Research by Experimentation
for Dependability on the
Internet of Things

Publishable Summary Report

Grant Agreement no: 317826
www.relyonit.eu

Date: 2015-03-10

Author(s) and affiliation: Carlo Alberto Boano, Kay Römer (TUG), Nicolas Tsiftes, Thiemo Voigt (SICS), Marco Zúñiga, Koen Langendoen (TUD), James Brown, Utz Roedig (ULANC), Alejandro Veiga, Rafael Socorro (ACCIONA), Xavier Vilajosana, Màrius Montón (WOS)

Work package/task: —
Document status: Final
Dissemination level: Public
Keywords: Dependability, Internet of Things, Wireless Sensor Networks, Radio Interference, Temperature, Protocols.

Abstract This document is the Publishable Summary Report within the Final Report of the FP7 RELYonIT project.

Disclaimer

The information in this document is proprietary to the following RELYonIT consortium members: Graz University of Technology, SICS Swedish ICT, Technische Universiteit Delft, University of Lancaster, Worldsensing, Acciona Infraestructuras S.A.

The information in this document is provided “as is”, and no guarantee or warranty is given that the information is fit for any particular purpose. The user uses the information at his sole risk and liability. The above referenced consortium members shall have no liability for damages of any kind including without limitation direct, special, indirect, or consequential damages that may result from the use of these materials subject to any liability which is mandatory due to applicable law.

Contents

List of Figures

Executive Summary

Internet of Things (IoT) applications in several domains of utmost importance for our society, such as surveillance of critical infrastructures, smart cities, smart grids, and smart healthcare, require dependable performance. Failure to meet application-specific guarantees on network performance parameters such as data delivery reliability and energy consumption may lead to reduced user satisfaction, increased costs, or to critical system failures. Unfortunately, existing IoT and underlying Wireless Sensor and Actor Network (WSAN) technologies mostly follow a best effort approach and do not offer guaranteed performance. The major hurdle to providing a dependable IoT is that the operation of WSANs linking the real world to the Internet is deeply affected by their surrounding environment. Environmental properties such as electromagnetic radiation and ambient temperature have significant impact on achievable network performance. These environmental conditions are not only hard to predict for a given deployment site, but they may also largely vary from one deployment site to another, thus hindering the scalable deployment of IoT applications.

RELYonIT creates a systematic framework that enables the development of dependable IoT applications by taking into account the challenging interaction of IoT platforms and communication protocols with the surrounding environment. To this end, RELYonIT has devised parametrizable environmental models that capture how environmental properties vary over time and platform models that capture how these environmental properties affect the operation of a hardware platform. A specification language allows a user to specify dependability requirements for a given application that drive the automatic selection and parametrization of environment-aware protocols such that performance requirements can be met for a given deployment environment and hardware platform or the infeasibility of these requirements is detected. If environmental properties have significantly changed at runtime, the framework makes sure that protocol parameters are automatically adapted to reflect the new environmental model.

To support this approach, RELYonIT has extended FIRE experimental infrastructures to provide the ability of recording and playing back realistic environmental effects, hence allowing repeatability of specific environmental conditions and simplifying the evaluation of novel, dependable IoT communication protocols. These experimental infrastructures help researchers to better explore the challenging interaction of wireless sensor and actuator networks with their surrounding environment without the need of carrying out labour-intensive and costly real-world experiments.

The project has deployed two pilot applications addressing real-world scenarios provided by our industrial partners ACCIONA and WorldSensing. In both deployments, WSAN nodes were exposed to challenging environmental conditions that strongly affected network performance when using state-of-the-art IoT solutions. When using RELYonIT solutions, not only the minimal packet delivery and lifetime performance requirements for our application scenarios were met, but often better performance could be offered, despite the hostile environment.

1 Project Context and Objectives

1.1 Motivation

The Internet of Things (IoT) provides a substrate to realize applications in several domains of utmost importance for our society, including surveillance of critical infrastructures, smart cities, smart grids, and smart healthcare. However, many of these applications are only possible if the IoT provides dependable performance. Application-specific guarantees on network performance parameters such as data delivery reliability and energy-consumption must be given for all system operation conditions, and failure to meet these requirements may lead to reduced user satisfaction, increased costs, or to critical system failures.

Unfortunately, existing IoT and underlying Wireless Sensor and Actor Network (WSAN) technologies mostly follow a best effort approach and do not offer guaranteed performance. The major hurdle to providing a dependable IoT is that the operation of WSANs linking the real world to the Internet is deeply affected by their surrounding environment. Environmental properties such as electromagnetic radiation, ambient temperature, and humidity have significant impact on achievable network performance. WSAN communications have indeed often to deal with radio interference from other communication networks such as WiFi and Bluetooth systems that may lead to a significant message loss, which in turn leads to an increase in latency and energy consumption. Environmental temperature also deeply affects the operation of WSAN: the electronics of wireless sensor nodes may experience substantial temperature variations over time and space. The latter can have a detrimental effect on network performance, as they can significantly affect clock drift, battery capacity and discharge, as well as the efficiency of low-power radios. These environmental conditions are not only hard to predict for a given deployment site, but they may also largely vary from one deployment site to another, thus requiring a costly customization that hinders the scalable deployment of IoT applications.

1.2 Goals

Within this context, RELYonIT aims to enable the cost-effective construction of IoT applications meeting performance guarantees in hostile environments by providing a set of generic methods and tools that address the challenging interaction of WSANs with their surrounding environment. To support the construction of these methods and tools, RELYonIT aims to augment existing FIRE experimental facilities with the ability to record and playback the effects of environmental conditions in a realistic and repeatable fashion. This allows not only an easier experimentation and evaluation of IoT communication performance, but also to perform visionary experimental research towards a dependable Internet of Things. By creating these solutions and by allowing IoT applications to meet their specific performance requirements, a large class of solutions that could not be realized before will become technically and economically feasible and the time and costs needed for developing dependable IoT applications will be substantially reduced.

1.3 Contributions

Towards these goals, RELYonIT has advanced the state of the art in IoT research by providing the following key contributions:

- Creation of a systematic framework that enables the development of dependable IoT applications by taking into account the challenging interaction of IoT platforms and communication protocols with the surrounding environment. The creation of such framework includes:
  - The design of generic and computationally lightweight environmental and platform models that capture, respectively, specific environmental properties and how the latter affect certain functionalities of a hardware platform;
  - The development of techniques to automatically learn model parameters such that the generic environmental and platform models mentioned above can be used in different deployment scenarios;
  - The design and development of environment-aware protocols that can be automatically configured to meet application-specific dependability requirements;
  - The development of tools to automatically select and parametrize communication protocols such that the specified dependability requirements can be met;
  - The creation of tools to track model conformity at runtime and to allow a runtime adaptation of models and protocol parameters;
  - The design of a specification language that allows to incorporate dependability requirements into the application specification.
- Extension of FIRE experimental infrastructures such as TWIST [21] to provide the ability of recording and playing back realistic environmental effects, namely temperature variations and radio interference, hence allowing repeatability of specific environmental conditions and simplifying the evaluation of novel, dependable IoT communication protocols.
- Experimental evaluation of the framework on two pilot deployments located in Barcelona and Madrid, Spain, addressing real-world scenarios provided by our industrial partners ACCIONA and WorldSensing.

Generic framework. RELYonIT provides a generic framework that enables dependable IoT applications by taking into account all relevant environmental properties and their impact on IoT platforms and communication protocols. The components of the framework are highlighted in Figure 1.1.

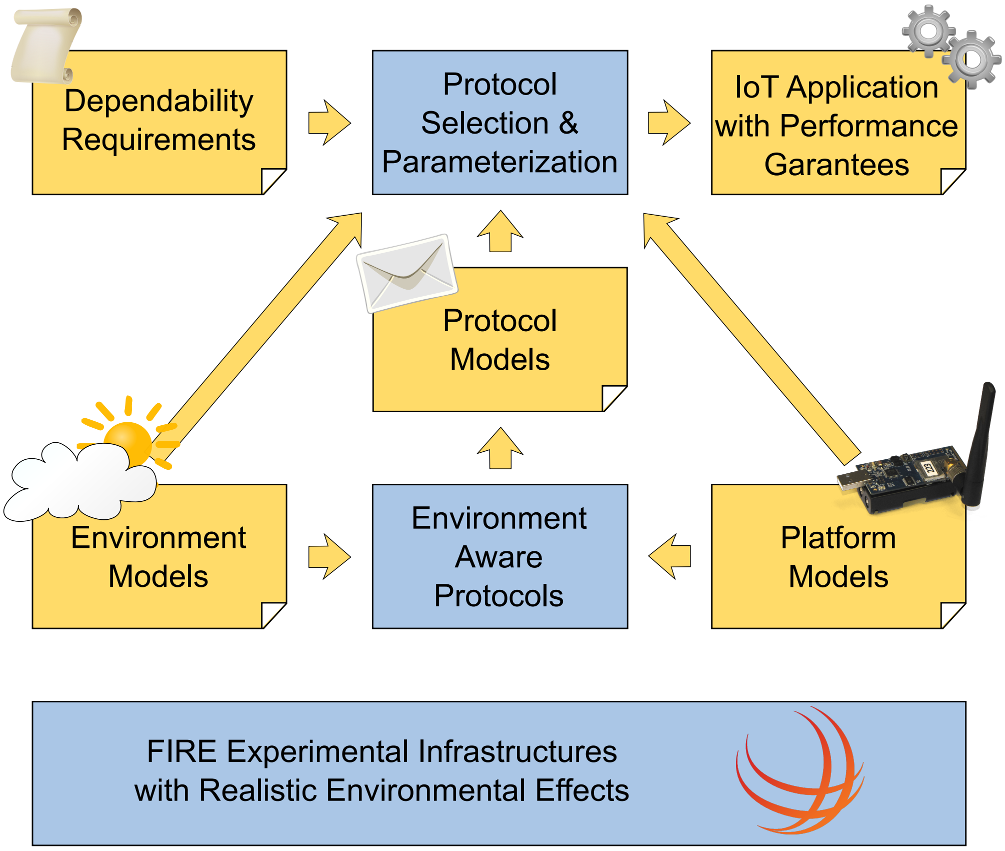

Dependability Requirements
Protocol Selection & Parameterization
IoT Application with Performance Garantees
Protocol Models
Environment Models
Environment Aware Protocols
Platform Models
FIRE Experimental Infrastructures with Realistic Environmental Effects

Figure 1.1: RELYonIT contributions to enable dependable IoT applications.

Within RELYonIT, we devised generic environmental models that capture how environmental properties vary over time as well as platform models that capture how these environmental properties affect the operation of a given hardware platform. These models have parameters that can be automatically learnt for a given platform and for a given deployment environment, which makes sure that the generic environmental and platform models can be used in any deployment scenario. We have further optimized existing protocol and designed new ones to provide performance guarantees for a given deployment environment and hardware platform by exploiting knowledge from the respective environmental and platform models. Finally, a specification language allows a user to specify dependability requirements for a given application that drive the automatic selection and parametrization of environment-aware protocols such that the performance requirements can be met for a given deployment environment and hardware platform or the infeasibility of these requirements is detected. If environmental properties have significantly changed at runtime, RELYonIT makes sure that protocol parameters can be automatically adapted to reflect the new environmental model.

Augmented experimental infrastructures. To provide such a dependability framework for the IoT, RELYonIT has extended existing FIRE experimental infrastructures with the ability to record and playback realistic environmental effects, namely temperature variations and radio interference. Such experimental infrastructures augmented with realistic environmental effects represent a significant contribution to the community, as they simplify experimentation and help researchers to better explore the challenging interaction of wireless sensor and actuator networks with their surrounding environment without the need of carrying out labour-intensive and costly real-world experiments. The augmented FIRE experimental infrastructures further allow repeatability of specific environmental conditions and hence enable the evaluation of novel, dependable IoT communication protocols, as well as a quantitative comparison of the performance of different approaches.

Pilot deployments. These contributions have been evaluated experimentally on two pilot deployments located in Barcelona and Madrid, Spain, addressing real-world scenarios provided by our industrial partners ACCIONA and WorldSensing. The deployment in Barcelona at Worldsensing's premises is based on a smart parking solution in which sensors embedded into the pavement detect the occupancy of parking spots and allow users to find the nearest free parking spot using a smart phone application. The deployment in Madrid at ACCIONA's premises is based on a wireless sensor network installed on the outdoor façade of buildings to measure the effectiveness of the insulating materials used for their construction – an important step in order to build more energy-efficient buildings.

In both cases, the wireless sensor nodes are exposed to extremely variable environmental conditions: the streets in Barcelona are characterized by bursty and heavy radio interference, whereas at the DEMOPARK facility in Madrid the nodes attached to the outdoor façade of buildings suffer high on-board temperature variations. Nevertheless, our results indicate that the RELYonIT system does not only meet the minimal packet delivery and lifetime performance requirements for our application scenarios, but can often even offer better performance, hence practically increasing the dependability of the deployed IoT applications despite the hostile and highly-dynamic environment.

2 S&T Results and Foregrounds

The RELYonIT project provides a number of technical contributions across different aspects, from the design of environmental and platform models and the development of environment-aware protocols that can be automatically configured to meet application-specific dependability requirements, down to the extension of FIRE experimental infrastructures providing the ability of recording and playing back realistic environmental effects.

We briefly describe next the major scientific and technical results of the RELYonIT project. Section 2.1 illustrates the design and implementation of IoT testbed extensions that enable the repeatable playback of environmental factors. Section 2.2 describes the design of generic and computationally-light environmental and platform models that capture, respectively, specific environmental properties and how the latter affect certain functionalities of a hardware platform. We further present techniques to automatically parametrize these models for specific deployment scenarios, as well as tools that track model conformity at runtime and alert the user as soon as environmental properties have significantly changed. Section 2.3 illustrates the design and development of environment-aware protocols that provide performance guarantees for a given deployment environment and hardware platform by exploiting knowledge from the respective environmental and platform models. We further present the design of protocol models that describe the performance of the newly designed environment-aware protocols as a function of the environment and the selected hardware platform. Section 2.4 describes the design of a specification language that allows to incorporate dependability requirements into the application specification and the development of tools to automatically select and parametrize communication protocols such that the specified dependability requirements can be met. We further illustrate how to allow a runtime adaptation of models and protocol parameters.

All the scientific and technical contributions developed within RELYonIT have been evaluated experimentally on two pilot deployments in collaboration with the industrial partners. The setup and execution of these experiments as well as their results are summarized in Section 2.5.

2.1 Testbeds with Realistic Environmental Effects

To understand how the environment affects the performance and the operation of IoT hardware, protocols, and applications, it is fundamental to be able to rerun experiments under identical environmental conditions, and the first contribution of RELYonIT is the design and implementation of two IoT testbed extensions that enable the repeatable playback of two environmental factors:

1. **TempLab** is a low-cost extension for sensornet testbeds that allows to study the impact of temperature on wireless sensor hardware and protocols [10];
2. **JamLab** is a low-cost extension for sensornet testbeds that allows the creation of realistic and repeatable interference patterns [5].

2.1.1 TempLab: a Testbed to Study the Impact of Temperature on Wireless Sensor Networks

To better study the impact of temperature variations on low-power wireless communications and protocols, we have designed TempLab [10], a novel testbed infrastructure with the ability of varying the on-board temperature of sensor nodes and reproducing the temperature fluctuations that can be normally found in outdoor deployments. TempLab can accurately reproduce temperature traces recorded in outdoor environments with an average error of only 0.1°C. Such infrastructure allows a quick debugging of protocol behaviour and plays an important role in examining and quantifying the effects of temperature variations on sensornet applications and protocols, as it can reveal system limitations that would not have been visible when experimenting with existing testbed installations. Experiments using TempLab have indeed revealed that high temperature variations can drastically change the topology of a network and lead to network partitions, reduce significantly the performance of MAC protocols, as well as increase the processing delay in the network.

Such a testbed solution essentially has one main function: the ability to control the on-board temperature of wireless sensor nodes. However, in order to accurately reproduce the temperature dynamics that can be found in typical deployments, the infrastructure needs to be able to reproduce specific temperature profiles on several nodes, and this requires: (i) temperature profiles to be reproduced, (ii) actuators to control the on-board temperature of each sensor node, and (iii) a controller that uses the actuators to instantiate the desired profiles.

In the remainder of Section 2.1.1, we describe testbed components, methods for implementing different temperature profiles, and show that TempLab can accurately reproduce temperature dynamics found in outdoor environments with fine granularity.

Temperature profiles. In order to support a wide range of experimentation techniques, TempLab can generate temperature profiles using three different approaches. Firstly, one can replay temperature traces collected in-situ at a given deployment site, such as those in Figure 2.1. However, traces are not always at one's disposal, or they may be incomplete: trace-based profiles can be used only if one or more sensor nodes deployed previously actually collected temperature data with a frequency sufficiently high to capture the dynamics of temperature changes. A second possibility is, therefore, to use a *model-based* temperature profile to have an estimation about the temperature dynamics at a certain location without the need of traces collected in-situ. Such model-based approach uses models to estimate the temperature profile of objects using basic environmental information such as the maximum solar radiation and the minimum temperature during a day, which is readily available from satellites and meteorological stations. A third possibility is to use TempLab to vary the temperature of sensor nodes using specific *test patterns*. Users that are not interested in recreating a specific profile and only need to verify whether a high temperature variation has an impact on the operation of a given protocol can feed TempLab with on-off patterns (e.g., a series of cold and warm periods) or jig-saw patterns that vary temperature with a specified frequency, allowing a quick debugging of protocols' behaviour.

Time-lapsing of traces. TempLab also offers the possibility of compressing the time-scale of an experiment to save evaluation time: one may indeed want to time-lapse the recreation of real-world traces and playback, for instance, in a few hours the profile of a full day.

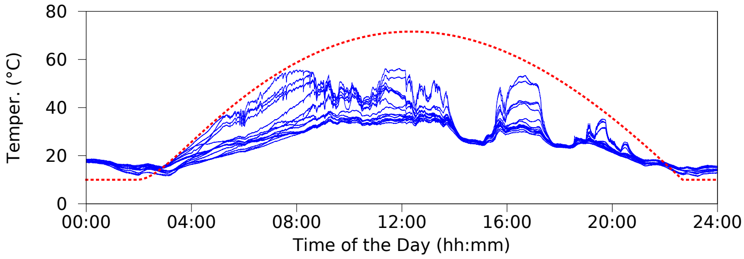


Figure 2.1: Daily temperature profiles of 16 nodes deployed outdoors (blue curve), and maximum temperature profile obtained with the model-based instantiation (red dashed curve) [10].

Actuators. To heat-up and cool-down the on-board temperature of sensor nodes, one or more actuators are required for each node. We design TempLab following an *out-of-band* approach (i.e., the sensor node is not involved in the control of its temperature) based on infra-red heating lamps and cooling enclosures that allow to vary the on-board temperature of wireless sensor nodes in the range [-5, +80] °C. TempLab can have two types of nodes with different capabilities as shown in Figure 2.2: *LO* and *PE* nodes. *LO nodes*, which stands for lamps-only nodes, are heating-only devices that have the capability of warming the sensor nodes between room temperature and their maximum operating range. They are based on IR heating lamps and they do not have any capability to cool-down the nodes below room temperature. *PE nodes*, which stands for Peltier enclosure nodes, are hard temperature-isolating Polystyrene enclosures with an embedded IR heating lamp and an air-to-air Peltier module to heat-up and cool-down the inner temperature of the casing. To control the intensity of the IR lamps and the operations of the Peltier module, we borrow existing home automation solutions and use *wireless dimmers* to vary the intensity of the lamps and *on-off wireless switches* to control the Peltier modules embedded in the enclosure. To make sure that the temperature control system does not interfere with the existing testbed communication, we select home automation solutions working on an ISM frequency band that is different from the one used by the sensor nodes. This approach can easily scale to large testbeds as PE and LO nodes only need to be plugged into wall power and require no further cabling. Furthermore, home automation solutions such as Z-Wave allow to connect up to 256 wireless dimmers in a multi-hop fashion, and can can in principle scale to large buildings with many devices.

Controller. To instantiate a temperature profile and control heat lamps and Peltier modules, TempLab uses different controllers running on a centralized testbed gateway computer. The simplest one is an open-loop controller that varies the intensity of the light bulbs in LO and PE nodes according to a pre-computed calibration function (this is possible if the impact of each dimming level on the on-board temperature of a node is known based on previous calibration). To precisely regenerate trace- or model-based temperature profiles, TempLab uses a closed-loop proportional-integral controller that tries to minimize the difference between the desired temperature profile and the on-board temperature of the sensor node of interest. To periodically convey temperature readings to the controller, TempLab uses the USB back-channel by means of a low-priority routine executing only when the processor is idle.

Accuracy in replaying real-world traces. We have collected and replayed a number of traces from real-world deployments and FIRE testbeds, showing that TempLab can accurately

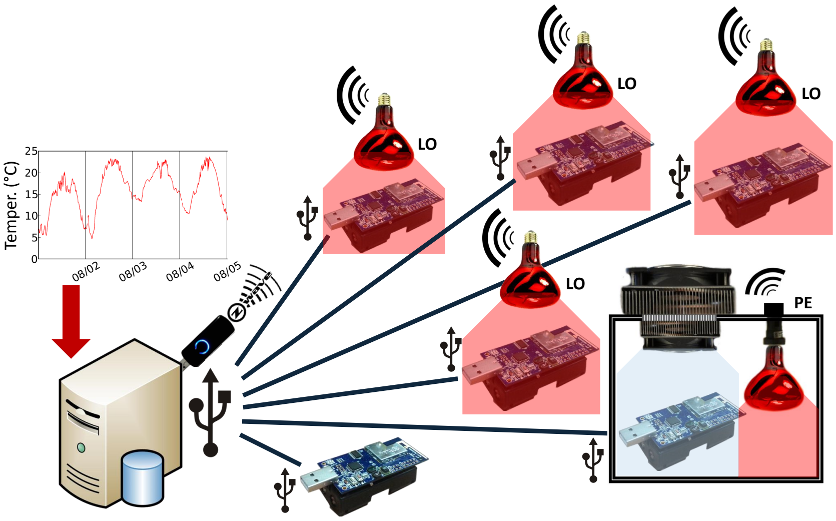


Figure 2.2: Sketch of TempLab's architecture [4].

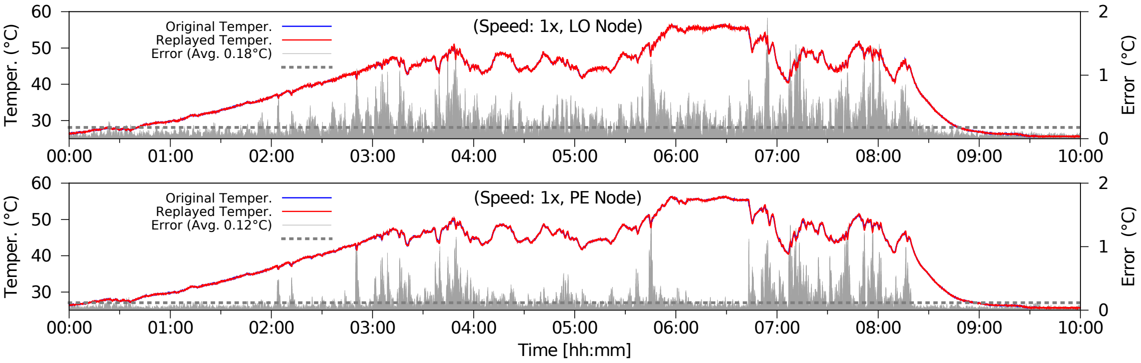


Figure 2.3: Accuracy of TempLab in replaying real-world traces [47].

reproduce a large variety of temperature patterns (full results available in [10, 47]). Figure 2.3 shows the ability of Templab to accurately reproduce the temperature traces recorded in an outdoor deployment in Uppsala, Sweden [49]. The average error in the temperature replay is only 0.18 and 0.12°C when using LO and PE nodes, respectively. When time-lapsing the same trace with a factor of 3, 5, or 10 using LO nodes, the average error still remains below 0.5, 1.1, and 1.9°C, respectively, showing very good performance. We have also replayed several temperature traces taken from the FIRE testbed facility in Santander, Spain, in different seasons, with an average error in the replay between 0.35 and 0.74°C (see [47]).

2.1.2 JamLab: a Testbed to Study the Impact of Radio Interference on Wireless Sensor Networks

JamLab is a low-cost extension for sensornet testbeds that allows the creation of realistic and repeatable interference patterns [5]. The key idea behind JamLab is to use off-the-shelf motes to record and playback interference patterns instead of bringing Wi-Fi access points, microwave

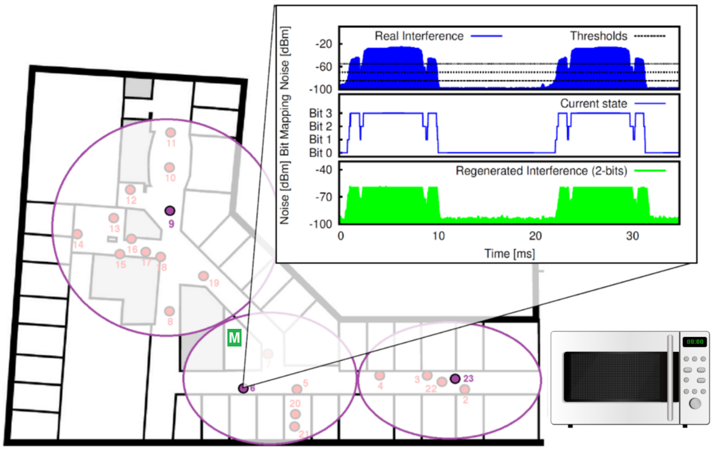


Figure 2.4: Overview of JamLab's architecture [4].

ovens, or other expensive equipment to the testbed: with JamLab, either a fraction of the existing nodes in a testbed are used to record and playback interference patterns, or a few additional motes are placed in the testbed area (see Figure 2.4). The sensor nodes used for interference generation are called *HandyMotes* or *jammers* and support two modes of operation:

- *Emulation*, where lightweight models are used to generate interference patterns that resemble those generated by a specific appliance (e.g., Wi-Fi device and microwave ovens);
- *Regeneration*, where each jammer autonomously samples the actual interference, compresses and stores it locally, and regenerates the recorded patterns at a later stage. This mode is especially useful to record realistic interference patterns in a crowded shopping center or on a lively street by placing a few jammers to record interference, and bringing them to the testbed to playback the recorded traces there.

For a correct replay of interference, the nodes that operate as jammers need to be carefully selected, such that each remaining node is covered by at least one jammer, i.e., the signal from the latter needs to be stronger than any other signal in order to ensure an effective impact on radio communications. Furthermore, to have a sufficient set of nodes available for the actual experimentation, the number of testbed nodes selected as jammer should be minimized. Such testbed configuration is typically carried out manually, a time-consuming task that did not guarantee the creation of optimal jammer configurations. Within RELYonIT, we have extended JamLab and created an automatic testbed configuration that allows an optimal selection of the jammers within the network. In particular, we developed a tool-chain to semi-automatically derive a suitable testbed configuration as illustrated in Figure 2.5. Such a configuration is generated in a three step process:

1. The signal strength of all potential links at all output power levels is recorded;
2. This signal strength data is employed to determine an optimized selection of jammers and suitable output power level for the jammers;
3. The jamming software is deployed on the previously selected nodes, whilst the remaining nodes can be used for experimentation.

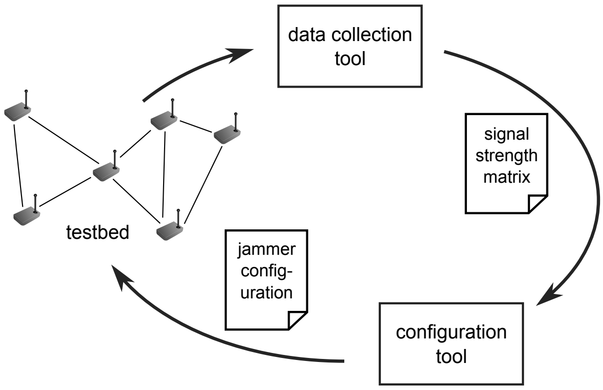


Figure 2.5: Tool-chain to support the jammer configuration process [47].

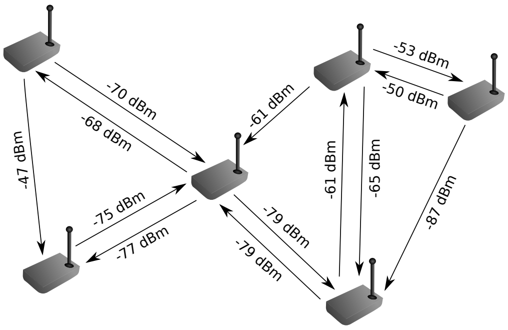


Figure 2.6: Signal strengths of an exemplary testbed [47].

The tool first builds a connectivity matrix with signal strength readings for each possible link between a pair of nodes (see Figure 2.6). The tool essentially consists of a program that is deployed on the nodes of the testbed and a central data collection application. During data collection, the nodes sequentially send a broadcast message, including the employed power level and node ID, at each available output power level. All other nodes monitor the radio and record the signal strength and the transmitted information for each message they receive. The collected data is integrated into a single signal strength matrix by the data collection application. The node component of this tool was implemented for Contiki and TinyOS to support a wide range of different environments.

The generated signal strength matrix is in turn used by a separate Python program running on a more powerful machine to derive a suitable configuration, by employing an optimization strategy. This tool generates a selection of jammers and determines a suitable output power setting for each of these jammers. This configuration is written to a file and can be subsequently employed to deploy the actual jamming software on the selected nodes. The latter step is currently not fully automated due to the heterogeneous programming interfaces of the different FIRE facilities. This jammer configuration tool-chain has been employed in the prototype experiments as well as in the final deployments described in Section 2.5 to support the testbed experiments. Most notably, it was used to assist with the generation of suitable jammer selections for the local testbed at the Graz University of Technology and in the FIRE facility TKN Wireless Indoor Sensor network Testbed (TWIST) at TU Berlin.

2.2 Environmental and Platform Models

Within RELYonIT we have first designed *environmental models* that capture the behaviour of important environmental properties such as external interference and temperature. We have then devised *platform models* that capture how the operation of a specific platform is affected by those environmental properties, such that by combining environmental and platform models we can characterize the behaviour of a given hardware platform as a function of the environment.

We kept our models generic, and we have investigated methods to automatically learn the parameters of these models for a given target environment and for a specific platform before the actual deployment of a system, so that the RELYonIT solutions can be applied in principle to any setting. Furthermore, we have investigated techniques to verify at runtime that the model parameters learned prior deployment are valid throughout the lifetime of the system.

2.2.1 Environmental Models

In RELYonIT we focus on two environmental phenomena: *temperature* and *radio interference.*

Temperature. We describe the behaviour of an environment $\mathcal{E}$ in terms of four thermal properties: hotness, periodicity, change of rate, and max/min temperature range [51].

Hotness. Given the probability mass function of temperature $p_i(t)$, the hotness of a node H_i can be defined by the expectation:

$$H_i = \sum_t t * p_i(t) \tag{2.1}$$

Periodicity. Identifying the periodicity of temperature patterns helps to exploit the times of the day in which temperature is low (at lower temperatures the performance of the network is better). Denoting f_i and g_i as the time series of the temperature observed by node i at two different days, we use cross correlation functions to quantify the periodicity of a node P_i:

$$(f_i \star g_i)[n] = \sum_{m=-\infty}^{\infty} f_i[m] g_i[n+m] \qquad P_i = \max_n (f_i \star g_i)[n] \tag{2.2}$$

Change of rate. To design protocols that can adapt to the environment we quantify how fast the environment changes, because the network properties will change at the same rate. To capture the maximum rate of change on a node i, we identify the steepest slope of the temperature series f_i:

$$R_i = \max_t (f_i(t + \Delta t) - f_i(t)) \tag{2.3}$$

Max/min temperature range. The maximum and minimum temperatures recorded on each node i bound the performance of the network. Given a trace f_i recorded at a node i, these properties are obtained as follows:

$$(\max_i^t, \min_i^t) = (\max\{f_i\}, \min\{f_i\}) \tag{2.4}$$

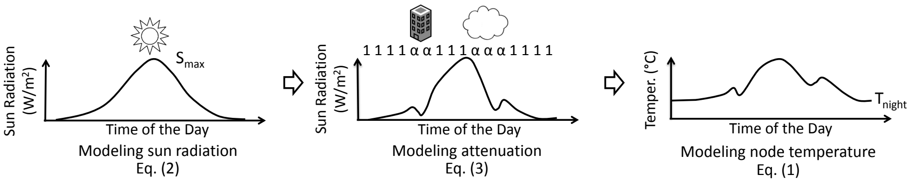


Figure 2.7: Model-based temperature profile generation [10].

Identifying the model parameters. Using thermodynamic equations, we derived a model suitable to create temperature profiles for nodes. We focus on outdoor deployments where infra-red radiation from the sun and air temperature are the most significant factors. In essence, objects heat up by absorbing solar radiation and cool down by constantly releasing energy to their surrounding. The balance between these processes determines the object temperature.

The model has three basic steps, as shown in Figure 2.7. First, we model the impact of sun radiation: this is done by assuming a clear sky, where the object absorbs the maximum possible infra-red radiation hitting its surface (top diagram in Figure 2.7). Second, we model the impact of events blocking sun radiation, such as clouds and buildings. These events decrease the temperature of the object (middle diagram in Figure 2.7). Finally, we combine these models to recreate the temperature profile of a node, based on a simple set of model parameters e (rightmost diagram in Figure 2.7). Further details can be found in [51].

Radio interference. When other devices operating in the same frequency band of WSANs are active, bursts of interference signals (busy periods) alternate with instants in which the channel is clear (idle periods). We have modelled radio interference using the Cumulative Distribution Function (CDF) of idle and busy periods (full details can be found in [51]). This approach has two key advantages: (i) it is suitable for low-power wireless networks, as knowledge on distributions of idle and busy periods can be exploited to design interference-aware protocols, and (ii) it is usable with resource-constrained wireless sensor nodes, as these distributions can be easily derived by performing RSSI sampling on off-the-shelf motes.

The distribution of idle and busy periods is strongly dependent on the type of devices generating interference. Some devices (e.g., microwave ovens) generate periodic interference patterns with relatively long idle periods, while others (e.g., Wi-Fi stations) generate interference patterns with short idle periods of a highly variable length. When several interfering sources are present, interference occurs continuously and independently at a constant average rate. In these cases, we model the CDF as an exponential distribution. Denoting $P_i(j)$ as the CDF of the idle periods formed by the interference pattern, and denoting $P_b(j)$ as the CDF of the busy periods formed by the interference pattern, we use the following models to capture idle and busy CDF:

$$P_i(i) = \begin{cases} 1 - e^{-\lambda \cdot j} & j \geq 0 \\ 0 & j < 0 \end{cases} \qquad P_b(i) = \begin{cases} 1 - e^{-\lambda \cdot j} & j \geq 0 \\ 0 & j < 0 \end{cases}$$

2.2.2 Platform Models

We have further devised platform models capturing how the operation of a specific platform is affected by temperature and radio interference. In particular, we have characterized the impact of temperature on low-power radio transceivers and the effects of radio interference on packet loss and on the operation of clear channel assessment [51].

Effect of temperature on radio transceivers. Our empirical measurements have shown that an increasing temperature has three main effects on the signal strength of radio transmissions; it (i) decreases the transmitted power, (ii) decreases the received power, and (iii) decreases the noise floor. We have modelled these three effects in a first-order model that we have validated using our TempLab experimental testbed infrastructure [7].

Denoting α, β, γ as constants with units dB/K, and T_t, T_r as the temperature in Kelvin of transmitter and receiver, the effect of temperature on SNR can be defined as:

$$\begin{aligned} SNR(dB) &= P_t - PL - P_n \\ &= (P_t - P_n) - (P_t - P_r) \\ &= (P_t - \alpha\Delta T_t) - (PL + \beta\Delta T_r) - (P_n - \gamma\Delta T_r + 10\log_{10}(1+\tfrac{\Delta T_r}{T_r})) \\ &= P_t - PL - P_n - \alpha\Delta T_t - (\beta-\gamma)\Delta T_r - 10\log_{10}(1+\tfrac{\Delta T_r}{T_r}) \end{aligned} \tag{2.5}$$

The proportional relation between ΔT and the constants α (effect on transmitted power), β (effect on received power) and γ (effect on noise floor) is based on the empirical observations obtained using our TempLab testbed [7]. The term $10\log_{10}(1+\frac{\Delta T_r}{T_r})$ is derived analytically from the well-known thermal equation. There are two important trends to highlight in this model. First, changes in temperature have a higher impact on the transmitted and received powers (linear relation of α and β), than on the thermal noise (logarithmic relation). Second, to some extent it is counter-intuitive that a higher temperature decreases the noise floor (negative sign of γ). That is, a higher temperature not only reduces the gain of the signal but also the gain of the noise, and hence, the received signal strength (RSSI) is lower for both.

Effect of radio interference on packet loss. The primary outcome of radio interference is typically an increase in the packet loss rate: in the presence of a sufficiently strong interference signal, the receiver node is no longer able to discriminate the good signal from the interfering one. The receiver node can indeed reject any interference that is C_{Rej} weaker than the signal of interest, with C_{Rej} being the so called co-channel rejection capability of the transceiver (with unit dB). Any interfering signal stronger than that may result, depending on its duration and strength, in either a corrupted or a completely lost (i.e., not even detected) packet. The first case occurs when radio interference corrupts only some of the bits in a frame, leading to cyclic redundancy check errors and a consequent dropped packet. In the vast majority of the cases, however, the radio does not even detect the presence of a frame. Given a pair (A, B) of wireless sensor nodes in which A transmits a train of n packets $P_1...P_n$ to B, a generic interfering signal will affect the reception of a packet P_i at node B as follows:

$$P_i = \begin{cases} received & if\;\; (I_i - R_i) \le C_{Rej} \\ not\ received & if\;\; (I_i - R_i) > C_{Rej} \end{cases} = \begin{cases} received & if\;\; (R_i + C_{Rej}) \ge I_i \\ not\ received & if\;\; (R_i + C_{Rej}) < I_i \end{cases}$$

where R_i is the RSSI of packet P_i at node B, I_i is the signal strength of the interfering signal at node B during the reception of P_i, and C_{Rej} is the co-channel rejection of B's transceiver.

Effect of radio interference on clear channel assessment. Interference may also affect the overall energy efficiency and delay in the wireless network other than through loss of a packet. MAC protocols employing carrier sense multiple access with collision avoidance (CSMA/CA) typically sense the channel for ongoing transmissions using clear channel assessment

(CCA) and transmit packets only if the channel is found to be idle. If a node detects the presence of an ongoing activity in the radio channel, the node defers or cancels the transmission, which may lead either to an unbounded latency or to the loss of a packet.

The clear channel assessment operation is typically based on the measured received signal strength compared against a programmable threshold T_{CCA} (in dBm units). Therefore, the success or failure of the transmission of a packet P_i in the presence of a CSMA-CA protocol for a node can be simply modelled as:

$$P_i = \begin{cases} transmitted & if \;\; I \leq T_{CCA} \\ not\ transmitted & if \;\; I > T_{CCA} \end{cases} \tag{2.6}$$

where I is the signal strength of the interfering signal and T_{CCA} is the programmed CCA threshold. Further details can be found in [51].

2.2.3 Learning Model Parameters

The parameters of environmental and platform models may differ from one environment to another or between different platforms, and obtaining the correct parameter values to characterize a certain deployment environment or platform is a non-trivial task. For environments, human domain experts with sufficient experience and deep knowledge of the environment may be able to estimate those parameters, but this requires substantial effort and needs to be verified. For platforms, values in datasheets may provide some insights but detail is often insufficient and there is little consideration of environmental conditions.

Hence, within RELYonIT, we have devised algorithms and tools to learn these parameters in order to support domain experts. The idea is to install a cost-efficient measurement system in the target deployment environment prior to the actual deployment of the WSAN. This measurement system consist of a small number of WSAN nodes that are installed for a short amount of time to sample the environmental and platform parameters of interest. The data collected can then be used to compute the model parameters from these observations: in some cases, particularly for platform parameters, the tools can be run in the lab under controlled conditions to reduce the time needed for data collection.

Within RELYonIT, we have developed tools to learn the characteristics of temperature and interference in a given environment, as well as to understand their impact on the RSSI measurements and the system timing of common WSAN platforms. Besides data collection, we also focused on model bounding and aggregation: the utility of a model is often low without model bounds (for instance, knowing the average clock drift of one particular platform provides little use without worst and best case bounds to configure guard times). Similarly, considering models produced by each mote individually adds an unnecessary complexity and combining models captured by different nodes into a single model instance can drastically ease their use. Further details on how we built the tools to learn model parameters can be found in [11].

2.2.4 Runtime Assurance Component

The tasks of the RELYonIT runtime assurance component are threefold:

1. Monitoring aspects of systems performance for indications of possible volitions of the environmental model (*violation detection*);

2. Verifying that model violations exist (*verification*);
3. Reporting violations of the environmental model and use the collected data for remediation (*reporting and remediation*).

Violation detection. The detection module runs alongside the application and regularly checks for signs that the environmental model has been violated. As embedded systems are resource-constrained, it is important to ensure that the method used for violation detection is lightweight in order to limit any potentially adverse effect on the application: especially the run-time of the component should be restricted to minimise the possibility of affecting the application's own scheduled operations. The detection module can be scheduled to run periodically based on a timer. However, when doing so, the effects on the application would be difficult to predict. Another approach is to infer that a violation has occurred by measuring other properties of the system. A number of aspects of the system may indeed degrade if the environmental model is violated: for example the energy consumption of the device may increase with radio interference due to an increased wake-up rate. Communication metrics may also change with the environment such as a drop in signal strength with temperature or a drop in packet reception rate (PRR) with increased interference: changes in either of which could signal to the runtime assurance component that the environmental model may have been violated. The advantage of this approach is that it typically requires minimal processing and memory overhead, as many of these metrics are already recorded by the system at runtime.

Verification. This module is executed by the application after the detection module detects a potential environmental model violation. Its task is to verify that a violation has actually occurred and to notify the reporting component to signal the violation. Verification of a violation often involves re-running the environmental data capture tool for the specific environmental aspect and the data recorded is then used to instantiate a real-time instance of the model that is compared to the one captured prior to application deployment. In the event that the deviation between the two models is above a model-defined threshold then a violation is considered to have occurred. The processing and memory overheads of model violation verification depend on the model being verified, although it is typically higher than that of the detection subcomponent. As the resource requirements for verification can vary from minor to extreme, scheduling when the verification algorithm should run is under full control of the application.

Reporting and remediation. This module of the runtime assurance framework is responsible for signalling an alarm and initiating a remediation operation as soon as a violation of the environmental model has occurred. The module is directly interfaced with the verification system and as such has no user callable functions. The implementation to signal the event to a controller is left up to the user, but may, for example take the form of a header bit flag piggy-backed in existing application packets sent to the sink. Alternatively, an entirely separate message may be sent to signal the event. When beneficial, the module should also transmit information gathered by the verification system that can be used for remediation, such as the current temperature or the generated interference model instance. Forwarding this information helps in remediating the current performance issues, and provides hints to the runtime adaptation and protocol parametrization modules (see Section 2.4).

We have created instances of this framework for both temperature and interference [15].

2.3 Environment-Aware Protocols

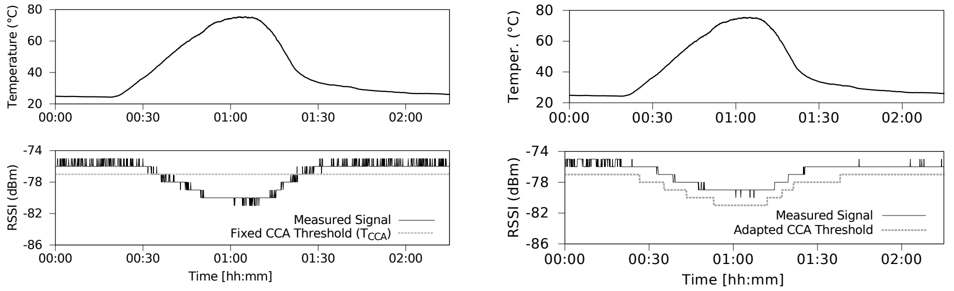
Within RELYonIT we have designed and implemented four new protocols that overcome interference and temperature effects, as well as a novel method that adjusts the packet size according to the observed interference patterns. For the evaluation of these protocols, testbed infrastructures played a key role: it would indeed not have been possible to obtain meaningful and practical results without the TempLab testbed [10], nor without the testbeds available from the FIRE initiative (in particular the TWIST testbed from TU-Berlin [21]) that we augmented using JamLab [5], as described in Section 2.1.2.

Our protocol-design process has followed a two-step approach [52]. First, where possible, we tried to optimize the performance of existing protocols. Second, if we found that the attempted optimization did not lead to significant improvements, we designed the protocol from scratch.

2.3.1 Optimized Protocols

Within RELYonIT, we have designed three optimizations at the data link layer: one that tackles temperature effects (TempMAC), and two that aim at overcoming the effects of radio interference (MiCMAC and Variable Packet Size).

Temperature-aware MAC (TempMAC). We have analysed in detail the adverse effects of temperature on communication protocols and experimentally shown that fluctuations of the on-board temperature of sensor nodes reduce the efficiency of data link layer protocols, leading to a substantial decrease in packet reception rate and to a considerable increase in energy consumption. The reasons for such performance degradation lies in the reduced effectiveness of clear channel assessment at high temperatures, which compromises the ability of a node to avoid collisions and to successfully wake-up from low-power mode. Exploiting the environmental and platforms model presented in Sections 2.2.1 and 2.2.2, we have proposed TempMAC [9], a protocol that dynamically adapts the clear channel assessment threshold to temperature changes, thus making data link layer protocols temperature-aware. An extensive experimental evaluation has shown that TempMAC considerably increases the performance of a network in the presence of temperature variations commonly found in real-world outdoor deployments, with up to 71% lower energy consumption and 194% higher packet reception rate.



(a) ContikiMAC: static CCA threshold (b) TempMAC: dynamic CCA threshold

Figure 2.8: Attenuation of signal strength at high temperatures: ContikiMAC uses a static CCA threshold and cannot avoid that the RSSI curve intersects the CCA threshold (a). TempMAC, instead, dynamically adapts the the CCA threshold to the current temperature (b) [9].

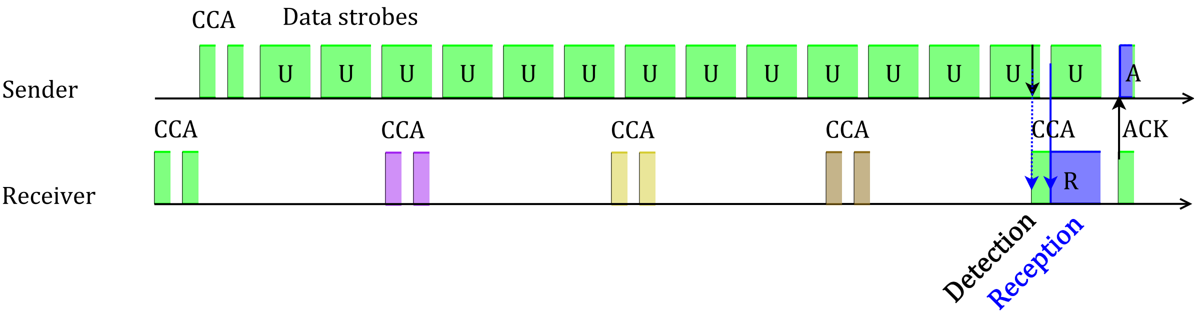


Figure 2.9: MiCMAC initial rendezvous (4 Channels): the sender strobes over one available channel until it receives an acknowledgement, for a maximum of 4 consecutive wakeup periods. The receiver wakes up periodically to sample the channel with two short CCA and hops through all available channels according to its own sequence [1].

MiCMAC. We have proposed a practical extension of low-power listening, MiCMAC [1], that performs channel hopping, operates in a distributed way, and is independent of upper layers of the protocol stack. The above properties make it easy to deploy in a variety of scenarios, without any extra configuration, scheduling, or channel selection hassle. We have implemented our solution in Contiki and evaluated it in the TWIST FIRE testbed while running a complete, out-of-the-box low-power IPv6 communication stack (UDP/RPL/6LoWPAN). Our experimental results demonstrate an increased resilience to emulated Wi-Fi interference (e.g., data yield kept above 90% when the ContikiMAC drops in the 40% range). In noiseless environments, MiCMAC keeps the overhead low in comparison to ContikiMAC, achieving performance as high as 99% data yield along with sub-percent duty cycle and sub-second latency for a 1-minute inter-packet interval data collection.

Variable packet size. We have exploited the interference models developed within RELYonIT to provide a better understanding of the achievable packet reception rate for a given environment. The underlying assumption is that by estimating the *idle* time between two consecutive *busy* periods (where interference is present), we can adjust the size of the packet to fit in-between the estimated *idle* time. We used this knowledge to select and configure protocols to mitigate much of the effect that interference causes to deliver the required network performance. For instance, an alternative packet size may be selected based on the interference level to attain higher delivery targets, or the transmission channel can be changed (based on the existing interference patterns) to always select the least affected channel [12].

2.3.2 Newly Designed Protocols

Within RELYonIT, we have designed two new protocols: one tackling interference at the data link layer (JAG) and one tackling interference and temperature effects at the data link and network layers (Evergreen).

Jamming-based AGreement (JAG). In the context of RELYonIT, we have designed, implemented, and evaluated JAG, a simple yet efficient agreement protocol for wireless sensor networks exposed to external interference. JAG introduces a jamming sequence as the last step of a packet handshake between two nodes to inform about the correct reception of a message

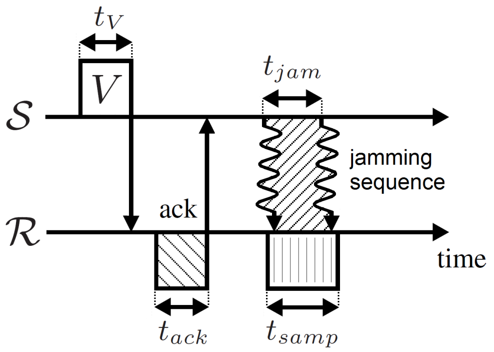


Figure 2.10: Illustration of JAG: the last acknowledgement of the 3-way handshake between nodes $\mathcal{S}$ and $\mathcal{R}$ is sent in the form of a jamming signal [6].

carrying the information to be agreed upon, as shown in Figure 2.10. The key insight behind this approach is that detecting a jamming sequence in the presence of external interference is more reliable than using acknowledgement packets to verify whether the information was successfully shared. In environments that experience high levels of external interference, the probability of successfully transmitting a sequence of packets and completing an handshake is small, even when using short ACK packets. Despite the minimal amount of information they carry, acknowledgements are embedded into IEEE 802.15.4 frames, and hence can be destroyed if any of the bits in the header, payload, or footer is corrupted by interference. Performance can be improved by means of redundancy (i.e., by sending multiple ACK packets), but this results in a significantly higher energy expenditure and latency, which is undesirable when using resource-constrained wireless sensor nodes. Using JAG, instead, one can minimize the energy expenditure and provide agreement guarantees under weaker and more realistic assumptions about the underlying interference pattern compared to message-based approaches. By appropriately tuning the length of the jamming sequence, one can parametrize JAG to obtain predictable performance and to guarantee agreement in a finite amount of time, even in the presence of external interference: a perfect fit for applications with timeliness requirements. We have evaluated JAG using JamLab [5] and shown that it outperforms message-based approaches in terms of agreement probability, energy consumption, and time-to-completion, as well as that it can also be used to obtain performance guarantees and meet the requirements of applications with real-time constraints.

Evergreen. We have developed a *comprehensive* new collection protocol named Evergreen that can overcome link dynamics independently of their source, which can be temperature, interference or even mobility. Evergreen is indeed designed to work—and not to break down—in extreme conditions including: high interference, large temperature variations, node mobility, and high data rates. In addition, Evergreen also aims to perform well in stable, more favourable environments achieving at least the same performance as existing protocols designed for those conditions. Full details about Evergreen's design can be found in [47]. In summary, Evergreen can tolerate changes in the network and surrounding environment thanks to the following characteristics:

1. *Link Quality Estimation (LQE)*: Evergreen's link quality estimation is designed to quickly recognize a lossy link and can be maintained in the absence of data packets.

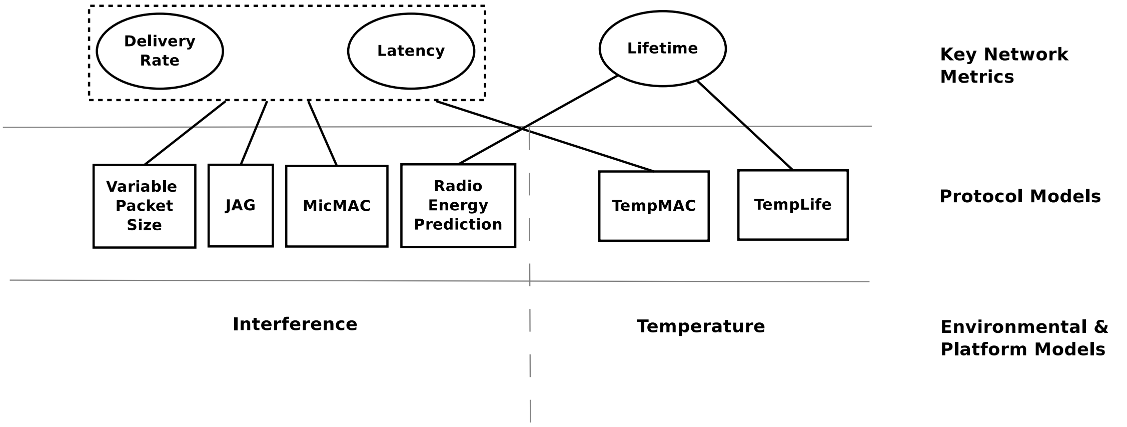


Figure 2.11: Overview of protocol models developed within RELYonIT.

2. *Alternative Forwarding Algorithm*: When the main algorithm of Evergreen cannot forward packets due to network changes, an alternative algorithm kicks in. The main algorithm is useful to find near optimal paths in stable network conditions whereas an alternative algorithm is used when the network is in a transitional phase and no route to a root node is known.

3. To maintain network state a node has to decide how frequently control messages are to be sent. Evergreen employs CTP's Trickle timer, but uses an additional set of comprehensive strategies for asserting when to increase the frequency of control messages.

4. Evergreen uses three different control messages depending on the state of the network, leading to a reduction in overhead and allows running at lower duty cycles, leading to increased network lifetime.

5. Evergreen further employs several other optimizations including packet aggregation, packet queue management, and short-circuit data forwarding.

2.3.3 Protocol Models and Validation

When several unknowns affect the performance of a system, it becomes increasingly complex to evaluate all the possible scenarios in testbeds. Arguably, the best way to estimate and adjust the performance of such a system is to model it mathematically. Given a particular environment, mathematical models can not only predict the performance of the system, but they can also be used to calibrate parameters before and at runtime (to adjust to continuously changing environments).

In RELYonIT, to allow a robust operation of future wireless networks under interference and temperature effects, we developed mathematical models for some of the protocols presented in Section 2.3.2. A summary of the designed protocols and their models are depicted in Figure 2.7. These protocol models allow us (i) to provide quality-of-service guarantees and (ii) to perform optimisations at runtime. All models have been extensively validated in testbeds, except for MicMAC which was evaluated through simulations. Full details about the devised protocol models and their validation can be found in [52].

2.4 Configuration and Runtime Support

A central component of the RELYonIT approach are software tools to automatically select, parametrize, and adapt IoT protocols. We have designed and implemented software artefacts that allow the user to select protocols and to generate parameter configurations based on models of the scenario and an abstract specification of dependability requirements.

2.4.1 Protocol Selection

A first step to build a dependable Internet of Things application is the selection of suitable protocols. In RELYonIT we support this task with a decision support system that, based on a careful analysis of different protocols, provides hints about which protocols are most suited for a specific environment and scenario.

The systematic approach we propose for selecting a protocol requires four basic steps. The first three steps consist of identifying the properties of the environment (mobility, temperature, interference), identifying the metrics of interest (delivery rate, throughput, energy consumption), and evaluating different network parameters (offered load, scalability, density). With this information, the final step is to compute a ranking to identify the best protocol [32].

Step 1: Identifying the properties of the environment. This step is required because environmental conditions are usually outside the user's control and they can have a dramatic impact in the underlying communication structure of the network. We identified three important environmental properties affecting the dynamics of the network: (1) *Temperature*, which changes the connectivity among nodes due its negative effect on link quality. (2) *Interference* – like temperature – affects the link quality, but it does so at a much faster rate (milliseconds as compared to several tens of minutes in the case of temperature). (3) *Mobility*, which leads to the most dramatic changes in terms of the underlying communication structure. Once these properties have been identified and quantified, the user should use a testbed or scenario that closely resembles these environmental conditions to test the protocols.

Step 2: Identify the metrics of interest. In principle, we would like a network to provide as much data as possible with as high throughput as possible and lasting for as long possible. There are three de-facto metrics used to capture this behaviour: (1) *Delivery ratio*, which is the fraction of packets arriving at the sink(s) compared to the total number of packets sent by the sources; (2) *Throughput*, which is the number of data packets received at the sink (root node) in a second; (3) *Energy consumption*, which is usually captured by the *duty cycle* of the radio. Often these three metrics present trade-offs: a higher delivery rate or high throughput, for example, usually requires a higher energy consumption.

Step 3: Evaluate protocols. The properties of the scenario (Step 1) are a hard input, and the desired performance (Step 2) is a hard output, hence we cannot change them. There is however a flexible input that allows the user to have a deeper insight into the performance of the system: the properties of the network. Once the scenario of interest has been assessed (Step 1) and the metrics quantified (Step 2), the user can start testing protocols. We identified three general properties that affect the performance of protocols but that are under the control of the user: (1) *Scalability*, which is related to the number of nodes in the network; (2) *Density*, which is defined as the average number of neighbours observed by all nodes; (3) *Offered load*, which is the total number of packets generated by the sources per unit of time.

Step 4: Select protocol. We propose a ranking method to select the right protocol: users assign weights to the metrics identified in Step 2 (depending on their relative performance), and then use the information collected in Step 3 to identify the protocol with the best performance according to the environmental parameters identified in Step 1.

2.4.2 Protocol Parametrization

The protocol parametrization framework is a central component of the RELYonIT software architecture, and its goal is to generate protocol configurations that meet previously specified requirements. The parametrization component relies on mathematical optimization to determine a near optimal protocol configuration for a specific application based on a user-generated requirement specification and instances of environment, platform, and protocols models.

Architecture. Figure 2.12 presents a bird's eye view of the RELYonIT parametrization architecture. The central component of the architecture is the static configuration framework for protocol parametrization: this component selects a suitable parameter set to ensure the dependable performance of IoT protocols. The tool receives an XML-encoded specification of the dependability requirements as input, which also contains the selection of protocols determined as explained in Section 2.4.1. A single requirement specification may contain several requirement sets for different modes of operation. In addition to the requirements specification, the configuration tool has access to a repository of protocol models. Based on the protocols defined in the requirement specification, a suitable protocol model is chosen. The individual protocol models may in turn rely on application-specific instances of environment and platform models. Based on these inputs a near-optimal parameter configuration for the respective protocol is generated by the static configuration tool, employing optimization techniques. The final output of the protocol parametrization tool is a protocol configuration for each employed protocol. These configurations are static and do not change at run-time. Nevertheless, it is possible to switch between configurations associated to different performance modes. The configurations are emitted in the form of a C source file that can be compiled as an individual module and can be linked with the run-time component and the actual application program.

Specifying dependability requirements. The static protocol parametrization component employs mathematical optimization to generate an optimal parameter configuration for one or more IoT protocols based on a user-defined dependability specification specified using XML [29]. Each dependability specification essentially defines a constrained optimization problem. Within the context of RELYonIT we only consider the single objective case where a single metric is optimized, while the user may specify a number of additional constraints on the same or other metrics. The system currently supports three metrics: (1) system lifetime, (2) data yield, (3) and latency. This results in optimization problems of the following form:

$$\begin{array}{ll} \text{Maximize/Minimize} & m_0(c) \\ \text{Subject to} & m_1(c) \geq, \leq t_1 \text{with probability } p_1 \\ & m_2(c) \geq, \leq t_2 \text{with probability } p_2 \end{array} \tag{2.7}$$

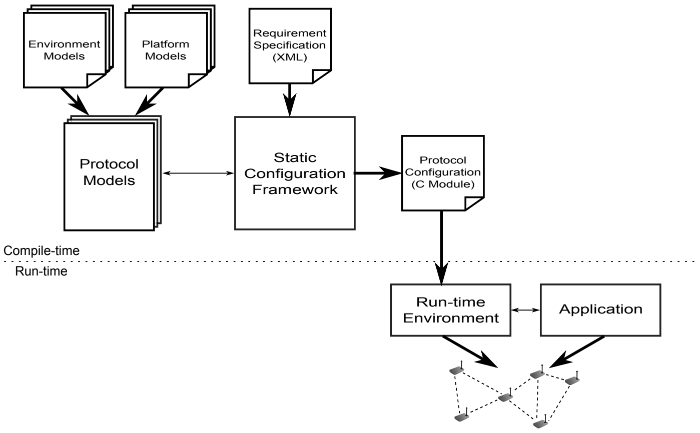


Figure 2.12: Architecture of the RELYonIT parametrization framework [32].

The goal is to find a set of protocol configuration parameters c that optimizes a single metric m_0. In addition, a variable number of constraints need to be fulfilled by ensuring that $m_1(c)$ and $m_2(c)$ are either larger or smaller than the respective thresholds t_1 and t_2 with a probability of at least p_1 or p_2. Note that m_1 and m_2 may be the same as m_0.

Optimization problem. To be suitable for automatic optimization, the optimization problem needs to be transformed into suitable input for the employed optimization strategies. Most optimization techniques cannot directly handle constraints: instead, constraints need to be integrated in the objective function. We achieve this by normalizing the optimization goal and all constraints to the $[0, 1]$ range and by computing the weighted sum where constraints are given a higher weight than the original objective. In our case, each constraint has twice the weight as the original optimization goal. This approach corresponds to penalty functions often used in stochastic optimization [44]. It ensures that invalid solutions are unlikely to be selected, but still allows the optimization algorithm to traverse infeasible regions of the search space.

In addition, the requirement specification allows to attach probabilities to individual constraints that only need to be fulfilled with the specified probability. This is not supported by standard optimization techniques, and to support this feature, we exploit the fact that most environmental parameters exhibit a periodic behaviour: this enables us to sample the environment at different points of time identifying time-frames with distinct environmental conditions.

Without support for different probabilities, each model would provide a single cost function $f_m(c)$ for each supported metric m. The function $f_m(c)$ returns the normalized cost of configuration c. To enable the use of different probabilities, each model needs to provide a set of functions, where each of these functions $f_{m,i}(c)$ has an associated probability p_i. Each of these functions uses a different instance of the environment model that represents a distinct time frame. During the optimization instead of a single function $f_m(c)$, each function $f_{m,i}(c)$ is evaluated for the current configuration. In a second step, the associated probabilities p_i of all functions that have as cost value that is above the threshold given in the requirement specification are summed up. For each constraint j, these and this probability is used instead of the original function value in the optimization process.

Assuming a weight of 2 for the constraints and a problem of the form of Equation 2.7, we end up with the following definition of the total fitness e for a given configuration c:

$$e(c) = \frac{\left(2 * \sum_{j=1}^{|M|-1} \rho\left(\sum_{i=1}^{|P|} \tau_{\text{op}_j}(f_{m_j,i}(c), p_i, t_j), q_j\right) + f_{m_0}(c)\right)}{3} \tag{2.8}$$

The set M contains the metrics employed by the current specification. As a convention, we assume that the metric of the objective function is named m_0 while the constraint metrics use the indices 1 to $|M| - 1$. The set P contains the probability values provided by the employed model instance. The function τ realizes the above mentioned comparison with the threshold value t. For the operators $<$ and $>$ it is defined as follows:

$$\tau_>(v, p, t) = \begin{cases} p & v > t \\ 0 & \text{otherwise} \end{cases} \qquad \tau_<(v, p, t) = \begin{cases} p & v < t \\ 0 & \text{otherwise} \end{cases} \tag{2.9}$$

The function ρ is used to determine the error between the desired probability of a constraint and the current probability of a constraint being fulfilled. If the difference would be negative, it is set to 0. Consequently, the function is defined as follows:

$$\rho(p, t) = \begin{cases} p - t & p < t \\ 0 & \text{otherwise} \end{cases} \tag{2.10}$$

The protocol parametrization component can utilize a number of different optimization strategies to solve the optimization problem, allowing the user to choose a strategy that is most appropriate for the specification and models at hand (e.g., exhaustive search, simulated annealing, and evolution strategies).

2.4.3 Runtime Adaptation

Dependable operation should be ensured even under unexpected deterioration of the environment. As the environmental models depend on data collected prior to the deployment, it is possible that the environment changes in unpredicted ways, which invalidates the model assumptions. In this case the previously generated configuration may be invalidated and is unable to ensure the expected performance. Such situations are reported to the user by the run-time assurance module described in Section 2.2.4, but user reaction may require some time.

To prevent a severe deterioration of the performance and dependability of the application, we have developed a run-time adaptation framework that tries to adapt the parameter values to the new environment in a best-effort fashion. The runtime adaptation mode is activated directly after the runtime assurance module raises an alarm and operates according to a specific configuration policy for the network deployment. While it is active, the runtime adaptation module gathers data about the current network performance and adjusts protocol parameters according to the configuration policy. The latter is not designed to meet user-specified performance requirements: instead, the configuration policy, in conjunction with runtime adaptation, is meant as a fall-back mechanism that provides acceptable performance under a wide variety of environmental conditions outside the range of the environmental model.

Configuration policies are generated off-line for each specific deployment. For this purpose, we implement a reinforcement learning algorithm in the Cooja simulator [33], which is able to emulate a network of nodes running exactly the same system firmware as is running on the real nodes in the deployment. The reinforcement learning algorithm explores a set of protocol parameters and learns which settings provide acceptable performance under various environmental conditions. More details on the runtime adaptation module can be found in [32].

2.5 Real-World Deployments

All the scientific and technical contributions developed within RELYonIT have been evaluated experimentally on testbed infrastructures and real-world deployments. After building an integrated prototype of all RELYonIT contributions, we have first stress-tested our prototype by generating repeatable interference patterns in the FIRE facility TWIST, and by replaying real-world temperature variations in the TempLab testbed. We have then designed experiments in the context of two use cases provided by our industrial partners:

- In Madrid, Spain, we used ACCIONA's DEMOPARK facility to evaluate the RELYonIT system in a realistic civil infrastructure monitoring scenario;
- In Barcelona, Spain, we used Worldsensing's Smart City facility to evaluate the RELYonIT system in an outdoor parking management scenario.

The two use cases were chosen such that they are exposed to orthogonal environmental impact. The parking use case has radio interference as the dominating environmental influence due to its location in an urban area rich of Wi-Fi interference but where sensor nodes are mostly shielded from direct sun radiation. In contrast, the civil infrastructure demo site is located in a rural area with little radio interference, but with sensor nodes being directly exposed to sunlight and suffering substantial on-board temperature variations.

The two use cases also focused on two orthogonal dependability requirements. Due to the difficulty of replacing batteries in thousands of sensors embedded into tarmac, energy consumption is the most critical requirement for the smart parking use case. As the civil infrastructure monitoring use case may be applied in safety-critical contexts, packet delivery rate is the most critical requirement for the civil infrastructure monitoring use case. We were hence able to *isolate* both the requirement of interest and the dominating environmental impact in each use case from other requirements and secondary environmental impacts, allowing us to clearly attribute performance observations to an environmental property.

2.5.1 Civil Infrastructure Monitoring Use Case

In order to carefully test how the insulating materials used in the construction of buildings reduce heat transfer, ACCIONA Infraestructures installed a testing facility to compare the effectiveness of different building materials and Heating, Ventilating, and Air Conditioning (HVAC) systems in a real-world setting. The facility, called DEMOPARK, has a total area of 1200 m^2 and is located in Fuente del Fresno, 20 km North of the city of Madrid, Spain. The facility contains several testing rooms to test assorted materials such as coatings reflecting the infra-red part of the solar spectrum, phase change materials integrated in lightweight insulation panels, as well as lightweight vacuum panels with ultra-low thermal conductivity.

Performance requirements. In order to allow precise studies on the insulation of a given material, it is fundamental that data loss is minimized, so that engineers have all the required information to draw conclusions about the effectiveness of material or HVAC system under study. Some of the tests to be carried out are heavily dependent on very small changes in the measured variables, so any gap in the collected data may lead to a false conclusion. For this reason, ACCIONA Infraestructures requires a minimum packet delivery rate of 85% (preferably

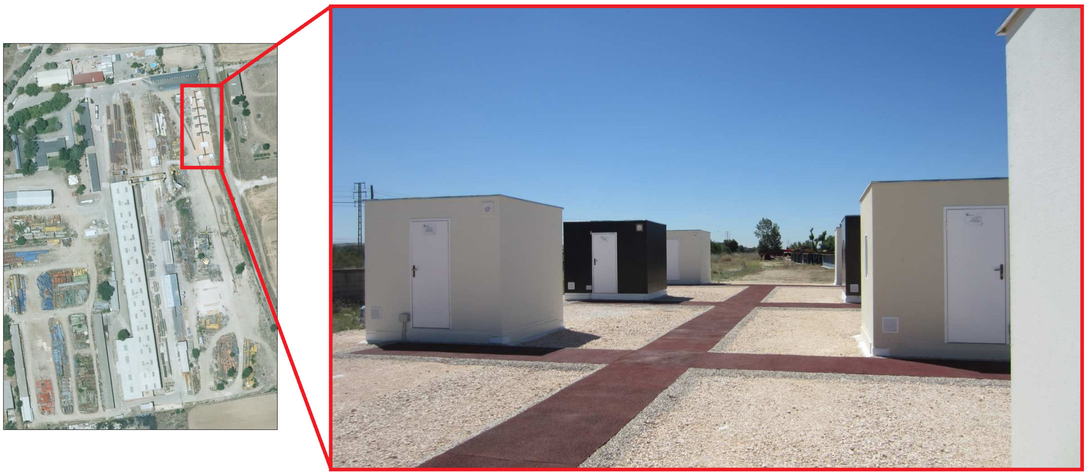


Figure 2.13: Overview of the ACCIONA facility with the buildings used to test insulating materials [46]. Aerial images taken from sigpac.mapa.es.

(a) Dawn (b) Midday (c) Sunset (d) Late evening

Figure 2.14: Images from the Webcam at different times of the day [46].

95%). The climate in Madrid's area is typically Mediterranean, and the air temperature outdoors can vary by up to 50 °C across one year. This facility is therefore a perfect fit to test the solutions produced by the RELYonIT consortium with respect to the impact of temperature variations on IoT performance.

Installation. In Figure 2.13 one can recognize 12 squared buildings of approximately 2.5 meters width and 2.8 meters height: we have installed 7 Maxfor CM5000 and Moteiv TelosB wireless sensor nodes (Tmote Sky replicas) on the different outdoor façades of the northern building in the facility. All nodes are connected via USB to allow a real-time gathering of the sensor data and communication statistics as well as to give us the possibility to have a bird-eye view on the data collection and verify whether the performance requirements have been met at any point in time. To monitor the activities in the building and the sun exposure 24/7, we also installed a Webcam pointed on the south-west façade of the building (see Figure 2.14).

Insufficient performance using state-of-the-art communication protocols. We have used this setup to carry out an extensive data collection sampling the properties of insulating materials. Our results show that, when employing state-of-the-art communication protocols (i.e., protocols that do not include RELYonIT solutions), communication performance varies

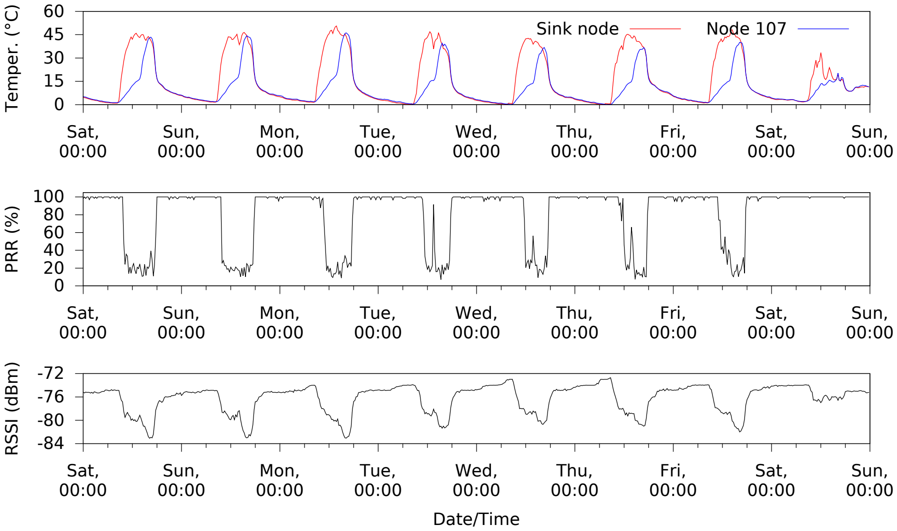


Figure 2.15: Performance of default's ContikiMAC in our deployment. Without resorting to the techniques developed within RELYonIT, temperature variations affect network performance significantly and one cannot achieve the desired performance [46].

dramatically between day and night, with several nodes being unable to meet their performance goals (a packet reception of at least 85%, better 95%). In particular, we have recorded daily on-board temperature fluctuations as high as 55 °C for some of the sensor nodes in our deployment, which caused an attenuation of the received signal strength at the sink in the order of several dB. This attenuation compromised the operations of CSMA-based data link layer protocols with fixed CCA threshold [7, 9, 10] and led to an average delivery rate in the network of only 61.38% of the packets. Figure 2.15 shows the performance of one of the links in the network during a data collection carried out between the end of December 2014 and the beginning of January 2015: this link has suffered a large decrease in performance and is completely compromised during daytime, as a consequence of a reduction of almost 10 dB in the received signal strength (the same behaviour has been recorded throughout the rest of the network).

Meeting performance requirements using RELYonIT solutions. Using the static protocol parametrization tool, we derived an optimal configuration of TempMAC, an optimized version of ContikiMAC [17] designed within RELYonIT to mitigate the impact of temperature variations on network performance. A long-term data collection carried out in the beginning of January 2015 has shown that, despite the same high temperature fluctuations observed in the previous experiment, the reception of packets is unaffected by environmental changes. The average packet reception rate in the network is higher than 85%, and hence within the specified performance requirements. Figure 2.16 shows that temperature variations do not affect the packet reception of the link: TempMAC, indeed, adapts the CCA threshold at runtime, avoiding the wake-up problem at high-temperatures [9].

The RELYonIT system was hence able to correctly predict and mitigate the impact of temperature variations on communication performance, allowing us to meet not only the minimal performance requirements for our application scenario, but even offer better performance [46].

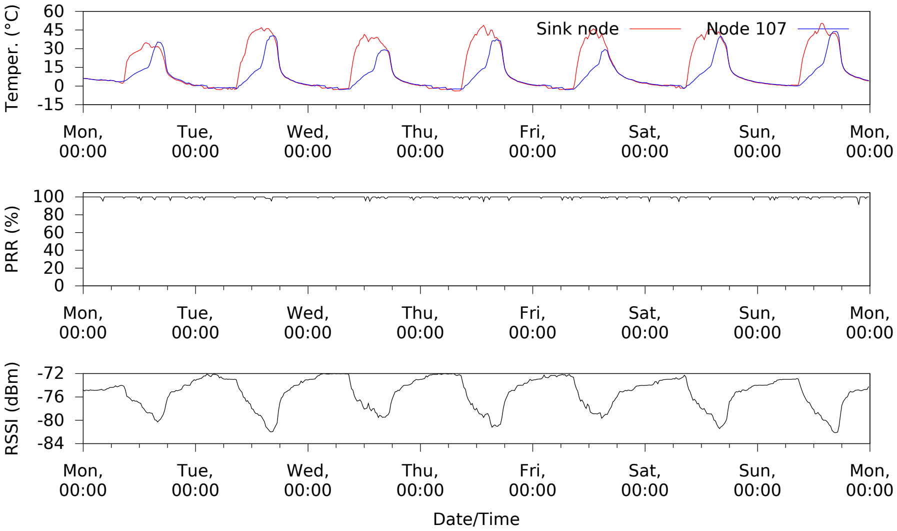


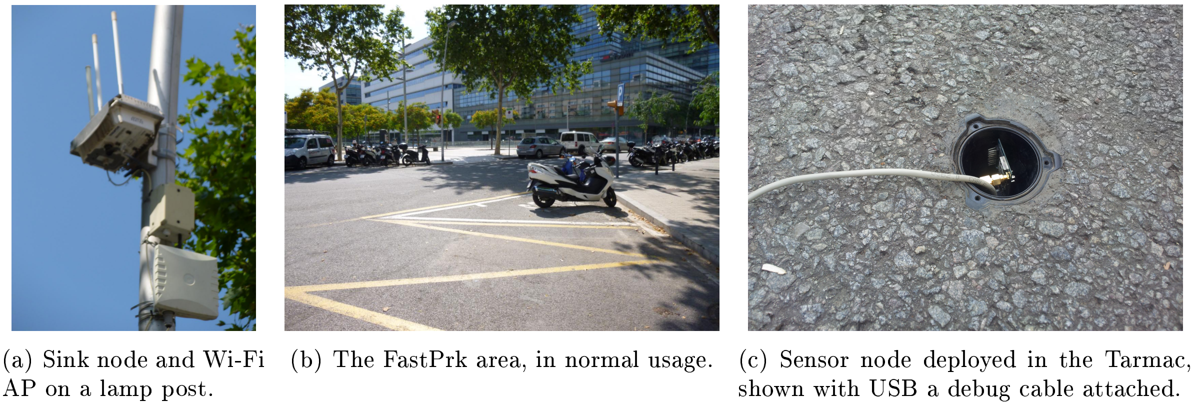
Figure 2.16: Performance of TempMAC in our deployment at the DEMOPARK facility in Madrid. The solution provided by RELYonIT minimizes the impact of temperature variations on packet reception, allowing to achieve the desired performance [46].

2.5.2 Outdoor Parking Management Use Case

Worldsensing's smart parking system, Fastprk, is a smart city system helping drivers to find a parking spot quicker, allowing cities to manage their parking spaces more efficiently. It is composed of a set of sensors installed in every parking spot and a set of gateways that collect and transmit the information to the Internet so that drivers can be informed about the number of available parking places in real-time.

Performance requirements. All sensor nodes in Fastprk are battery-powered, and replacing the battery of all nodes installed in a city is an expensive, labour-intensive operation. To avoid excessive maintenance costs, the company hence requires that all nodes achieve a minimum lifetime of 4 months. To optimize their profit, Worldsensing would like to achieve an average battery lifetime of 18 months in the nodes deployed in the tarmac. This is difficult to achieve given the high number of wireless appliances deployed within the city, especially Wi-Fi devices. The latter may cause an increased wake-up rate of the sensor nodes at different times of the day, leading to an early battery depletion.

Installation. We have carried out a pilot deployment in the Smart City testbed at the 22@Barcelona innovation district consisting of five spots in a load/unload zone in the corner of a larger parking area. These spots are used for short stops for loading or unloading vehicles, and as such see a large amount of use with vehicles arriving and leaving far more often than would normally be seen in a traditional parking space. Battery-powered Tmote Sky wireless sensor nodes running the Contiki operating system are situated in the tarmac at points surrounding a central base station (shown in Figure 2.17) such that the distance between any node and the sink is the same for all nodes and that each node reports changes in sensor readings back to the central sink. All nodes except for the sink are embedded in the road surface, and therefore do not receive much interference from other radio sources. The sink, however, is mounted 4


(a) Sink node and Wi-Fi AP on a lamp post.

(b) The FastPrk area, in normal usage.

(c) Sensor node deployed in the Tarmac, shown with USB a debug cable attached.

Figure 2.17: The experimental area and the sink node in situ [46].

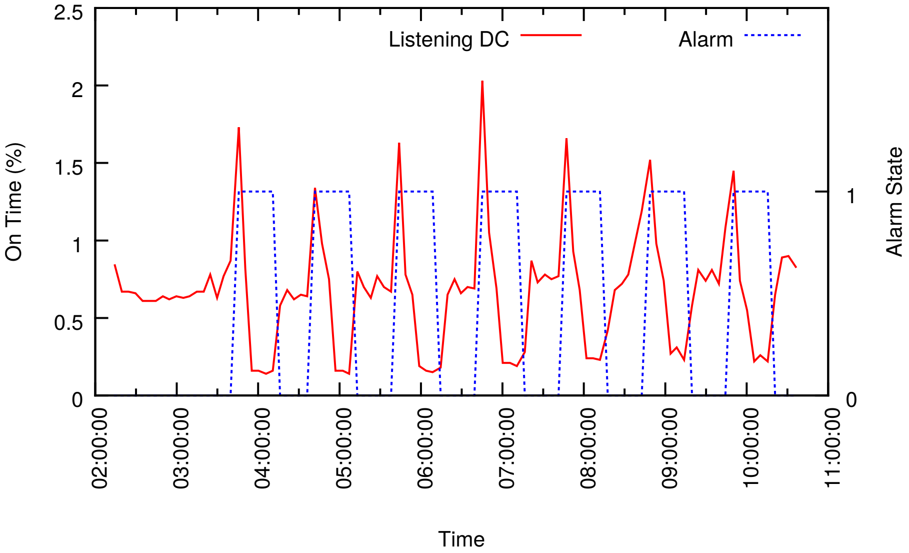


Figure 2.18: Sink-only deployment at Barcelona with a channel check rate of 8. The sink is affected by periodic artificial interference [46].

meters in the air attached to a lamp post, less than a meter away from a commercial Wi-Fi access point mounted on the same lamp post (see Figure 2.17(c)). Hence, the sink receives significant interference and we want to verify the effectiveness of RELYonIT solutions with respect to interference mitigation on energy consumption.

Impact of radio interference on battery lifetime. Low-power listening is a common technique adopted by data link layer protocols to reduce energy consumption and prolong battery lifetime in which nodes periodically wake-up to sample the wireless channel and detect radio activity. However, this technique is highly susceptible to *false wake-ups* caused by environmental noise being detected as activity on the channel, which causes nodes to spuriously wake-up for receiving transmissions that do not actually exist [41].

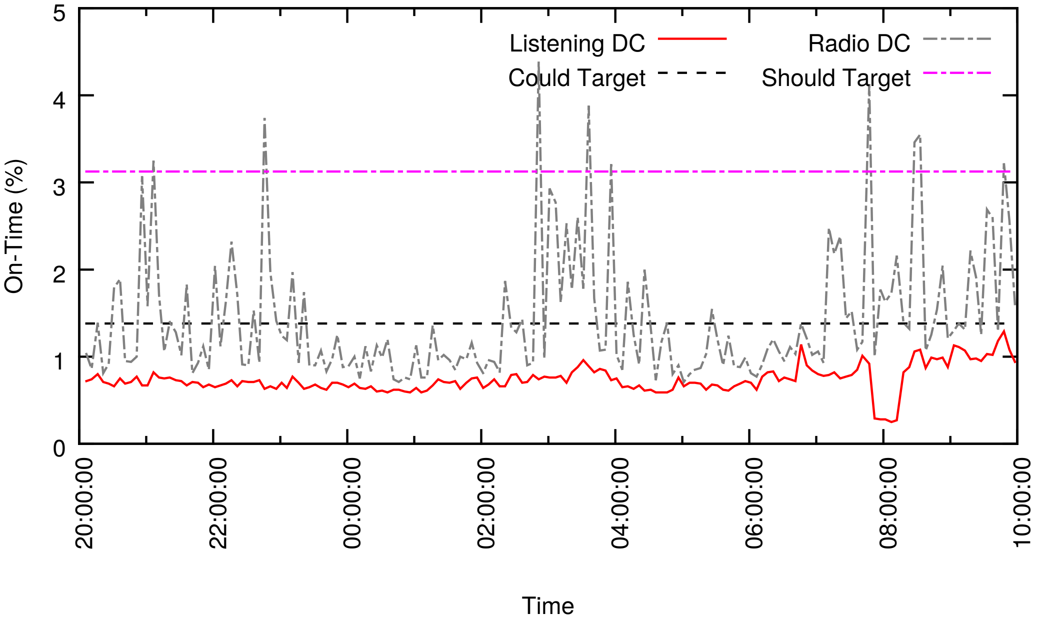


Figure 2.19: Total radio duty cycle over the duration of the parking demonstration [46].

Our experiments show that radio interference can have a huge impact on the number of false wake-ups and hence on the energy consumption of the wireless sensor nodes, given that the radio is the most power-hungry component of a wireless sensor node. Figure 2.18 examines idle energy consumption under artificially generated interference [46]. The average radio on-time is approximately 0.65% in absence of interference, but bursts of interference can cause a sudden spike in the actual idle listening that can significantly reduce battery lifetime.

The runtime assurance component developed within RELYonIT detects the violation of the target energy efficiency (we set an idle energy consumption target of 1.38% to achieve a lifetime of 18 months as desired by the industrial partner) and triggers runtime adaptation. The latter dynamically adjust the channel check rate to bring the idle listening on-time down. Figure 2.19 illustrates the duty cycle of all modes of the radio: idle listening, transmitting, and receiving. The total radio duty cycle averaged 1.42%, just slightly higher over the 1.38% target corresponding to 18 months of battery lifetime ("could" requirement), and significantly less than the duty cycle required to satisfy the "must" requirement of four months and the "should" requirement of 8 months. Full results of our experiments can be found in [46].

In summary, experimental results have shown that the components developed within the RELYonIT project successfully provide probabilistic bounds on the performance of protocols. The pilot deployments have indeed shown that the integrated prototype provides communication performance within the use case requirements of each application despite the harsh environmental conditions, and proved that the RELYonIT system successfully upholds the performance within the required range. With the increased reliability, higher performance, and increased battery lifetime that these results entail, our industry partners envision that the outcome of the RELYonIT project will have a positive impact on their future business models.

3 Potential Impact

In this chapter we present the potential impact of the work carried out within RELYonIT and describe the research impact in Section 3.1, the impact on industry in Section 3.2, as well as the socio-economic impact in Section 3.3. We further detail on the exploitation of results (Section 3.4), as well as on the main dissemination activities carried out throughout the project (Section 3.5).

3.1 Impact on Research

RELYonIT has advanced the state of the art in IoT research by providing a systematic framework that enables dependable applications by taking into account relevant environmental properties and their impact on IoT platforms and communication protocols, as well as by creating environment-aware protocols that can be automatically configured to meet application-specific dependability requirements.

To provide such a dependability framework for the IoT, RELYonIT has extended existing FIRE testbed infrastructures with the ability to record and playback realistic environmental effects, namely temperature variations and radio interference. Such testbeds augmented with realistic environmental effects simplify experimentation and help researchers to better explore the challenging interaction of wireless sensor and actuator networks (WSAN) with their surrounding environment without the need of carrying out labour-intensive and costly real-world experiments. The augmented testbeds further allow repeatability of specific environmental conditions and hence enable the evaluation of novel, dependable IoT communication protocols, as well as a quantitative comparison of the performance of different approaches. Hence, among others, the work produced by RELYonIT enables researchers to rapidly evaluate the dependability of communication protocols against specific environmental conditions that can be found in the real-world. For example, the JamLab tool has been used by a number of international researchers working in the WSAN field to evaluate their protocols, and their contributions have been published in several top-tier scientific conferences. These contributions include a burst forwarding technique to allow high throughput data transport in lossy wireless networks [18], an extension of low-power listening that performs channel hopping [1], a contention resolution mechanism for low-power wireless networks [34], as well as a technique to select the best access point available to transfer data in the presence of interference [19].

RELYonIT also attracted the attention of the research community on the problems caused by challenging environmental conditions on the performance of communication protocols. This especially applies to the important role played by ambient temperature on the performance of low-power wireless communication protocols. Our results showing that temperature can dramatically affect the communications between wireless sensor nodes aroused significant interest at conference presentations and demonstrations, allowing new research in the area (e.g., [25, 40]) and motivating further research on other environmental factors such as mobility.

Furthermore, due to the modular approach of the RELYonIT framework, other researchers can easily exploit ideas at different levels (e.g., use the optimized or newly-developed protocols independently of the overarching tool-chain), as well as improve existing mechanisms and contributing new ones. For example, new models and protocols can be easily added to the framework that can hence be used as a basis to research dependable configurations for a large variety of IoT applications.

3.2 Impact on Industry

RELYonIT tools can significantly help in increasing the dependability of IoT applications by allowing the network to meet specific performance requirements. These tools can bring enormous advantages for companies offering end-to-end services on top of ultra low-power wireless sensor technology.

Already within the project, we have analysed the benefits introduced by RELYonIT solutions in two specific application scenarios (see also D-4.4 [46]):

1. *smart parking*, where the energy-consumption of sensor nodes was kept below a minimum threshold despite the congestion in the radio spectrum, allowing to prolong the battery lifetime of sensor nodes and hence to significantly reduce expensive maintenance operations;

2. *civil infrastructure monitoring*, where the effects of temperature variations on communications were minimized, allowing the wireless network to sustain a high packet delivery rate and meet not only the minimal performance requirements for the specific scenario, but also the desired ones.

These results have shown that the increased dependability of IoT communications does not only allow industry to refine and improve the quality of their existing line of products, but also to investigate new markets that were before unexplored because of the intrinsic unreliability of low-power wireless communications in challenging environments. Our industrial partner Worldsensing, for example, is extending their set of products with resilient multi-hop long-range nodes for industrial applications such as structural health monitoring. They are furthermore considering to expand their business to markets such as tracking single individuals in a city by means of scanning their portable devices (smart phones, Bluetooth hands free, smart watches, or any sort of wearable device). This new family of products can apply to pedestrian pattern detection, branding and advertisements, security and privacy, with a huge market space still to be exploited.

RELYonIT has also an important role in supporting the fast development of dependable IoT applications for challenging and hazardous environments. Fast time to market plays an important role in this rapidly evolving domain, and RELYonIT solutions allow European industry to dominate the market of innovative and complex IoT applications by producing cutting-edge products that are able to operate reliably even under the most difficult environmental conditions. Our industrial partner ACCIONA Infraestructuras, for example, will apply RELYonIT improvements on their future construction projects in different climate areas (e.g., tropical, desert, and alpine) to reliably study the insulation properties of new materials used in the construction of buildings.

RELYonIT solutions also address the lack of reliability of low-power wireless communications that has significantly impaired the quick adoption of wireless sensor technology by industry in the past [28, 30]. The ability to meet specific performance requirements will allow wireless sensor networks and the Internet of Things to move from pure monitoring to control applications. As control tasks often require a high dependability from the underlying wireless infrastructure than pure monitoring tasks, the RELYonIT results will allow application domains for the Internet of Things to expand and include more critical domains such as the smart grid, as well as traffic and air-plane control. Increased dependability will also enable industry to expand the realm of the Internet of Things to more critical application domains in surveillance.

For example, the RELYonIT results have inspired industrial research activities at Fraunhofer IIS in Dresden, Germany, where the focus is on industrial automation systems with wireless technologies in harsh environments. Further RELYonIT results have been exploited in the context of dependable wireless systems in the automotive fields (DEWI Artemis Project) in a joint use case with AVL List GmbH, Austria, and in the context of water management systems in a joint project the China Electronic Technology Corporation (CETC).

3.3 Impact on Society

The number of devices that will be connected to the Internet will rise to 50 billion by 2020 [27], with the majority being small embedded devices with sensing capabilities. This will result in an Internet of Things (IoT) in which low-power wireless sensor networks will represent the bridge between the physical and the digital world and will actually become an integral part of the daily life of millions of people. Many experts believe that this revolution will have much larger impact than previous ones, even larger than the Internet revolution that interconnected computers in the late 90's [48].

The IoT will therefore embrace a system of wireless networks that can deliver to end-users a plethora of services and attractive applications (e.g., smart cities, smart grids, and smart healthcare). At the same time, these systems will heavily rely on the dependable and predictable operation of networked embedded wireless sensors and actuators. Hence, in the years to come, wireless sensor networks will be expected to meet application-specific dependability requirements, and to minimize the impact of the environment on their performance.

The results obtained within RELYonIT play a fundamental role in this domain. By providing a systematic framework and tool-chain to enable dependable IoT applications that take into account all relevant environmental properties and their impact on IoT platforms and communication protocols, RELYonIT has the potential to drive the adoption of low-power wireless technology, therefore amplifying the effects that the latter has on society.

Dependable IoT applications can greatly contribute to address some of the most urgent challenges of our modern society such as the increasing size and age of the population, the need for public safety and security, as well as environmental sustainability. Smart health care can improve the living comfort of elderly and ill people, smart cities can make the life in dense urban environments more comfortable, smart surveillance can increase safety and security, and smart grids can improve the efficiency of production, distribution, and consumption of energy. The use cases we have considered and validated within the project are representative sample of those application domains and have shown that RELYonIT technology can indeed be exploited to make IoT applications more dependable also in the presence of adverse environments. Furthermore, the project has been in contact with public authorities to use the RELYonIT results towards more sustainable cities with optimized traffic and minimal air pollution. Specifically a joint workshop has been held with representatives of the Styrian State Government, Austria, where the potential of using RELYonIT results within this context has been explored. Similarly, a joint workshop with the Ambassador of Singapore to Austria has been held in Graz, where, among others, the potential of RELYonIT technology to improve the air quality and traffic distribution in dense urban environments has been explored.

3.4 Exploitation

In this section, we describe the exploitation activities of the industrial partners Worldsensing and ACCIONA Infraestructuras during project execution, and further highlight future internal and external exploitation opportunities.

3.4.1 Exploitation Efforts of Industrial Partner Worldsensing

On a regular basis, meetings have been organized to align the different business units of Worldsensing on the results of the RELYonIT project. This allowed to promptly identify which components of RELYonIT could be beneficial for the existing line of products.

As shown by the demonstrator in Barcelona [46], a primary beneficiary of RELYonIT improvements is Worldsensing's prime product, FastPrk [50]. The latter is targeting the outdoor smart parking market, being it privately owned (such as shopping malls) or public (such as town halls). The technology is composed of sensors installed in each parking spot that communicate wirelessly with an Internet-enabled gateway to inform about the absence/presence of a car. FastPrk addresses two problems: (i) the obvious headache of losing a lot of time, money, and health by not being able to find a parking spot quickly, as well as (ii) the traffic problem and occupation optimization in cities. For such monitoring system, reliability of communications in any environment is a fundamental aspect, and the marketability of the overall product strongly depends on the ability of the IoT application to meet specific performance requirements. As we have shown in our final integrated experiment [46], RELYonIT technology helps in significantly reducing the energy-consumption of sensor nodes despite the congestion in the radio spectrum, allowing to prolong the battery lifetime of sensor nodes and hence to significantly reduce expensive maintenance operations.

Besides FastPrk, RELYonIT results can also be applied in a number of products of Worldsensing's Loadsensing business unit (currently named Industrial Division). Loadsensing is the line of products designed to wirelessly monitor infrastructure assets such as bridges or tunnels where ambient conditions can be extreme. For such monitoring systems, reliability of communications is a fundamental aspect. Furthermore, the Industrial Division branch also focuses on structural health monitoring and hazard control, with the deployment of wireless sensors to monitor different parameters of buildings, tunnels, pillars or other structures. These markets' requirements are very high in terms of reliability and quality control, and RELYonIT solutions can help in increasing the gap causing current deployment to still use wired communication. Worldsensing is confident to achieve the high-grade requirements for these domains also using wireless solutions, allowing the company to enter and profitably exploit this huge potential market.

The enhancement on radio communications can also be suitable for WOS products such as SpiderNano. This new product family is a seismic data acquisition unit with real-time characteristics and high throughput for geological markets, such as CO_2 reservoir control, and dam monitoring. These systems are typically deployed in hazardous environments like jungles, deserts, or tundra in case of oil prospecting. The product is currently under development, and its engineering team has been periodically updated with RELYonIT achievements, so that the latter could be embedded in the ongoing product design.

Furthermore, the increase in reliability due to interference control and temperature compensation enables a very low energy consumption and will hence give the company the chance of adding side-products such as in-field assets and/or person management and tracking to their portfolio.

3.4.2 Exploitation Efforts of Industrial Partner ACCIONA

A number of face to face meetings among the different ACCIONA business units took place regularly in order to analyse the results of the RELYonIT project and determine how to take advantage of its results. Already during the execution of the project, it was clear that the visible increase in the dependability of low-power wireless systems introduced by RELYonIT solutions was of potential interest for all three business units of ACCIONA, namely ACCIONA Services, Transmediterranea, and Infaestructures. In particular, our business meetings identified especially two application scenarios within the infrastructure construction field in which RELYonIT technologies could be immediately applied:

- *Pressure measurement application in semi-submersible offshore concrete structure*, in which the key aspect is to collect pressure measurements at different locations. As ACCIONA is currently building the Oceanic Platform of the Canary Islands (`http://www.plocan.eu/en/home-2.html`), dependable wireless communications despite temperature variations is a primary concern.

- *Thermal properties measurement application of traditional insulation systems for energy efficient buildings*, in which the key aspect is to carefully test the insulating properties of materials used in the constructions of buildings. As ACCIONA aims to build energy-efficient buildings, and as these tests are typically carried out in specialized laboratories, ACCIONA would like to go one step further and test such insulating properties directly into the field. This however, requires a dependable data collection.

After a thorough discussion among the members of the consortium, it was agreed to include a new use case demonstration within the project, although not initially planned by the original work plan. This led to the deployment of a network within the DEMOPARK facility in Madrid [46] to evaluate the robustness of RELYonIT's newly-designed communications protocols to temperature variations. The impressive results achieved in the final integrated experiment at the DEMOPARK, i.e., the ability to minimize the impact of temperature changes up to 55 °C and achieve not only the minimal dependability requirements on packet delivery rate, but even offer better performance, convinced the company to adopt RELYonIT improvements in all the wireless sensor networks to be deployed in future material tests. This especially applies to the networks in which the nodes are deployed outdoors, e.g., on the façades, walls, and roofs of buildings.

After project completion, a stable collaboration group will be created, which includes one researcher from the New Material group and one researcher from the ICT group. This will facilitate the utilization of RELYonIT technology and know how, as well as simplify the technical support between the involved business units. Such new working group will coordinate additional deployments to further validate RELYonIT results on outdoor infrastructures. In particular, wireless sensor networks will be deployed on buildings in which new insulating materials have been installed (both newly-constructed or refurbished ones), and the communications among the nodes will be thoroughly analysed and compared against the expected performance requirements. The expectation is that the use of RELYonIT solutions will allow the network to lose a minimal number of packets while being operative, despite the temperature variations that will occur in the deployment scenario. Depending of the success of the whole process, the managers of the different business units involved will make a decision about the feasibility of incorporating this service into the formal procedures within the company. If the outcome is positive, the use of the RELYonIT results will be adopted by units in respective geographical zones (e.g., if a first successful test is carried out in Australia, all the construction units there will implement these solutions). The use of dependable IoT solutions will give ACCIONA a significant advantage in a highly competitive market.

3.5 Main Dissemination Activities

Starting from the very beginning of the project, the goals and the results of RELYonIT were continuously disseminated to industry, academia, and to the general public through a number of different means. The following sections describe different levels of dissemination activities for promoting the wide adoption of our research and technology, ranging from publications at leading conferences, workshops, and journals in our research field (Section 3.5.1) as well as demonstrations at a number of scientific events (Section 3.5.2) to presentations and press releases arousing the interest of potential future users in industry and regular updates on our project Website (Section 3.5.3).

3.5.1 Publications

A significant number of RELYonIT-related articles has been published at a number of leading conferences, workshops, and journals in our research field. This includes top-tier conferences such as the ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN), the IEEE International Real-Time Systems Symposium (RTSS), the ACM Conference on Embedded Networked Sensor Systems (SenSys), the IEEE International Conference on Mobile Ad-Hoc and Sensor Systems (MASS), as well as flagship journals on WSN research such as the ACM Transactions of Sensor Networks (TOSN). Most of the articles have been co-authored by two or more project partners. In the following, we list in chronological order a selection of our major scientific publications:

- C.A. Boano, M.A. Zúñiga, K. Römer, and T. Voigt. **"JAG: Reliable and Predictable Wireless Agreement under External Radio Interference"**. In Proc. of the 33rd

IEEE International Real-Time Systems Symposium (RTSS). San Juan, Puerto Rico, December 2012 [6].

- F. Hermans, O. Rensfelt, T. Voigt, E. Ngai, L.-Å. Nordén, and P. Gunningberg. **“SoNIC: Classifying Interference in 802.15.4 Sensor Networks”**. In Proc. of the 12th ACM/IEEE Conference on Information Processing in Wireless Sensor Networks (IPSN), Philadelphia, Pennsylvania,USA, April 2013 [22]. *Best Paper Runner-Up.*

- C.A. Boano, H. Wennerström, M.A. Zúñiga, J. Brown, C. Keppitiyagama, F.J. Oppermann, U. Roedig, L.-Å. Nordén, T. Voigt, and K. Römer. **“Hot Packets: A Systematic Evaluation of the Effect of Temperature on Low Power Wireless Transceivers”**. In Proc. of the 5th Extreme Conference on Communication (ExtremeCom). Thórsmörk, Iceland, August 2013 [7]. *Best Paper Award.*

- C. Keppitiyagama, N. Tsiftes, C.A. Boano and T. Voigt. **“Temperature Hints for Sensornet Routing”**. In Proc. of the 11th ACM Conference on Embedded Networked Sensor Systems (SenSys), poster session. Rome, Italy, November 2013 [25].

- C.A. Boano, M.A. Zúñiga, J. Brown, U. Roedig, C. Keppitiyagama, and K. Römer. **“TempLab: A Testbed Infrastructure to Study the Impact of Temperature on Wireless Sensor Networks”**. In Proc. of the 13th International Conference on Information Processing in Sensor Networks (IPSN), Berlin, Germany, April 2014 [10].

- B. Al Nahas, S. Duquennoy, V. Iyer, and T. Voigt. **“Low-Power Listening Goes Multi-channel”**. In Proc. of the 10th IEEE International Conference on Distributed Computing in Sensor Systems (DCOSS), Marina Del Rey, CA, USA, May 2014 [1].

- W.-B. Pöttner, H. Seidel, J. Brown, U. Roedig, and L. Wolf. **“Constructing Schedules for Time-critical Data Delivery in Wireless Sensor Networks”**. ACM Transactions on Sensor Networks (TOSN), vol. 10, no. 3, August 2014 [35].

- J. Brown, U. Roedig, C.A. Boano, and K. Römer. **“Estimating Packet Reception Rate in Noisy Environments”**. In Proc. of the 9th IEEE International Workshop on Practical Issues in Building Sensor Network Applications (SenseApp) in conjunction with the 39th IEEE Conference on Local Computer Networks (LCN). Edmonton, Canada, September 2014 [13].

- C.A. Boano, K. Römer, and Nicolas Tsiftes. **“Mitigating the Adverse Effects of Temperature on Low-Power Wireless Protocols”**. In Proc. of the 11th IEEE International Conference on Mobile Ad hoc and Sensor Systems (MASS). Philadelphia, PE, USA, October 2014 [9].

- F.J. Oppermann, C.A. Boano, M. Zimmerling, and K. Römer. **“Automatic Configuration of Controlled Interference Experiments in Sensornet Testbeds”**. In Proc. of the 12th ACM Conference on Embedded Networked Sensor Systems (SenSys), poster session. Memphis, TN, USA. November 2014 [31].

- V. Iyer, F. Hermans, and T. Voigt. **"Detecting and Avoiding Multiple Sources of Interference in the 2.4 GHz Spectrum"**. In Proc. of the 12th European Conference on Wireless Sensor Networks (EWSN). Porto, Portugal, February 2015 [24]. *Best Paper Award.*

A complete list can be of RELYonIT publications can be found in [2] and [3], as well as on `http://www.relyonit.eu/`.

3.5.2 Demonstrations

The progress and results of the RELYonIT project were also presented by means of practical demonstrations at a number of scientific events. Besides disseminating results and ideas to the research community, this also provided us with an excellent opportunity to receive feedback on the employed approach and developed solutions, as well as to establish contacts for future collaborations. In the following, we list in chronological order the practical demonstrations that we presented at scientific events:

- J. Brown, U. Roedig, C.A. Boano, K. Römer, and N. Tsiftes. **"How Temperature Affects IoT Communication"**. In Adjunct Proc. of the 11th European Conference on Wireless Sensor Networks (EWSN), demo session. Oxford, United Kingdom, February 2014 [14]. *Best Demo Runner-up.* See also Figure 3.1(a).
- M. Bor and F.J. Oppermann. **"RELYonIT: Research by Experimentation for Dependability on the Internet of Things"**. Demonstration at the Future Internet Assembly 2014. Athens, Greece, March 2014. See also Figure 3.1(b).
- C.A. Boano, K. Römer, J. Brown, U. Roedig, and M.A. Zúñiga. **"A Testbed Infrastructure to Study the Impact of Temperature on WSN"**. In Proc. of the 11th IEEE Conference on Pervasive Computing and Communications (PerCom), demo session. Budapest, Hungary, March 2014 [8]. See also Figure 3.2.

3.5.3 Publicity

In addition to the previously outlined efforts to disseminate RELYonIT ideas and results to the scientific community, the project also made some efforts to inform the general public about project goals, ideas, and results.

Project Website. Already in an early stage of the project, we established the project website at the address `http://www.relyonit.eu` and constantly updated it with information about the project, its goals, the achieved results, and all latest developments. The use of the `.eu` top-level domain strongly links the project to the European Union and the Commission as the project's co-founder.

Fact Sheet. We created a fact sheet of the project and distributed it at several events. The PDF can also be downloaded from the RELYonIT Website at `http://www.relyonit.eu/fileadmin/user_upload/factsheet.pdf`.

Magazines. The project consortium published the following two magazine articles summarizing scientific results and ongoing activities:

- “RELYonIT: No dependability, no Internet of Things”. FIRE Magazine: Net-Xperiment Future, March 2014 [37];
- “RELYonIT: Dependability for the Internet of Things”. IEEE IoT Newsletter, January 2015 [36].

Press Releases. Press releases served as a way to arouse the interest of potential future users in industry. Several press releases were released by academic and industrial partners on their corporate Websites throughout the project, such as (in chronological order):

- “Verlässliche Werkzeuge für das Internet der Dinge” – University of Lübeck [38];
- “School of Computing and Communications Secures €650k for Future Internet Research” – Lancaster University [26];
- “A Dependable Framework for Internet of Things that will Guarantee Performance” – SICS Swedish ICT [42];
- “RELYonIT – robusta lösningar för Sakernas Internet” – SICS Swedish ICT [43];
- “Smartes Parken und Co.: Wie Informationstechniker der TU Graz das Internet der Dinge zuverlässiger machen” – Graz University of Technology [20].

These press releases were then re-posted and disseminated by a large number of digital newspapers and magazines. A complete list can be found under “Publicity” in [2] and [3].

Media Coverage. Furthemore, within the 28 months of project duration, RELYonIT attracted significant attention in a number of national newspapers such as Die Welt, as well as at international ICT fairs such as CeBIT. Among others, the RELYonIT project was covered by the following media:

- “Wenn der Mähdrescher mit dem Trekker spricht” – Die Welt newspaper [45] (National German newspaper);
- “Internet der Dinge – Maschinen melden sich via Web” – CeBIT ICT fair [16];
- “Smart Cities: Deine Stadt spricht zu Dir!” – New Scientist newspaper [39];
- “Die Welt im Netz” – IBM insider [23];
- ORF (Austrian Broadcasting) – Interview within “Ö1 Digital Leben”;
- Further articles appeared in Die Presse (National Austrian newspaper), der Standard (National Austrian newspaper), Kronenzeitung (National Austrian newspaper);

A complete list of the media coverage can can be found under “Publicity” in [2] and [3].

3.5.4 Further Dissemination Activities

Invited talks and presentations. In addition to the publication of papers and posters at scientific conferences and workshops, the project partners also used the opportunity to introduce the project goals and results as part of *twenty-nine* invited talks and presentations. A complete list can be found under "Presentations" in [2] and [3].

Organization of events. Conferences and workshops provide an adequate platform for the presentation of novel findings, techniques, and their application in theory and practice. These international events are commonly organized by universities and also create valuable opportunities for researchers from academia and industry to get in touch and exchange ideas or discuss visions. RELYonIT project partners participated in the organization of *six* major WSN and IoT conferences: a complete list can be found under "Organization of Events" in [2] and [3].

Teaching activities. Ideas and concepts developed in the RELYonIT project were also integrated into *eleven* of the courses in the teaching curriculum at the academic partners. In addition, *seven* student theses were based on topics with a strong relevance to the project. A complete list of teaching activities can be found under "Teaching" in [2] and [3].

Collaborations with external partners. To supplement the work conducted within the RELYonIT project, we sought cooperation with *eleven* research institutions, among which ETH Zurich, Vienna University of Technology, Fraunhofer IIS, Darmstadt University of Technology, and the China Electronic Technology Corporation (CETC). We have also sought cooperation with a number of European projects such as EVARILOS, makeSENSE, AmpliFIRE, and the DEWI Artemis Project. The complete list of collaborations with external partners can be found in [2] and [3].

Exchange of ideas. Beyond the research and administrative collaborations with other projects, partners of RELYonIT actively approached *seventeen* other research projects and institutions to introduce the RELYonIT project and its results in order to exchange ideas related to the research topics of the project. A complete list of institutions and projects can be found in [2] and [3].

How Temperature Affects
IoT Communications
RELYonIT
RESHAPING INFRASTRUCTURE FOR INNOVATION
RELYonIT
Research by Experimentation for Dependability on the Internet of Things

Figure 3.1: The RELYonIT demonstration at the 11th European Conference on Wireless Sensor Networks (EWSN) held in Oxford, United Kingdom, during February 2014 (a) and the RELYonIT demonstration at the Future Internet Assembly (FIA) during March 2014 (b).

Figure 3.2: The RELYonIT demonstration at the 11th IEEE Conference on Pervasive Computing and Communications (PerCom) held in Budapest, Hungary, during March 2014.

4 Contact Details

Grant Agreement Number: 317826
Project Acronym: RELYonIT
Full Name: Research by Experimentation for Dependability on the Internet of Things

Project Website: `http://www.relyonit.eu`
Project Logo:

Project Coordinator:
Name: Kay Römer
Institution: Technische Universität Graz (TUG)
E-mail: roemer@tugraz.at

Partners:

- SICS Swedish ICT AB (SICS)
  Contact: Thiemo Voigt, thiemo@sics.se

- Technische Universiteit Delft (TUD)
  Contact: Koen Langendoen, K.G.Langendoen@tudelft.nl

- University of Lancaster (ULANC)
  Contact: Utz Roedig, u.roedig@lancaster.ac.uk

- Worldsensing (WOS)
  Contact: Xavier Vilajosana, xvilajosana@worldsensing.com

- Acciona Infraestructures S.A. (ACCIONA)
  Contact: Rafael Socorro, rafaelclaret.socorro.hernandez@acciona.com

Bibliography

[1] B. Al-Nahas, S. Duquennoy, V. Iyer, and T. Voigt, "Low-Power Listening Goes Multi-Channel," in *Proceedings of the* 10^{th} *IEEE International Conference on Distributed Computing in Sensor Systems (DCOSS)*, 2014.

[2] M. Baunach, C. A. Boano, K. Langendoen, P. M. Montero, M. Montón, F. J. Oppermann, U. Roedig, K. Römer, R. Socorro, T. Voigt, and M. Zúñiga, "D-5.1 – report on 1st year cooperation, dissemination and joint activities," http://www.relyonit.eu/, RELYonIT: Research by Experimentation for Dependability on the Internet of Things, Grant Agreement no: 317826, Tech. Rep., Nov. 2013.

[3] M. Baunach, C. A. Boano, K. Langendoen, A. Veiga, M. Montón, F. J. Oppermann, U. Roedig, K. Römer, R. Socorro, T. Voigt, and M. Zúñiga, "D-5.2 – report on 2nd year cooperation, dissemination and joint activities," http://www.relyonit.eu/, RELYonIT: Research by Experimentation for Dependability on the Internet of Things, Grant Agreement no: 317826, Tech. Rep., Jan. 2015.

[4] C. A. Boano, "Dependable wireless sensor networks," Ph.D. dissertation, Graz University of Technology, Graz, Austria, Oct. 2014.

[5] C. A. Boano, T. Voigt, C. Noda, K. Römer, and M. A. Zúñiga, "JamLab: Augmenting sensornet testbeds with realistic and controlled interference generation," in *Proceedings of the* 10^{th} *International Conference on Information Processing in Sensor Networks (IPSN)*, Apr. 2011, pp. 175–186.

[6] C. A. Boano, M. A. Zúñiga, K. Römer, and T. Voigt, "JAG: Reliable and predictable wireless agreement under external radio interference," in *Proceedings of the* 33^{rd} *IEEE International Real-Time Systems Symposium (RTSS)*. IEEE, Dec. 2012, pp. 315–326.

[7] C. A. Boano, H. Wennerström, M. Zúñiga, J. Brown, C. Keppitiyagama, F. J. Oppermann, U. Roedig, L.-Å. Nordén, T. Voigt, and K. Römer, "Hot Packets: A systematic evaluation of the effect of temperature on low power wireless transceivers," in *Proceedings of the* 5^{th} *Extreme Conference on Communication (ExtremeCom)*, Aug. 2013, pp. 7–12.

[8] C. A. Boano, K. Römer, J. Brown, U. Roedig, and M. A. Zúñiga, "Demo abstract: A testbed infrastructure to study the impact of temperature on wsn," in *Proceedings of the* 11^{th} *IEEE Conference on Pervasive Computing and Communications (PerCom), demo session*. IEEE, Mar. 2014, pp. 154–156.

[9] C. A. Boano, K. Römer, and N. Tsiftes, "Mitigating the adverse effects of temperature on low-power wireless protocols," in *Proceedings of the* 11^{th} *International Conference on Mobile Ad hoc and Sensor Systems (MASS)*. IEEE, Oct. 2014.

[10] C. A. Boano, M. Zúñiga, J. Brown, U. Roedig, C. Keppitiyagama, and K. Römer, "TempLab: A testbed infrastructure to study the impact of temperature on wireless sensor networks," in *Proceedings of the* 13^{th} *International Conference on Information Processing in Sensor Networks (IPSN)*, Apr. 2014, pp. 95–106.

[11] J. Brown, I. E. Bagci, U. Roedig, M. A. Zúñiga, C. A. Boano, N. Tsiftes, K. Römer, T. Voigt, and K. Langendoen, "D-1.2 - Report on Learning Models Parameters," http://www.relyonit.eu/, RELYonIT: Research by Experimentation for Dependability on the Internet of Things, Grant Agreement no: 317826, Tech. Rep., Nov. 2013.

[12] J. Brown, U. Roedig, C. A. Boano, and K. Römer, "Estimating packet reception rate in noisy environments," in *Proceedings of the* 9^{th} *IEEE International Workshop on Practical Issues in Building Sensor Network Applications (SenseApp)*, Sep. 2014.

[13] ——, "Estimating packet reception rate in noisy environments," in *Proceedings of the* 9^{th} *International Workshop on Practical Issues in Building Sensor Network Applications (SenseApp)*. IEEE, Sep. 2014, pp. 583–591.

[14] J. Brown, U. Roedig, C. A. Boano, K. Römer, and N. Tsiftes, "Demo abstract: How temperature affects iot communication," in *Adjunct Proceedings of the* 11^{th} *European Conference on Wireless Sensor Networks (EWSN), demo session*, Feb. 2014, pp. 40–41.

[15] J. Brown, J. Vidler, I. E. Bagci, U. Roedig, C. A. Boano, F. J. Oppermann, M. Baunach, K. Römer, M. A. Zuniga, F. Aslam, and K. Langendoen, "D-1.3 - Report on Runtime Assurance," http://www.relyonit.eu/, RELYonIT: Research by Experimentation for Dependability on the Internet of Things, Grant Agreement no: 317826, Tech. Rep., Nov. 2014.

[16] Deutsche Messe AG, "Internet der Dinge – Maschinen melden sich via Web," 2013. [Online]. Available: http://www.cebit.de/de/ueber-die-messe/news-trends/rueckblick-cebit-2013/cebit-2013-trendthemen/internet-der-dinge

[17] A. Dunkels, "The ContikiMAC radio duty cycling protocol," Swedish Institute of Computer Science, Kista, Sweden, Tech. Rep. T2011:13, Dec. 2011.

[18] S. Duquennoy, F. Österlind, and A. Dunkels, "Lossy links, low power, high throughput," in *Proceedings of the* 9^{th} *ACM Conference on Embedded Networked Sensor Systems (SenSys)*, Nov. 2011, pp. 12–25.

[19] H. Fotouhi, M. A. Zúñiga, M. Alves, A. Koubâa, and P. J. Marrón, "smart-HOP: a reliable handoff mechanism for mobile wireless sensor networks," in *Proceedings of the* 9^{th} *European Conference on Wireless Sensor Networks (EWSN)*, Feb. 2012, pp. 131–146.

[20] Graz University of Technology, "Smartes parken und co.: Wie informationstechniker der tu graz das internet der dinge zuverlässiger machen," Jan. 2015. [Online]. Available: http://presse.tugraz.at/pressemitteilungen/2015/27.01.2015.htm

[21] V. Handziski, A. Köpke, A. Willig, and A. Wolisz, "TWIST: a scalable and reconfigurable testbed for wireless indoor experiments with sensor networks," in *Proceedings of the* 2^{nd}

International Workshop on Multi-hop Ad-Hoc Networks: from Theory to Reality (REALMAN), May 2006, pp. 63–70.

[22] F. Hermans, O. Rensfelt, T. Voigt, E. Ngai, L.-A. Norden, and P. Gunningberg, "Sonic: Classifying interference in 802.15.4 sensor networks," in *Proceedings of the* 12^{th} *International Conference on Information Processing in Sensor Networks (IPSN)*, Apr. 2013, pp. 55–66.

[23] IBM Deutschland GmbH, "Die Welt im Netz," *IBM Insider*, no. 1, 2013. [Online]. Available: http://www-01.ibm.com/software/de/insider/ausgaben/2013/IBM_S1_12064_BAO_Insider_1-13_LowRes.pdf

[24] V. Iyer, F. Hermans, and T. Voigt, "Detecting and avoiding multiple sources of interference in the 2.4 ghz spectrum," in *Proceedings of the* 12^{th} *European Conference on Wireless Sensor Networks (EWSN)*, Feb. 2015, pp. 35–51.

[25] C. Keppitiyagama, N. Tsiftes, C. A. Boano, and T. Voigt, "Poster abstract: Temperature hints for sensornet routing," in *Proceedings of the* 11^{th} *ACM Conference on Embedded Networked Sensor Systems (SenSys), poster session*. ACM, Nov. 2013, pp. 25:1–25:2.

[26] Lancaster University, "School of computing and communications secures €650k for future internet research," Nov. 2012. [Online]. Available: http://www.lancaster.ac.uk/sci-tech/news/001597/school-of-computing-and-communications-secures-650k-for-future-internet-research

[27] N. Lomas, "Online gizmos could top 50 billion in 2020," Bloomberg Businessweek, http://www.businessweek.com/globalbiz/content/jun2009/gb20090629_492027.htm, Jun. 2009.

[28] P. J. Marrón, S. Karnouskos, D. Minder, and A. Ollero, Eds., *The Emerging Domain of Cooperating Objects*. Springer, 2011. [Online]. Available: http://www.springer.com/engineering/signals/book/978-3-642-16945-8

[29] L. Mottola, T. Voigt, F. J. Oppermann, K. Römer, and K. Langendoen, "D-3.1 – report on specification language," http://www.relyonit.eu/, RELYonIT: Research by Experimentation for Dependability on the Internet of Things, Grant Agreement no: 317826, Tech. Rep., Aug. 2013.

[30] ON World Inc., "Wireless Sensor Networks - Growing Markets, Accelerating Demands," http://www.onworld.com/wsn/wirelesssensors.htm, Jul. 2005.

[31] F. J. Oppermann, C. A. Boano, K. Römer, and M. Zimmerling, "Automatic configuration of controlled interference experiments in sensornet testbeds," in *Proceedings of the* 12^{th} *ACM International Conference on Embedded Networked Sensor Systems (SenSys), poster session*. ACM, Nov. 2014, pp. 342–343.

[32] F. J. Oppermann, C. A. Boano, M. Baunach, K. Römer, F. Aslam, M. Zúñiga, I. Protonotarios, K. Langendoen, N. Finne, N. Tsiftes, and T. Voigt, "D-3.2 – report on protocol selection, parameterization, and runtime adaptation," http://www.relyonit.eu/, RELYonIT: Research by Experimentation for Dependability on the Internet of Things, Grant Agreement no: 317826, Tech. Rep., Jan. 2015.

[33] F. Österlind, A. Dunkels, J. Eriksson, N. Finne, and T. Voigt, "Cross-level sensor network simulation with cooja," in *Proceedings of the First IEEE International Workshop on Practical Issues in Building Sensor Network Applications (SenseApp 2006)*, Tampa, Florida, USA, Nov. 2006.

[34] F. Österlind, L. Mottola, T. Voigt, N. Tsiftes, and A. Dunkels, "Strawman: Resolving collisions in bursty low-power wireless networks," in *Proceedings of the* 11*th* *International Conference on Information Processing in Sensor Networks (IPSN)*, Apr. 2012, pp. 161–172.

[35] W.-B. Pöttner, H. Seidel, J. Brown, U. Roedig, and L. Wolf, "Constructing schedules for time-critical data delivery in wireless sensor networks," *ACM Transactions on Sensor Networks (TOSN)*, vol. 10, no. 3, August 2014.

[36] RELYonIT Consortium, "RELYonIT: Dependability for the internet of things," in *IEEE IoT Newsletter*, Jan. 2015.

[37] ——, "RELYonIT: No dependability, no internet of things," in *FIRE Magazine: Net-Xperiment Future*. European Commission's DG CONNECT, Mar. 2014, p. 23.

[38] K. Römer, "Verlässliche Werkzeuge für das Internet der Dinge," Nov. 2012. [Online]. Available: http://www.uni-luebeck.de/aktuelles/pressemitteilung/artikel/verlaessliche-werkzeuge-fuer-das-internet-der-dinge.html

[39] N. Schlüter, "Smart Cities: Deine Stadt spricht zu Dir!" *New Scientist*, Mar. 2013.

[40] F. Schmidt, M. Ceriotti, N. Hauser, and K. Wehrle, "If you can't take the heat: Temperature effects on low-power wireless networks and how to mitigate them," in *Wireless Sensor Networks*, ser. Lecture Notes in Computer Science, T. Abdelzaher, N. Pereira, and E. Tovar, Eds., Feb. 2015, vol. 8965, pp. 266–273.

[41] M. Sha, G. Hackmann, and C. Lu, "Energy-efficient low power listening for wireless sensor networks in noisy environments," in *Proceedings of the* 12*th* *International Conference on Information Processing in Sensor Networks (IPSN)*, Apr. 2013, pp. 277–288.

[42] SICS Swedish ICT, "A dependable framework for Internet of Things that will guarantee performance," in *Activities 2012–2013*, 2013. [Online]. Available: https://www.sics.se/sites/default/files/pub/sics-activities-2012-2013-print.pdf

[43] ——, "Relyonit – robusta lösningar för sakernas internet," Jan. 2015. [Online]. Available: http://www.mynewsdesk.com/se/sics/pressreleases/robusta-loesningar-foer-sakernas-internet-1109629

[44] A. E. Smith and D. W. Coit, "Penalty functions," in *Handbook of Evolutionary Computation*, T. Bäck, D. B. Fogel, and Z. Michalewicz, Eds. Bristol, UK: IOP Publishing Ltd., 1997.

[45] K. Starke, "Wenn der Mähdrescher mit dem Trecker spricht," *DIE WELT*, Feb. 2013. [Online]. Available: http://www.welt.de/sonderthemen/cebit-2013/article113928760/Wenn-der-Maehdrescher-mit-dem-Trecker-spricht.html

[46] N. Tsiftes, N. Finne, Z. He, T. Voigt, F. Aslam, I. Protonotarios, M. A. Zúñiga, K. Langendoen, C. A. Boano, F. J. Oppermann, K. Römer, M. Baunach, J. Brown, U. Roedig, I. E. Bagci, J. Vidler, A. Veiga, R. S. Hernández, M. Montón, and J. C. Pacho, "D-4.4 - Final Integrated Prototype and Experiment," http://www.relyonit.eu/, RELYonIT: Research by Experimentation for Dependability on the Internet of Things, Grant Agreement no: 317826, Tech. Rep., May 2014.

[47] N. Tsiftes, T. Voigt, F. Aslam, I. Protonotarios, M. A. Zúñiga, K. Langendoen, C. A. Boano, F. J. Oppermann, K. Römer, M. Baunach, J. Brown, U. Roedig, P. M. Montero, R. S. Hernández, M. Montón, and J. C. Pacho, "D-4.3 - First Integrated Prototype and Experiment," http://www.relyonit.eu/, RELYonIT: Research by Experimentation for Dependability on the Internet of Things, Grant Agreement no: 317826, Tech. Rep., May 2014.

[48] J.-P. Vasseur and A. Dunkels, *Interconnecting Smart Objects with IP - The Next Internet.* Morgan Kaufmann, 2010. [Online]. Available: http://TheNextInternet.org/

[49] H. Wennerström, F. Hermans, O. Rensfelt, C. Rohner, and L.-A. Nordén, "A long-term study of correlations between meteorological conditions and 802.15.4 link performance," in *Proceedings of the* 10^{th} *IEEE International Conference on Sensing, Communication, and Networking (SECON)*, Jun. 2013, pp. 221–229.

[50] Worldsensing. The FastPrk website. [Online]. Available: http://www.fastprk.com

[51] M. A. Zúñiga, C. A. Boano, J. Brown, C. Keppitiyagama, F. J. Oppermann, P. Alcock, N. Tsiftes, U. Roedig, K. Römer, T. Voigt, and K. Langendoen, "D-1.1 - Report on Environmental and Platform Models," http://www.relyonit.eu/, RELYonIT: Research by Experimentation for Dependability on the Internet of Things, Grant Agreement no: 317826, Tech. Rep., Jun. 2013.

[52] M. A. Zúñiga, I. Protonotoarios, S. Li, K. Langendoen, C. A. Boano, F. J. Oppermann, K. Römer, J. Brown, U. Roedig, L. Mottola, and T. Voigt, "D-2.2 & D-2.3 - Report on Protocol Models & Validation and Verification," http://www.relyonit.eu/, RELYonIT: Research by Experimentation for Dependability on the Internet of Things, Grant Agreement no: 317826, Tech. Rep., Nov. 2014.