand, F. Pasveer, and G. Dolmans. Evaluating the Performance of eMTC and NB-IoT for Smart City Applications. In *2018 IEEE International Conference on Communications (ICC)*, May 2018.

[50] G. Szabo, G. Pongracz, I. Godor, R. Coster, and M. Sintorn. Service aware adaptive DRX scheme. In *2014 IEEE Globecom Workshops (GC Wkshps)*, Dec 2014.

[51] C. Y. Yeoh, A. B. Man, Q. M. Ashraf, and A. K. Samingan. Experimental assessment of battery lifetime for commercial off-the-shelf NB-IoT module. In *2018 20th International Conference on Advanced Communication Technology (ICACT)*, Feb 2018.