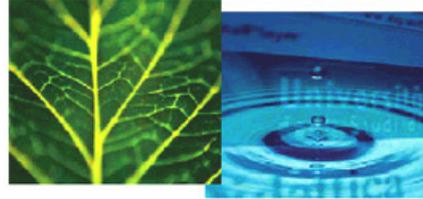


PhD Dissertation



International Doctorate School in Information and
Communication Technologies

DISI - University of Trento

MEASURING, UNDERSTANDING, AND ESTIMATING
THE INFLUENCE OF THE ENVIRONMENT
ON LOW-POWER WIRELESS NETWORKS

Ramona Marfievici

Advisor:

Prof. Gian Pietro Picco

Università degli Studi di Trento

Co-Advisor:

Dr. Amy Lynn Murphy

Fondazione Bruno Kessler, Trento

April 2015

To my parents
Părinților mei

“Be good!”

—E.T. the Extra-Terrestrial

The following document, written under the supervision of **Prof. Gian Pietro Picco** and **Dr. Amy L. Murphy**, was reviewed by:

Prof. Mário Alves Politécnico do Porto, Portugal

Prof. Alberto Cerpa University of California, Merced, USA

Prof. Marco Zúñiga Technische Universiteit Delft, The Netherlands

Abstract

After a decade and a half of research in academia and industry, wireless sensor networks (WSNs) are seen as a key infrastructure able to monitor the environment in which they are immersed, thanks to their miniaturization, autonomy, and flexibility. Still, outdoor deployments of WSNs (e.g., in forests) are notoriously difficult to get right, partly due to the fact that their low-power wireless communication is greatly affected by the characteristics of the target environment (e.g., temperature, humidity, foliage). In the absence of quantitative evidence about the target application environments, the asset that drives a successful and reliable outdoor deployment is the experience gained from previous deployments, lab-like testbeds, or simulators that however often do not resemble the real-world environments.

The general goal of this dissertation is to *support the principled design and deployment of WSNs by improving the understanding of how the natural outdoor environment affects the network stack, and providing tools and modeling techniques to address this impact*. This constitutes the premise for WSNs to be a credible tool for domain experts (e.g., biologists) operating in this field. Our own practical need to design and deploy a reliable WSN system for wildlife monitoring in the mountains near Trento, Italy, pushed our goals towards a deployment and application oriented perspective, whose ultimate objectives are: supporting the WSN deployment; informing the selection or design of protocols, to ensure they are well-suited to the target environment; deriving models to push the envelope of what can be predicted or simulated beforehand.

To achieve these goals we must start from the first step—assessing quantitatively the characteristics of the low-power wireless links *in-field*, i.e., in the environment where the WSN must be deployed. To this end, we contribute with TRIDENT and HARPOON, *tools* for in-field connectivity and routing performance assessment that support principled, repeatable, automated, and flexible collection of measurements in the target environment without the need for a tethered infrastructure and without requiring coding from the end user. Then, using these tools we collect a large set of data traces from six campaigns across different years, environments and seasons, whose *analysis* quantified the impact of the environmental factors on the network stack, focusing primarily on the physical and

routing layers. Finally, we exploit the data traces to create *models* for both estimating the link quality at run-time and reproducing realistic network conditions in simulators.

We argue that the tools we expressly designed for gathering in-field empirical traces, the understanding and quantitative characterization of data traces from real environments, and the modeling, together significantly advance the state of the art by rendering the process of designing and deploying a WSN more repeatable and predictable.

Keywords

Wireless Sensor Networks, Low-power Wireless Communications, Tools, Measurements, Models

Contents

1	Introduction	1
2	Background and Related Work	7
2.1	Long-lived Real-World Deployments	7
2.2	Tools for In-Field Measurements	8
2.3	Empirical Studies	9
2.3.1	Metrics	9
2.3.2	Empirical Studies on Link Properties	10
2.3.3	Impact of the Environment on Low-Power Wireless Networks	11
2.3.4	Empirical Studies on Data Collection Protocols	13
2.4	Models for Characterizing the Behavior of Low-Power Wireless Links	14
3	Measuring: Tools	17
3.1	Requirements	18
3.1.1	Type of Data Collected	18
3.1.2	Type of Experiments Supported	20
3.1.3	Support to Operators	20
3.1.4	Non-functional Requirements	21
3.2	TRIDENT	22
3.2.1	Experiments Execution	22
3.2.2	Toolset Overview	25
3.2.3	Platform-specific Details	26
3.3	HARPOON	30
3.3.1	Experiments Execution	31
3.3.2	Toolset Overview	32
3.4	Summary	33
4	Understanding: Data Traces	35
4.1	Experimental Setup	37

4.1.1	Time and Location of Experiments	37
4.1.2	Hardware Platform	38
4.1.3	Node Placement	39
4.2	Physical Layer	40
4.2.1	Experiment Execution	40
4.2.2	Network-wide Packet Delivery Rate	41
4.2.3	Transitional Region	42
4.2.4	Received Signal Strength Indicator and Link Quality Indicator . . .	44
4.2.5	Link Classification	45
4.2.6	Day vs. Night	49
4.2.7	Link Asymmetry	52
4.2.8	Link Transitions	52
4.2.9	Confirming Observations at Physical Layer	53
4.3	Routing Layer	60
4.3.1	Periodic Data Collection with CTP	61
4.3.1.1	Test Execution	61
4.3.1.2	Impact of Environment on CTP	62
4.3.2	Periodic Data Collection with CTP and ORW	64
4.3.2.1	Test Execution	65
4.3.2.2	Confirming Observations on CTP	66
4.3.2.3	Impact of Environment on CTP and ORW	66
4.4	Application Layer	72
4.4.1	Test Execution	72
4.4.2	Impact of Environment on the Application Layer	73
4.5	Summary	74
5	Estimating and Reproducing: Models	75
5.1	Estimating the Impact of Environmental Parameters	76
5.1.1	Experimental Setup	76
5.1.2	Observations	77
5.1.2.1	Impact of Temperature on Link Quality	77
5.1.2.2	Impact of Humidity on Link Quality	84
5.1.3	Models	85
5.1.3.1	Existing Model of the Effect of Temperature on Link Quality	85
5.1.3.2	A Model of the Effect of Temperature and Humidity on Link Quality	86
5.2	Reproducing Realistic Network Conditions	88

5.2.1	Goals	89
5.2.2	Theoretical Model	89
5.2.3	Evaluation of the Model	92
5.3	Summary	95
6	Conclusions and Outlook	97
	Bibliography	99

List of Tables

3.1	TRIDENT per-round configuration parameters.	24
3.2	In-field commands.	25
3.3	Platform support in TRIDENT.	27
3.4	HARPOON per-round configuration parameters.	31
4.1	Campaigns.	37
4.2	Values for per-round configuration parameters for PHY experiments using the CROSS.	41
4.3	Percentage of asymmetric links on CROSS	52
4.4	Transitions on CROSS	53
4.5	Values for per-round configuration parameters for PHY experiments using the GRID.	54
4.6	Asymmetrics links on GRID	60
4.7	Transitions on GRID	60
4.8	Per-round configuration parameters and their values for GRID topology campaigns.	61
4.9	Delivery rate statistics	68
4.10	Duty cycle statistics	68
4.11	Retransmissions count statistics	68
4.12	<i>ORW</i> L2 duplicates	72
5.1	RSSI-temperature linear function slopes on GRID.	78
5.2	LQI-temperature linear function slopes on summer GRID.	81
5.3	Our proposed model.	88
5.4	Boano et. al [18] model.	88
5.5	Day vs. night <i>PDR</i> distributions example.	89
5.6	Comparison between empirical traces and simulated traces: overall <i>PDR</i> (%).	94

5.7	Comparison between empirical and simulated traces: summer and winter day link classification.	94
5.8	Comparison between empirical and simulated traces: summer and winter night link classification.	95
5.9	Comparison between empirical traces and simulated traces: number of transitions.	96

List of Figures

1.1	Dissertation Contributions.	4
2.1	Connected, transitional and disconnected region example.	10
3.1	Sample TRIDENT experiment showing two rounds, staggered transmissions using single-packet and burst probes, and per-round synchronization.	22
3.2	The TRIDENT toolset.	25
3.3	The “results view” of the TRIDENT in-field assistant, showing one-hop inbound connectivity and <i>PDR</i> for node 0.	26
3.4	ZigBee transmits each broadcast packet three times.	29
3.5	Distribution of packet arrival time.	29
3.6	The HARPOON toolset.	32
4.1	Experimental sites on Mount Bondone.	38
4.2	WSN nodes in the field.	39
4.3	Network topologies for experiments.	39
4.4	Distribution of link distances.	40
4.5	Network-wide average <i>PDR</i> , for all combinations of site (OPEN, SPRUCE, BEECH), season (summer vs. winter) and power (high vs. low).	41
4.6	Transitional areas in OPEN (left), SPRUCE (center), and BEECH (right).	43
4.7	<i>RSSI</i> vs. <i>PDR</i> in OPEN (left), SPRUCE (center), and BEECH (right).	45
4.8	<i>LQI</i> vs. <i>PDR</i> in OPEN (left), SPRUCE (center), and BEECH (right).	46
4.9	Link classification.	47
4.10	Cumulative distribution function of <i>PDR</i> in OPEN (left), SPRUCE (center), and BEECH (right).	47
4.11	Seasonal variations, at high power (left) and low power (right).	48
4.12	An example of day vs. night variations: the <i>PDR</i> of link 2 → 6 and 6 → 2 in OPEN, during summer, at low power. The node distance is 39 m.	49
4.13	Day vs. night variations, at high power (left) and low power (right).	51

4.14	Network-wide average PDR in summer: comparison of the experiments in Section 4.2.2 and 4.2.9, performed one year apart and with different topologies.	55
4.15	Network-wide average PDR in winter: comparison of the experiments in Section 4.2.2 and 4.2.9, performed one year apart and with different topologies.	56
4.16	Meteorological conditions: temperature and humidity (left) and precipitation (right).	56
4.17	Daily variations in OPEN, at high and low power. The sudden drop to zero corresponds to the 1-hour period in which, every two days, we downloaded the data from the flash of the nodes, and replaced their batteries.	57
4.18	Winter link classification, at high power (left) and low power (right).	58
4.19	Day vs night variations, at high power (left) and low power (right).	59
4.20	Delivery ratio for <i>CTP</i> in OPEN (left) and SPRUCE (right) during summer.	63
4.21	<i>CTP</i> beacons in OPEN and SPRUCE, high (left) and low (right) power during summer.	63
4.22	Parent changes in OPEN and SPRUCE, at high power during summer.	63
4.23	Delivery ratio for <i>CTP</i> in OPEN (left) and SPRUCE (right) during winter.	65
4.24	<i>CTP</i> beacons in OPEN and SPRUCE, high (left) and low (right) power during winter.	65
4.25	Delivery ratio for <i>CTP</i> on GRID (4th campaign) and GRID_SMALL (6th campaign).	67
4.26	<i>CTP</i> and <i>ORW</i> delivery ratio on GRID_SMALL.	67
4.27	Delivery ratio for <i>CTP</i> and <i>ORW</i>	69
4.28	Environmental conditions during the experiments.	69
4.29	<i>CTP</i> ACK fails and lost packets variations with temperature, at node 5 (left) and node 6 (right).	70
4.30	<i>CTP</i> ACK fails and lost packets variations with humidity, at node 5 (left) and node 6 (right).	70
4.31	Duty cycle of <i>CTP</i> and <i>ORW</i>	71
4.32	Retransmissions of <i>CTP</i> and <i>ORW</i>	71
4.33	The effect of environment on contact detection.	73
5.1	RSSI and temperature relation.	78
5.2	RSSI and PDR of TMote Sky and Waspote links over a 2-day time period.	79
5.3	TMote Sky RSSI-temperature relation in sauna.	80
5.4	Waspote RSSI-temperature relation in sauna.	81
5.5	Relationship between <i>LQI</i> and temperature during summer.	82

5.6	Relationship between LQI and temperature during winter.	83
5.7	LQI measured by TMote Sky over a 2-day time period.	83
5.8	LQI -temperature in sauna.	83
5.9	PDR-temperature variations during winter(left) and summer (right). . . .	84
5.10	$RSSI$ -humidity in sauna.	85
5.11	Example of two states.	90

Chapter 1

Introduction

“Begin at the beginning”, the king said, very gravely,
“and go on till you come to the end: then stop.”

Lewis Carroll, *Alice in Wonderland*

In 1999, researchers working on “Smart Dust” [37] pioneered the concepts of a large number of resource scarce devices cooperating to achieve a sensing task. Since then, a whole range of algorithms, protocols and programming abstractions have been developed for these wireless sensor networks (WSNs), along with several operating systems and hardware platforms. Moreover, the validity and the performance of the proposed solutions has been extensively tested in laboratories, in simulators and controlled testbeds. To a large part driven by the functionality and performance demonstrated in research laboratories, WSNs are nowadays seen as flexible systems able to provide dense and cost-effective monitoring for application domains that span engineering, scientific, medical and other disciplines. This has created, from the domain experts, enormous expectations around this technology, as an enabler of previously impossible scenarios as well as credible replacement for established solutions.

Motivation. Since the inception of WSNs, many target applications meant that the network would be deployed in *outdoor* settings, where temperature, humidity, foliage, obstacles and other factors are known to affect communication especially in the 2.4 GHz ISM band [1] that has become a popular choice among researchers and practitioners. In fact, examples of successful deployments covering a wide range of scenarios from environmental monitoring [9, 10, 54, 65, 74, 83], to habitat monitoring [25, 49, 56, 73], from precision agriculture [43] to the study of tree canopy climate [48, 77] exist. However, these experiences also clearly demonstrate: (i) how difficult it is to run and manage a WSN in the real-world rather than in an indoor controlled testbed, (ii) that all too often these systems fail to provide the expected results once deployed in their *outdoor* target environ-

ments. This is partly due to the fact that the behavior of their physical communication layer and, as a consequence, of the entire stack, is greatly affected by the characteristics of the environment. However, the lack of proper tools for large-scale, in-field connectivity assessment, and consequently of publicly available traces means that there is limited quantitative evidence for understanding channel behavior in the target environment.

This lack of understanding prevents the development of channel models realistic enough to capture the characteristics of the target application environments, which in turn are needed to predict and estimate link quality at run-time, or reproduce network conditions accurately when simulating the application behavior. As a result, WSN design and deployment is still mostly an *art*, based on rules-of-thumb guidelines gleaned from experience, or lab-like testbeds which have little in common with the real target environments, especially when these are outdoor.

A concrete application. In this dissertation, we are motivated by a real-world wildlife monitoring application and WSN deployment [56] we are pursuing in collaboration with biologists studying the social interactions/contacts among animals, namely, roe deer in the mountains near Trento, Italy.

The state of the art for these studies is based on GPS collars enabling the tracking of animal movements [16], interactions among animals being inferred from the intersection of individual trajectories [34]. Unfortunately, GPS is energy-hungry, hence the sampling rate is typically very low. This fact, along with inaccuracies due to partial sky views, introduces uncertainty and approximation in the trajectories, and therefore on the spatial proximity and interaction among individuals inferred from them. Instead, animal-borne WSN nodes enable biologists to record *directly* the interactions among animals. The low-power wireless radio is used as a proximity sensor, and contacts among animals are inferred based on the message exchanges among nodes. Nevertheless, to be useful to biologists, the interactions recorded with this WSN must be correlated to the characteristics of the environment, as the latter affects message reception. For instance, roe deer, one of the species under study, dwell in various environments with different vegetation, and their social interactions must be tracked year-long, across different weather conditions. A contact between two roe deer at 40 m may be detected in an open field but not in a dense forest. How do the various environmental factors affect the quality of communication and as a consequence the performance of contact detection among the WSN nodes? In our application, the answer to this question affects directly the sensing and therefore the study of social interactions. However, the answer is of more general interest, given the many applications exploiting a WSN in an outdoor environment as knowledge about the effect of the environment on the connectivity may provide insights about the reliability and lifetime of the WSN.

Goals and contributions. The general goal of this dissertation is to

support the principled design and deployment of WSNs by improving the understanding of how the natural outdoor environment affects the network stack, and providing tools and modeling techniques to address this impact.

This constitutes the premise for WSNs to be a credible tool for domain experts (e.g., biologists) in outdoor environments. For this it is necessary to understand and characterize quantitatively and empirically the behavior of the WSN in the target environment.

Today, tools enabling the collection of network connectivity measurements in outdoor real-world environments are lacking. Therefore, large realistic network connectivity datasets are missing in the literature, further limiting the understanding of the behavior of low-power wireless links and the development of realistic models able to predict (e.g., for protocol design) or to reproduce (e.g., for simulation) the behavior of the network links. Moreover, the lack of quantitative evidence about target environments leaves the WSN developers in the dark, without specific guidelines to drive their deployments and tune their systems and protocols.

The main contributions of this dissertation, as depicted in Figure 1.1, can be summarized as:

1. a **toolset** for in-field connectivity and routing protocol performance assessment that supports principled, repeatable, automated and flexible collection of measurements in the target environment, relies only on the WSN nodes without any external infrastructure, and does not require any coding by the end user;
2. the **analysis of a large set of data traces** collected *in vivo* (i.e., in actual real-world environments) in which we quantify the impact of the environmental factors, both in terms of morphology (e.g., vegetation present), as well as daily, seasonal and yearly variations at the physical, routing and application layer;
3. **models** based on these traces, with the distinctive goal to:
 - (a) describing the influence of temperature and humidity on low-power wireless links for **estimating** link quality at run-time;
 - (b) describing the **long-term** behavior of low-power wireless links collected in-field for **reproducing** realistic network conditions in simulators.

The central pillar of this dissertation are the empirical data traces acquired in-field. These entail in-field experiments, notoriously effort-demanding and time-consuming. Unfortunately, state-of-the-art tools (i.e., SCALE [20], SWAT [70], and RadiaLE [5]), require

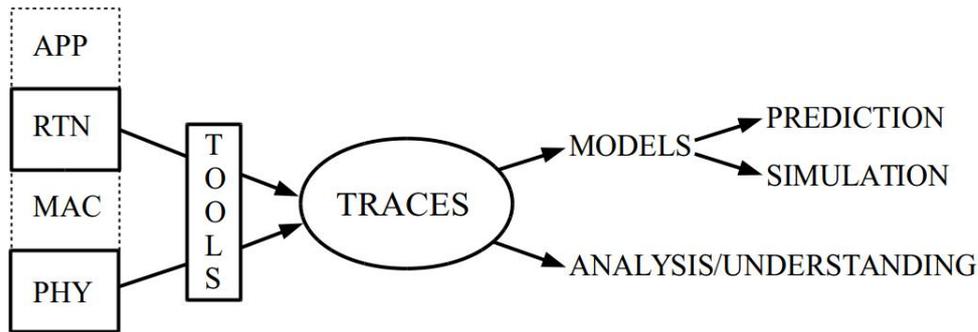


Figure 1.1: Dissertation Contributions.

an infrastructure made by powered devices to which the motes are wired for gathering experimental data — a luxury one can rarely afford in-field. Therefore, the first contribution of this dissertation is embodied in two **tools**, TRIDENT and HARPOON, expressly designed to support principled, repeatable, automated and flexible collection of connectivity measurements and routing protocol performance assessment in the target environment. Unlike similar tools, TRIDENT and HARPOON do not require any communication infrastructure besides the WSN nodes. Our tools cover the entire process concerned with in-field connectivity and routing protocol performance assessment, from the design of the experiments to the download and analysis of the data gathered. Our tools are designed to be easy to use by domain experts (e.g., the biologists we collaborate with) who can perform their experiments without *any* coding effort. TRIDENT is provided in two variants, targeting TMote Sky motes running TinyOS, and Wasmotes running the standard ZigBee stack, covering popular platforms in research and industry, respectively. The tool supports the acquisition of physical layer parameters (e.g., *PDR*, *RSSI*, *LQI*, noise floor), while HARPOON supports the acquisition of routing layer metrics (e.g., data yield and duty-cycle), which account for the reliability of communication, along with the overhead and energy consumption necessary to ensure this reliability in the target environment. HARPOON supports two routing protocols: the Collection Tree Protocol (*CTP*), a representative of commonly employed tree-based routing schemes, and the Opportunistic Routing Protocol (*ORW*), a more recent protocol based on opportunistic routing approach. If the protocol allows, HARPOON acquires additional parameters to assess the effectiveness of specific mechanisms (e.g., number of beacons for *CTP*, number of neighbors for *ORW*). Both tools support the acquisition of *environmental parameters* (e.g., temperature and humidity) from on-board sensors.

The second contribution of this dissertation is the **analysis of a large set of data traces** collected using TRIDENT and HARPOON. The location of the experiments was chosen to be representative of the environment where our target wildlife monitoring application is going to be deployed, and to cover different vegetation conditions. All the

experimental sites are on Mount Bondone, near Trento, Italy. In collaboration with the biologists in our team, we identified three characteristic locations in this area: a meadow with essentially no trees (OPEN), an evergreen forest with sparse vegetation (SPRUCE), and a deciduous forest with dense vegetation (BEECH). The design of the experiments was informed by the biologists interest in understanding how low-power wireless links and as a consequence the contact detection are affected by: *presence and density of vegetation*, *seasonal* and *daily* variations. Our experiments investigated, during six campaigns, across multiple years, what happens if the *same* WSN is immersed in different combinations of the above environmental factors. We quantify the trends emerging at the physical layer and show that: (i) this influence is mirrored at the routing and application layer, (ii) the macro-trends across environments allow us to infer the relative trends in reliability, overhead and energy consumption at the routing layer.

Whilst there are other empirical studies showing that the physical layer is affected by the environment, our study extends up to the routing layer, which bears a direct impact on the application performance. To the best of our knowledge, this represents the first empirical study that characterizes, from a quantitative standpoint, the overall behavior of a WSN in different environments.

As a third contribution, we build on the above analysis to exploit the set of data traces to create two **models**: (i) for **estimating** the link quality and run-time, and (ii) for **reproducing** realistic network conditions in simulators.

First, based on our empirical observations that the link quality of a WSN depends on the characteristics of the surrounding environment, we focus on two factors: temperature and humidity. We study the impact of these two factors on the physical layer parameters in the OPEN field environment as the degradation of the link quality here is not influenced by any other environmental factors (e.g., obstacles, foliage). We conducted a study on the impact of temperature on the *RSSI* and *PDR* of two hardware platforms, TMote Sky and Waspote, and show that the latter is affected by temperature to a much lesser extent w.r.t. the former. Then, we develop a model describing the influence of temperature and humidity on the link quality. This helps estimating the link quality at run-time considering the particular temperature/humidity profile of the target environment and informing the application layer performance.

Second, based on the observations that the link quality distribution of WSN in link quality classes (e.g., dead, poor, intermediate, good, perfect) follows a specific pattern in each of our studied environment, we capture the pattern in a model based on Markov chain theory. Then we train the model with our experimental traces and integrate it in mainstream simulators. Using our analytical model, the simulator generates link quality distributions into classes with similar quality for a specific combination of environment and

season. The model is key for efficiently reproducing realistic network conditions for large-scale simulations of long-term behavior of protocols/applications by accounting for the influence of the environment on the network beforehand. Thus, we contribute at reducing the gap between simulation and real-world performance of protocols and applications.

Dissertation organization. We argue that the tools, described in Chapter 3, we expressly designed for gathering in-field empirical traces, the understanding and quantitative characterization of data traces from real environments, presented in Chapter 4, and the modeling introduced in Chapter 5, together significantly advance the state of the art by rendering the process of designing and deploying a WSN more repeatable and predictable. However, a lot remains to be done and we explore possible future work in Chapter 6.

Chapter 2

Background and Related Work

What's past is prologue.

William Shakespeare, *The Tempest*

This chapter presents background on experiences from outdoor real-world WSNs deployments and their related challenges. Software tools developed to measure and analyze low-power wireless links along with systematic empirical studies on low-power wireless transmissions are reviewed. Finally, existing models characterizing the behavior of low-power wireless links are presented.

2.1 Long-lived Real-World Deployments

Taking a closer look at prior long-lived real-world deployments reveals that putting a functional WSN in place is a non-trivial task. Several research groups have already shared part of their experience in the WSNs deployment field, e.g., X-sense [9], Permadaq [10], Koala [54], Luster [65], WildScope [56], LOFAR-agro [43], GreenOrbs [48], WildScope [56], Great Duck Island [73], Permasense [74], Redwoods [77]. The lessons learned from these deployments highlight a tremendous gap between “*it works in the lab*” and “*it works in the real-world*” [81]. They also emphasize the need to understand the target environment and the impact it will have on the operation of the WSN, as well as the need to take into account the domain knowledge and application requirements to enable successful long-term outdoor deployments. Moreover, they advocate for the need of deployment-time tools to guide the deployment, support in-field evaluation of alternatives and ensure the deployment is up and running before leaving the field. Moreover, there are several examples, TASK [15], Sensorscope [63], Heathland [79], Vigilnet [80], Marionette [84], where a protocol developed on a testbed does not work well in the actual deployment.

However, all these real-world reported experiences focus on the WSN performance in terms of high-level network properties such as end-to-end throughput, data yield and latency and a characterization of the low-power wireless links and routing protocol performance in these scenarios is largely missing. Moreover, the reports that complemented these deployments do not aim to characterize the environment where they are deployed and its variations.

2.2 Tools for In-Field Measurements

The WSN community has recognized the relevance of acquiring information about the properties of the wireless links in the specific environment and built tools to empirically experiment with links. SCALE [20], SWAT [70], RadiaLE [5], and IRIS [28] were conceived for this.

SCALE is built using the EmStar programming model [31] and collects only *PDR*. Each node runs a software stack, allowing for sending and receiving probe packets in round-robin, retrieving packet-statistics and sending them through serial communication. All nodes are connected to a central PC via serial cables and multiplexors. The PC runs different processes, one for each node in the experiment, that perform data collection. Based on the collected data, other processes running on the PC allow for connectivity assessment through the computation of the *PDR* for each unidirectional link. Thus, the network connectivity can be visualized at run-time.

SWAT collects *PDR* and hardware-based metrics, i.e., *RSSI*, *LQI*, *noise* floor. It uses the same type of infrastructure as SCALE, nodes being connected through serial connections or using a back-channel to a central PC. SWAT stores collected data into a database and provides modules for calculating and visualizing various metrics derived from statistics.

RadiaLE makes use of bursty and synchronized traffic patterns for physical layer measurements collection. Nodes are connected to a control station via a combination of USB cables and active USB hubs constituting a USB tree. This tree is used as a reliable logging/control channel between the nodes and the central PC. RadiaLE supports the operator to follow the experiment progress in real-time by displaying the network map, link quality metrics and node status. Moreover, it logs the measurements in a database and enables the evaluation of Link Quality Estimators (LQEs) that can be configured and evaluated based on the collected data from a given experiment.

Unfortunately, SCALE, SWAT, and RadiaLE, require a secondary, wired networking infrastructure for gathering experimental data - a luxury one can rarely afford in real-world settings.

On the other hand, IRIS provides an integrated solution for experiment management and on site data analysis. It supports the automated experiment installation on nodes, the management of measurements, i.e., data organization and logging in different formats (binary, CSV or WiseML format [2]), and customized logging with the help of user-defined functions for direct manipulation of raw data. Furthermore, the tool allows to control the experiment flow and the interaction with the deployed WSN. IRIS is equipped with a set of function templates for data processing and interfaces for visualizing the experiment data.

Unfortunately, IRIS does not support the configuration of the experiment and neither generates the code for the experiments, thus the user has to get into code intricacies and provide the code to be run by IRIS.

In Chapter 3, we present TRIDENT and HARPOON, our tools expressly designed to simplify the chore of in-field connectivity and routing protocol performance assessment. These tools rely only on the WSN nodes whose connectivity needs to be ascertained, without any external infrastructure. They cover the entire workflow concerned with the experiments which can be configured easily without requiring any coding.

2.3 Empirical Studies

The performance of a WSN deployment is obviously directly affected by the quality of the links enabling the communication among nodes. In the case of commonly-used IEEE 802.15.4 radios, the link quality is in turn easily affected, as empirically demonstrated, by environmental factors (i.e, temperature, humidity, rainfall, snow, wind, foliage, obstacles), which cause variations over time and space. Therefore, from a communication perspective, information about the properties of the low-power wireless links in the specific environment at hand is crucial to build reliable systems that run on top of 802.15.4.

2.3.1 Metrics

Researchers have identified several low-level metrics to help them understand network dynamics. The set of basic metrics that were examined by previous empirical studies to capture the low-power link characteristics are: *PDR* (Packet Delivery Ratio), *RSSI* (Received Signal Strength Indicator), *LQI* (Link Quality Indicator) and *SNR* (Signal to Noise Ratio). *PDR* is computed as the ratio of packets received over the number of transmitted packets. It provides an immediate and easy-to-compute indicator of link quality. In addition to *PDR*, physical layer information is another direct indicator of link quality. Most off-the-shelf low-power radio chips provide the *RSSI* and *LQI*, both measured over the first eight symbols (32 bits, 125 μ s) of the received packet. *RSSI*

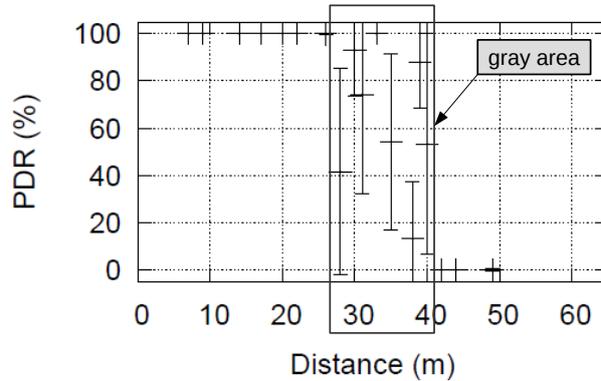


Figure 2.1: Connected, transitional and disconnected region example.

measures the RF signal strength of a received packet. *Noise* floor is the power of the ambient channel energy, when there are no transmissions, and is useful to indirectly determine the presence of interference. *SNR* of a received packet is computed as the difference between the two *RSSI* measures (i.e, *RSSI* and *noise* floor). *LQI* is computed over the 8 bits following the start frame delimiter (SFD) and for the CC2420 radio ranges from 50 (minimum) to 110 (maximum).

2.3.2 Empirical Studies on Link Properties

A significant body of works is related to the understanding of link dynamics, showing how their quality fluctuates over time [22, 69, 87] and space [20, 29, 61, 86, 87, 88, 91], that connectivity is typically asymmetric due to differences in *noise* floor [22, 88] and to imperfections in hardware [91] that may cause variations in the power output during transmissions, and/or antenna orientation [61, 88].

These works demonstrated that the transmission range is not isotropic and is defined by three regions- connected, transitional, and disconnected, based on the distance of the receiver from sender and on the *PDR*, as shown in Figure 2.1. The connected region is the closest to the sender and its links are of good quality. At the other extreme, the disconnected region is the farthest from the sender,; it does not contain links usable for communication. The transitional region, also called the “gray area” is a mix of the two, and contains links that exhibit a high variance. Each of these regions has an irregular shape, with bounds changing over time and with specific features [20, 61, 87, 90].

The transitional region was analyzed during several empirical studies because links within this region are considered unreliable [20, 61, 69, 87, 90, 91]. Zuniga and Krishnamachari [91] quantified the impact of channel multipaths and hardware variance on unreliable and asymmetric links from the transitional region and provide analytical expressions for the boundaries of this region. Cerpa et al. [20] performed indoor and outdoor

measurements using Mica1/Mica2 platforms and different power levels and reported the width of transitional region to range from 50% to 80% of the transmission range. On the other hand, Zhao and Govindan [87], while performing measurements with almost the same settings as of Cerpa et al. [20], reported a smaller width for the transitional region ranging from 20% to 35% and a range for the percentage of intermediate links from 35% to 50%. Moreover, Srinivasan et al. [69] claimed that the number of intermediate links observed with recent platforms, Micaz and TelosB, is lower, from 5% to 60%, than that observed with old platforms. Mottola et al. [53] refuted this observation while conducting experiments in road tunnels using TMoteSky platform. Nevertheless, these studies are based on different network settings in terms of radio type, power, environment and traffic load but all showed that link quality varies drastically over space.

Studies show that the temporal variations of link quality are due to changes in the environment, constructive/destructive interference, obstacles and human presence [21, 87, 61, 75]. Cerpa et al. [21] and Zhao and Govindan [87] show that links from disconnected and connected regions tend to be stable over time, while Srinivasan et al. [69] report on link burstiness and confirm observations made by Cerpa et al. [22].

Moreover, Cerpa et al. [20, 21] analyzed the asymmetry of low-power links and showed that links from transitional region tend to be asymmetric and argued that link asymmetry is not correlated with distance. Srinivasan et al. [69] studied the temporal variation of link asymmetry and found that very few links are long-term asymmetric. Experiments of Mottola et al. [53] refuted this and show that link asymmetry tends to persist in their specific environment.

With few exceptions [20, 53, 69], the above studies were carried out indoor, in controlled environments and present radically different (and sometimes contradicting) results. In contrast, our experiments presented in Chapter 4, are carried out in-field and outdoor. Moreover, these works are generally focused on the study of individual links, usually over short-term variations, to provide guidelines on fine-grained design decisions. In contrast, we are concerned with characterizing the aggregate behavior of the network. In doing this, however, we do borrow some commonly-used concepts, definitions, and techniques from these works, e.g., including the notion of transitional area, the definition of link classes and asymmetric links.

2.3.3 Impact of the Environment on Low-Power Wireless Networks

Several researchers have shown that outdoor WSNs are affected by meteorological factors, i.e., temperature, humidity, rain, fog. Thelen et al. [76] described how radio propagation is favored by high humidity in their potato field deployment and attribute the enhancement to a change in the reflection coefficient on top of the plant foliage at their deployment

site. Works of Anastasi et al. [4] and Son et al. [67] suggest that fog and rain may have a severe impact on the transmission range of WSN nodes, especially w.r.t. packet reception. A loss of connectivity in their outdoor deployments, with rainfalls from 0.4 to 1.4 mm and temperature daily variations between 10 and 50°C, was reported by Sun and Cardell-Oliver [72]. Similarly, Capsuto et al. [17] reports a drop in signal strength during rain and snowfall.

Work by Holland et al. [36] concludes that temperature and humidity have no impact on link quality and data yield while Bannister et al. [7] show that high temperature negatively affects communication, based on data from a radio survey in the desert and applied to simulations of localization and data collection. Daily variations in *RSSI* of up to 6 dBm were reported by Lin et al. [47]. In particular, Boano et al. [13] quantified the impact of these conditions on different platforms in an outdoor industrial setting, showing that light rainfall has a negligible effect on signal strength while heavy rainfall can disrupt connectivity. Using a controlled experiment, Boano et al. [12] show a decrease in *RSSI* as temperature increases and reasons that changes in temperature affect crystal accuracy that induce frequency shifts, and thermal transceiver noise, that may degrade performance. More recent work of Wennerström et al. [82] reports a clear degradation in *PDR* and average link quality during summer and confirm that daily fluctuations of ambient temperature have a strong impact on quality of the communication. Motivated by this study, Boano et al. [18] carry out a systematic set of experiments in controlled settings analyzing how temperature affects the *RSSI* in transmitters and receivers of nodes with CC2420 and CC2520 radios and show that the trend can be captured in a simple first-order model. Schmidt et al. [64] built “HotBox” and reproduced the setup from [18]. They reinforced that temperature has a significant influence on communication quality but reported that heating the receiver produces a larger impact on link quality rather than heating the sender as Boano et al. [18] showed.

Temporal relationship between *RSSI* variability and plant canopy development is observed by Rankine et al. [59] in an old growth stand of deciduous forest. In an effort to understand the causes of losses in a forest deployment, Dong et al. [24] observes periodic performance variations with links severely degrading during night and recovering during day. Moreover, wind movement through vegetation has been observed to increase the variance of *RSSI* due to the changes in multi-path propagation as trees sway in the winds. Hashim and Stavrou [35] show that high wind speeds produces radio signal fading of the 2.4 GHz radio frequency.

All these works, however, simply report the degradation/improvement of the wireless signal as a consequence of changes in the environmental factors like temperature and humidity and do not provide a deeper analysis of what the implications are on routing

protocols and the application when operating a WSN outdoors.

Using the same GreenOrbs deployment as [24], Tong et al. [78] makes one step further and looks at the routing dynamics analyzing possible causes of *CTP* parent change events during several days and their correlation across time and location. Keppitiyagama et al. [40], investigates the effect of temperature fluctuations on RPL protocol [85], in a temperature-controlled testbed, and shows that the performance of ContikiRPL suffers from the short term validity of the ETX [23] predictions under temperature fluctuations.

In Chapter 4 we confirm many of the above mentioned findings. However, in comparison, our work assesses the overall trends induced by the environment in a more holistic way, by taking into account yearly, seasonal and daily variations. It also explores the effect of environments with different characteristics (e.g., vegetation). Finally, we extend up to the routing layer, which bears a more direct impact on the application performance, and the application layer, the one directly relevant to the end user. Moreover, we investigate if the observed influence of the environment at the physical layer is mirrored at the upper layers.

2.3.4 Empirical Studies on Data Collection Protocols

Several studies evaluate the performance of data collection protocols [27, 30, 32, 51, 58] and benchmark the proposed solutions against *CTP*. Gnawali et al. [32] reports *CTP*'s *data yield* ranges from 90.5% to 99.9%, a median *duty-cycle* of 3% across the nodes of a network in experiments in which the network generates data at 30 packets/minute and delivers them to the sink.

The performance of *ORW* protocol was evaluated by Ghadimi et al. [30] in both simulations and testbed-based experiments. They show that *ORW* improves *duty-cycle* and delays significantly while achieving similar reliability and transmission counts when compared to *CTP*. An average decrease in *duty-cycle* by about 50% and up to 90% for individual nodes is reported along with a decrease in delay by 30% to 90% depending on the network density. Moreover, they show that the optimal *ORW duty-cycle* is at lower wakeup rates when compared to *CTP*. At high wakeup rates *ORW* loses some of its benefits and both protocols show similar performance in terms of energy and delay.

The testbed evaluation in Puccinelli et al. [58], where Broadcast-free Collection Protocol is benchmarked against *CTP*, shows a reduction of the *duty-cycle* of the same order of the one achieved by *ORW*.

When compared the Backpressure Collection Protocol [51] against *CTP* on a 40-nodes testbed, a performance by more than 60% in terms of min-max rate and a reduction by more than 30% in terms of average packet transmissions was observed.

Nevertheless, all previous work on analyzing the behavior of routing protocols has been

done in simulations and testbeds. The only exception, represented by Gnawali et al. [32], evaluates *CTP*'s performance on a large-scale, long-term deployment, using *CTP* in an application that collects power and utilization data for the computing infrastructure of a building. In contrast, our experiments are carried outdoor. Not only that we investigate the performance of the data collection protocols but also assess the effectiveness of their specific mechanisms. Moreover, throughout our study we investigate and evaluate the coupling between subtle effects at the physical layer and routing layer behavior.

2.4 Models for Characterizing the Behavior of Low-Power Wireless Links

Several models cover outdoor and indoor signal propagation characterization, taking into account the number, delay and power of indoor multipath components. The log-normal path loss model [60] has been widely use to describe the propagation of radio signals for low-power wireless networks and small-scale models that estimate signal loss over small distances and small time intervals [55] have been proposed as well.

Another class of models, on the other hand, estimate the packet delivery function. These models build functions relating physical layer parameters (e.g., *RSSI*, *SNR*) to *PDR*. The delivery probability is modeled as a function of interference by Reis et al. [62], while Kashyap et al. [39] uses curve-fitting of packet reception probability and *SNR* to model the packet reception. Woo et al. [86] derives a packet loss model based on aggregated statistical measures, assuming a Gaussian distribution of *PDR* for a given transmitter-receiver distance, which later was refuted by Zuniga and Krishnamachari [91]. They use the log-normal path loss model to derive expressions for the distribution, expectation, and variance of *PDR* as a function of distance. A set of non-parametric statistical models for characterizing links and groups of links associated with a particular receiver, transmitter, radio is developed by Cerpa et. al [21].

Recent efforts provide models for estimating the signal strength as a function of temperature. Bannister et al. [7] quantifies the loss of *RSSI* due to temperature for Telos-class nodes, but only for a limited temperature range (i.e., 24 to 65°C) and for a single radio chip. Using TempLab [11] testbed infrastructure, Boano et al. [18] captures the impact of the temperature on the signal strength of transmitters and receivers of two radio chips, i.e., CC2420 and CC2520, and show that the decrease in *RSSI* is consistent among different platforms. Then, a first order platform model describing the impact of the temperature on wireless sensor nodes is proposed. Nevertheless, TempLab temperature profiles are over 0°C, covering just a segment of the outdoor temperature profiles.

Deployments and empirical studies mentioned in Section 2.1 and Section 2.3 indicated

a wide chasm between real-world channel behavior and existing radio channel models in simulators [38]. The latter are often too simplistic, and can hardly capture the complexity of the real-world. To increase the realism of simulations, several models have been proposed and incorporated into existing simulators. Using a multi level approach involving hidden Markov models and mixtures of multivariate Bernoullis, Kamthe et al. [38] proposed to model the long- and short-term scale behavior of links. Zuniga and Krishnamachari [91] modeled the behavior of wireless links depending on the radio and channel characteristics for static and low-dynamic environments, while Lee et al. [45] proposed a statistical model created from noise traces. On the other hand, instead of attempting to create more precise and realistic radio models for simulators, Boano et al. [14] proposed to augment existing simulation tools with the playback of realistic interference traces.

In Chapter 5 we assess the impact of the temperature and humidity on the link quality and propose a model to estimate the link quality at run-time. Moreover, we capture the pattern of links distribution in link quality classes in a model and integrate it in mainstream simulators providing more realistic network conditions for simulations.

Chapter 3

Measuring: Tools

Door meten tot weten.

By measurement to knowledge.

Heike Kamerlingh Onnes

Real-world deployments of WSNs are notoriously difficult to get right, partly due to the fact that their low-power wireless communication is greatly affected by the characteristics of the target environment. Communication in the 2.4 GHz ISM band has peculiar characteristics, that have been studied by many researchers; a summary is provided in Section 2.3.

Unfortunately, the tools supporting connectivity assessment [5, 20, 70] were conceived to study the properties of low-power wireless links using an infrastructure made by powered devices to which the motes are wired, and thus are not directly applicable in most deployed systems where nodes are only wirelessly connected. Further, they are of limited use for characterizing any outdoor environment where it is not practical to run wires for data collection.

This chapter presents TRIDENT and HARPOON, two tools expressly designed to simplify the chore of in-field connectivity assessment and routing performance assessment, respectively. Our tools rely only on the WSN nodes, without any external infrastructure.

TRIDENT and HARPOON are useful to WSN researchers and practitioners, who may use them towards any of the aforementioned goals. However, the tools are designed to be easy to use also for domain experts who do not have a very deep knowledge about the inner working of the WSN, and definitely do not take part in programming it. The experiments can be configured easily without requiring coding, and the data collected with straightforward procedures. TRIDENT and HARPOON cover the entire workflow concerned with the connectivity and protocol performance assessment experiments.

We originally developed TRIDENT for the popular research-oriented platform consti-

tuted by the TMote Sky motes running TinyOS. However, connectivity assessment is relevant also to industry-oriented platforms, for which we chose Wasp mote devices running the standard ZigBee stack as a representative. Interestingly, supporting the latter platform is not simply a matter of porting the code from the former; the fact that the ZigBee stack is “closed”, unlike the TinyOS one, forced us to find ways to reliably measure the main metric of *PDR*, which cannot be derived directly otherwise. HARPOON was originally supporting only *CTP* and later incorporating *ORW*, one of its opportunistic approach competitors.

The chapter is organized as follows. In Section 3.1 we outline the key requirements for both tools established by ourselves in designing them. Then, in Section 3.2 and in Section 3.3, we describe their design and the execution of experiments. Finally, we provide an overview of each toolset.

3.1 Requirements

In this section, we describe the requirements we set for both our tools. The following define distinctive traits and the scope of our work.

3.1.1 Type of Data Collected

Metrics. We want to support in-field collection of several metrics typically used to perform connectivity and routing performance assessment.

TRIDENT provides as the key metric the *Packet Delivery Rate (PDR)*, i.e., the ratio of packets received over those sent. *PDR* provides a direct assessment of the ability of a link to reliably communicate packets. A number of metrics are extracted directly from the *radio chip*: Received Signal Strength (*RSSI*), Link Quality Indicator (*LQI*), and *RSSI noise floor*, sampled after sending or receiving a packet. These metrics provide insights about physical layer parameters influencing *PDR*, and their correlation with it. Moreover, *noise floor* is useful to indirectly determine the presence of interference.

HARPOON provides as key metrics: the *delivery rate*, i.e., the number of messages received by the *sink/root* node over the total number of messages generated by the *source* nodes, and the *duty-cycle*, i.e., the portion of time spent with the radio chip turned on. The *delivery rate* provides a direct assessment of the reliability of communication. The *duty-cycle* is a proxy for the energy consumed because the radio chip consumes far more power than the other hardware components. If the protocol under tests allows, we acquire additional parameters to assess the effectiveness of specific mechanisms: the number of *duplicates* dropped before reaching the routing layer, the number of *duplicates* received at the routing layer, the number of failed *link-layer acknowledgements* at every node, the

number of *forwarded* messages, the number of *acknowledgements* received, the number of times the sending queue was full. These parameters can be recorded for each node in the network.

Both tools support the acquisition of *environmental parameters* (e.g., temperature and humidity) from on-board sensors. These are useful to determine how the environment affects communication.

Aggregated vs. per-packet samples. The reason for which the connectivity or routing assessment is performed determines the nature of data gathered. If a long-term observation is necessary, the amount of data recorded can rapidly become prohibitive for a resource-constrained platform. Therefore, the tools should support the ability to store only an aggregate of the metrics collected, computed over a well-defined sequence of packet exchanges.

On the other hand, recording the individual metrics associated with *each* packet provides a fine-grained detail that is necessary in some cases, e.g., when the goal is to ascertain the time-variant characteristics of links with the resolution necessary to inform the design of network protocols or when the goal is to analyze snapshots of the logical topology constructed by the routing protocols.

The choice between aggregated or per-packet samples should therefore be left to the user, balancing the goals of the assessment against the storage limitations of the platform at hand.

In HARPOON the aggregation is done either per-network or per-node. The former, computed over all the nodes, may be a choice when the goal is to assess the overall behavior of the protocol in the target environment, while, the latter, providing a fine-grained detail is necessary when the goal is to ascertain the impact of each individual node on the connectivity graph or the energy consumption of forwarding nodes.

Single packets vs. bursts. Another related dimension is the way the channel is observed. Connectivity assessment is often performed by sending *probe* packets with an inter-message interval (IMI) relatively high (e.g., seconds), representative of several WSN applications. Nevertheless, some applications (e.g., recording data from accelerometers) require the transmission of bursts of packets. Moreover, a well-known property of wireless channel is that the transmission of packets sent with a small-enough IMI [69] is more reliable. The tool should allow the user to choose whether to use single packets or bursts of packet probes. TRIDENT supports both single packets and bursts while HARPOON only single packets.

3.1.2 Type of Experiments Supported

Interactive vs. batch. Connectivity and routing assessment may be needed for reasons yielding different requirements as described in Section 3.1.

If the ultimate goal is to support a WSN deployment by helping determine a node placement enabling communication, this is often achieved by performing tests of short duration (e.g., few minutes). These are useful to quickly evidence which nodes experience low *PDR* values or if there are any network partitions; this information is used by the operator to relocate nodes and reassess connectivity with another short test. To effectively support this process, the tool must provide a way to quickly represent the *PDR* associated with the WSN links.

On the other hand, connectivity and routing assessment is performed also to *characterize* the target environment. This requires long-term observations (e.g., days); the continuous presence of an operator would be impractical. The tool must provide the option to run *automatically* a battery of tests, including different settings, defined by the operator but executed without her involvement.

In our experience, the two methods of operation are often used in conjunction. Indeed, before starting a long-term batch experiment, a short-term interactive one is performed, to make sure that all nodes are functioning properly, and that the baseline connectivity is appropriate to the purpose of the experiment.

Mobile nodes. Mobile WSN applications, e.g., involving nodes placed on humans, animals, robots or vehicles, are becoming increasingly popular. Therefore, the tool should support experiments where some of the nodes are mobile, to assess the connectivity between these and the fixed nodes. An interesting possibility, partially explored in one of our group previously work [19], is to use mobile nodes as a way to perform a preliminary exploration of the deployment area. As the mobile node moves across the field and exchanges messages with fixed nodes, it samples the connectivity of a high number of locations, cumbersome to explore individually only with fixed nodes.

3.1.3 Support to Operators

In-field, wireless interaction with the nodes. In-field operators must interact with the nodes for various purposes. The primary reason is to retrieve the results of experiments, stored on the nodes taking part in them. Another key operation is the re-tasking of the nodes with a new set of experiments. The operator may also need to interact with the nodes for the sake of monitoring the correct execution of the experiment, e.g., to retrieve statistics about the experiments performed or the battery level. Other useful operations are the ability to put selected nodes (or the entire WSN) in stand-by when

they are not involved in an experiment, and wake them up later on. In principle, some of these operations can be performed by directly accessing the node; for instance, data can be downloaded via USB. However, while this operation is trivial in a lab, it becomes cumbersome when nodes are deployed in-field in a harsh environments, e.g., outdoor in winter, or in places that are not easily accessible. Therefore, all of the interactions with the nodes should be performed over-the-air, by leveraging the wireless channel.

Data storage and processing. Connectivity and routing assessment experiments may generate a huge quantity of data. Handling these as individual files becomes rapidly impractical. Further, the raw data gathered often needs to be processed in an automated way to simplify the interpretation. The tools should therefore integrate a database for storing experimental data, enabling structured access and querying, and the definition of stored procedures providing a layer of abstraction in data manipulation and interpretation.

3.1.4 Non-functional Requirements

No infrastructure. This is a defining non-functional requirement. In-field experiments cannot afford the luxury of a secondary network: the experiment execution must rely only on the WSN nodes under test.

No coding required. Our desire to support domain experts implies that using the tools should *not* require writing not even a single line of code. The configuration of experiments should occur entirely via the user interface; at most, domain experts must learn how to flash motes with a pre-canned binary before going in-field.

Ease of use. The logistics of in-field experiments makes them effort-demanding and time-consuming. The situation should not be exacerbated by a complex or cumbersome interaction with the tools. Simple configuration files or a graphical user interface, providing intuitive support for all the phases of the experiments, is therefore an obvious requirement.

Flexibility. The experiment settings, including number of nodes, their nature and role in the experiment, power and channel settings, number of messages, inter-message interval, number of test repetitions, etc., should be designed in a way that allows user to combine them freely, to explore different portions of the parameter space.

Decoupling from hardware platform. The tools should work on standard nodes without modifications of hardware. Nevertheless, as there are many WSN platforms available, supporting a new one in the tool should be simplified by its software architecture, by confining platform-specific details in well-identified components.

Decoupling from protocol. The tools should support an easy integration of a new routing protocol by confining protocol-specific instrumentation in well-defined components.

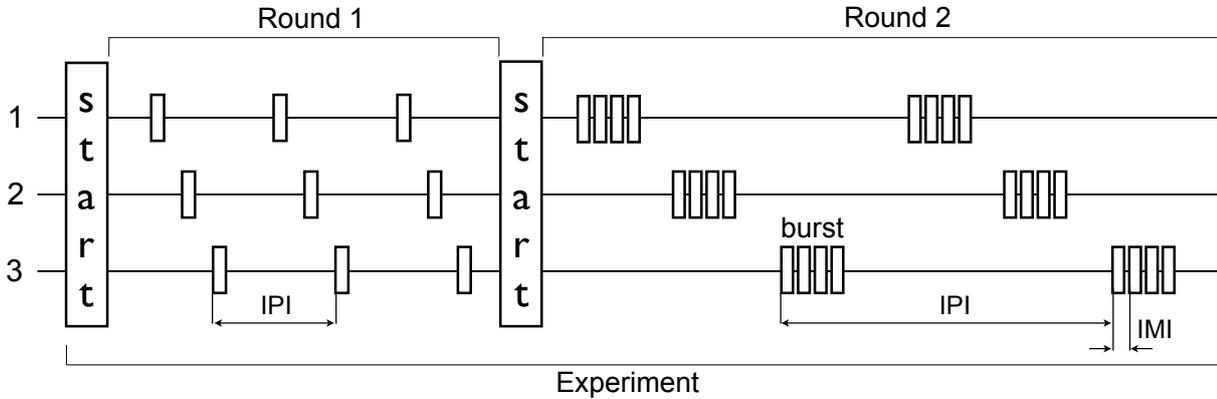


Figure 3.1: Sample TRIDENT experiment showing two rounds, staggered transmissions using single-packet and burst probes, and per-round synchronization.

3.2 Trident

TRIDENT is the expressly designed tool to simplify the chore of in-field connectivity assessment. The tool inherits all the requirements described in Section 3.1.

Next we present the design of TRIDENT. We begin with a description of the execution of connectivity assessment experiments, then provide an overview of the TRIDENT toolset.

3.2.1 Experiments Execution

Definitions. An entire TRIDENT experimental campaign is defined as a sequence of *rounds*, as shown in Figure 3.1. Each round has its own configuration parameters, detailed next, including whether it used single-packet probes or burst probes, i.e., multiple packets transmitted in rapid sequence. The time between the beginning of two consecutive probes from the same node is the *inter-probe interval (IPI)*. For burst probes, we also define the *inter-message interval (IMI)* as the time between two messages belonging to the same burst. Both IPI and IMI are configurable on a per-round basis.

Probing the links. Connectivity is assessed by probing the communication links with packet transmissions and evaluating the received packets and their properties. Therefore, TRIDENT experiments must define precisely *when* each node transmits and listens.

All nodes behave the same: transmit a probe, pause for the IPI duration, repeat this process a configurable number of times. In between transmissions, nodes can be configured to listen for packets from other nodes. TRIDENT does not duty-cycle the radio during the experiments, ensuring that no packets are lost due to the radio state. Moreover, to avoid collisions among simultaneously transmitted packets, which can confound the link evaluations, TRIDENT ensures that only one sender transmits at any given time.

This is achieved by having nodes begin their probe sequence in a staggered way based on their node identifiers. Specifically, the transmit time for the i -th probe of node n is defined as

$$t_{n,i} = t_{start} + nT_{IPI} + iT_{IPI}$$

where t_{start} is the start time of the round, T_{IPI} is the value of the IPI, n is the node identifier, and N is the overall number of nodes. n and N are setup statically during the experiment design phase.

Staggering transmissions by assigning each nodes its transmit slot, requires the nodes to be time-synchronized. In TRIDENT this is achieved at the beginning of each round, as shown in Figure 3.1. This synchronization allows the system to compensate for clock drift during a long running experimental campaign.

Master node. Time synchronization is initiated by a special node, called the *master*. The latter acts in general as a coordinator towards the rest of the WSN nodes, as well as the “access point”, enabling the operator to change the configuration of experiments. The master has the same binary code of the other nodes; its special role is determined by its identifier, $n = 0$.

The parameters describing the round configuration are also disseminated by the master node during synchronization at the beginning of each round. The option to change parameters in each round allows the interleaving of rounds with different power levels, or interleaving single-packet and bursty rounds, as shown in Figure 3.1.

This choice has multiple consequences. First, only the master node is aware of the experimental campaign, and therefore is the only one affected by changes to the latter. As the master can receive an entire experimental campaign configuration over-the-air, physical access to nodes is not required to change or initiate a campaign. Second, prior to starting a long-running campaign, the operators can interactively run a number of small experiments, each time uploading the round description to the master, instructing the master to initiate the round, then collecting the results. After analysis, another short experiment can be carried out, or the long-running experiment can be initiated. This is all possible because the nodes are experiment-agnostic and the master can be controlled over-the-air.

Per-round configuration parameters. Table 3.1 shows the per-round configuration parameters available in TRIDENT, communicated by the master before starting a round. We already mentioned some of them, e.g., the overall number of nodes, the role (sender or listener), the type of probe, the values of IPI and IMI. Additional parameters include the radio channel, transmission power, overall number of probes transmitted per node, and number of packets per burst. The logging method must also be defined, choosing between storing information about every packet, or only the average over the entire round,

Table 3.1: TRIDENT per-round configuration parameters.

Parameter
number of nodes (N)
list of senders
list of listeners
transmit power
probing channel
control channel
number of probes per sender
single-packet probe
inter-probe interval (IPI)
inter-message interval (IMI)
aggregate logging
number of repetitions

based on the needs of the experiment and the available storage. Finally, rounds and/or entire experiments can be repeated a configurable number of times, to increase statistical relevance.

As for metrics, not shown in the table, *PDR*, *RSSI*, and (if available) *LQI* are collected by default. If the platform allows, sender and receiver can acquire per-packet environmental conditions such as noise floor, temperature, and humidity. These values are recorded according to the per-round logging policy.

Interacting with the nodes under experiment. Changing the per-round configuration parameters is not the only option to interact in-field with nodes. Table 3.2 describes the commands available to the operator. As already mentioned, the operator can upload the description of an entire campaign configuration on the master, which then uses it to orchestrate the various rounds with the appropriate per-round configuration. However, the operator can also start and stop the execution of the experiment; these commands are propagated network-wide, with a mechanism similar to the one used to mark the start of a single round. Alternatively, the master node can be used to start and stop the experiments, e.g., using the on-board user button.

The master node can instruct all nodes to automatically enter a sleep state upon the end of an entire experimental campaign, allowing them to save their remaining battery power. In TinyOS, nodes in this state use the default low-power listening (LPL) MAC with a long sleep interval, currently set to 1 s. By duty-cycling the radio, nodes save battery power, but can be woken up later, e.g., to initiate over-the-air data download

Table 3.2: In-field commands.

Command	Target	Description
UPLOAD	master	load a campaign configuration
START	network	start the execution of an experiment
STOP	network	stop the execution of an experiment
POLL	1-hop	query battery level
SLEEP	1-hop	place nodes in low-power listening
WAKE-UP	1-hop	remove nodes from low-power listening
DOWNLOAD(n)	node	download logs from selected node
ERASE(n)	node	erase flash of selected node
SNIFF	operator	toggle packet sniffing

or to start a new experimental campaign. Alternately, the operator can also put to (or wake-up from) sleep a subset of the nodes, and query for their battery level.

Other commands enable the operator to download the experiment logs from a selected node, and to erase its flash memory after successful data transfer is verified. Finally, passive packet sniffing can be activated on the node managed by the operator, enabling the latter to ascertain whether all nodes properly started the experiment.

3.2.2 Toolset Overview

Figure 3.2 depicts the main structure of TRIDENT. WSN nodes, the subject of the experiment, are programmed with a *platform-specific mote-level runtime* that is experiment-agnostic; its behavior is established by the operator without requiring any coding.

This configuration is performed through the GUI of a dedicated component, the *experiment planner*, which resides on the operator's laptop. The experiment planner essentially enables the operator to quickly and easily define the various details of the experiment, by properly setting the values for the parameters in Table 3.1. This step does not need to be performed in-field, as the planner enables only the definition and storage of experiments.

The actual upload of experiments to the master, and from there to the rest of the

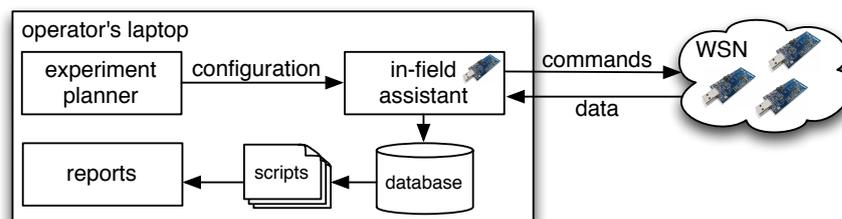


Figure 3.2: The TRIDENT toolset.

WSN, is instead supported by the *in-field assistant*, which also enables the execution of the other commands in Table 3.2. Communication with the WSN is enabled by a mote acting as a gateway, connected to the USB port of the operator’s laptop.

The in-field assistant provides also a simple visualization of the connectivity map built from available collected traces. An example is shown in Figure 3.3, visualizing the quality of the inbound links for node 0 according to a intuitive green/yellow/red color-coding, whose semantics in terms of *PDR* is configurable.

This feature is particularly useful when TRIDENT is used for a short-term assessment, as it quickly informs the operator about areas with connectivity problems, whose nodes can then be re-arranged. A similar visualization is provided also for mobile nodes; once the in-field assistant is fed with a sequence of locations, it can ”replay” the maps, showing to the operator how connectivity evolved due to mobility.

Finally, the *database* and associated plotting *scripts* simplify the storage of the collected data and its offline analysis. The database contains generic and customizable stored procedures for data manipulation. The set of pre-canned scripts allows the user to quickly plot trends derived from data collected, e.g., currently including network-wide *PDR*, spatial and temporal variations of the metrics, correlation of *PDR* with *RSSI* and *LQI*.

3.2.3 Platform-specific Details

TRIDENT currently supports two hardware platforms: TMote Sky and Wasp mote. The former directly integrates the CC2420 radio chip, while the latter relies on an extension module for radio communications, in our case the XBee S2 integrating the ZigBee-

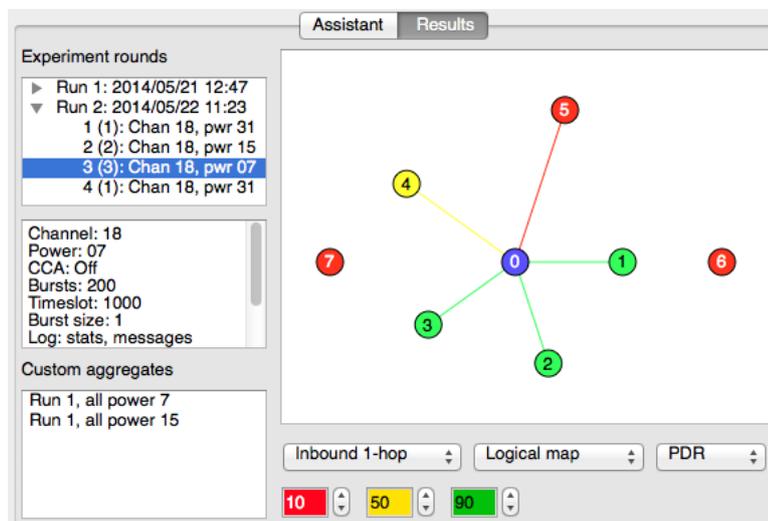


Figure 3.3: The “results view” of the TRIDENT in-field assistant, showing one-hop inbound connectivity and *PDR* for node 0.

compliant EM250 system-on-chip. Both transceivers implement the 2.4 GHz IEEE 802.15.4 physical layer.

The software architecture of TRIDENT confines the differences mostly in the platform-specific runtime installed on motes, although a few changes are needed also to parts of the in-field assistant handling communication with the WSN and parsing the logs for visualization. We refer to these platform-specific portions of the software as the *backend*, and summarize the differences in Table 3.3. The Wasp mote variant provides less features than the TMote Sky counterpart, as TinyOS allows low-level access to the radio chip while Wasp mote provides only the high-level interface of the ZigBee application support sublayer (APS). These trade-offs are discussed in the rest of the section, along with other implementation details.

Available metrics. The two platforms provide different metrics. TinyOS records *RSSI* and *LQI* for each received message, while the XBee radio module reports only *RSSI*. Moreover, unlike ZigBee, the low-level API available to TinyOS applications allows requesting *RSSI* when the channel is idle to measure the noise floor.

The temperature and humidity sensor of TMote Sky provides important information for our studies of the environmental effects on connectivity. In principle, the same holds for Wasp mote, given the wide range of sensors available for this platform. However, we have not yet implemented support for them in TRIDENT, although this does not pose any particular technical problem.

Experiment execution. On both platforms the experiment configuration is installed in the non-volatile storage of the master node. The TMote Sky backend supports uploading the configuration wirelessly or via USB, and stores it in a dedicated flash partition, while the Wasp mote backend relies on a configuration file on the SD card.

Table 3.3: Platform support in TRIDENT.

hardware	TMote Sky	Wasp mote + XBee
software	TinyOS	Arduino
PHY layer	802.15.4 (2.4 GHz)	802.15.4 (2.4 GHz)
radio chip	CC2420	EM250
TX power	-25 . . . 0 dBm	-8 . . . 2 dBm
metrics	RSSI, LQI, noise	RSSI
burst probes	yes	no
storage	flash chip, 1 MB	microSD, 2 GB
aggregate logging	per round, on motes	on operator's laptop
sensors	temperature, humidity	—

As described in Section 3.2.1, the network is time-synchronized at the beginning of each round to avoid collisions among probes. The backends implement different techniques. In the case of TMote Sky, dissemination relies on a TDMA scheme where each node has its own time slots to repropagate commands, in a way similar to the mechanism outlined for probe transmission in Section 3.2.1. The common time reference needed for both the TDMA dissemination phase and to calculate t_{start} is established with TinyOS packet-level time synchronization service [50], yielding sub-millisecond precision.

As ZigBee does not provide such a synchronization service, we rely on the standard multi-hop broadcast feature, basically a network flooding. However, based on the ZigBee Pro feature set [89], broadcast packets are *always* sent 3 times in a row, increasing reliability at the expense of energy consumption. These broadcast packets are separated by a 500 ms interval plus a random delay between 0 and 40 ms. Therefore, in the worst case where the start synchronization message is received only upon third attempt, the time synchronization error goes slightly above 1 s per hop.

To secure a collision-free transmission schedule the IPI should be set long enough to compensate for these synchronization errors and also for the time drift of the nodes. On TMote Sky the synchronization error is negligible; the typical time drift of 100 ppm results in two nodes drifting apart by 36 ms in half an hour. Therefore the use of 250 ms time slots during 30-minute rounds can be considered safe, counting also the time needed for internal processing of the received packets. Instead, on Wasmote the second-per-hop error should be compensated; we achieve this by using 3 s time slots.

Moreover, transmission of probes as one-hop broadcast requires an additional second, again due to the triple transmission performed by the ZigBee network layer. Therefore, it is impossible to send bursts of packets with Wasmote; on TMote Sky, bursts are instead supported with a configurable IMI, set to 20 ms by default.

Another implication of ZigBee compliance is that nodes must join the wireless PAN (personal area network) before sending application data. A multi-hop ZigBee network is built around its coordinator with the standard join procedure, including channel scanning and handshaking, one hop at a time. This process requires up to minutes, depending on the network diameter. In case the channel is changed in between rounds, this affects the minimum interval between them, as the network topology must be rebuilt from scratch.

Determining the link-level PDR. In principle, the value of PDR can be obtained straightforwardly, since the number of transmitted and received packets is known for each link. However, Wasmote introduce additional complexity due to the triple transmission of broadcast packets mandated by the ZigBee network layer. As shown in Figure 3.4, since duplicates are automatically filtered at the receiver, the link-level PDR cannot be determined by simply counting the delivered packets at the application layer. Consider

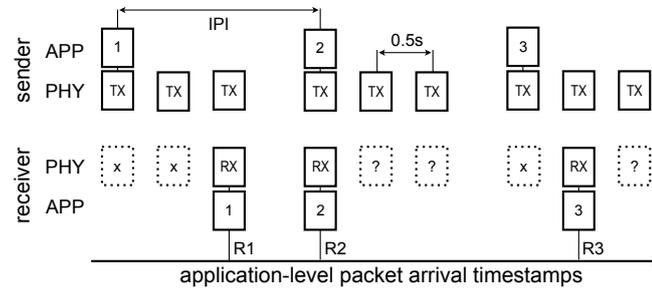


Figure 3.4: ZigBee transmits each broadcast packet three times.

two hypothetical experiment runs, one where each probe sent is always received upon first attempt, and one where it is received always upon the third attempt. The link-level *PDR* would be a meager 33% in the second case, yet the application-level *PDR* (the only directly measurable) would be 100% in both cases, providing a false representation of the quality of the wireless link.

Nevertheless, we can infer the number of failed attempts by looking at the packet arrival time, based on the fact that the three broadcast transmissions in ZigBee are spaced relatively far apart (500 ms). It is therefore possible to determine, upon receiving a broadcast packet, whether this was the first, second, or third transmission. We confirmed this experimentally: Figure 3.5 shows the distribution of the packet arrival interval (modulo the nominal IPI) measured at the application level for an intermediate-quality link. For a perfect link, all packets would be received with the same IPI, 15 s in this case; however, this is not the case when packets are lost. Consider an application-level packet i , received on first attempt. If the previous packet, $i - 1$, was received only on second or third attempt, the IPI between the two packets is smaller than 15 s. A similar reasoning holds in case the next packet $i + 1$ is not received upon first attempt, yielding an IPI greater than 15 s. Clearly, the histograms to the left and right of the central one in Figure 3.5 can be generated also by intermediate combinations, e.g., if i is received upon second attempt and $i + 1$ upon third, the IPI will be around 14.5 s.

Packet loss can be inferred by comparing the application-level packet arrival times-

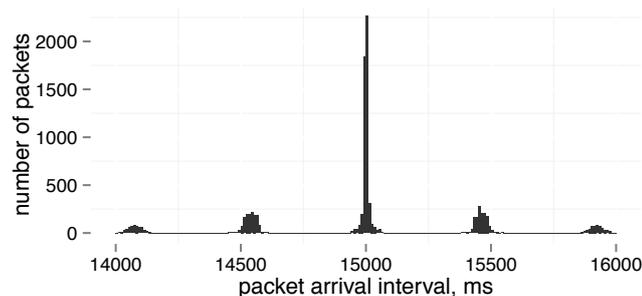


Figure 3.5: Distribution of packet arrival time.

tamps (e.g., R_1 and R_2 in Figure 3.4), provided there is at least one packet in the received sequence known to have arrived upon first attempt. This can be stated certainly when at least one pair of packets (not necessarily consecutive) has either the minimum or maximum possible recorded arrival interval (modulo the IPI), i.e., placed leftmost or rightmost on the histogram of Figure 3.5. Indeed, this means that the one of the packets was delivered on first attempt and the other on last, as in the case of R_1 and R_2 .

In principle, it may happen that no such IPI is recorded during the whole test. In practice, this is unlikely to happen for intermediate-quality links. However, one can still infer the characteristics of the link based on the application-level *PDR*. If the latter is very good, one can assume that the majority of the packets were received on the first attempt and base the analysis on this fact. For very bad links, it may be impossible to measure the actual *PDR* precisely when there are just few packets received from the whole sequence. **Storing the experiment data.** Due to the storage limitations of TMote Sky, per-probe logging can be replaced by storing per-round averages of the recorded values, computed on the nodes themselves. Waspote does not have strict storage capacity limitation; full logs are always stored and the log analysis performed on the operator’s laptop. However, we plan to implement on-board log processing also on Waspote, to reduce the downloading time of large logs.

3.3 Harpoon

HARPOON is the expressly designed tool to simplify the chore of in-field routing performance assessment. Currently, HARPOON supports two routing protocols: *CTP* (Collection Tree Protocol) [33], arguably the de-facto protocol for data collection applications, and *ORW* (Opportunistic Routing Protocol) [41], a more recent opportunistic approach and competitor. HARPOON collects common metrics and parameters for both protocols, as specified in Section 3.1.1 and a set of protocol-specific ones. For *CTP*, we acquire the *parent in the tree* at every node at the time of sending a message, along with the total number of *parent changes* observed throughout the experiment. For *ORW*, we acquire the *first neighbor*, the number of *dummy messages* and the number of *neighbors* for each node.

Similarly to TRIDENT, the tool supports interactive vs batch experiments and supporting a new routing protocol in the tool is simplified by its software architecture. The set of options to interact in-field with the nodes is much reduced in comparison with TRIDENT, changing the per-round configuration parameters and downloading the data traces are the only options available to the operator.

Next we describe the design of HARPOON. First we give a description of the execution

of routing experiments, then provide an overview of the HARPOON toolset.

3.3.1 Experiments Execution

As in TRIDENT, an entire experimental campaign with HARPOON is defined as a set of *rounds*, each with its own configuration parameters. Each round has its own configuration parameters, detailed next. The time between two consecutive messages from the same *source* node is the *inter-probe interval (IPI)*.

Probing the network. Routing performance is assessed by having a set of nodes acting as *sources* sending messages towards a *sink* node. We instrumented the routing protocols implementation to provide our test application with the necessary hooks to gather the statistics and parameters mentioned in Section 3.1.1. We embed this information within the application messages and report them to the sink. Unlike the connectivity assessment, transmissions are not centrally scheduled, the generation of messages being randomly scattered over time, as in most data collection applications [49]. During the experiment, HARPOON duty cycles the radio of the source nodes. We adopt the *CTP* default setting of having the sink node always on.

Sink node. The parameters describing the round configuration are disseminated by the *sink* node at the beginning of each round. This node is the equivalent of the *master* node from TRIDENT. The sink receives an entire experimental campaign configuration via USB connection from operator’s laptop and dissemination to the WSN nodes is done using broadcast communication. This way, physical access to the nodes is not required in order to change or initiate an experimental campaign. Not only that it acts like a coordinator of the network, but the sink node also acts as an ”aggregator”, the data traces being stored in its local memory.

Per-round configuration parameters. Table 3.4 shows the per-round configuration parameters available in HARPOON, communicated by the sink before starting a round. These are: the overall number of nodes (N), the transmission power and radio channel, inter-probe interval (IPI), round duration (RD), wake-up interval (WI).

Table 3.4: HARPOON per-round configuration parameters.

Parameter
of nodes (N)
transmission power
probing channel
inter-probe interval (IPI)
round duration (RD)
wake-up interval (WI)

the inter-probe interval (*IPI*), the round duration (*RD*) and the wake-up interval (*WI*). Rounds and/or entire experiments can be repeated a configurable number of times, to increase the statistical relevance.

As for metrics, not shown in the table, *delivery rate*, *duty-cycle* and parameters mentioned in Section 3.1.1 are collected by default. If the hardware platform allows, source nodes can acquire per-packet environmental conditions and battery level. These values are recorded according to per-packet or per-node logging policy.

3.3.2 Toolset Overview

Figure 3.6 shows the main structure of HARPOON. WSN nodes, the subject of the experiment, are programmed with HARPOON application, the main component of the toolset. When compiled for the sink, a node whose role is determined by its identifier, $n = 0$, it adds an extra behavior than for a normal node, to broadcast the round configuration parameters and to store the messages received from the other nodes of the network. The configuration is performed through a file in which the operator can quickly and easily define the experiment, by properly setting the values for the parameters in Table 3.4. This operation does not need to be performed in-field.

The actual upload of the experiment to the sink is done via a USB connection and from there to the rest of the WSN nodes. The sink node is used to start and stop the experiments using the on-board user button.

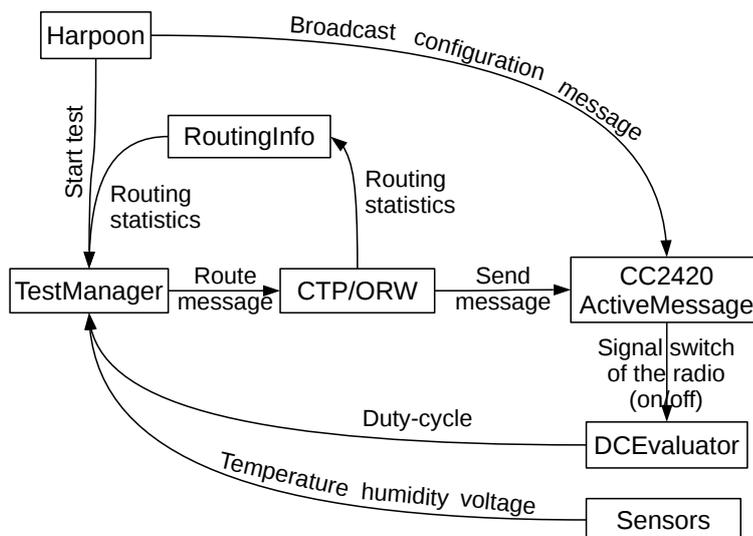


Figure 3.6: The HARPOON toolset.

TestManager is the component responsible for rounds management. At node level, the component generates messages with a pre-defined application period, IPI, for the length of the round duration. Moreover, the component is responsible for gathering routing statistics, the duty cycle metric and environmental data from specialized components and embedding them in specific fields of message to be forwarded to the routing layer. When the round duration expires, *TestManager* stops and passes the flow back to HARPOON.

The *RoutingInfo* component collects metrics, common and specific parameters from the routing protocols and provides an interface to HARPOON to retrieve these information. As mentioned before, *CTP* and *ORW* are the two protocols supported by our tool. Thus, we instrumented their implementation to provide HARPOON with the necessary hooks to gather the metrics and parameters.

The *duty-cycle* is collected and computed using the *DCEvaluator* component. For this, HARPOON reuses the duty cycle measurement component provided with the *ORW* implementation [42]. Environmental conditions and battery voltage are collected using the *Sensors* component.

Finally, a set of pre-canned *scripts* allows the user to quickly parse the logs and generate an output results file for offline analysis, which contains a summary of per-round configuration parameters and the per-node and per-network collected metrics and statistics. The set of plotting scripts allows the user to quickly plot trends derived from data collected; e.g., currently including network *delivery rate* and average network *duty-cycle* in time, individual nodes *duty-cycle* in time, network and individual nodes retransmissions in time.

As a final note, it is worth noting that, to integrate *ORW* protocol into our tool, we solved two of its major bugs. First, we had to patch *ORW* to accept broadcast messages and therefore to accept the configuration message to ensure the same starting procedure for all tested protocols. Second, an *ORW* node stops sending data after 255 packets have been sent if the receiver and sender are only one-hop away. Therefore, we had to patch *ORW* in order to support experiments with nodes sending more than 255 packets.

3.4 Summary

In this chapter we presented TRIDENT and HARPOON, tools that enable principled, repeatable, automated and flexible connectivity and routing protocol performance measurements, among low-power wireless devices immersed in real-world environments. Unlike similar tools in the literature, TRIDENT and HARPOON do not require any communication infrastructure besides the WSN nodes. Moreover, they are designed to be easy to use for domain experts, which can perform their experiments without coding. Our tools support

several configuration parameters and metrics, enabling the investigation of many aspects of low-power wireless communication and routing protocols.

During the years we continuously modify our tools based on lessons we learn from in-field experience in our deployments and experimental campaigns. TRIDENT has already been used in-field by domain experts (e.g., biologists) without technical knowledge about WSNs in the Ecuador forest [19].

The experimental campaigns we describe in the next chapter were all performed using TRIDENT and HARPOON.

Chapter 4

Understanding: Data Traces

What if Christmas, does not come from a store?

What if Christmas...perhaps...means a little bit more.

Dr. Seuss, *How the Grinch Stole Christmas!*

The performance of a WSN deployment is obviously directly affected by the quality of the links enabling communication among nodes. In the case of commonly-used IEEE 802.15.4 radios, the link quality is in turn easily affected by a variety of environmental factors, which cause variations over time and space. This is a well-known fact that has been studied extensively in the context of protocol design for WSNs. For instance, a number of techniques exist for estimating the quality of links and determining their best use in MAC and routing protocols [6, 26]. These techniques are themselves based on the findings of studies that empirically determined key properties of low-power wireless communication, as we show in Section 2.3.

The goal of supporting the efficient design of network protocols has led the research community to focus primarily on the study of *small-scale*, short-term variations of the individual links, as this is what matters for determining, say, forwarding decisions in a routing protocol. Further, the need to analyze in detail these variations has led to experimental setups constituted mostly by highly controlled lab settings. Nevertheless, the few exceptions in the literature [13, 19, 24, 53, 78, 82] show that real-world environments may yield quite different link properties.

In this dissertation, we take a different look at the problem of assessing link quality, motivated by deployment and application issues rather than protocol design. Specifically:

1. we focus on the *large-scale* changes of the overall network, as induced by different environmental conditions. Therefore, we characterize the *aggregate* constituted by the entire network, rather than the individual links.

2. we assess the above *in vivo* (i.e., in an actual real-world environment) in contrast to the *in vitro* (i.e., in labs or controlled setups) experiences reported in the literature.

The original motivation for the work described here stems from the wildlife monitoring application presented in Chapter 1. As already mentioned, to be useful to biologists, the interactions recorded by our WSN must be correlated to the characteristics of the environment. We were asking ourselves how do the various environmental factors affect the quality of communication? A similar question recently surfaced for GPS-based tracking [16], where it has been shown that the environment significantly affects the precision of GPS data, and post-processing is necessary to avoid biasing the scientific observations. In our application, the answer to this question affects directly the reliability of sensing interactions among animals. However, the answer is of more general interest, given the many applications exploiting a WSN in an outdoor environment. In this context, knowledge about the effect of the environment on communication provides insights about the reliability and lifetime of the WSN.

This chapter reports about the experimental campaigns through which we investigated these issues. Their design was informed by the biologists in our team, interested in determining how low-power wireless links are affected by:

- *Presence and density of vegetation*: we selected habitat components of a heterogeneous landscape (e.g., mountain range of roe deer): two types of forest, beech (dense vegetation, deciduous) and spruce (sparser vegetation, evergreen), as well as open field.
- *Seasonal variations*: experiments in summer and winter.
- *Daily variations*: experiments span the 24 hours.

The above entail in-field experiments, notoriously effort-demanding and time-consuming. Chapter 3 describes TRIDENT and HARPOON, the tools we used in our in-field campaigns, expressly designed to simplify the in-field assessment of connectivity and protocol performance.

In a nutshell, our experiments investigate what happens if the *same* WSN is “immersed” in different combinations of the above environmental factors. The experiment execution is the same across the various test environments. We describe the details of our experimental setup in Section 4.1.

The questions we answer are, for instance: What changes between operating a given WSN in an open field vs. in a forest, during winter vs. summer, or during day vs. night? Some qualitative answers can be derived from existing results, especially if one focuses separately on the impact of each environmental parameter (e.g., foliage, temperature, humidity, snow). Nevertheless, our aim in this chapter is to *quantify* the extent of changes, based on the combined effect of the various parameters in different, real outdoor environ-

ments. To this end, we rely on the metric of *packet delivery rate* (PDR), i.e., the ratio of packets received on a link over those sent. However, we aggregate and analyze the PDR in different ways and against several combinations of environmental factors to elicit our findings. Nevertheless, we also consider the *received signal strength* ($RSSI$) and the *link quality indicator* (LQI) as a number of approaches to evaluate the quality of network links rely on these two physical layer parameters. These are reported in Section 4.2, where we discuss the impact of the environment on the physical layer. In Section 4.3 we show how this impact is mirrored at the routing layer, for two routing protocols, namely CTP and ORW . Finally, in Section 4.4 we show how the impact is mirrored at the application layer—the one directly relevant to the end users using our application.

4.1 Experimental Setup

4.1.1 Time and Location of Experiments

Findings reported in this chapter were gathered in six experimental campaigns, described in Table 4.1. The first and second campaigns including only physical layer (PHY) experiments, were performed during the winter and late spring of 2011. A third, fourth and fifth campaign was performed in the summer of 2012 and winter of 2013, to validate the findings at PHY in a different time period, and explore the impact on the routing (RTN) and application (APP) layer. The sixth campaign was performed during the summer of 2014, to explore the impact of the environment on two different routing protocols, namely CTP and ORW . As the weather environment conditions in late spring and summer campaigns were very similar, hereafter we refer to both as “summer”, for simplicity.

The location of the experiments was chosen to be representative of the environment where our target WSN application is going to be deployed, and to cover different conditions of vegetation. All experimental sites are on Mount Bondone, near Trento, Italy. Upon suggestion of the biologists on our team, we identified three locations in this area:

Table 4.1: Campaigns.

Nr.	Season	Time	Environment	Topology	Layer
1	Winter	February 21-March 27, 2011	OPEN, SPRUCE, BEECH	CROSS	PHY
2	Summer	May 24-May 31, 2011	OPEN, SPRUCE, BEECH	CROSS	PHY
3	Summer	July 11-August 4, 2012	OPEN, SPRUCE, BEECH	CROSS_EXTENDED	APP
4	Summer	July 9-August 9, 2012	OPEN, SPRUCE	GRID	PHY, RTN
5	Winter	March 1-April 18, 2013	OPEN, SPRUCE, BEECH	GRID	PHY, RTN
6	Summer	August 15-September 18, 2014	OPEN	GRID_SMALL	RTN



Figure 4.1: Experimental sites on Mount Bondone.

OPEN is a wide meadow with essentially no trees, while SPRUCE and BEECH are forests characterized by the corresponding tree type. SPRUCE is an evergreen forest with sparse vegetation while BEECH is a deciduous forest with dense vegetation. Pictures from our deployment sites, in winter and summer, are shown in Figure 4.1. Due to some logistical issues, we performed the sixth campaign on an OPEN field on a different site, in Pergine, Italy.

4.1.2 Hardware Platform

TMote Sky [57] nodes, equipped with the ChipCon 2420 radio chip compliant with IEEE 802.15.4, and on-board omnidirectional antenna, were used for the PHY and RTN experiments. These are popular choices that allowed us to reuse experience from previous uses of TRIDENT in our studies [19] and to compare with experiments reported in the literature. Instead, for the APP experiments, the hardware used is custom-made for our application, integrating a GPS unit and a GSM/GPRS modem [56]. However, at its core is similar to TMote Sky, from which it differs in two respects. First, we use the TI MSP430F2618 MCU instead of MSP430F1611. The former provides a larger program memory (116 kB vs. 48 kB) and is better suited to our software architecture, which must manage many hardware components and their complex application functionality. Second, it uses a 2-Mbit FRAM (Ferromagnetic RAM) memory chip instead of the commonly used Flash memory. FRAM consumes less power than Flash, and offers faster write access and higher limits on write-erase cycles. As in TMote Sky nodes, our node supports low-power wireless communication via a CC2420 transceiver and on-board inverted-F microstrip antenna.

Across all campaigns, each node was placed in an IP65 water-proof box with a transparent cover, containing the D-size Lithium batteries powering the node, as shown in Figure 4.2(a).

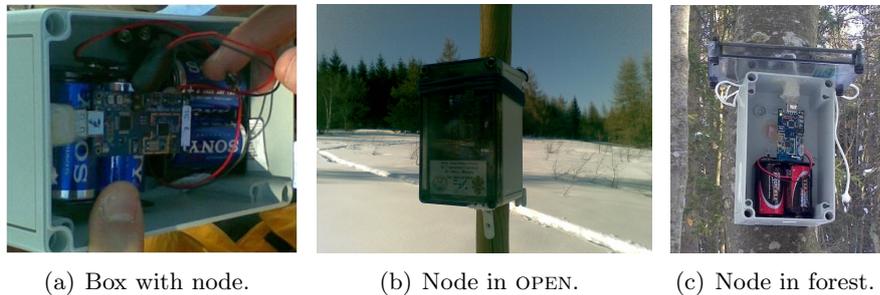


Figure 4.2: WSN nodes in the field.

4.1.3 Node Placement

The main topologies of our WSN deployments, shown in Figure 4.3, consist of 8 nodes arranged in a CROSS and 16 nodes distributed in a GRID. The CROSS, similar to [19], represents a good trade-off between deployment effort and coverage of different link distances, in our case ranging from 7 m to 64 m. The GRID yields a rich set of links at distances varying from 10.2 m to 60 m, and since the grid contains twice the nodes of the CROSS, as shown in Figure 4.4, it has more statistical relevance. It is worth noting that since the CROSS was our initial choice, the analysis presented in Section 4.2 is done using the CROSS and then we use the GRID to confirm the trends.

In OPEN, nodes were attached to 3-meter tall wooden poles, planted vertically in the ground. In the two forests, nodes were lashed to trees. Boxes were latched onto the poles and trunks of the trees with elastic ropes at 1.7 m from the ground, as depicted in Figure 4.2(b) and Figure 4.2(c). We selected the three sites to have a similar terrain inclination. Care was taken to ensure the same vertical mounting (i.e., the same antenna orientation) for all nodes while placing them in the boxes.

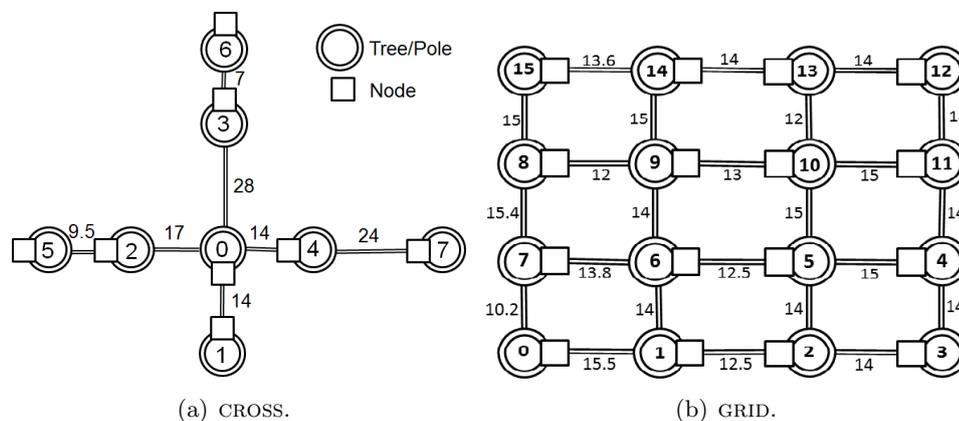


Figure 4.3: Network topologies for experiments.

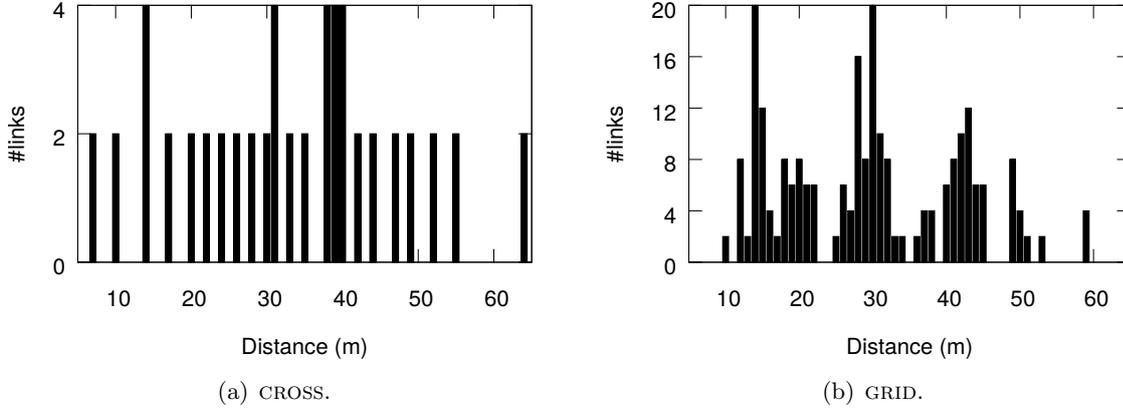


Figure 4.4: Distribution of link distances.

4.2 Physical Layer

The objective of the experiments described here is to understand if and how the environmental conditions, both in terms of morphology (e.g., vegetation present) as well as daily, seasonal and yearly variations, affect the low-power wireless links of a WSN, from the standpoint of the physical layer. The experiments we report about were performed during our first and second campaign using the CROSS topology. In Section 4.2.9, we report about experiments performed one year later, during campaign four and five, with a different topology, namely the GRID.

4.2.1 Experiment Execution

The in-field assessment of connectivity is performed using TRIDENT. Our experiment is composed of 30 minute rounds, in which each node of the network sends 215 packets with IPI equal to 8 s, at a rate of 1 packet/s. All nodes were configured both as listeners and receivers. No MAC protocol was used, given our goal of characterizing physical connectivity. Table 4.2 summarizes the per-round configuration parameters we used in TRIDENT for our CROSS campaigns.

Each node aggregated PDR , $RSSI$, LQI and $noise$ floor values. Moreover, nodes acquired temperature and humidity from the SensirionSHT11 [3] on-board sensors, so as to accurately monitor the conditions experienced by the network. Rounds were executed one after another without intervention for 2 days, after which they were stopped and data traces downloaded from the flash memory of the nodes. Moreover, we interleaved rounds at -1 dBm (power 27, hereafter “high power”) with rounds at -8 dBm (power 14, hereafter “low power”). We used channel 18 in all cases. Overall, the results of 96 rounds of 30 minutes were collected in each environment (i.e., OPEN, SPRUCE, BEECH), for each season (i.e., winter, summer).

Table 4.2: Values for per-round configuration parameters for PHY experiments using the CROSS.

Parameter	Value	Parameter	Value
# of nodes (N)	8	# of probes per sender	215
list of senders	0...7	single-packet probe	yes
list of listeners	0...7	inter-probe interval	8 s
transmit power	27(-1 dBm), 14(-8 dBm)	inter-message interval	N/A
probing channel	18	aggregate logging	yes

4.2.2 Network-wide Packet Delivery Rate

We begin our analysis with the roughest indicator of link quality, and yet the most intuitive and directly informative: the *packet delivery rate* PDR (i.e., the ratio of packets received over those sent) computed over the entire set of links. Despite its simplicity, this provides an immediate and easy-to-compute macro-indicator telling us how the same network behaves, once immersed in different environments.

Figure 4.5 shows the results of our experiments with the CROSS topology during the first and second campaign. A few trends are clearly identifiable. First, the quality of communication decreases as we progress from OPEN to SPRUCE to BEECH- i.e., as the quantity of trees and foliage increases. The trend is more marked during the summer at both powers: going from OPEN to SPRUCE, and from SPRUCE to BEECH shows differences in PDR of the order of 15-20%. Second, the seasonal variation (summer vs. winter) also includes dramatic changes in link quality. These can be as high as 30%, as in the case of SPRUCE at high power, but in any case winter is always consistently worse. Third, the two observations above, combined, yield an interesting outcome, namely, during winter the differences in PDR between the two forests, SPRUCE and BEECH, are negligible. As we discuss later, this is not true for other parameters related to link quality. However, through the macroscopic lens provided by the network-wide average PDR , it appears like

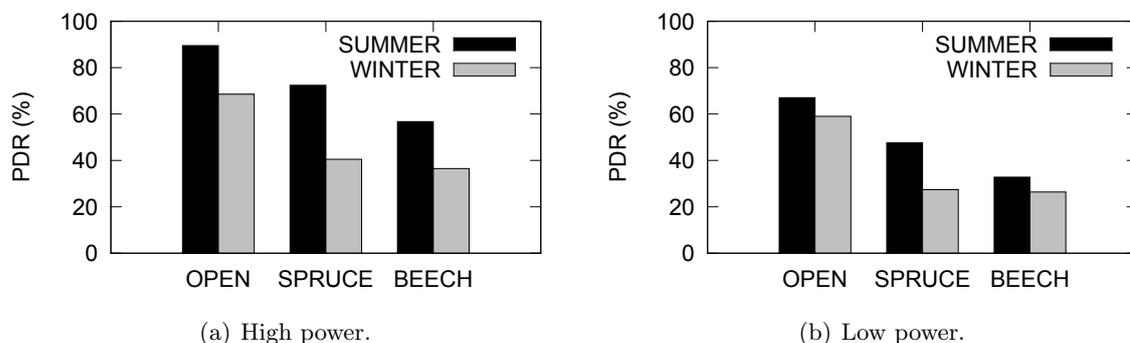


Figure 4.5: Network-wide average PDR , for all combinations of site (OPEN, SPRUCE, BEECH), season (summer vs. winter) and power (high vs. low).

the combination of snow and vegetation yields the same effect regardless of the density of the latter. In this respects, it should be pointed out that, although during winter the foliage is not present in BEECH (deciduous) while it is in SPRUCE (evergreen), the density of trees and branches is still much higher in the former. Finally, all of the trends above hold both for high and low power although, quite obviously, the *PDR* values are lower in the latter case.

We dissect further the environment impact on the PHY layer, with tools more sophisticated than the average *PDR*.

4.2.3 Transitional Region

Several studies [29, 87, 69, 91] have classified low-power wireless links in three distinct reception regions—connected, transitional, and disconnected—based on the distance of the receiver from the sender and on the *PDR*. The connected region is the closest to the sender: its links are of good quality, and often stable and symmetric. At the other extreme, the disconnected region is the farthest from the sender: it does not contain links practically usable for communication. The transitional region (also referred to as “gray area”) is a mix of the two, and contains links that exhibit a high variance in packet reception rate, as well as asymmetric links. Unfortunately, the transitional region often spans a large fraction of the communication range. The beginning of the transitional region and its span determine how challenging it is to ensure reliable communication in a WSN deployment.

Figure 4.6 shows the transitional areas for OPEN, SPRUCE, and BEECH, in different seasons and with different power settings, as described in Section 4.2.1. Several observations can be made.

OPEN provides the situation most favorable to communication. The transitional area begins (i.e., the connected area ends) much farther than in the forest deployments, for all combinations of seasons and power settings. The presence of vegetation appears to impair communication, by reducing the span of the connected area, and increasing the transitional one.

Seasonal variations are also evident. Indeed, the presence of snow on the ground and on tree branches during winter has a detrimental effect to communication. In OPEN, the transitional area begins closer to the sender at both powers, although the phenomenon is more marked when using low power. Similar considerations hold for the two forests, although the impact on communication of seasonal variations is less dramatic w.r.t. the one of vegetation. In comparison to OPEN, however, the transitional area not only starts earlier, but is also wider.

The two forests are similar in terms of beginning and span of the transitional region.

Differences are more marked in summer, when the denser and broader leaves in BEECH yield a smaller connected region, and a wider transitional one.

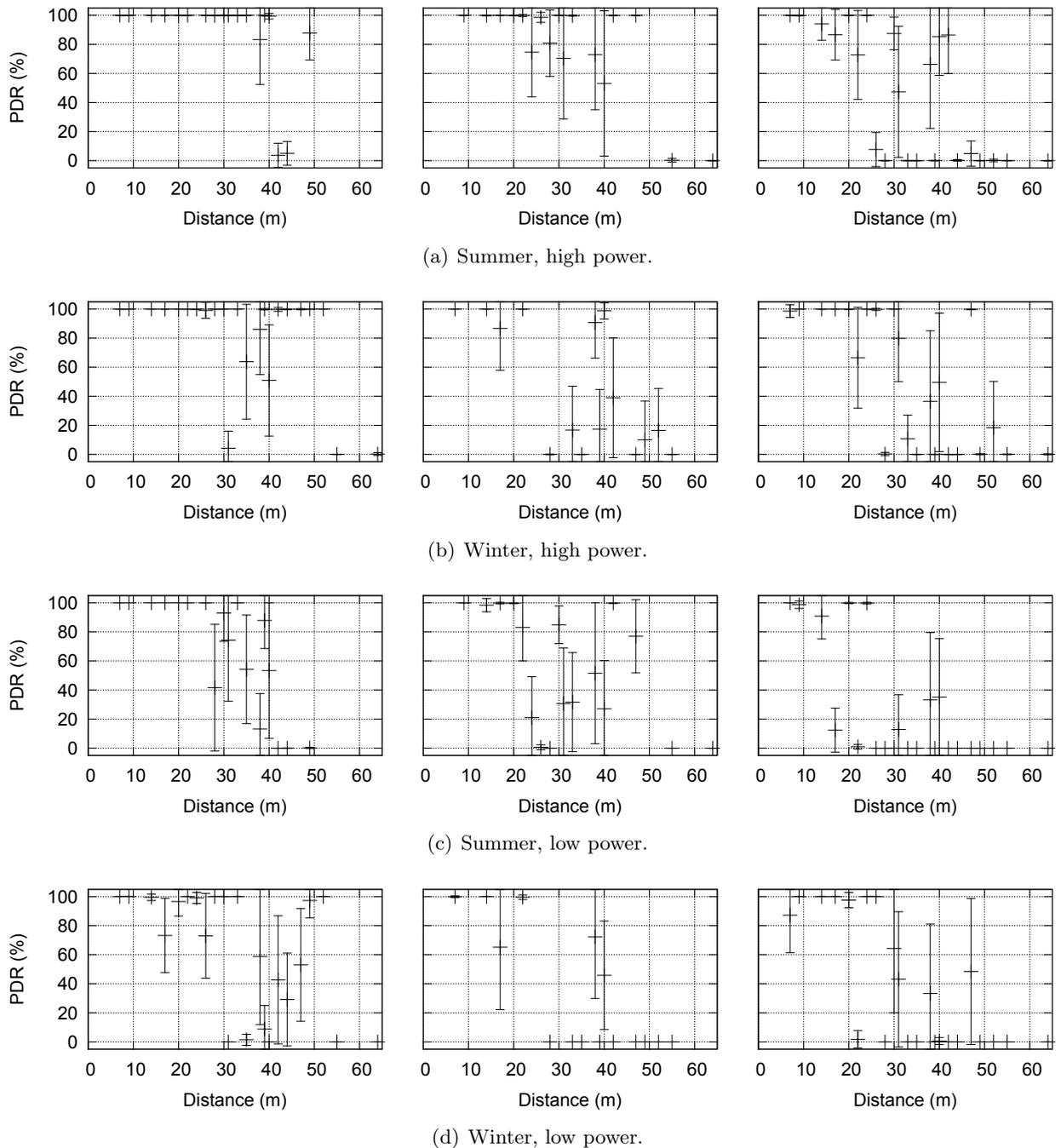


Figure 4.6: Transitional areas in OPEN (left), SPRUCE (center), and BEECH (right).

4.2.4 Received Signal Strength Indicator and Link Quality Indicator

Even if *PDR* is considered the most direct metric for link quality, physical layer parameters, like *RSSI* and *LQI*, are a direct measurement of the wireless channel quality when a packet is received and therefore are used to compute receiver-side link quality estimators. Thus, one should expect some level of correlation between physical information and link quality.

To explore this we aggregated all the packet traces collected on the CROSS and plot the average and standard deviation of *PDR* values with the same *RSSI* and *LQI* for each environment, season, and power.

A simple visual inspection of Figure 4.7 shows that we can identify two thresholds: below the lower threshold is the disconnected area, with *PDR* consistently low and less than 10%, while above the higher threshold is the connected area, with the *PDR* consistently high and above 90%. In between the two thresholds there is a gray area of many different *PDR*, with no clear correlation with *RSSI*. Based on this, there are a couple of observations we can make. Independent of the environment and season, the threshold for receiving packets is approximately -93 dBm, which is very close to the sensitivity threshold of the CC2420 radio, according to the data sheet. On the other hand, the higher threshold, is dependent on the environment. One can see that for OPEN, both in summer and winter, this is within a 2 dBm range of -85 to -83 dBm while for SPRUCE is within a 3 dBm range of -83 to -80 dBm and for BEECH is within a 6 dBm range of -83 to -77 dBm. Moreover, the same behavior as in the case of *PDR* transitional region can be observed. Thus, during winter the two forests look comparable having the same *RSSI* thresholds for *PDR* to become stable and $\geq 90\%$, around -80 dBm. During summer the difference between the two forests is more marked, the presence of leaves in BEECH has a detrimental effect on the *RSSI* threshold for achieving a consistently high *PDR*, which goes as high as -77 dBm, while in SPRUCE is around the value of -80 dBm. One can see that the distance between the two thresholds is wider as we progress from OPEN to SPRUCE and BEECH. In OPEN the *PDR* goes from 0% to 100% within 10 dBm while in the forests within 15 to 20 dBm.

Similar considerations hold for *LQI* vs. *PDR*. In Figure 4.8, one can easily see that winter “flattens” the *LQI-PDR* curve. The presence of snow on the ground and on the branches during winter increases the standard deviation. One can easily identify the presence of two thresholds and see that the distance between the two increases as we progress from OPEN to SPRUCE and BEECH. Moreover, in OPEN the average standard deviation of *PDR* is smaller than in the two forests, suggesting a better correlation of *LQI* and *PDR*. From this perspective, the two forests look comparable during winter, with BEECH having a higher variance than SPRUCE.

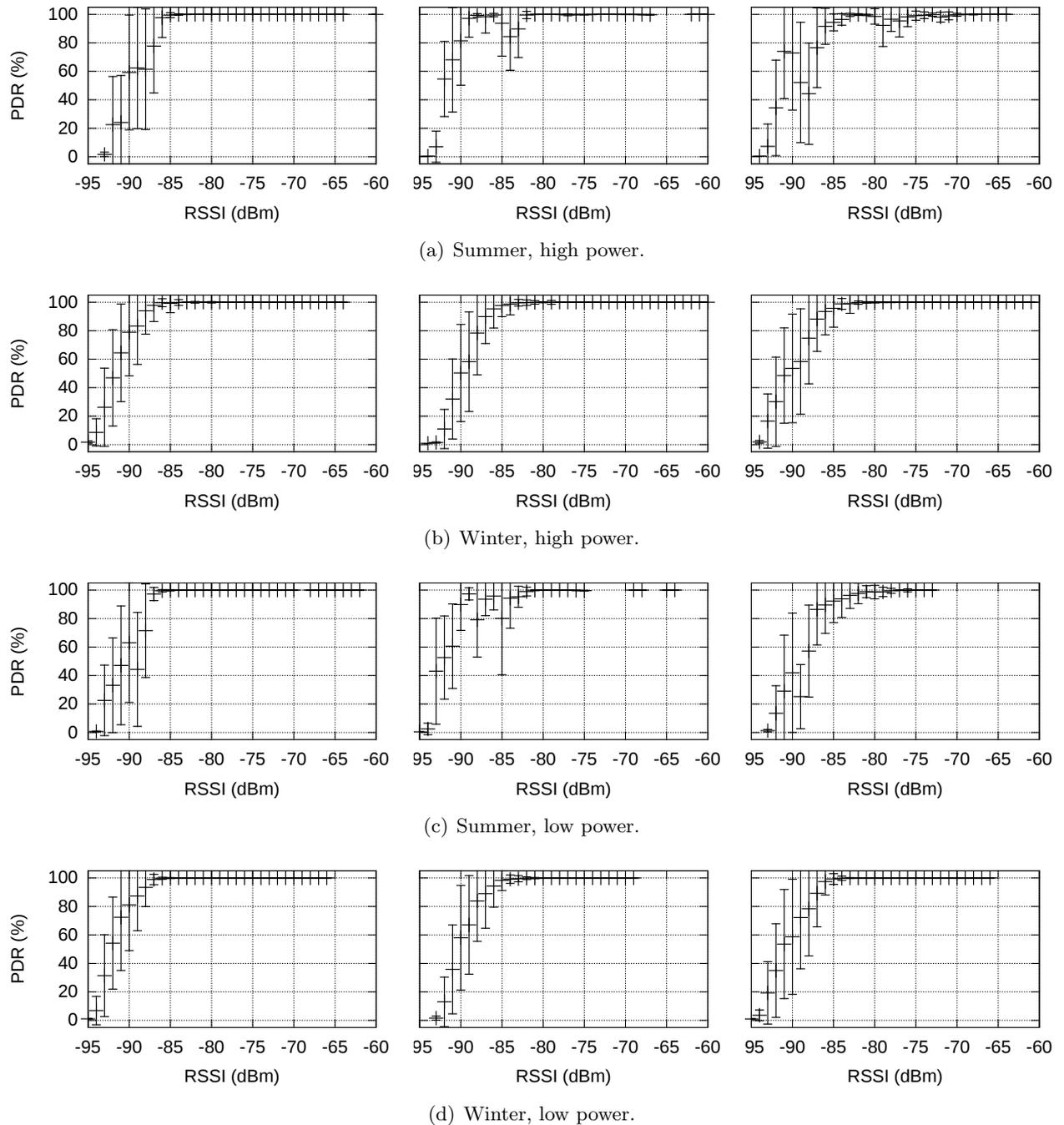


Figure 4.7: *RSSI* vs. *PDR* in OPEN (left), SPRUCE (center), and BEECH (right).

4.2.5 Link Classification

Looking at the transitional area allows one to grasp quickly the extent of communication range, but does not yield insights on the *fraction* of links characterized by a given quality.

We provide this “view” in two ways. The first is to compute the cumulative distribution

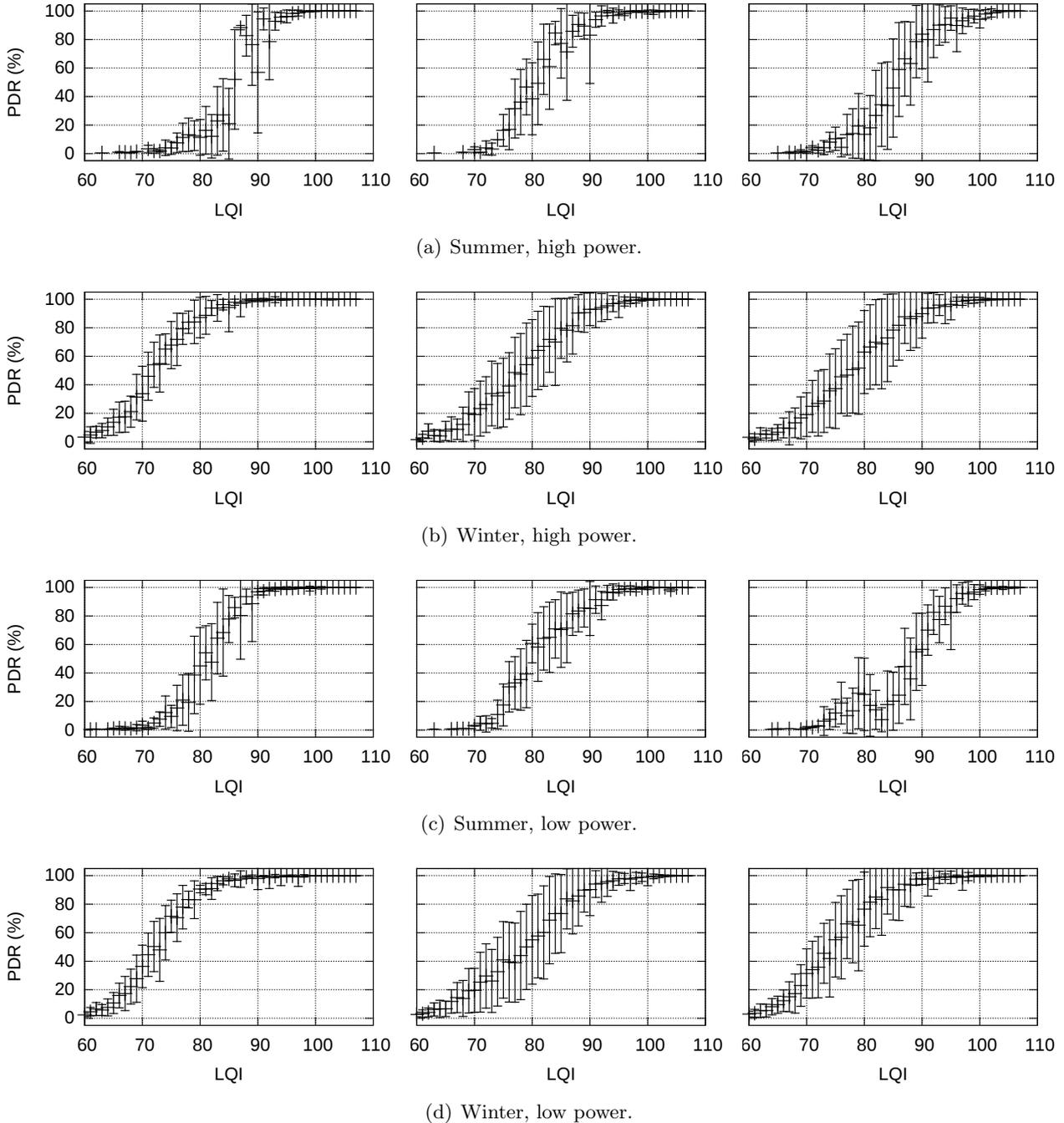


Figure 4.8: LQI vs. PDR in OPEN (left), SPRUCE (center), and BEECH (right).

function (CDF) of the links w.r.t. their PDR , as shown in Figure 4.10. The other is to adopt the classification used in [69]. Figure 4.9 shows five classes defined based on the PDR values: *dead* ($PDR = 0\%$), *poor* ($PDR < 10\%$), *intermediate* ($10\% \leq PDR \leq 90\%$), *good* ($90\% < PDR < 100\%$) or *perfect* ($PDR = 100\%$).

Figure 4.10 shows clearly the relative performance of the three environments. As we

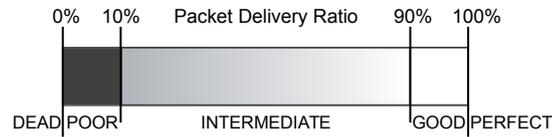
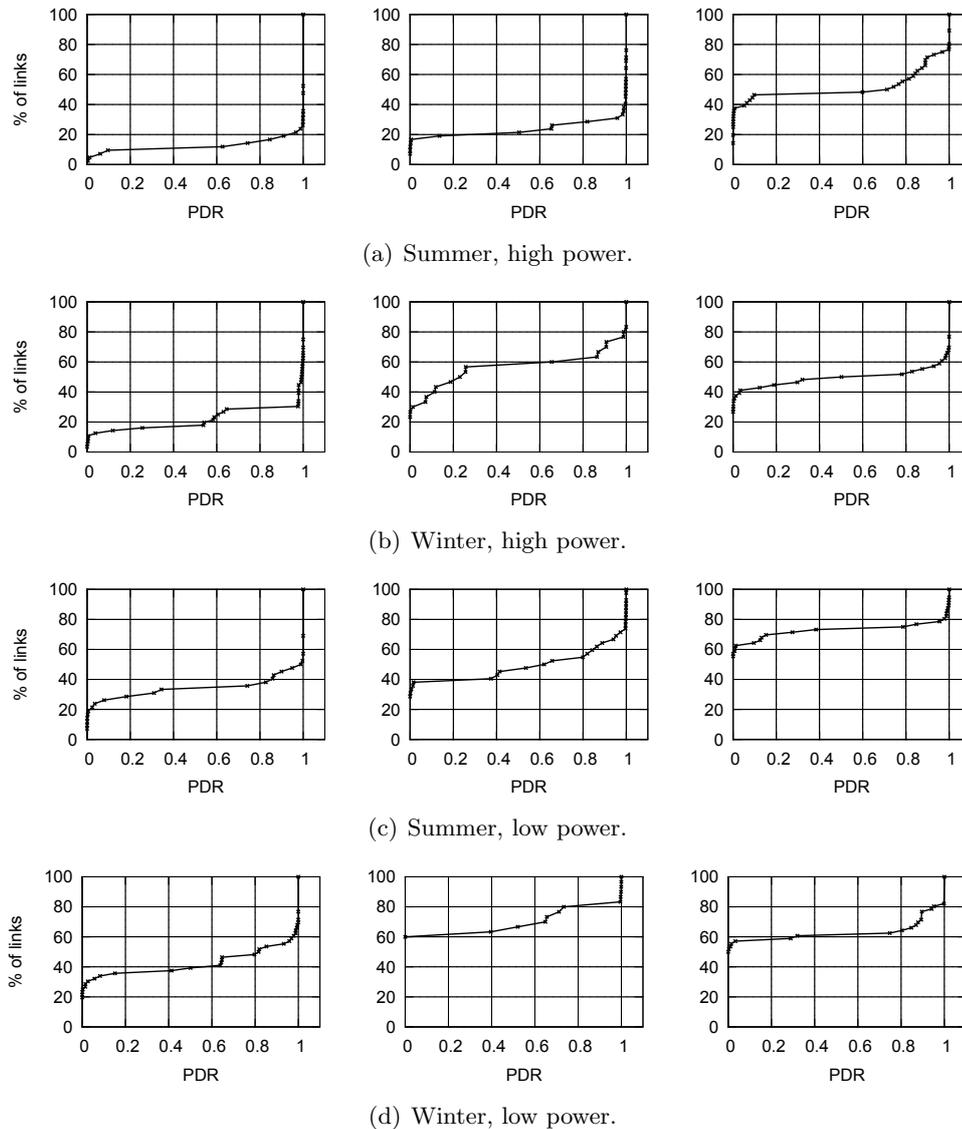


Figure 4.9: Link classification.

progress from OPEN, to SPRUCE, and finally to BEECH, the CDF “shifts” upwards, due to an increasing fraction of bad (i.e., dead, poor, intermediate) links. This can be seen in Figure 4.11, where especially the number of dead links increases noticeably as we go from

Figure 4.10: Cumulative distribution function of PDR in OPEN (left), SPRUCE (center), and BEECH (right).

OPEN to BEECH and, dually, the sum of good and perfect links decreases.

Figure 4.11(a) also shows that winter has a negative impact on OPEN. During summer, at high power, 83% of all links are good or perfect, but only 71% during winter. Further, the number of perfect links drops abruptly from 48% to 25%. This is consistent with our previous considerations about the transitional area “moving” closer to the sender during winter. On the other hand, the impact is much less marked at low power, although the number of dead links doubles.

Similar considerations hold for SPRUCE, in Figure 4.11(b), although the sum of good and perfect links is smaller than in OPEN, as already mentioned. However, the sum of intermediate, good, and perfect links accounts for 63% in winter and 82% in summer, respectively 40% and 66% with low power.

In BEECH, shown in Figure 4.11(c), communication at high power in winter is better

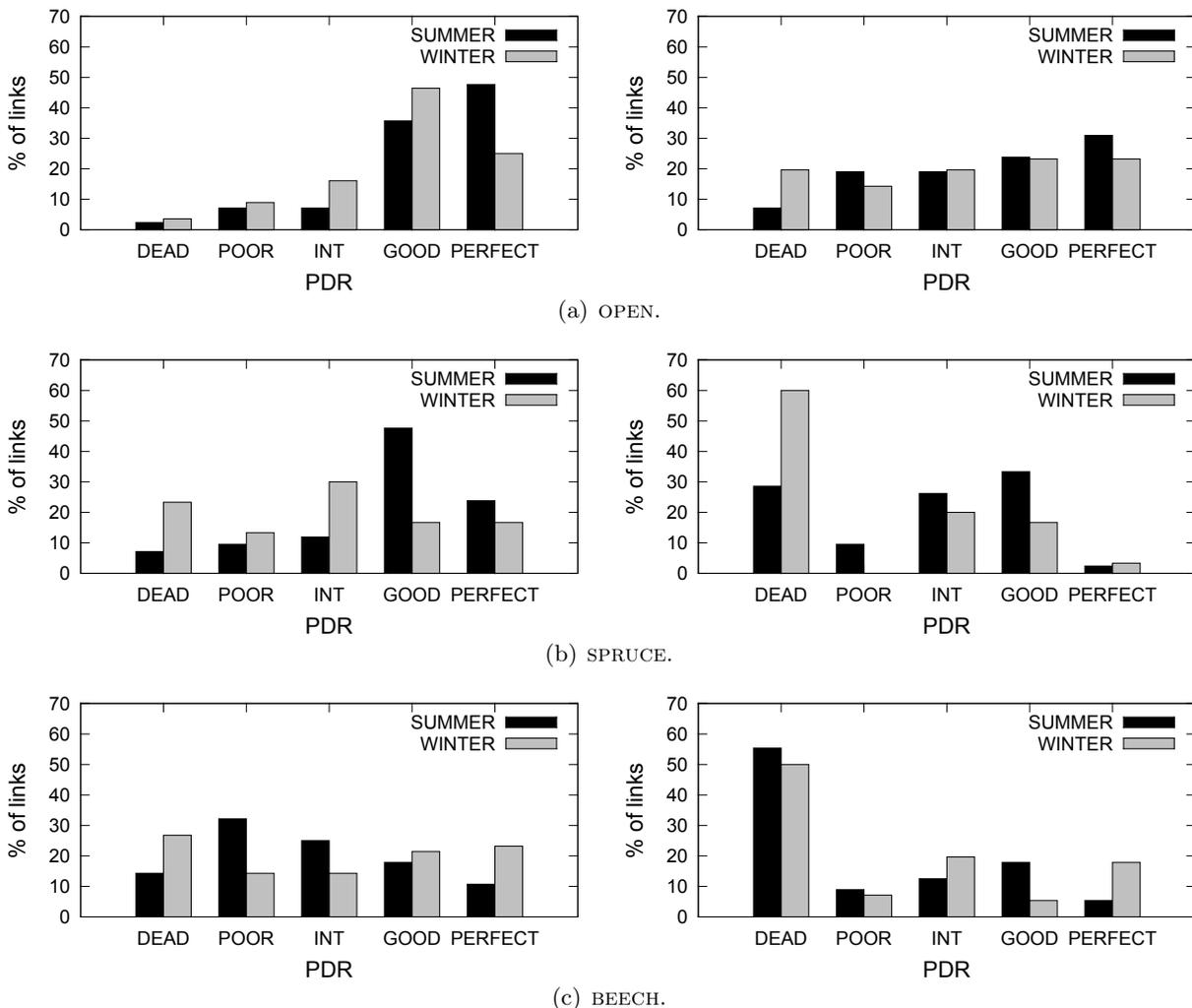


Figure 4.11: Seasonal variations, at high power (left) and low power (right).

than in summer, with 44% links being good or perfect during winter compared to 28% during summer, when more intermediate links are also present. The reason for this behavior is the fact that, during winter, foliage is not present in BEECH, and therefore, despite the presence of snow on the dense tree branches, communication is less impaired. A similar trend, albeit less marked, is shown at low power, with many links moving from good to perfect.

The difference between the two forests is less marked in winter: snow reduces the link quality in both SPRUCE and BEECH, but in the latter this fact is partially compensated by the absence of foliage. On the other hand, BEECH is worse in the summer, due to the presence of denser and broader leaves.

Now, let us turn our attention to Figure 4.7 and Figure 4.8, which plots $RSSI$ and LQI vs. PDR . During summer, in OPEN field a link with an average $RSSI$ above -85 dBm can be safely considered a perfect link but links with an average $RSSI$ between -90 dBm and -85 dBm are hardly distinguishable between poor, intermediate and good. More, LQI presents a saturation that makes it incapable to distinguish between good and perfect links. On the other hand, in comparison with the $RSSI$, LQI shows a smoother decay that enables a better classification of poor, intermediate and good links. Hence, the observations related to these figures confirm that the $RSSI$ and LQI alone cannot accurately classify the entire spectrum of link qualities independently but based on them one can easily identify the range of intermediate links and this can be used as a crude measure of the transitional region.

4.2.6 Day vs. Night

We now turn our attention to variations induced by the interleaving of night and day, which affects environmental parameters such as temperature and humidity, which in turn affect link quality. An example is provided by Figure 4.12, showing the PDR between

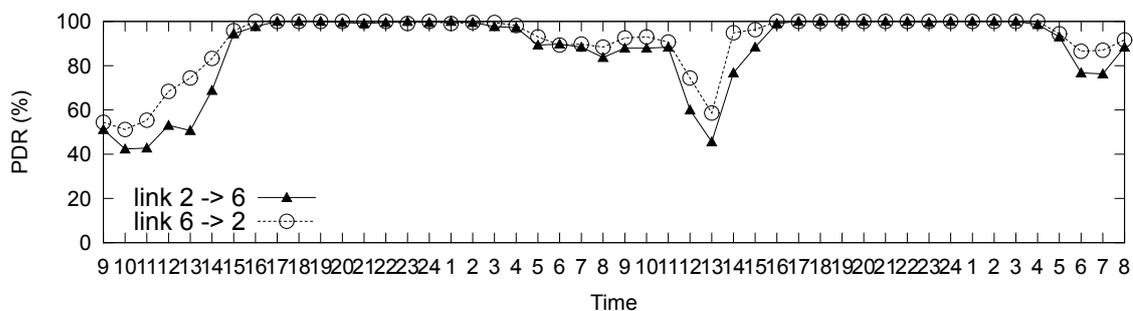


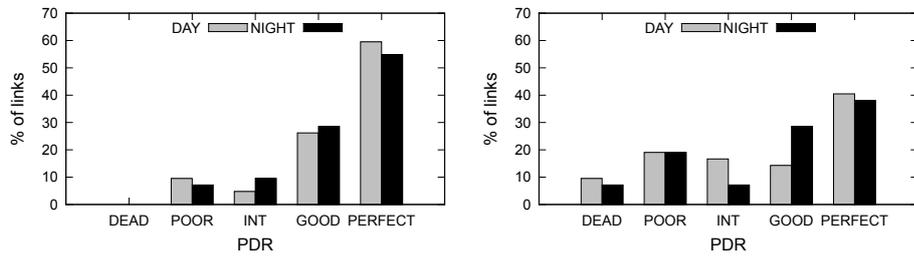
Figure 4.12: An example of day vs. night variations: the PDR of link 2 \rightarrow 6 and 6 \rightarrow 2 in OPEN, during summer, at low power. The node distance is 39 m.

node 2 and 6. The link is perfect at night, but its *PDR* drops significantly (as low as 40%, going from perfect to intermediate) during the day. To analyze in more detail this phenomenon for the entire network, unlike Section 4.2.5 we focus only on the charts showing link classification, due to the fact that these better highlight how links “move” across classes when changing from day to night. Figure 4.13 shows the results of our analysis.

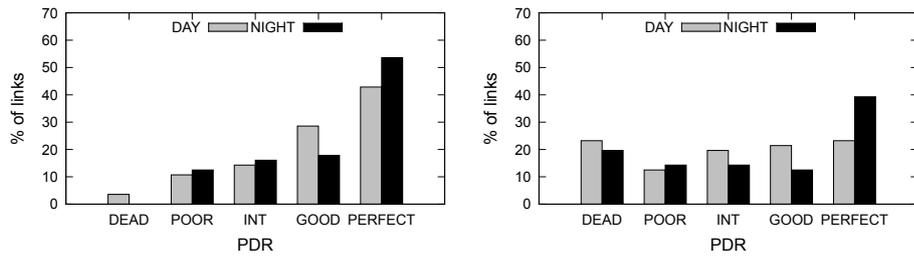
In OPEN, variations are very limited during summer, especially at high power. At low power, instead, the number of good and perfect links is higher during the night (66% vs. 54%), as several links move from intermediate to good. During winter, at high power, a significant fraction (10%) of links move from good to perfect when changing from day to night. This trend is even more marked at low power; the number of perfect links doubles and the intermediate links decrease.

In SPRUCE, during summer the number of dead, poor, and intermediate links remains basically unaltered, but perfect links increase by almost 15%. Similar trends are observed at low power, although less dramatic. Interestingly, however, in this case some links move from poor to dead, and others from intermediate to good and perfect. In a sense, the transition from day to night has the effect of “polarizing” the network into dead and perfect links, although overall the variations are somewhat limited. Similar considerations, albeit with more marked variations, hold for winter at high power. At low power, instead, the network is dominated by a significant fraction (60%) of dead links during both day and night, although the shift towards perfect links can still be observed.

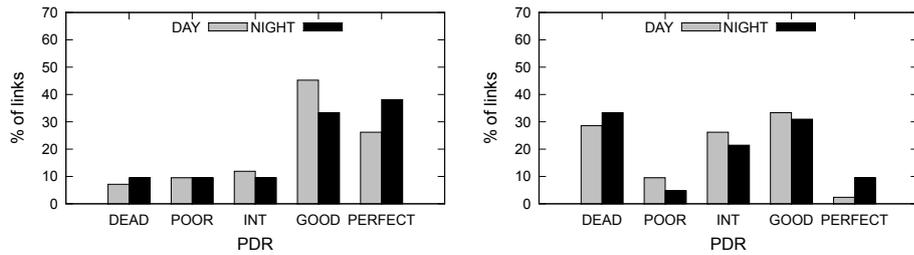
As for BEECH, the denser vegetation and broader foliage has the effect of “damping” the daily variations, introducing different trends. During summer, at high power, there is a polarization effect similar to SPRUCE. However, while there is an evident (almost 20%) decrease in intermediate links, there is also a decrease in dead and perfect links. The net effect is that it is the fraction of poor and good links that increases significantly, instead of dead and perfect as in SPRUCE. On the other hand, at low power the change from day to night induces almost no variation, due to the thick foliage that “protects” the links from environmental variations. During winter, thanks to the absence of foliage we observe a polarization towards dead and perfect links as in SPRUCE, although variations are smaller. Finally, at low power the trends we observed in the other cases are reversed: the number of intermediate links doubles, while perfect links are reduced by almost two thirds. We conjecture that this is an effect of the dense branches with snow, creating different “micro-ambients” for communication, although this aspect requires further, finer-grained investigation.



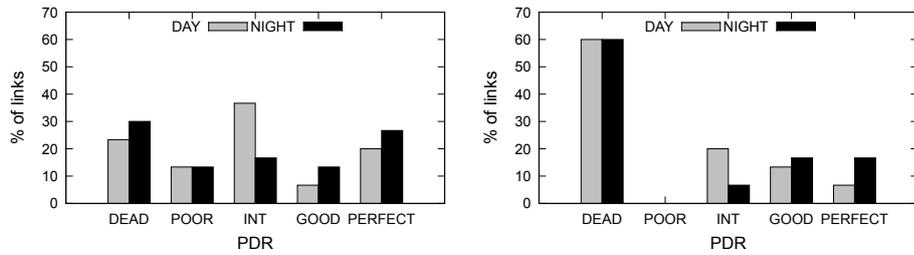
(a) OPEN, summer.



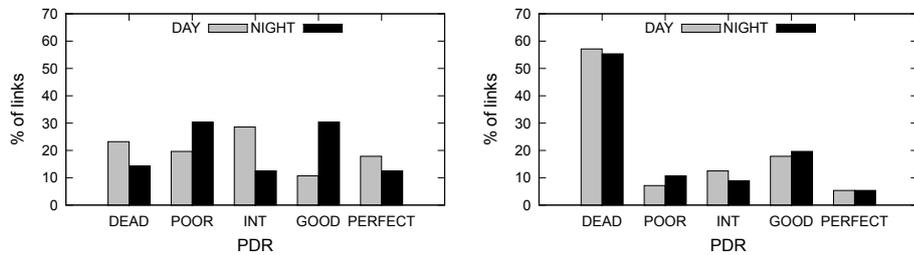
(b) OPEN, winter.



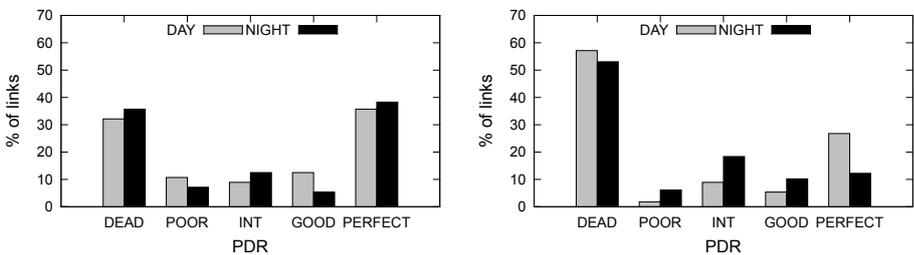
(c) SPRUCE, summer.



(d) SPRUCE, winter.



(e) BEECH, summer.



(f) BEECH, winter.

Figure 4.13: Day vs. night variations, at high power (left) and low power (right).

Table 4.3: Percentage of asymmetric links on CROSS

	high power		low power	
site	summer	winter	summer	winter
OPEN	N/A	N/A	9.61%	1.92%
SPRUCE	3.84%	N/A	5.76%	N/A
BEECH	N/A	1.92%	1.92%	1.92%

4.2.7 Link Asymmetry

We continue our analysis by concisely reporting about the presence of asymmetric links, as these are known to complicate the design of network stacks for WSNs. We follow the commonly accepted rule to consider a link between node i and j asymmetric if $|PDR_{ij} - PDR_{ji}| > 40\%$ [69].

Table 4.3 shows the data from our experiments, where we observed a small number of asymmetric links. These are of course more common at low power, but even in the worst case (OPEN, during summer, at low power) they account for less than 10% of the overall number of bidirectional links. Also, during summer at low power, the number of asymmetric links decreases when going through OPEN, SPRUCE, and BEECH: in other words, it follows the opposite trend of the network-wide PDR , as discussed in Section 4.2.2. The presence of asymmetric links is independent of the day/night variations we discussed in Section 4.2.6. Given that there is no interference and the environment is relatively uniform (i.e., the placement of nodes does not bear a big influence, as reported for instance in [53]), we conjecture this trend may be ascribed to different *noise* floor at different nodes, or to miscalibrated hardware, as suggested by some authors [91].

4.2.8 Link Transitions

We conclude our analysis by reporting about the dynamics of links in our environments as the frequency of link quality changes can have serious adverse effects on existing routing protocols, causing losses in the data or longer periods of disconnection while topology adjust, as we further elaborate in Section 4.3. Moreover, this might inform about the effort needed to accurately estimate the link quality and maintain the topology in different environments. For this we look at the total number of link transitions between classes, as we defined them in Section 4.2.5.

Table 4.4 shows the data from our experiments. The first important observation is that the links are more stable in OPEN, in the absence of vegetation, than in the forests, independent of the season. The second observation is that summer creates more transitions than winter in all environments, and their number increases as we progress from OPEN to SPRUCE and BEECH with the presence of vegetation and the increase of its

Table 4.4: Transitions on CROSS

site	high power		low power	
	summer	winter	summer	winter
OPEN	279	221	352	316
SPRUCE	528	336	546	387
BEECH	594	345	647	412

density.

4.2.9 Confirming Observations at Physical Layer

In this section, we investigate whether the results at the PHY layer obtained using a CROSS topology during the first and second campaign, in 2011, hold in experiments performed at a different time, 2013, and with a different setup, a GRID topology.

Experimental setup. These experiments are part of our fourth and fifth campaign, as in Table 4.1, and were performed during summer and winter. Due to some logistical issues, during summer, we performed the tests only in OPEN and SPRUCE. The site for BEECH and SPRUCE was the same, while the one for OPEN was different but with the same characteristics as the one in Section 4.1, which meanwhile had become a private property.

We also used a different topology this time, namely the GRID of 16 nodes from Figure 4.3(b). The nodes used are the same as in Section 4.1, in terms of hardware, packaging and setup. The only difference was that, to maintain rounds of 30 minutes with an increased number of nodes, we configured each node to transmit 115 packets per round. This allowed us to verify that the trends observed in the previous experiments on CROSS held also in the new experiments. Table 4.5 summarizes the per-round configuration parameters we used in TRIDENT for our experiments. Each node aggregated *PDR*, *RSSI*, *LQI* and *noise* floor over the 30-minute round. Rounds were executed one after the other without intervention for 2 days, after which they were stopped for one hour (at 9 AM) to download the data from the flash and replace the batteries. A total of 330 30 minutes experiments were run in each environment.

During the fourth and fifth campaign we used two sources of meteorological data besides the information collected using the on-board sensors. First, a meteo station [71] located 200 m from OPEN which provides temperature, relative humidity and precipitation measurements. Each of these values were sampled every 15 minutes. Second, two LASCAR EL-USB-2+ [44] temperature, relative humidity data loggers with readings from 0 to 100%RH and -35 to +80°C. Each of these values were sampled every 5 minutes. Thus, the data loggers provide measurements with a higher temporal resolution than available

Table 4.5: Values for per-round configuration parameters for PHY experiments using the GRID.

Parameter	Value	Parameter	Value
# of nodes (N)	16	# of probes per sender	115
list of senders	0 . . . 15	single-packet probe	yes
list of listeners	0 . . . 15	inter-probe interval	16 s
transmit power	27(−1 dBm), 14(−8 dBm)	inter-message interval	N/A
probing channel	18	aggregate logging	yes

from the meteo station and since they are collocated with our WSNs, the measurements accurately reflect the meteorological conditions experienced by the network.

Network-wide Packet Delivery Rate. When we went back to the field, the first question we wanted to answer was: To what extent the trends we observed in the previous campaigns remained the same? The experiments we report about in this section were performed one year later, with a different topology and, in the case of OPEN, in a different albeit similar site. We clearly hoped for close-enough results, but what we distilled from the analysis of the experimental data went beyond our expectations.

Indeed, during summer, as Figure 4.14(a) shows, the difference in network-wide *PDR*, for *all* the combinations of environments we tested, remains within 2–3%. During winter, at high power the difference in network-wide *PDR* is within 10% in the forests and 12% in the OPEN, while at low power, it remains within 5–6% in the forest and 10.5% in the OPEN. As Figure 4.15(a) shows, the biggest difference in network wide *PDR* is in OPEN, more at low power, in the environment in which the links are more affected by night and day variations. Two considerations are worth making. First, although the WSN topologies we used are different (CROSS vs. GRID), we chose the node distances in the latter to approximate the ones in the former, as shown in Figure 4.4 from Section 4.1.3. Second, and most important, the difference in network-wide *PDR* across the two sets of experiments is so small that it suggests that, when one looks at the *aggregate* link quality of the entire network, indeed the impact of the environment on a WSN deployment is relatively stable, on a seasonal scale. Moreover, looking at the meteorological data from the meteo station, we observed that in winter 2013, when we run the GRID campaign it did snow during the last 2.5 days out of the whole week we run the experiments while during the previous winter campaign it snowed for an entire day during the experiments, as it can be seen in Figure 4.16, which might impaired the communication.

In general, the WSN behavior in the one year apart campaigns is very similar. For example, Figure 4.17 shows the *PDR* variation in the summer campaign, confirming over

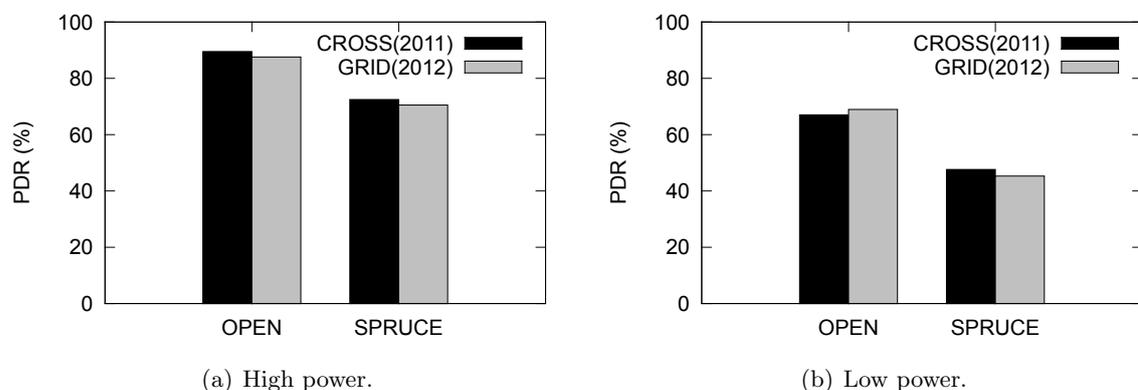


Figure 4.14: Network-wide average PDR in summer: comparison of the experiments in Section 4.2.2 and 4.2.9, performed one year apart and with different topologies.

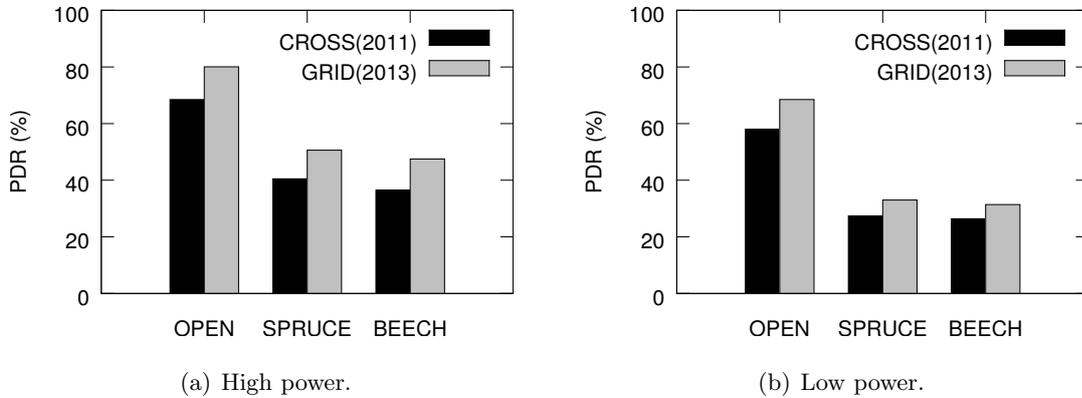


Figure 4.15: Network-wide average PDR in winter: comparison of the experiments in Section 4.2.2 and 4.2.9, performed one year apart and with different topologies.

the longer 7-day period the observations in Section 4.2.6, i.e., daily variations in OPEN affect the WSN only marginally at high power, but significantly at low power. Next we focus our attention on comparing the results from winter campaigns as this is the season in which we collected data traces from all three environments.

Link classification. Before analyzing into details each environment, it is worth making some observations from looking at the fraction of links characterized by a given quality: OPEN is still better than SPRUCE and BEECH at both powers and the two forests are still comparable during winter after one year and using a GRID topology.

Figure 4.18 shows the difference between the performance of the three environments

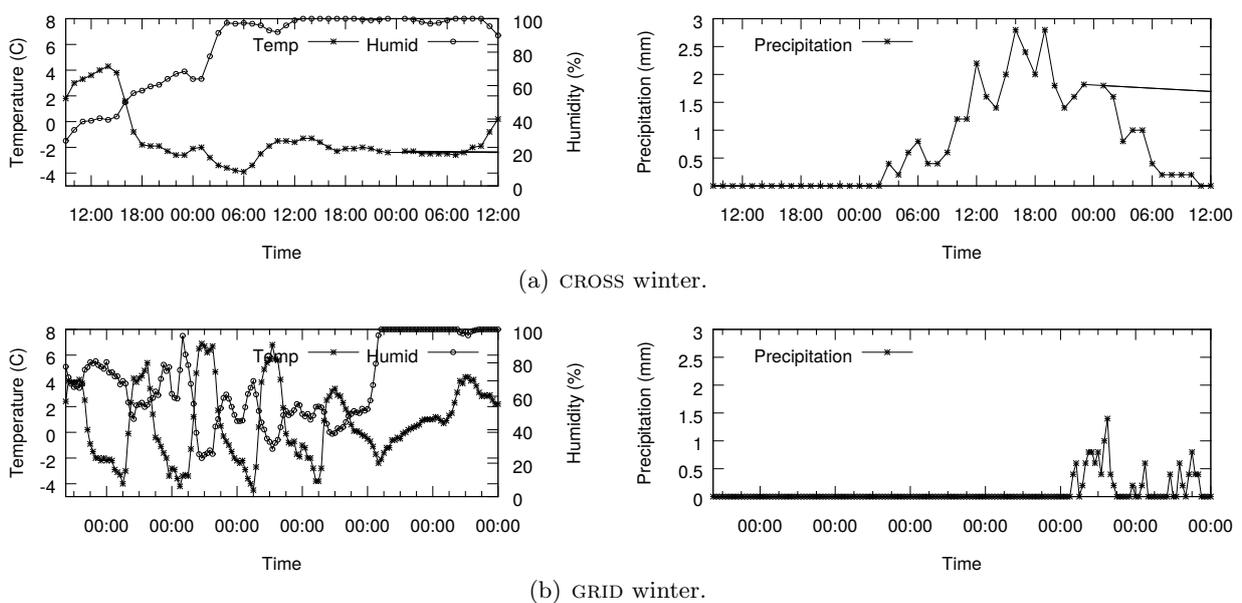


Figure 4.16: Meteorological conditions: temperature and humidity (left) and precipitation (right).

on CROSS and GRID winter campaigns. Looking at the link distribution in classes, one can see that the difference in network-wide PDR between the two winter campaigns is not due to the links jumping between extreme classes, but from one class to the adjacent one, up or down. For example, when the network-wide PDR at high power in OPEN increased with 11% between the years is because the links became stronger moving from dead to poor and from good to perfect.

In OPEN, at high power, of all links 72.62% are good and perfect still less than during summer grid 83.64% (trend confirmed) and very close to the 71% of winter CROSS. If we look at the number of perfect and good links, we observe more good than perfect links, with a difference between the two classes of 8% compared to the 11% difference computed for the CROSS winter campaign. At low power, of all links, 49.45% are good and perfect, less than during the GRID summer (i.e., 57%), and their sum is higher than the total number of dead and poor links.

In SPRUCE, at both powers, the sum of good and perfect links is lower than in OPEN (i.e., 35% and 22%). Moreover, as in previous CROSS winter campaign, the number of intermediate links is higher than in OPEN, which confirms that as we move from one environment to another we see more intermediate links. Looking at links distribution in classes in both forests we see that they are still comparable in winter, due to the absence of leaves in BEECH. Also, in BEECH, the sum of good and perfect links at high power is 48.33% and at low power is 25.71% with a difference of only 6.5% between the two winter campaigns.

Day vs. night. We now turn our attention to variations induced by night and day, and investigate if our trends still hold, for the winter campaigns.

In OPEN, at both powers, trends from the previous winter campaign are confirmed: the number of intermediate links decreases during night and number of good and perfect links increases. At high power, from day to night, the sum of good and perfect links

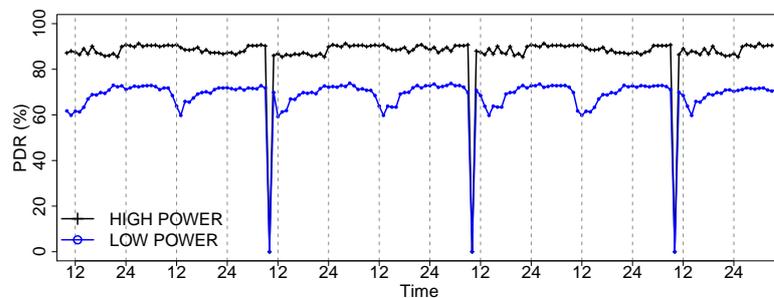


Figure 4.17: Daily variations in OPEN, at high and low power. The sudden drop to zero corresponds to the 1-hour period in which, every two days, we downloaded the data from the flash of the nodes, and replaced their batteries.

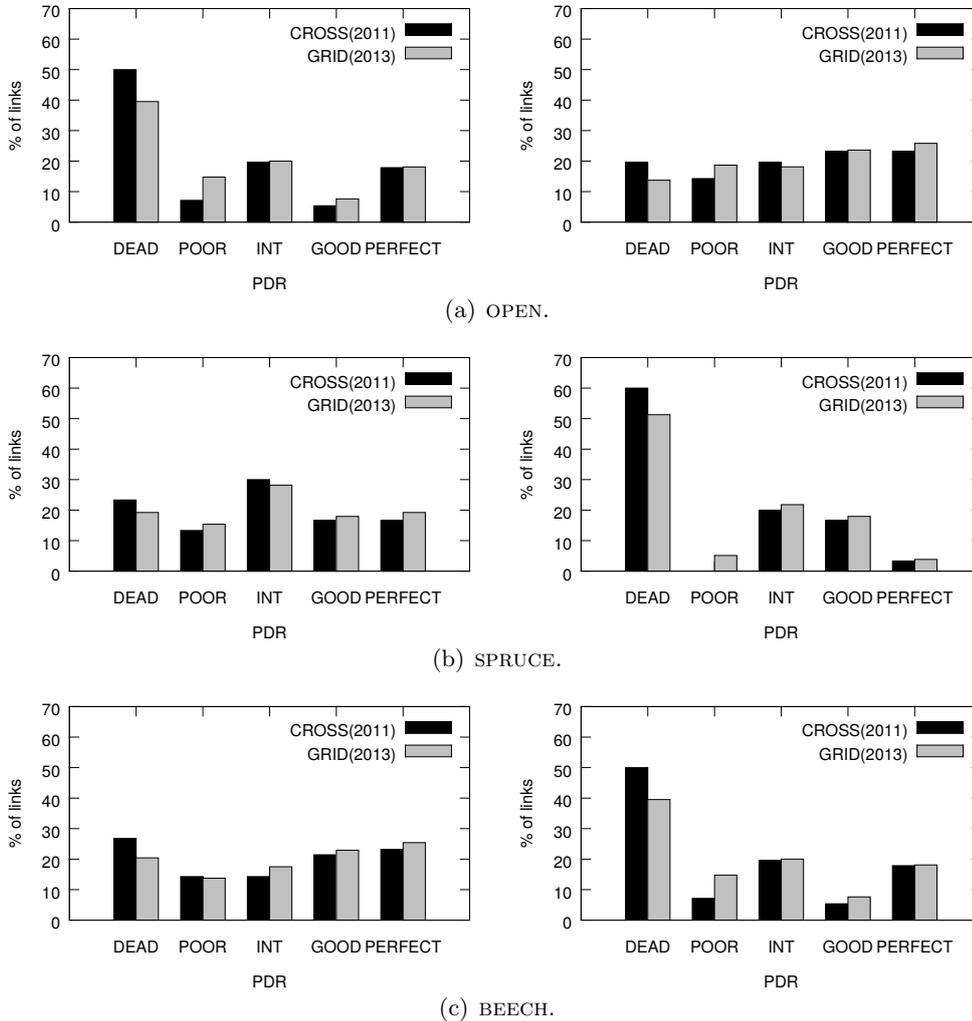


Figure 4.18: Winter link classification, at high power (left) and low power (right).

increases by 8.75%, while at low power the number of perfect links increases by 1.7%. This is also supported by the fact that the average *PDR* during nighttime is higher than during daytime.

In SPRUCE, at high power, during night, the number of dead links and the number of good links increases, and the number of intermediate links decreases by 14%, the network being polarized towards dead and good/perfect links. At low power, the shift of links is also from intermediate towards dead or good/perfect.

In BEECH, at high power, during night, we see the same polarization effect as in the SPRUCE, towards dead and perfect and a decrease in the number of intermediate links which we could not see during winter 2011 (when the number of intermediate links increased during night). One explanation of this inversed trend can be that during winter CROSS campaign, in BEECH it started snowing during the first night of the experiments

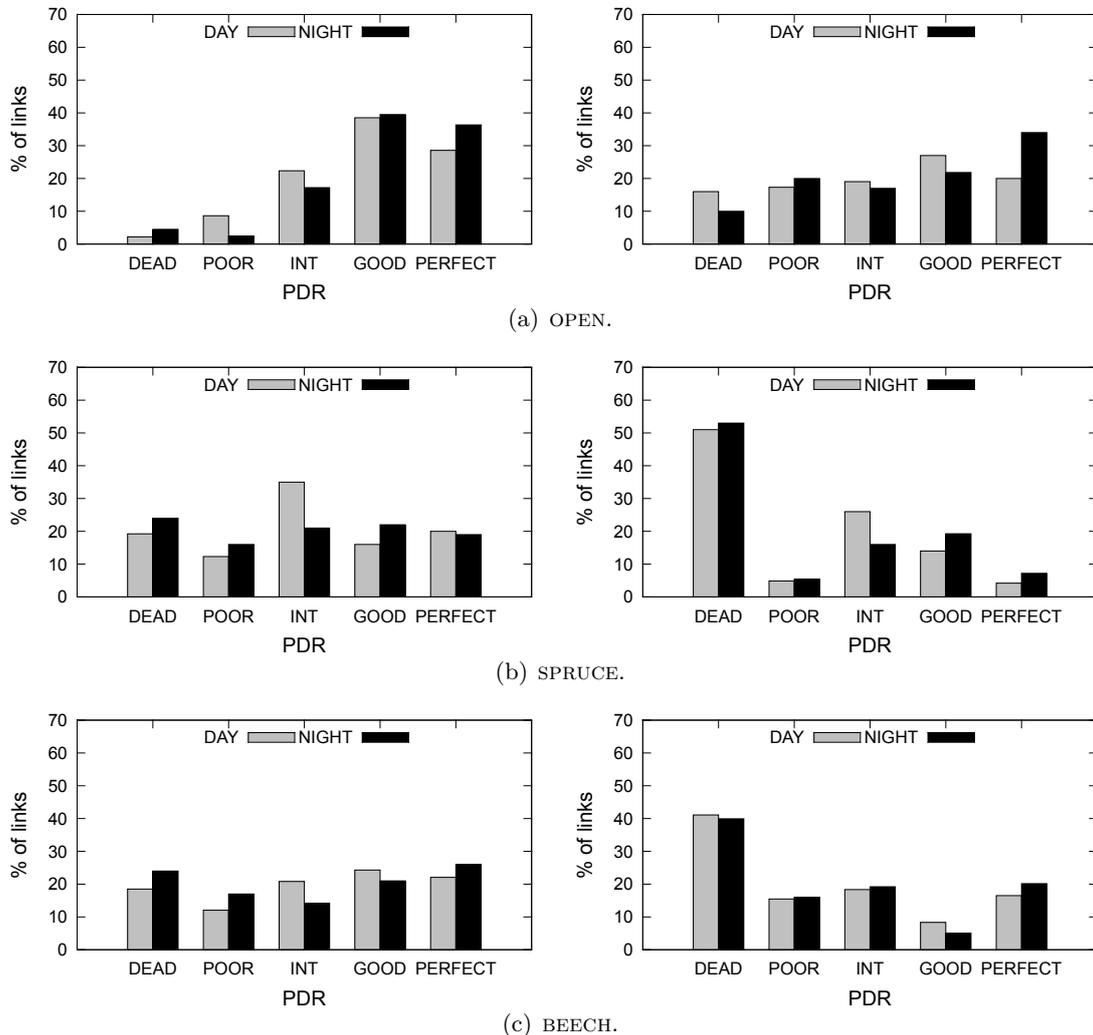


Figure 4.19: Day vs night variations, at high power (left) and low power (right).

which might have increased the number of the intermediate links. At low power, the network is polarized towards poor and perfect but the variations are quite small in comparison with the variations observed in winter CROSS campaign when the number of intermediate links almost doubled and the number of perfect links reduced by almost two thirds.

Link asymmetry. As in our previous campaign, we observed a small number of asymmetric links in contrast with some reported studies. They are more common at low power, as can be seen in Table 4.6 but even in the worst case scenario of OPEN low power, they account for less than 8% all number of bidirectional links and less than during summer (i.e., 10%). Also, the number of asymmetric links decreases when going from OPEN to forests, following an opposite trend of the network-wide PDR and, as discussed in Section 4.2.7 the the presence of asymmetric links is independent of day/night variations.

Link transitions. As Table 4.7, the observations on links dynamics hold for the GRID

site	high power		low power	
	summer	winter	summer	winter
OPEN	N/A	3.75%	10%	7.91%
SPRUCE	3.57%	2.5%	7.14%	6.25%
BEECH	N/A	2.91%	N/A	6.66%

Table 4.6: Asymmetries links on GRID

site	high power		low power	
	summer	winter	summer	winter
OPEN	3134	2444	3864	2940
SPRUCE	5428	3505	5980	4386
BEECH	N/A	3664	N/A	4782

Table 4.7: Transitions on GRID

campaigns. Thus, summer induces more link transitions than winter in all the environments and as we progress from OPEN to the forests the number of transitions increases, as the presence of the vegetation increases.

4.3 Routing Layer

In this section, we investigate whether the trends we discussed in Section 4.2 bear an impact on the routing layer. To this end, we consider two routing protocols. The first is the Collection Tree Protocol (*CTP*) [33], a representative of commonly employed tree-based routing schemes. The second is the Opportunistic Routing Protocol (*ORW*), a more recent opportunistic routing approach.

At a high-level, the behavior of the *CTP* protocol is characterized by two core elements: a routing metric used by every node to select a parent node in the tree and a set of reliability mechanisms to improve the fraction of messages correctly delivered to the root of the tree. The version of *CTP* we used leverages the *expected transmission count* (ETX) [23] as a basis for its routing metric. This metric is computed based on beacons broadcasted by every node. In *CTP*, nodes select their parent in the tree by minimizing the *end-to-end* ETX, i.e., the number of retransmissions required to deliver a message to the root, possibly across multiple hops. Reliability is mainly achieved by using randomly-scattered retransmissions and link-layer acknowledgements. In *CTP*, the latter is entirely implemented in software, instead of using CC2420 hardware-level acknowledgements, to retain better control of retransmission timeouts.

On the other hand, *ORW* uses a completely different approach to build routes to the root. *ORW* uses the *expected duty cycle* (EDC) [41] as routing metric, which estimates the expected duty cycle needed to reach the root. Compared to *CTP*, *ORW* aims at reducing the duty cycle and not the number of retransmissions. In *ORW*, nodes which want to transmit a packet will just send the packet, until any of the candidates wakes up, with its own EDC and a predefined threshold for it for the next hop. Any node that wakes up first and has a lower EDC value than the predefined EDC threshold, receives the packet, acknowledges the receive and forwards the packet following the same procedure.

Table 4.8: Per-round configuration parameters and their values for GRID topology campaigns.

Parameter	Value
number of nodes (N)	16
transmission power	27 (-1 dBm), 14 (-8 dBm)
probing channel	18
inter-probe interval (IPI)	30 s
round duration (RD)	30 minutes
wake-up interval(WI)	default BoX-MAC

4.3.1 Periodic Data Collection with CTP

Findings reported in this section were gathered during the fourth and fifth experimental campaign, as described in Table 4.1, Section 4.1.1, using HARPOON, our tool for routing performance assessment. Networks were deployed on the same sites as for the PHY experiments presented in Section 4.2.9.

4.3.1.1 Test Execution

We used the 16 nodes GRID topology as in Section 4.2.9, as we wanted *i*) to have enough nodes to build the topology from, and *ii*) to have a uniform node placement to avoid biasing the tree construction. We first deployed it in SPRUCE and BEECH, where node placement was constrained by distances between trees, and then reproduced it in OPEN with the same distances. As we already mentioned, this yielded a rich set of links at distances varying from 10.2 m to 63 m. The nodes are the same as in Section 4.1, in terms of hardware, packaging, and setup.

We ran experiments using HARPOON on this topology. Nodes were configured with the standard settings for *CTP* and *BoX-MAC* [52]. Nodes acted as sources sending messages towards the sink (node 0) every 30 s. Unlike the PHY tests, transmissions in HARPOON are not centrally scheduled. We used a packet size of 105 B, including a 94 B payload, representative of many WSN applications. As before, experiments were divided in 30-minute rounds, performed by interleaving high power and low power settings. Table 4.8 summarizes the per-round configuration parameters we used in HARPOON during our experiments. The collection tree was reset before each round, therefore, we ignore the first few minutes of data to account for the tree building phase. A total of 330 30-minutes experiments were run in each environment. Unfortunately, during summer, when analyzing the data we discovered that a few nodes (2, 4, 6, 14) malfunctioned, and removed them from the data set. During winter campaign all nodes were properly functioning.

4.3.1.2 Impact of Environment on CTP

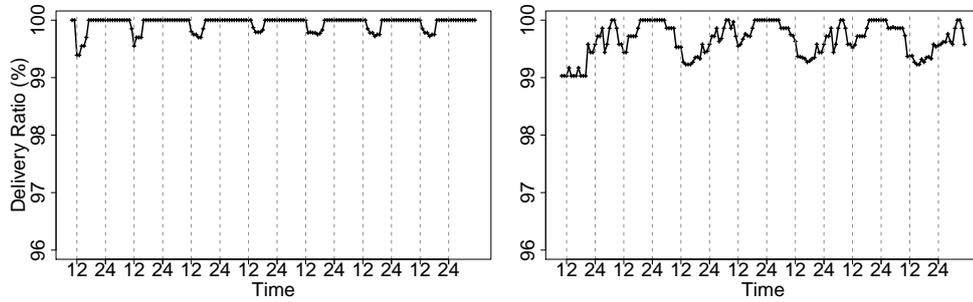
We looked at three metrics to assess the impact of environment-induced differences in connectivity on the operation of *CTP*: 1. *delivery rate*, i.e., fraction of messages received at the sink over those collectively sent by the sources; 2. *number of CTP beacons* sent; 3. *number of tree parent changes* occurred. The first metric measures the reliability of communication, while the other two measure the overhead necessary to ensure such reliability.

Figure 4.20 shows that during summer, the trends in delivery rate are very different in OPEN and SPRUCE. The former remains stable at 100%, except during the central day hours, where the delivery rate drops as low as 96% for low power. On the contrary, in SPRUCE the range of variation in delivery rate is less dramatic, with drops limited to 98.5%, even for low power. However, variations are much more frequent, and yield perfect reliability only for very short periods and only at high power. We argue that this behavior is induced by the environment. In OPEN, the absence of vegetation yields better and more stable links. Specifically, *i*) the link quality of the network in SPRUCE is lower than in OPEN (Figure 4.14) and, *ii*) in SPRUCE there are many bad links, while in OPEN there are a majority of good and perfect links (Figure 4.13(a) and 4.13(c)). On the other hand, in SPRUCE the negative effect induced by the vegetation is partially compensated by the fact that the latter “protects” links from environmental changes. In other words, links are worse on average, but subject to less abrupt variations.

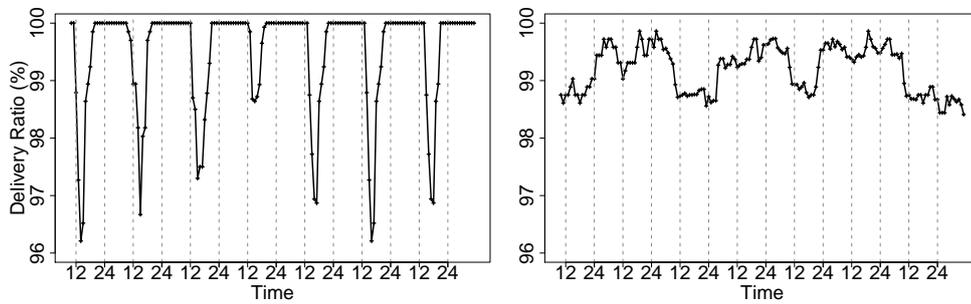
Figure 4.21 shows instead the number of beacons as a function of time. Two observations can be made. First, this number is higher—almost double—for SPRUCE than in OPEN. Second, the variation over time in the former appears to be somewhat regular and independent of environmental variations (e.g., due to night and day), which are instead more visible in OPEN.

These observations can be explained with arguments similar to those for delivery rate. Indeed, *CTP* must frequently update the link quality information in SPRUCE simply because links are more brittle; we verified this hypothesis in our data sets from PHY layer experiments on GRID, where SPRUCE experiences many changes across link classes as shown in Table 4.7 of Section 4.2.9. This yields an increased energy expenditure for the control traffic required to maintain essentially the same (good) level of reliability of OPEN. In the latter, links are more stable; therefore, less updates are required, triggered to a greater degree by variations in the environment (e.g., day/night patterns).

These considerations are confirmed by looking at parent changes, shown in Figure 4.22 for high power only. Overall, the number of (average) parent changes we observed is exactly double—48 in SPRUCE vs. 24 in OPEN, due to the more challenging environment. However, nodes obviously experience different variations, due to the quality of their links.



(a) High power.



(b) Low power.

Figure 4.20: Delivery ratio for *CTP* in OPEN (left) and SPRUCE (right) during summer.

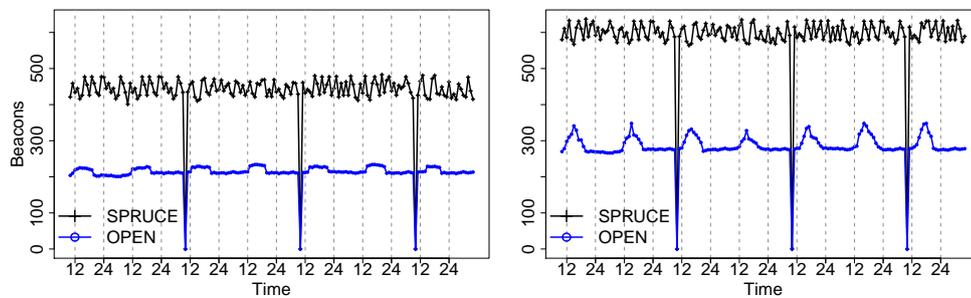


Figure 4.21: *CTP* beacons in OPEN and SPRUCE, high (left) and low (right) power during summer.

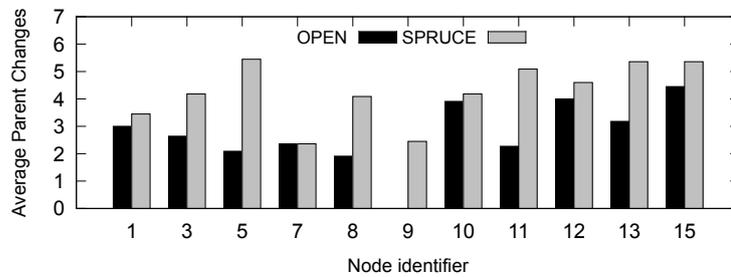


Figure 4.22: Parent changes in OPEN and SPRUCE, at high power during summer.

Daily patterns are less relevant here because the overall number of changes is smaller than in the case of beacons.

We ran the same experiments with HARPOON and *CTP* during winter. In order to compare against the summer campaign, we use the data collected during 7 days and do not take into consideration nodes 2,4,6,14 (which malfunctioned during the summer campaign). We can only compare OPEN and SPRUCE as we did not run *CTP* experiments in BEECH during summer.

Figure 4.23 shows the trends in delivery rate during winter. In OPEN, at both powers, the delivery rate drops during the hot hours of the day, to 97.77% at high power and 92.88% at low power. Compared with summer *CTP* campaign, the delivery rate is lower now. This could be explained by the lower link quality in OPEN winter although links are a bit more stable, as shown in the transitions Table 4.7.

On the other hand, in SPRUCE, the rate drops are less dramatic than in the OPEN but the delivery rate is still lower than during summer. In fact the *PDR* in SPRUCE during winter is lower than during summer and the links are exposed to less abrupt variations in temperature and humidity than the ones in the OPEN, with the exception of the 4th day when it snowed during the experiments.

If we look at the number of beacons, see Figure 4.24, in OPEN and SPRUCE we observe the same trend as during summer, there is a positive correlation of the number of beacons with the day-night cycles in the OPEN and in the forest the number of beacons is somewhat independent of the day and night variation. Moreover, the number of beacons used by the control traffic in the forest is higher than in the OPEN, almost 1.4 times higher (an explanation can be in the correlation with the number of transitions which is still higher in the evergreen SPRUCE than in the OPEN field even during winter, as in Table 4.7).

If we compare the number of average parent changes in OPEN and SPRUCE two observations can be made: during winter the number is smaller than in summer in both environments (24 in OPEN summer compared to 18 in winter and 48 in SPRUCE summer compared to 31 in winter). Again, this can be explained with arguments similar to those above. Indeed, during winter SPRUCE has more stable links than during summer.

4.3.2 Periodic Data Collection with *CTP* and *ORW*

When we started our investigations, *CTP* was arguably the most used protocol for data collection applications for the past ten years. Given that *ORW* is a more recent protocol, it has not been used as widely as *CTP* and it has not been thoroughly investigated either in outdoor real-world scenarios. Considering the current situation we asked ourselves: *i)* how does the environment impact the operation of *ORW*, and *ii)* which data collection protocol is better in terms of reliability and in terms of energy consumption of the network.

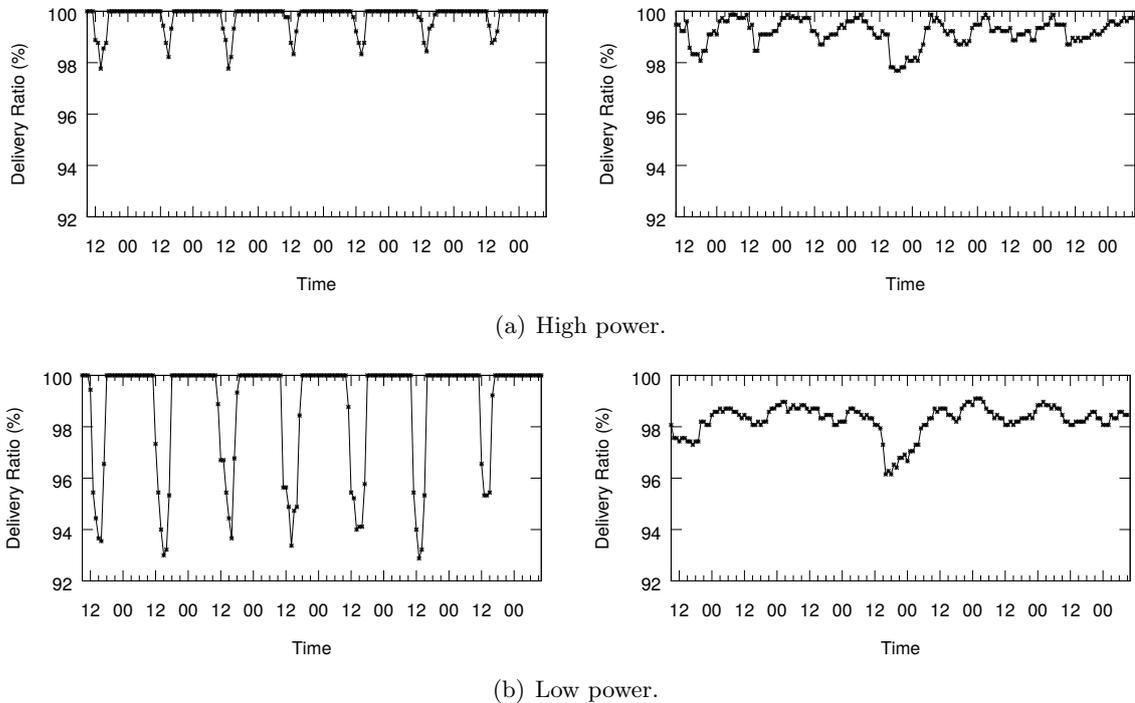


Figure 4.23: Delivery ratio for CTP in OPEN (left) and SPRUCE (right) during winter.

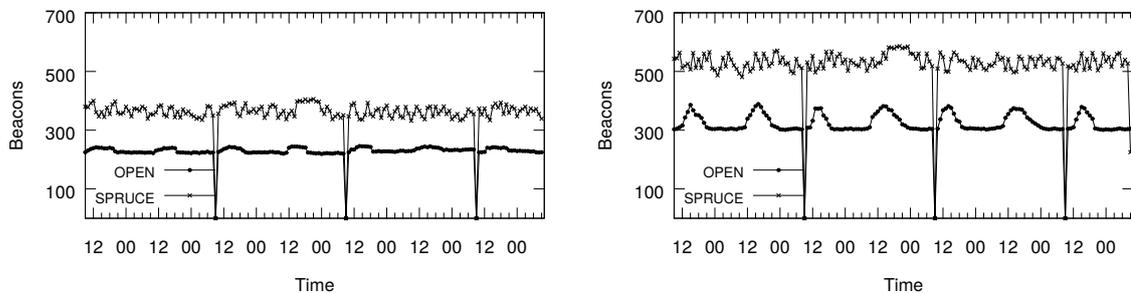


Figure 4.24: CTP beacons in OPEN and SPRUCE, high (left) and low (right) power during winter.

4.3.2.1 Test Execution

To answer the above questions we performed an in-field experimental comparison, immersing two networks running HARPOON, with *CTP* and *ORW*, using the same setup for both protocols. These experiments are part of our sixth campaign, as shown in the campaigns Table 4.1. Due to the reduced number of TMote Sky nodes we had at the time of the experiments, we needed to cut out the uppermost and the leftmost edges of the GRID from Figure 4.3(b) and thus used a GRID_SMALL topology of 9 nodes, still yielding a rich set of links with lengths from 13 to 55 m. Experiments were run in an OPEN field, on a different site, in Pergine, Italy, but having the same characteristics as the ones on

Mount Bondone.

We configured HARPOON with the same per-round configuration parameters as in our previous *CTP* campaigns, see Table 4.8, using 9 nodes and two different non-overlapping channels for the protocols, i.e., channel 18 for *CTP* and 26 for *ORW*.

We focused on two key metrics to assess the impact of the environment on the two protocols: *reliability* and *energy consumption*. Reliability is evaluated through the *delivery rate* while the energy consumption through the average *duty-cycle* in the network. The two are related, as the latter is a measure of the effort to maintain the reliability. Both metrics are collected and computed using HARPOON.

For a fair comparison between the two protocols, we skip the first three minutes when measuring duty-cycle, as *CTP* shows a high duty-cycle during this initial period due to its link probing.

4.3.2.2 Confirming Observations on CTP

When we went back to the OPEN field, the first question we wanted to answer was: To what extent trends that we observed in the previous *CTP* campaigns remained the same? The experiments we report about in this section were performed one year later, with a sparser topology, and in a different albeit similar site.

Indeed, during summer, as Figure 4.25 shows, the difference in average delivery ratio, remains within 3% at high power and 10% at low power. Two considerations are worth making. First, the environmental conditions were different during the two summers, with the one of sixth campaign on GRID_SMALL being much more humid. Second, and most important, although the WSN topologies we used were the same, a grid, their density was different. Indeed, the GRID of summer 2012 had 16 nodes and *CTP* might had benefit from a greater density than GRID_SMALL topology. As in the case of the physical layer, the difference in the delivery ratio across the two sets of *CTP* experiments is so small, that it suggests that, when one looks at the aggregate delivery ratio of the network, indeed the impact of the environment on a WSN deployment is relatively stable, on a seasonal scale. More, as in our previous experiments, *CTP* showed spikes of reduced reliability, going as low as 80% at low power during the hottest hours of the day. These spikes grow to 72% during the hours when the humidity exceeds 75%, as depicted in Figure 4.26.

4.3.2.3 Impact of Environment on CTP and ORW

Next, we compare the performance of *CTP* and *ORW* in our scenario using the GRID_SMALL topology. Table 4.9 summarizes our results in terms of reliability. With all settings, *ORW* is more reliable than *CTP*, and maintains a reliability above 99.66% independent of the

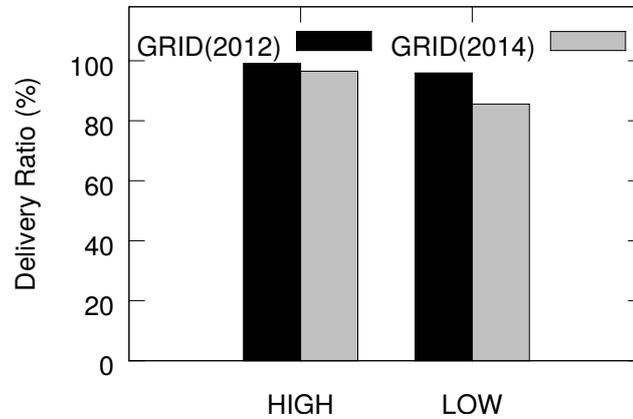


Figure 4.25: Delivery ratio for *CTP* on GRID (4th campaign) and GRID_SMALL (6th campaign).

power in contrast with *CTP*, which as the power decreases leads to a reliability of 85.58%. On the other hand, this yields an increased energy expenditure for the *ORW* required to maintain the almost perfect level of reliability. As depicted in Table 4.10, *CTP* and *ORW* have different energy trade offs. The average *duty-cycle* of the two networks shows that *CTP* is more energy efficient than *ORW* with an average of 4.5% in comparison with 6%, which is consistent with the literature [41, 30] when using the default settings for *BoX-MAC* protocol. Table 4.11 shows that, on average, *CTP* produces twice the number of retransmissions of *ORW*, accounting for between 1.92% to nearly 2.65% at low power, while *ORW* retransmits on average 1.13%, independent of the power.

We now analyze the effect of the environment, with temperature and humidity variations as shown in Figure 4.28, on the performance of both protocols. To highlight the impact of the temperature and humidity on our outdoor WSNs, we focus on a set of

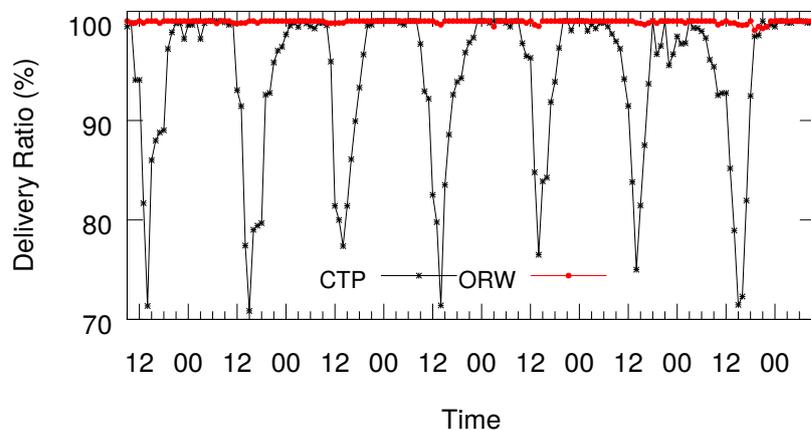


Figure 4.26: *CTP* and *ORW* delivery ratio on GRID_SMALL.

Table 4.9: Delivery rate statistics

Power	Delivery rate			
	<i>CTP</i>		ORW	
	avg	std	avg	std
High power	96.52	0.29	99.95	0.01
Low power	85.58	0.17	99.66	0.03

Table 4.10: Duty cycle statistics

Power	Duty-cycle			
	<i>CTP</i>		<i>ORW</i>	
	avg	std	avg	std
High power	4.45	0.03	5.85	0.02
Low power	4.65	0.04	5.78	0.02

experiments run during daytime, from 10:00AM to 5:30PM, on September 14, as during nighttime, both protocols show a high reliability over 99.2%. On close inspection, in Figure 4.27, one can observe that during the hottest hours of the experiments, the reliability of *CTP* decreases as low as 95%, in contrast with *ORW* which does not deliver under 99.63%. Our experiments bring further evidence that *CTP* has a decrease in reliability, to 87%, when the humidity increases over 40%, during the 5PM experiment. On a closer inspection the decrease in reliability can be explained by looking at the number of packets lost and failed ACKs at the node level. Figure 4.29 clearly shows an increase of these two with the increase of the temperature. Moreover, Figure 4.30 shows an inverse relationship of the two parameters with the humidity.

In terms of *duty-cycle*, the behavior of the two protocols does not show relevant changes between experiments with different powers. Figure 4.31 shows that *ORW duty-cycle* is higher when compared to *CTP*, having a rather stable value around 6% with few oscillations which we ascribe to node failures in the network. On the other hand, *CTP* maintains a low and stable *duty-cycle* of 4.5% in the “comfort zone” of a high reliability (over 95%) while, in the worst case scenario, when temperature and humidity affect the links or the network gets partitioned, *CTP* increases the *duty-cycle* up to 6% as *ORW*.

Next, our results show, see Figure 4.32, that *CTP* exhibits a higher count of retransmissions than *ORW*, and the number of retransmissions is positively correlated with the temperature. Additionally, *ORW* achieves the same count of retransmissions, 1.13 on average, independent of the environmental conditions and the power of the experiments.

Table 4.11: Retransmissions count statistics

Power	Retransmissions count			
	<i>CTP</i>		<i>ORW</i>	
	avg	std	avg	std
High power	1.92	0.02	1.13	0
Low power	2.65	0.04	1.13	0

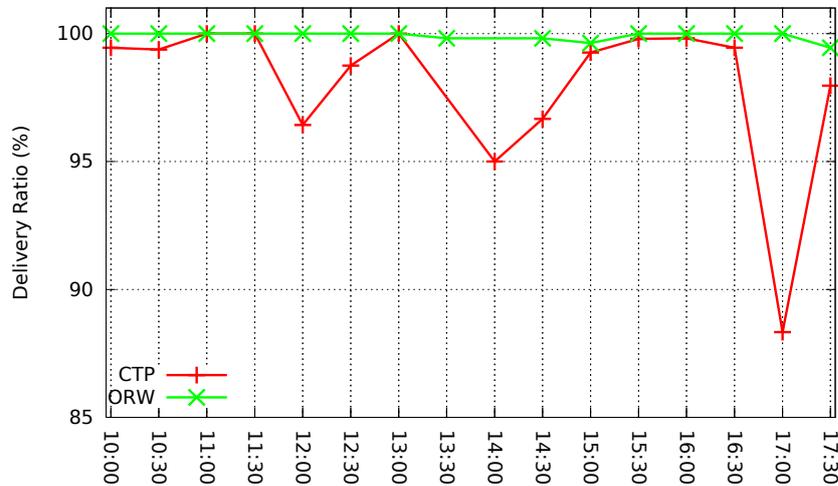
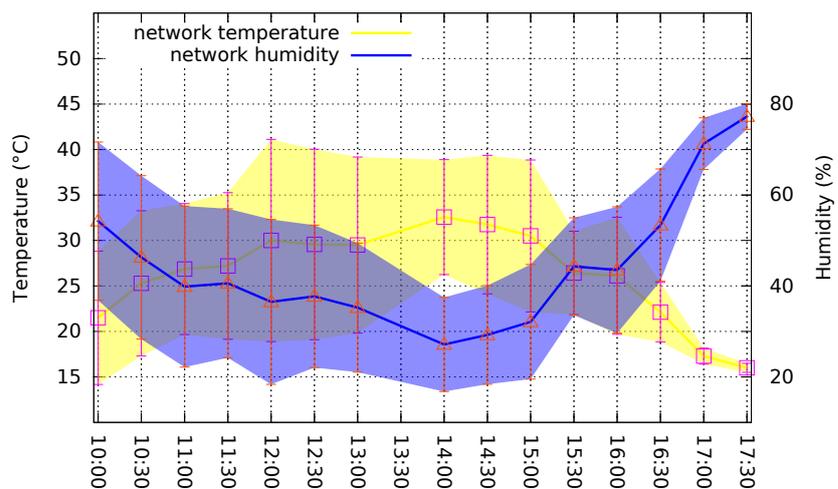
Figure 4.27: Delivery ratio for *CTP* and *ORW*.

Figure 4.28: Environmental conditions during the experiments.

The low count of retransmissions can be explained by the mechanisms implemented in *ORW*, when upon receiving a single ACK, the sender concludes that a single forwarder has been selected and does not initiate further retransmissions. The outliers observed during the *CTP* experiments can be explained by having nodes that lost the connectivity and rejoined the network later plus the fact that the protocol uses a very aggressive policy, retransmitting a packet up to 32 times, by default.

Finally, we report on the number of Layer 2 duplicates generated by the *ORW* network and the impact of the node failures throughout the course of the experiments as our results refute the ones reported by [41]. We report these numbers in two cases: *wf* (with failures) and *wof* (without failures). Table 4.12 shows that for increased node

failure rates, denoted by High power-wf and Low power-wf, in which the number of possible forwarders is reduced, the number of duplicates is smaller than in the absence of node failures. Moreover, the standard deviation shows that the number of duplicates

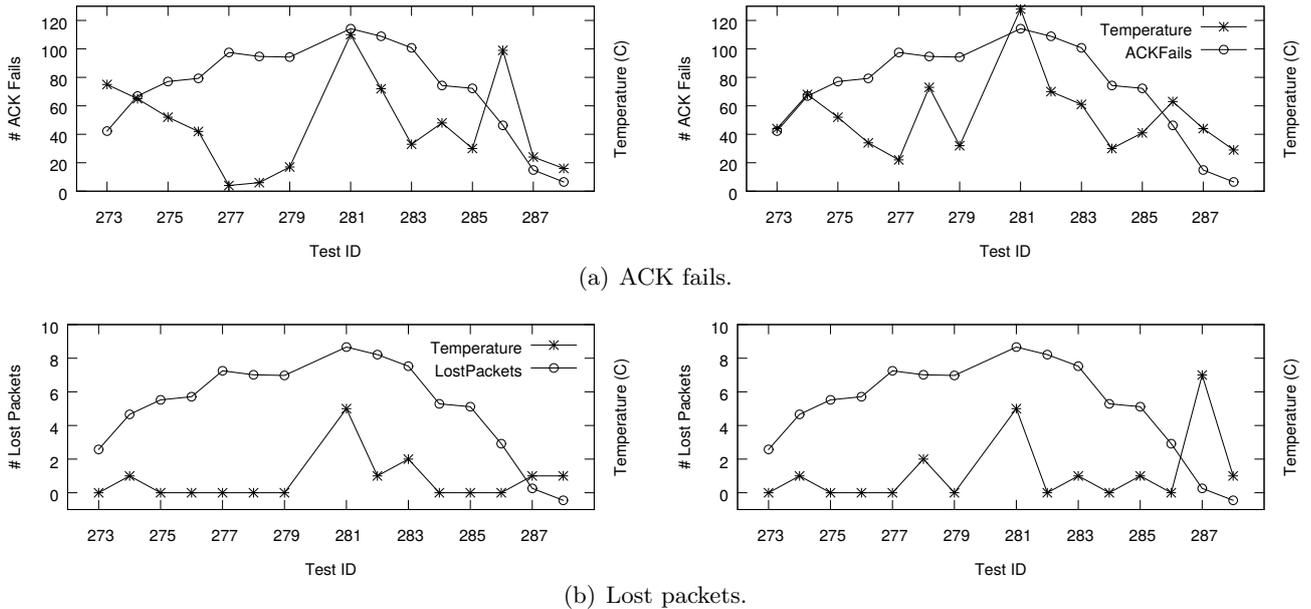


Figure 4.29: *CTP* ACK fails and lost packets variations with temperature, at node 5 (left) and node 6 (right).

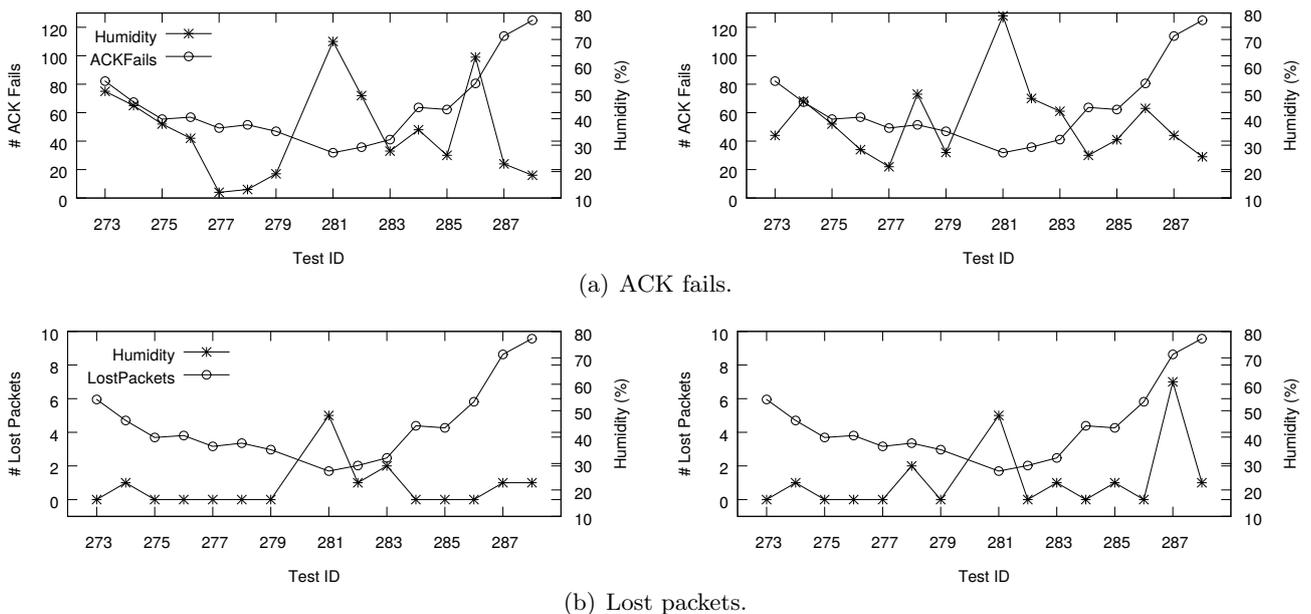
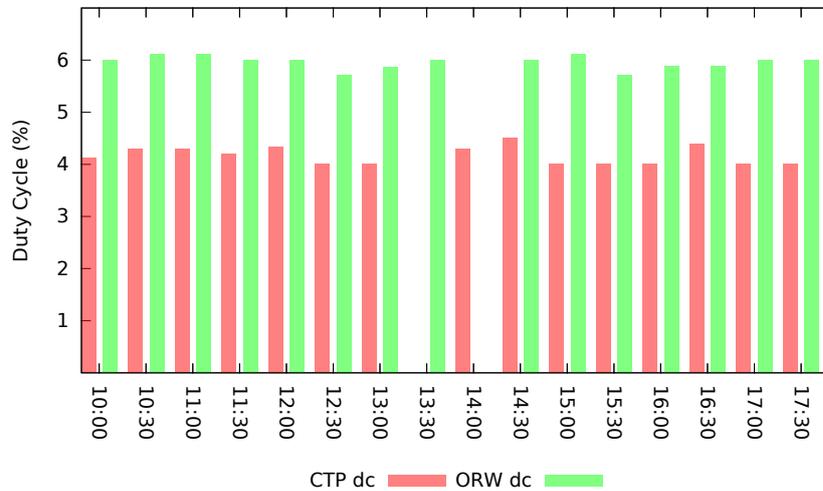
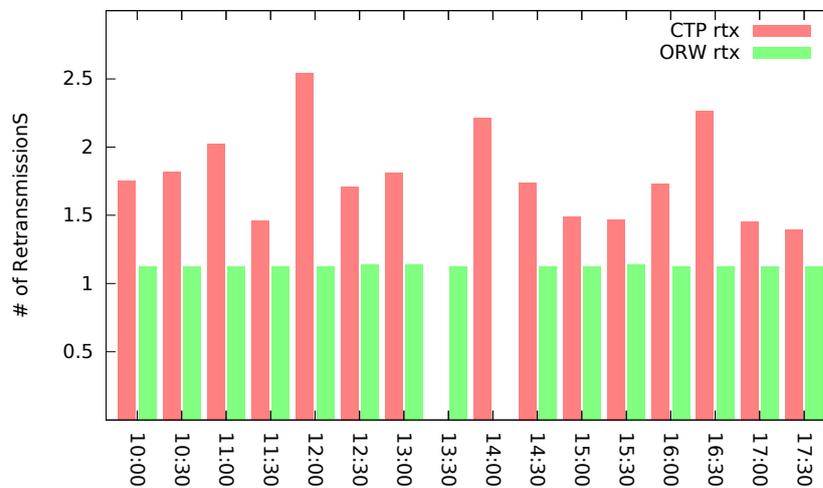


Figure 4.30: *CTP* ACK fails and lost packets variations with humidity, at node 5 (left) and node 6 (right).

Figure 4.31: Duty cycle of *CTP* and *ORW*.Figure 4.32: Retransmissions of *CTP* and *ORW*.

across all our experiments is stable around the average. For the high power experiments, *ORW* generates almost 13000 duplicates which translates into 3 duplicates per packet on each network node, while at low power, 2.5 duplicates are generated per packet and the reduction can be correlated with the reduction of the transmission range.

Table 4.12: *ORW* L2 duplicates

Power	Duplicates			
	min	max	avg	std
High power-wf	7054	15232	11982	187
High power-wof	10648	15232	12862	79
Low power-wf	3698	12566	9228	139
Low power-wof	8441	12566	10621	63

4.4 Application Layer

In this section, we investigate whether the trends we discussed in Section 4.2 bear an impact at the application layer. We consider our wildlife monitoring application that, as mentioned in Chapter 1, motivated our study. Here nodes are animal-borne, and therefore mobile.

The goal is to detect *contacts* among animals, viz. the mobile nodes they carry. Nodes determine whether they are in proximity of others through periodic beaconing. Time is discretized into epochs, of length 60 s in our case, which determine the temporal resolution of contact detection. Each node sends a beacon at the epoch start and then listens for beacons from other nodes. A contact begins at a node upon receiving the first beacon, and ends when a beacon from the same node is not received within a given, user-defined time interval. In our case, the latter is equal to the epoch length; missing a single beacon causes the corresponding contact to be closed. We consider a contact detected when recorded by at least one of the two nodes involved.

4.4.1 Test Execution

These tests were executed during our third campaign, as in Table 4.1. We deployed our application in the same sites described in Section 4.2, with the exception of OPEN which is the same as in Section 4.3. In the experiments we describe here, we wanted to assess the bias induced by the environment, therefore the application is run in a static topology, without the bias and complexity induced by mobility. We used the same CROSS topology described in Section 4.1.3, see Figure 4.3(a). However, in this case we “stretched” it to obtain bigger distances, as this was required to test the application functionality, which was our primary goal when performing the experiments. This resulted in a CROSS_EXTENDED topology with link distances up to 93 m, about 30% longer than in previous experiments.

The hardware we used is the one custom-made for our application, as described in

Section 4.1.2. The tests were performed only with high power. As in Section 4.3, we execute the real application: nodes are therefore not synchronized and collisions may occur.

4.4.2 Impact of Environment on the Application Layer

Figure 4.33 shows, for each environment, the number of detected and missed contacts. The chart is built by discretizing time into 1-minute intervals (i.e., the epoch duration). We count, for each pair of nodes, whether in a given interval they are part of a contact (detected) or not (missed). The chart confirms the overall trends we reported in previous sections. The percentage of missed contact is lowest in OPEN and higher in the two forests, with SPRUCE better than BEECH. However, while the trend is the same (OPEN is better than SPRUCE, which is better than BEECH), the performance difference between environments is quite different w.r.t. the physical layer. In Figure 4.5(a), the *PDR* decrease in going from one environment to the other was 15-20%. Here, the difference in missed contacts between OPEN and SPRUCE is about 10%, while the one between SPRUCE and BEECH is about 40%.

The reason is that the two experiments measure different things in different setups. In Section 4.2 we analyzed the (aggregate) *PDR*, measuring directly packet transmission in a controlled scenario where collisions are absent. Here, we look at application data (with its own semantics, only partially related to packet transmission) in a much less controlled scenario. As a concrete example: missing a single packet causes a very small difference in the aggregate *PDR*, but it may determine a contact as closed in the application considered.

In other words, and similarly to what we already mentioned in Section 4.3 about our data collection experiments, the observations about the physical layer in Section 4.2 in general cannot not be *directly* translated in quantitative terms to the application layer.

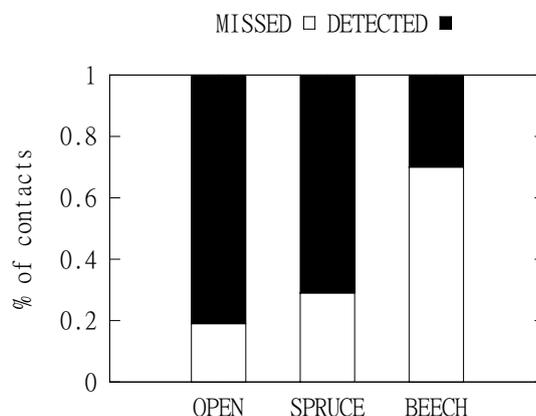


Figure 4.33: The effect of environment on contact detection.

However, there is clearly a relationship between the two, and knowledge of the quantitative tradeoffs in the former should inform design (if not prediction) of the latter, as we discuss in the next chapter.

4.5 Summary

In this chapter we presented an experimental study on the impact of environmental factors on outdoor WSN deployments. Motivated by a real-world application, we investigated, during six campaigns, the difference emerging when “immersing” the same WSN in different environments, characterized by varying degrees of vegetation, as well as seasonal and daily variations. We quantified the trends emerging at physical layer and show that: (i) this influence is mirrored at the routing and application layer, (ii) the macro-trends across environments allow to infer the relative trends in reliability, overhead and energy consumption at the routing layer. Whilst there are other empirical studies showing that the physical layer is affected by the environment, our study is the first that extends up to the routing layer, which bears a more direct impact on the application performance, and the application layer, the one directly relevant to the end user. To the best of our knowledge, this represents the first empirical study that characterizes, from a quantitative standpoint, the overall behavior of a WSN in different environments.

The observations we made in this chapter can directly inform deployment decisions, and possibly application- or network-level strategies, from a qualitative point of view.

In the next chapter, we build on this analysis and exploit the set of data traces to create models to better estimate the performance of a WSN in its target deployment scenario.

Chapter 5

Estimating and Reproducing: Models

Essentially, all models are wrong,
but some are useful.

George E. P. Box

Our experimental campaigns and data traces analysis in Chapter 4 show that the link quality of a WSN depends on the characteristics of the surrounding environment. In this chapter we build on this analysis and exploit the set of data traces to create two models with distinct goals: (i) estimating the link quality at run-time, and (ii) for reproducing realistic network conditions in simulators. First, in Section 5.1, based on our empirical evidence that the environmental factors have a strong impact on the low-power wireless links, we focus on two factors: temperature and humidity. We study the impact of these two factors on the physical layer parameters in the OPEN field environment and then conduct a study on the impact of temperature and humidity on the *RSSI* and *PDR* of TMote Sky and Wasp mote platform in controlled settings. Then, we develop a model describing the influence of temperature and humidity on the link quality that helps estimating the link quality at run-time considering the particular temperature/humidity profile of the target environment. Second, in Section 5.2, based on the observations that the link quality distribution of WSN in link quality classes (i.e., dead, poor, intermediate, good, perfect) follows a specific pattern in each of our studied environment, we capture the pattern in a model. Then we train the model with our experimental traces and integrate it in mainstream simulators. The model is key for efficiently reproducing realistic network conditions for large-scale simulations of long-term behavior of protocols/applications by accounting for the influence of the environment on the network beforehand.

5.1 Estimating the Impact of Environmental Parameters

Our analysis in Section 4.2.6 shows variations induced by the interleaving of day and night, which affect parameters such as temperature and humidity, which in turn affect link quality. Our goal is to model and estimate the impact of these factors on the performance of the WSNs. As in the forests is very difficult to separate the effects on link quality of the environmental parameters factors (e.g., temperature and humidity) from the effects of vegetation (e.g, tree trunks, foliage), we fix the environment to the OPEN field. Since our OPEN field deployment sites are at a remote location in the mountains, our data traces were collected in an interference-free environment. Here we measured the hardware-based link quality metrics, namely *RSSI*, *noise* floor and *LQI*, and the *PDR*, during the PHY layer OPEN field campaigns from Table 4.1, with a focus on correlations between link measurements and temperature and humidity, for the TMote Sky platform.

One year later, we extended our study to a second hardware platform, namely Waspote, and analyzed the behavior of the two (i.e., TMote Sky and Waspote) under the same environmental conditions, albeit using a small-scale setup in an OPEN field during winter. As our outdoor experiments were performed on a relatively short range of temperature and humidity, we conducted experiments with both platforms, in a sauna environment that allowed us to test and control a wide range of temperature and humidity in a fine-grained way.

During our outdoor experiments, apart from acquiring temperature and humidity from on-board sensors reflecting the conditions directly experienced by the nodes in the box, we used two other sources of meteorological data. First, a meteo station [71] located 200 m from OPEN, which provides temperature, relative humidity and precipitation measurements. Each of these values were sampled every 15 minutes. Second, two LAS-CAR EL-USB-2+ [44] temperature, relative humidity and dew point data loggers with readings from 0 to 100%RH and -35 to +80°C. Each of these values are sampled every 5 minutes. Thus, the data loggers provide measurements with higher temporal resolution than the available meteo station, and since they are collocated with our sensor networks, the measurements accurately reflect the meteorological conditions experienced by our networks.

5.1.1 Experimental Setup

Open field experiments with TMote Sky platform. Data traces were collected during the first and second campaign during summer and winter using the 8 nodes CROSS topology and during the fourth and fifth campaign using the 16 nodes GRID topology. The experimental setup is described in Section 4.1. The in-field collection was performed using TRIDENT, and the aggregates of the link metrics and environmental parameters

over 30 minute rounds were collected during 2 days experiments on CROSS and 7 days on GRID. 250,314 data points were collected overall.

Open field experiments with different platforms. The data traces were gathered in an experimental campaign performed during the winter (March 9 - March 11) of 2014. A single link between two nodes communicating at 0 dBm and placed at 40 m apart in an OPEN field was observed for a 2-day period. The same setup was used for a link between a pair of TMote Sky and Wasmote nodes. The two links were observed in exactly the same conditions, by placing the two pairs of nodes side-by-side on different channels (i.e., 26 and 17). All nodes were configured using TRIDENT, and experiments were divided in 30 minute rounds like in our previous campaigns. All the raw packets were collected, for both platforms, along with the per-round statistics. Overall, the results of 96 rounds of 30 minute and 138,240 data points were collected.

Controlled experiments in sauna. Using a sauna environment and a setup with two pairs of two TMote Sky and Wasmote nodes at 3 m apart communicating at 0 dBm, we were able to study the impact of temperature on *RSSI* and *noise* floor in a very fine-grained way. During the experiments, we exposed the link to four cycles. During the first cycle, the environment was dry with a humidity of 32%RH, and with a constant temperature of 24°C. Then, during the second cycle, we slowly increased the temperature in an almost linear fashion up to 55°C, followed by a third cycle with an increase in humidity to 70%RH. In the fourth cycle we tried to keep the temperature constant and vary the humidity up and down with 5 to 15%RH. The collection of connectivity traces was performed using TRIDENT, and our experimental campaign was composed of four 60 minute rounds, corresponding to the four cycles described above, in which each node sent 7200 packets at a data rate of 1 packet every 250 ms. Low power (-8 dBm) and channel 26 and 17 were used. Each node collected all the packets and temperature and humidity were sampled every minute. Overall, the results of 4 rounds of 60 minute were collected, and a total of 288,000 data points were collected.

5.1.2 Observations

The objective of the experiments described above is to understand if and how the environmental conditions, both in terms of temperature and humidity, affect the low-power wireless links of a WSN, from the standpoint of the physical layer.

5.1.2.1 Impact of Temperature on Link Quality

RSSI and noise floor. We begin our analysis with one of the most common indicator of link quality: *RSSI*. Figure 5.1 shows the results from our 7 days experiment on GRID

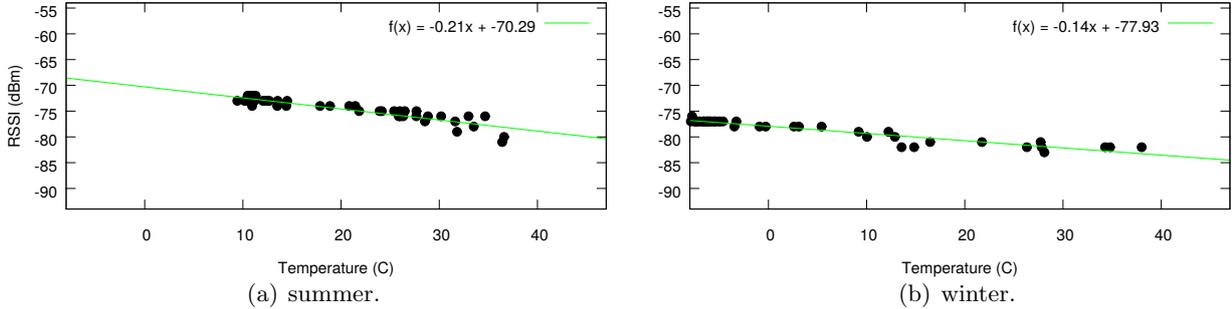


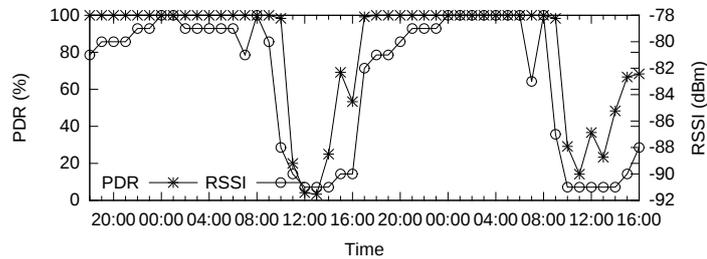
Figure 5.1: RSSI and temperature relation.

topology, for a link of 40 m, at power -1 dBm, during summer and winter, using the same pair of nodes. For both seasons, it is clearly visible that the relationship between *RSSI* and temperature can be approximated as a linear function, our observations being in accordance with [18]. Using linear regression analysis we observe that all 240 links of the GRID topology follow the same linear trend over the entire experiments duration. Table 5.1 shows the complete statistics (minimum, maximum, average and standard deviation values) of the slopes for the linear *RSSI*-temperature function, for both seasons and powers. Based on this, there are a couple of observations we can make. Independent of the season and power, there is a negative correlation between the *RSSI* and temperature, given the negative sign of the slope. Also, links at high power show smaller slopes than the links at low power. This is intuitive as high power links also succeed to maintain higher *PDR* values as shown in Section 4.2.2, having more energy to compensate for the negative effect of temperature. Moreover, the decrease of the *RSSI* with temperature is faster in summer than during winter, suggesting that during winter the link is “helped” by other factors (i.e., humidity).

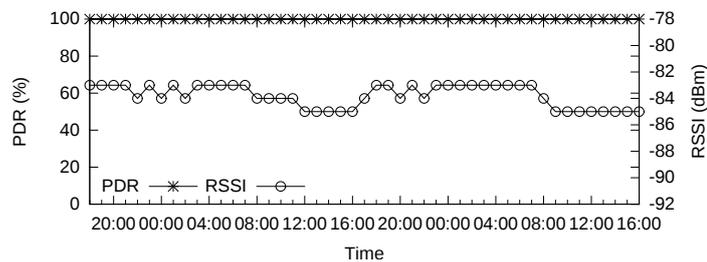
We now turn our attention to the traces collected in the OPEN field during winter for the two platforms, and investigate if the correlation between *RSSI* and temperature holds across platforms. An example of how the *RSSI* of the two considered links vary over time, and the corresponding temperature variation (computed as combined temperature of sender and receiver as in [18]) inside the plastic boxes holding the motes, can be seen

Season	Power	min	max	avg	stddev
Summer	High power (-1 dBm)	-0.062	-0.19	-0.124	0.038
Summer	Low power (-8 dBm)	-0.072	-0.23	-0.157	0.052
Winter	High power (-1 dBm)	-0.012	-0.17	-0.11	0.041
Winter	Low power (-8 dBm)	-0.03	-0.21	-0.13	0.043

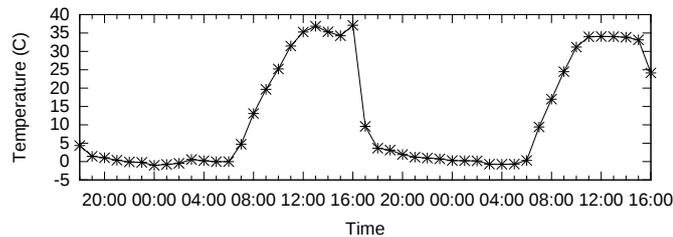
Table 5.1: RSSI-temperature linear function slopes on GRID.



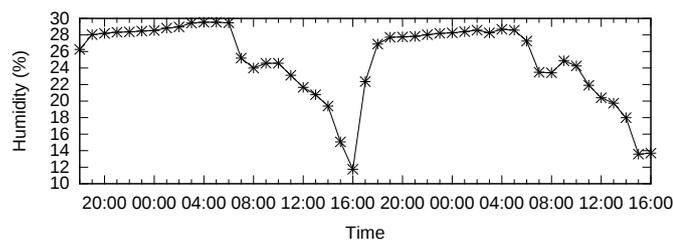
(a) TMote PDR and RSSI.



(b) Wasp mote.



(c) In box temperature.



(d) In box humidity.

Figure 5.2: RSSI and PDR of TMote Sky and Wasp mote links over a 2-day time period.

in Figure 5.2. This shows a degradation of 13 dBm in *RSSI* for the TMote Sky, see Figure 5.2(a), for a temperature increase of 40°C in the box (with a mere 11°C increase outside it). This is enough to turn a perfect link ($PDR = 100\%$) into a dead one ($PDR = 0\%$). On the other hand, Figure 5.2(b) shows that, for Wasp mote, temperature is also negatively correlated with *RSSI*, but to a much lesser extent w.r.t. TMote Sky, showing a degradation of 2 dBm and allowing the *PDR* to remain at 100%. Although both platforms

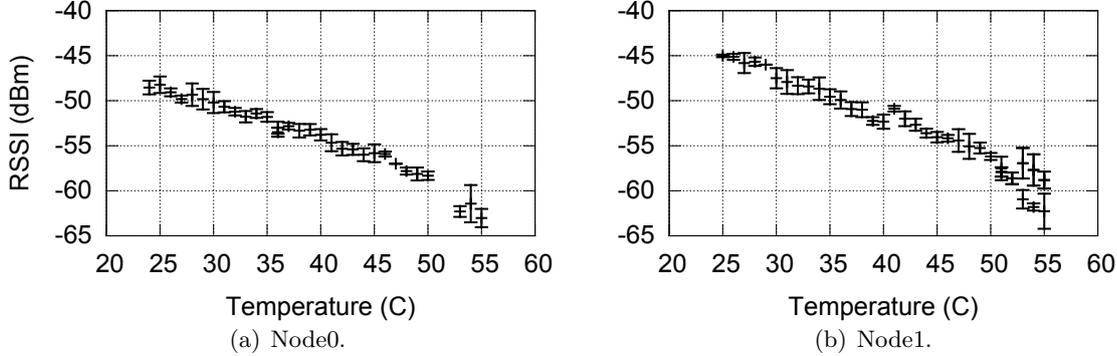


Figure 5.3: TMote Sky RSSI-temperature relation in sauna.

are based on radio chip compliant with IEEE 802.15.4, their behavior is very different. These sharp differences reassert that case-by-case quantitative assessment and analysis with support tools (e.g., TRIDENT) is key.

Moreover, the trend observed in our OPEN field deployments showing that the *RSSI* decreases approximatively linear with the temperature holds for our controlled setup in sauna. Figure 5.3 shows the relation between *RSSI* and temperature and the stddev *RSSI* for each temperature point. A 30°C temperature increase in the sauna caused a 12 dBm decrease in *RSSI* of Node0 and 14 dBm for Node1, respectively. Using linear regression analysis we observed a steeper decrease of *RSSI* with the temperature than in the outdoor OPEN field, with a slope of -0.44. The data collected during the sauna experiments allowed us to further investigate the importance of the temperature parameter and how it correlates with the *RSSI*. To measure the correlation, we used Spearman’s rank correlation [68], computed as a score between -1 and +1 where 0 indicates no correlation at all. It measures how well two variables monotonically increase/decrease in relation to one another. It does this by computing the linear dependence of the ranked variables as opposed to the variables values themselves. Using Spearman’s rank correlation, we computed a strong negative correlation between the two variables of -0.85, by taking into consideration the *RSSI* values and the temperature measured at a fixed humidity value.

On the other hand, looking at the *RSSI* relation with the temperature for the Wasp-mote platform, we observed a different behavior. Figure 5.4 shows clearly that there are steps of 1 dBm in the relationship between *RSSI* and temperature that correspond, in our environment, to an increase of 7°C in temperature.

Our experiments in the outdoor OPEN field and the sauna controlled setup have clearly shown that: the *RSSI* decreases with an increase in the temperature; the decrease is not consistent in the same environment across seasons; and that the decrease is not consistent among the two platforms studied. Which again means that: in-field traces are needed

and our TRIDENT tool supports this step.

Also in the case of *noise* floor, higher temperature decrease the *noise* floor and the relationship with the temperature is approximately linear. This effect was reported also by [7, 18] and explained by Bannister [7] to be due to the losses in the signal amplifier. Unlike for *RSSI* we observed similar slopes among different nodes, but with a less pronounced decrease compared to *RSSI*. The average computed slope is -0.057 with a standard deviation of 0.008.

LQI. Using the same methodology as for *RSSI*, we analyze the relation between *LQI* and temperature.

Figure 5.5 and Figure 5.6 illustrate the impact of the temperature on *LQI* of the same set of representative links from the GRID topology, during summer and winter. The *LQI* decreases with the increase in the temperature for both seasons, but it is only during summer when the *LQI* decreases in an approximately linear fashion with the temperature, during winter visible non-linearities can be observed. Even more, during summer, across the slopes of the linear functions describing the *LQI*-temperature relation are different among good, perfect and intermediate quality links, as defined in Section 4.2.5, with the latter exhibiting bigger slope values, on average -0.7, compared to -0.12 for the former, see Table 5.2. *LQI* small variation for high quality links, even exposed to 40°C fluctuations in temperature, confirms the observations from [?, 69].

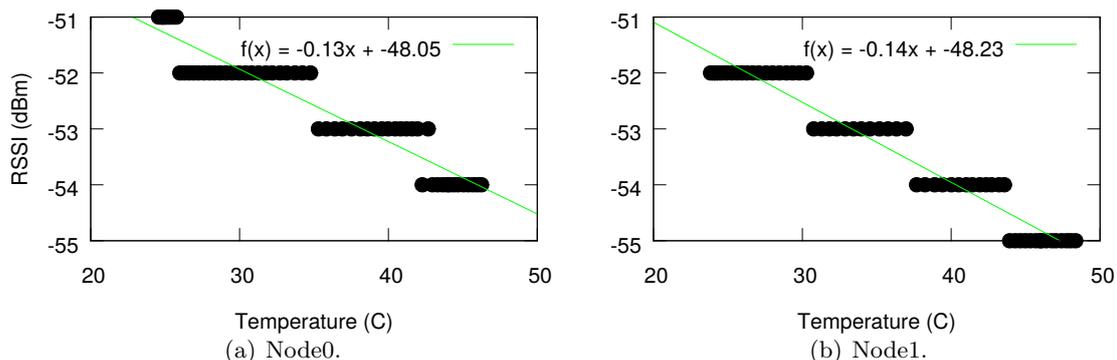


Figure 5.4: Waspnote RSSI-temperature relation in sauna.

Power	Link quality	min	max	avg	stddev
High power (-1 dBm)	Good/perfect	-0.02	-0.10	-0.07	0.003
High power (-1 dBm)	Intermediate	-0.23	-0.41	-0.33	0.017
Low power (-8 dBm)	Good/perfect	-0.05	-0.18	-0.12	0.016
Low power (-8 dBm)	Intermediate	-0.52	-0.76	-0.70	0.021

Table 5.2: LQI-temperature linear function slopes on summer GRID.

Furthermore, if we look at the single link of 40 m between the two nodes measured at 0 dBm, for the TMote Sky platform, Figure 5.7 shows that, a 40°C temperature increase in the box causes an excursion in LQI from a very high value (i.e., 106) to a very low one (i.e., 62). As the XBee radio module does not report the LQI , insights about LQI variations with the temperature for the Wasmote platform cannot be presented.

Our experiments in the sauna controlled setup confirm that for high quality links, LQI variation with the temperature is very low, as can be seen in Figure 5.8. The bidirectional link experiences a very small degradation of LQI , from 108 to 106, for a temperature excursion of 30°C.

PDR. We continue our analysis with the roughest indicator of link quality, yet the most directly informative: the PDR . We already shown two examples of representative links from our OPEN field campaigns, one of 39 m from the OPEN field summer at low power, in Figure 4.12, and another of 40 m from OPEN winter at high power, in Figure 5.2(a). For both links the PDR is perfect during night and drops significantly, as low as 40% and 5% respectively, during the hot hours of the day.

In accordance with our previous analysis, we next look at how PDR related with the temperature. Figure 5.9 depicts the impact of the temperature on PDR , for a representative link of 40 m, during winter and summer, at low power, in the OPEN field. During both seasons in OPEN field, we observe three regions: *constant* the PDR of the link does not change with the changes in the temperature; *linear regression* the PDR decreases in an approximately linear fashion with the temperature and the decrease starts at the same

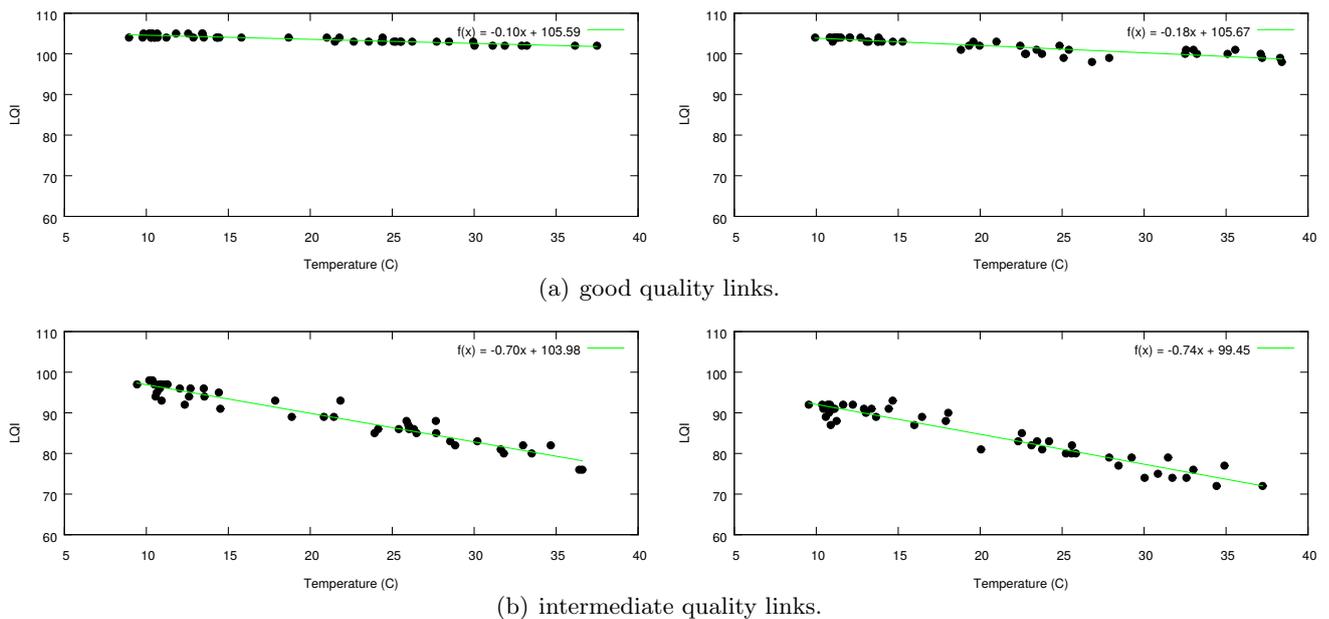


Figure 5.5: Relationship between LQI and temperature during summer.

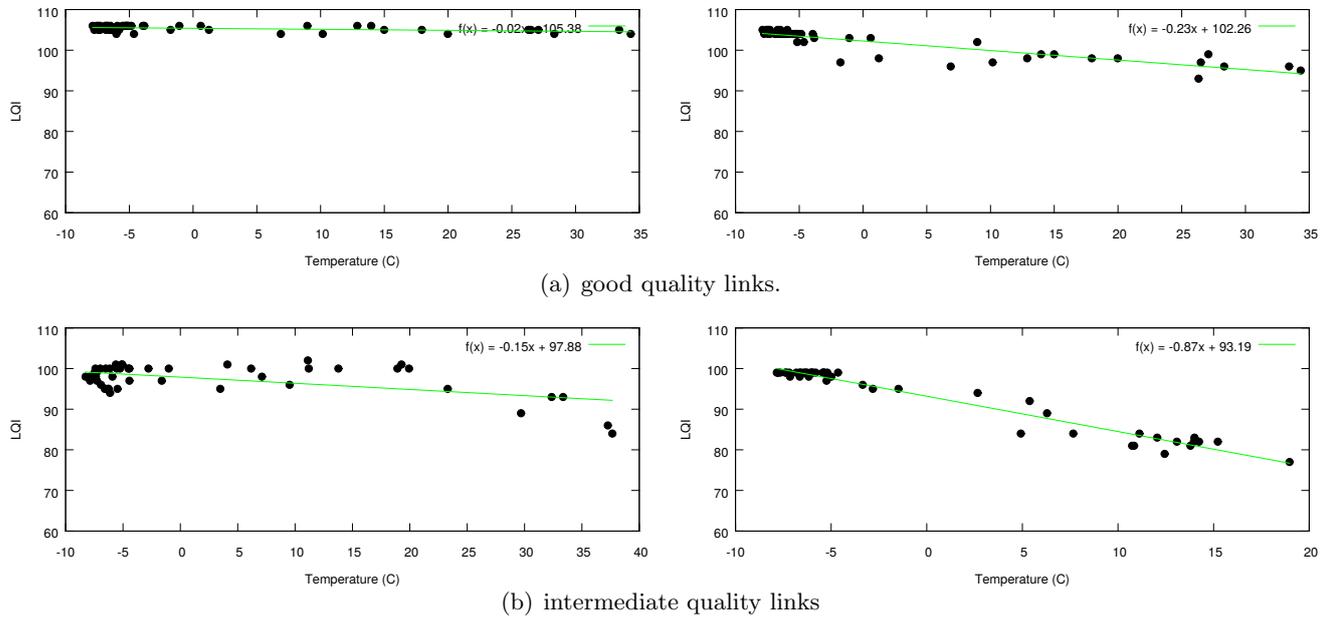


Figure 5.6: Relationship between LQI and temperature during winter.

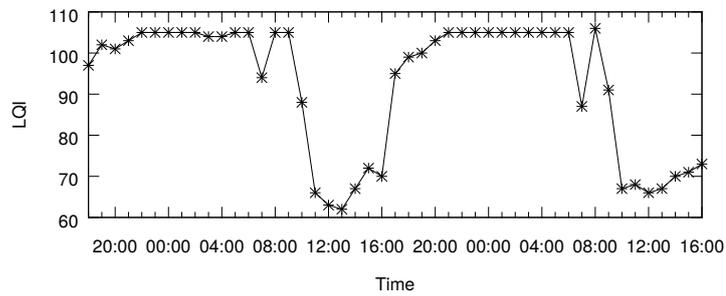


Figure 5.7: LQI measured by TMote Sky over a 2-day time period.

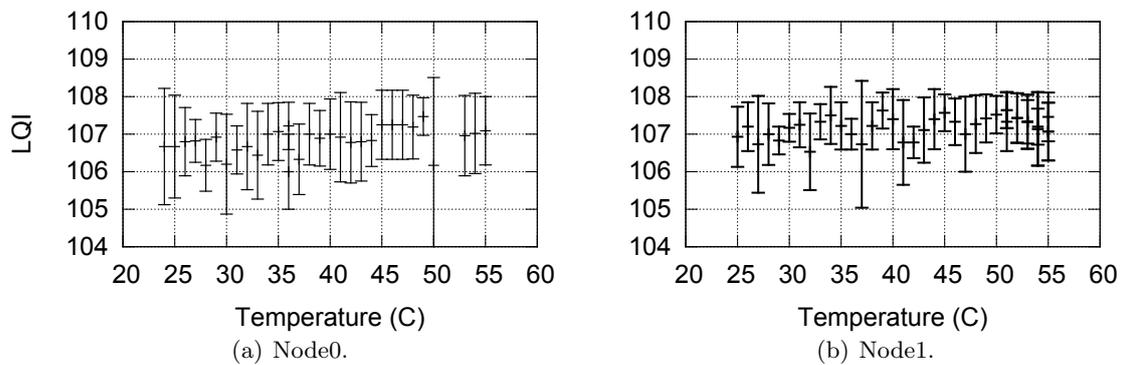
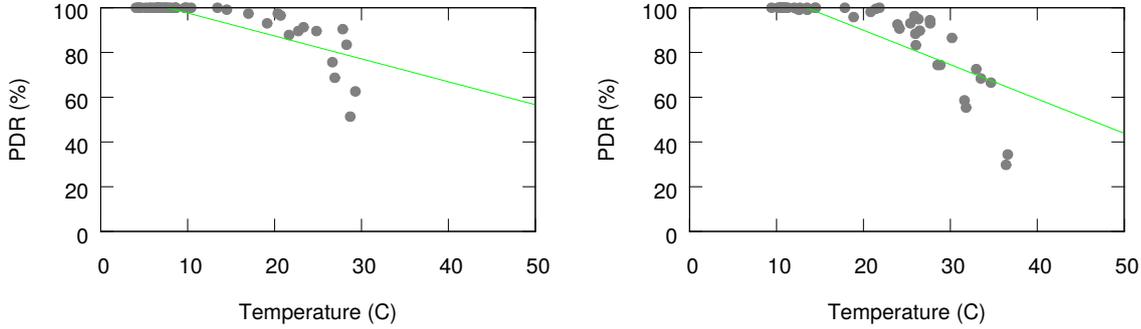


Figure 5.8: LQI -temperature in sauna.



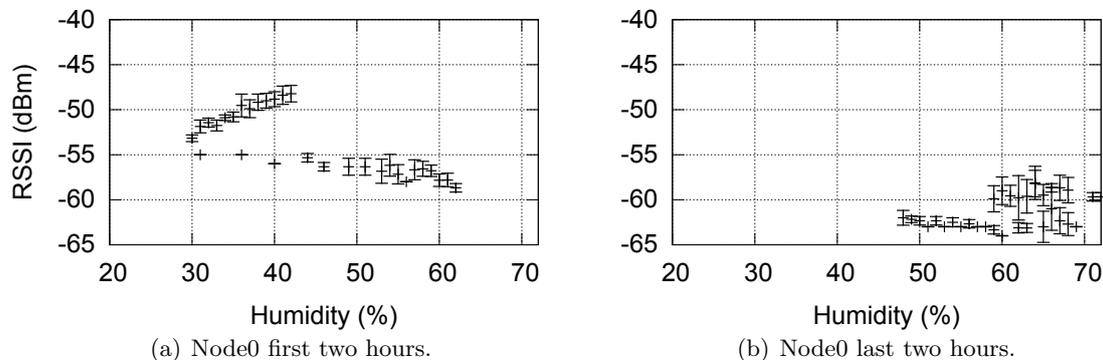
point with the decrease in $RSSI$; *no relation*: we cannot define a linear function anymore. Using the 240 links of the GRID topology we identified the two temperature thresholds that define the *linear regression* region: $T_{start} = 15 - 18^{\circ}C$ and $T_{end} = 25 - 28^{\circ}C$. It is worth noting that, for the low power links, while running the linear regression analysis we identified the slopes of the PDR -temperature relation to approximate the ones of the $RSSI$ -temperature.

In the sauna controlled setup, even if the link had a $RSSI$ decrease of 15 dBm, from -45 dBm to -60 dBm and the LQI varied, the link was perfect, delivering 100% of the packets.

Figure 5.9: PDR-temperature variations during winter(left) and summer (right).

5.1.2.2 Impact of Humidity on Link Quality

We now turn our attention to the variations of link quality induced by humidity. We begin by looking at the link in Figure 5.2 from Section 5.1.2.1, which shows a degradation of PDR and $RSSI$ with the increase of temperature corresponding to a decrease of humidity. Looking only at Figure 5.2(a) and Figure 5.2(d) one might say that there is a strong positive correlation between link quality and humidity. But, this is likely explained by the fact that we measure the relative humidity which is directly dependent on temperature and these two parameters correlate to a high degree. Therefore, in order to decouple the impact of the two parameters, temperature and humidity, we use the data from the sauna experiments. As during our experiments in the sauna we could not produce the same linear behavior for humidity as for temperature, we run the analysis separately for the first and the last two hours of experiments respectively, shown in Figure 5.10. Here we see a positive correlation of the $RSSI$ with the humidity to a threshold, identified at 42%RH, after which humidity starts to negatively affect the $RSSI$. The outliers in the interval 26 to 42% humidity correspond to the round of the experiment when the temperature was increasing and humidity decreasing, thus the impact of the two factors keeping the $RSSI$ at an almost constant value. Moreover, two other observations are worth mentioning: the

Figure 5.10: *RSSI*-humidity in sauna.

increase of the *RSSI* with the humidity has a much higher slope than the decrease of *RSSI* with humidity, in both cases the differences between the minimum and maximum humidity value is 17%, but from 25% to 40%RH the *RSSI* increases with 10 dBm, while for an increase in humidity from 42% to 62%RH the *RSSI* has a decreasing step of only 5 dBm.

Similarly as for temperature, we look at the *RSSI* and humidity value for specific temperature values and observe a Spearman correlation of +0.67 for humidity values lower than 42% and -0.33 for higher values. In fact, this might explain the milder decrease of *RSSI* with the temperature in winter, when the humidity helps the *RSSI*.

5.1.3 Models

In this section we assess to which extent the first-order model proposed by Boano et al. [18] matches the real-world data traces we collected during our OPEN field campaigns in summer and winter and propose a model to estimate the link quality at run-time considering the particular temperature/humidity profile of the target environment.

5.1.3.1 Existing Model of the Effect of Temperature on Link Quality

The first-order model proposed by Boano et al. [18] computes the effect of temperature on the signal to noise ratio, taking into consideration the effect of the temperature on the signal strength of the transmitted power, received power and noise floor, using the following equation:

$$SNR = P_t - PL - P_n - \alpha \Delta T_t - (\beta - \gamma) \Delta T_r - 10 \log \left(1 - \frac{\Delta T_r}{T_r} \right)$$

denoting PL as the path loss between a transmitter-receiver pair, P_t as the transmis-

sion power, and P_n as the noise floor at the receiver. α , β , γ are the slopes, with units dB/K , of the linear trends observed, platform dependent, and obtained through empirical study on the TempLab testbed [11] with values: $\alpha = 0.065$, $\beta = 0.088$ and $\gamma = 0.037$. T_t and T_r are the temperature in Kelvin at the transmitter and receiver.

First, the accuracy of the proposed model depends on identifying the right values of the parameters that are considered platform dependent. Authors proposed a set of values for the slopes α , β , γ computed in their testbed. Our empirical studies show that not only that the values of the slopes are platform dependent but also that their values depend on the season, therefore we cannot use the same values for both seasons.

Second, we tested the accuracy of the model by feeding the model with the data traces collected from our OPEN field experiments, using the slopes for TMote Sky, computed by the authors. We started with the simple computation of the attenuation of the signal in our environment for the link used as example in Section 5.1.2.1. In the real-world, the link was exposed to a difference in temperature of 40°C and had an attenuation of 6 dBm. Using the proposed model, the computed attenuation for the link is 4.64 dBm. This translates in an underestimation of the attenuation of the signal of 22.66%. As a next step, we fed the model with all the data traces collected from the OPEN field and computed the deviation of the model generated values with respect to the in-field measured values, for each of the 240 links in the GRID topology, see Table 5.4. During summer the model underestimates the attenuation of the signal by 12.6% on average and a stddev of 3.1%. During winter the model underestimates the attenuation of the signal with 19.3% on average and a stddev of 4.5%. Translated in *RSSI* values, the differences between the estimated attenuation values and the real attenuation experienced in-field by the link correspond to a difference between a perfect and a dead link.

5.1.3.2 A Model of the Effect of Temperature and Humidity on Link Quality

We build on top of our empirical observations and propose a model for low-power wireless links. Our goal is to develop a simple model to estimate the attenuation of the link quality under environmental settings with temperature and humidity variations. As shown in our empirical experiments, temperature has a high impact on the *RSSI* and a lower impact on the *noise* floor, decreasing both of them and the decrease can be approximated with a linear function. Humidity has proven to have an impact on the *RSSI* and noise floor, increasing both of them to the extent of empirically determined 42%RH, in our environments.

We focus on modeling the estimation of attenuation of the signal for TMote Sky platform as the observations from Section 5.1.2 showed different behaviors among platforms and because the Wasmote platform does not allow the collection of *noise* floor measure-

ments.

Modeling approach. The value of the attenuation represents the difference of the *RSSI* and the *noise* floor. Therefore, in our model, we can convert the values into an indicator computed by subtracting the *noise* floor from the *RSSI* and defined it as a function of temperature and humidity of the target environment as follows:

$$RSSI\textit{Attenuation} = \begin{cases} (\alpha - \beta)\Delta T - (\gamma - \theta)\Delta H & \text{if } H < 42\% \\ (\alpha - \beta)\Delta T + (\gamma - \theta)\Delta H & \text{if } H \geq 42\% \end{cases}$$

Parameters ΔT and ΔH characterize the environment in which the network is deployed. ΔT represents the difference between the maximum and minimum temperature, same as ΔT_r in Boano et al. model [18], while ΔH represents the difference between the maximum and minimum humidity, to which the network is exposed. In case the humidity of the environment is under 42% the formula can be used with the sign $-$, the humidity compensating for the attenuation induced by the temperature. In case the humidity of the environment exceeds 42% the formula can be used with the sign $+$, the increased value of humidity helping the increase of the attenuation. Parameters α and γ are the slopes of the *RSSI*-temperature and *RSSI*-humidity linear function, while, β and θ are the slopes of the linear trends observed *noise* floor with respect to temperature and humidity. As we already shown, these parameters are platform dependent and their values must be computed using in-field data traces. Nevertheless, using the model with both signs can give an upper and lower bound of the attenuation in the target environment.

Evaluation of the model. To evaluate the performance of our approach, we used the model to estimate the attenuation of the signal for each link from our OPEN field GRID topology, during summer and winter. We have the temperature and humidity values to which the network was exposed and the in-field computed slopes (i.e., α , γ , β , θ) for the linear functions of *RSSI* and *noise* floor with respect to temperature and humidity for each link. Then, we compared these results against the in-field measured ground truth and against the first-order model presented in Section 5.1.3.1. Table 5.3 shows a comparison between the estimated values using our model and the real values for the attenuation, in terms of minimum, maximum, average values, for both seasons and powers, high (HP) and low (LP). Table 5.4 shows the comparison for the model in Section 5.1.3.1 using the slopes $\alpha = 0.065$, $\beta = 0.088$ and $\gamma = 0.037$.

Our results show that the accuracy we obtain with our approach is significantly superior to existing approaches [18] and reiterates the fact that the most accurate estimates are obtained using in-field data traces from the target environments.

Season	Power	min	max	avg
Summer	HP (-1dBm)	2.5%	4.2%	3.5%
Summer	LP (-8dBm)	2.9%	4.8%	4.1%
Winter	HP (-1dBm)	3.3%	4.5%	3.9%
Winter	LP (-8dBm)	3.8%	5.1%	4.3%

Table 5.3: Our proposed model.

Season	Power	min	max	avg
Summer	HP (-1dBm)	5.1%	7.3%	6.5%
Summer	LP (-8dBm)	9.2%	14.5%	12.6%
Winter	HP (-1dBm)	5.8%	8.1%	7.2%
Winter	LP (-8dBm)	15.05%	22.87%	19.3%

Table 5.4: Boano et. al [18] model.

5.2 Reproducing Realistic Network Conditions

WSNs protocols and applications are often evaluated through simulations that make simplified assumptions about the link layer. As demonstrated by several empirical studies, presented in Section 2.3.2, the real characteristics of low-power wireless links differ greatly from those used in nowadays models. Moreover, low-power wireless links are often modeled in the abstract, using very general models that often do not capture the characteristics of target application environments and do not provide the ability to tailor the model to the target environment. Thus, the significant differences between the models used in simulators and the real behavior leads to erroneous performance evaluation of upper-layer protocols (routing layer and above). To alter this, we must improve our simulators by incorporating models that would help reproducing realistic network conditions and realistic link layer characteristics.

Based on the observations in Section 4.2.5 that the link quality distribution of WSN in link quality classes (e.g., dead, poor, intermediate, good, perfect) follows a specific pattern in each of our studied environment, across years, seasons and topologies, our approach is to explore using probabilistic models to recreate behavior that is representative of what is observed in our outdoor real-world network. Thus, we capture the pattern in a model that we train with our experimental data traces, collected during the campaigns presented in Chapter 4, and integrate it in mainstream simulators. Using this model, the simulator can generate link quality distributions into classes and their variations across classes with similar quality for a specific combination of environment and season. To evaluate our approach and model we reproduce “synthetically” our summer and winter experiments on the GRID topology in the simulator. Then, we assess the connectivity characteristics of the simulated network and links, running TRIDENT with the same configuration parameters as in our in-field experiments. We compute the link quality distributions in classes and the number of transitions and compare against the in-field distributions and transitions. In this regards, we prove that the model is key for efficiently reproducing realistic network conditions for simulations of long-term behavior of protocols/applications by accounting for the influence of the environment on the network beforehand.

5.2.1 Goals

In this section, we present the issues that need to be addressed when modeling link quality distribution patterns and the variations across classes. We use *PDR* to characterize the link quality and exploit it to get insights on the *fraction* of links characterized by a given quality. In this respect, we use the same link classification as in Section 4.2.5. Considering all 30-minute rounds of our experiments, we show in Table 5.5 the link quality distributions for day and night in OPEN field on the GRID topology. The table clearly shows that the fraction of dead and poor links is quite stable between night and day in both seasons, while several links move from intermediate to good during summer and from good to perfect during winter. Moreover, we showed in Section 4.2.9 that the patterns are almost the same across different years and topologies, CROSS and GRID, with a small percentage of links migrating from one class to the adjacent one from one year to another.

A closer look at our experimental data, see Figure 5.2 shows long term dynamics with rounds of nearly constant *PDR* of the links which persist during the night or during the day. Moreover, in Section 4.2.8 we showed in Table 4.4 that the summer creates more transitions between link quality classes than winter and that their number increases as we progress from OPEN to SPRUCE and BEECH with the presence of vegetation and the increase of its density. During summer, in 24 hours, a link experiences on average 3 transitions, while during winter it experiences 5 transitions.

In order to realistically simulate the behavior of the links, a model that is flexible enough to replicate the observed link quality distributions in classes is required. Moreover, the parameters of the model should be estimated from the available data traces.

5.2.2 Theoretical Model

The core idea of our model is to capture and describe the probability for the link quality distribution in classes to transit from a state (i.e., link quality distribution in classes during a 30 minute round) to another. For this we propose to model network’s transitions from one state to another using a *hidden Markov model* (HMM).

The theoretical details of our model are described next. We denote a set of *states*:

Table 5.5: Day vs. night *PDR* distributions example.

Season	Time	DEAD	POOR	INT	GOOD	PERFECT
Summer	Day	9.52%	19.05%	16.67%	14.29%	40.48%
Summer	Night	7.14%	19.05%	7.14%	28.57%	38.10%
Winter	Day	23.21%	12.50%	19.64%	21.43%	23.21%
Winter	Night	19.64%	14.29%	14.29%	12.50%	39.29%

$\{s_1, s_2, \dots, s_n\}$. The process moves from one state to another generating sequences of states: $\{s_{i1}, s_{i2}, \dots, s_{ik}, \dots\}$. The Markov chain property, the probability of each subsequent state depends only on what was the previous state is: $P(s_{ik} | s_{i1}, s_{i2}, \dots, s_{ik}) = P(s_{ik} | s_{ik-1})$. The *states* are not visible but each state randomly generates one of M *observations* (or *visible states*): $\{v_1, v_2, \dots, v_m\}$.

To define our hidden Markov model the following probabilities have to be specified:

- matrix of *transition probabilities* $A = (a_{ij})$, $a_{ij} = P(s_i | s_j)$;
- matrix of *observation* (emission) *probabilities* $B = (b_i(v_m))$, $b_i(v_m) = P(v_m | s_i)$;
- and a vector of *initial probabilities* $\pi = (\pi_i)$, $\pi_i = P(s_i)$.

In the end, the behavior of our model is going to be represented by: $M = (A, B, \pi)$.

In Figure 5.11 we represent a hidden Markov model with two states:

- two *states*: s_1 and s_2 , which correspond to two different rounds from the OPEN summer day link quality distribution in classes;
- five *observations*, which correspond to our link quality classes (dead, poor, intermediate, good, perfect);
- *transition probabilities*: $P('s_1' | 's_1') = 0.3$, $P('s_2' | 's_1') = 0.7$, $P('s_1' | 's_2') = 0.2$, $P('s_2' | 's_2') = 0.8$
- *observation probabilities*: $P('dead' | 's_1') = 0.09$, $P('poor' | 's_1') = 0.19$, $P('intermediate' | 's_1') = 0.16$, $P('good' | 's_1') = 0.15$, $P('perfect' | 's_1') = 0.41$ and $P('dead' | 's_2') = 0.07$, $P('poor' | 's_2') = 0.19$, $P('intermediate' | 's_2') = 0.07$, $P('good' | 's_2') = 0.29$, $P('perfect' | 's_2') = 0.38$
- *initial probabilities*: $P('s_1') = 0.4$, $P('s_2') = 0.6$.

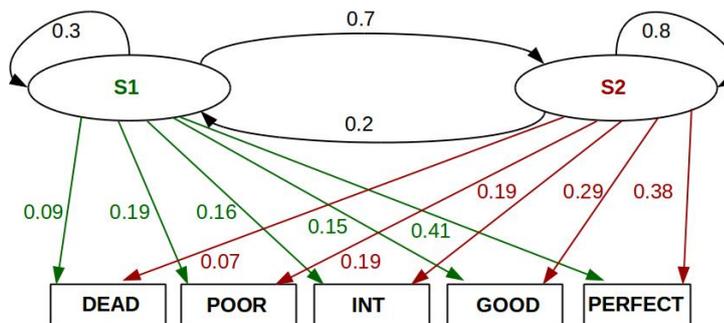


Figure 5.11: Example of two states.

Estimate model parameters. Next we show how to estimate model parameters from our data traces. We formulate our problem as follows: given some training *observation sequences* $O = o_1 o_2 \dots o_k$ (i.e., data traces from our 30-minute rounds of experiments) and a general *structure* of a hidden Markov model (number of hidden and visible states), determine the model parameters $M = (A, B, \pi)$ that best fit the training data.

$O = o_1 o_2 \dots o_k$ denotes a sequence of observations $o_k \in v_1, \dots, v_m$.

Estimation of the parameters of a hidden Markov model has most often been performed using maximum likelihood estimation [46], so we follow the Baum-Welch algorithm [8] to efficiently derive the local maximum likelihood. This is an iterative expectation-maximization algorithm to find the local maximum of $P(O | M)$, using an expectation step and a maximization one. Each iteration of Baum-Welch is guaranteed to increase the log-likelihood of the data.

We start by denoting:

$$a_{ij} = P(s_i | s_j) = \frac{\text{expected number of transitions from state } s_j \text{ to state } s_i}{\text{expected number of transitions out of state } s_j}$$

$$b_i(v_m) = P(v_m | s_i) = \frac{\text{expected number of time observation } v_m \text{ occurs in state } s_i}{\text{expected number of times in state } s_i}$$

$$\pi_i = P(s_i) = \text{expected frequency in state } s_i \text{ at time } k = 1.$$

Expectation step. The Baum-Welch algorithm starts by defining the variable $\xi_k(i, j)$ as the probability of being in state s_i at time k and in state s_j at time $k + 1$, given the observation sequence $o_1 o_2 \dots o_k$.

$$\xi_k(i, j) = P(q_k = s_i, q_{k+1} = s_j | o_1 o_2 \dots o_k).$$

$$\xi_k(i, j) = \frac{P(q_k = s_i, q_{k+1} = s_j, o_1 o_2 \dots o_k)}{P(o_1 o_2 \dots o_k)}$$

$$\xi_k(i, j) = \frac{P(q_k = s_i, o_1 o_2 \dots o_k) a_{ij} b_j(o_{k+1}) P(o_{k+2} \dots o_k | q_{k+1} = s_j)}{P(o_1 o_2 \dots o_k)}$$

$$\xi_k(i, j) = \frac{\alpha_k(i) b_j(o_{k+1}) \beta_{k+1}(j)}{\sum_i \sum_j \alpha_k(i) b_j(o_{k+1}) \beta_{k+1}(j)}$$

Following the algorithm, we define the variable $\gamma_k(i)$ as the probability of being in state s_i at time k , given the observation sequence $o_1 o_2 \dots o_k$:

$$\gamma_k(i) = P(q_k = s_i | o_1 o_2 \dots o_k)$$

$$\gamma_k(i) = \frac{P(q_k = s_i, o_1 o_2 \dots o_k)}{P(o_1 o_2 \dots o_k)}$$

Thus, having defined:

$$\xi_k(i, j) = P(q_k = s_i, q_{k+1} = s_j | o_1 o_2 \dots o_k) \text{ and } \gamma_k(i) = P(q_k = s_i | o_1 o_2 \dots o_k)$$

From here, the next step is to compute:

- expected number of transitions from state s_i to state s_j : $\sum_k \xi_k(i, j)$
- expected number of transitions out of state s_i : $\sum_k \gamma_k(i)$
- expected number of times observation v_m occurs in state s_i : $\sum_k \gamma_k(i)$, k is such that $o_k = v_m$
- expected frequency in state s_i at time $k = 1$: $\gamma_1(i)$.

Maximization step. Having γ and ξ , one can define update rules as follows:

$$a_{ij} = P(s_i | s_j) = \frac{\text{expected number of transitions from state } s_j \text{ to state } s_i}{\text{expected number of transitions out of state } s_j} = \frac{\sum_k \xi_k(i, j)}{\sum_k \gamma_k(i)}$$

$$b_i(v_m) = P(v_m | s_i) = \frac{\text{expected number of time observation } v_m \text{ occurs in state } s_i}{\text{expected number of times in state } s_i} = \frac{\sum_k \xi_k(i, j)}{\sum_{k, o_k = v_m} \gamma_k(i)}$$

$$\pi_i = P(s_i) = (\text{expected frequency in state } s_i \text{ at time } k = 1) = \gamma_1(i)$$

Having the estimated model parameters as described above, in order to generate a trace from our model, one can generate a state sequence using the transition probabilities.

5.2.3 Evaluation of the Model

We first incorporated our model for the generation of realistic link traces in COOJA [92] simulation environment, a simulator for the Contiki sensor node operating system. We have created a library of models that we trained with links quality distributions in classes from our three studied environments: OPEN, SPRUCE, BEECH, for summer and winter. Proper handling of the files with the pre-computed models for a specific environment enables the simulator to generate links with different reception rates and sequences of

link distributions from the experimental data for a specific combination of environment-season.

Methodology. To validate the accuracy of our approach we “synthetically” reproduced our summer and winter campaigns in the simulator and run TRIDENT to assess the connectivity characteristics of the network. We used the trained models and simulated a network of 240 links on the same GRID topology (i.e., same link distances) as the one used in our in-field campaigns, shown in Figure 4.3. Then, we assessed the connectivity characteristics of the simulated network using our TRIDENT tool with the same per-round configuration parameters as in-field, see Table 4.2 in Chapter 4. For each link, we sampled sequences of 30 minute rounds, for a total number of 330 rounds equivalent to 7 days of in-field experiments. Following the same methodology as for the in-field data traces, from the simulated sequences, we computed the overall *PDR*, the link distribution in link quality classes and the number of transitions. Then, we compared against the in-field data traces statistics, at both high power (HP) and low power (LP). It is worth noting that when sampling state sequences from the model, there is a possibility that the overall proportion of time a links stays in a particular state is different from the in-field trace.

Results. Table 5.6 shows a comparison between the empirical traces and the simulated traces from the model, for each environment and season, at high power (HP) and low power (LP), in terms of overall *PDR* (%). Due to logistical issues, for BEECH forest we did not run any outdoor experiments with the GRID during summer (N/A).

In the case of simulated traces we also report the \pm StdDev for 100 repetitions. Using our models, the worst case differences in overall *PDR* between the empirical traces and the model traces are 12.7% and 10.2%, considering the StdDev, in SPRUCE and BEECH, where the number of intermediate links is higher. Nevertheless, looking at the simulated traces we observe the same trend as in the empirical traces, that the quality of communication decreases as we progress from OPEN to SPRUCE to BEECH- i.e., as the quantity of trees and foliage increases. It is worth noting that the *PDR* interval for the intermediate links being very large, from 10% to 90%, for environments with a high percentage of intermediate links, like SPRUCE and BEECH, the standard deviation from the empirical traces is expected to be higher than in the OPEN.

Table 5.7 and Table 5.8 show a comparison between the empirical and simulated traces from the model, in terms of link distribution in link quality classes, in our three environments, during summer and winter, night and day, at low power. We chose to report the results of the simulations at low power as the number of transitions is bigger than at high power, as shown in Section 4.2.8. The average difference between the simulated distribution of links and the empirical distribution of links in both seasons, during summer and night, is less than 1.2%, whereas the average standard deviation of the simulated

Table 5.6: Comparison between empirical traces and simulated traces: overall PDR (%).

Season	Power	OPEN		SPRUCE		BEECH	
		Empirical trace	Model trace	Empirical trace	Model trace	Empirical trace	Model trace
Summer	HP	87.58	89.10 (1.7)	70.51	72.40 (4.3)	N/A	N/A
	LP	68.95	67.50 (3.1)	45.29	48.30 (3.9)	N/A	N/A
Winter	HP	80.05	75.56 (3.2)	50.60	42.60 (4.7)	47.5	42.5 (5.2)
	LP	68.5	65.3 (2.9)	33	27.05 (4.1)	31.4	28.60 (4.7)

traces is 2.13%. The worst case difference in the distribution of links is 5.66% in OPEN field. Nevertheless, our model has very close approximation on the percentages of links in each class. Even if the the worst case difference might seem high it is important to mention that our model is not meant to be an exact replica of the environment but an approximation to it.

Moreover, as our goal was not only to model the distribution in link quality classes but also the variations across classes, we computed the number of transitions from one class to another for the simulated traces and show the differences with respect to the empirical traces in Table 5.9. The maximum difference between the empirical and simulated traces from model is less than 4% in terms of average number of transitions over 100 simulations.

We also used the Kolmogorov-Smirnov (K-S) test to compare the CDFs of the empirical

Table 5.7: Comparison between empirical and simulated traces: summer and winter day link classification.

Season	Class	OPEN		SPRUCE		BEECH	
		Empirical trace	Model trace	Empirical trace	Model trace	Empirical trace	Model trace
Summer	DEAD	9	10 (1.4)	29	30 (1.0)	N/A	N/A
	POOR	19	17 (2.8)	10	8 (2.1)	N/A	N/A
	INT	16	14 (4.2)	26	29 (3.2)	N/A	N/A
	GOOD	15	17 (1.8)	33	31 (1.8)	N/A	N/A
	PERFECT	41	42 (1.6)	2	2 (1.1)	N/A	N/A
Winter	DEAD	23	25 (3.2)	60	52 (0.8)	57	57 (1.8)
	POOR	12	14 (3.6)	0	2 (0.6)	2	1 (0.4)
	INT	19	16 (1.9)	20	24 (3.4)	9	10 (3.4)
	GOOD	21	20 (2.4)	13	15 (1.5)	5	7 (1.2)
	PERFECT	23	25 (1.7)	7	7 (0.8)	27	26 (0.3)

and simulated traces. For the number of links in our topology (i.e., 240) and a confidence interval of 5% the K-S table [66] has a threshold value of 0.087. In our case, the model passed the K-S test with 0.07 (<0.087), hence both CDFs can be considered similar.

Model limitations. Our proposed model is a purely data-driven approach and the model is built using empirical data traces from a small scale deployment of 240 links, with lengths ranging from 10 m to 60 m. However, using our model is possible to simulate large-scale networks that have links of the same lengths as ours and with the same distribution of link lengths.

We are aware that, the interval for the intermediate links is too wide (i.e, $10\% \leq PDR \leq 90\%$) and affects the results of the overall PDR computed when using our models. Nevertheless, the value of our proposed model is not in the results per-se, which are site specific, but in the data-driven methodology we developed that exploits real data traces to create models that are incorporated in simulators. Moreover, our model can be used for simulations of protocols by accounting for the influence of the environment on the network beforehand.

5.3 Summary

In this chapter we built on the analysis and exploit the set of data traces collected during our campaigns to create two models with the distinctive goal to: (i) describe the influ-

Table 5.8: Comparison between empirical and simulated traces: summer and winter night link classification.

Season	Class	OPEN		SPRUCE		BEECH	
		Empirical trace	Model trace	Empirical trace	Model trace	Empirical trace	Model trace
Summer	DEAD	7	8 (1.1)	33	34 (1.1)	N/A	N/A
	POOR	19	18 (2.4)	5	4 (0.2)	N/A	N/A
	INT	7	8 (3.8)	21	18 (4.6)	N/A	N/A
	GOOD	29	30 (1.7)	31	33 (1.1)	N/A	N/A
	PERFECT	38	36 (1.1)	10	11 (1.3)	N/A	N/A
Winter	DEAD	19	17 (2.9)	60	56 (1.7)	54	52 (1.4)
	POOR	14	15 (3.2)	0	2 (0.6)	6	5 (0.6)
	INT	14	13 (1.1)	6	8 (2.5)	18	20 (2.6)
	GOOD	12	14 (2.0)	17	16 (1.2)	10	11 (1.1)
	PERFECT	39	41 (1.0)	17	18 (0.7)	12	12 (0.4)

Table 5.9: Comparison between empirical traces and simulated traces: number of transitions.

Season	Power	OPEN		SPRUCE		BEECH	
		Empirical trace	Model trace	Empirical trace	Model trace	Empirical trace	Model trace
Summer	HP	3134	3254 (147)	5428	5601 (261)	N/A	N/A
	LP	3864	3710 (187)	5980	6189 (275)	N/A	N/A
Winter	HP	2444	2502 (103)	3505	3391 (255)	3664	3536 (273)
	LP	2940	2835 (236)	4386	4557 (309)	4782	4595 (301)

ence of the temperature and humidity on low-power wireless links, and (ii) describe the long-term behavior of low-power wireless links collected in-field. The first model helps estimating the link quality at run-time considering the particular temperature/humidity profile of the target environment. The second model is key for efficiently reproducing realistic network conditions for large-scale simulations of long-term behavior of protocols/applications. Thus, we contribute at reducing the gap between simulation and real-world performane of protocols and applications.

Chapter 6

Conclusions and Outlook

Phil: Do you know what day is today?

Rita: No, what?

Phil: Today is tomorrow. It happened!

Groundhog Day

In this dissertation, we argued that it is possible to enable the principled design and deployment of WSNs by improving the understanding of how the natural outdoor environment affects the network stack, and providing tools and modeling techniques to address this impact. To support our argument, we built two **tools**, TRIDENT and HARPOON, for in-field connectivity and routing protocol performance assessment that support principled, repeatable, automated, and flexible collection of measurements in the target environment, rely only on the WSN nodes without any external infrastructure and do not require any coding by the end user. Using these tools we collected in vivo a large set of data traces from sites that cover different vegetation conditions, during winter and summer, and we **analyzed the data traces** and quantified the trends emerging at physical, routing and application layer. This represents the first empirical study that characterizes, from a quantitative standpoint, the overall behavior of a WSN immersed in different outdoor real-world environments. Then, we built on the analysis to exploit the set of data traces to create two **models**: for estimating the link quality at run-time, and for reproducing realistic network conditions in simulators. The first model, helps estimating the link quality considering the particular temperature/humidity profile of the target environment. The second model is key for efficiently reproducing realistic network conditions for large-scale simulations of long-term behavior of protocols/applications by accounting for the influence of the environment on the network.

The contributions of this dissertation can be applied today to aid the design and deployment of WSNs. The tools and the quantitative assessment of the characteristics

of low-power wireless links and protocol performance in-field, are key for: supporting the WSN deployment, by characterizing the target environment and determining where to place the motes to ensure communication among them; informing the selection of routing protocols and tuning of their parameters, to ensure they are well-suited to the target environment; informing application-level strategies from a qualitative point of view and tuning of parameters, to ensure the reliability of the application—the one directly relevant for the end user (e.g., biologists).

Beyond what we can do today, our contributions are key for informing protocol design improvements and methods to overcome the effect of the environmental conditions. We already have the model describing the long-term behavior of low-power wireless links incorporated in mainstream simulators. This enables performance tests of newly designed protocols under realistic conditions without the need for expensive in-field experimentation. Nevertheless, in-field performance assessment is also supported by using our tools. Further, as we showed in Chapter 4 and Chapter 5, as new hardware platforms and protocols are developed, it is important to understand to what extent they are affected by environmental factors. To this end, our tools and methodologies can be used in quantifying this impact and enable meaningful comparison of their performance in the target environments.

The experiments we described in this dissertation required significant effort, as they had to be replicated in different outdoor sites, often in harsh conditions. Nevertheless, the set of data traces can be extended with other experiments to confirm (or refute) our observations and to validate the proposed models. Other seasonal variations (i.e., autumn and spring) and environmental conditions (i.e., rain and fog) could unveil additional trends and observations. In this respect, supporting the in-field tests without infrastructure, are an invaluable asset that greatly simplifies the experimental work. Moreover, all the collected data traces constitute a step towards populating a large repository of real-world connectivity and routing traces, missing today from the literature. This would provide valuable information to system and application developers, as well as a unique opportunity for the research community at large to study and exploit large real-world datasets.

Another line of research is to use our tools and methodologies for developing models for other environments. To this end, our model for estimating the link-quality at run-time focuses on temperature and humidity as factors determining the link quality. While this is true in OPEN fields, environments for which we tailored our model, several factors other than temperature and humidity determine the link quality in forests (i.e, foliage, tree trunks). We already reported significant differences between link quality and protocol performance in OPEN w.r.t. to forests where the heterogeneous vegetation creates microclimates that amplify the complexity. We already have the tools and the methodologies

to collect data traces from new target environments. Moreover, our model for estimating link quality can be used as a baseline for a new model that can be build by adding new parameters that factor in the complexity of the new environments.

A more ambitious line of research is to use the observations we made in this dissertation as the quantitative stepping stone enabling the prediction of routing and application performance. In other words, to define a methodology that, given the combination of environmental conditions relevant to the application, yields a good estimate of its performance, based on the principled execution and analysis of in-field tests. We already have the tools (e.g., TRIDENT and HARPOON) supporting the first step of the methodology. However, models are required to link the results at the physical layer to the routing and application layer.

Until that time comes, we argue that the tools we expressly designed for gathering in-field empirical traces, the understanding and quantitative characterization of data traces from real environments, and our models, together significantly advance the state of the art by rendering the process of designing and deploying a WSN more repeatable and predictable.

Bibliography

- [1] Texas instruments inc. cc2420 - 2.4 ghz ieee 802.15.4, zigbee-ready rf transceiver.
- [2] WiseML description (TR-RS-WISEMLDESCRIPTION-1.0). Technical report, Delft University of Technology, 2010.
- [3] S. AG. Sht1x humidity and temperature sensor datasheet, version 2.04 edition, may 2005, 2005.
- [4] G. Anastasi, A. Falchi, A. Passarella, M. Conti, and E. Gregori. Performance measurements of motes sensor networks. In *Proc. of the 7th Int. Symp. on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM)*, 2004.
- [5] N. Baccour, M. Ben Jamaa, D. do Rosario, A. Koubaa, H. Youssef, M. Alves, and L. B. Becker. A testbed for the evaluation of link quality estimators in wireless sensor networks. In *Proc. of the 8th Int. Conf. on Computer Systems and Applications (AICCSA)*, 2010.
- [6] N. Baccour, A. Koubâa, L. Mottola, M. Zuniga, H. Youssef, C. A. Boano, and M. Alves. Radio link quality estimation in wireless sensor networks: A survey. *ACM Trans. on Sensor Networks*, 8(4), 2012.
- [7] K. Bannister, G. Giorgetti, and E. K. S. Gupta. Wireless sensor networking for “hot” applications: Effects of temperature on signal strength, data collection and localization. In *Proc. of the 5th Workshop on Embedded Networked Sensors (HotEmNets)*, 2008.
- [8] L. E. Baum, T. Petrie, G. Soules, and N. Weiss. A maximization technique occurring in the statistical analysis of probabilistic functions of markov chains. *Ann. Math. Statist.*, 41(1), 1970.
- [9] J. Beutel, B. Buchli, F. Ferrari, M. Keller, M. Zimmerling, and L. Thiele. X-sense: Sensing in extreme environments. In *Proc. of Design, Automation and Test in Europe (DATE)*, 2011.

-
- [10] J. Beutel, S. Gruber, A. Hasler, R. Lim, A. Meier, C. Plessl, I. Talzi, L. Thiele, C. F. Tschudin, M. Woehrle, and M. Yuecel. Permadaq: A scientific instrument for precision sensing and data recovery in environmental extremes. In *Proc. of the 8th Int. Conf. on Information Processing in Sensor Networks (IPSN)*, 2009.
- [11] C. Boano, M. Zuniga, J. Brown, U. Roedig, C. Keppitiyagama, and K. Roemer. Templab: A testbed infrastructure to study the impact of temperature on wireless sensor networks. In *Proc. of the 13th Int. Conf. on Information Processing in Sensor Networks (IPSN)*, 2014.
- [12] C. A. Boano, J. Brown, Z. He, U. Roedig, and T. Voigt. Low-power radio communication in industrial outdoor deployments: The impact of weather conditions and ATEX-compliance. In *Proc. of the 1st Int. Conf. on Sensor Networks Applications, Experimentation and Logistics (SENSAPPEAL)*, 2009.
- [13] C. A. Boano, J. Brown, N. Tsiftes, U. Roedig, and T. Voigt. The impact of temperature on outdoor industrial sensornet applications. *IEEE Trans. on Industrial Informatics*, 6(3), 2010.
- [14] C. A. Boano, K. Römer, F. Österlind, and T. Voigt. Realistic simulation of radio interference in COOJA. In *Proc. of the 8th European Conf. on Wireless Sensor Networks (EWSN)*, 2011.
- [15] P. Buonadonna, D. Gay, J. M. Hellerstein, W. Hong, and S. Madden. TASK: sensor network in a box. In *Proc. of the 2nd European Workshop on Wireless Sensor Networks (EWSN)*, 2005.
- [16] F. Cagnacci, L. Boitani, R. A. Powell, and M. S. Boyce. Animal ecology meets GPS-based radiotelemetry: a perfect storm of opportunities and challenges. *Philosophical Trans. of The Royal Society Biological Sciences*, 365(1550), 2010.
- [17] B. Capsuto and J. Frolik. A system to monitor signal fade due to weather phenomena for outdoor sensor systems. In *Proc. of the 5th Int. Conf. on Information Processing in Sensor Networks (IPSN)*, 2006.
- [18] B. Carlo Alberto, W. Hjalmar, Z. Marco Antonio, B. James, K. Chamath, O. Felix Jonathan, R. Utz, N. Lars-Åke, V. Thiemo, and R. Kay. 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)*, 2013.
- [19] M. Ceriotti, M. Chini, A. L. Murphy, G. P. Picco, F. Cagnacci, and B. Tolhurst. Motes in the jungle: Lessons learned from a short-term WSN deployment in the

- Ecuador cloud forest. In *Proc. of the 4th Int. Workshop on Real-World Wireless Sensor Networks Applications (REALWSN)*, 2010.
- [20] A. Cerpa, N. Busek, and D. Estrin. SCALE: A tool for simple connectivity assessment in lossy environments. Tech. rep. UCLA, 2003.
- [21] A. Cerpa, J. L. Wong, L. Kuang, M. Potkonjak, and D. Estrin. Statistical model of lossy links in wireless sensor networks. In *Proc. of the 4th Int. Conf. on Information Processing in Sensor Networks (IPSN)*, 2005.
- [22] A. Cerpa, J. L. Wong, M. Potkonjak, and D. Estrin. Temporal properties of low power wireless links: modeling and implications on multi-hop routing. In *Proc. of the 6th Int. Symp. on Mobile Ad Hoc Networking and Computing (MobiHoc)*, 2005.
- [23] D. S. J. De Couto, D. Aguayo, J. Bicket, and R. Morris. A high-throughput path metric for multi-hop wireless routing. In *Proc. of the 9th Annual Int. Conf. on Mobile Computing and Networking (MobiCom)*, 2003.
- [24] W. Dong, Y. Liu, Y. He, T. Zhu, and C. Chen. Measurement and analysis on the packet delivery performance in a large-scale sensor network. In *Proc. of the 32nd Int. Conf. on Computer Communications (INFOCOM)*, 2013.
- [25] V. Dyo, S. A. Ellwood, D. W. Macdonald, A. Markham, N. Trigoni, R. Wohlers, C. Mascolo, B. Pásztor, S. Scellato, and K. Yousef. Wildsensing: Design and deployment of a sustainable sensor network for wildlife monitoring. *ACM Trans. on Sensor Networks*, 8(4), Sept. 2012.
- [26] F. Fabbri, M. Zuniga, D. Puccinelli, and P. Marrón. On the optimal blacklisting threshold for link selection in wireless sensor networks. In *Proc. of 9th Eur. Conf. on Wireless Sensor Networks (EWSN)*, 2012.
- [27] F. Ferrari, M. Zimmerling, L. Thiele, and L. Mottola. The bus goes wireless: Routing-free data collection with qos guarantees in sensor networks. In *Proc. of the 4th Int. Workshop on Information Quality and Quality of Service for Pervasive Computing (IQ2S)*, 2012.
- [28] R. Figura, M. Ceriotti, C.-Y. Shih, M. Mulero-Pâzmány, S. Fu, R. Daidone, S. Jungen, J. J. Negro, and P. J. Marrón. Iris: Efficient visualization, data analysis and experiment management for wireless sensor networks. *EAI Endorsed Transactions on Ubiquitous Environments*, 14(3), 2014.

- [29] D. Ganesan, B. Krishnamachari, A. Woo, D. Culler, D. Estrin, and S. Wicker. Complex behavior at scale: An experimental study of low-power wireless sensor networks. Technical report, UCLA, 2002.
- [30] E. Ghadimi, O. Landsiedel, P. Soldati, S. Duquennoy, and M. Johansson. Opportunistic Routing in Low Duty-Cycle Wireless Sensor Networks. *ACM Trans. on Sensor Networks*, 10(4), 2014.
- [31] L. Girod, N. Ramanathan, J. Elson, T. Stathopoulos, M. Lukac, D. Estrin, and C. Ma. Emstar: a software environment for developing and deploying wireless sensor networks. In *Proc. of the 2004 USENIX Technical Conference*, 2004.
- [32] O. Gnawali, R. Fonseca, K. Jamieson, M. Kazandjieva, D. Moss, and P. Levis. CTP: An Efficient, Robust, and Reliable Collection Tree Protocol for Wireless Sensor Networks. *ACM Trans. on Sensor Networks*, 10(3), 2013.
- [33] O. Gnawali, R. Fonseca, K. Jamieson, D. Moss, and P. Levis. Collection tree protocol. In *Proc. of 7th Conf. on Embedded Networked Sensor Systems (Sensys)*, 2009.
- [34] T. J. Habib, E. H. Merrill, M. J. Pybus, and D. W. Coltman. Modelling landscape effects on density-contact rate relationships of deer in eastern alberta: implications for chronic wasting disease. *Ecological Modeling*, 222(15), 2011.
- [35] M. H. Hashim and S. Stavrou. Measurements and modelling of wind influence on radiowave propagation through vegetation. *IEEE Trans. on Wireless Communications*, 5(5), 2006.
- [36] M. Holland, R. Aures, and W. Heinzelman. Experimental investigation of radio performance in wireless sensor networks. In *Proc. of the IEEE Workshop on Wireless Mesh Networks (WiMesh)*, 2006.
- [37] J. M. Kahn, R. H. Katz, and K. S. J. Pister. Next century challenges: Mobile networking for “smart dust”. In *Proc. of the 5th Int. Conf. on Mobile Computing and Networking (MobiCom)*, 1999.
- [38] A. Kamthe, M. A. Carreira-Perpiñán, and A. E. Cerpa. Improving wireless link simulation using multilevel markov models. *ACM Trans. on Sensor Networks*, 10(1), Dec. 2013.
- [39] A. Kashyap, S. Ganguly, and S. R. Das. Measurement-based approaches for accurate simulation of 802.11-based wireless networks. In *Proc. of the 11th Int. Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM)*, 2008.

- [40] C. Keppitiyagama, N. Tsiftes, C. A. Boano, and T. Voigt. Temperature hints for sensor network routing. In *Proc. of the 11th Int. Conf. on Embedded Networked Sensor Systems (SenSys)*, 2013.
- [41] O. Landsiedel, E. Ghadimi, S. Duquennoy, and M. Johansson. Low power, low delay: Opportunistic routing meets duty cycling. In *Proc. of the 11th Int. Conf. on Information Processing in Sensor Networks (IPSN)*, April 2012.
- [42] O. Landsiedel and O. Gnawali. Dc evaluator component.
- [43] K. Langendoen, A. Baggio, and O. Visser. Murphy loves potatoes: Experiences from a pilot sensor network deployment in precision agriculture. In *Proc. of the 20th Int. Conf. on Parallel and Distributed Processing (IPDPS)*, 2006.
- [44] LASCAR. El-usb-2+.
- [45] H. Lee, A. Cerpa, and P. Levis. Improving wireless simulation through noise modeling. In *Proc. of the 6th Int. Conf. on Information Processing in Sensor Networks (IPSN)*, 2007.
- [46] B. G. Leroux. Maximum-likelihood estimation for hidden markov models. *Stochastic Processes and their Applications*, 40(1), 1992.
- [47] S. Lin, J. Zhang, G. Zhou, L. Gu, J. A. Stankovic, and T. He. Atpc: Adaptive transmission power control for wireless sensor networks. In *Proc. of the 4th Int. Conf. on Embedded Networked Sensor Systems (SenSys)*, 2006.
- [48] Y. Liu, Y. He, M. Li, J. Wang, K. Liu, and X. Li. Does Wireless Sensor Network Scale? A Measurement Study on GreenOrbs. *IEEE Trans. on Parallel and Distributed Systems*, 24(10), 2013.
- [49] A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk, and J. Anderson. Wireless sensor networks for habitat monitoring. In *Proc. of the 1st Int. Workshop on Wireless Sensor Networks and Applications (WSNA)*, 2002.
- [50] M. Maroti and J. Sallai. Packet-level time synchronization (TEP 133). <http://www.tinyos.net/tinyos-2.1.0/doc/html/tep133.html>, 2008.
- [51] S. Moeller, A. Sridharan, B. Krishnamachari, and O. Gnawali. Routing without routes: The backpressure collection protocol. In *Proc. of the 9th Int. Conf. on Information Processing in Sensor Networks (IPSN)*, 2010.
- [52] D. Moss and P. Levis. Box-macs: Exploiting physical and link layer boundaries in lowpower networking. Technical report, Tech. report, Stanford, 2008.

- [53] L. Mottola, G. P. Picco, M. Ceriotti, S. Guna, and A. L. Murphy. Not all wireless sensor networks are created equal: A comparative study on tunnels. *ACM Trans. on Sensor Networks*, 7(2), 2010.
- [54] R. Musaloiu, C.-J. M. Liang, and A. Terzis. Koala: Ultra-low power data retrieval in wireless sensor networks. In *Proc. of the 7th Int. Conf. on Information Processing in Sensor Networks (IPSN)*, 2008.
- [55] N. Patwari and P. Agrawal. Effects of correlated shadowing: Connectivity, localization, and rf tomography. In *Proc. of the 7th Int. Conf. on Information Processing in Sensor Networks (IPSN)*, 2008.
- [56] G. P. Picco, D. Molteni, A. L. Murphy, F. Ossi, F. Cagnacci, M. Corra, and S. Nicoloso. Geo-referenced proximity detection of wildlife with wildscope: Design and characterization. In *Proc. of the 14th Int. Conf. on Information Processing in Sensor Networks (IPSN)*, 2015.
- [57] J. Polastre, R. Szewczyk, and D. Culler. Telos: Enabling ultra-low power wireless research. In *Proc. of the 4th Int. Symp. on Information Processing in Sensor Networks (IPSN)*, 2005.
- [58] D. Puccinelli, S. Giordano, M. Zuniga, and P. J. Marrón. Broadcast-free collection protocol. In *Proc. of the 10th ACM Conf. on Embedded Network Sensor Systems (SenSys)*, 2012.
- [59] C. Rankine, A. Sanchez-Azofeifa, and M. MacGregor. Seasonal wireless sensor network link performance in boreal forest phenology monitoring. In *Proc. of the 11th Int. Conf. on Sensing, Communication, and Networking (SECON)*, 2014.
- [60] T. S. Rappaport. *Wireless Communications: Principles and Practice*. IEEE Press, 1st edition, 1996.
- [61] N. Reijers, G. Halkes, and K. Langendoen. Link layer measurements in sensor networks. In *Proc. of the 1st Int. Conf. on Mobile Ad-hoc and Sensor Systems (MASS)*, 2004.
- [62] C. Reis, R. Mahajan, M. Rodrig, D. Wetherall, and J. Zahorjan. Measurement-based models of delivery and interference in static wireless networks. In *Proc. the Conf. on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM)*, 2006.

- [63] T. Schmid, H. Dubois-ferrière, and M. Vetterli. Sensorscope: Experiences with a wireless building monitoring sensor network. In *Proc. of the 1st Workshop on Real-World Wireless Sensor Networks (REALWSN)*, 2005.
- [64] 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 *Proc. of the 12th European Conf. on Wireless Sensor Networks (EWSN)*, pages 266–273, 2015.
- [65] L. Selavo, A. Wood, Q. Cao, T. Sookoor, H. Liu, A. Srinivasan, Y. Wu, W. Kang, J. Stankovic, D. Young, and J. Porter. Luster: Wireless sensor network for environmental research. In *Proc. of the 5th Int. Conf. on Embedded Networked Sensor Systems (SenSys)*, 2007.
- [66] N. Smirnov. Table for estimating the goodness of fit of empirical distributions. *Ann. Math. Statist.*, 19(2), 1948.
- [67] D. Son, B. Krishnamachari, and J. Heidemann. Experimental analysis of concurrent packet transmissions in wireless sensor networks. In *Proc. of the 4th Conf. on Embedded Networked Sensor Systems (SenSys)*, 2006.
- [68] C. Spearman. The proof and measurement of association between two things. *American Journal of Psychology*, 100(3-4), 1904.
- [69] K. Srinivasan, P. Dutta, A. Tavakoli, and P. Levis. An empirical study of low-power wireless. *ACM Trans. on Sensor Networks*, 6(2), 2010.
- [70] K. Srinivasan, M. A. Kazandjieva, M. Jain, E. Kim, and P. Levis. SWAT: enabling wireless network measurements. In *Proc. of the 6th Int. Conf. on Embedded Networked Sensor Systems (SenSys)*, 2008.
- [71] M. Station. Bondone. Online, Viote.
- [72] J. Sun and R. Cardell-Oliver. An experimental evaluation of temporal characteristics of communication links in outdoor sensor networks. In *Proc. of the 5th ACM Workshop on Real-World Wireless Sensor Networks (RealWSN)*, 2006.
- [73] R. Szewczyk, J. Polastre, A. M. Mainwaring, and D. E. Culler. Lessons from a sensor network expedition. In *Proc. of the 1st European Workshop on Wireless Sensor Networks (EWSN)*, 2004.

- [74] I. Talzi, A. Hasler, S. Gruber, and C. Tschudin. Permasense: Investigating permafrost with a wsn in the swiss alps. In *Proc. of the 4th Workshop on Embedded Networked Sensors (EmNets)*, 2007.
- [75] L. Tang, K. Wang, Y. Huang, and F. Gu. Channel characterization and link quality assessment of IEEE 802.15.4-compliant radio for factory environments. *IEEE Trans. Industrial Informatics*, 3(2), 2007.
- [76] J. Thelen, D. Goense, and K. Langendoen. Radio wave propagation in potato fields. In *Proc. of the 1st Workshop on Wireless Network Measurements*, 2005.
- [77] G. Tolle et al. A macroscope in the redwoods. In *Proc. of the 1st Int. Conf. on Embedded Networked Sensor Systems (SenSys)*, 2005.
- [78] Z. Tong, D. Wei, H. Yuan, M. Qiang, and L. Yunhao. Understanding routing dynamics in a large-scale wireless sensor network. In *Proc. of the 10th Int. Conf. on Mobile Ad-hoc and Sensor Systems (MASS)*, 2013.
- [79] V. Turau, M. Witt, and C. Weyer. Analysis of a real multi-hop sensor network deployment: The heathland experiment. In *Proc. of 3rd Int. Conf. on Networked Sensing Systems (INSS 2006)*, 2006.
- [80] P. Vicaire, T. He, Q. Cao, T. Yan, G. Zhou, L. Gu, L. Luo, R. Stoleru, J. A. Stankovic, and T. F. Abdelzaher. Achieving long-term surveillance in vigilnet. *ACM Trans. on Sensor Networks*, 5(1), 2009.
- [81] M. Welsh. Sensor networks for the sciences: Lessons from the field. Keynote address, presented at DCOSS 2008, 2008.
- [82] H. Wennerstrom, F. Hermans, O. Rensfelt, and C. Rohner. A long-term study of correlations between meteorological conditions and 802.15.4 link performance. In *Proc. of the 5th Extreme Conference on Communication (ExtremeCom)*, 2013.
- [83] G. Werner-Allen, K. Lorincz, M. Welsh, O. Marcillo, J. Johnson, M. Ruiz, and J. Lees. Deploying a wireless sensor network on an active volcano. *IEEE Internet Computing*, 10(2), Mar. 2006.
- [84] K. Whitehouse, G. Tolle, J. Taneja, C. Sharp, S. Kim, J. Jeong, J. Hui, P. Dutta, and D. Culler. Marionette: using rpc for interactive development and debugging of wireless embedded networks. In *Proc. of the 5th Int. Conf. on Information Processing in Sensor Networks (IPSN)*, 2006.

-
- [85] T. Winter, P. Thubert, A. Brandt, J. Hui, R. Kelsey, P. Levis, K. Pister, R. Struik, J. Vasseur, and R. Alexander. RPL: IPv6 routing protocol for low-power and lossy networks. RFC 6550, 2012.
- [86] A. Woo, T. Tong, and D. Culler. Taming the underlying challenges of reliable multi-hop routing in sensor networks. In *Proc. of the 1st Int. Conf. on Embedded Networked Sensor Systems (SenSys)*, 2003.
- [87] J. Zhao and R. Govindan. Understanding packet delivery performance in dense wireless sensor networks. In *Proc. of the 1st Int. Conf. on Embedded Networked Sensor Systems (SenSys)*, 2003.
- [88] G. Zhou, T. He, S. Krishnamurthy, and J. A. Stankovic. Impact of radio irregularity on wireless sensor networks. In *Proc. of the 2nd Int. Conf. on Mobile systems, applications, and services (MobiSys)*, 2004.
- [89] Zigbee Alliance. ZigBee-2007 PICS and stack profiles, 2008.
- [90] M. Zuniga and B. Krishnamachari. Analyzing the transitional region in low power wireless links. In *Proc. of the 1st Conf. on Sensor and Ad Hoc Communications and Networks (SECON)*, 2004.
- [91] M. Zuniga and B. Krishnamachari. An analysis of unreliability and asymmetry in low-power wireless links. *ACM Trans. on Sensor Networks*, 3(2), 2007.
- [92] F. Österlind, A. Dunkels, J. Eriksson, N. Finne, and T. Voigt. Cross-level sensor network simulation with cooja. In *Proc. of the Int. Workshop on Practical Issues in Building Sensor Network Applications*, 2006.

