The Power of Models: Modeling Power Consumption for IoT devices

Borja Martinez, Màrius Montón, *Member, IEEE*, and Joan Daniel Prades,

Abstract

Making more energy efficient technologies is still far from having those envisaged ubiquitous deployments (so called the Internet of Things, or the Industrial Internet), which will enable optimal industrial operation, and will contribute to improve the social welfare.

Today, it is possible to build a device which features this industrial wireless performance, and is able to in-node analyze the acquired data. However, energy-dimensioning the device in order to meet the application requirements is not an easy task, especially when the reliability claimed for industrial applications faces up to the uncertainty introduced by energy harvesting.

Modeling and dimensioning the energy consumption of an application at pre-deployment or pre-production stages is of utmost importance considering the critical requirements of IoT applications in terms of reduced cost, life-time, and available energy.

This paper presents a comprehensive model for the power consumption of wireless sensor nodes that accounts for all the energy expenditures at system-level: communications, acquisition and processing. The model is only based on parameters that can be empirically quantified, once the platform (i.e., technology) and the application (i.e., operation conditions) are defined. This results in a new framework for the study and analysis the energy live-cycle within the applications, suitable to determine in advance the specific weight of application parameters and to understand the tolerance margins and trade-offs in the system.

Index Terms

Low power models, Sensor system networks, networkable sensors, sensor system integration

B. Martínez is with Microelectronics and Electronic Systems Dpt., Universitat Autònoma de Barcelona, Spain and Worldsensing, Barcelona, Spain. (e-mail: borja.martinez@uab.cat)

M. Montón is with Microelectronics and Electronic Systems Dpt., Universitat Autònoma de Barcelona, Spain and Worldsensing, Barcelona, Spain.

J.D. Prades is with MIND-IN^{2}UB, Department of Electronics, Universitat de Barcelona, Spain.

The Power of Models: Modeling Power Consumption for IoT devices

I. INTRODUCTION

ENERGETICALLY autonomous wireless sensors are the backbone of the Internet of Things (IoT) [1]. To implement this concept each sensor node must be able to harvest, buffer and consume the energy available in the environment, in an efficient manner [2].
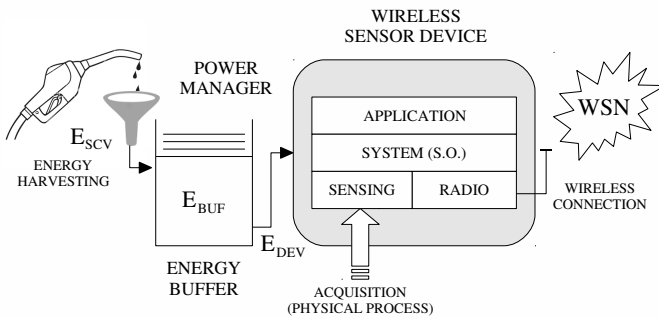

Fig. 1. Generic energy model for IoT device.

Fig. 1 represents the three main blocks to optimize in the design of energy efficient nodes. First, the *power management unit* collects energy from the ambient and converts it into usable electrical power, properly adapted to feed the following blocks [3]. Second, the energy is *buffered* in a battery, a supercapacitor, or any other device capable of storing and releasing energy [4]. Third, the energy is consumed in the *device* to carry out the required sensing, processing and communication tasks [5].

In this architecture, energy follows the *energy flow model* [3], described in Eq. (1); being $\mathcal{E}$ and $\mathcal{P}$, energy and power terms, respectively:

$$\mathcal{E}_{BUF}^{(t=0)} + \int_{\tau=0}^{t} \mathcal{P}_{SCV}(\tau)d\tau \geq \int_{\tau=0}^{t} \mathcal{P}_{DEV}(\tau)d\tau = \mathcal{E}_{DEV}(t) \quad (1)$$

In this general model, the energy initially stored in the buffer $\mathcal{E}_{BUF}$ and the additional energy scavenged from the medium $\mathcal{E}_{SCV} = \int \mathcal{P}_{SCV}d\tau$ must be greater than the energy that the device requires to operate $\mathcal{E}_{DEV} = \int \mathcal{P}_{DEV}d\tau$ throughout the whole operation time t of the system. A natural constraint follows from *energy causality*, which dictates that energy cannot be used before it is available [6]; so Eq. (1) condition must hold strictly at any time.

This is analogous to the classic *producer-consumer problem* in computing (also known as the bounded-buffer problem) [7]. The problem involves two processes, the *producer* and the *consumer*, sharing a common, fixed-size, buffer used as a queue. The producer's job is to generate pieces of data and store them into the buffer. At the same time, the consumer is removing data from the buffer. The problem is assuring that the producer won't try to add data into a full buffer, and the consumer won't try to remove data from an empty buffer.

Thus, the energy flow in a wireless sensor device follows this model closely. However, the consumer's goal in energy-limited systems is not always to dispatch tasks from the queue as fast as possible. In fact, most of the time, the optimization objective will be more conservative in order to optimize power consumption. Only in seldom occasions, the device will require an aggressive configuration, thus maximizing the processing performance [8]. This leads to alternative cost functions that determine the *energy policies* of the device [9].

Recently, the topic of *neutral design policies* is thriving in the Wireless Sensor Network (WSN) research community (see [10] and citations within). The concept of neutrality accounts for the fact that the energy used, over the long term, should be, at most, equal to that harvested ($\mathcal{E}_{SCV}(t) \geq \mathcal{E}_{DEV}(t), t \to \infty$). In other words, the energy stored in the first part of ($\mathcal{E}_{BUF}^{(t=0)}$ in Eq. (1)) is negligible after some time running. This is a general condition for the sensors to be energetically self-sufficient, that is, unattended devices will ideally last for an unlimited period of time [11].

In spite of its conceptual simplicity, the realization of such finely tuned powering schemes is extremely challenging and requires the optimization of many design parameters that affect the system performance in a complex way. In this context, *system-level consumption models* are a valuable tool to support the design of energy constrained devices.

In the past few years, a vast literature has emerged on energy constrained WSN. Most of the developed work is focused on network activity, i.e., how communications issues affect the device consumption. Other works are concerned with the role of processors in the energy expenditure, or even with specific details about the sensing process itself. However rarely this topic has been addressed from a system level perspective. For this reason, the applicability of these models into practical designs is still limited. Nevertheless, they provide useful insights into some of the design trade-offs.

Among the first group, mainly focused in the *communications side*, it is worth noting the work presented in [12], which find optimal transmission policies. according to the expected energy income. However this work is focused on throughout maximization that is not always required in a more general case. More related to our approach is [13], that develops a method for dimensioning the energy spent during the communication process. The basic idea is to estimate the power consumption of wireless sensors based on the individual contributions of each of the building blocks involved in the communication. Neither sampling techniques nor the cost of

the application itself are however considered.

Regarding *processing power*, several measurement-based methods can be found in the literature. Most of these models, like those presented in [14] or [15], use data obtained from a physical target device and associate the processor instructions with the corresponding energy cost. The total energy is the aggregate cost of all executed instructions that can be obtained by running the application in an emulator. These works are focused on the accurate energy profiling of the CPU and processor peripherals (FLASH, RAM, ADCs), but any of them take into consideration other components that have a significant weight in energy consumption, mainly related to communications. The main advantage of these measurement-based methods is the high accuracy in the energy estimation obtained due to the use of actual values measured in the target platform.

Some recent approaches to model the use of scavenging techniques in industrial wireless applications deal with the *sampling energy*, although the topic is still far from being addressed in deep. For instance, [16] assumes a dependence between the harvesting pattern and the applications needs, drawing a best effort policy: an application wakes up the system and transmits a packet when enough energy has been harvested. The sampling contribution is estimated to compute the total energy, but the work does not provide a clear modeling of the application energy requirements nor a detailed network energy consumption analysis. Besides, [15] evaluates the cost of capturing the sample from the processor side. The model is built based on the type of the executed assembly instructions, the number of accesses to the memory as well as the analog-to-digital converter; but, the power consumption associated with external sensors and components is not computed. In general, the energy consumption by the sensor has been underestimated in the literature, and this part can contribute strongly in the consumption of the device, particularly for active sensors (i.e., needing some excitation).

Finally, from a different perspective, [17] analyzes several *system-level design* aspects of wireless embedded systems. This survey identifies the synergies between wireless sensor networks and non-intrusive electrical-signal-based monitoring and fault diagnosis for industrial systems. The main scope is to provide a system overview of applications in a network architecture. That paper also provides detailed analyses to address the real-world challenges in designing and deploying WSNs in practice, including wireless-link-quality dynamics, noise and interference on communication range and reliability. However, the impact of these system level issues on the energy consumption is not clearly addressed.

As it may be inferred from this brief analysis, the models developed so far only cover partial areas of the design space. However, the energy budget is a shared resource and, in general, a systematic study of how energy is distributed in the whole system has not been tackled in the literature.

In this work we aim to formalize a system-level energy consumption model (the $\mathcal{E}_{DEV}$ term in Eq. (1)) that can be easily simulated and numerically evaluated. The development of the presented methodology is in part motivated by the lack of a rigorous and systematic approach to model the
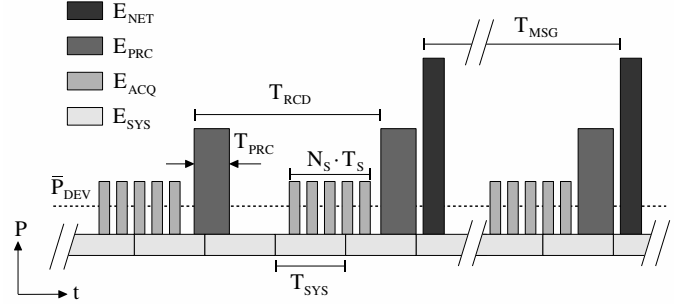

Fig. 2. Characteristic time evolution of energy usage. The vertical dimension represents the instantaneous power consumption of the device. Consequently, shaded areas depict the accumulated energy for each task. The dashed line in the figure represents the average power of the device $\bar{P}_{DEV}$.

energy consumption for smart-sensor devices with the required system-wide view.

Our model combines ideas and strategies proposed in different works with new evaluation approaches in order to end up with a whole system model. From a practical perspective, the model is only based on operational parameters that can be easily quantified, either by design or by empirical estimation. We also demonstrate with two study cases that the model can effectively assist on the design and simulation of WSN systems, providing concrete answers to abstract problem formulations, as energy causality, neutrality or sustainability.

II. Bottom-Up Modeling of Energy Consumption

Most of the industrial monitoring applications follow a common operational pattern: data is acquired by some sensor of the system, processed in a controller unit and some information is then sent through a wireless channel. This process repeats over time, and the role of its duty cycle is fundamental in the energy consumption: the smaller the duty cycle (which can be achieved by shortening the active time or by lengthening idle periods) the lower the average power. [DEL] the larger the duty cycle the lower the average power.

Based on this assumption, the power required to operate a wireless sensor device can be broken down into three main blocks: for data sensing or *acquisition* $\mathcal{P}_{ACQ}$, for data handling or *processing* $\mathcal{P}_{PRC}$, and data communication, or *networking* $\mathcal{P}_{NET}$. Additionally, a tiny fraction of the available reservoir is intended for system management tasks, such as running a real-time operating system (RTOS) or rising the system at periodic wake-ups. The needs of these management tasks are gathered in this $\mathcal{P}_{SYS}$ contribution. These elements together form Eq. (2): the general expression for the device's power consumption $\mathcal{P}_{DEV}$.

$$\mathcal{P}_{DEV} = \mathcal{P}_{NET} + \mathcal{P}_{ACQ} + \mathcal{P}_{PRC} + \mathcal{P}_{SYS} \qquad (2)$$

Fig. 2 shows a characteristic sequence of tasks for a monitoring application. The system wakes-up periodically to get a record with elapsed time of T_{RCD}. For each cycle, three main steps are executed: i) capture a set of N_S samples with

T_S period; ii) run some process/analysis of the *record* [1] of acquired samples, and iii) report gathered data, update server information or trigger an alarm when an anomaly is detected, generating some radio traffic with T_{MSG} intervals.

In Fig. 2, the vertical dimension represents the instantaneous power consumption of the device. Consequently, shaded areas depict the accumulated energy for each task. $\mathcal{E}_{NET}$ stands for the energy drained for communication tasks, $\mathcal{E}_{ACQ}$ for acquisition and $\mathcal{E}_{PRC}$ for processing. Running in the background, the operating system or scheduler executes different synchronization and coordination tasks, which may include network management. This systematic activity is carried out within T_{SYS} cycles, and demand an associated energy $\mathcal{E}_{SYS}$. For illustration purposes, the dashed line in the figure represents the average power of the device $\bar{\mathcal{P}}_{DEV}$.

The proposed model is based on an atomic breakdown of each building block of Eq. (2), interpreted within the Fig. 2 framework. The instantaneous power consumption is integrated over the duration of the corresponding task, and its characteristic *temporal scale* or period of repetition is then averaged out. In the next sections, we go into details on the analysis of each building block.

A. Modeling Network Energy

1) Point to Point Communications: The simplest model for wireless communications consists of an interference-free, single-hop scenario. The Medium Access Control (MAC) layer is idealized; i.e., apart from transmission and reception, it does not introduce further energetic inefficiencies due to collisions and idle times for floor acquisition. In this case, the power consumption can be estimated for each device independently. As any attempt at transmission is supposed to arrive to the destination, the model does not need to cope with interferences caused by other devices, congestion or any other collective issue.

Under these assumptions, the average power of the communications block can be expressed in terms of the energy required to send a radio message $\mathcal{E}_{MSG}$, and the time between consecutive messages $T_{MSG}^{(i)}$, as shown in Eq. (3). The index i of the summation runs for all messages on the averaging period N_{MSG}.

$$\bar{\mathcal{P}}_{NET} = \sum_{i=0}^{N_{MSG}} \frac{\mathcal{E}_{MSG}}{T_{MSG}^{(i)}} \qquad (3)$$

The energy per message $\mathcal{E}_{MSG}$ is a parameter that depends mainly on the specific radio technology. Two main factors govern this contribution: radio power and transmission time. Radio power tends to be maximized to increase its range, although it is legally limited in each Industrial, Scientific and Medical (ISM) band. Instead, transmission time is a parameter determined mainly by the modulation: depending on how a message is spread over time, it balances the complex trade-off between bit-rate (and thus consumption), range, reliability and immunity to interferences. The study of this topic is of out of

---

¹A *record* is defined as the process of waking up, taking a set of samples and storing it into memory, ready to be processed.
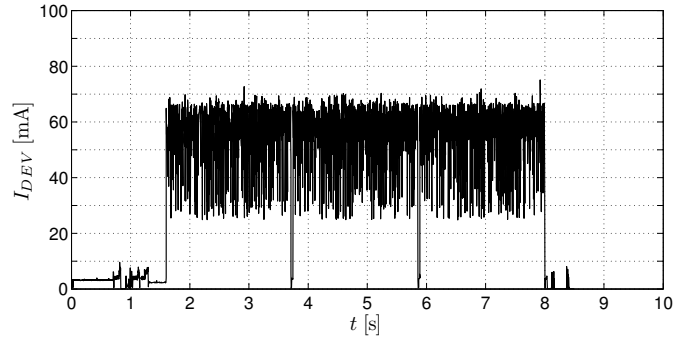

Fig. 3. Measured SIGFOX transmission energy consumption in terms of the current drained I_{DEV}. At $T = 1.6s$ starts the transmission of the payload, which is repeated 3 times. [Legend Rev.]

the scope on this work, but it is a fundamental step of the system design flow.

Regarding the time between messages, T_{MSG}^0 can be considered a constant parameter for periodic reporting applications. In this case, Eq. (3) reduces to a simpler expression given by Eq. (4)

$$\bar{\mathcal{P}}_{NET} = \frac{\mathcal{E}_{MSG}}{T_{MSG}^0} \qquad (4)$$

More generally, sensors nodes generate endogenous traffic, each one according to some distribution or stochastic process and the T_{MSG}^0 becomes a random variable. The message production rate, characterized by a certain probability distribution, depends basically on the underlying physical process. Then, the time elapsed between consecutive messages in Eq. (3) should be characterized by an appropriate statistical estimator $\widehat{E}[\ ]$. Typically it is used the expected value of the distribution defined as $\widehat{T}_{MSG}$, leading to Eq. (5). This approximation should be good enough for long-term averaging.

$$\bar{\mathcal{P}}_{NET} \cong \frac{\mathcal{E}_{MSG}}{\widehat{E}[T_{MSG}]} = \frac{\mathcal{E}_{MSG}}{\widehat{T}_{MSG}} \qquad (5)$$

The energy cost associated with each transmission may in turn depend on multiple factors:

- *Retransmissions*: some opportunistic approaches just retransmit the same message several times in order to increase the probability of delivery success. In this case, the energy per message is multiplied by the number of attempts N_R.

$$\bar{\mathcal{P}}_{NET} = N_R \cdot \mathcal{E}_{MSG}/T_{MSG} \qquad (6)$$

Fig. 3 shows a snapshot of a radio transmission using a SIGFOX transceiver [18], an illustrative example of this approach.

- *Radio power*: most radio transceivers allow programmers some control over transmission power, providing a trade-off between energy cost and distance range. Typically, the output level is selected from among a set of discrete values N_P, leading to a quantized energy scale [19].

$$\bar{\mathcal{P}}_{NET} = \mathcal{E}_{MSG}^{(N_P)}/T_{MSG} \qquad (7)$$

- *Spreading factor*: alternatively, some radio technologies can operate with different spreading factors N_{SF} [20]. The spreading factor increases the communication range, but lowers the bit-rate of the transmission: as the transmission time increases, more energy is required (see Fig. 4). This behavior can be easily modeled by some suitable function $h(\ )$, as detailed in the next section.

$$\bar{\mathcal{P}}_{NET} = h(N_{SF}, \mathcal{E}_{MSG})/T_{MSG} \tag{8}$$

Despite the simplicity of this scenario, it covers a large number of applications. In recent years, wireless low power communications are evolving towards a wide-area, low bit-rate, low-cost approaches, operating over distances long enough to avoid multi-hop techniques. Companies like SIGFOX, Semtech with LoRa and Weightless are examples of use for these technologies [18][20][21]. The wide range, combined with a low data-rate orientation, allows modeling the radio activity of these devices within the point-to-point, unidirectional link and retransmission free assumptions.
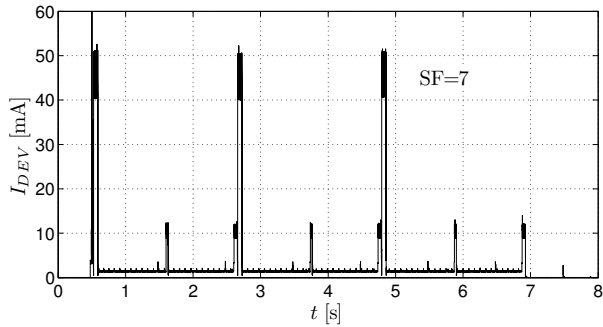

(a) Transmission using Spreading Factor 7
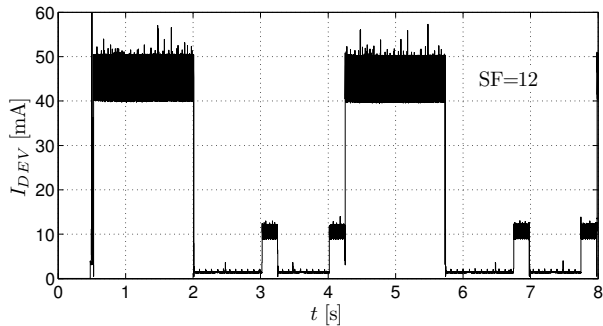

(b) Transmission using Spreading Factor 12

Fig. 4. Measured Cycleo transmission energy consumption in terms of the current drained I_{DEV}, for two spreading factor levels: SF=7 and SF=12. [Legend Rev.]

2) Time Synchronized Networks: These technologies form a second main category of industrial low-power radios. In this work, we adopt the model derived in [22] for Time Slotted Channel Hopping networks (TSCH). For this reason, in this section we only outline the basic features required for building a higher level system model.

TSCH networks show an ultra low power consumption profile due to the low power nature of IEEE802.15.4 compliant radios, and due to the fact that nodes are synchronized and
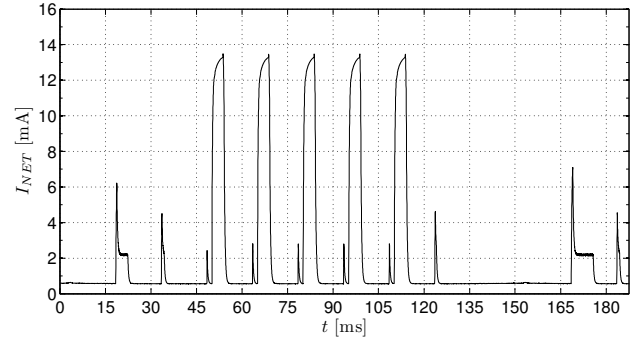
(a) Slot-frame composed by 10 slots with 5 of them *active*.
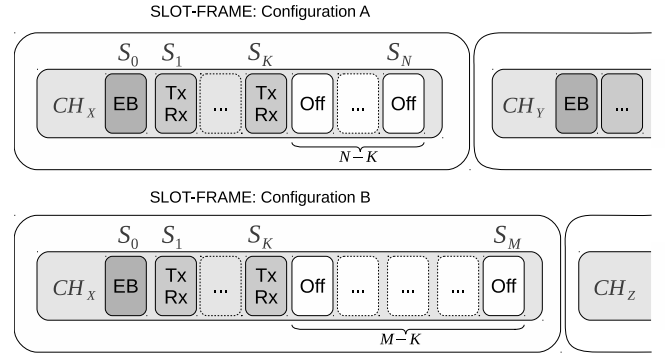
(b) Example of two TSCH slot-frame configurations.

Fig. 5. Consumption charactersitics of a TSCH network. (a) Measured TSCH network transmission energy consumption in terms of the current drained I_{NET}. (b) Scheme of the time allocation in two slot-frame configurations. [Legend Rev.]

actions occur at specific moments in time, enabling nodes to optimize the usage of their resources.

In a TSCH network, slots are grouped into slot-frames which repeat over time. Each type of slot has an energy consumption profile related to the hardware and the activity it is performing (e.g. transmit, receive, sleep, etc.) as shown in Fig. 5a.

The model used is based on profiling the energy consumption $\mathcal{E}_{SLOT}^{(i)}$ in each of those slots, counting the number of slots of each type, and calculating the total energy of the slot-frame. The average power can be obtained by dividing the total energy $\mathcal{E}_{SF}$ by the slot-frame period T_{SF}, as indicated by Eq. (9). Since slot-frames repeat cyclically, T_{SF} represents the characteristic temporal scale of the network.

$$\bar{\mathcal{P}}_{NET} = \frac{\mathcal{E}_{SF}}{T_{SF}} = \frac{1}{T_{SF}} \sum_{i=1}^{N_{SLOTS}} \mathcal{E}_{SLOT}^{(i)} \tag{9}$$

For the purpose of the presented methodology, the key feature is the impact of the network configuration on the energy consumption of the application. In a TSCH network, the slot-frame period is determined by the number of slots in the slot-frame and the time assigned to each slot $T_{SF}=N_{SLOT} \cdot T_{SLOT}$. As T_{SLOT} is a fixed network parameter, the slot-frame length N_{SLOT} determines how often actions repeat, which usually depends on application requirements.

A node is only *active* in a few time-slots in the slot-frame, which are used to send or receive information. During the

rest of non-active slots, the node remains switched off (*sleep*). Energy consumption can be reduced by increasing the length of the slot-frame, i.e., by inserting more *sleep* slots or by disabling some *active* slots so that they become *sleep* slots. If the number of *active* slots remains constant and the slot-frame size increases, the ratio of *sleep* slots to total time slots increases. Thus the average energy spent by the node is lower. The same effect is obtained by changing *active* slots to *sleep* while maintaining the slot-frame size constant. However, the reduction of activity comes at the cost of less bandwidth and increased latency. Reliability is also compromised, as less redundant links to neighbors are expected.

Fig. 5b shows an example of two TSCH slot-frame configurations with a different number of slots N and M, assuming $M > N$. The first slot is used for network discovery by means of *Enhanced Beacons*. Then K *data* slots for transmission and reception are common in both configurations. Configuration A has $N - K$ *sleep* slots (unused), while Configuration B has $M - K$ *sleep* slots, meaning that a node running in this configuration will be idle for longer periods.

B. Modeling Data Acquisition Energy

Monitoring applications can be classified in two categories: regular sensing, i.e. with a fixed acquisition interval, and event-driven sensing, i.e., characterized by some stochastic distribution. In event-driven sensing, a random event triggers the acquisition of a series of samples from the sensor. This event can be internal to the sensor (e.g. a random trigger in *compressed sensing* [23]) or it can be a request for acquired data coming from en external source (e.g. radio request for data [24]).

Then, we can model the energy consumption of the acquisition component using Eq. (10).

$$\mathcal{E}_{ACQ} = \begin{cases} \mathcal{E}_{SMP} \cdot N_S & (Regular) \\ \mathcal{E}_{SMP} \cdot N'_S \cdot Pr(e) & (Event) \end{cases} \tag{10}$$

In this expression, $\mathcal{E}_{SMP}$ is the energy needed to acquire one sample (see Fig. 6), and N_S is the number of samples taken during one regular sensing interval. For event-driven applications, $Pr(e)$ is the probability of an event occurring in one sensing interval, and N'_S is the number of samples taken following the occurrence of an event.

Obviously, the model can be generalized in order to account for more than one regular sensing interval (with different periods and sampling requirements), as well as various event types.

C. Modeling Local Data Processing Energy

To estimate the energy drained from the battery by an application task, we adopt a method proposed and validated originally in [25]. Starting from a high level description of the algorithm (e.g. Matlab/Octave), the number of operations to process the original sensed signal is recorded, accounting basically for the number of arithmetic operations: additions, multiplications, divisions and comparisons, which are the main actors in signal processing loops. Thus, depending on the
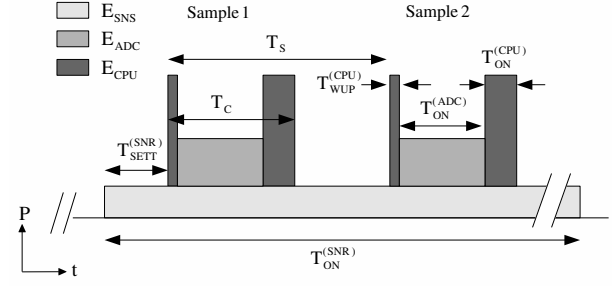
(a) Model consumption scheme
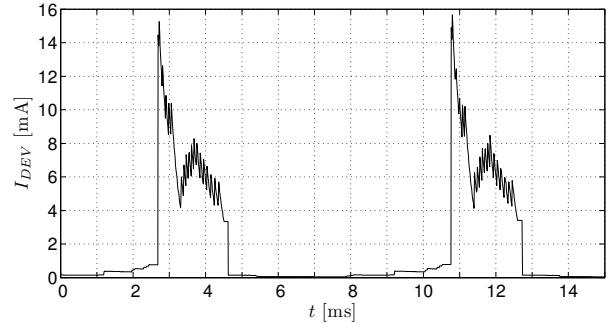
(b) Measured consumption

Fig. 6. Typical energy consumption breakdown for acquiring a couple of samples . In the model scheme (a), T_C is the total capture time; T_S the sampling period; $T_{SETT}^{(SNS)}$ the sensor setting time (time for the sensor to stabilize and start capturing data); $T_{WP}^{(CPU)}$ processor Wake-up time and start acquisition; $T_{ON}^{(ADC)}$ time of the ADC Conversion; $T_{ON}^{(CPU)}$ time the processor reads sample from ADC and stores it in memory (b) Shows a real record of the current consumed to acquire two samples. [Legend Rev.]

selected hardware architecture, these counters are mapped into the corresponding number of microcontroller (μC) clock cycles, and subsequently the latter is mapped into the corresponding energy expenditure.

This method offers accurate results as long as the CPU tasks rely mainly on arithmetic instructions (as digital signal processing algorithms do). However, it is no longer applicable when the μC is involved in non-arithmetic based tasks, like dealing with a protocol stack. In that case, alternative methods based on Instruction Set Simulators (ISS) can be used. [14].

III. EXTRACTION OF TECHNOLOGICAL PARAMETERS

For the sake of analytical tractability, we propose an intermediate fitting step to find a simple closed-form expression of each of the individual contributions. As this step requires experimental measurements of the actual platform, it can be skipped in early stage developments (i.e., when the platform is not yet available), or if the required time and resources do not justify the additional benefit. Yet, this step is highly recommended since fitting can significantly improve the accuracy of the model.

Before starting with measurements, it is worth noting a few considerations. First, consumption is directly measured in terms of *current*, not power. Therefore, an appropriate conversion must be applied to use all the above formulation. The actual power drained from the buffer is measured according to

$\mathcal{P}_{DEV}{=}I_{DEV}{\cdot}V_{BUF}$, where I_{DEV} is defined as the current measured at the output of the energy buffer, and V_{BUF} the voltage in its terminals (see Fig. 1) Moreover, the reference values in the datasheets of the components (chips, batteries, scavengers, etc.) are typically expressed in intensity units. However power on the component side is relative to the local voltage. Then, for each individual component i connected to the power domain j, the power is obtained according to: $\mathcal{P}^{(i,j)}{=}I^{(i)}V^{(j)}$

Obviously, all magnitudes must be compared in the same power domain, i.e., voltage level. As measurements are easier at the buffer output, before power reaches the regulators or DC/DC converters or regulators,, it is recommended to operate on the buffer side. In fact, for any practical estimation, what actually really matters is the load at the energy buffer.

The class of DC/DC converter and regulator determines how is the jump from one power domain to another. *Switched-mode* converters basically preserve power (with some losses being parametrized by their efficiency factor η). Then, to interpret component currents as battery loads the proper conversion is given by:

$$Switched\ DC/DC:$$
$$\mathcal{P}_{OUT}{=}\eta\mathcal{P}_{IN} \Rightarrow I_{DEV}^{(i)} = I^{(i)}\frac{1}{\eta}\left(\frac{V^{(j)}}{V_{BUF}}\right)$$

Linear converters roughly preserve currents, provided that the minimum required voltage dropout δ is respected ($V_{OUT}{>}V_{IN} + \delta$). Then, currents measured on the buffer side and the actual currents of the device are approximately the same:

$$Linear\ DC/DC:$$
$$I_{OUT} \cong I_{IN} \Rightarrow I_{DEV}^{(i)} \cong I^{(i)}$$

While respecting these rules, models can be described in current units I instead of power $\mathcal{P}$, and charge units Q instead of energy $\mathcal{E}$ (i.e., normalized by the voltage). This method avoids continuous conversions and facilitates experimental measurements. When all the components share the same power domain, the conversion is almost direct.

A. Network Profiling

Profiling a valid model for the wireless communications requires first figuring out the functional dependence on selected control parameters. The procedure is in essence the same for all practical networks. First, one should identify a suitable control parameter N_X. Then, one can fit experimental data to the analytical function or polynomial approximation $\mathcal{H}(N_X)$ chosen to model $h()$ in Eq. (8). The following examples illustrate the procedure.
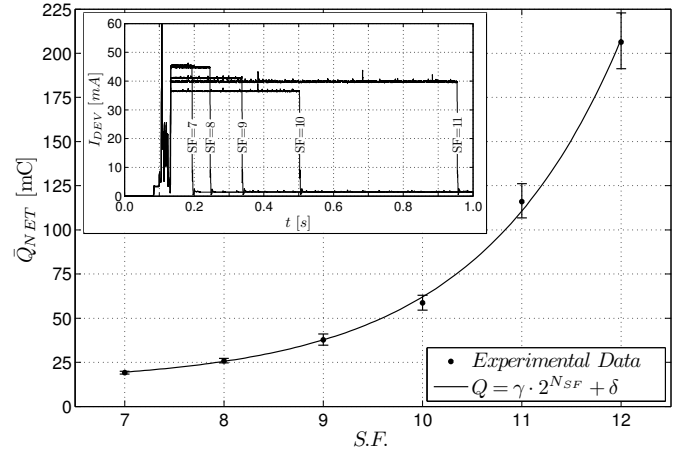

Fig. 7. Fitting LoRa spreading-factor to Eq. (11). [Legend Rev.]

1) Point to Point Communications: For the case of single-hop networks we are going to use LoRa by Semtech as an illustrative example. As mentioned before, LoRa uses different spreading factors to tune the range and consumption of the transmissions. Although it is a very specific technology, similar fitting strategies can be applied to other technologies.

To increase the range, LoRa uses configurable Spreading Factor (SF), or the ratio between the chip rate and the symbol rate. This SF parameter can be configured from SF6 to SF12 (64 to 4096 chips/symbol since in LoRa modulation is performed by representing each bit of payload information by multiple chips of information), with an increase in the link budget of 14 dB in the highest SF. This ends with a reduction in bit-ratio, which affects the time needed to send a payload, and hence, the power consumed in each transmission.

Fig. 4 shows the current measured for transmissions with different spreading factors. Basically, each step in the spreading factor scale doubles the time that radio spends in *active* state. This suggests that the charge per message can be characterized by an exponential function in the form $\mathcal{H}(N){=}\mathcal{O}(h(N)){=}2^N$, leading to Eq. (11) as a tentative fitting function:

$$\bar{Q}_{NET} \cong \bar{Q}_{MSG} \cdot 2^{N_{SF}} + \bar{Q}_B \qquad (11)$$

In the experiment shown in Fig. 7, the model is fitted with a trial set of $\sim$100 samples for each modulation. The error bars represent the empirical dispersion obtained. Results demonstrate that the postulated model is in full agreement with measurements within the experimental error. Numerical values obtained are $Q_{MSG} = 6.1\,\mu C$ and $\bar{Q}_B = 13.0\,\mu C$ with RMS Relative Error $= 6.4\%$.

2) Time Synchronized Networks: To obtain a suitable fitting function, Eq. (9) points out that the average current is the ratio between the charge of the slot-frame Q_{SF} and the period T_{SF}. The charge can be roughly estimated based on the number of *active* slots and the charge per *active* slot $Q_{SF}{\cong}Q_{MSG}{\cdot}N_{ACT}$; whilst the length of the slot-frame is determined by the number of slots and the duration of each $T_{SF}{=}T_{SLOT}{\cdot}N_{SLOTS}$ [22]. Then, the average current can be

approximated by Eq. (12), where $\bar{I}_B$ represents the background activity of the μC to control the network (periodic wake-ups, synchronization messages, etc.), and it can be considered constant.
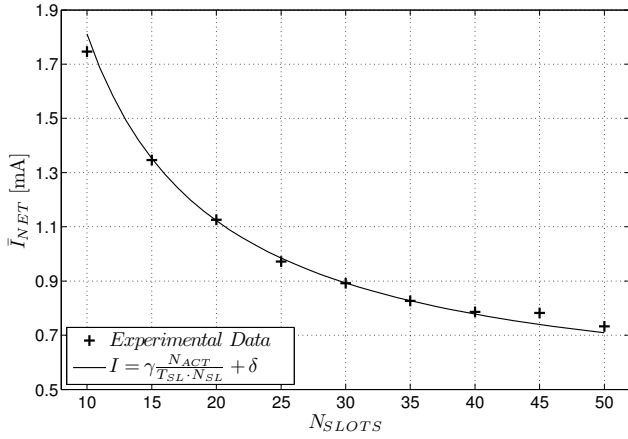

Fig. 8. Fitting parameters of TSCH networks to Eq. (12). [Legend Rev.]

$$\bar{I}_{NET} \cong \frac{\bar{Q}_{MSG} \cdot N_{ACT}}{T_{SLOT} \cdot N_{SLOTS}} + \bar{I}_B \qquad (12)$$

In Eq. (12), the duration of a timeslot T_{SLOT} is a fixed network parameter. $\bar{Q}_{MSG}$ represents the average charge required per packet, and depends basically on the radio technology. This means that, once the number of *active* slots in the slotframe N_{ACT} are scheduled, the number of slots N_{SLOTS} in the frame becomes the control parameter for the network energy, giving the characteristic functional dependency $I_{NET} \propto 1/N_{SLOTS}$ as observed in Fig. 8.

Fig. 8 shows the empirical fitting of Eq. (12) obtained for a GINA platform. This platform is based on a 16 bit MSP430F2618 μC with a IEEE802.15.4-compliant AT86RF231 radio transceiver by Atmel and a ST-LIS344ALHTR 3-axis accelerometer. Details about the platform and numerical results can be found in [26].
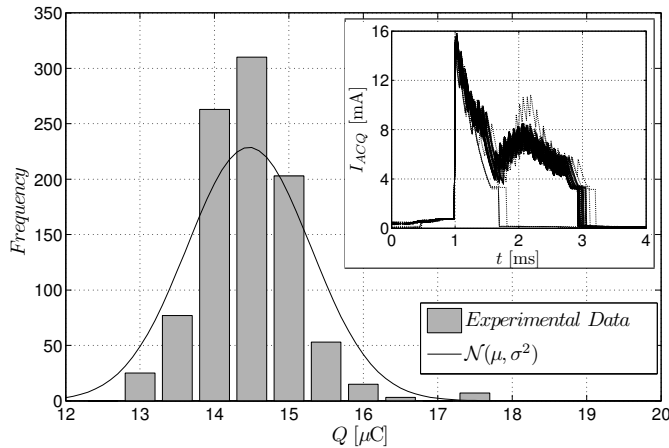

Fig. 9. Fitting sampling charge of a sensor to a normal distribution $\mathcal{N}(\mu, \sigma)$. [Legend Rev.]

B. Sampling Characterization

The current drained by the acquisition block can be reasonably approximated by Eq. (13), where the charge to get N_S samples of a record is averaged over the time elapsed between consecutive records T_{RCD}, i.e., the wake-up period. In this expression, $\bar{Q}_{SNR}$ can be interpreted as the average charge to get one sample, and comprise both the sensor and the processor contributions (including the ADC conversion) represented in Fig. 6a, while $\bar{I}_Q$ accounts for the stand-by or quiescent current of the sensor.

$$\bar{I}_{ACQ} \cong \frac{\bar{Q}_{SNR} \cdot N_S}{T_{RCD}} + \bar{I}_Q \qquad (13)$$

While in many situations the sampling period T_S is determined by the underlying physical magnitude and filtering requirements (e.g. anti-aliasing Low-Pass Filters), the time between consecutive records T_{RCD} is scheduled from the application layer, thus providing a mechanism for balancing energy consumption and sensing accuracy.

The value of $\bar{Q}_{SNR}$ can be obtained by fitting a set of experimental samples. To illustrate the procedure, Fig. 9 summarizes an experiment performed to characterize a digital magnetometer sensor. The inset plot shows the current measured during the acquisition process, with multiple samples superimposed to portray the variability between them. The charge of each individual sample is obtained by integrating the measured current. The main plot in Fig. 9 shows the empirical distribution of a dataset of $\sim$1000 samples, fitted to a normal distribution with $\mu_Q \pm \sigma_Q = 14.5 \pm 0.8 \mu C$. The average charge per sample can thus be approximated to the mean value of the distribution $\bar{Q}_{SNR} \approx \mu_Q$.
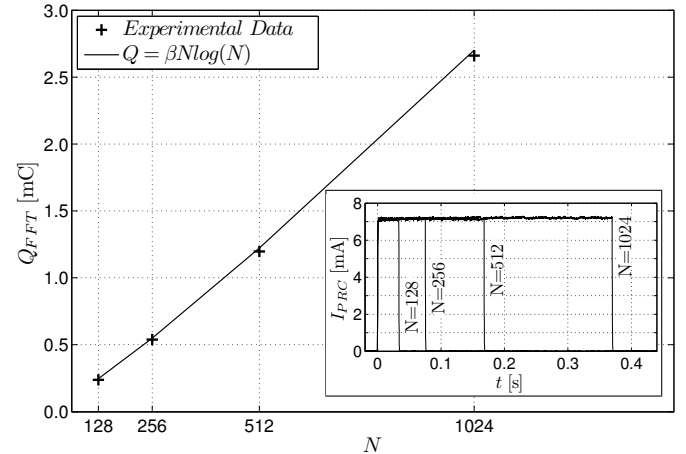
C. Processing Profiling

Fig. 10. Fitting parameters of FFT computing to Eq. (14). The inset shows a record of the current consumed to perform a FFT calculation with increasing number of samples. [Legend Rev.]

To extract a valid model for the processing contribution, it is important first to identify the parameters that rule the algorithm behavior. For already known algorithms a natural choice is the (worst-case) *time complexity* $\mathcal{T}(n)$. On the other

hand, if the algorithm is custom designed or its complexity is unknown, the fitting function should be inferred from the experimental data or simulator.

To illustrate this idea with an example we use the *Fast Fourier Transform* (FFT) algorithm. The FFT has a well known $\mathcal{T}(N)=\mathcal{O}(f(N))=N\,log(N)$ complexity. Accordingly, the associated processing time should be proportional to this relation. Considering $\bar{Q}_{OP}$ to be an estimate of the average cost per arithmetic operation, Eq. (14) can fit the processing power consumption with reasonable accuracy, as shown in Fig. 10.

$$\bar{I}_{PRC} \cong \frac{k\,\bar{Q}_{OP} \cdot N\,log(N)}{T_{RCD}} + \bar{I}_{SYS} \qquad (14)$$

In Eq. (14), k is a constant factor that depends on the algorithm's implementation, in this case, it is related to the number of arithmetic operations per FFT point. In turn, $\bar{I}_{SYS}$ includes all system related functionalities of the μC, such as running the operating system itself, managing periodic interrupts, etc.

D. Putting the Pieces Together

The last step in the modeling process consist of merging all contributions into one single expression. Followed by examples, Eq. (15) combines the contributions coming from sections III-B, III-A and III-C, keeping technological and application parameters as independent variables.

$$\bar{I}_{DEV} = \frac{\alpha N_S}{T_{RCD}} + \frac{\beta \cdot \mathcal{T}(N_P)}{T_{RCD}} + \frac{\gamma \cdot \mathcal{H}(N_A)}{T_{MSG}} + \delta \qquad (15)$$

Constants α, β, γ and δ depend only on the particular choice of sensor, μC and radio technologies. Recalling the meaning of each individual contribution from the fitting process, α can be interpreted as the charge per sample $\bar{Q}_S$, β is related with the cost per operation $\bar{Q}_{OP}$ associated with the specific μC and algorithm, while γ is an estimator of the average charge per message $\bar{Q}_{MSG}$. All constant contributions related with system activity have been gathered in the δ term. Arranged in this way, Eq. (15) allows for a straightforward evaluation of alternative technologies by simply finding the characteristic values for this set of parameters.

In turn, T_{MSG}, N_i or T_{RCD} are application parameters that can be tuned in order to meet the specifications, once the specific technology has been established. In Eq. (15), N_S stands for the number of samples effectively acquired in each sampling interval, N_P parametrizes the amount of data to be processed, and N_A represents any parameter related to radio activity.

The benefits drawn from this methodology relay on reaching Eq. (15) starting from the vague condition asserted in Eq. (2). Once this general expression is properly interpreted according to the specific platform (technological parameters) and application (operational parameters), the outcome naturally emerges when exploiting by simulation the analytical model built, i.e., the particular instance derived from Eq. (15). Next sections present two case-studies to demonstrate the applicability of the presented methodology.
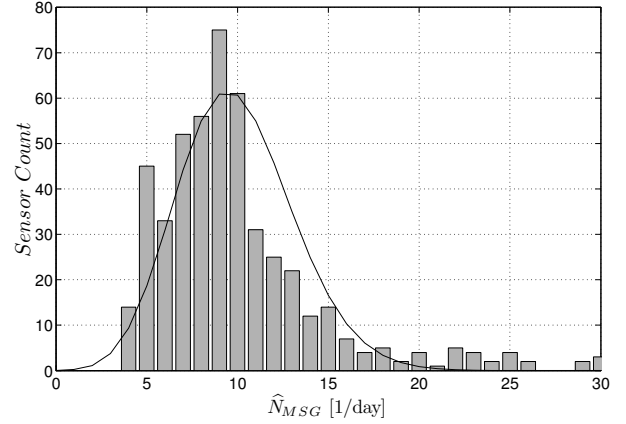

Fig. 11. Traffic generated by $\sim$500 parking spot sensors. Data was collected over 100 days.

IV. CASE STUDY I: PERIODIC REPORTING APPLICATIONS

A. Network Scenario

New radio transceivers are evolving towards long range modulation techniques while maintaining low energy consumption, thus being suitable for battery powered devices and becoming true enablers of the IoT.

A notable example of this new technological paradigm is Weightless, an industry consortium originally founded by NEUL with more than 1.000 members [21]. Weightless fosters the development of wide-area communication in white spaces at sub-GHz ISM band, covering ranges of up to 10km. This communication scheme is based on costly and powerful base-stations managing the whole network. The devices are synchronized and can send packets only in their time-slot. The packets are acknowledged to reduce packet loss and give feedback about transmission parameters (transmission power, channel, spread-factor, etc.).

Another example of wide-range wireless connectivity for M2M (Machine-to-Machine) is SIGFOX [18], which uses a simple radio technology known as Ultra Narrow Band (UNB), and operates in the license-free ISM frequency band of 868 MHz. As a MAC protocol, it uses retransmission of each packet in order to avoid packet loss. This technology is not bidirectional yet (only up-link is available).

Cycleo (now Semtech) is another provider of low-power, wide-area equipment operating in the sub GHz ISM band [20]. It is based on their proprietary approach called LoRa. This network topology is based on a base-station listening in several bands. Synchronized devices can send packets in its own time-slot and wait for the ACK of every packet. This ACK packets can carry some feedback information and the notification of an incoming downlink packet.

All these examples are practical realizations of the network scenario studied in this case study.

B. Application Scenario

Smart-parking systems provide an excellent application example for this case study, based on the previous network approaches. On-street parking sensors are small devices used

to monitor the availability of parking spots. Each device periodically wakes-up to check the state of the spot. When a car parks above, its presence is detected and the sensor reports the event to a gateway. The bandwidth required for this application is particularly low, and thus perfectly suited for long-range radio technologies. The typical interval time between radio messages oscillates from some minutes to several hours, and the information required per message is very small, as the state can be codified with just 1 bit.

Data generated from one single parking sensor is unpredictable; however, when data coming from a set of similar sensors is aggregated, a typical Poisson-like distribution emerges. Fig. 11 shows an example of the empirical distribution found in a parking application. The histogram was obtained from the events gathered by a set of sensors belonging located in the same area and operating over several days. The solid line represents the fitted Poisson distribution. In this case, the expected value of the number of messages per day is given by the Poisson mean $\mathrm{E}[N_{MSG}]=\lambda_{MSG}$, and it is used to estimate the average elapsed time between messages $\widehat{T}_{MSG}=1/\lambda_{MSG}$, required for Eq. (3).

C. Model Description

In this kind of application, a record of N_S samples is acquired with a fixed interval time T_{RCD}. In addition, information is reported to the data collector center, with an update period characterized by T_{MSG}. In this simple approach, the reported information is aggregated in a unique message. We also assume that this message is retransmitted a certain number of times N_R to increase the probability of success, thus following the SIGFOX approach, as an alternative to implementing an acknowledgment scheme over a downlink of a LoRa-like approaches. In this case, the functional dependence associated with network retransmission is trivial $\mathcal{H}(N_R)=N_R$, and γ is easily interpreted as the charge per message $\bar{Q}_{MSG}$.

Usually, the number of samples acquired N_S equals the number of samples processed N_P, thus we redefine this number as N from this point on. This means that the N parameter affects simultaneously consumption terms of both sensing and processing tasks. However, as this particular case is just a reporting application, the cost associated with processing is very low, and the associated term can be omitted. Then, Eq. (16) combines Eq. (6) and Eq. (13) in a basic instance of Eq. (15).

$$\bar{I}_{DEV} = \frac{\alpha N}{T_{RCD}} + \frac{\gamma N_R}{T_{MSG}} + \delta \qquad (16)$$

D. Simulation Results

Fig. 12 presents a simulation obtained by applying Eq. (16) to different sampling T_{RCD} and reporting period T_{MSG} configurations. The bars present the contribution to the energy consumption of the network and sampling components, according to the time elapsed between consecutive messages and the sampling rate. The floor level is associated with system management (e.g. periodic interrupts of the operating system). Although it is constant, this contribution carries important
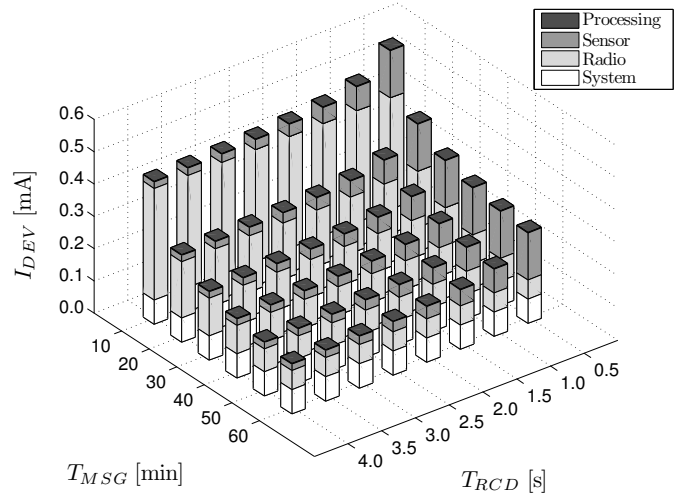

Fig. 12. Simulated consumption for a simple periodic-reporting application as function of T_{MSG} and T_{RCD}. [Legend Rev.]

weight in this application. The main reason for this is the low radio activity of the application that makes radio contribution a non-dominant term (in contrast with the usual assumption in the literature). Processing cost is also represented, despite being negligible for this particular applications.

Asymptotic behavior of Eq. (16) is evident in both axes of Fig. 12. By maintaining a fixed recording interval time, the overall energy consumption is reduced when radio activity is lower. However, the amount of energy that can be saved is limited by the asymptotic decreasing. At a certain point, increasing the elapsed time between messages does not significantly reduce the consumption. Analogously, as the recording interval increases, the energy savings decrease.

This graphical representation is useful as a tool for determining which control parameters are accountable for the highest energy savings, as well as figuring out the achievable gains and limits for optimization.

E. Validation

In order to validate the previous estimations of the power consumption, a set of measurements have been performed on a real hardware system. The employed platform was composed of a Cortex-M4 32 bit μC, the Telecom Designs TD1202 long-range radio module, and the integrated Honeywell HMC5883 digital compass. The RTOS wokes-up periodically with a *systick* period of 1 ms.

The application was configured with different sampling intervals T_{RCD}, and different reporting periodicity T_{MSG}. Fig. 13 compares the experimental results (dots) with those predicted by the model in Eq. (16) (vertical bars). Error bars account for the statistical deviation in the fitting process. Clearly, the prediction of the model agree with the experimental determinations of the system consumption, within the experimental uncertainty.

This example has shown the work-flow with a few concrete steps to have a valid consumption model of a given platform.

It also showed the right outcome of the model in form of the consumption foreseen for a specific application.
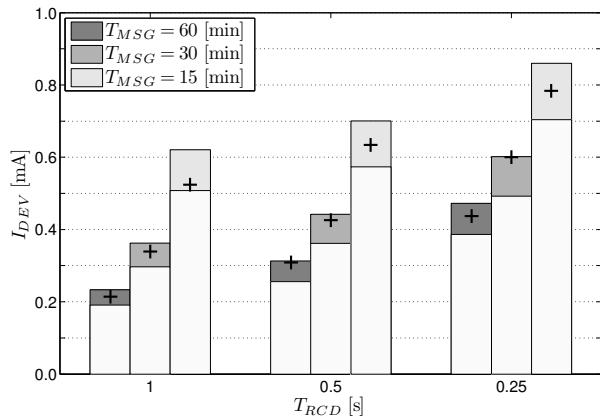

Fig. 13. Model validation of a simple periodic-reporting application [Legend Rev.]

V. CASE STUDY II: TIME SLOTTED CHANNEL HOPPING NETWORKS

A. Network Scenario

Industrial Wireless Mesh networks are being consolidated by standardization efforts under the Time Slotted Channel Hopping (TSCH) scheme. This technique has been adopted by major industrial low-power wireless standards such as WirelessHART [27], ISA100.11a [28], and, more recently, as a part of the IEEE802.15.4e standard [29]. As of today, several commercial low-power wireless networking providers are offering almost 100% reliable MAC layers, e.g [30], that provide radio duty cycles well below 1%, thereby reducing power consumption, and increasing network lifetime. This is facilitating the introduction of new monitoring and actuating devices that aim to improve the security, process automation, efficiency and productivity of the industries; Furthermore it devises a clear roadmap for the Industrial Internet paradigm. Nowadays industrial wireless communications is considered a mature technology.

B. Application Scenario

Notable examples of industrial wireless systems are vibrational analysis of rotary machines [26], structural health monitoring through harmonic analysis, e.g. accelerometers for monitoring power-line towers [31], and vibrating wire strain gauges [32] for measuring infrastructures. All these examples involve some kind of frequency analysis that can be performed by means of the FFT algorithm, which has been chosen for illustrative purposes in this second case-study.

C. Model Description

This second example models an application that requires arithmetic computing for the FFT. On top, the processor manages communications through a TSCH network. Therefore, Eq. (17) combines the contribution of Eq. (13), Eq. (12) and Eq. (14)

$$\bar{I}_{DEV} = \frac{\alpha N}{T_{RCD}} + \frac{\beta N log(N)}{T_{RCD}} + \frac{\gamma N_{ACT}}{T_{SLOT} N_{SLOTS}} + \delta \quad (17)$$

The main parameter involved in network consumption now is N_{SLOTS}, which is related to the number of *active* and *sleep* slots. Assuming a fix number of *active* slots, by incrementing N_{SLOTS} we are introducing *sleep* slots to the schedule; therefore reducing the average consumption, although sacrificing bandwidth and latency.

Following Fig. 2, the processor periodically wakes up, takes N_S samples, and analyzes them. Again, the number of points computed by the FFT N_P, and the number of samples read by the ADC N_S are the same. So, this parameter simultaneously affects both contributions, and it is denoted simply by N. Fixed N is the duty-cycled behavior of the application and it makes the time between records T_{RCD} the fundamental parameter for controlling the average power. Specifically, as the time between records increases, less power is consumed. Therefore, the time interval between consecutive records determines the time scale for power averaging.
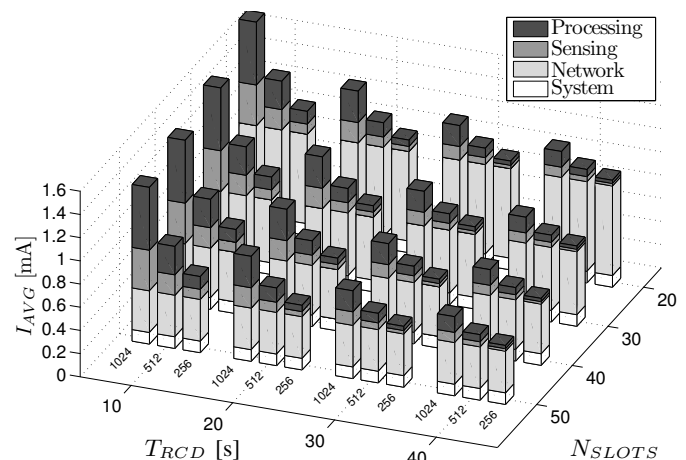
D. Simulation Results

Fig. 14. Simulated consumption for an application using a TSCH network , as function of N_{SLOTS} and T_{RCD}, for different number of samples in the FFT calculation N_{FFT}. [Legend Rev.]

Fig. 14 presents the consumption values obtained by applying Eq. (17) to different network N_{SLOTS} and recording period configurations T_{RCD}, considering N_{FFT} = 256, 512 and 1024 samples per record. The bars present the contribution to the energy consumption of the network, sampling and processing components; according to the number of slots per slot-frame and the recording interval.

Again, asymptotic behavior appears in both axes of Fig. 14. While maintaining a fixed interval time, the overall energy consumption is reduced by increasing the number of slots in a slot-frame. Still, the asymptotic behavior sets a limit to the power savings: at a certain point, increasing the number of slots in the network does not significantly reduce the energy consumed.

E. Validation

Experiments were carried out using a GINA platform [33], and running the OpenWSN protocol stack [34]. The GINA platform comprises several inertial sensors for angular rate and linear acceleration along with a general purpose microprocessor. Specifically, constants α, β, γ and δ of Eq. (17) have been characterized for the Texas Instruments MSP430f2618 16-bit μC, Atmel AT86RF231 IEEE802.15.4 radio, and the ST-LIS344ALHTR 3-axis accelerometer sensor.

Fig. 15 compares the experimental results (dots and experimental dispersion bars) with those predicted (vertical bars) by Eq. (17) for different application configurations: different number of slots in a slot-frame N_{SLOTS}, different recording intervals T_{RCD}, and different number of samples collected and processed N_{FFT}. Again, the here-presented model quantitatively predicts the experimental trends.

This example serves to further demonstrate the applicability of the model. A device operating in a real scenario has always a degree of uncertainty associated with environmental conditions. Variations in the Packet Delivery Ratio (PDR) are just a representative example, but also other issues as temporal link interruptions, unexpected system restarts, etc. Under these circumstances, further refinements for improving the accuracy of estimates may not be necessary. Instead, the strength of the model lies in its capability to support better-informed decisions and avoid risks in the early stages of development.
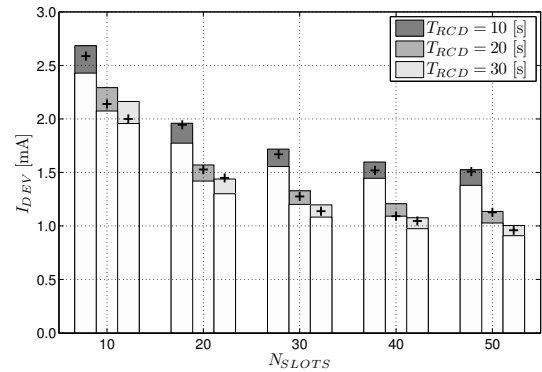
VI. CONCLUSIONS

The present paper defines a general methodology to model energy consumption of wireless network devices as a system. The model takes into account all the components that play a fundamental role in a realistic industrial application: standard networking, mainly standard networking sensing and processing technologies. Our approach bridges the gap between theoretical analysis and practical applicability by proposing a straight forward method to estimate a few key parameters related to the technology used and the operation conditions of a specific application.

The utility of the approach is illustrated with two case studies. The agreement between experiments and predictions demonstrate that the model is valid and applicable to real applications and platforms. It also shows that measuring a set of application-specific parameters is enough the make accurate estimations of the power consumption.
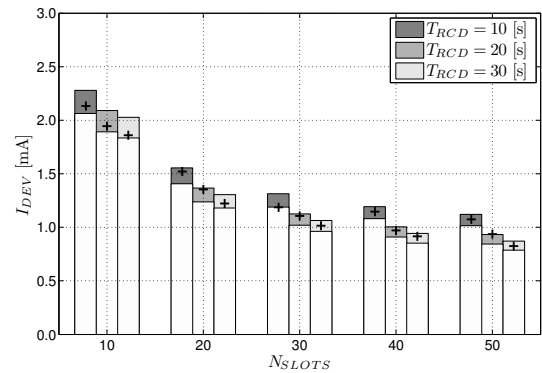
With this model, application engineers can foresee the impact of different application parameters on power consumption, even without a complete implementation of the application. Hence, this framework can help engineers to study the viability of a new application in terms of power consumption, energy harvesting needs, battery requirements, etc.

(a) $N_{FFT} = 1024$

(b) $N_{FFT} = 512$

(c) $N_{FFT} = 256$

Fig. 15. Comparison between model predictions (bars) and experimental measurements (dots) of a TSCH-based application, as a function of different configuration parameters. [Legend Rev.]
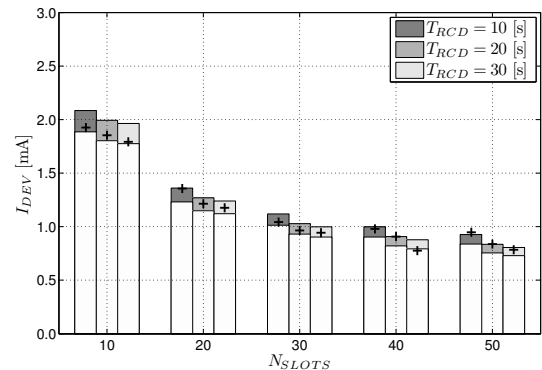
REFERENCES

[1] L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," *Computer Networks*, vol. 54, no. 15, pp. 2787 – 2805, 2010.

[2] R. Vullers, R. Schaijk, H. Visser, J. Penders, and C. Hoof, "Energy harvesting for autonomous wireless sensor networks," *Solid-State Circuits Magazine, IEEE*, vol. 2, no. 2, pp. 29–38, Spring 2010.

[3] A. Kansal, J. Hsu, S. Zahedi, and M. B. Srivastava, "Power management in energy harvesting sensor networks," *ACM Transactions on Embedded Computing Systems (TECS)*, vol. 6, no. 4, p. 32, 2007.

[4] V. Boicea, "Energy storage technologies: The past and the present," *Proceedings of the IEEE*, vol. 102, no. 11, pp. 1777–1794, Nov 2014.

[5] M. Belleville, H. Fanet, P. Fiorini, P. Nicole, M. Pelgrom, C. Piguet, R. Hahn, C. V. Hoof, R. Vullers, M. Tartagni, and E. Cantatore, "Energy autonomous sensor systems: Towards a ubiquitous sensor technology," *Microelectronics Journal*, vol. 41, no. 11, pp. 740 – 745, 2010, {IEEE} International Workshop on Advances in Sensors and Interfaces 2009.

[6] D. Gunduz, K. Stamatiou, N. Michelusi, and M. Zorzi, "Designing intelligent energy harvesting communication systems," *Communications Magazine, IEEE*, vol. 52, no. 1, pp. 210–216, January 2014.

[7] E. W. Dijkstra, "Information streams sharing a finite buffer." *Inf. Process. Lett.*, vol. 1, no. 5, pp. 179–180, 1972. [Online]. Available: http://dblp.uni-trier.de/db/journals/ipl/ipl1.html

[8] K. Tutuncuoglu and A. Yener, "Optimum transmission policies for

battery limited energy harvesting nodes," *Wireless Communications, IEEE Transactions on*, vol. 11, no. 3, pp. 1180–1189, March 2012.

[9] N. Michelusi, K. Stamatiou, and M. Zorzi, "Transmission policies for energy harvesting sensors with time-correlated energy supply," *Communications, IEEE Transactions on*, vol. 61, no. 7, pp. 2988–3001, July 2013.

[10] N. Bui and M. Rossi, "Staying alive: System design for self-sufficient sensor networks," *ACM Transactions on Sensor Networks*, 2015.

[11] X. Jiang, J. Polastre, and D. Culler, "Perpetual environmentally powered sensor networks," in *Information Processing in Sensor Networks, 2005. IPSN 2005. Fourth International Symposium on*, April 2005, pp. 463–468.

[12] P. Blasco, D. Gunduz, and M. Dohler, "A learning theoretic approach to energy harvesting communication system optimization," in *Globecom Workshops (GC Wkshps), 2012 IEEE*, Dec 2012, pp. 1657–1662.

[13] Q. Wang, M. Hempstead, and W. Yang, "A realistic power consumption model for wireless sensor network devices," in *Sensor and Ad Hoc Communications and Networks, 2006. SECON '06. 2006 3rd Annual IEEE Communications Society on*, vol. 1, 2006, pp. 286–295.

[14] M. Bazzaz, M. Salehi, and A. Ejlali, "An accurate instruction-level energy estimation model and tool for embedded systems," *Instrum. Meas., IEEE Trans. on*, vol. 62, no. 7, pp. 1927–1934, July 2013.

[15] V. Konstantakos, A. Chatzigeorgiou, S. Nikolaidis, and T. Laopoulos, "Energy consumption estimation in embedded systems," *Instrum. Meas., IEEE Trans. on*, vol. 57, no. 4, pp. 797–804, April 2008.

[16] R. Torah, P. Glynne-Jones, M. Tudor, T. O'Donnell, S. Roy, and S. Beeby, "Self-powered autonomous wireless sensor node using vibration energy harvesting," *Measurement Science and Technology*, vol. 19, no. 12, p. 125202, 2008.

[17] V. Gungor and G. Hancke, "Industrial wireless sensor networks: Challenges, design principles, and technical approaches," *Industrial Electronics, IEEE Transactions on*, vol. 56, no. 10, pp. 4258–4265, Oct 2009.

[18] SigFox, "About SigFox," http://www.sigfox.com/en/about, 2014.

[19] W. Webb, "Weightless: The technology to finally realise the m2m vision," *Int. J. Interdiscip. Telecommun. Netw.*, vol. 4, no. 2, pp. 30–37, apr 2012.

[20] Semtech, "LoRa Product Family," http://www.semtech.com/wireless-rf/lora.html, 2014.

[21] Weightless, "Weightless SIG for M2M and Internet of Things IOT," http://www.weightless.org/.

[22] X. Vilajosana, Q. Wang, F. Chraim, T. Watteyne, T. Chang, and K. Pister, "A realistic energy consumption model for tsch networks," *Sensors Journal, IEEE*, vol. 14, no. 2, pp. 482–489, Feb 2014.

[23] D. Donoho, "Compressed sensing," *Information Theory, IEEE Transactions on*, vol. 52, no. 4, pp. 1289–1306, April 2006.

[24] G. Marrocco, "Pervasive electromagnetics: sensing paradigms by passive rfid technology," *Wireless Communications, IEEE*, vol. 17, no. 6, pp. 10–17, December 2010.

[25] D. Zordan, B. Martinez, I. Vilajosana, and M. Rossi, "On the performance of lossy compression schemes for energy constrained sensor networking," *ACM Trans. Sen. Netw.*, vol. 11, no. 1, pp. 15:1–15:34, aug 2014.

[26] B. Martinez, X. Vilajosana, F. Chraim, I. Vilajosana, and K. Pister, "When scavengers meet industrial wireless," *Industrial Electronics, IEEE Transactions on*, vol. PP, no. 99, pp. 1–1, 2014.

[27] *WirelessHART Specification 75: TDMA Data-Link Layer*, HART Communication Foundation Std., Rev. 1.1, 2008, hCF_SPEC-75.

[28] ISA, *ISA-100.11a-2011: Wireless Systems for Industrial Automation: Process Control and Related Applications*, International Society of Automation (ISA) Std., May 2011.

[29] *802.15.4e-2012: IEEE Standard for Local and metropolitan area networks–Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs) Amendment 1: MAC sublayer*, IEEE Std., 16 April 2012.

[30] L. Doherty, W. Lindsay, and J. Simon, "Channel-specific wireless sensor network path data," in *ICCCN*, 2007, pp. 89–94.

[31] T. Yin, H. Lam, H. Chow, and H. Zhu, "Dynamic reduction-based structural damage detection of transmission tower utilizing ambient vibration data," *Engineering Structures*, vol. 31, no. 9, pp. 2009 – 2019, 2009.

[32] J. Brownjohn, "Structural health monitoring of civil infrastructure," *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, vol. 365, no. 1851, pp. 589–622, 2007.

[33] A. Mehta and K. Pister, "WARPWING: A Complete Open-Source Control Platform for Miniature Robots," in *International Conference on Intelligent Robots and Systems (IROS)*. IEEE/RSJ, 2010.

[34] T. Watteyne, X. Vilajosana, B. Kerkez, F. Chraim, K. Weekly, Q. Wang, S. D. Glaser, and K. Pister, "OpenWSN: a Standards-based Low-power Wireless Development Environment," *Transactions on Emerging Telecommunications Technologies*, vol. 23, no. 5, pp. 480–493, 2012.