Model-based energy characterization of IoT system design aspects

Alexios Lekidis[1] and Panagiotis Katsaros[1]

Department of Informatics,
Aristotle University of Thessaloniki
54124 Thessaloniki, Greece
{alekidis, katsaros}@csd.auth.gr

Abstract. The advances towards IoT systems with increased autonomy support improvements to existing applications and open new perspectives for other application domains. However, the design of IoT systems is challenging, due to the multiple design aspects that need to be considered. Connectivity and storage aspects are amongst the most significant ones, as IoT devices are resource-constrained and in many cases battery-powered. On top of them, it is also essential to consider privacy and security aspects that are linked to the protection of the IoT system, as well as of the data exchanged through its connectivity interfaces. Ensuring security in an IoT system, though, is an evident need and a complex challenge, due to its impact in the battery lifetime. In this paper, we propose a methodology to manage energy consumption through a model-based approach for the energy characterization of IoT design aspects using the BIP (Behavior, Interaction, Priority) component framework. Our approach is exemplified based on an Intelligent Transport System (ITS) that uses Zolertia Zoul devices placed in traffic lights and road signs to broadcast environmental and road hazard information to crossing vehicles. The results allow to find a feasible design solution that respects battery lifetime and security requirements.

Keywords: Internet of Things (IoT) · energy characterization · security · model-based design

1 Introduction

The combination of connected intelligence in systems of the Internet of Things (IoT) with energy-constrained devices featuring limited computational resources poses new challenges in application design. To better control the consumption of battery lifetime, application developers have to consider a number of design aspects not only at the application level, but also at the system level. These aspects include the system's connectivity, the data processing and the data storage. Furthermore, security and data privacy aspects should be also taken into account, since the IoT has attracted the interest of malicious actors, who may attempt to tamper with the provided functionality of an IoT system. Widely known attacks as the Distributed Denial of Service (DDoS) Mirai botnet [3] on

OVH[1] have recently demonstrated the feasibility of opening system ports with default authentication credentials through remote TELNET or SSH connections.

Important risks emerge due to: (i) the available connectivity interfaces and IoT protocol implementations and (ii) the exposed web services that allow continuous service delivery but usually are not designed with security in mind. The former refer to the absence of security measures in IoT protocols (e.g. MQTT [7]), which opens possibilities to eavesdrop or tamper with the exchanged data. For the latter, the absence of security protection is due to the additive computational power and storage memory required for the implementation of security mechanisms (e.g. encryption). This may lead to memory overflow, as well as to considerably reduced battery lifetime.

In overall, the design of IoT systems is characterized by a high complexity due to the multiple overlapping design aspects, which have to be taken into account and the initial system requirements that often are not directly feasible within the IoT system architecture. As a consequence, the system requirements are usually refined multiple times until they converge to those that can be eventually implemented. The overall gap in system design could be bridged by a method for the estimation and characterization of the system's energy life-span, with respect to the design aspects considered in the initial system requirements.

To this end, we propose a systematic model-based approach that leverages the Behavior, Interaction, Priority (BIP) component framework [5] towards an energy profiling scheme for the overlapping system design aspects. Our scheme enables the prediction of tight bounds for the various design aspects by utilzing a software-based solution of the Contiki IoT operating system, called powertrace [8], which are then used to include energy constraints in the BIP models [11]. Such scheme allows to overcome challenges related to energy monitoring, as it currently requires direct hardware interactions, which in most devices are not supported [8]. Furthermore, it provides an automated energy cost analysis for IoT design aspects compared to manual energy consumption calculations, which rely on the manufacturer characteristics that might also be inaccurate according to actual system measurements [17].

The method is illustrated through an Intelligent Transport System (ITS), i.e. an application scenario selected from one of the main IoT system domains[2]. In the ITS system, road hazard and environment information are broadcasted by infrastructure components, as traffic lights and road signs. To facilitate such communication the infrastructure components include Zolertia Zoul modules[3]. In this context, we use the system requirements defined by the European Transportation Committee (ETSI)[4] to evaluate the feasibility of ITS system design aspects that are linked to IoT connectivity protocols for data processing and

[1] French cloud computing company - https://ovhcloud.com
[2] https://www.technavio.com/blog/intelligent-transport-system-iot-promote-smart-safe-urban-mobility
[3] https://zolertia.io/zoul-module/
[4] https://www.etsi.org/

storage schemes, as well as to security protocols such as lightweight implementations of TLS and DTLS. In overall, this paper has the following contributions:

- an energy profiling technique for estimating the energy impact of various design aspects in an IoT application;
- an energy-aware model allowing a feasible design solution for IoT applications given the energy cost for each design aspect;
- a use case of our energy profiling technique on an ITS system deployed on Zolertia Zoul motes.

The rest of the paper is organized as follows. Section 2 provides a brief introduction to the powertrace IoT energy measuring module, as well as our previous work on energy-aware models using the BIP framework. Section 3 illustrates the proposed energy profiling technique for deriving estimates for the IoT system design aspects, which are later used in Section 4 to evaluate the use of communication, data storage and security aspects in the ITS case study. Finally, Section 5 provides conclusions and perspectives for future work.

2 Background

2.1 Measuring energy consumption with Powetrace

Powertrace [8] is a Contiki library that allows the annotation of Contiki programs with primitives, for monitoring the energy flow in IoT devices. It identifies four individual operating modes that contribute to a device's energy consumption:

- Low Power (LPM): the device is idle waiting for an event
- CPU: the device microcontroller is used for calculations/data processing
- Radio transmission (Tx): indicating data transmission
- Radio reception (Rx): indicating data reception

The energy consumption depends on the time that a device remains in each of the above modes. To measure this time, the library provides code primitives that can be used for every IoT device type. A data logger supporting energy analytics is utilized to store the data. Examples of such analytics are the duty cycle or the device lifetime. The former refers to the percentage of time that a device remains in one operating mode, whereas the latter refers to the total time duration that a device operates autonomously. The period that powertrace uses to measure and log the data is user-configurable and has an impact on the performance and accuracy of the mechanism. Finally, the energy calculation in powertrace also supports hardware-specific parameters, such as the real-time timer (RTIMER[5]) that is used to measure the hardware clock cycles of the device per second.

2.2 Energy-aware modeling of IoT systems

In a related article [11] we have presented a systematic methodology to characterize the impact of various application design parameters to the total energy consumption of IoT devices. This methodology allows the estimation of tight

[5] http://anrg.usc.edu/contiki/index.php/Timers#Step_5_-_Introduction_to_rtimer

energy bounds for the energy consumption of an IoT application through the use of Statistical Model Checking (SMC) [14].

Since energy consumption is closely related to connectivity design aspects, the parameters found to have a high impact are related to IoT connectivity. Nevertheless, energy consumption is also impacted by additional equally important aspects and hence we focus here on revisiting the initial energy model and associated design parameters. The identified parameters in [11] were classified into (i) the application, (ii) the MAC and (iii) the physical network layers.

Our methodology is driven by a model for the system design that is based on the BIP (Behavior-Interaction-Priority) [5] executable modeling language. BIP is a particularly expressive, component-based framework with rigorous semantics. It allows the construction of complex, hierarchically structured models from atomic components, which are characterized by their behavior and interfaces. The components are transition systems enriched with data. Transitions represent state changes from a source to a destination control location. Each time a transition is taken, component data (variables) may be assigned new values, which are computed by user-defined functions (in C/C++). Atomic components are composed by layered application of interactions and priorities. Interactions express synchronization constraints and define the transfer of data between interacting components. Priorities are used to filter amongst possible interactions and to steer system evolution so as to meet performance requirements, e.g. to express scheduling policies. A set of atomic components can be composed into a generic compound component by the successive application of connectors, representing sets of interactions, and priorities.

3 Characterization of IoT system design aspects

In this section, we present a method (Fig. 1) to bridge the gap between IoT system requirements and the energy impact that they have in the context of the IoT architecture. Our method allows (1) the energy characterization of all aspects contributing to the energy consumption and (2) the design of the application with respect to those aspects.

Given as input a set of requirement specifications for the relevant design aspects, and the high-level design of the application expressed in a Domain Specific Language (DSL) [12], our method proceeds as follows:

1. **Transformation for the System Model**: The actions comprising this step are two-fold. First, the Contiki code behaviour of the application modules is specified in the DSL-based description, which is used to generate an Application Model in BIP. This model is later enhanced with the OS/kernel model that is formed from a library of BIP components. The two models are composed by incorporating information specified in the DSL description for how the application modules are deployed to the IoT system's devices.
2. **Code generation from IoT application templates**: This step leverages the DSL description and an XML-based configuration file with the parameters that are presented in Section 3.1 that affect the energy consumption.

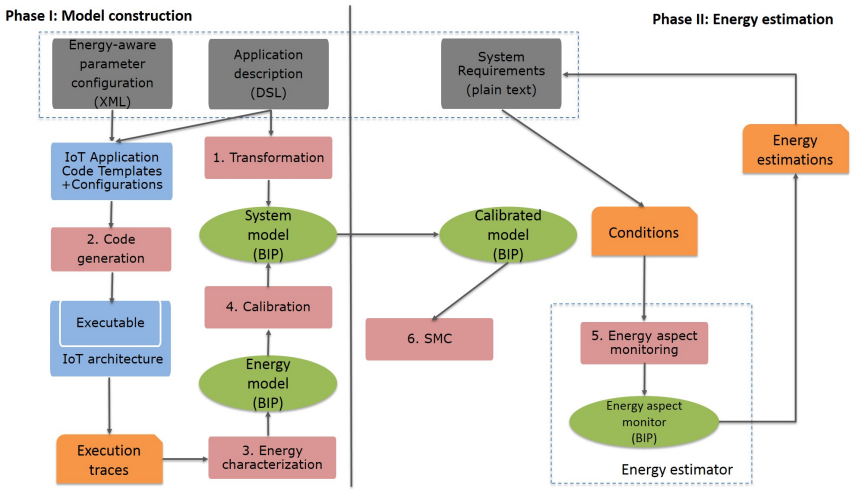

Fig. 1. Design phases of the proposed method

Both are used as inputs to instantiate Contiki code templates for the IoT application and form an executable program for the devices.

3. **Energy characterization**: The analysis of powertrace execution traces is provided to a distribution fitting technique (Section 3.2). This technique allows to associate a probabilistic distribution to the data, in order to prepare them for being injected to a BIP energy model for IoT systems. This model is presented in [11] and includes the influential hardware/software energy constraints from the execution of the IoT application.
4. **Calibration for the construction of an energy-aware System Model**: This step concerns with the addition of parameters to the BIP model for the runtime characterization of the IoT application, as well as the generation of glue code for the composition of the BIP System Model with the energy model.
5. **Energy aspect monitoring:** The Calibrated BIP Model is also connected to energy monitors (Section 3.3) for the design aspects of interest. This allows the simulation-based analysis of energy consumption such that the designer can check the feasibility of the various (combinations of) design aspects with respect to the available resources.
6. **Statistical model checking (SMC):** At the end the designer can verify if the Calibrated BIP Model satisfies the system requirements through SMC.

3.1 Energy-relevant parameters for IoT design aspects

Given that the connectivity aspects were covered in our previous work [11], this section focuses on the data processing and security aspects.

Data processing aspects

The amount of available memory in IoT devices, allows for the implementation of a limited number of IoT application features, with respect to those envisioned in system requirements. The selection of which features are necessary and should be implemented requires a characterization of their energy constraints. Given these constraints, the main parameters that influence data processing are:
Resource processing: Resources provide important information in IoT applications through the interaction with hardware sensors. Sensors are or are not built-in, in which case they are mounted in the device as peripherals (e.g. Phidgets[6] sensors for Zolertia Z1). Resource communication depends on their processing time, which when lengthy, can lead to timeout and retransmissions on the requester side. This increases substantially the energy consumed in processing and transmission mode. CoAP provides a mechanism to optimize the energy consumed in such scenario by acknowledging the resource request and then sending the response when it becomes available [16]. Overall, peripheral communication is unpredictable and thus can influence strongly IoT device energy consumption.
Routing protocol: Many IoT applications configure edge entities to route all data that require processing in remote cloud servers. This emerged a new type of IoT applications, called Software-Defined Networks (SDN) [13], where control and logic is placed on the sensors and data processing on remote cloud servers. To allow energy-efficient data routing to cloud servers, a number of protocols were developed with RPL [18] being the one widely used in IoT. RPL builds an Destination Oriented Directed Acyclic Graph (DODAG) that provides the best route from each leaf node to the edge to direct all data encapsulated in network packets. Other routing protocols, include the cognitive RPL (CORPL) [2], a variation of RPL for cognitive radio networks [1], as well as the Channel-Aware Routing Protocol (CARP) [4], a non-standardized protocol that is used for underwater communication due to its link quality considerations.
Memory block management: The limited memory available in IoT devices requires new techniques to allocate memory dynamically. Apart from the commonly used heap memory allocation using the *malloc* library[7], IoT systems also employ dynamic block allocation through the *mmem* library[8]. The latter defragments the managed memory area, which in turn allows the IoT application designer to manage the features to be implemented by estimating their block size. Additionally, by avoiding device operations when in a low memory state, a balanced energy consumption can be achieved.

Security aspects

Security is receiving substantial attention in IoT application development, due to the underlying risks especially for safety-critical systems as connected vehicles or avionics. Mechanisms as encryption or authentication offer protection against imminent threats, though they should also have a lightweight energy footprint to respect the constraints of IoT systems.

[6] http://wiki.zolertia.com/wiki/index.php/Phidgets
[7] https://en.cppreference.com/w/c/memory/malloc
[8] http://www.eistec.se/docs/contiki/a02115.html

Security level: The protection of IoT devices is managed by the security level that they offer. Security levels are categorized according to the system requirements for security aspects. The currently available levels are:
[**SL-0**] No security
[**SL-1**] Encryption only
[**SL-2**] Authentication and encryption
A higher security level results in better protected schemes, but leads to higher energy consumption. The security level is based on the system requirements.
Security protocol: The protocols allowing secure data exchange are implemented in different layers of the IoT protocol stack. Each of these protocols contributes to a security level, but it also uses a specific communication mechanism between the IoT devices. As an example, TLS uses a handshake mechanism to establish a connection by agreeing on the connection parameters and by the exchange of a secret cipher key. A similar procedure is applied for protocols of other layers such as the IPsec in the IP layer. Overall, even though these protocols offer solid encryption/authentication mechanisms they introduce a substantial overhead on the energy consumption. This is due to the time that a device remains on the processing (i.e. CPU) mode for encrypting/decrypting the packets, as well as the additive transmissions for establishing a connection.
Session key size: In traditional Internet systems, security is handled through sufficiently large key sizes through the commonly used AES encryption. Instead, in IoT a large key size (i) would increase the processing demand for encryption/decryption of messages and (ii) would prolong the time the IoT device remains in transmission mode, since the key should be distributed to the other IoT devices upon connection establishment. These considerations along with the dynamicity of the IoT environment lead to the conclusion that the key size should be considered as an important aspect when providing security for IoT devices.

3.2 Energy characterization

Energy characterization (step 3 in Fig. 1) is performed through distribution fitting, a technique to derive models that characterize input data. In our scope, distribution fitting considers that the target model is a probability distribution. This technique allows to characterize the energy evolution over a certain period, to reflect the actual energy consumption in the IoT system under study.

The technique itself is based on the randomness of input data and thus cannot be applied to deterministic or statistically correlated data. Instead of this, the data should be independent, such that one outcome of a random sample does not affect the outcome of another. This holds for energy data as IoT devices have asynchronous and not correlated changes, which is a consequence of relying in event-driven operating systems as the Contiki OS [8].

The fitting process is using well-known methods, such as moments matching and maximum likelihood. The moments matching method estimates the model parameters by using as many moments as the number of missing parameters of the candidate distribution. These moments depend on the probability law that the chosen candidate distribution follows. On the other hand, maximum

likelihood finds the parameters that maximize the likelihood function. Then, the fitted distributions are validated against the input energy data using goodness-of-fit tests, such as the Kolmogorov-Smirnov (K-S).

An example fitted distribution characterizing the energy consumed while a device is in Tx mode is illustrated in Fig. 2. Horizontal axis reflects the range in which energy values can vary, whereas the vertical illustrates the Probability Density Function (PDF). In this example, the distribution that is selected as a best fit is Generalized Pareto with $\kappa = 0.40227, \sigma = 1.6739, \mu = 35.105$ moments. For energy samples given by: $X = [x_1, x_2, ..., x_n]$, the distribution parameters θ_1 and θ_2 that maximize the likelihood function are computed as follows:

$$L(\theta_1, \theta_2; x_1, \ldots, x_n) = \prod_{i=1}^{n} \theta_1 * \frac{\theta_2^{\theta_1}}{{x_i}^{\theta_1+1}} = {\theta_1}^n * \theta_2^{n*\theta_1} \prod_{i=1}^{n} \frac{1}{{x_i}^{\theta_1+1}} \quad (1)$$

During the validation phase, the goodness-of-fit tests have given 0.09415 error for Kolmogorov-Smirnov (K-S).

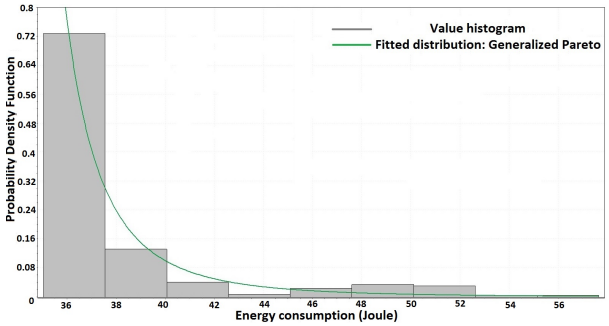

Fig. 2. Fitted energy distribution for the transmission (Tx) mode

The fitted distributions are calibrating the energy model in the form of probabilistic variables. These variables take values based on a non-deterministic selection that is following the probability law of the fitted distribution.

3.3 Energy aspect monitoring

An energy aspect monitor is instantiated according to the number of aspects that influence the IoT system. Following Section 3.1, these parameters lead to three instances of the component, namely the connectivity, data processing and security monitors. Each instance interacts with the energy model using a dedicated BIP connector as illustrated in Fig. 3. The monitor component has two main characteristics: (i) acting as an interaction advisor, such that when present it can consider the energy cost of each design aspect (ii) implementing all the required equations for evaluating if the conditions that are derived by the system requirements are met.

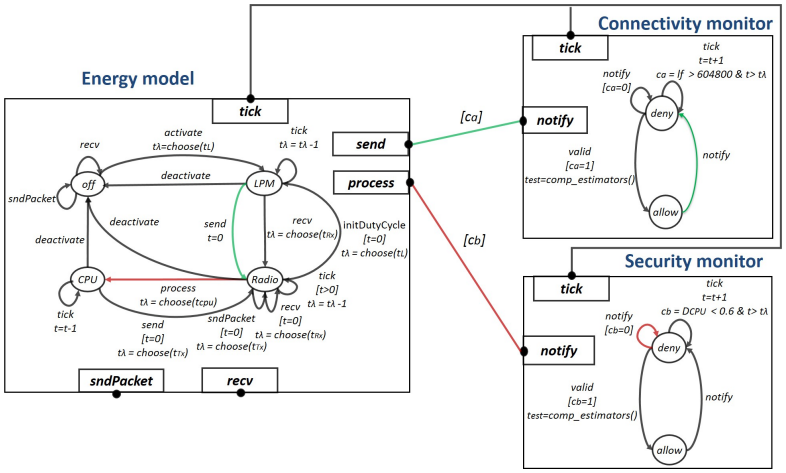

Fig. 3. Energy aspect monitor interactions with energy model

The aspect monitor is initially informed when the device switches operating mode by receiving the time value of the probabilistic distribution that was selected through the *tick* transition. With this value it can estimate if the condition can be met or not. In the example of Fig. 3 there are two monitors (i.e. connectivity and security) and each one evaluates a different condition:

1. Condition A (associated with connector value c_a): The device is sustained for full working day on battery power
2. Condition B (associated with connector value c_b): The processing time of security operations should not be higher than 60% of the overall duty cycle

Each condition is computed by the monitor during its own *tick* transition. When the condition is met, the boolean value is set and the monitor can proceed through the *valid* internal transition (connectivity monitor of Fig. 3). Otherwise, it will remain on the *deny* state until satisfying the condition. On the other hand, when the allocated time on the energy model has elapsed, it has first to interact with the monitor (through the *notify* transition) prior of interacting with the Contiki system model [11]. During this interaction (depicted in red in Fig. 3) the monitor uploads the condition value, which either allows the energy model to proceed (c_a or $c_b = true$), or notifies the energy model that the condition is invalidated and thus the system violates the requirements. When the condition is met as with condition A in Fig. 3, the time value counted by the *tick* transition of the energy model (t_{est}) is send to the monitor through the condition connector, which calculates the estimated energy consumption (in Joule) using the following equation:

$$E_{est_y} = I_y * V_y * t_{est} \tag{2}$$

where in every tuple $\{Iy, Vy\}$, y indicates the device's operating mode and I, V indicate respectively the current (in Ampere) and voltage (in Volts).

Energy estimation

After simulating the Calibrated BIP Model, the monitor provides analytics regarding the estimated values for energy characteristics, which differ from the actual ones that are logged in the execution traces of Figure 1 using the power-trace module (Section 2). These characteristics cover all the IoT design aspects of Section 3.1 and are given by the following equations:

$$E_{est_{total}} = E_{est_{conn}} + E_{est_{proc}} + E_{est_{sec}} + \sum_{i=1}^{N_{LPM}} I_{LPM} * V_{LPM} * \Delta t_{LPM_i} \quad (3)$$

Except from the last parameter (energy in LPM mode), each of the remaining equation parameters reflects the aspects presented in Section 3.1, defining their contribution to the energy consumption. Δt_y indicates the time intervals in which the device remains in an operating mode and D_y indicates the duty cycle for each mode ($y = LPM$ for LPM mode).

$$D_y = \frac{\sum_{i=1}^{N_y} I_y * V_y * \Delta t_{y_i}}{E_{est_{total}}} \quad (4)$$

where N_y the relative number of occurrences that the device has visited the operating mode y. The rest of the equation parameters are computed as:

$$E_{est_{conn}} = \sum_{j=1}^{N_{Tx}} I_{Tx} * V_{Tx} * \Delta t_{Tx_j} + \sum_{k=1}^{N_{Rx}} I_{Rx} * V_{Rx} * \Delta t_{Rx_k} \quad (5)$$

$$E_{est_{proc}} = \sum_{z=1}^{N_{CPU}} I_{CPU} * V_{CPU} * \Delta t_{CPU_z} + \sum_{w=1}^{N_{PER}} I_{PER} * V_{PER} * \Delta t_{PER_w} \quad (6)$$

Energy consumption for security aspects is linked to both connectivity and data processing aspects, however the contribution percentage for each one varies and depends on the energy parameters of the IoT application. Hence:

$$E_{est_{sec}} = \Delta E_{conn} + \Delta E_{proc} \quad (7)$$

where ΔE_{conn} and ΔE_{proc} indicate the additional overhead that is added by security aspects. Finally, the device lifetime is computed by the following equation:

$$lf_{est} = \frac{C_{batt} * Vcc}{E_{est_{total}}} \quad (8)$$

where C_{batt} indicates the overall capacity of the battery for autonomous operation (in Ampere hours) and Vcc the operating voltage (in Volts).

The designer can then update the system requirements iteratively according to the difference between the estimated model parameters and the actual ones.

4 Case-study: Energy characterization of ITS design aspects

In this section, we illustrate our method through a case study in the smart mobility IoT domain by presenting an Intelligent Transport System (ITS). This case study provides environmental condition awareness to different parts of a city through the ITS data exchange scheme defined by the ETSI EN 302 637-2 [10] and ETSI EN 302 637-3 [9] standards. The case study (Fig. 4) aims to measuring and characterizing energy design aspects in a real ITS system that was deployed as a prototype within our premises. Since ITS architectures handle sensitive user-data, they require the existence of security mechanisms to prevent attacks from malicious actors. Characteristic ITS attack examples aim to take control of the network, such as DDoS [15], spoofing or frame replay [6]. Hence, to protect the system against such attacks we have implemented a lightweight security library that includes the TLS, DTLS and IPSec protocols.

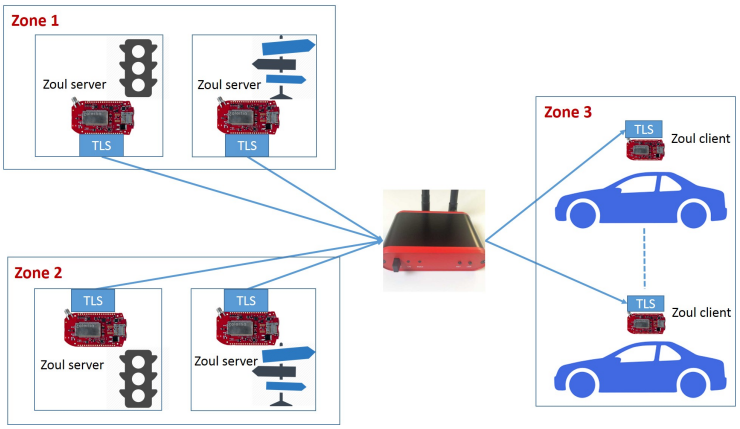

Fig. 4. Topology of the ITS prototype

The deployed architecture of Fig. 4 allows Zolertia Zoul devices that are located in certain zones to send environmental data (i.e. temperature, humidity) to an Orion border router[9], which is configured as data forwarder for supporting awareness of vehicles that are about to enter these areas. Specifically, in Fig. 4 vehicles from Zone 3 are about to enter Zone 1 and 2, hence after establishing a secure connection with the border router they ask for real-time data analytics about the environmental conditions. The border router also runs a web-server, where vehicles have access and can get live updates. Furthermore, the Zoul modules run the Contiki OS and are placed into ITS devices, as traffic lights and road signs that we have configured in our prototype (Fig. 5).

[9] https://zolertia.io/product/orion-router/

Fig. 5. Deployment of the ITS prototype

4.1 Application of the proposed method

Step 1: Transformation for the System Model
We used the described energy-relevant parameter XML configuration along with the DSL description to generate the BIP System Model.

Step 2: Code generation from IoT application templates
In this step, we have considered the parameters of Section 3 combined with the connectivity parameters presented in [11]. The value range of all parameters that influence energy consumption is shown in Table 1. The default value for each parameter is specified by the ITS system requirements.

Energy model parameter	Associated aspect	Default value	Variation range
RDC protocol	Connectivity	X-MAC	[Contiki-MAC, X-MAC, LPP, nullRDC]
RDC frequency	Connectivity	8 Hz	[2-32] Hz (even number)
Packet retransmissions	Connectivity	4	[0-5] $\in \mathbb{Z}$
Service protocol	Connectivity	CoAP	[CoAP, MQTT, HTTP]
Header size	Connectivity	48 bytes	[32-64] bytes (even number)
Interference	Connectivity	0	[0-1] $\in \mathbb{R}$
Resource processing	Data processing	Application resources: Temperature, Humidity	Available resources: Temperature, Humidity, Motion, Light, Accelerometer
Routing protocol	Data processing	RPL	[RPL, CORPL, CARP, none]
Memory block management	Data processing	Mmem	[Malloc, mmem]
Security level	Security	SL-2	[SL-0, SL-1, SL-2, SL-3]
Security protocol	Security	TLS	[TLS, DTLS, IPSec]
Session key size	Security	256	[128, 192, 256] bits

Table 1. Parameters of the energy-aware configuration

Step 3: Energy characterization
We have used the technique of Section 3 to derive probabilistic dsitributions, aiding in the calibration of the Energy Model with energy constrains for the ITS case study. The derived probabilistic distributions were found to follow the Generalized Pareto or the Cauchy distributions. These distributions were used in *Step 4* to calibrate the energy model (presented in Fig. 3).

Step 5: Energy aspect monitoring
We identified two conditions for the ITS system, that are presented in Section 3. The system requirements for the ITS[10] lie on the usage of security level SL-2, meaning strong encryption (256-bit key size) and authentication mechanisms for the communication (Default value in Table 1). Hence, the energy monitors for all aspects are enabled to ensure that conditions A and B in Section 3 hold. As an additional step, the estimations for possible changes in the IoT architecture can be validated through the use of *SMC* in *Step 6*.

4.2 Experiments

In this section, we demonstrate the experiments for evaluating the aforementioned requirements. To automate these experiments we developed a tool that given the parameter configuration XML, executes the system model for all combinations of parameters and saves the energy estimations that satisfy at least one of the conditions in dedicated files. The current and voltage values that were used for the calculation of the total energy, duty cycle and device lifetime for Section 3 equations, were obtained from the IoT devices' datasheet. The experiments were conducted by leaving the Zoul devices on battery power for an entire working day and then charging them to reach their full battery capacity.
Condition A. By experimenting with multiple variations for the parameters of Table 1 we concluded that the largest contribution to the energy consumption is given by the connectivity and security aspects. Specifically, the use of TLS increases substantially the device energy consumption (Actual Energy in Fig. 6) in the processing mode. This invalidates condition A, since the security monitor allows energy consumption up to 60 Joules for sustaining the device for an entire day of continuous operation. Hence, this scenario is excluded from the feasible ones in the energy estimation feedback report that is returned to the user. Instead, the experiments with 128 key size and no authentication scheme allowed condition A to be satisfied (Estimated Energy in Fig. 6), as the duty cycle in processing mode is significantly reduced.
Condition B. As with condition A, B was also not met for the Zoul devices. Additionally, the scenario that led condition A to be met i.e. 128 key size and no authentication scheme did not satisfy condition B, since it led to a duty cycle: $D_{CPU} = 67\%$. However, the combination of this scenario with an increased RDC frequency to 32 Hz resulted in meeting condition B, as $D_{CPU} = 58\%$.

5 Conclusion

We presented a novel method for estimating the energy consumption for various design aspects of IoT applications. The method is based on the principles of rigorous system design by using the BIP component framework. It takes as input the application design description in a DSL and an XML-based set of energy

[10] https://www.etsi.org/e-brochure/Work-Programme/2017-2018/files/basic-html/page17.html

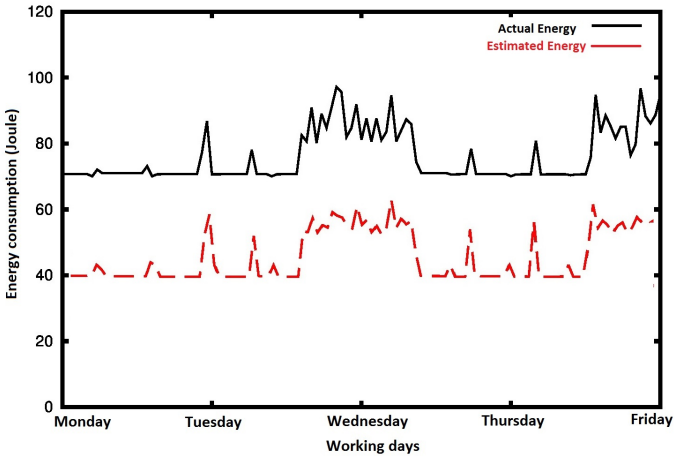

Fig. 6. Actual energy consumption compared to the estimated

parameters, and generates a system model in BIP calibrated with energy constraints. These constraints are obtained by energy characterization mechanisms applied to the execution traces of the deployed IoT application. The calibrated model is then monitored through model conditions that allow to verify if the system requirements are met and also to estimate scenarios where they can be met. The estimations are given as feedback to the IoT system designer.

As a proof of concept, the described method has been applied to an Intelligent Transport System. The system consists of road signs and traffic lights that are informing vehicles for climate conditions upon entering their area. This system requires the presence of strong security mechanisms to respect the privacy of exchanged data and to avoid security threats. We have verified conditions related to the IoT device lifetime and the CPU duty-cycle for security mechanisms. The results allow to provide a feasible design solution for the ITS application by considering the energy cost of each IoT design aspect.

Currently, the energy aspect monitoring technique requires extensive tests for all the combinations of energy parameters in each IoT application. We plan to improve this by testing only the relevant scenarios according to the system requirements. This will allow faster estimations for the IoT application designer.

References

1. Aijaz, A., Aghvami, A.H.: Cognitive machine-to-machine communications for Internet-of-Things: A protocol stack perspective. IEEE Internet of Things Journal **2**(2), 103–112 (2015)
2. Aijaz, A., Su, H., Aghvami, A.H.: CORPL: A Routing Protocol for Cognitive Radio Enabled AMI Networks. IEEE Trans. Smart Grid **6**(1), 477–485 (2015)

3. Antonakakis, M., April, T., Bailey, M., Bernhard, M., Bursztein, E., Cochran, J., Durumeric, Z., Halderman, J.A., Invernizzi, L., Kallitsis, M., et al.: Understanding the mirai botnet. In: USENIX Security Symposium. pp. 1092–1110 (2017)
4. Basagni, S., Petrioli, C., Petroccia, R., Spaccini, D.: CARP: A channel-aware routing protocol for underwater acoustic wireless networks. Ad Hoc Networks **34**, 92–104 (2015)
5. Basu, A., Bensalem, B., Bozga, M., Combaz, J., Jaber, M., Nguyen, T.H., Sifakis, J.: Rigorous component-based system design using the BIP framework. IEEE software **28**(3), 41–48 (2011). https://doi.org/10.1109/MS.2011.27
6. Chim, T.W., Yiu, S., Hui, L.C., Li, V.O.: Security and privacy issues for inter-vehicle communications in VANETs. In: Sensor, Mesh and Ad Hoc Communications and Networks Workshops, 2009. SECON Workshops' 09. 6th Annual IEEE Communications Society Conference on. pp. 1–3. IEEE (2009)
7. Collina, M., Corazza, G.E., Vanelli-Coralli, A.: Introducing the QEST broker: Scaling the IoT by bridging MQTT and REST. In: Personal indoor and mobile radio communications (pimrc), 2012 ieee 23rd international symposium on. pp. 36–41. IEEE (2012)
8. Dunkels, A., Osterlind, F., Tsiftes, N., He, Z.: Software-based on-line energy estimation for sensor nodes. In: Proceedings of the 4th workshop on Embedded networked sensors. pp. 28–32. ACM (2007). https://doi.org/10.1145/1278972.1278979
9. ETSI, E.: 302 637-3 V1. 2.2 (2014-11) Intelligent Transport Systems (ITS). Vehicular Communications
10. ETSI, T.: Intelligent transport systems (its); vehicular communications; basic set of applications; part 2: Specification of cooperative awareness basic service. Draft ETSI TS **20**, 448–451 (2011)
11. Lekidis, A., Katsaros, P.: Model-Based Design of Energy-Efficient Applications for IoT Systems. In: Proceedings of the 1st International Workshop on Methods and Tools for Rigorous System Design, MeTRiD@ETAPS 2018. pp. 24–38 (2018). https://doi.org/10.4204/EPTCS.272.3
12. Lekidis, A., Stachtiari, E., Katsaros, P., Bozga, M., Georgiadis, C.K.: Model-based Design of IoT Systems with the BIP Component Framework. Software Practice and Experience (2018). https://doi.org/10.1002/spe.2568
13. Nastic, S., Sehic, S., Le, D.H., Truong, H.L., Dustdar, S.: Provisioning software-defined IoT cloud systems. In: 2014 2nd International Conference on Future Internet of Things and Cloud (FiCloud). pp. 288–295. IEEE (2014)
14. Nouri, A., Bensalem, S., Bozga, M., Delahaye, B., Jegourel, C., Legay, A.: Statistical model checking QoS properties of systems with SBIP. International Journal on Software Tools for Technology Transfer **17**(2), 171–185 (2015). https://doi.org/10.1007/s10009-014-0313-6
15. Parno, B., Perrig, A.: Challenges in securing vehicular networks. In: Workshop on hot topics in networks (HotNets-IV). pp. 1–6. Maryland, USA (2005)
16. Shelby, Z., Hartke, K., Bormann, C.: The constrained application protocol (CoAP). Tech. rep. (2014)
17. Vilajosana, X., Wang, Q., Chraim, F., Watteyne, T., Chang, T., Pister, K.S.: A realistic energy consumption model for TSCH networks. IEEE Sensors Journal **14**(2), 482–489 (2014). https://doi.org/10.1109/JSEN.2013.2285411
18. Winter, T., Thubert, P., Brandt, A., Hui, J., Kelsey, R., Levis, P., Pister, K., Struik, R., Vasseur, J.P., Alexander, R.: RPL: IPv6 routing protocol for low-power and lossy networks. Tech. rep. (2012)