Energy-aware transceiver for energy harvesting wireless sensor networks

Amine Didioui

ANNÉE 2014

THÈSE / UNIVERSITÉ DE RENNES 1
sous le sceau de l'Université Européenne de Bretagne

pour le grade de

DOCTEUR DE L'UNIVERSITÉ DE RENNES 1

Mention : Traitement du Signal et Télécommunications

Ecole doctorale Matisse

présentée par

Amine DIDIOUI

préparée à l'unité de recherche UMR6074 IRISA/CEA-LETI
Institut de recherche en informatique et systèmes aléatoires - CAIRN
Commissariat à l'énergie atomique et aux énergies alternatives

---

Energy-Aware Transceiver for Energy Harvesting Wireless Sensor Networks

Thèse soutenue à Grenoble
le Lundi 13 Octobre 2014

devant le jury composé de :

Dominique DALLET
Professeur des Universités à l'Institut Polytechnique de Bordeaux / Rapporteur

Fabien MIEYEVILLE
Maitre de Conférences HDR à l'Ecole Centrale de Lyon / Rapporteur

Sylvain BOURDEL
Professeur des Universités à l'Université de Grenoble / Examinateur

Renaud BRIAND
Maitre de Conférences à l'ESTIA - Université de Pau / Examinateur

Rose-Marie SAUVAGE
Responsable du domaine Nanotechnologies à la Direction Générale de l'Armement / Examinateur

Olivier SENTIEYS
Professeur des Universités à l'Université de Rennes1 / Directeur de thèse

Carolynn BERNIER
Ingénieur-Chercheur au CEA-LETI / Co-directeur de thèse

ABSTRACT

Technological advances achieved over the past decade in the fields of microsystems and wireless communications have enabled the development of small size and low cost sensor nodes equipped with wireless communication capabilities able to establish a wireless sensor network (WSN). Each sensor node is typically equipped with one or several sensing unit, a data processing unit, a wireless communication interface and a battery. The challenges raised by WSNs has lead to the emergence of a new research domain which focuses on the study and deployment of such a networks in order to offer the required remote monitoring and control solutions for complex and unreachable environment. WSNs have found application in a wide range of different domains, including home and structural health monitoring, military surveillance, and biomedical health monitoring. These applications usually impose stringent constraints on the WSN lifetime which is expected to last several years. To reach this objective, it is necessary to reduce the overall energy consumption of the sensor node and to find an additional source of energy as well. To address the last point, energy harvesting from the environment seems to be a an efficient approach to sustain WSNs operations. However, energy harvesting devices, which must also be small, are usually unable to ensure a continuous operation of sensor nodes. Thus, it is necessary to adapt the WSN consumption and activity to the low and unpredictable energy scavenged. The work presented in this thesis focuses on the issue of simulation and power consumption of autonomous sensor nodes. We have first developed, HarvWSNet, a co-simulation framework combining WSNet and Matlab that provides adequate tools to accurately simulate heterogenous protocols (based on discrete-time events) and energy harvesting systems (based on continuous-time events). We have demonstrated that HarvWSNet allows a rapid evaluation of energy-harvesting WSNs deployment scenarios that may accelerate the time-to-market for these systems. Thanks to the accurate energy models (battery, supercapacitor, etc.) implemented in this platform, we have studied and evaluated a large scale deployment of solar and wind energy-harvesting WSNs. Our second contribution focuses on the implementation of energy-aware reconfiguration strategies in the radio transceiver which is usually considered as the most energy hungry component in a sensor node. These strategies are intended to reduce the excessive power consumption of the radio transceiver when the channel conditions are favorable. To this end, we have a new simulation framework called EnvAdapt (based also on WSNet) dedicated to the evaluation of reconfigurable radio transceivers for WSNs. In EnvAdapt, we have implemented the required radio transceiver behavioral and power consumption models that allows the eval-

uation of the impact of radio transceiver reconfiguration on the communication performance and lifetime of WSNs.

RÉSUMÉ

Les progrès technologiques accomplis durant ces dernières décennies dans les domaines des microsystèmes et des radiocommunications nous permettent de réaliser des composants communicants miniaturisés à faible coût afin de constituer des réseaux de capteurs sans fil. Typiquement, chacun de ces composants intègre une ou plusieurs unités de mesures (capteur), une unité de traitement de données, une unité de communication radio et une batterie. De ce fait, un nouveau domaine de recherche s'est créé pour étudier le déploiement de ces réseaux afin d'offrir des solutions de surveillance et de contrôle à distance, notamment dans des environnements complexes ou inaccessibles. Les domaines d'application de ces capteurs sont très variés, allant de la domotique au militaire en passant par le médical et les infrastructures civiles. Souvent, ces applications impliquent des contraintes sévères en terme d'autonomie qui idéalement devrait atteindre plusieurs dizaines d'années. Pour atteindre cet objectif, il est à la fois nécessaire de réduire la consommation énergétique du nœud capteur et de trouver d'autres solutions d'alimentation en énergie pour le nœud. Pour adresser ce deuxième point, la récupération d'énergie à partir de l'environnement (solaire, vibratoire, thermique, etc.) semble représenter une solution idéale pour alimenter un noeud capteur, bien que celui-ci doive s'adapter aux faibles quantités d'énergie récupérées par ces systèmes, ainsi qu'à leurs variations et intermittences. Ces travaux de thèse s'intéressent donc à la problématique de la simulation et de la réduction de la consommation des nœuds de capteurs sans-fil et autonomes en énergie. Dans un premier temps, nous avons développé la plateforme HarvWSNet, un environnement de co-simulation alliant le simulateur de réseaux WSNet et Matlab permettant ainsi la modélisation précise et la simulation hétérogène des protocoles de communication (typiquement à évènements discrets) et des systèmes de récupération d'énergie (qui possèdent typiquement un comportement à temps continu). Nous avons démontré que cette plateforme permet de réaliser très rapidement des études de pré-prototypage de scénarios applicatifs de déploiement et ainsi réduire le temps de conception de ces nouvelles technologies. Grâce à la modélisation précise des éléments du système de récupération d'énergie (batterie, supercapacité, etc.) permise par cette plateforme, nous avons étudié et évalué la durée de vie de déploiements à large échelle de réseaux de capteurs alimentés par des systèmes de récupération d'énergie (solaire et éolien). La deuxième contribution de cette thèse concerne l'étude et l'implémentation de stratégies de reconfiguration dans

l'interface de communication radio, qui est souvent la principale source de consommation d'énergie d'un capteur, afin de permettre au nœud et/ou au réseau de minimiser sa consommation lorsque le bilan de liaison RF est favorable. A cette fin, nous avons proposé une approche originale grâce au développement d'un simulateur de réseau dédié, EnvAdapt (basé sur WSNet). Dans cette nouvelle plateforme, des modèles de consommation des différents blocs du transceiver radio et des algorithmes de reconfiguration ont été implémentés afin d'étudier l'impact de la reconfiguration des performances de la radio sur la qualité de service et l'autonomie d'un réseau de capteurs.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

1

MOTIVATION, AND CONTRIBUTIONS OF THE THESIS

The beginning of the 90^{th} was marked by the appearance of the internet as we know it today. The deployment of this heavy network contributed to the modification of our life positively. In the 2000, we assisted to the emergence of the web 2.0 and social networks. This internet targeting the "connection of humains" represents a natural extension of internet in humain life. Years later, the idea of connecting everyday objects appeared naturally. This concept was mostly catalyzed by the achievements of semiconductor industry with the miniaturization of integrated circuits.

In 2005, the ITU (International Telecommunication Unit) published a comprehensive report on the "Internet of Things" (IoT) and its expansion to real world and to objects from technical, economical and ethical views [1]. The main aim of the IoT is to integrate everyday objects that surround us to the virtual world of information technology and make them a proactive actors of the internet. Figure 1.1 shows the new "Any THING" axis where the thing-to-thing and Human-to-Thing connectivity is added to complete the Human-to-Human and Human-to-Machine interaction. This axis clearly opens up a new services opportunities and new information sources that will be introduced in the connected network.

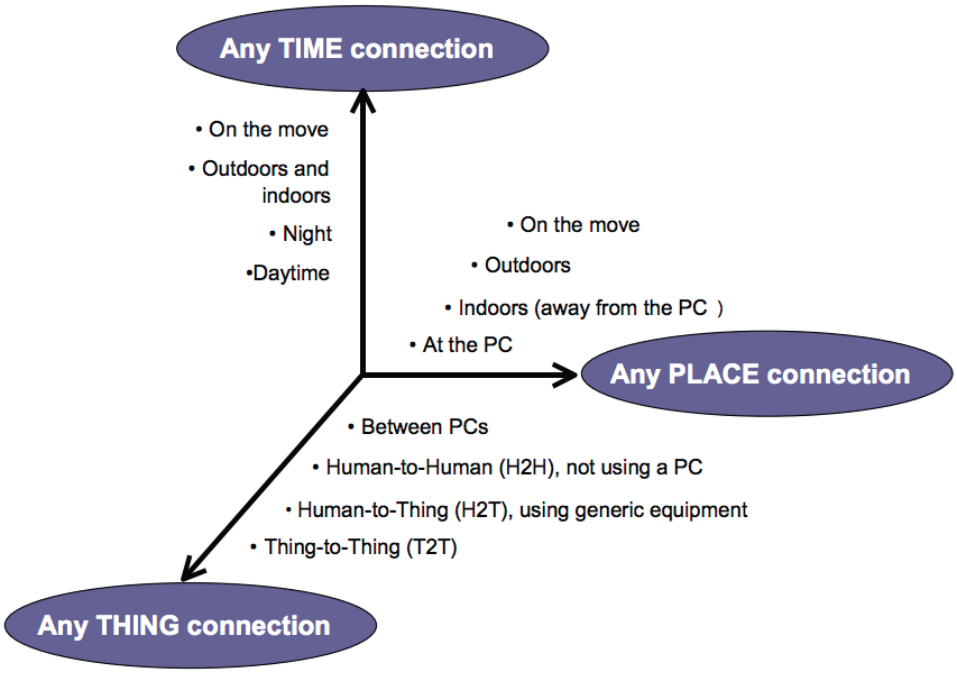


Figure 1.1: The ITU view of the new telecommunication environment [1].

1.1 MOTIVATION

IoT technology is expected to be a major growth field for both the semiconductor and internet markets. Such technology is expected to improve the productivity, efficiency, and security of various applications in our society [22]. Moreover, and more importantly, one of the most crucial building block of the IoT paradigm is wireless sensor networks (WSN) which consists of unattended nodes that sense the environment, collect data measurements, and relay them to a management entity. The major challenge facing the vision of IoT is to build large-scale autonomous WSNs that will interact with the objects and cloud services. In fact, WSNs are inherently resource-constrained components with a limited computational power, limited amount of memory, and short communication and sensing ranges [23]. All these constraints are often due to the limited storage capacity of a sensor node. The evolution of battery technologies is unable to follow the increasing processing and communication requierment of battery power devices such as WSNs. As shown in Figure 1.2, there exists a huge gap between the energy density resulting from battery technology evolution, and the increasing energy requirements of processing devices. A major challenge therefore is to find another source of energy that can sustain sensor node operations such as energy harvesting.

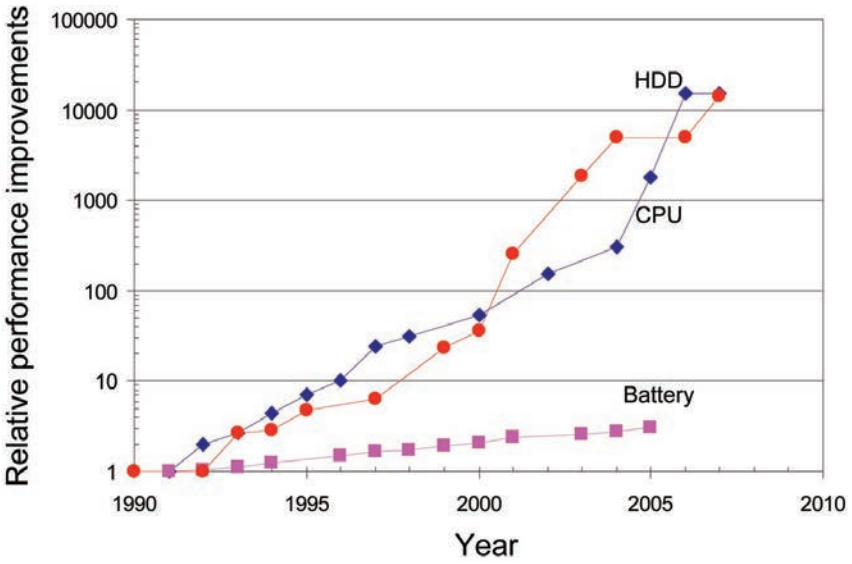


Figure 1.2: Relative improvement in the energy density of lithium ion batteries vs. the areal density of hard disk drives and the number of transistors in Intel microprocessors [2].

A typical sensor node such as eZ430-RF2500 [24] can last about 3 years on two AA batteries for a communication duty cycle of 1s. In real life applications, sensor nodes are required to last more than 15 years without service interruption. The energy consumption in sensor nodes is shared between the following components:

(1) consumption due to sensor transducer, (2) consumption due to microcontroller processing, and, finally, (3) consumption due to the radio transceiver. The most energy hungry component in a sensor node is the radio transceiver, for example, Hill et *al.* [25] found that each bit transmitted by a sensor node consumes about as much power as executing 800 to 1000 instructions.

1.2 THESIS CONTRIBUTIONS

The work presented in this thesis manly focuses on the lifetime issue of WSNs. With regard to the previously mentioned challenges facing the proliferation of WSNs and IoT, we describe in the following the main contributions of this thesis.

- **Contribution 1: Simulation of Energy Harvesting WSNs - Chapter 3**

In order to evaluate the deployment scenarios of energy harvesting sensor nodes, we propose, HarvWSNet, a co-simulation framework based on WSNet and MATLAB that provides adequate tools for evaluating energy-harvesting WSNs lifetime. The framework allows for the simulation of multi-node network scenarios while including a detailed description of each node's energy harvesting and management subsystem, and its time-varying environmental parameters.

- **Contribution 2: Power-Reconfigurable Transceiver for WSNs - Chapter 4**

Wireless transceivers must perform adequately when the channel conditions are in worst case. However, when signal and interference conditions are favorable, the transceiver may consume more energy than required. Therefore, a recon- figurable transceiver (transmitter + receiver), able to adjust its corresponding power consumption to the time-varying signal and interference conditions, could have a decisive impact on the lifetime of the sensor node. In this chapter, we propose a reconfigurable transceiver model whose purpose is to enable the study of reconfiguration strategies for future environment- and energy-aware WSNs. The model used are based on Figure of Merits of measured circuits and can easily be implemented in a wireless network simulation in order to validate the value of a reconfigurable architecture in real-world deployment scenarios.

- **Contribution 3: EnvAdapt Simulation Framework for Power Reconfigurable Transceivers - Chapter 5**

We propose, EnvAdapt, a simulation framework for WSN, implementing the required models and reconfiguration strategies for tunable transceivers. The main objective of this framework is to evaluate the transceiver reconfiguration strategies and their impact on WSNs lifetime with respect to the channel models and other network protocols algorithms of the MAC and routing layers.

2

ENERGY IN AUTONOMOUS WIRELESS SENSOR NETWORKS

Contents

2.1 INTRODUCTION

With the recent advances achieved over the last decade in highly integrated microelectronics, sensors and actuators, and low power wireless communication technologies, wireless sensor networks (WSN) have gained significant interest from both academic and industrial communities due to their potential to monitor and control the physical world from remote locations which can be difficult to reach.

Wireless sensor networks (WSN) have an enormous potential for changing industry function, enhancing people's daily lives, and monitoring ecological environment. They have already been used in a wide variety of applications such as industrial machine and process control, building and facility automation, fire rescue and medical monitoring, wild life tracking and agriculture monitoring. However, implementing such networks remains a challenging task due to hard requirements in terms of data processing and transmission capabilities as well as computational power and energy resources.

In this chapter, we present a general state of the art related to the major topics covered by this thesis: application domains of WSNs, energy-harvesting technologies and power consumption issues in WSNs.

2.2 HISTORY OF WSNS AND RELATED APPLICATIONS

To understand the requirements and tradeoffs of WSNs, it is helpful to briefly examine their origin and history. The early research and development in WSNs was initially driven by military applications. The first development of WSNs started at around 1980 by the Distributed Sensor Networks (DSN) program at the Defense Advanced Research Projects Agency (DARPA) in the USA [26]. This research program explored the challenges involved in implementing many spatially distributed low-cost and autonomous sensor nodes that collaborate with each other. The research results and testbeds generated many WSNs systems with uses mainly in the military domain. Examples of WSN applications include acoustic tracking sensors for low-flying aircrafts and Remote Battlefield Sensor Systems (REMBASS) [27]. At this time, the solutions proposed were very expensive and were dedicated only to military purposes. Even though researchers in the 1980s and early 1990s had in mind a global vision of a WSN, the processing workstations and communications technologies were not yet available to support their goals. The WSN research remained in the defense area until the end of 1990s and early 2000s when the first low-cost sensor nodes were developed [28].

From these years onward, the increasing miniaturisation of microelectromechanical systems and radio frequency devices paved the way to a new era of mobile networks and resulted in the emergence of many other potential applications, ranging from industrial to health care. One outcome of these advances was the Smart Dust project [28] developed at the University of California at Los Angeles, which focused on the development of highly miniature sensor nodes called *motes*. This project demonstrated that a complete system composed of a micro-controller, wireless transceiver, sensors, and interface circuits can be integrated in a single tiny CMOS chip that has the size of a grain of sand. Another research effort was the PicoRadio project [29] initiated in 1999 to support the development of low-cost and low-energy ad-hoc wireless networks that can power themselves from the energy sources available in their operating environment, such as vibrational or solar energy.

During the same time, the IEEE catalyzed the development of WSNs by defining the new IEEE 802.15.4 standard targeting low cost, low-power and low-data rate wireless personal networks. This standard defined the specifications of new wireless Medium Access Control (MAC) and Physical layer (PHY). To stimulate the WSN market, a group of companies formed an alliance called the ZigBee Alliance that focuses on the development of the ZigBee protocols based on IEEE 802.15.4 standard. The main objective of the ZigBee Alliance is to describe a protocol stack

architecture that guarantees interoperability among devices produced by different companies. In addition to ZigBee, many other standards have been proposed such as Enocean, 6LoPAN, Wireless HART, ANT, RF4CE, etc. [30].

Currently, WSNs are viewed as the most important technology of the 21st century due to the variety of possible applications [26]. A WSN may consist of many types of sensors that monitor different physical information including humidity, temperature, light, movement, mechanical stress, etc. As shown in Figure 2.1 [3], a huge number of applications are possible.

Figure 2.1: Future applications of WSNs [3].

These applications can be broken down into five main categories (Figure 2.2): industrial, home monitoring, health care, environmental, and military applications.

- **Industrial applications:** The industrial sector is continuously looking for more efficient ways to improve production quality and reduce costs while minimizing waste of energy and materials. Thus, it is not surprising that several organizations, such as CISCO [31], are excited about the possibilities of using WSNs to monitor and enhance each step of product including production, delivery and consumption. Sensor nodes can offer real-time access to information about equipment or plants, and prevent disruption of infrastructures [32]. Unlike wired networks, WSNs are excellent candidates for monitoring of the entire life of a product, step by step from raw material

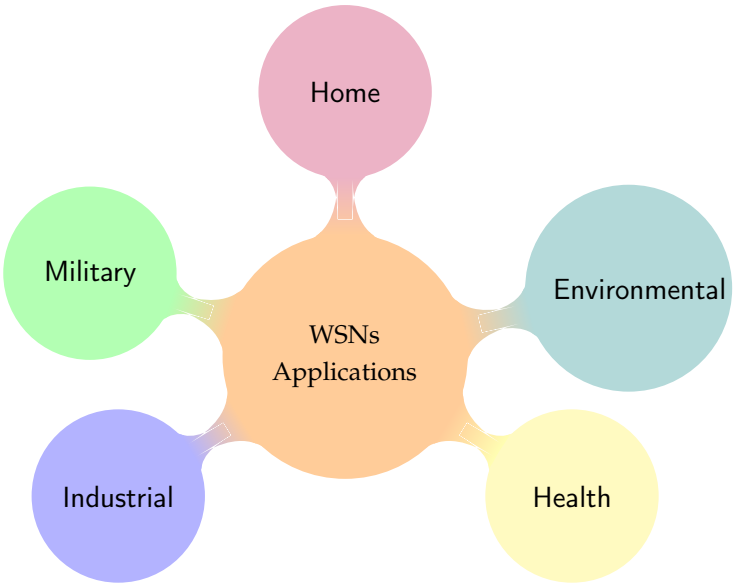


Figure 2.2: Application domains of WSNs.

provision used for its fabrication to its final assembling. The equipment to be monitored by WSNs can include assembly machines [33], product parts, logistics and transportation (containers or vehicles) [34], warehouses palettes localisation [35], and end-user assets.

- **Home monitoring:** The field of home monitoring represents another potential area for using WSNs. Sensor nodes can transform a traditional home into a "*smart home*" that is aware of its occupants' special needs and respond accordingly. A future smart home will be able to regulate the rooms' temperature, control air quality, adjust lighting, and even play music that fits with a particular resident. This automation would obviously minimize the building's energy consumption as-well. Sensor networks can also improve the security of the house by sending an alarm message to the resident when it detects intruders, gas leakage, fire occurrence or other security incidents. Moreover, a smart home may even be able to provide data to firefighters about structural integrity of the building and prevent areas that may collapse. The flexibility and easy installation of WSNs can ease the implementation of such houses and contribute to the digital future of humanity.

- **Health care:** WSN technology could potentially provide better health-care delivery for different segments of the population. Health professionals are always looking for new techniques that provide the best care to patients. Thus, WSNs may change the interaction between the medical staff and their patients. The medical domain has produced many implementations of sensor nodes ranging from equipment and patient health monitoring to wireless medicine injection and control [36]. There are many health monitoring prod-

ucts implementing WSNs such as SmartVest [37], AMON [38], Wealthy [39]. These systems transmit wirelessly and continuously physiological parameters about the health of the patient (e.g. monitoring of vital signs, blood glucose, organ monitor, cancer detector [23], etc.) so that the health care professionals can diagnose and assure timely interventions.

- **Environmental monitoring:** The growing concerns about ecology and the impact of humanity on climate change and global warming is pushing scientists to use WSNs for environment protection. WSNs can facilitate the measurement of a large variety of environmental data [40] for a huge number of applications such as agriculture, meteorology, geology, zoology, etc. The advantage of using WSNs in such applications is mainly due to the need for acquiring large amounts of data in each region that would be difficult to obtain using existing technologies. Based on this data, researchers and engineers can build efficient models that can describe the behavior of the environment and potentially predict disasters.

- **Military applications:** As stated previously, the beginning of WSNs development focused on military applications [41]. One of the most important characteristics of WSNs that make them prominent for military applications is their ability to autonomously reorganize themselves to form a network capable of routing measurement data to the commanders. From defense's perspective, WSNs represent an important technology necessary for maintaining soldiers safe in the battlefield [42]. In addition, WSNs could provide more useful information such as enemy troop movement detection, monitor the health of soldiers, coordinate resources and defense, gunfire origin detection, monitor critical equipment, etc.

Today, many of the WSNs applications described in this section are widespread. Given the usefulness of such networks in our lives, WSNs will continue their proliferation in every domain, and the number of their applications in the future will be limited only by imagination [43]. In this section, we reviewed the history and some of the well-known WSNs applications. More specifically, we considered five application fields namely industrial, home monitoring, healthcare, environmental, and military. In the next section, we present the architecture of a sensor node architecture and the corresponding building-blocks.

2.3 REQUIREMENTS AND CHALLENGES OF WSNS - TOWARD AUTONOMOUS WSNS

In this section, we focus on the requirements and challenges facing the building and deployment of WSNs. As discussed in the previous sections, in order to han-

dle a wide range of applications types, an autonomous sensor node hardware and implementation is required to address the following requirements:

- **Low Cost:** The utility and scalability of a WSN highly depends on its density, which means large numbers of sensor nodes in the deployment area. Therefore, to make the deployment cost effective, the cost of individual sensor nodes should be extremely low.
- **Low Power:** For high density WSNs, battery replacement can be difficult, expensive, or even impossible in harsh environments. Nodes must be able to ensure a permanent connectivity and coverage for long periods, ideally up to 10 years, without service interruption.
- **Small Size:** WSNs are desired to replace cables that are often impracticable. Therefore, the size of the sensor node must be small enough so that the WSN monitors the environment in an unobtrusive and transparent manner.

The real requirements and challenges of a WSN heavily depends on the application domain. Nevertheless, many applications are concerned with two main challenges that will be addressed in this thesis. Firstly, since most sensor nodes are battery-powered, it is important to tackle the limitations of batteries and their replacement. Thus, to guarantee an energy autonomy for sensor nodes, energy harvesting represents an efficient solution that may results in a self-powered node. Secondly, reducing the excessive power consumption of the radio transceiver is of primary importance to assure a long lifetime to networks that are unable to harvest energy from their environment or that harvest very little energy or in an intermittent manner.

2.4 ARCHITECTURE OF A WSN NODE

As shown in Figure 2.3, a sensor node is composed of four major blocks: processing unit, communication interface, sensors and power supply.

- **Microcontroller:** Since the main functionalities of a sensor node consists in communicating, processing and gathering sensor data, sensor nodes must be equipped with a microcontroller unit (MCU). The MCU is responsible for execution of algorithms and communication protocols, controlling of the sensor and actuators, processing of gathered data, and it can be used also for the management of the energy harvesting block [44]. The MCU is required to be energy-efficient with a very short wake-up time.
- **Transceiver:** Wireless communications between sensor nodes and also with the base station are carried by means of a wireless transceiver interface. A

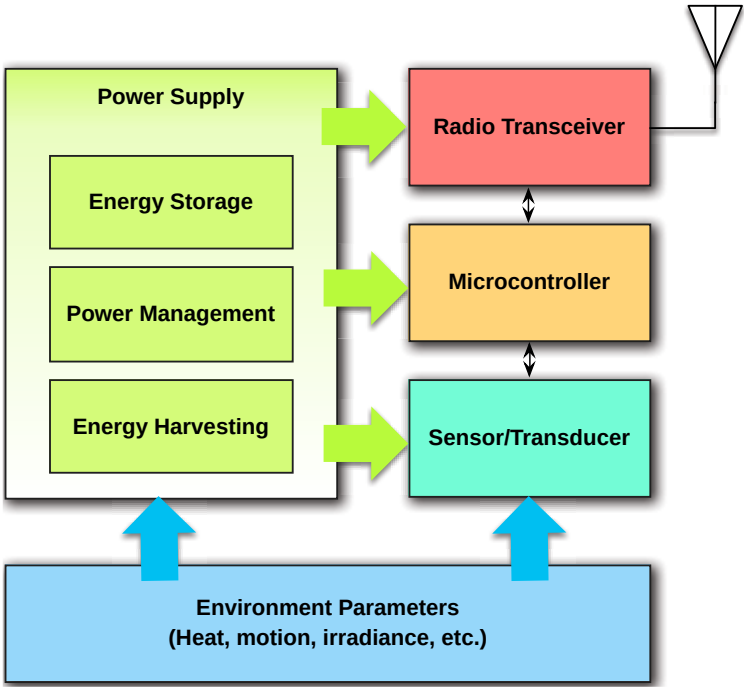


Figure 2.3: Block diagram of a sensor node.

wireless transceiver implements all the necessary functionalities to convert bits data to radio waves and vice-versa. There exists many wireless standards operating in different frequency bands such as ZigBee, Bluetooth Low Energy (BLE), Ultra-Wide Band (UWB), etc.

- **Sensors and actuators:** Environment and target application is monitored using sensors which are responsible of converting physical world information into electrical signals. There exists a plethora of sensor nodes that measure environmental parameters such as light, temperature, gas content in the air, sound, etc. These sensors can be classified as either analog or digital depending on the characteristics and speed requirements of the microcontroller and actuators used.

- **Power supply:** The power supply block has the functionality of supplying the required energy to power the sensor node. While typically consisting of a battery, it might also consists of an energy harvesting block which scavenges environmental energy, a power manager which is usually based on DC-DC converters that regulate the harvested energy, and finally a storage element responsible for storing the energy needed by the sensor node when the environmental energy to harvest is unavailable.

EXAMPLE: SIMPLICITI SENSOR NODE

In order to assess the lifetime issue in a WSN, we consider the TI eZ430-RF2500 sensor node which combines the MSP430 MCU and the CC2500 low power radio

transceiver. This node consumes about 6-10µA during deep sleep mode and 15-20mA in full operation mode. Figure 2.4 shows the current consumption profile of the sensor node during temperature measurement and data transmission to a base station.

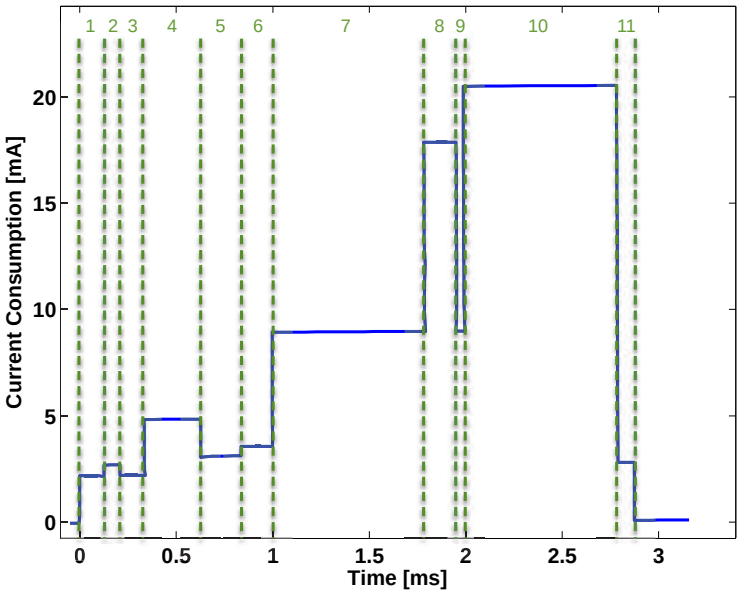


Figure 2.4: Current consumption profile of the eZ430-CC2500 platform.

1	Temperature Measurement
2	Vcc Measurement
3	Temperature & Vcc Calculation
4	Oscillator activation
5	Transceiver Power On
6	Data prepare
7	PLL synchronization
8	Clear channel assessment
9	Rx to Tx switching
10	Data Tx
11	Sleep mode switching

Table 2.1: eZ430-CC2500 consumption states.

The execution time of this application is about 2.8ms and the average current consumed by the sensor node $I_{ON\text{-}avg}$ is 12.33mA. To assess the lifetime of the application, we consider two AAA batteries with a nominal capacity of 1000mAh. The application lifetime is calculated by:

$$\begin{aligned} \text{Application Lifetime} &= \frac{C_{\text{battery}}}{I_{\text{Application-avg}}} = \frac{1000\ [mAh]}{12.33\ [mA]} \\ &\simeq 3 \text{ days} \end{aligned} \tag{2.1}$$

To reduce this excessive current consumption, the sensor node should reduce the amount of transmission of its temperature measurement to a base station (i.e. at a period of 1 second). Thus, the sensor node lifetime is given by:

$$\begin{aligned} \text{Application Lifetime} &= \frac{C_{\text{battery}}}{I_{\text{Application-avg}}} = \frac{1000\ [mAh]}{\frac{(I_{\text{ON-avg}} \times t_{ON}) + (I_{\text{OFF-avg}} \times t_{OFF})}{t_{ON} + t_{OFF})}} \\ &= \frac{1000\ [mAh]}{\frac{(12.33[mA] \times 2.88[ms]) + (1.3[\mu A] \times (1[s] - 2.88[ms]))}{1[s]}} \\ &\simeq 3 \text{ years} \end{aligned} \tag{2.2}$$

In this example, the calculated lifetime of the sensor node is 3 years. This result seems to be overestimated for two reasons: The first one is due to the hypothesis that the battery voltage is always sufficient to supply the sensor node. The second is related to the leakage of the battery which was not taken into account. This example clearly shows that a network supplied only by batteries has a very limited lifespan. Therefore, in order to maximize network operation without battery replacement, it is necessary to reduce the power consumption of each component of the sensor node or add an additional energy source such as energy harvesting.

2.5 ENERGY HARVESTING AND STORAGE IN WSN

Energy harvesting is a technique where ambient energy is captured from multiple sources and converted into electrical energy. During the last decades, solar and wind energy have been widely used to provide electrical energy to industry and urban applications due to their high power yield. Recently, researchers have actively investigated the adaptation of such renewable energies to WSNs. Since WSNs are expected to be deployed in harsh or inaccessible environments for many years, several research works have focused on the diversification of energy sources for powering autonomous sensor nodes. In this section, we briefly overview existing environmental energy sources and storage elements for WSNs.

2.5.1 *Sources of Environmental Energy*

Environmental energy harvesting represents a viable option to increase the autonomy of WSNs where batteries are hard or impossible to recharge or replace.

Several factors make them a key choice for perennial operation of sensor nodes: (1) The recent advancement in low-power electronics has significantly reduced the power consumption of sensor nodes. Hence, ambient harvested energy may provide a long-term solution and reduce the dependency on batteries. (2) Energy autonomous WSNs are easy to install and do not require any wiring cables. Consequently, the heavy installation cost and battery replacement can be reduced greatly [45]. There are various sources of ambient energy available for energy harvesting as shown in Figure 2.5. These energy sources can be categorized into five types: *mechanical*, *thermal*, *solar*, *wind*, and *radiofrequency*.

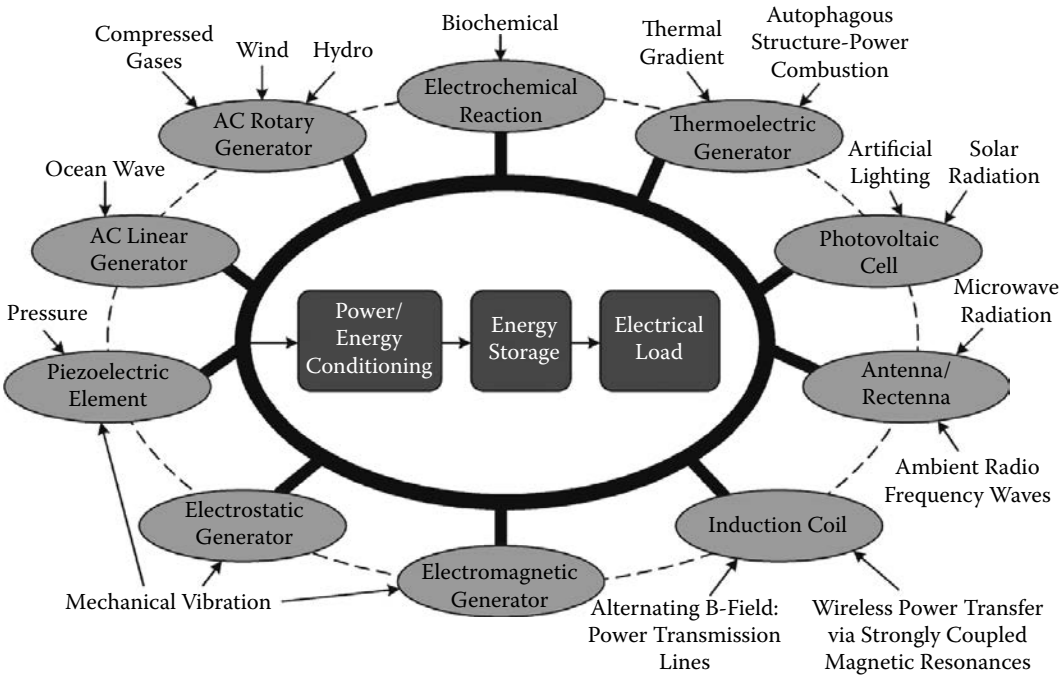


Figure 2.5: Energy harvesting sources and their energy harvesters [4].

2.5.1.1 *Solar Energy Harvesting*

Solar energy is the most popular ambient energy due to its high power density [46]. Consequently, a huge number of research work has focused on the design of micro-solar PV and energy harvesting hardware to efficiently make use of solar energy [6] [5] [8]. Solar harvesting consists in converting the incident light into electrical power using photovoltaic (PV) panels. Figure 2.6 shows a typical solar energy harvester which consists of a PV panel and DC-DC converter that maximizes the energy transfer to a load.

The two first prototypes of solar harvesting sensor nodes were Heliomote [6] and Prometheus [5]. In these systems, built with off-the shelf components, the solar panel is directly connected to the storage device (supercapacitor & Li-ion

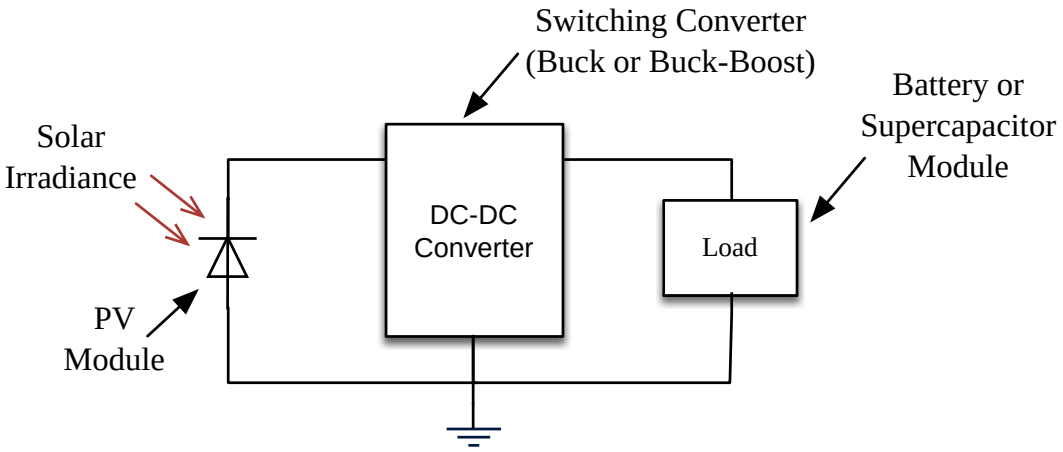
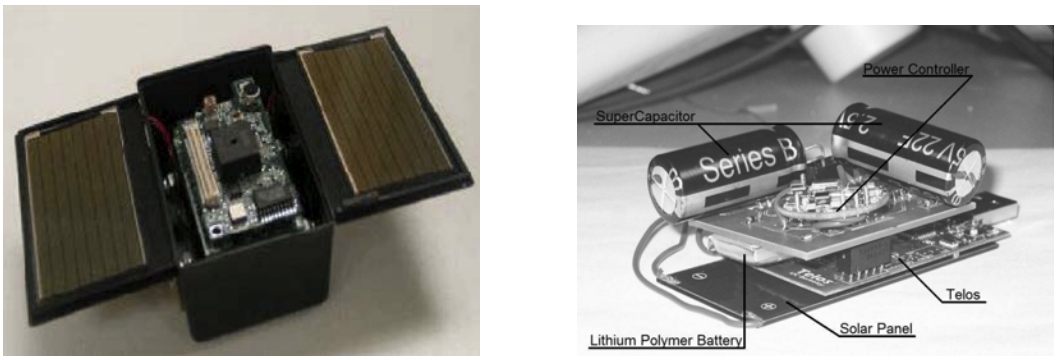


Figure 2.6: Simple diagram of a solar energy harvester.

battery in Prometheus, and NiHM battery in Heliomote). The Prometheus and Heliomote systems are depicted in Figure 2.7.



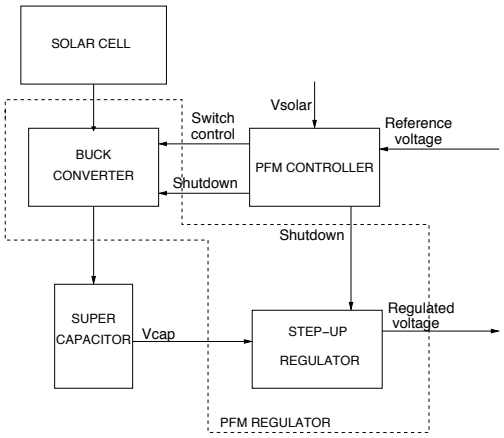
(a) Heliomote [6].

(b) Prometheus [5].

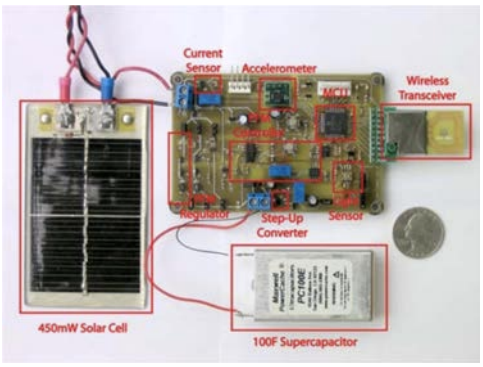
Figure 2.7: Examples of solar energy harvesting sensor nodes (a) Prometheus [5] and (b) Heliomote [6].

To achieve a high conversion efficiency, a power management system is required in order to adapt the operating point of the PV panel to the variations of light conditions, so that the maximum power (P_{MPP}) is always maintained. Everlast and Ambimax are examples of solar energy harvesting sensor nodes implementing such power management systems. The Everlast sensor node implements a system that estimates the maximum voltage power point (V_{MPP}) based on the open-circuit voltage (V_{OC}) of the solar panel. The global architecture of this system is represented in Figure 2.8. The pulse frequency modulated (PFM) regulator consists of a buck-converter and a step-up regulator. Its main function is to transfer the energy to the supercapacitor stage. The Everlast sensor node claims a lifetime of 20 years at a duty cycle of 50 % and a data rate of 1 Mbits/s.

Ambimax [8] (Figure 2.9) is a sensor node which obtains its energy from both solar and wind harvesting systems. The MPPT system of Ambimax uses a photo-



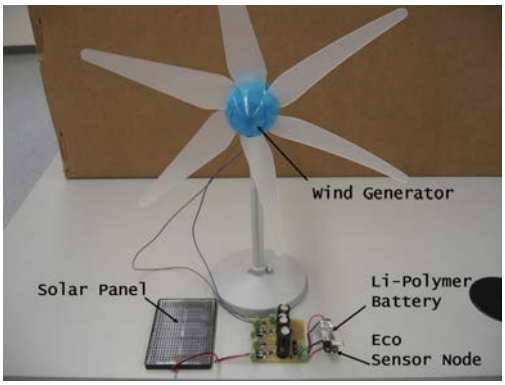
(a) Block diagram of the Everlast's energy harvesting subsystem [7].



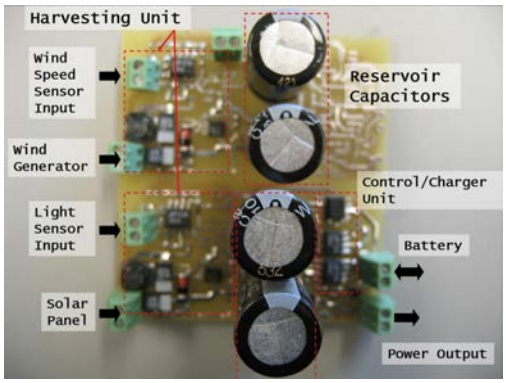
(b) Everlast prototype board [7].

Figure 2.8: The Everlast prototype node and its block diagram [7].

sensor to detect the ambient light conditions coupled with a hysteresis comparator that forces the solar panel to work in its maximum power point.



(a) Ambimax sensor node [8].



(b) Ambimax hardware board [8].

Figure 2.9: The Ambimax prototype node and its hardware board [8].

PowWow [9] is an energy harvesting sensor node based on an MSP430 microcontroller and a CC2420 RF transceiver. This sensor node has the ability to harvest energy from solar and thermal energy using a power manager based on based on the LTC3108. This power manager adapts the sensor node communication duty cycle according to the estimation of harvested energy and the consumed energy provided by an energy monitor. The experiments showed that the thermal energy-harvesting system [47] can power the sensor node using the wasted heat of a personal computer adapter.

These sensor nodes prototypes have successfully proven that solar energy harvesting is able to provide electrical power to wireless sensor nodes. In practice however, implementation of a solar energy harvesting system must take into ac-

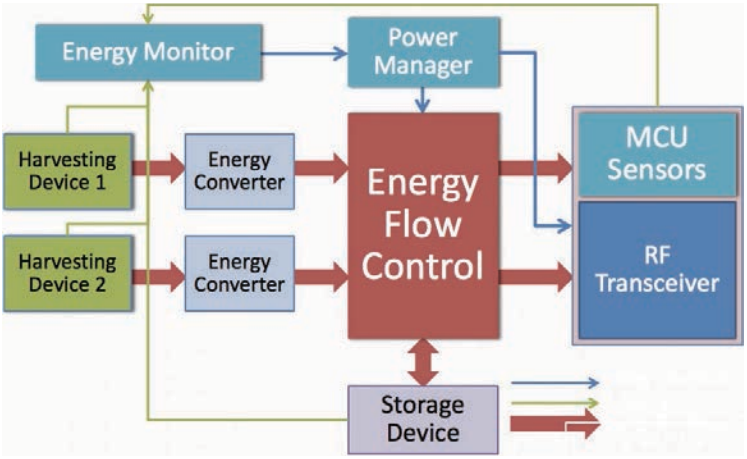


Figure 2.10: Generic architecture of the PowWow sensor node [9].

count many factors such as availability of the light source, electrical characteristics and the size of the PV cell used, capacity of the storage element, and power consumption of the other sensor node components.

2.5.1.2 *Wind Energy Harvesting*

Similarly to solar energy, wind energy harvesting has been widely used for high-power applications where wide turbines are used to generate electrical energy. In the WSNs context, there has been only few research efforts that attempt to develop small scale wind energy harvesters to power WSNs deployed in indoor applications such as underground tunnels. In these research works, we distinguish two approaches for wind energy harvesting: The first one is a direct approach where traditional wind turbines are used, and the second one is an indirect approach which uses piezoelectric materials to generate electrical power from air-flow energy.

The Ambimax sensor node [8] (Figure 2.9) and the work presented in [48] uses a direct approach to sustain electrical power for the sensor node. The results have shown promising results of power densities that can be harvested from air velocity. The global architecture of a wind energy harvester is represented in Figure 2.11. It consists of three main blocks: (1) a wind turbine coupled with an electrical generator, (2) a power management unit, which consists of power conditioning circuitry (AC-DC/DC-DC converters and MPPT) and energy storage stage, and (3) the sensor node which consists of a microcontroller, sensors, and radio transceiver.

Similarly, in other research works, Priya et *al.* [10] have developed a windmill that uses a piezoelectric bimorph transducer to generate electricity from air-flow energy. Figure 2.12 shows the schematic of the piezoelectric bimorph and the fabricated windmill.

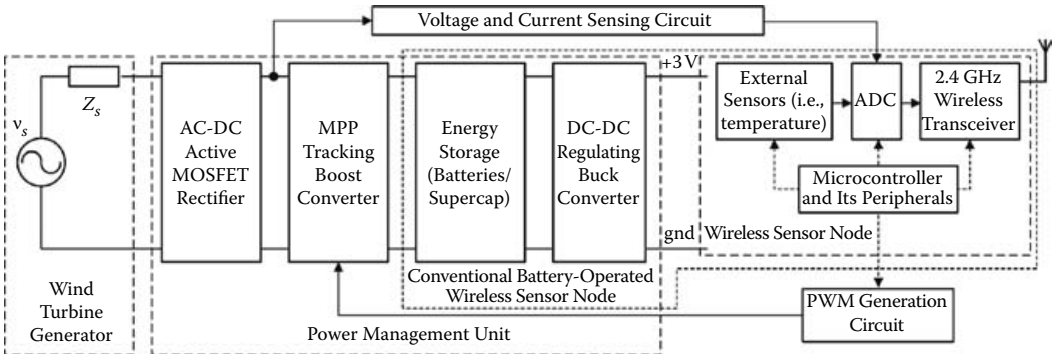
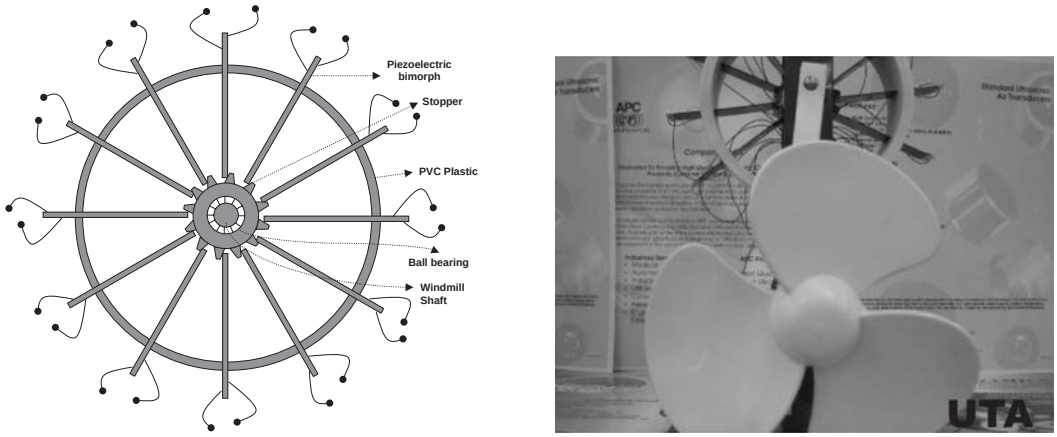


Figure 2.11: Block diagram of a wind energy harvesting for wireless sensor node [8].



(a) Piezoelectric bimorph transducer.

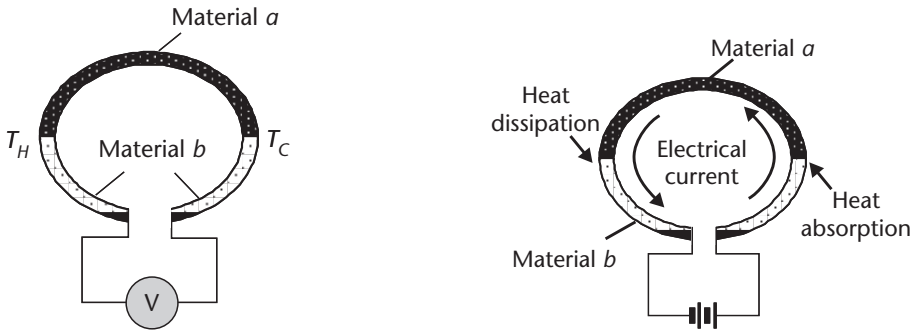
(b) Piezoelectric windmill prototype.

Figure 2.12: Piezoelectric bimorph and windmill prototype used for wind energy harvesting [10].

The power densities delivered by air-flow harvesters may exceed the required energy by a sensor node, thus, an adequate power management system is mandatory to regulate the power delivered by these systems. However, the size of the wind turbine is still an obstacle for the proliferation of wind energy harvesters compared to the dimension of a sensor node.

2.5.1.3 *Thermoelectric Energy Harvesting*

Thermal energy is the most abundant and is found in many applications and environments. Typical examples of these include heat from industrial machines, home appliances, vehicle exhausts, human body [49], etc. Converting such energy into electrical energy may lead to the highly desirable self-powered WSNs. In this context, thermoelectric generators (TEG) were developed in order to produce electrical power from heat flow by exploiting two main effects: Seebeck and Peltier effects.



(a) The Seebeck effect: a voltage generated by the temperature difference across the junctions.

(b) The Peltier effect: heat absorption (or dissipation) at junctions due to electrical current.

Figure 2.13: Thermoelectric effects [11].

The Seebeck effect consists of generating a voltage along a semiconductor when it is subjected to a temperature gradient as shown in Figure 2.13a. On the other hand, the Peltier effect, inverse of the Seebeck effect, reflects the fact that when carriers flow through a conductor/semiconductor, they also carry heat as shown in Figure 2.13b.

Several researchers and companies have developed different approaches and microsystems to recover heat flow energy. In [50], the authors developed a thermoelectric converter for low power-electronic circuits, targeting power consumption ranges from hundreds of microwatts to a few milliwatts. Seiko has developed the Thermic wristwatch (Figure 2.14) which uses 10 thermoelectric modules to generate some microwatts to run its clock movement from the thermal gradient provided by body heat.

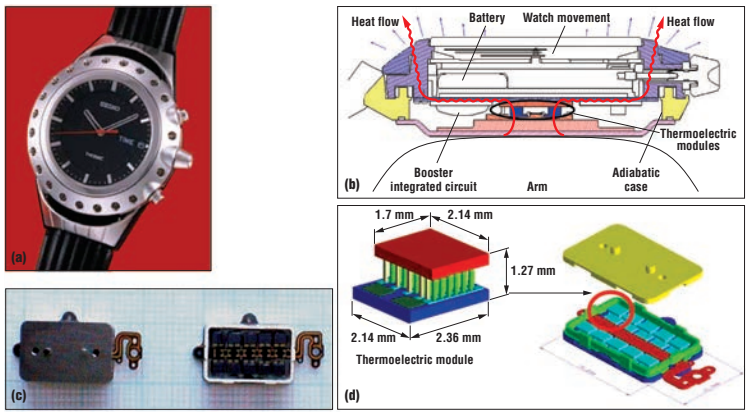


Figure 2.14: The Seiko Thermic wristwatch: (a) the product; (b) a cross-sectional diagram; (c) thermoelectric modules; (d) a thermopile array [12].

2.5.1.4 *Piezoelectric Energy Harvesting*

Energy harvesting from vibrations and movement has recently become promising for powering sensor nodes. Indeed, many urban environments, such as railways, highways, bridges and also human bodies are subjected to vibrations. To power sensor nodes from these ambient vibrations, many researchers have successfully built and tested several prototypes based on piezoelectric materials [51] [52] [53]. A typical example of piezoelectric energy harvesting is shown in Figure 2.15 where the piezoelectric generator was included in the heel of a shoe to generate electrical power as humans walk about [13].

Figure 2.15: Piezoelectric-powered shoes with mounted electronics[13].

Enerbee [54] has developed a piezo-magnetic system able to generate electrical energy from omni-directional movements. The energy harvested can power a ZigBee or Bluetooth Low Energy sensor node.

2.5.1.5 *Radiofrequency Energy Harvesting*

Every day, millions of people around the world are using a huge number of wireless technologies such as cell-phones, hand-held devices, WiFi access points, etc. All of these devices are using radio-frequency waves to communicate. Thus, harvesting these RF signals may be a promising solution if the harvested energy is sufficient to power sensor nodes. In this context, several researchers around the world have explored the feasibility of RF harvesting systems. In [55], the authors have implemented an RF energy harvester in the Mica2 sensor node. The sensor node adapts its communication duty cycle to the harvested energy. In [14], the authors showed that energy transmitted by a nearby TV station, at a distance of 4km, can power a temperature sensor as shown in Figure 2.16. The antenna of a size of around 30 cm by 20 cm, can harvest 60 μW, which gives an energy density of $0.1\mu W/cm^2$. Furthermore, some commercial products have also surfaced such as Powercast [56], aiming to bring RF energy harvesting to WSNs deployments.

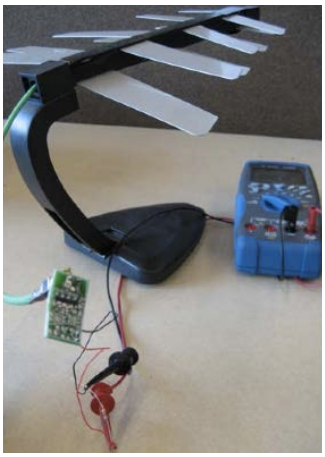


Figure 2.16: Prototype of RF energy harvesting from a nearby TV tower [14].

2.5.2 *Energy Storage Components*

Energy storage is a very critical issue when conceiving an autonomous WSNs. Actually, there are only two energy storage technologies available to the mass market: batteries and supercapacitors. Due to their high energy density, batteries are the most common storage element and they are usually used as a primary buffer stage for most WSNs. However, batteries have a limited number of charge/discharge cycles and require elaborate charging and control circuits to prevent them from overcharge/over-discharge or even overheating and combustion. On the other hand, supercapacitors offer a longer lifetime due to the infinite number of charge/discharge cycles. They are usually used as secondary storage elements due to their low energy density. However, supercapcitors are expensive and suffer from higher self-discharge.

2.5.3 *Architecture of Energy Management Systems for WSNs*

As noted before, energy is a very scarce resource in WSNs which puts stringent efficiency requirements on the power management system which is responsible of sensor node power supply. The energy management system has mainly three functions: (1) transfer effectively the harvested energy to the storage load or to the sensor node, (2) adapt the impedance of the energy transducer to the external load, and (3) manage and regulate the output voltage of one or many scavengers. The power management system must be able to adapt to the activity of the sensor node and supply large peak currents during transmission or reception while minimizing quiescent currents during the sensor node inactivity.

In the literature, only few power management systems have been proposed. In [8], Park et *al.* propose to power the Ambimax sensor node by combining both

solar and wind energy harvesting. The power management system is based on a MPPT tracking using hysteresis comparator and a boost regulator responsible for battery charging. In [57], the authors propose a power estimator and management system able to identify the most power consuming elements and energy availability prediction for WSNs. In [15] Christmann et *al.* propose an energy harvesting and power management platform combining four energy harvesters which extracts photovoltaic, thermal, vibration, and RF energy. The platform includes also the power management blocks for each energy source. Figure 2.17 shows the global architecture of the Managy platform.

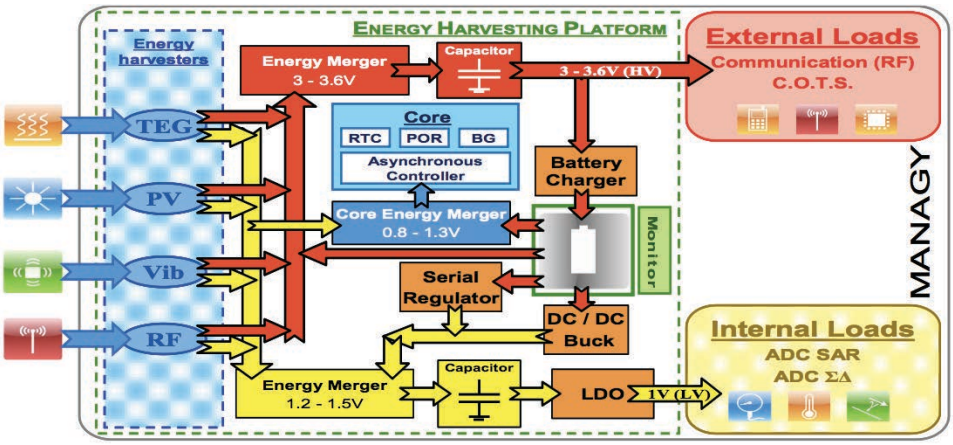


Figure 2.17: Architecture of the Managy Energy Harvesting Platform [15].

2.6 TRANSCEIVER POWER CONSUMPTION

This section focuses on the problematic of power consumption in the radio transceiver which is responsible for wireless communication between the sensor nodes. A transceiver consists of a transmitter, a receiver, and all the circuitry responsible for signal processing such as modulator/demodulator, filtering, signal conversion, etc. Given the limited amount of data exchanged between sensor nodes, the wireless transceivers for WSNs usually employ low data rates ranging from 100kb/s to few MB/s. Generally, these transceivers work typically in half duplex mode to reduce the usage of radio channel which is also considered as scarce resource. In fact, wireless transceivers can be classified into three categories which depend on the frequency band used. The first category concerns conventional narrow band transceivers which operate in the 2.4 GHz ISM frequency band and implement one of the low-rate standards such as ZigBee or Bluetooth with its low energy variant (BLE). The second category is formed by the sub-GHz transceivers operating in the ISM 433 MHz, 868 MHz and 915 MHz bands and are based on IEEE 802.15.4 and proprietary standards [58]. Finally, the third group consists of ultra-wide band transceivers operating in bands ranging from 3.1 GHz to 10.6 GHz, and occupying

a bandwidth of several GHz. Recently, a new standard IEEE 802.15.6 [59] targeting wireless body area networks (WBAN) and including a PHY layer which supports both narrow band and ultra wide band communications was proposed.

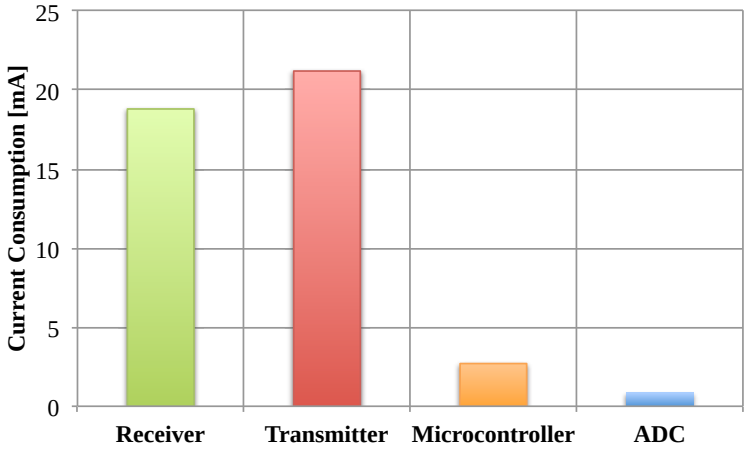


Figure 2.18: Current consumption distribution in the eZ430-RF2500.

The wireless transceiver is typically the most energy hungry component in a WSN node. For the eZ430-RF2500 sensor node, the current consumption distribution in the active mode is shown in Figure 2.18.

Since in the active mode the wireless transceiver consumes the largest fraction of the power of the node, the natural way to reduce its power consumption is to use a duty-cycling technique which consists in turning on the sensor node components for a very short period to perform the target functions, and then placing the components in a deep sleep mode. In this context, a huge number of MAC [60] [61] [62] and routing protocols [63] [64] [65] have been designed to exploit this technique. The results obtained in these works showed a drastic reduction of the average power consumption by orders of magnitude. In practice however, there are two main problems with duty-cycling: Firstly, determining the exact wake-up time for sensor nodes is significantly affected by the problem of clock drift. In the WSN literature, this is known as *rendezvous* time [66]. Secondly, several critical application such as emergency detection, disaster relief, target tracking, and so on, require a permanent interaction between the sensor nodes and the base station without the latency inherent to duty-cycling.

Clearly, reducing the power consumption of the active mode of the transceiver is a key method to reach the requirements of autonomous WSNs. Therefore, the second part of this research is devoted to the study of new-low power strategies that focus on the reduction of the transceiver power consumption as function of the conditions of communication channel.

2.7 CONCLUSION

Wireless sensor networks hold great promise in applications ranging from environmental and structural monitoring to industrial control and human health monitoring. However, almost all of WSNs available on the market still operate on batteries. Such a situation presents a substantial roadblock to the proliferation of WSNs due to the long lifetimes and small form factors needed by sensor nodes that batteries fail to provide. Energy harvesting will thus be the key choice for the development of autonomous WSNs. The characteristics of the most promising energy harvesting technologies and the fundamental energy subsystems have been overviewed in this Chapter. In the next Chapter, we present a co-simulation framework able to evaluate the lifetime of sensor nodes with heterogeneous harvesting capabilities.

3

ENERGY HARVESTING CO-SIMULATION FRAMEWORK FOR WSN'S

Contents

3.1 INTRODUCTION - PROTOTYPING VS. SIMULATION

Lifetime evaluation of sensor networks is a difficult task. Indeed, to evaluate the feasibility of an energy harvesting WSN applicative scenario, all of the system components, including the energy harvesting subsystem, the wireless sensor node

and the networking protocols, must be validated simultaneously. To this end, several approaches for performance evaluation of networks and systems are used: mathematical models, prototyping, hardware/software emulation, and computer simulations. Figure 3.1 shows the main approaches used for WSNs evaluation and their hybrid approaches. Each approach has its strength and weaknesses. The first approach is mathematical models which provides a description of systems based on a large number of variables each with its probability distribution and their interconnecting equations [67] [68]. This complexity makes it usually difficult to identify the global behavior of the system [69].

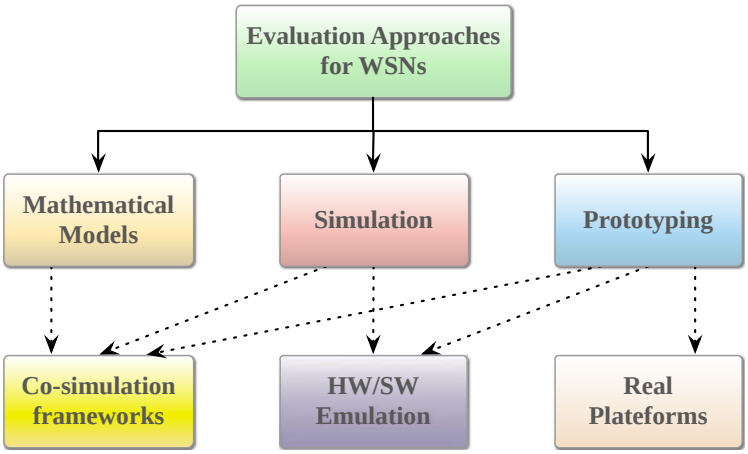


Figure 3.1: Evaluation approaches for WSNs.

The second approach is prototyping which is considered as the natural way for evaluating wireless protocols, applications and network lifetime. This approach consists in building a testbed network using existing hardware platforms and evaluating its performance in a real environment. Usually, a testbed provides the most accurate data about the system because it runs real hardware and takes into account the factors that impact the sensor lifetime such as the available energy in the environment of the sensor node, leakage currents, radio propagation environment, etc. However, setting up a real-world large scale testbed composed of hundreds of wireless sensor nodes is a time consuming task and requires access to a huge deployment surface as well as important financial and human resources. Currently, there exists numerous real-world plateforms such as Senslab [70], Sensorscope [71], GreenOrbs [72] and Wisebed [73]. The main objective of these infrastructures is to provide researchers the ability to build and deploy their applications with all contraints of a real deployment without wasting time. For example, Senslab lets the user to select remotely the number of nodes, radio characteristics, network topology, implementation time, etc. [74]. Even if these infrastructures represent a key choice for algorithms and protocols testing, they are, however, limited to existing hardware and still use batteries as source of energy for sensor nodes.

Since prototyping in realistic environments is laborious and making testbeds of such systems usually impossible due to technical and financial constraints, computer simulation remains the most affordable and the most used methodology to evaluate sensor networks [75]. The main advantages of simulation over other approaches are as follows:

- It can be used in the evaluation of existing and non-existing hardware and protocol stacks;
- It can be used to study the internal states and parameters of components such as microcontrollers, transceivers, energy storage devices, etc., to which we typically do not have access;
- It can be easily controlled in order to study specific factors that affect the WSN's deployment scenario;
- It can be cheap. Simulation does not require expensive testbeds or large amount of human resources;
- It provides the capability of running the same scenario several times with different combinations of environmental and hardware factors in order to assess their impact on the network performance;
- It allows a fast evaluation of network lifetime that can last many years in a real-world deployment.

The successful evaluation of WSN scenarios using simulation requires that the simulation results match the behavior of the corresponding WSN in a real-world deployment. Consequently, this implies that accurate models of various parts of sensor node and its environment, such as radio links, interference, mobility, energy storage, etc., be implemented in the network simulator.

Due to the lack of realistic environment and hardware models used in simulation, the majority of existing WSNs simulators are unable to reproduce the behavior of the real system. This is particularly due to the assumptions made and the simplified models used such as linear battery model, omnidirectional and symmetrical radio transmissions or propagation without shadowing or fading phenomena. These assumptions lead to simulation results that fail to replicate the experimental results [76]. In [75], the authors surveyed papers published in Mobihoc Conference between 2000 and 2005 and found a lack of rigor in the models used and in the simulation methodology. The survey concludes that more than 85% of the published papers are not repeatable because of scarcity of documentation concerning the simulators used, erroneous simulation input parameters and inappropriate propagation and mobility models. They highlighted that in the process of protocol

evaluation, researchers must validate the simulation models using a real testbed experimentation. Similarly, *Andel et al.* [77] questioned the lack of accuracy of the simulation studies. They point out that "properly validating simulation models against the intended or real-world implementation and environment can mitigate many of the problems of simulation" such as unrealistic models and incorrect simulation parameters.

To fully understand the complexity of designing a WSN and developing scenarios which work in real-life, it is necessary to develop accurate models that capture the real behavior of a sensor node and calibrate them using real-world prototypes. A compromise between simulation and prototyping is emulation. Emulation is a hybrid approach that combines both software simulations and hardware experimentation. In [78] and [79], the authors discuss the advantages and drawbacks of emulation with respect to simulation and prototyping. However, existing emulation platforms such as PowerTOSSIM [80] implement simplified energy models and do not support the simulation of energy harvesting systems.

3.2 AN OVERVIEW OF AVAILABLE SIMULATION FRAMEWORKS FOR EH-WSNS

An autonomous WSN consists of nodes that harvest environmental energy to perpetually sustain interaction with real-world applications as shown in Figure 3.2.

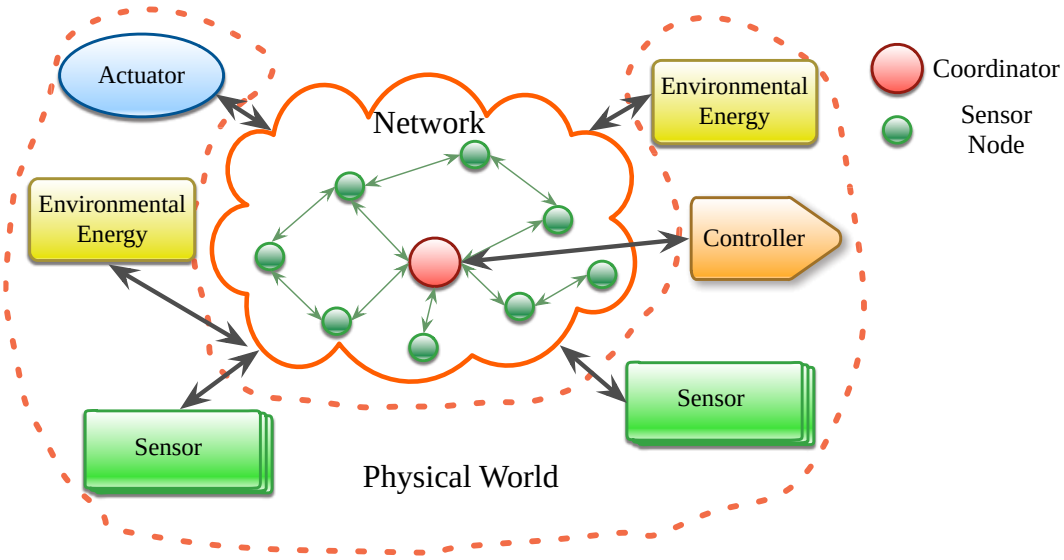


Figure 3.2: Illustration of the interaction between a WSN and the physical world.

To evaluate the performance of protocols and the feasibility of WSN deployment scenarios, researchers around the world have developed in recent years a wide variety of simulation frameworks. These simulators focus mainly on protocol modeling and differ in many aspects such as coding language (C, C++, Java,

etc.), graphical interface, scalability, performance, accuracy, re-usability, hardware & software emulation, documentation, etc.

The most popular WSN simulators are NS-3 [81], OPNET [82], WSNet [83], and Castalia [84] a simulator based on OMNet++. To meet the demand of simulating a large number of sensor nodes, all of these simulators are based on event-driven simulation technique. In event-driven simulation, the simulation time and node states are updated only when an event occurs (i.e. packet reception, transmission, etc.). This approach allows the simulator to process a huge number of events but fails to deal with continuous-time systems such as energy harvesters.

Moreover, most of these network simulators offer only a linear battery model [84] where the energy is decremented at each power consuming event. It is difficult, using such a simple model, to account for the complex behavior of an energy harvesting unit also including the storage devices. Modeling such a unit requires taking into account the time varying environmental parameters, the harvester and converter efficiencies, the power management algorithms and the battery charge and discharge behavior. In addition, the network and energy harvesting events may be completely uncorrelated.

To enable the simultaneous simulation of network protocols and continuous-time physical systems, there are two main approaches: The first approach consists in extending the features of existing simulators and enhancing their capabilities for continuous-time simulation. The second approach involves the integration of specific tools designed for the simulation of physical systems within a co-simulation framework.

3.2.1 *Extended Simulators*

In this section, we briefly introduce several tools based on the extended simulator approach.

3.2.1.1 *TrueTime Toolbox*

TrueTime [85] is a Matlab/Simulink-based simulator that provides models for the simulation of real-time networks and is widely used to simulate networked embedded control systems. TrueTime has been extended to support networks such as Bluetooth and ZigBee [86], and recently integrated with PiccSIM [87] that will be described in later.

Compared to existing WSN simulators, the network simulation models used in TrueTime are quite simplistic and support only the simulation of physical and medium access layers. This limits its usage in the simulation of WSNs that require higher-layer network protocols.

3.2.1.2 *Proweler*

Prowler [88] is a probabilistic discrete-event simulator running under Matlab which can simulate WSNs, especially MICA [89] motes, from the application to the physical layer. This simulator provides an accurate radio layer including fading models with packet collisions, static and dynamic asymmetric radio links. However, this simulator lacks several important features such as accurate energy models and many radio and MAC functionalities of the IEEE 802.15.4 standard.

3.2.1.3 *GreenCastalia*

GreenCastalia [90] is an extension of the energy model of the Castalia simulation framework targeting the simulation of energy harvesting WSNs. The energy module of GreenCastalia consists of three submodules:

- Energy harvester, which models the energy harvesting system;
- Energy storage, which represents an energy storage device such as batteries or supercapacitors;
- Energy manager, which models the control of storage charging and utilization.

GreenCastalia is a promising simulator and a good starting point for the simulation of energy-harvesting WSNs, but its current implementation is based only on simple models that fail to capture the real-world behavior of an energy harvesting system, such as the characteristics of the power management system (i.e. AC-DC or DC-DC converters), or the limited number of battery charge/discharge cycles.

3.2.1.4 *Energy Harvesting Module for the NS-3 Simulator*

In [91], the authors propose an extension of the energy module of the NS-3 simulator to support the simulation of energy harvesting WSNs. The energy module implemented in NS-3 is based on analytical models of a solar energy harvester and supercapacitors. The simulations performed show a good accuracy between the measurements and the analytical models, however, the global architecture of this simulator leaks the modularity capabilities essential to support the simulation of multi-source energy harvesting and multiple energy storage models.

3.2.1.5 *IDEA1*

IDEA1 [92] is a simulation framework aiming at the performance evaluation of sensor networks at a high level while taking into account every hardware and software component of a sensor node. IDEA1 is written in SystemC and is similar to existing emulators such as TOSSIM [93] and Avrora [94]. This simulator contains a library of numerous microcontrollers (Atmel ATMega128, PIC16LF88) and RF transceivers (TI CC2420, CC1000) that are modeled as finite state machines. In addition, these components are modeled as finite state machines. The accuracy of this simulator has been validated with experimental results and its performance has been compared to the NS-2 simulator.

3.2.2 *Co-simulation Frameworks*

This section is devoted to the review the available co-simulation platforms combining both a network simulator and a physical system simulator. The main challenge facing each co-simulation framework is to accurately model and simulate each component of a sensor node with a very low simulation time overhead implying an efficient synchronization protocol between the two tools.

3.2.2.1 *COSMO*

COSMO [95] is a co-simulation framework based on Matlab and OMNet++ which was developed to study the impact of 3-D indoor channels on wireless networks. Channel models developed in Matlab are compiled to shared libraries and headers files, and integrated into OMNet++. This approach does not require a synchronization technique, but may be, however, time consuming due to the process of recompilation of the OMNet++ or Matlab models being modified. Moreover, COSMO is not adapted to the simulation of EH-WSN requiring interactive simulation and clock synchronization between the WSN simulator and the energy harvesting system.

3.2.2.2 *Piccsim*

Piccsim [96] stands for *Platform for integrated communications and control design, Simulation, Implementation and Modeling*, is a co-simulation platform dedicated for research on wireless networked control systems. This co-simulation platform consists of one computer running NS-2 simulator and another computer running the PiccSIM toolchain and MATLAB/SIMULINK. The co-simulation and the communication between the two computers are coordinated by TCP/UDP packets using the Piccsim Toolchain. However, this platform does not support yet the simulation of energy harvesting WSNs.

3.2.2.3 *OPNET-Simulink*

The OPNET-SIMULINK co-simulation platform [97] combines OPNET and Matlab to study the effect of node movement on wireless networked control systems. Similarly to Piccsim, this platform requires two computers, the first running OPNET and the second running Matlab. Both simulators are executed in parallel and are synchronized using UDP sockets. In both of these co-simulation platforms, much development effort is needed to implement the synchronization mechanism between the two computers.

3.3 CHALLENGES FOR ENERGY HARVESTING WSN MODELLING AND SIMULATION

The simulation of WSNs involves modeling its different building blocks including hardware (i.e. micro-controller, transceiver), software (i.e. OS, middle-ware, firmware), and energy sources. Different languages and computation models are often used to describe these blocks. This often results in multi-language heterogeneous simulators. While homogeneous approaches exist, where a single language is used to describe the entire system, the main issue here consists in model simplifications that may not correctly describe the behavior of each subsystem, especially for a heterogeneous system like an EH-WSN.

In the case of energy harvesting WSN, the main problem faced is to simulate the network protocols in parallel with the environmental parameters, energy harvester, conditioning circuits and storage blocks. Such an integral role of energy in WSNs motivates many questions. Is there an energy module model that can be reused for all simulation scenarios in WSNs? If so, what are the parameters of this model? In our opinion, the variety of environmental energy sources and storage elements implies that there cannot be a unique energy module model suitable for all WSNs simulation scenarios. If this is the case, then the next question is: Does the chosen energy module model, from the huge number of energy transducers and storage elements available, fulfill the network lifetime objective for the applicative scenario? Energy harvesting subsystems are not easy to model in existing WSN simulators. Some, particularly those involving realistic properties of the energy harvester and the environmental data [15] [98], are very challenging to implement in a network simulator using a high level programming language such as C/C++ or Java. How can we ease this difficulty? How can we make the energy module module models accurate and realistic?

These are some of the questions that form the basis of the research contributions presented in this chapter.

3.4 THE HARVWSNET CO-SIMULATION FRAMEWORK

3.4.1 *Co-simulation framework*

The co-simulation framework presented in this chapter aims to provide a comprehensive solution to simulate energy harvesting WSNs. The co-simulation framework, HarvWSNet, combines the two following simulators:

- Matlab is the most frequently used simulator in research and industry. This simulator offers a wide range of accurate models and lots of powerful toolboxes dedicated to the simulation and analysis of continuous-time systems;
- WSNet is a precise WSNs simulator which offers sufficiently detailed models of the radio channel and physical layer [21].

In HarvWSNet, the MATLAB environment is used for modeling the energy harvesting system and for data visualization while WSNet is used to implement the communication protocol layers and simulate the network behavior. Since WSNet does not provide any facility to directly connect to Matlab, the major extension of the WSNet simulator was the addition of a communication interface by mean of TCP sockets that permit data to be exchanged periodically with Matlab. The goal of HarvWSNet is to evaluate the power charge/discharge profile of energy harvesting nodes in realistic scenarios. To this end, the simple battery model of WSNet is replaced by a function which calls directly the energy harvesting system implemented in MATLAB. The global architecture of the proposed framework is represented in Figure 3.3.

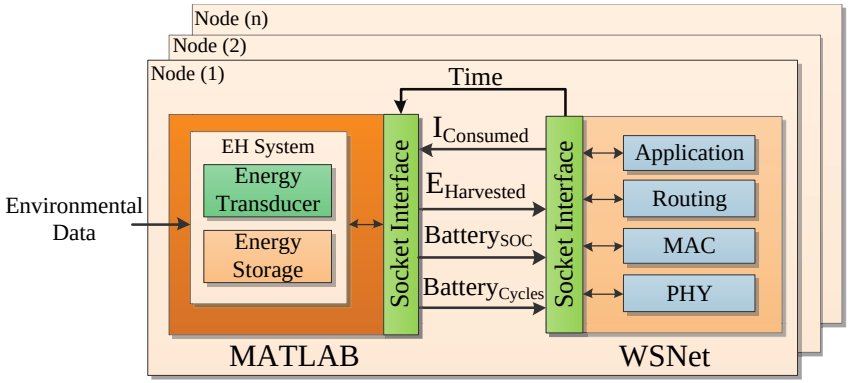


Figure 3.3: Structure of HarvWSNet.

In the co-simulation process, both MATLAB and WSNet operate paralelly and interactively. A simple synchronization process between these tools was implemented in HarvWSNet which requires only a single computer. The simulation is controlled by WSNet which defines the master clock and is responsible for clock synchronization.

Even if, in HarvWSNet, Matlab is used to model only the energy harvesting subsystem, the link between WSNet and Matlab could be exploited for other tasks, e.g. to simulate complex channel models or elaborate RF transceiver models or for data visualization and hence open up a wide range of additional capabilities for WSNet.

3.4.2 *Energy Harvesting Subsystems*

The energy harvesting system implemented in Matlab is completely modular and can adapt to any type of harvested energy from simple to more complex models. In addition, the architecture of the energy harvesting system supports multi-source energy harvesting capabilities, and allows the sensor node to harvest from different energy sources. Figure 3.4 illustrates the general architecture of the energy harvesting system.

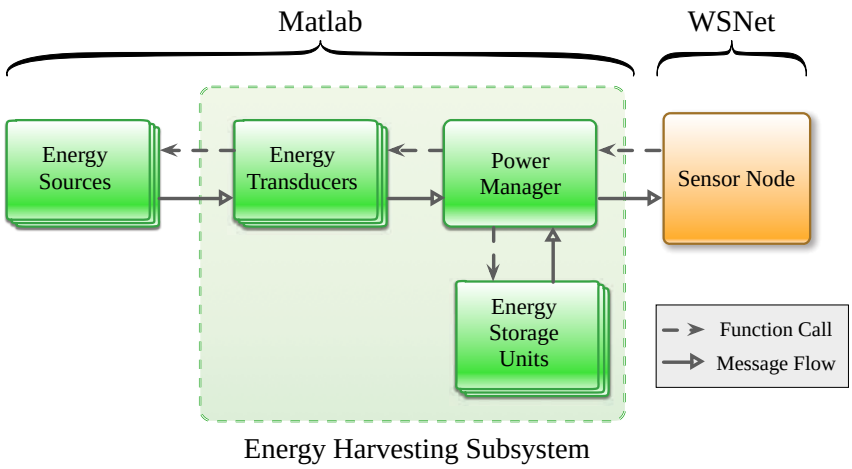


Figure 3.4: General structure of the energy harvesting system.

The energy harvesting system consists of four blocks:

- Energy source contains data files that fully describe the energy available in the environment of the sensor node such as solar irradiance, temperature, vibrations, etc. These data files are called directly by the energy transducer.
- Energy transducer converts the data fed by the energy source block to an electrical energy and includes also all the circuitry (i.e. AC-DC, DC-DC converters, MPPT functions) responsible for enhancing the transducer's efficiency and producing the desired output voltage.
- Power manager is responsible for controlling and routing the energy from the energy transducer to the energy storage units and/or to the sensor node. This block may have energy measurement capabilities that track the voltage

status of the energy transducer and energy storage elements. The power manager provides to WSNet information about the instantaneous power available for the powering of the sensor node.

- Energy storage, composed of either batteries or supercapacitors, are devices that store the energy collected by the energy transducer and supply energy to the sensor node if/when it is not powered by the collected ambient energy.

To accelerate the simulation process, the Matlab model can be compiled into a MEX function which is also called directly by WSNet through the TCP socket.

3.4.3 *Co-simulation process*

An energy harvester is a dynamic system in which the battery state-of-charge (SOC) and energy harvested change in a continuous manner with respect to time. On the other hand, WSNet is a discrete-event driven simulator that handles events that are unevenly distributed in time due to the stochastic nature of packet generation. We therefore define two types of events at which WSNet invokes Matlab. The first, called *synchronization events*, are scheduled at fixed intervals and dedicated to maintaining the synchronization of the simulation clocks between WSNet and Matlab. The second type of events, called *communication events*, arise when a node is in one of the following states: transmitting, receiving and listening, or in any other power consuming state.

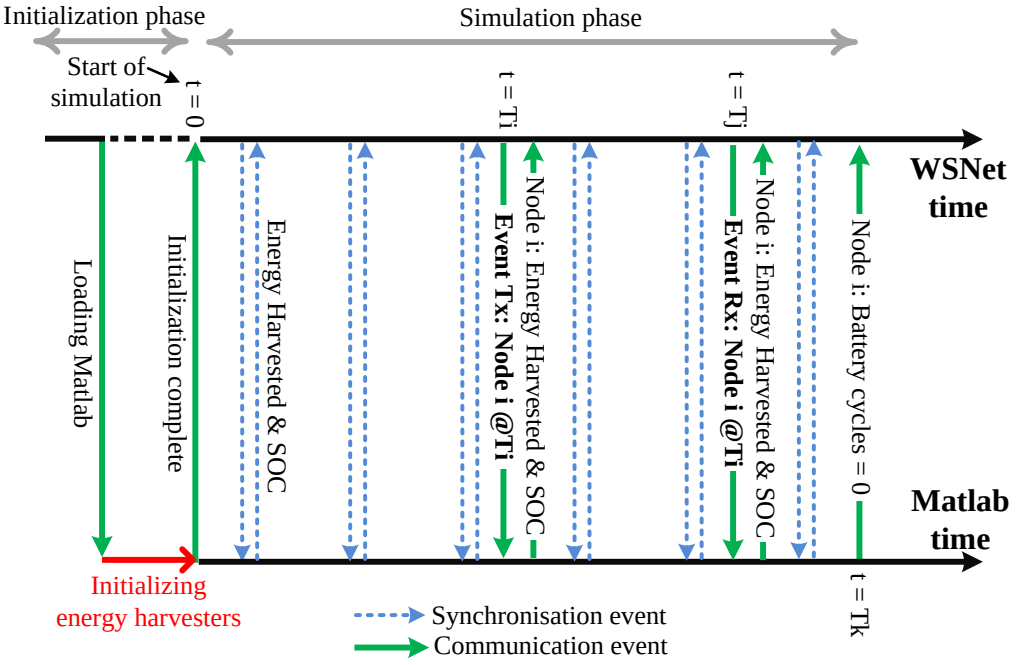


Figure 3.5: Interactive co-simulation of WSNet and MATLAB.

The following steps and Figure 3.5 provide an overview of a typical exchange between WSNet and Matlab: During the initialization phase which takes place at the initial time but before the actual simulation starts, WSNet creates the nodes specified in the simulation configuration file and requests a connection to the Matlab engine via sockets. Once the connection is setup, Matlab initializes the energy harvesting system that corresponds to each simulated node and associates the predefined environmental data in accordance with the location of each node. After initialization, WSNet begins the simulation at time zero and passes the clock to Matlab each T seconds allowing the energy harvester of each node to calculate the scavenged power, the battery SOC and the number of remaining charge/discharge cycles. The period T is controlled by WSNet and is chosen as a function of the communication duty-cycle and the environmental data type. When a node transmits or receives a packet, WSNet sends a notification to Matlab and waits until the Matlab calculation is done before continuing. The notification includes the node number, the kind of event (e.g. reception or transmission), the event duration, and the current to draw from the battery. Otherwise, if the node is not receiving nor transmitting data, the energy harvesting system considers that the node is in idle mode and draws the current corresponding to this state. When the battery associated to a node reaches its maximum number of charge/discharge cycles, Matlab sends a notification to WSNet indicating that the node is not operational and should be removed from the network.

In the following sections, we describe two case studies using HarvWSNet, the first one focuses on the lifetime estimation of a solar energy harvesting WSN while the second is dedicated to the evaluation of the feasibility of a structural health monitoring WSN using air-flow energy harvesting. The models used in the two case studies are based on the datasheets of COTS devices.

3.5 CASE STUDY 1 : SOLAR POWERED GREEN HOUSE TEMPERATURE MONITORING

The application scenario described in this section is a solar powered sensor network for temperature monitoring in a green house. The main constraint in this scenario is the energy consumption.

3.5.1 *Solar Energy Harvester Module*

The aim of this section is to develop a model that not only is as realistic as possible but also that closely resembles existing energy harvesting platforms. The flexibility of the HarvWSNet framework eases the development of highly complex models

that can be refined to take into account advanced power management and prediction strategies. Since the applicative scenario that will be described below is temperature monitoring application in a greenhouse, the energy scavenger chosen is a solar harvester consisting of a photovoltaic (PV) panel, a battery and a power manager as illustrated in Figure 3.6. Thus, the environmental data used are solar irradiance and temperature. This information is loaded at the initialization phase (Figure 3.5).

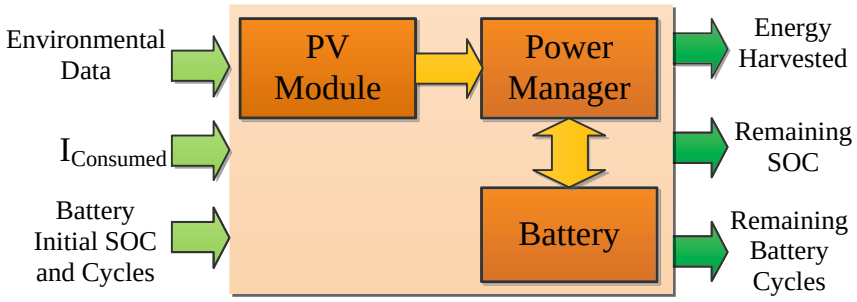


Figure 3.6: Global architecture of the energy harvesting system.

When MATLAB receives a synchronization or a communication event, the solar harvester system calculates the solar energy scavenged, the state-of-charge (SOC) and remaining cycles of battery as function of the environment data, the current drawn by the node, the duration and type of power consuming event, and both the initial SOC and battery cycles. The models for the PV panel, battery and power manager will be described next.

3.5.2 *Solar Harvester Modeling*

Using the irradiance and temperature data, the solar harvester estimates accurately the current and voltage that is delivered to the sensor node or to the battery. Figure 3.7 represents the circuit diagram of the solar harvester which consists of a photovoltaic panel and a maximum power point (MPP) block.

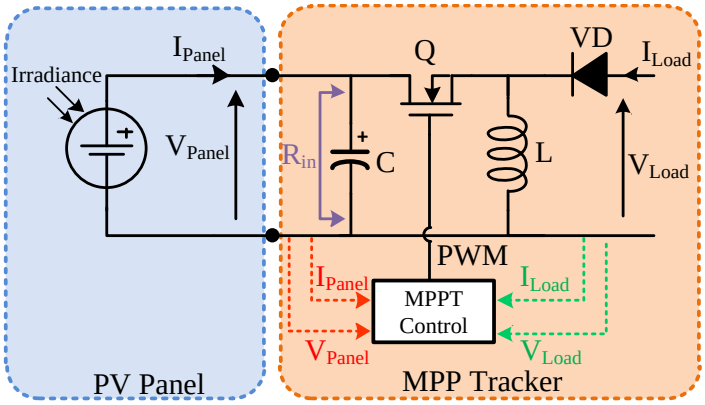


Figure 3.7: Block diagram of solar energy harvesting system.

3.5.2.1 *PV Panel Modeling*

A PV panel is composed of numerous connected PV cells. The electrical model for a PV cell is typically modeled using the one-diode model as shown in Figure 3.8, where I_{PH} denotes the photo-generated current and R_S and R_P are series and shunt resistances.

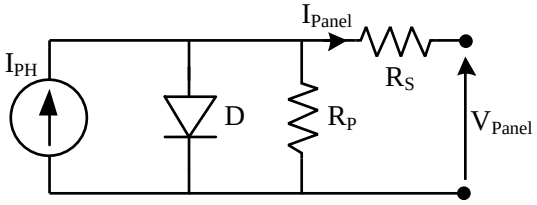


Figure 3.8: Equivalent electrical circuit of a photovoltaic cell.

Manufacturers typically provide only few operational data for photovoltaic panels, such as the open circuit voltage (V_{oc}), the short circuit current (I_{sc}), the maximum power point voltage (V_{mpp}) and current (I_{mpp}), the temperature drift coefficients at open circuit voltage and short circuit current. The PV panel *MULTICOMP MC-SPo.8-NF-GCS* provides a nominal output power of 800 mW under full outdoor irradiance. Assuming that the resistance R_P is infinite, the current voltage characteristic of the PV panel may be written as [99]:

$$I_{Panel} = I_{PH} - I_S \cdot \left[\exp\left(\frac{q\,(V_{Panel} + I_{Panel} \cdot R_S)}{N_S\, a k T} \right) - 1 \right] \tag{3.1}$$

where N_S is the number of cells in series, I_S is the saturation diode current, a is the diode quality factor, T is the absolute temperature and q and k are the electron charge and Boltzmann constant, respectively. The three parameters I_{PH}, I_S and R_S can be determined from the operational data provided on the manufacturer datasheet [100].

Since the available solar energy varies with time of day, a maximum power point (MPP) tracking mechanism is essential to take full advantage of available solar energy by continuously delivering the highest possible power to the battery or to the node. As shown in Figure 3.9, the I-V characteristic of the PV panel is nonlinear and changes with irradiance.

3.5.2.2 *MPPT Modeling*

The MPP tracker block consists of two parts: a DC-DC buck-boost converter and an MPP tracking (MPPT) control algorithm. The function of the DC/DC converter is to provide appropriate power to the load (battery or node) and compute the MPP according to the control algorithm. The converter input resistance R_{in}, which is the load resistance presented to the PV panel, varies according to the duty cycle D

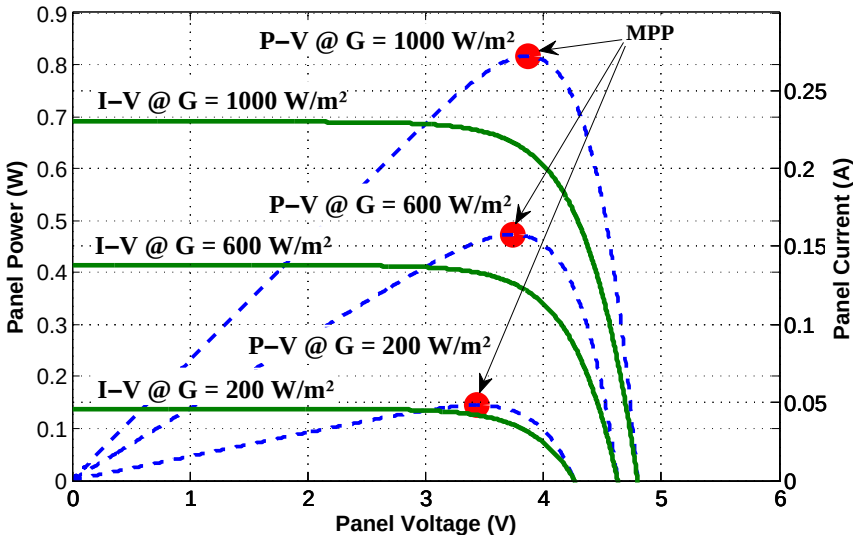


Figure 3.9: PV panel current-voltage and power-voltage characteristic.

of the pulse width modulation (PWM) signal delivered by the MPPT control block and is therefore given as [101].

$$R_{in} = \left(\frac{1-D}{D}\right)^2 \frac{V_{Load}}{I_{Load}} \tag{3.2}$$

where V_{Load} and I_{Load} are respectively the converter output current and voltage. The output voltage delivered to the load is given by

$$V_{Load} = \frac{D}{1-D} V_{PV} \tag{3.3}$$

The MPPT controller is implemented using the widely used algorithm Perturb and Observe (P&O) [102]. This algorithm measures the power generated by the PV panel and finds and tracks its maximal value by varying the converter duty cycle.

3.5.3 *Battery Model*

There exist several models in the literature that capture battery charge/discharge behavior with reasonable reliability. For its simplicity and accuracy, we consider only the electrical circuit model based on previous work done by [103]. This model can predict both the $I-V$ performance and the lifetime, and includes two important battery properties: the capacity fading effect due to charge-discharge cycles and temperature. The equivalent circuit model of the battery is represented in Figure 3.10, where V_{oc}, $V_{battery}$ and $I_{battery}$ are, respectively, the output open circuit voltage, the output voltage and the current of the battery.

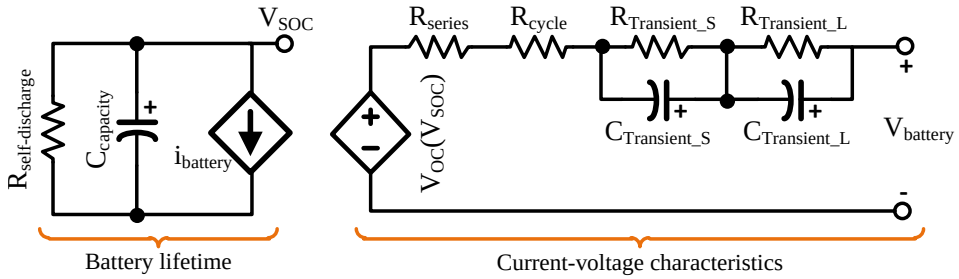


Figure 3.10: The electrical battery model.

The battery used in the application below (*TCL PL-383562*) has a capacity of 850 mAh, with 400 cycles of charge and discharge. The expression of battery state of charge (SOC) is described by

$$SOC = SOC_{init} - \int \frac{I_{battery}}{C_{usable}}\, dt \tag{3.4}$$

where the SOC_{init} is the initial battery SOC and C_{usable} the usable battery capacity which is given as

$$C_{usable} = C_{init}.(1 - \text{Calendar life losses} - \text{Cycle life losses}) \tag{3.5}$$

where C_{init} is the initial battery capacity, *Calendar life losses* is the storage losses occurring when the battery is not used and *Cycle life losses* is the capacity fading due to cycle number.

The parameters of the two RC networks vary as a function of the SOC. The total impedance of the RC network is

$$\begin{aligned} Z_{Battery}(SOC) &= R_{series} + R_{cycle} \\ &+ R_{Transient\_L}\left(1 - \exp\left(-\frac{t}{R_{Transient\_L}C_{Transient\_L}}\right)\right) \\ &+ R_{Transient\_S}\left(1 - \exp\left(-\frac{t}{R_{Transient\_S}C_{Transient\_S}}\right)\right) \end{aligned} \tag{3.6}$$

Therefore, the battery I-V characteristic can be expressed as a function of voltage drop due to the battery equivalent internal impedance $Z_{Battery}$ and V_{oc}.

$$V_{battery} = V_{oc} - i_{battery} \times Z_{Battery}(SOC) \tag{3.7}$$

The battery is charged each day by the PV panel and is discharged by the load during the night. A battery cycle occurs when the battery is completely discharged and then fully recharged. We consider that the lifetime of a battery is the number of charge-discharge cycles that the battery will sustain. The battery is supposed to be unusable when its capacity drops to 80% of its original rated capacity.

3.5.4 *Power Manager*

The power manager adopted is based on an ideal controller which manages the energy harvested and node power supply mode (solar harvester or battery) while ensuring that the battery is not overcharged nor over-discharged (10% $\leqslant$ SOC $\leqslant$ 95%). Additionally, this power manager periodically indicates to WSNet the energy source used, either solar harvester or battery. This information allows the sensor node to adapt its operations to the available power supply. The flowchart of the power manager is shown in Figure 3.11.

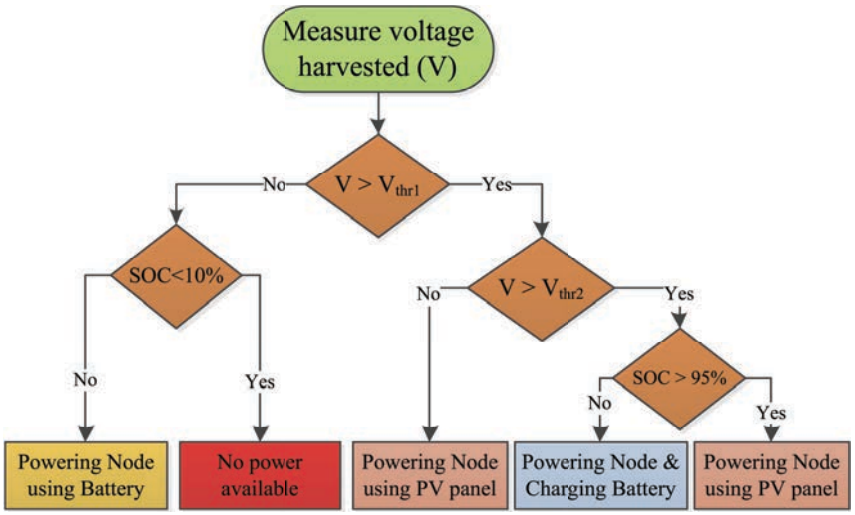


Figure 3.11: Flowchart of the power manager.

Here, $V_{Thr1} = 3.6V$ and $V_{Thr2} = 4V$ are chosen with respect to the operating voltage of the sensor node and the battery. At reception of a synchronization or communication event from WSNet, the power manager measures the voltage delivered by the solar harvester. This voltage and the battery terminal voltage are compared to the configured thresholds V_{Thr1} and V_{Thr2}, the power manager determines which power source, either solar harvester or battery, should power the node. If the scavenged voltage is higher than V_{Thr2}, the power manager connects the solar harvester to the node and charges the battery only if the battery SOC is below 95%. Otherwise, if the PV panel voltage is lower than V_{Thr2} but higher than V_{Thr1}, the power manager connects the node directly to the PV panel. In the case where the voltage is below V_{Thr1}, the node is powered using the battery. Figure 3.12 shows the evolution of the solar harvester and battery voltages during 4 days, from July 1 to July 4, based on the solar irradiance data of Grenoble given by [104].

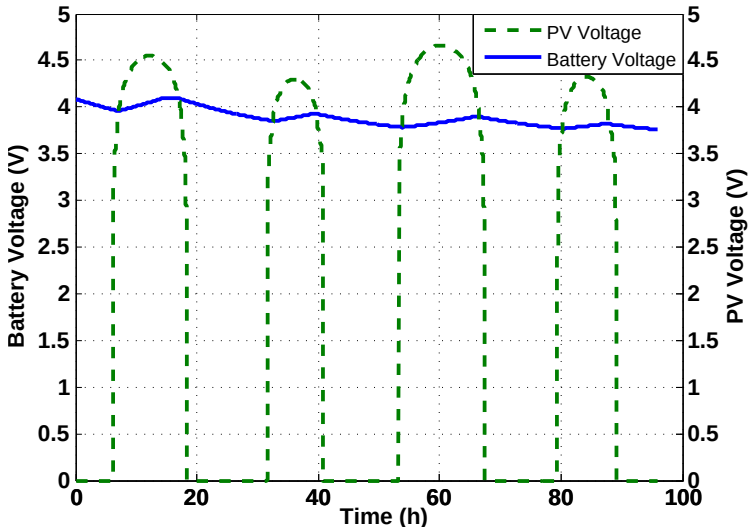


Figure 3.12: Evolution of voltage in the energy harvesting system over 4 days.

3.5.5 *Simulation Results*

In this section, we study the lifetime of a WSN for a temperature monitoring scenario located in a greenhouse using HarvWSNet. The simulated network consists of 8 sensor nodes distributed in a circular area around a base station as illustrated in Figure 3.13.

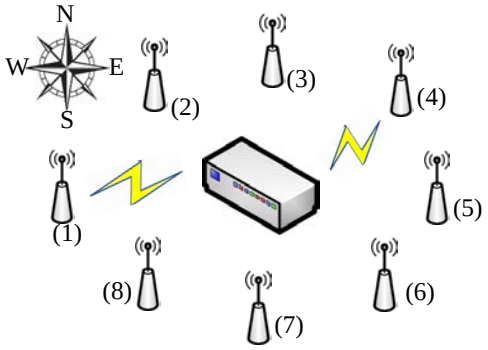


Figure 3.13: Simulation set-up.

The communication distance between a sensor node and the base station is 40 m at a frequency of 2405 MHz and a data rate of 250 kbps (IEEE 802.15.4). The MAC protocol used is CSMA/CA. All simulated sensor nodes are identical and integrate the following functions: a temperature sensor, a micro-controller for data processing, an RF transceiver to communicate with the base station, and the energy harvester described in Section 3.5.1 for power.

We associate different one year irradiance data with each node depending on its position and solar panel orientation. The base station is assumed to have an unlimited power source. We consider the power consumption model of the board

eZ430-RF2500 combining an RF transceiver CC2500, a micro-controller MSP430 and a temperature sensor [24]. The nodes are configured to sense and transmit the temperature with a 1 second transmission period to the base station. The average current consumed during sleep mode is given as

$$\begin{aligned} I_{sleep} &= (I_{idle}(CC2500) + I_{idle}(MSP430)) \times (T_{App} - T_{Tx}) \\ &= (1.3[\mu A]) \times (1[s] - 0.002383[s]) \\ &= 1.296\ \mu A.s \end{aligned} \tag{3.8}$$

Therefore, the average current consumption of the application over 1 second is

$$\begin{aligned} I_{avg} &= (I_{sleep} + I_{Tx}(MSP430))\ /1[s] \\ &= (1.296[\mu A.s] + 35.508[\mu A.s])\ /1[s] \\ &= 36.80\ \mu A \end{aligned} \tag{3.9}$$

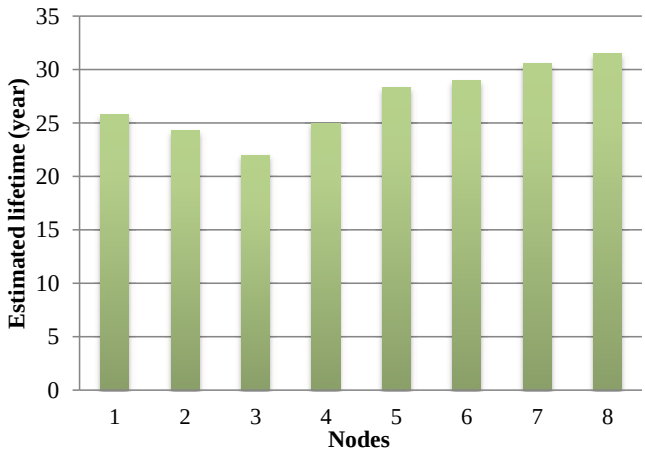


Figure 3.14: Estimated lifetime of each sensor node.

Figure 3.14 shows the estimated lifetime for each sensor node assuming that both the power manager and the MPP tracker consume 1 mW. We observe that certain nodes have longer lifetimes due to the solar data used which depends on solar panel orientation. The nodes facing south receive higher solar irradiance than nodes facing the north. Concerning battery lifetime, we have observed that most battery charge-discharge cycles are consumed during the winter months where the nodes do not receive sufficient solar irradiance. Even if the previous observations are predictable, HarvWSNet is the only simulation platform capable of producing accurate lifetime projections. Most importantly, for the same scenario described above, the processing time taken to simulate one year using WSNet with a simple linear battery model is about 3 hours, compared to 4,5 hours using HarvWSNet. This extra processing time is clearly worth the price since the evaluation of an energy harvesting node's lifetime is impossible using a linear battery model.

3.6 CASE STUDY 2: WIND-POWERED SHM APPLICATION

3.6.1 *Scenario Description*

As stated in Chapter 2, one of the many applications that can benefit from WSN is structural health monitoring (SHM). SHM consists of accurately diagnosing the health of structures in order to prevent potential damages of its materials and asses its remaining lifetime.

The main goal of this case study is to investigate, using HarvWSNet, the feasibility of a SHM for an industrial client who wishes to monitor the state of its signalisation equipment installed on the ceiling of a subway metro line. In all, there are 183 such signalisation stations along the 24.3 kilometers of the line, or one approximately every 130 meters. The client wishes to monitor the local temperature around each station and detect if a particular equipment has suffered an impact. A solution consists in attaching to each station a self-powered wireless SHM (Structural Health Monitoring) node equipped with a temperature sensor and accelerometer. A possible source of energy in the subway tunnel is the wind generated by each train passage.

The pre-prototyping simulation must be able to quickly respond to the following questions: Can the wind generated by the passage of subway trains near each wind harvester furnish enough energy to power the nodes? If so, what network architecture is necessary to support this application? Can nodes aggregate and relay data towards the nearest base station in a multi-hop mesh-type network? Are intermediate base stations necessary? Note that to minimise costs, the client prefers a minimal number of base stations. Given a network architecture, can the application guarantee a minimum data gathering period of 1 second, a minimal transmission period of 20 seconds, and a maximum data transmission delay of 1 hour? Is the network able to function 24 hours, even if the trains are at a stop between midnight and 5h00 in the morning? How long can the network continue working if the train traffic is suddenly stopped?

3.6.2 *Scenario component models*

In order to model and simulate this scenario, a number of models were required. These include models for the wind energy harvesting module, for the RF signal propagation in the tunnel, models for the MAC, routing and power management protocols, and a model for the train mobility and for the node power consump-

tion. Whenever possible and for better precision, the models are built based on measured results as described below.

3.6.2.1 *Wind Energy Harvester Model*

In this scenario, train movement in the tunnel generates wind that is harvested to supply the sensor nodes. The block diagram of the Matlab model for the wind-flow energy harvester is given in Fig. 3.15. Measurements made by [105] in a Rome metro tunnel with a small 6.3cm diameter micro-wind turbine and an adaptive rectifier placed on a side wall of the tunnel showed that approximately $E_{harvested} = 60mJ$ can be harvested over a 20 second period at each train passage, assuming that the rectifier is loaded by an MPPT (Maximum Power Point Tracking) block and that the node is placed far from a tunnel entrance. Therefore, in the model, at each train passage event, a corresponding input power profile is injected into the MPPT block whose role is to present the correct input impedance while providing a charging current for the 4.7F supercapacitor. Finally, a boost DC/DC converter provides the 3.3 Volts required by the node.

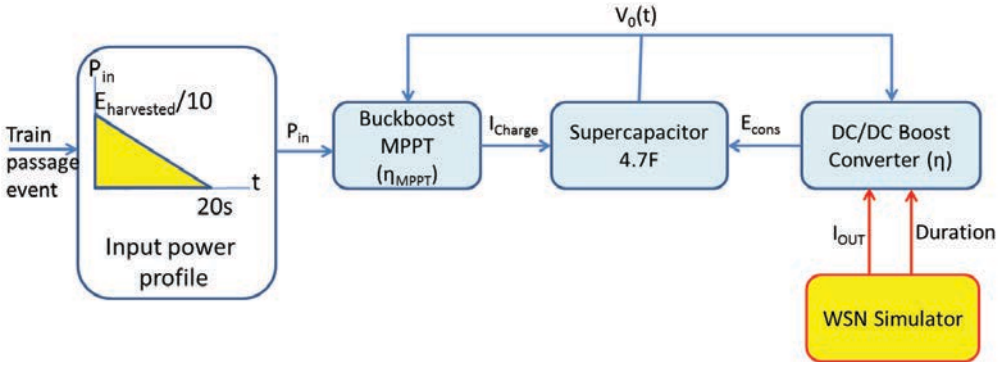


Figure 3.15: Wind-flow Energy Harvester Matlab Model.

The MPPT block transfers the input power to the supercapacitor with an efficiency of MPPT η_{MPPT} which depends on the supercapacitor voltage $V_0(t)$. The definition of η_{MPPT} is obtained by fitting onto the measurements presented by [106] of a buck-boost converter operated in Fixed Frequency-Discontinuous Current Mode (FF-DCM) with a switching frequency of $f = 28kHz$ which has the best performance when $V_0(t)$ is in the range of $0.9V$ to $2.3V$ (i.e. the most common range of supercapacitor voltages in our scenario.)

$$\begin{aligned}\eta_{MPPT} =& 0.3424V_0(t) + 0.5542, \quad V_0(t) < 0.88 \\ =& 0.5133V_0(t)^3 + 2.7098V_0(t)^2 - 1.0308V_0(t) \\ &+ 0.7871, \quad V_0(t) >= 0.88\end{aligned} \tag{3.10}$$

The resulting model (3.10) is compared to measurements [106] in Figure 3.16.
The buck-boost block generates a charging current I_{Charge} defined by

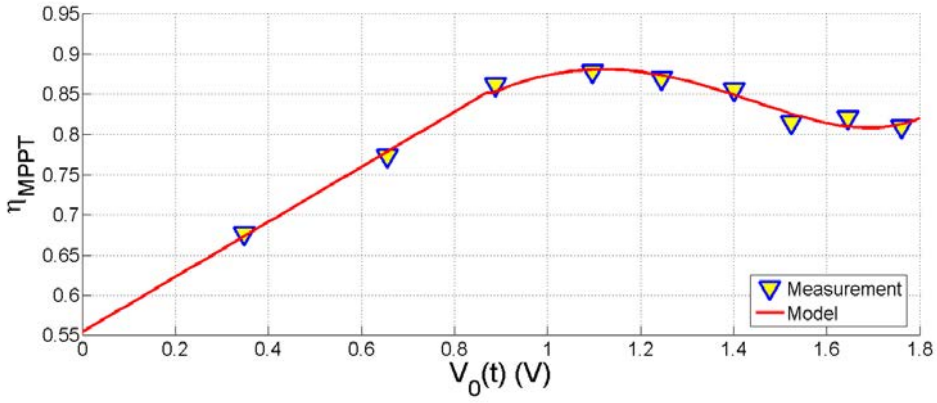
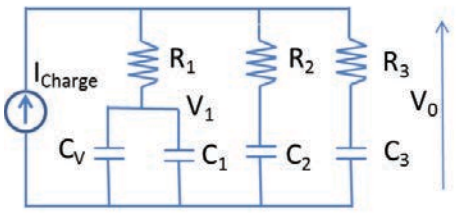


Figure 3.16: Buck-boost converter efficiency η_{MPPT} as a function of output voltage $V_0(t)$ with switching frequency of 28kHz.

$$I_{Charge} = \int_{t}^{t+T_{Harv}} \frac{\eta_{MPPT} P_{in}}{V_0(t)} dt \tag{3.11}$$

where P_{in} is the input power and T_{Harv} is the harvesting period (defined below).

A crucial part of our harvester model concerns the supercapacitor which suffers from large redistribution currents which, as explained in [16], are often mistaken as leakage currents. In our work, correctly modeling these currents is essential since large leakage currents could completely discharge the supercapacitor during the hours when no trains are circulating whereas redistribution currents tend to zero when steady-state is reached. To correctly account for this effect, [16] proposes the multi-time constant supercapacitor model (Fig. 3.17).



Parameter	Value
CV	0.945F/V
C1	2.62F
C2	1.45F
C3	3.88F
R1	0.178Ω
R2	94.2Ω
R3	1030Ω

Figure 3.17: Supercapacitor Model proposed by [16].

The values of the branch voltages and currents can be obtained by solving the corresponding system of differential equations for each value of I_{charge}. This is achieved simply thanks to the *ode()* function in Matlab. When $I_{charge} = 0$, only charge redistribution currents are observed within the supercapacitor. In this work,

a supercapacitor of 4.7F is used with the model parameters given by [16]. At full-charge, this supercapacitor has a maximum voltage, V_{MAX}, of 2.3V.

For simplicity, supercapacitor discharge events are treated differently. When required, the energy consumed E_{cons} is directly removed from the first branch of the supercapacitor model and the value of $V_1(t)$ is updated as follows:

$$E_{C_V//C_1}(t) = E_{C_V//C_1}(t-1) - E_{cons} \tag{3.12}$$

$$E_{C_V//C_1}(t) = \frac{1}{2} * (C_V * V_1(t-1) + C_1) * V_1(t)^2 \tag{3.13}$$

where $E_{C_V//C_1}(t)$ is the energy of the $C_V//C_1$ branch at time t.

Finally, a DC/DC boost converter is used to supply a stable voltage of 3.3V for the sensor node. The Texas Instruments DC/DC Boost Converter TPS61221 [17] with the start-up voltage of 0.7V is used as the basis for our model. If the node has drained a given current I_{OUT} for a given $Duration$, the total energy consumed, assuming a DC/DC efficiency η, is given by

$$E_{cons} = \frac{I_{OUT} * 3.3 * Duration}{\eta} \tag{3.14}$$

where η is modeled (3.15) by fitting onto the measured efficiency of the TPS61221, supposing that supercapacitor voltage varies around 1.2V (Fig. 3.18).

$$\begin{aligned} \eta =& 114.16 \log(I_{OUT}) + 245.63, \quad I_{OUT} < 30mA \\ =& 2.363 \log(I_{OUT})^3 - 2.719 \log(I_{OUT})^2 \\ &+ 0.3273 \log(I_{OUT}) + 82.462, \quad I_{OUT} > 30mA \end{aligned} \tag{3.15}$$

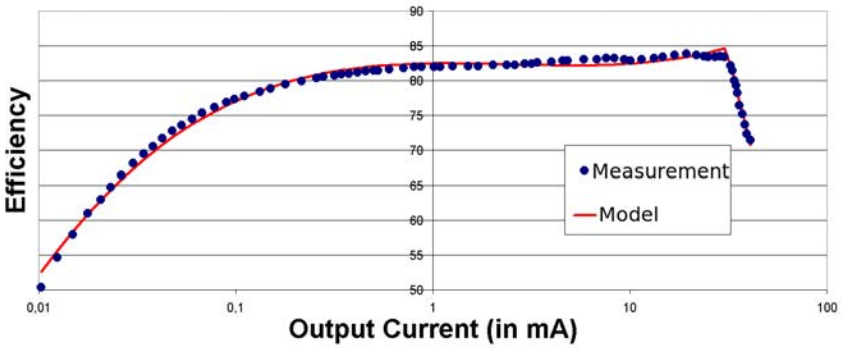


Figure 3.18: Efficiency of DC/DC boost converter block η as a function of output current I_{OUT}. Measurements are from [17].

3.6.2.2 *In-Tunnel Propagation Channel Model*

RF signal propagation in tunnels has been considerably explored in the literature. Differently from indoor and outdoor propagation channels, tunnels provide a better than free-space average pathloss but with deep fading zones which are often

of a periodic nature for larger distances. Recently, a geometric channel model for tunnels based on the Ray Tracing approach was proposed [107]. The model inputs are the RF channel center frequency, the tunnel width and height, and finally the permittivity and conductivity of the wall material. In our application, the tunnel vertical and horizontal dimensions are 4m and 7m, respectively, and the wall permittivity and conductivity are respectively $9\epsilon_0$F/m and 0.1S/m. The 2.4 GHz channel model that will be used in the rest of this work is given in Fig. 3.19. Finally, the presence of a subway train between two sensor nodes is assumed to cause an additional pathloss $\mathtt{Train}_{\mathtt{ATT}}$ whose value is swept between 0dB and 100dB (complete signal loss).

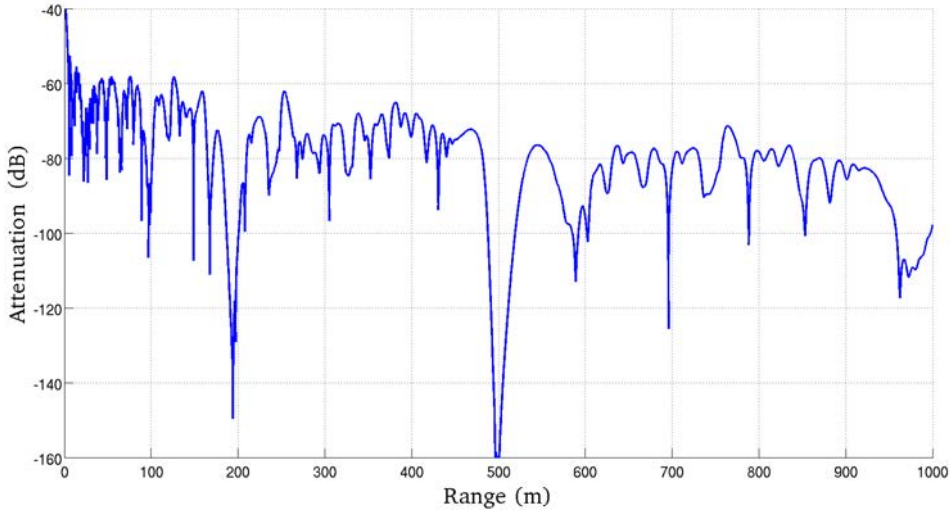


Figure 3.19: In-tunnel channel attenuation model at 2.4 GHz.

3.6.2.3 *Network Architecture and Routing Model*

The average inter-node distance of 130 meters all along the length of the tunnel clearly points to the use of a peer-to-peer data aggregating and relaying network architecture. In this first pre-prototyping attempt, nodes are therefore connected in a chain-like manner to a single base station which is placed at one end of the tunnel to simplify network maintenance. Since the objective of this work is to validate the 24 hour viability of the scenario assuming the wireless network is established, the route discovery protocol is not modeled. Second, since there is no node mobility, routing tables are established at the beginning of the simulation and are simply based on maximum received signal power. Since all nodes have the same emission power of 0dBm, nodes will therefore forward their packets to the uplink neighbor which has the lowest pathloss link. Note that, due to the channel model presented previously, this node may not be an immediate neighbor.

3.6.2.4 *Dedicated TDMA-like MAC Protocol*

While all nodes in our application employ the 2.4 GHz IEEE802.15.4 physical layer, in order to minimize idle listening time, a TDMA-like protocol is preferred over CSMA/CA. Each node is assigned to one of the 183 equal time slots of each transmission period T_{period}. In its assigned time slot, the node transmits to its uplink node (defined in its routing table) a packet that contains its own sensor data aggregated to that of all of its downlink nodes. Nodes know in which time slot(s) they must wake up in order to receive the data packets they must relay. Time slots are assigned such that a relay node receives all of its incoming data before aggregating and sending its own data. If all packet transmissions are successful, all of the data generated by the 183 nodes are relayed to the base station within a single T_{period}. While this protocol requires that all nodes remain synchronized, it is not difficult to imagine a light-weight mechanism (not modeled here) by which all nodes follow the time drift of the root node.

Each data packet is constructed based on the IEEE802.15.4 physical layer (bit rate of 250kbps) which has a 6 byte PHY header and a maximum PSDU (Physical Layer Service Data Unit) size of 127 bytes. Our MAC protocol overhead includes a 5 byte header and 2 byte CRC (FCS field). Each packet can therefore convey up to 120 bytes of useful data. Every T_{period}, each node fabricates a 5 byte data unit in which are mapped the node ID, timestamp, sensor data and supercapacitor state. A PHY data packet can therefore contain up to 24 such data units. Every data packet is immediately acknowledged using an 11 byte ACK packet. The data contained in unacknowledged packets is placed in a FIFO queue for retransmission in the follow transmission period. Since every receiving node knows precisely when it must receive a packet (DATA or ACK), idle listening is minimized using a 100µs timeout period.

Due to data aggregation, nodes which are nearer to the base station will have a greater number of data to transmit. In order to avoid data congestion, nodes may send up to 16 consecutive packets in a single time slot whose duration is fixed at 100ms. This is the time required for transmission of 16 maximum length packets ($133bytes@250kbps = 4.25ms$) including packet acknowledgement (352µs), transceiver turnaround (192µs), radio standby (0.77ms), maximum data processing time (10ms), and guard time (9.5ms). Figures 3.20 and 3.21 illustrate the MAC protocol used in this scenario.

3.6.2.5 *Mobility Model for Trains*

As in any subway system, the train passage period varies during the day (Table 3.1). No trains circulate between midnight and 5am. Trains are modeled as nodes

Figure 3.20: Definition of transmission period.

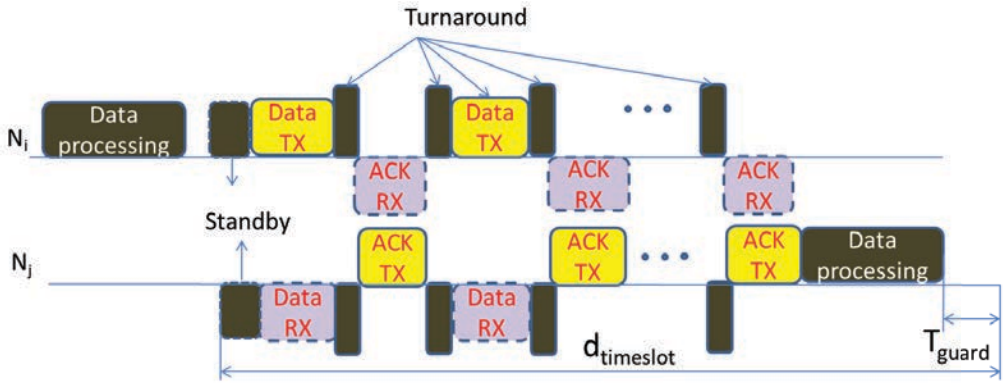


Figure 3.21: Dedicated MAC protocol - Timeslot definition

that travel at a constant speed of 50km/h in both tunnel directions. There is no wall separating the two train tracks.

Time	Train passage period
0am – 5am	No train
5am – 7 : 30am, 10am – 12am, 2pm – 5pm, 7pm – 8pm	240s
7 : 30am – 10am, 12am – 2pm, 5pm – 7pm	90s
8pm – 12pm	480s

Table 3.1: Train passage period.

3.6.2.6 *Power Aware Protocols*

In this application, the use of a simple network-level power manager (PM) is considered. Its aim is to adapt the transmission period of the network as a function of the residual energy of the nodes. The global PM algorithm running on the base station observes the 2-bit supercapacitor state data forwarded in each data unit and adapts the transmission period of the entire network based on the worst observed energy status (Figure 3.2). This decision is transmitted using the ACK packets which gradually reach all of the downlink nodes. A node applies the new transmission period decision immediately upon reception.

Supercapacitor voltage	Energy status	Transmission period T_{period}
$[1.9V - 2.3V]$	Full	T_0
$[1.5V - 1.9V]$	High	$2T_0$
$[1.1V - 1.5V]$	Low	$4T_0$
$[0.7V - 1.1V]$	Depletion	$8T_0$

Table 3.2: Network-level Power Manager Algorithm.

A simple node-level local power manager is also considered. It is assumed that a node can change its sensing period T_{SENSE} according to the trend of its supercapacitor voltage. Starting from a default T_{period}/T_{SENSE} ratio of 20, this ratio is updated exponentially within the limits $[1, 40]$ whenever the supercapacitor voltage increases or decreases by $20mV$.

3.6.2.7 *Power consumption model*

The power consumption model of the node is based on measurements [108] made on a TI-EZ430-SEH platform [109]. The current consumption and, if applicable, duration of each state is given in Table 3.3.

Parameters	**Values**	**Parameters**	**Values**
I_{sleep}	$1.16\mu s$	T_{sleep}	simulation
I_{TX}	$22.6mA$	T_{TX}	simulation
I_{RX}	$19.32mA$	T_{RX}	simulation
I_{Turn}	$20mA$	T_{Turn}	$196\mu s$
$I_{Standby}$	$8.23mA$	$T_{Standby}$	$0.77ms$
I_{PM}	$0.88mA$	T_{PM}	$2.5ms$
$I_{Sensing}$	$1.38mA$	$T_{Sensing}$	$19,76ms$
I_{Proc}	$1.6mA$	T_{Proc}	$61.4\mu s/byte$

Table 3.3: Power consumption model.

I_{sleep}, I_{TX}, I_{RX}, I_{Turn} and $I_{Standby}$ are consumed currents of nodes in sleep, transmission, reception, turn around and standby states and T_{sleep}, T_{TX}, T_{RX}, T_{Turn} and $T_{Standby}$ are the corresponding times passed in those states. $I_{Sensing}$, I_{Proc} and I_{PM} are consumed currents for data sensing, data processing and power management and $T_{Sensing}$, T_{Proc} and T_{PM} are the respective durations of these tasks. Here the power management task mainly consists in reading the supercapacitor voltage on one of the MSP430 ADC ports.

3.6.3 *Pre-prototyping with HarvWSNet*

In the present scenario, Matlab and WSNet interact on three types of event: harvesting period event, transmission period event, and train passage event (Figure 3.22). A harvesting period event is used to update the supercapacitor model of each nodes to account for charge redistribution and harvester charging currents when applicable. At each transmission period event, WSNet provides Matlab with the current consumption profile of the nodes in the preceding period and in return receives the energy status of the nodes (i.e. residual supercapacitor voltage). Finally, when a train passes a sensor node, a train passage event is triggered in which Matlab updates the harvested power profile of the given node.

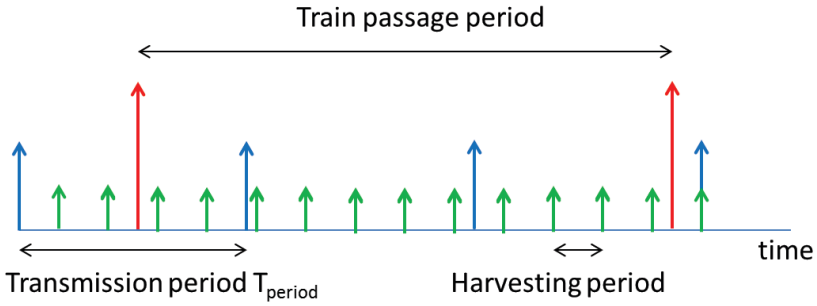


Figure 3.22: WSNet and Matlab Interaction.

The choice of harvesting period is a trade-off between simulation duration and accuracy of supercapacitor voltage update. When the harvesting period is small (1s), Matlab and WSNet communicate frequently heavily impacting the simulation duration. In contrast, when the harvesting period is large (20s), the simulation duration is reduced by a factor of 10 but the resulting model is less precise. With a 2mV difference observed after 363 train passages (i.e. the number of train passages per day), setting $T_{Harv} = 20s$ appears acceptable for the validation of the scenario.

3.6.4 *Simulation Results*

In this section, a network of 183 wireless sensor nodes (Node 0 - Node 182), each equipped with a wind-flow energy harvester is considered. Each node is separated from its neighbors by a random distance between $[60m, 190m]$ along the 24km subway tunnel which gives an average node distance of 130m. For figure clarity, Figure 3.23 shows the routed network for only the last 13 nodes closest to the terminal base station (Node 183) which is not considered in the scenario.

Unless otherwise stated, the following simulations are run with the default parameters given in Table 3.4. T_{SENSE} and T_{PMS} are the node's sensor activity and

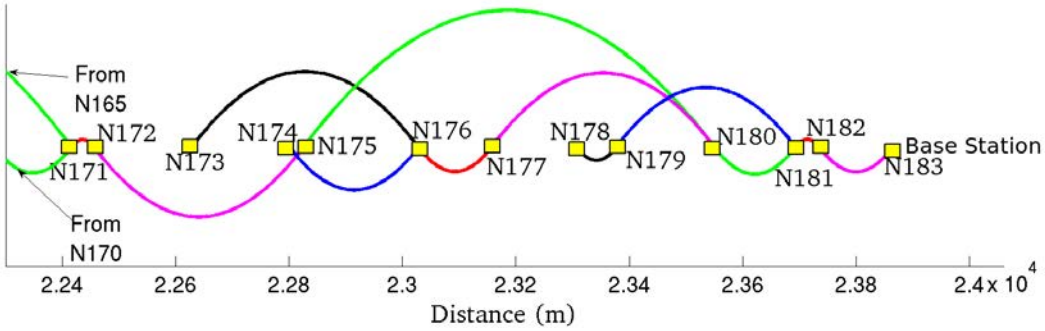


Figure 3.23: Node position and network connectivity (last 13 nodes).

power management periods, respectively. Note that in case no global (network-level) power manager is used, $T_{period} = T_0$ (i.e. network functions with a constant transmission period). In addition, if a local (node-level) power management algorithm is not used, $T_{SENSE} = T_{period}$ (i.e. nodes sense data once per transmission cycle). V_{init} is the value of each node's supercapacitor voltage at the start of the simulation. The receiver sensitivity $Sens_{RX}$ is defined as the input power level such that the Packet Error Rate (PER) is 1%. Default simulation duration is 24h. Simulation begins at 5am and finishes at 5am of the following day.

Table 3.4: Default values of simulation parameters.

Parameter	**Value**	**Parameter**	**Value**
T_0	20s	T_{period}	T_0
T_{SENSE}	T_{period}	T_{PMS}	T_{SENSE}
V_{init}	1.6V	$E_{harvested}$	60mJ
$Train_{ATT}$	100dB	$Sens_{RX}$	−95dBm

A nominal simulation of the scenario is considered with the parameters in Table 3.4 without power management (neither global nor local). The total energy consumed per node over a 24 hour period is shown in Figure 3.24. Obviously, nodes closest to the base station and situated on the main connectivity chain have greater energy consumption due to data aggregation. According to Figure 3.24, Node 181 has the greatest energy consumption. Figure 3.25 shows the residual supercapacitor voltage after a 24h simulation. Logically, the more a node consumes energy, the lower its residual voltage is. Again, we observe that the energy constraint of node 181 is the most important. The network can be considered viable if every supercapacitor voltage after 24 hours is greater than its initial voltage (1.6V). Indeed, this means that, over several days, this voltage will tend to increase until a steady-state is reached.

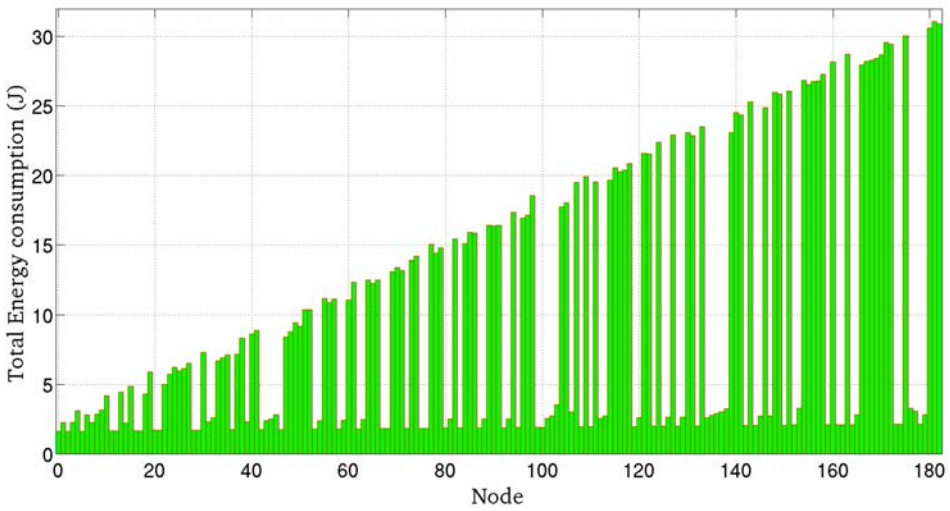


Figure 3.24: Total energy consumed per node over 24 hours

Figure 3.26 shows the energy consumed in each node state including transmission, reception, sleep, sensing and others (which includes radio turn around and standby, power management, and data processing). For nodes having less data to transmit (e.g. node 178), energy for transmission and reception is relatively less important. In contrast, these states occupy an important part of the energy consumption ratio (i.e. over 85%) for nodes 181 and 182. Note that the energy consumption of node 181 is a little bit higher than that of node 182 though node 182 has a greater number of data to transmit. This is due to the fact that node 181 is an inter-node router and hence dissipates more energy due to radio overheads (standby, turn around, PHY protocol, acknowledgments).

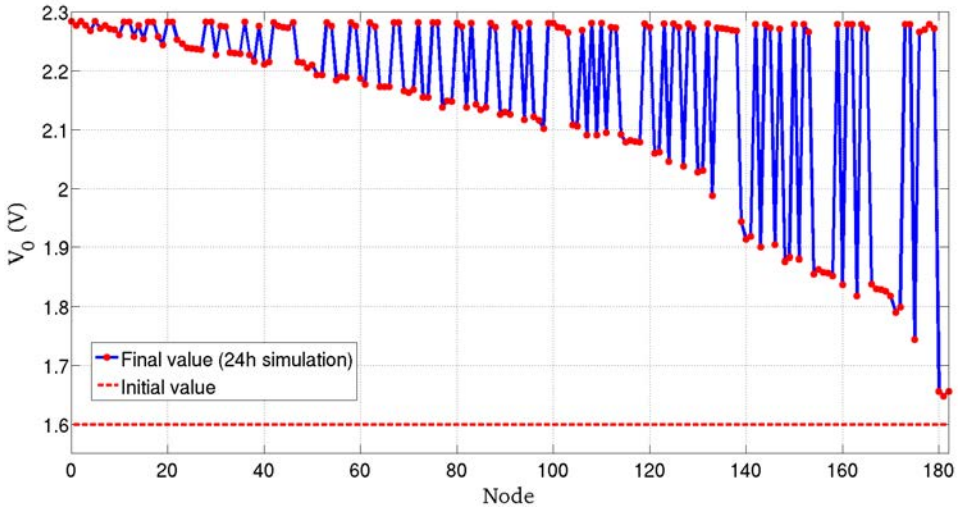


Figure 3.25: Supercapacitor voltage after 24h simulation.

Using the HarvWSNet framework, many other simulation were run in order to verify all of the other requirements of the scenario including data delay and network lifetime in case train traffic suddenly stops. To study the sensitivity of

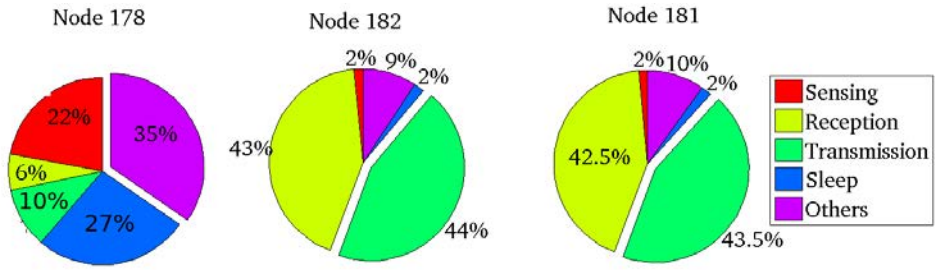


Figure 3.26: Energy consumption per node state.

the application to the amount of harvested energy, this last value was varied from 5mJ to 60mJ per train passage. This allowed to observe the value of the global power management algorithm. Indeed, if a global PM is used, Figure 3.27 shows the variations of T_{period} over a 24 hour period for different amounts of supplied energy. When the energy supplied is small, the network functions most of the time at maximum T_{period} to save energy. However, at 60mJ and 30mJ, the network can function with smaller T_{period} when the frequent train passages tend to recharge the supercapacitors.

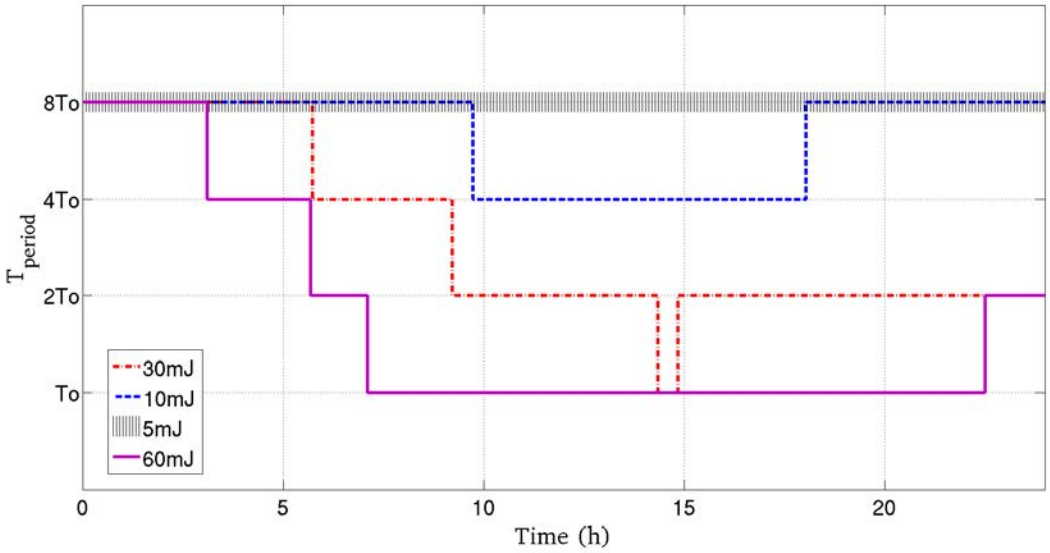


Figure 3.27: Transmission period of network with global PM.

3.7 CONCLUSION

While simulations are widely used to evaluate protocols and algorithms for WSNs, available network simulators offer only a simple linear battery model which can not be used to accurately estimate the lifetime of deployed energy harvesting nodes in the real world. In this chapter, we have proposed HarvWSNet a co-simulation framework for energy harvesting wireless sensor networks. HarvWSNet is a global tool that combines the strengths of WSNet and Matlab to eval-

uate energy-aware protocols and hardware for WSNs. WSNet is used for the multi-node network simulation while MATLAB is used for the simulation of the energy harvesting system. Both tools are run interactively and synchronized using TCP sockets. To demonstrate the efficiency of HarvWSNet, two case studies were defined. The first one simulates solar harvesting WSNs, where a solar energy harvesting model is implemented in MATLAB based on COTS devices. The second one was used to pre-prototype a wind-harvesting WSN for a subway tunnel structural health monitoring application. It is shown that the wind generated by the train passages can be harvested to supply the sensor nodes and support a client's monitoring application. The simulation results also show the value of using power-aware protocols to improve network robustness and performance. This work clearly shows that pre-prototyping can ease the development and minimize time-to-market of novel energy harvesting wireless sensor network applications.

4

POWER-RECONFIGURABLE TRANSCEIVER FOR WIRELESS SENSOR NETWORKS

Contents

4.1 INTRODUCTION

The previous chapter focused on increasing the autonomy of WSNs by capturing and transforming the environmental energy into electrical energy that sustains

the sensor node's operation. Indeed, energy harvesting is able to reduce the dependency on battery power but may be unable to carry the burden when ambient energy is scarce or unavailable for harvesting. Therefore, extending the lifetime of WSNs also implies minimizing the power consumption of the RF wireless transceiver, often the most power-hungry component of a wireless sensor node, as shown in Figure 4.1.

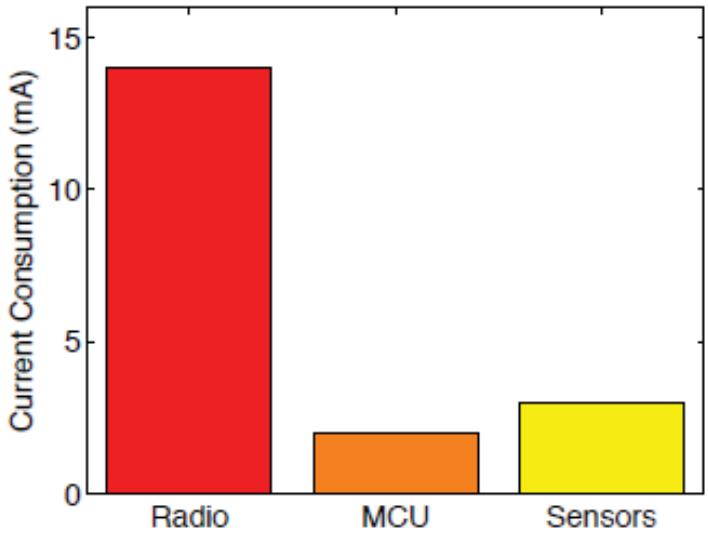


Figure 4.1: Active mode current consumption in a typical WSN node [18]

In the literature, several techniques for minimizing the power consumption in WSNs have been proposed. A great deal of research has addressed the power consumption savings that can be obtained using low power MAC and routing protocols in addition to savings that can be obtained by a judicious choice of physical layer (bit rate, modulation, etc.). At the transceiver level, low-power design approaches for analog [58] [110] [111] [112] and digital [113] [114] [115] circuits have been extensively studied. However, in all of this work concerning the physical layer implementation, the design methodology typically incorporates large specification margins to account for the worst-case channel conditions, simulation corners, and thermal effects. However, a radio transceiver may rarely face such worst-case environment and consequently, consumes more power than required under all operating conditions.

Recently, researchers have proposed to improve the network energy efficiency by dynamically adjusting the transmission power which typically has a high impact on the node's power consumption. An estimation of the link quality is used in order to maintain a "good-enough" link between nodes [116]. However, on the other side of the link, receiver reconfiguration has received much less attention in the literature. Most modern receivers are designed to achieve a minimum signal-

to-noise ratio (SNR) when the channel conditions are in worst-case (e.g. noise, interference, multipath). However, when the channel conditions are favorable, the transceiver may consume more energy than required to meet the target minimum bit-error-rate (BER). Since worst-case conditions may be rarely or only periodically experienced in the actual network deployment, a reconfigurable receiver, able to adjust its performance and power consumption to the instantaneous propagation conditions of the signal and interference, could provide considerable power savings.

The primary research goal of this chapter is to analyze the global problematic facing reconfigurable transceivers. Indeed, it is not enough to simply design reconfigurable circuits. It is of primary importance to identify which new system components are required to build a reconfigurable transceiver and understand which are the main design issues faced by RF system and circuit designers.

4.2 SENSE & REACT APPROACH AND CHALLENGES

Traditionally, RF transceiver building blocks are designed to function at a fixed performance level defined during the system design process for a target standard, and cannot be modified except through physical intervention. In practice, however, the propagation environment varies with time which means that the received signal strength varies due to varying propagation losses, multipath effects, and interference levels. Conservative RF transceivers operating in these conditions generally consume much more energy than needed during the majority of time. It is therefore desirable to dispose of a flexible radio which can scale its performance with respect to the channel conditions and application requirements. By continuously adapting the transceiver performance to the channel circumstances, such reconfigurable radios could result in significant energy savings.

A reconfigurable radio has four key features that distinguish it from a traditional radio: channel sensing, channel analyzing, performance mode deciding and performance adjusting. Figure 4.2 illustrates how these features allow a reconfigurable radio to conceptually react to the radio environment. This illustration is referred to as the *sense & react* cycle that is continuously processed by a reconfigurable radio to observe and analyze the channel condition, select the best performance mode, and adapt its performance.

SENSE Sensing refers to the ability of an RF transceiver to monitor the quality of a radio channel and estimate its related parameters. For example, a hardware sensing block might provide measurements of noise floor and interference levels

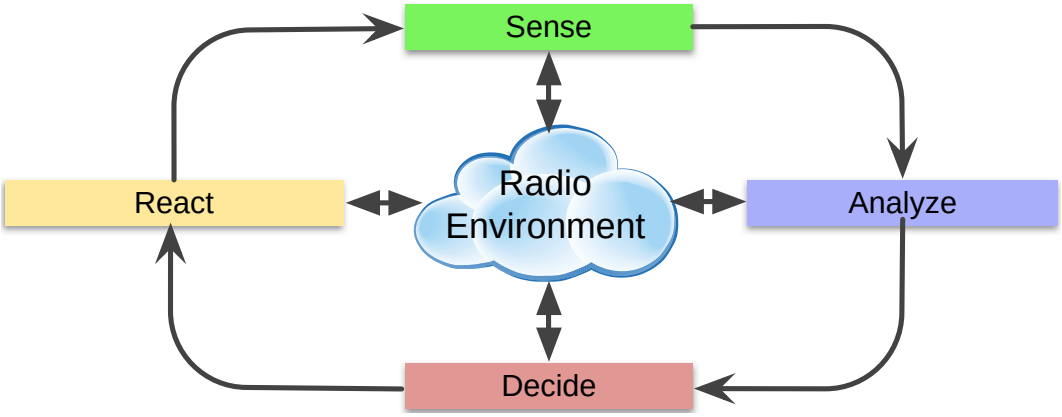


Figure 4.2: Functional architecture of a typical sense & react RF transceiver.

in the radio channel. The data output by the sensing block must be sufficiently accurate to allow exact conclusions to be reached regarding the channel usage in the radio environment. Moreover, channel sensing must be fast and frequent in order to track the variations of the radio environment. Such requirements may place stringent requirements on a hardware implementation in terms of sensing bandwidth, processing power, RF circuitry, etc.

ANALYZE The RF transceiver must also be able to analyze the output of the sensing block. For example, an analysis block placed directly within the RF transceiver might be able to distinguish between the desired signal, the interference level and the noise floor based on the sensed radio spectrum. Alternatively, an analysis block may process information extracted from high level protocols (MAC or NTW). In the analysis block, the data retrieved from the sensing block is used to build a metric that indicates the link quality. For example, this could be the average signal strength of a packet, the noise in the desired channel or the interference in adjacent channels. As described in Section 4.6.1, there is a huge variety of possible link quality metrics that can be defined.

DECIDE The third step of the *sense & react* cycle is to select the performance mode according to the calculated link quality metric.

REACT Based on the the performance mode chosen, the react control block defines the parameters of each reconfigurable component of the transceiver for an upcoming transmission or reception of a frame or a packet.

While *sense & react* represents an interesting approach to drastically reduce the power consumption of wireless systems without affecting the QoS required by the application, building a fully-functional reconfigurable radio is a difficult challenge. The RF transceiver is required to achieve a sufficient level of reconfigura-

bility in order to provide multiple performance modes. This is an additional design constraint for RF, analog and digital circuit designers and may place severe requirements on the noise, linearity, dynamic range, and power consumption of the transceiver building blocks. In fact, instead of designing a single receiver, the design team must now actually design a set receivers, each with its unique requirements. Care must be taken to minimize prototype design and test time.

Furthermore, the units responsible for channel sensing, analyzing and setting the performance mode are required to have a high processing speed and good robustness in order to make a decision with a relatively low delay and detect transmission problems before the communication is too degraded. In addition, the additional power consumption associated with these blocks must be relatively low to avoid impacting the global system performance.

An additional challenge consists in identifying the blocks which may benefit from the reconfiguration function. For example, consider Figure 4.3 which shows a break down of the power consumption of the main blocks of a transceiver. Notice how the synthesizer occupies an important part of the power budget along with the RF functions (LNA, mixer and PA). The challenge here is identifying the appropriate blocks that will offer considerable power savings while minimizing performance impact.

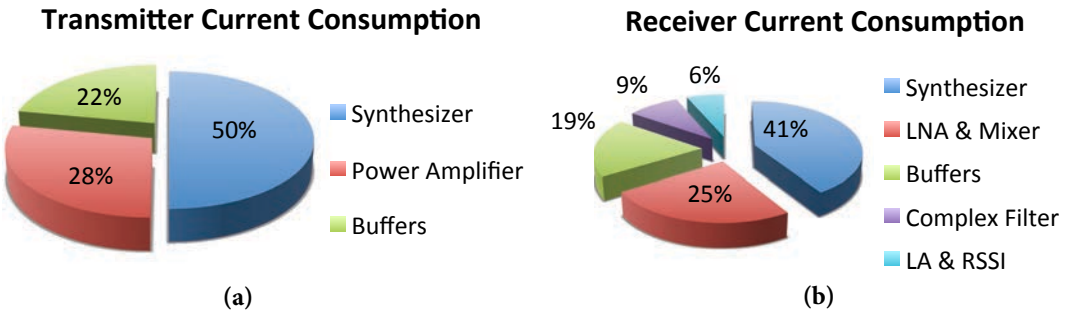


Figure 4.3: Distribution of the current consumption in a transceiver: (a) receiver, (b) transmitter [19].

4.3 PRIOR WORK ON RECONFIGURABLE RF SYSTEMS

As traditional RF transceivers typically function with fixed parameters and are designed to be application-specific, researchers around the world have deeply explored the feasibility of reconfigurable wireless systems in which the transmission and reception parameters (e.g. operating frequency band, modulation, waveform, performance, etc.) can be controlled dynamically. This flexibility can theoretically be achieved by a software defined radio (SDR) in which some or all of the physi-

cal layer functions are controlled by a software signal processing algorithms [117]. The architecture of an SDR is shown in Figure 4.4.

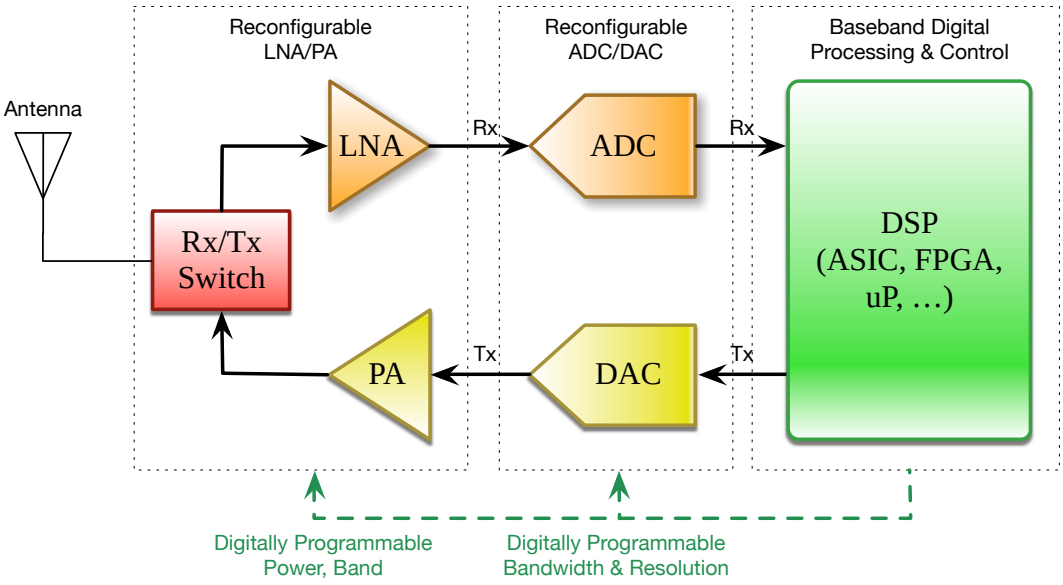


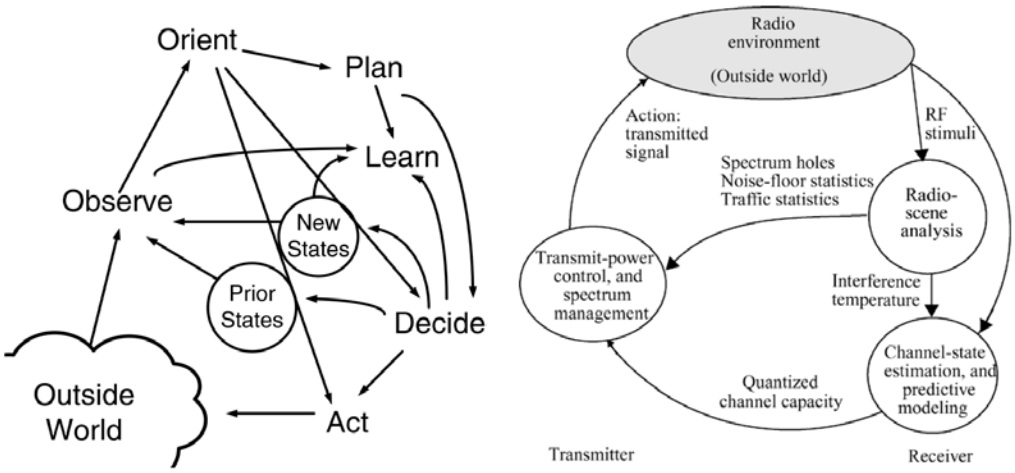
Figure 4.4: Architecture of an SDR.

In an SDR, the only analog components are an LNA and an ADC in the receiver, and a PA and a DAC in the transmitter. All other radio functions are realized using digital signal processor (DSP). One widely used SDR implementation is GNU Radio [118] which is a software package that provides many signal processing blocks and discrete components to build a software radio.

Given the recent advances in CMOS technology and data converters, SDR appears like a good approach for all types of radios, from cellular to WSNs. However, since minimal filtering is applied in the RF and analog parts, such a radio requires an extremely high-resolution ADC featuring a very high dynamic range in order to digitize any received signal frequency at the antenna. For example, to receive signals up to 3 GHz, the ADC required should run at 6 GHz (Nyquist rate) with a dynamic range of more than 100 dB. This requires more than 16 bits of resolution in the ADC to meet the required SNR [119]. Unfortunately, given the state-of-the-art of existing ADCs [120], the power required by such an ADC is far from compatible with a battery-powered sensor node even equipped with energy harvesting capabilities [121].

Another more flexible radio concept called *Cognitive Radio* has been deeply studied in the literature during the last decade. The concept of cognitive radio was first introduced by *Mitola* [122] and extended later by *Haykin* [123]. Cognition stands for the capability of a system to sense, analyze and adapt its parameters to the

surrounding environment.



(a) Mitola defined the cognitive radio as an extension of the SDR [122].

(b) Haykin has extended the cognitive radio concept and defined the physical layer communications and signal processing aspects associated with it [123].

Figure 4.5: A comparaison between the cognitive radio concepts of Mitola and Haykin.

The original proposals of *Mitola* (Figure 4.5-a) and *Haykin* (Figure 4.5-b) and existing research in the area of cognitive radio focused mainly on dynamic spectrum access. Recently, a certain level of flexibility has been implemented in some standards such as 802.11n which supports different constellation sizes and code rates, and also operates on different frequencies and bandwidths [124]. However, very little work has emerged in the field of power-reconfigurable circuit design. The following sections present a state of the art of RF blocks and low-power strategies aiming to implement some power/performance flexibility in RF transceivers.

4.3.1 *Prior Work on Reconfigurable Transmitters*

In the transmission of data from a wireless transmitter to a receiver, the transmission power and data rate determine the range over which the signal can be intercepted by the receiver and is therefore important in determining the performance of a network. Existing low power radio technologies dedicated to WSNs such as IEEE802.15.4 allow a transmission range in the order of 10 meters to a few hundred meters in line of sight at data rates from a few kbits/sec to 1 Mbit/sec. This order of distances is sufficient for several sensor network deployment scenarios. In order to transmit data over larger distances, WSNs typically use multi-hop relaying techniques.

Today's COTS radio transmitters include some flexibility in the transmission parameters such as radiated power and transmission rate. For example, the TI CC2420 [125] provides eight programmable transmission power steps ranging from -24 dBm to 0 dBm. The transmission power and the corresponding current consumption are represented in Table 4.1.

Table 4.1: TI CC2420 transmission power and corresponding current & power consumption.

PA Level	Output Power [dBm]	Current Consumption [mA]	Power Consumption [mW]
3	-25	8.5	15.3
7	-15	9.9	17.8
11	-10	11.2	20.1
15	-7	12.5	22.5
19	-5	13.9	25
23	-3	15.2	27.3
27	-1	16.5	29.7
31	0	17.4	31.3

This flexibility in the transmission power has been intensively exploited in order to reduce the excessive power consumption of the transmitter and optimize the traffic carrying capacity of the network by minimizing the interference generated with respect to the nodes located in the transmission range. A good deal of research activity has focused on the establishment of control policies that allow the sensor nodes to adapt their transmission power level with respect to the other nodes in their neighborhood [126] [127] [128]. The approach used, known as transmission power control (TPC), allows a transmitter to adapt the radiated power to a minimum level, which essentially depends on the proximity to the receiver, while ensuring a good link reliability.

For example, if the MAC protocol allows an exchange of control signals (e.g. RTS, CTS, beacons, preamble, etc.) between the receiver and the transmitter, the receiver can send the link quality estimation (LQE) together with these signals so that the transmitter can adapt its transmission power for future data transmissions. Alternatively, acknowledgment ACK packets can also be used to vehicle this information. Hence, with commercially available circuits, a reconfigurable system can be defined which minimizes power consumption while adapting to the varying channel conditions.

4.3.2 *Prior Work on Reconfigurable Receivers*

Receiver reconfiguration has received much less attention in the literature. State-of-the-art research in reconfigurable receivers has, to the best of our knowledge, only focused on the design of reconfigurable circuits.

Authors in [129] propose a power scalable digital baseband for a low-power IF receiver that reduces its power consumption by varying the word length and sampling frequency based on the values of estimated interference and signal-to-noise ratio. Thanks to an exhaustive search by Matlab for different levels of interference and SNR, the optimal values of bit-width and sampling frequency are stored in a look-up table. The power consumption figures in the different modes are obtained using Synopsys Prime Power. The authors show that the power consumption of the digital base band can vary from 0.49mW to 3.3mW.

In [20], an energy-aware RX front end for low-power 2.4 GHz applications is presented. By pushing the requirements of the receiver to the low noise amplifier (LNA), the authors obtain an important amount of controllable power consumption with a single reconfigurable block. The LNA was designed to achieve high gain and low power consumption while allowing variable performance modes as shown in Table. 1. Although the LNA of wireless receiver is well-characterized in these 4 modes, the article does not clearly show how the wireless receiver will pilot this block under varying signal to noise ratio conditions.

Table 4.2: LNA characteristics from [20] (CMOS 0.18µm)

Function mode	1	2	3	4
Current (mA)	5	3	1.9	
SSB Noise figure (dB)	8.8	9.3	12.7	16.5
IIP3 (dBm)	-31	-27	-23	-19
Voltage gain (dB)	35.6	34.7	28.7	24.5

In [130] the authors propose a reconfigurable receiver in which the gain and linearity of an LNA are independently controlled based on the calculated error vector magnitude (EVM). The authors claim that independently (orthogonally) tuning the gain and linearity allows an extra power savings of 22% as compared to a traditional non orthogonal adaptive system.

With the recent development of wake-up receivers, many researchers have also focused on the design of power scalable on/off keying (OOK) receivers for low

power event-driven applications. In [131], the authors propose an OOK receiver scalable in data rates from 1Kbits/s at 64µW to 100Kbits/s at 146µW.

In [132] the authors propose a reconfigurable low noise amplifier (LNA) with three different levels of performance. The reconfiguration is achieved using an external DAC that controls the supply voltage. Other power-scalable blocks such as an analog channel filter and variable gain amplifier (VGA) [133] and analog-to-digital converters (ADCs) [134] have been proposed with the aim of one day enabling true receiver reconfiguration.

However, in all of this aforementioned work, little attention is paid to the control functions employed to pilot the reconfigurable blocks, nor to the impact of receiver reconfiguration on the global network performance. Indeed, while this innovative research is indeed invaluable, evaluating the power savings that could be obtained using a reconfigurable receiver is difficult since the time-varying signal and interference propagation conditions so highly depend on the WSN deployment scenario.

4.4 RECONFIGURABLE TRANSCEIVER ARCHITECTURE

As discussed above, prior work has mainly focused on implementing circuits with some capacity for reconfiguration. However, very little work has addressed the challenge of defining a complete reconfigurable transceiver system. In fact, excluding [129] and [130], most prior work has focused uniquely on the *React* part of the *Sense & React* approach by defining tuning knobs for their circuits.

Our contribution in this section involves the specification at system-level of a transceiver architecture that implements all of the states of the *Sense & React* approach described in Section 4.2. The objective is to identify the major challenges facing transceiver reconfiguration.

The conceptual diagram of the reconfigurable transceiver is represented in Figure 4.6. It is subdivided into an analog front-end section and a digital base-band section both including several tuning knobs (red arrows). With respect to the *Sense & React* functional architecture described in Figure 4.2, the *Sense* function can be performed in the analog or digital sections of the transceiver or in the higher-level protocols, or in any combination thereof. This information is provided (green arrows) to the *Reconfiguration Manager* which performs the *Analyze*, *Decide* and *React* functions. In this conceptual model, no limits are placed on the implementation

of the *Reconfiguration Manager* which may be a pure analog function, dedicated digital hardware or a software component running on a micro-controller.

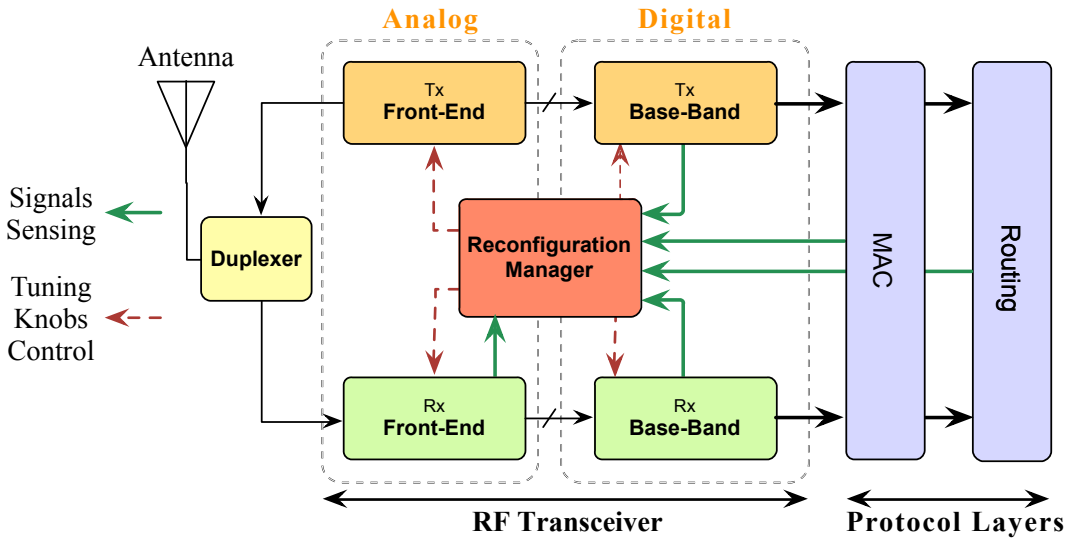


Figure 4.6: Conceptual diagram of the reconfigurable transceiver.

The conceptual diagram of the reconfiguration manager is shown in Figure 4.7. It is composed of the following blocks:

- A metric calculation block builds the Link Quality Estimation (LQE) indicator that reflects the link quality by capturing the signals and the information provided by all of the components that implement some type of sense function. The possible definitions of the LQE are discussed in detail in Section 4.6.1.
- The decision algorithm is the intelligent part of the reconfiguration manager which takes as input the LQE metric and establishes a decision on a new performance configuration for the transceiver.
- The reconfiguration decision is applied by the tuning controller that adjusts the transceiver performance using the built-in tuning knobs.

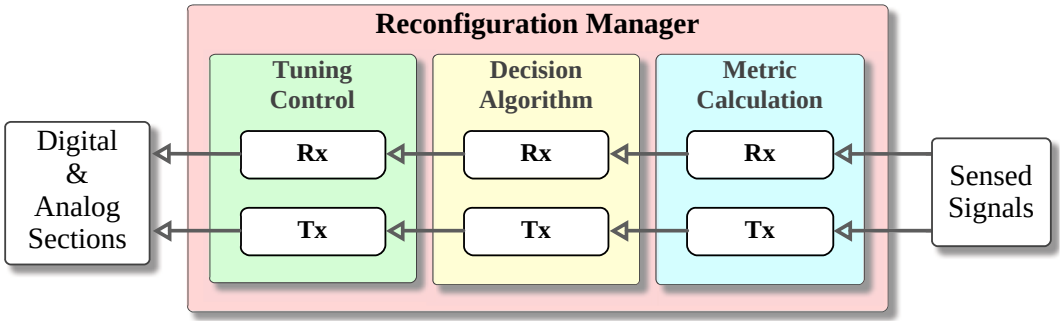


Figure 4.7: Conceptual diagram of the reconfiguration manager.

In the next sections, we present the architecture of the reconfigurable transmitter and receiver, and the corresponding system model.

4.4.1 *Reconfigurable Transmitter System Model*

To design a successful reconfigurable transmitter, it is necessary to understand its real behavior and investigate the sources of power dissipation. As shown in Figure 4.3-(a) the power consumption of the direct up-conversion transmitter is shared between two parts: the first one is due to the components responsible for signal generation and processing (i.e. PLL, DAC, mixers, filters, etc.), and the second part is due to signal transmission by the power amplifier.

As discussed in Section 4.3.1, a transmission power control (TPC) protocol can be used to reduce the excessive power consumption of the transmitter. This technique is commonly used in cellular and WLAN systems but has also been investigated for low power systems such as WSNs. Transmission power control can be realized simply by controlling the transmission power of the PA which dominates the power consumption of a transmitter. However, not all TPC systems offer power-aware reconfiguration. Indeed, a PA is not always designed to offer constant efficiency over its entire range of output powers. If the PA efficiency degrades with lower output power (which is often the case), power savings offered by TCP are minimized.

Conceptually, in a power-aware reconfigurable transmitter, other blocks than the PA can offer power/performance trade-offs. Of course, differently from the PA which is common to all architectures, the possibility of defining other power-aware reconfigurable blocks will depend on the transmitter architecture which can take various forms: direct, single or multiple IF up-conversion, polar modulation, direct VCO modulation, etc. Alternatively, a pulsed ultra-wide band (UWB) transmitter can offer a power/performance trade-off by inherently lowering its power consumption with lower bit rates if the RF blocks are switched off between each emitted pulse [135].

For example, in a Zero-IF up-conversion type architecture for narrow-band communications, other blocks offering power-aware reconfiguration could include the VCO which typically presents degraded phase noise for lower bias currents. Additionally, power-aware reconfiguration could be offered by varying the output symbol rate (and therefore bit rate) of the digital base-band block. By modifying the bandwidth of the signal and associated sample clocks, power savings could be obtained in the digital base-band as well as in the DAC. Exploring the wide variety

of transmitter architectures and identifying all potentially power-reconfigurable blocks is beyond the scope of the present work.

However, considering that the PA offers the greatest potential for power-aware reconfiguration, in this work, we focus on the establishment of system-level models and reconfiguration strategies that allow the reduction of the power consumption due to this block. The conceptual diagram of a direct-conversion reconfigurable transmitter is presented in Figure 4.8 and consists of the following stages: a digital base-band, an analog base-band and an analog front-end.

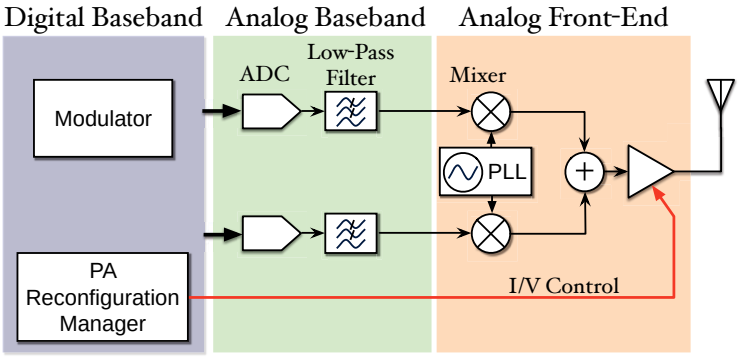


Figure 4.8: Conceptual diagram of the reconfigurable transmitter.

In this transmitter, the PA output power control is realized using the reconfiguration manager that adjusts the performance of the PA as function of the calculated LQE (e.g. based on the RSSI measured by the receiver.) Figure 4.9 shows a possible implementation (among many others) of a reconfigurable PA. It consists of a one stage PA and a tuning block that varies the efficiency and transmitted power by controlling the gate and drain bias [136].

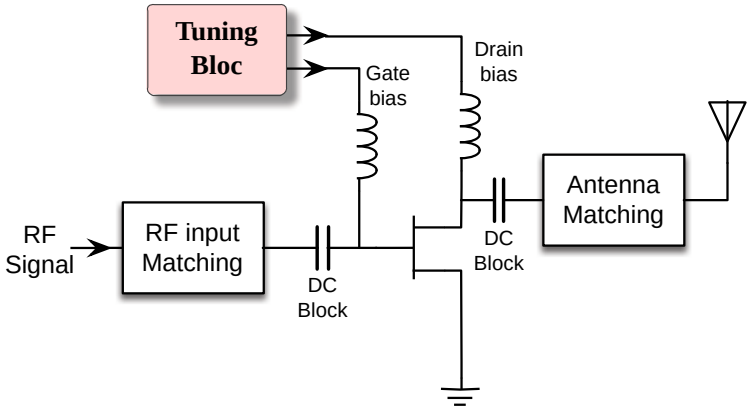


Figure 4.9: Schematic of the reconfigurable PA.

4.4.2 *System Model of the Reconfigurable Receiver*

The reconfiguration of the receiver appears more stringent than transmitter reconfiguration due to the non-uniform distribution of the receiver specifications (gain, noise, linearity) among the various building blocks. Indeed, as shown in the example of Figure 4.3-(b), the most power hungry components include the LNA, mixer, synthesizer, and ADC. Receiver reconfiguration becomes interesting only if the whole power consumption of the receiver chain is divided by a factor of 2 or more. Therefore, the reconfiguration might cover all of these blocks in order to drastically reduce the power consumption during reception. Furthermore, the implementation of tunability in these components typically comes at the risk of communication performance as well as power consumption penalty. Therefore, the major challenge of this work is to investigate the feasibility of such reconfiguration and evaluate the impact of receiver building block tuning on the power consumption as well as the communication performance of a WSN.

When designing a low power RF receiver, there are a number of architectural design choices that must be made (e.g. Zero-IF or Low-IF architecture, linear or non-linear amplification, analog or digital channel selection, etc.). The receiver chain blocks where reconfiguration can be usefully applied will therefore vary with the implementation and affect the reconfiguration strategy. We limit our study to a Zero-IF architecture with digital channel selection. The model could, of course, be adapted to any other receiver architecture.

As stated previously, a reconfigurable receiver necessarily includes a number of reconfigurable blocks, a means for calculating a reconfiguration metric that estimate the channel conditions (LQE), and a decision algorithm for applying the reconfiguration strategy. The proposed reconfigurable receiver is presented in Figure 4.10 and consists of an analog front-end which filters and amplifies the signal received at the antenna, an analog base-band that down-converts, further amplifies, pre-filters and digitizes the signal by the ADC, and a digital base-band responsible for signal processing and demodulation.

In this example, the reconfigurable blocks, identified by a red arrow, are the LNA, mixer, voltage controlled oscillator (VCO) and analog to digital converter (ADC). These blocks possess tuning knobs that allow the reconfiguration manager to set the target performance mode, tailored to the channel conditions. The reconfiguration manager is located in the digital base-band. Alternatively, the reconfiguration manager could be implemented as an algorithm running on an embarked micro-controller. Based on this example, we will investigate below what are the

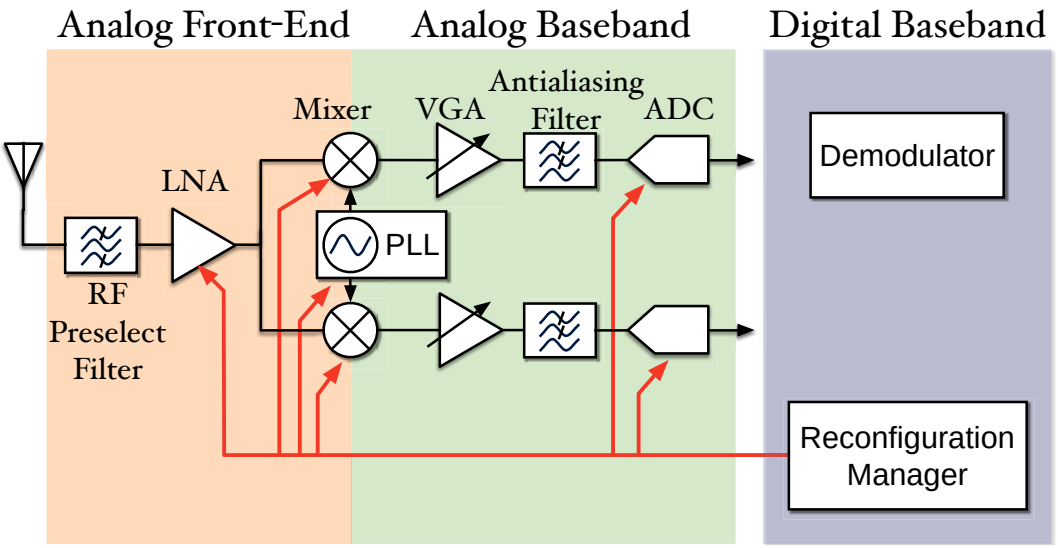


Figure 4.10: Architecture of the reconfigurable receiver.

minimum transceiver parameters required to achieve the a minimum power consumption for several power reconfigurable modes.

4.5 FIGURE-OF-MERIT BASED RECONFIGURABLE TRANSCEIVER

To describe an RF transceiver and evaluate its performance at the system-level, a set of metrics that have become industry standards over the past years are used. These metrics can generally be divided into two groups. The first group consists of metrics such as sensitivity, linearity, noise figure, and adjacent channel selectivity which can be extracted from the target standard specifications and hence quantify the global performance of the RF transceiver. Indeed, once these specifications are set, the RF system architect can further specify the parameters and margins of the transceiver's building blocks. The second group consists of metrics such as power consumption or bandwidth of a component which are not specified by the standard but are mainly used to compare the performance of different blocks and select the appropriate design for the target application.

The design of power reconfigurable RF circuits is a brand new field of research. Hence, currently, there is no common specification methodology for defining the performance of scalable analog blocks. The modeling approach adopted in this chapter for the transceiver's reconfigurable blocks is based on the figure-of-merits (FoMs) of measured blocks reported in the literature. FoMs are widely used in the literature to compare the performance and evaluate the efficiency of analog circuits. They represent an easy way to compare the characteristics of each design and predict the evolution of its performance in the future [137]. A FoM combines

several characteristics of a component such as power consumption, linearity, noise figure, gain, operating frequency, etc. The value obtained from the combination of these parameters represents a performance measurement independent of the circuit topology or its application.

In the following, we therefore will assume that the FoM correctly captures the design space of a given block. For each block, we therefore choose an average FoM for a given technology node and deduce the block characteristics for different levels of performance. Our adaptive models can later be refined when more circuits, specifically designed for power adaptability, become available. As a first step, we define three power consumption/performance modes for each block.

4.5.1 *Figure-of-Merit Based Reconfigurable Transmitter*

As stated above, the reconfigurable block considered in the transmitter is the PA. To establish the performance modes of a PA, the FoM should take into account the main key parameters that characterize a PA like power gain G, output power P_{out}, carrier frequency f, and power-added-efficiency (PAE). The following FoM_{PA} is used for benchmarking PA designs [137]

$$FoM_{PA} = G \times P_{out} \times PAE \times f^2 \ [W \times GHz^2] \tag{4.1}$$

The PAE represents the efficiency of a PA and is usually expressed as

$$PAE = \frac{P_{OUT} - P_{IN}}{P_{DC}} \tag{4.2}$$

where P_{IN} and P_{OUT} are respectively the input and output power, and P_{DC} is the DC component of the power drawn by the PA from the power supply. The linearity of a PA depends mainly on the operating class of the amplifier, making it difficult to compare amplifiers of different classes. To remain independent of the design class, the linearity was not included in the FoM_{PA}.

The average FoM_{PA} chosen for the 65nm technology node is $66GHz \times W^2$[19] [138]. Based on this figure of merit, three performance modes are defined for the PA and are summarized in Table 4.3.

4.5.2 *Figure-of-Merit Based Reconfigurable Receiver*

In this section, we detail the performance of each reconfigurable building block composing the receiver shown in Figure 4.10. The reconfigurable components are: LNA, mixer, VCO and ADC.

Table 4.3: Power amplifier performances.

Performance Mode	P_{dc} [mW]	P_{out} [dBm]
High	9	5
Moderate	6	0
Low	3.3	-5

4.5.2.1 *Reconfigurable LNA and Mixer model*

In order to fix the performance of the LNA, we use the FoM_{LNA} defined by the ITRS [139] as

$$FoM_{LNA}[GHz] = \frac{G[lin.] \cdot IIP_3[mW] \cdot f[GHz]}{P_{DC}[mW] \cdot (F[lin.] - 1)}, \tag{4.3}$$

where G is the power gain, F is the noise floor, IIP_3 is the third-order intercept point, f the operating frequency, and P_{DC} is the power consumption. Based on [140], an average FoM_{LNA} of 15 GHz is chosen.

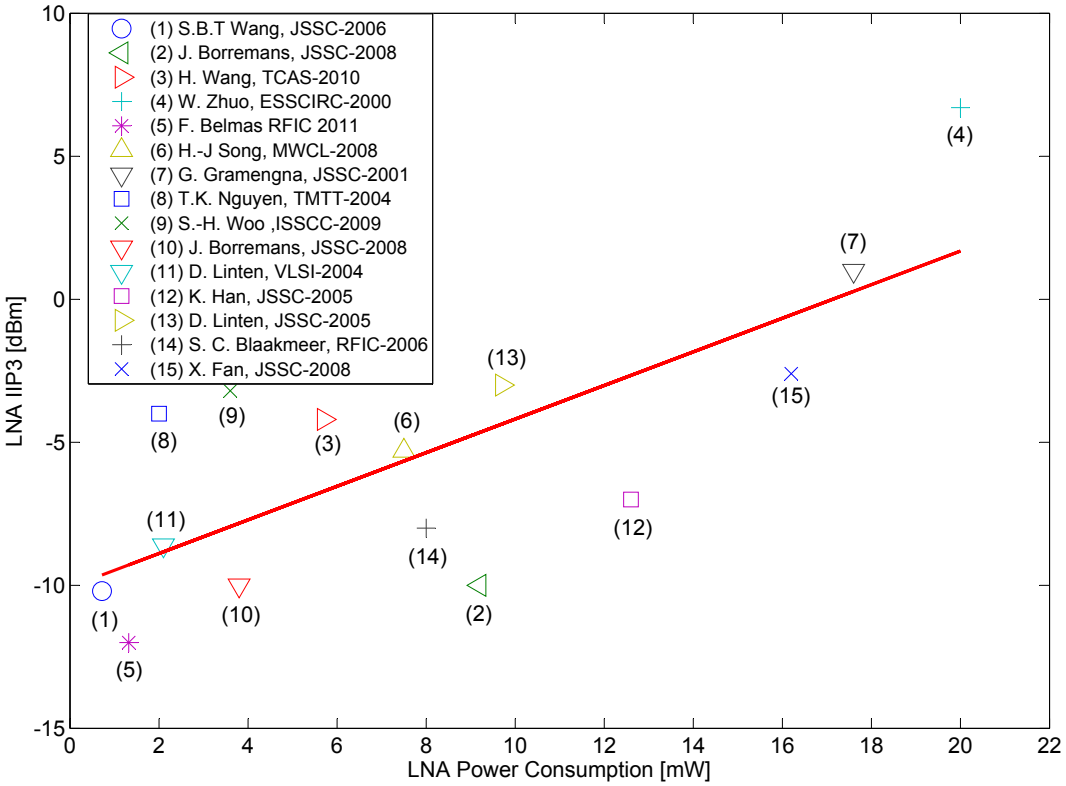


Figure 4.11: IIP_3 versus power consumption in recently published LNAs.

Figure 4.11 shows the relation between IIP_3 and power consumption in recently published LNAs (Low-Noise Amplifiers). The IIP_3 shows a tendency to decrease as power consumption decreases. Hence, in the reconfigurable receiver, the power consumption of the LNA can be reduced if interference levels are low in the node environment.

In the literature, there are various figure-of-merits that have been developed in order to quantify the performance of the mixer. The FoM_{Mixer} presented in [141] takes into account conversion gain, linearity, power consumption and noise factor. The FoM_{Mixer} used is given as

$$FoM_{Mixer}[dB] = 10\log\left(\frac{G[lin.]\cdot IIP_3[mW]}{P_{DC}[mW]\cdot(F[lin.]-1)}\right) \tag{4.4}$$

Based on [141], an average FoM_{Mixer} of -17 dB is chosen. Using (4.3) and (4.4), the LNA and mixer characteristics are given in Table 4.4 for three performance modes: high, moderate and low.

Table 4.4: LNA and Mixer Performances.

Component	Mode	Pdc [mW]	G [dB]	NF [dB]	IIP_3 [dBm]
LNA	High	5.9	16	1.5	-4
	Moderate	1	14	2.8	-6.5
	Low	0.1	12.5	6	-9.8
Mixer	High	3.5	10	14.5	-7
	Moderate	1.2	9	15	-11
	Low	0.2	7	16	-15.5

4.5.2.2 *Reconfigurable VCO Model*

The power consumption of a PLL can be decomposed among the charge pump, divider, phase detector, and the VCO. The contribution of the VCO power consumption to the total power consumption of the PLL can be up to 78% in some modern frequency synthesizer designs [142]. The power consumption of a VCO is usually inversely proportional to its phase noise which, in turn, is responsible for in-channel interference due to reciprocal mixing. Thus, reducing the VCO power consumption in the case of low or inexistent adjacent interferers will result in a

reduction of the overall power consumption of the PLL. Of course, the possibility of reconfiguring the VCO assumes that the PLL lock time and stability are guaranteed in each mode.

Similarly to the mixer, LNA, and PA, it is possible to combine the parameters of a VCO into a FoM that allows a realistic comparison of various VCO architectures. The ITRS defines several FoMs for VCOs in [139]. The most common includes the phase noise, DC power consumption, and carrier frequency. The FoM_{VCO} is given as

$$FoM_{VCO}[dBc/Hz] = 10\log\left[\left(\frac{f_c}{\Delta f}\right)^2 \cdot \frac{1}{L(\Delta f) \cdot P_{DC}[mW]}\right], \tag{4.5}$$

where f_c is the frequency carrier, Δf is the offset frequency, and $L(\Delta f)$ is the phase noise relative to the carrier measured over a 1Hz bandwidth. From [143], an average FoM_{VCO} of 178 dBc/Hz is chosen. The performance modes of the VCO are summarized in Table 4.5.

Table 4.5: VCO performances.

Mode	Pdc [mW]	$L(\Delta f)$ @ 1 MHz [dBc/Hz]
High	5.8	-114
Moderate	3.2	-110
Low	0.1	-95

A -20 dB/decade phase noise slope is assumed until a floor value 10dB inferior to the 1 MHz offset value is reached. The power consumption of other PLL components including the charge pump, filter, divider and the buffer is assumed to be 1.2 mW [142].

4.5.2.3 *Reconfigurable ADC Model*

In our model, we assume that the ADC is preceded by an anti-aliasing filter that limits the bandwidth to 10 MHz and that the ADC has a fixed sampling rate $f_{Sampling}$ of 20 MHz. An average FoM_{ADC} of 167 dB is extracted from data given in [120] assuming a FoM_{ADC} defined as

$$FoM_{ADC}[dB] = (6.02 \cdot ENOB + 1.76) + 10\log\left(\frac{f_{Sampling}}{P_{DC}}\right) \tag{4.6}$$

where ENOB is the effective number of bits. Since the signal is oversampled by an approximate factor of 5, the equivalent noise figure of the ADC is given as [144]

$$NF_{ADC} = 10\log\left(1 + \frac{V_{p-p}^2}{6kTR_sL_q^2f_s}\right) \tag{4.7}$$

where V_{p-p} is the full-scale voltage, R_s is the reference source resistance, L_q is the number of quantization levels, k is the Boltzmann constant and T the temperature. For V_{p-p} of 1V, the ADC performance modes are summarized in Table 4.6. Note that in this first level model, the bit stream entering the digital base-band varies from 3 to 9 bits depending on the performance mode. The additional power savings due to the corresponding dynamic range reconfiguration of the digital base-band are ignored.

Table 4.6: ADC performances.

Mode	Pdc [nW]	ENOB [bit]	NF_{ADC} [dB]
High	1.5×10^3	9	28
Moderate	200	6	46
Low	3.5	3	64

4.5.2.4 *VGA and anti-aliasing filter models*

The remaining components of the analog base-band, preceding the ADC, are the variable gain amplifier (VGA) and the anti-aliasing filter. The VGA is used to automatically adjust the signal to the full scale of the ADC, while the anti-aliasing filter is used to remove high frequency spectral components above the Nyquist frequency prior to the sampling process. Assuming that the receiver complies with the IEEE 802.15.4 standard, the filter bandwidth is fixed at 2 MHz. The required VGA power range is set by the difference between the minimum and the maximum signal input to the radio. For the 802.15.4 standard, the power range is set 0 dB and 33 dB. The VGA IIP3 and NF are 6 dBm and 25 dB, respectively. Similarly, the filter IIP3 is fixed at 30 dBm and the NF at 20 dB. The total power consumption of the VGA and baseband filter is assumed to be 2 mW (I and Q paths combined). In our model, we assume that the variable gain amplifier (VGA) is controlled by an ideal automatic gain control (AGC) algorithm that automatically sets the total receiver gain such that the input signal reaches the ADC full-scale voltage. This first-level

model neglects potential power consumption savings that may be obtained when the VGA changes gain setting.

4.5.2.5 *Reconfigurable Receiver Model*

Assuming that the gain and noise figure values of each block are given for the correct source and load impedances, the overall NF and IIP3 of the receiver is

$$\begin{cases} NF_{total} = NF_1 + \dfrac{NF_2 - 1}{G_1} + \cdots + \dfrac{NF_k - 1}{G_1 G_2 \cdots G_{k-1}}, \\ \dfrac{1}{IIP3_{total}} = \dfrac{1}{IIP3_1} + \dfrac{G_1}{IIP3_2} + \cdots + \dfrac{G_1 G_2 \cdots G_{n-1}}{IIP3_n}, \end{cases} \tag{4.8}$$

where NF_i, $IIP3_i$ and G_i are respectively the noise figure, third-order intercept point and power gain for the i^{th} block in the cascade chain. The global performance of the receiver based the models defined above is summarized in Table 4.7.

Table 4.7: Global receiver performances.

Mode	Pdc [mW]	NF [dB]	IIP3 [dBm]
High	22.6	4.6	-26
Moderate	10.5	7	-27
Low	4.5	14.7	-29

4.6 LINK QUALITY ESTIMATION

In the previous sections, we have postulated that a reconfigurable transceiver which adapts its performance can save a great amount of the energy dissipated when the channel conditions are favorable. This adaptivity is possible only if the radio has a reasonably good knowledge of the link conditions and communication performance. Hence, the unit responsible for link quality measurement represents a vital building block in the process of reconfiguration of the adaptive transceiver and selection of the optimal performance mode. In this section, we first review existing link quality estimation (LQE) techniques used for channel characterisation and their usage domains.

4.6.1 *Link Quality Estimation Techniques*

The communication performance in WSNs is affected by several factors that degrade throughput and may lead to high power consumption due to long periods of idle listening and/or numerous packet retransmissions. The unreliability of the communication links is mainly due to the following three factors:

- The surrounding environment leads to multipath fading and shadowing effects that contribute to signal strength attenuation;
- The interference due to concurrent transmissions from neighboring sensor nodes or networks using the same frequency band may hinder the desired transmission;
- The hardware of the radio transceiver may corrupt the sent and received signals due to the different imperfections of its building blocks.

The aforementioned factors motivated the need for measurement and characterization of the wireless link quality. In the literature, several research papers have focused on the study of the behavior of unreliable communication links using *Link Quality Estimation* techniques. Most of this prior work has focused on the establishment of link quality metrics for routing protocols and management of sensor node neighborhoods. The objective of the LQE mechanism is to provide a sensor node with a thorough knowledge of the surrounding channel conditions in order to select reliable links before data transmission.

Several LQE mechanisms have been reported in the literature. *Baccour et al.* [116] has classified these as either hardware-based or software-based estimators. An alternative LQE classification methodology might consist in distinguishing between single-frame and multiple-frame metrics. In [116], a detailed survey on link quality estimators dedicated to WSNs is presented. According to *Baccour et al.* the link estimation is evaluated through a three-step process, including link monitoring, link measurements, and metric evaluation.

1. **Link monitoring strategy**: The authors in [116] identify three kinds of link monitoring namely active, passive, and hybrid. In *active monitoring* mode, a sensor node evaluates the link quality by sending probe packets at a certain rate to its neighbors either in broadcast or unicast mode. Of course, a trade-off exists between energy efficiency and link estimation accuracy in these type of estimators. Estimators with a low sampling rate tend to be more energy efficient but less accurate than estimators with high sampling

rate which employ a large number of packet probes. Alternatively, in *passive monitoring* mode, a sensor node evaluates the link quality by listening to the existing packet traffic without sending probe packets. This LQE category tends to be more energy-efficient as it does not require additional communication overhead when used in duty-cycled WSNs. However, when the WSN operates at low data rate or with unbalanced traffic, which is often the case in many WSNs deployment scenarios, passive monitoring may lead to inaccurate link quality estimation due to the lack of up-to-date link measurements. Finally, *hybrid monitoring* combines both active and passive link monitoring techniques. This mix results in a better trade-off between energy-efficiency and accuracy.

2. **Link measurements**: Link measurement is the process of data collection from sent packets or received packets/acknowledgments. This can be done at transmitter- or receiver-side. Receiver-side link measurements use information extracted from received packets, such as packet numbers or time stamps. The measurements of transmitter-side link quality relies for example on packet retransmission count or other transmission attributes.

3. **Metric evaluation**: The retrieved data collected from the link measurements are analyzed to produce a metric that indicates the link quality. This could be, for example, the RSSI or the packet reception rate (PRR).

4.6.2 *LQE Requirements*

An efficient LQE metric must meet several requirements. These requirements ensure that the selected metric is able to estimate efficiently the link quality under any channel condition. Four important requirements are presented next [116].

- **Accuracy**: The main challenge facing LQE is the ability to capture the real behavior of the link. However, evaluating the accuracy of the link quality estimation is a complicated task due to the indefinite notion of "quality" when applied to volatile radio links. Indeed, there is no universal metric that measures the "*quality*" of the link. Many different quantifies of different nature can be used as LQE: e.g. RSSI (representing a power value in dBm), LQI (integer value), PRR (a probability). Hence, the accuracy of the LQE can only be studied indirectly using a comparative study. In the context of transceiver reconfiguration, the accuracy of the LQE metric will greatly impact the transceiver mode chosen and may induce a degradation in the communication performance.

- **Reactivity**: The LQE metric must react rapidly to changes in the link quality. The reactivity of a metric depends on two factors: (i) the observation window

w[1], and (ii) the link monitoring scheme. A large w is not able to capture short-time fluctuations. On the other hand, small w allows the LQE to be more reactive but may not capture the average link behavior and may result in needless overhead. In the context of reconfigurable transceivers, a reactive metric is essential for the process of reconfiguration to the fluctuation of the wireless link.

- **Stability**: On the contrary to reactivity, this property expresses the ability of an LQE to tolerate short-term changes in link quality. The stability of an LQE can be characterized by the coefficient-of-variation of the link estimate. The authors in [145] propose an easy approach to evaluate the LQE stability by calculating a weighted moving average with a large smoothing factor.

- **Power efficiency**: As power consumption is the most important parameter to take into account in WSNs, the LQE selected should have a low computation overhead in order achieve a minimum energy consumption. For example, estimation techniques based on learning and high beaconing rates are inappropriate for such networks and must be avoided because of their high power consumption and communication overhead.

In the context of reconfigurable transceivers, the reconfiguration manager might need different prediction windows in order to select the appropriate performance mode. Short time windows might be used to adapt the transceiver performance during packet reception or to choose the parameters of the next transmission while longer windows might allow the reconfiguration manager to estimate the general link quality based on a record of received/transmitted packets and consequently reduce the processing overhead due to frequent link quality estimation.

In the next section, we survey the most relevant LQEs suitable to be used in the process of transceiver reconfiguration.

4.6.3 *Overview of Existing LQE*

While several LQE techniques have been proposed in the literature, these have generally been used for higher layer protocols, e.g. topology control mechanisms, jamming detection, routing, handover, etc. As stated previously, *Baccour et al.* classified these link estimators into two categories: hardware and software LQEs. Generally, software-based LQEs are accurate in scenarios where the RF environment varies slowly but show poor performance due to the potentially long estimation time in volatile environments. Alternatively, hardware-based LQEs are fast but in-

1 The observation window w can be either a time window or a predefined number of sent/received packets depending on the context.

accurate [116].

An alternative, and perhaps more appropriate LQE classification methodology might consist in distinguishing between single-frame and multiple-frame metrics since this classification is independent of the implementation details. In this section, we review the most popular existing LQEs.

1. **Hardware LQEs**: Hardware LQEs are provided directly by the transceiver hardware, at the reception of a packet, without any additional processing cost. Moreover, these metrics can be measured without any specific traffic such as periodical broadcasts or beaconing. This category of LQEs is typically based on the measurement of the signal strength of a received signal, e.g. Received Signal Strength Indicator (RSSI) or the Link Quality Indicator (LQI) when available. Additionally, only a small number of samples are required to measure the link quality. However, several researchers have highlighted that hardware metrics fail to capture the real link quality [146]. Firstly, RSSI and LQI are subject to several fluctuation from the pro propagation channel [147]. Secondly, the link measurement highly depends on the hardware of existing transceivers. For example, the IEEE 802.15.4 standard does not specify clearly the RSSI mechanism in the radio transceiver. Thus, the information given by these metrics may differ from one transceiver to another due to hardware implementation of each manufacturer. For example, the RSSI obtained from the transceiver AT86RF230 [148] is given as

$$P_{RF} = RSSI\_Base\_VAL + 3 \times (RSSI - 1)[dBm] \tag{4.9}$$

where P_{RF} is the input RF power in dBm, $RSSI\_Base\_VAL = -91$ dBm, and $RSSI$ is the value read from the radio registers.

In the CC2420 radio transceiver [125], the RSSI is calculated in the $RSSI\_VAL$ register. The RSSI of the received signal specified by TI is given as

$$P = RSSI\_VAL + RSSI\_OFFSET[dBm] \tag{4.10}$$

where P is the received signal power and $RSSI\_OFFSET$ is approximately -45 dBm and can be found empirically during system development.

There exist several research activities that attempt to improve the efficiency of the measurements, e.g. RSSI calibration [149], however, these introduce a high calculation overhead not suitable for battery powered WSNs.

Another disadvantage of these metrics is that they are calculated based on a limited number of symbols (8 in the case of the CC2420), and are available only at the end of reception of successfully received packets, but not for

dropped or lost packets [150]. Finally, these metrics give only information about the signal power in the desired channel and are unable to predict the interference due to adjacent channels.

2. **Software LQEs**: In contrast to hardware LQEs which are obtained directly from the transceiver, software LQEs rely on historical packet count statistics to accurately estimate the channel quality. Several software LQEs have been proposed in the literature. In the following, we review the most relevant. The *Packet Reception Rate* (PRR) is a metric which is calculated at the receiver side and has been widely used in routing protocols [151] [152]. This metric is defined as the ratio of the number of correctly received packets to the number of transmitted (or expected) packets, for each time window ω.

$$PRR(\omega) = \frac{\text{Number of received packets}(\omega)}{\text{Number of expected packets}(\omega)} \tag{4.11}$$

The PRR has often been used as a reference metric to evaluate the accuracy of hardware LQEs. Each hardware LQE that correlates with PRR is considered as a good metric [116]. In [153], Cerpa et *al.* showed that the efficiency of the PRR mainly depends on the size of the observation time window ω. They showed that the larger is ω, the more accurate is the link estimation. Since PRR varies frequently, a new estimator has been introduced called Window Mean with Exponentially Weighted Moving Average (WMEWMA) [145]. The main goal of this metric is to smooth the fluctuations of the PRR. The WMEWMA is given by the following:

$$WMEWMA(\alpha, \omega) = \alpha \times WMEWMA(\alpha, \omega - 1) + (1 - \alpha) \times PRR \tag{4.12}$$

where α is the smoothness factor which is comprised between 0 and 1. This factor controls the importance of the current value of the PRR (with $\alpha < 0.5$) or the last value of WMEWMA (with $\alpha > 0.5$).

Similarly to the PRR, the *Required Number of Packet transmissions* (RNP) [153] metric is a receiver side LQE which estimates the link quality based on the average number of packet transmissions and retransmissions that are required for a successful reception in a time window ω.

$$RNP(\omega) = \frac{\text{Number of transmitted packets}(\omega)}{\text{Number of successfully received packet}(\omega)} - 1 \tag{4.13}$$

The RNP metric assumes that the link layer implements an ARQ (Automatic Repeat Request) protocol, which allows a node to retransmit a packet until it is correctly received. The number of correctly received packets is determined

only by the transmitter as the number of acknowledged packets. In [153], the authors claimed that the RNP characterizes the link quality more accurately than the PRR. However, RNP does not take the distribution of packet loss into account, which makes it unstable and less reliable in link quality estimation [116].

The *Expected Transmission Count* ETX metric introduced in [152] estimates the link quality based on the number of transmissions that are necessary to send a packet successfully. It uses active monitoring where each node collects statistical data by sending probe packets. This metric takes into account the link assymetry by combining the estimation of uplink PRR (denoted $\mathrm{PRR}_{\mathrm{uplink}}$) and downlink PRR (denoted PRRdownlink). The ETX calculated in time window ω is given as:

$$\mathrm{ETX}(\omega) = \frac{1}{\mathrm{PRR}_{\mathrm{uplink}} \times \mathrm{PRRdownlink}} \tag{4.14}$$

However, these models are relatively slow as they require a minimum number of packets (typically more than 5 packets) to build up a reliable link estimation.

3. **Hybrid LQEs**:

   A few hybrid estimators combining both software and hardware metrics such as PRR, link asymmetry, LQI and SNR have been proposed in the literature [154] [155] [156].

4.6.4 *Novel Link Quality Estimators for Reconfigurable Receivers*

Within the proposed reconfigurable transceiver, the reconfiguration mechanism must be driven by a link metric which correctly estimates the channel conditions. In the next section, we focus on the transceiver reconfiguration process, and more especially on link quality estimation and decision making. We define two new link quality estimators that can be used by the transceiver to update its parameters during the network run time while ensuring a minimum associated power consumption.

4.6.4.1 *Example Hardware Link Quality Estimator*

As discussed in Section 4.6.1, existing hardware metrics such as RSSI or LQI fail to estimate the link quality efficiently. The motivation of this section is to develop a new hardware link quality estimator that:

1. Estimates the channel noise and out-of-channel interference.

2. Allows the RF receiver to estimate the link quality for each symbol of the received packet.

3. Is sufficiently fast to allow the receiver to adapt dynamically (during packet reception) to the changing conditions of the channel.

4. Does not require a high computation and energy overhead.

5. Allows the receiver to select rapidly the right performance mode.

In order to build a good estimation of link quality, hardware LQE is required to approximate the true output SINR, defined as SINR^{GEN} (5.21) in the following chapter. However, in a real transceiver, it is hard to calculate this SINR^{GEN} due to the mixing of RF interference with the desired signal due to transceiver impairments. Therefore, the proposed LQE is built using a high-pass filtering path in the digital baseband as shown in Figure 4.12.

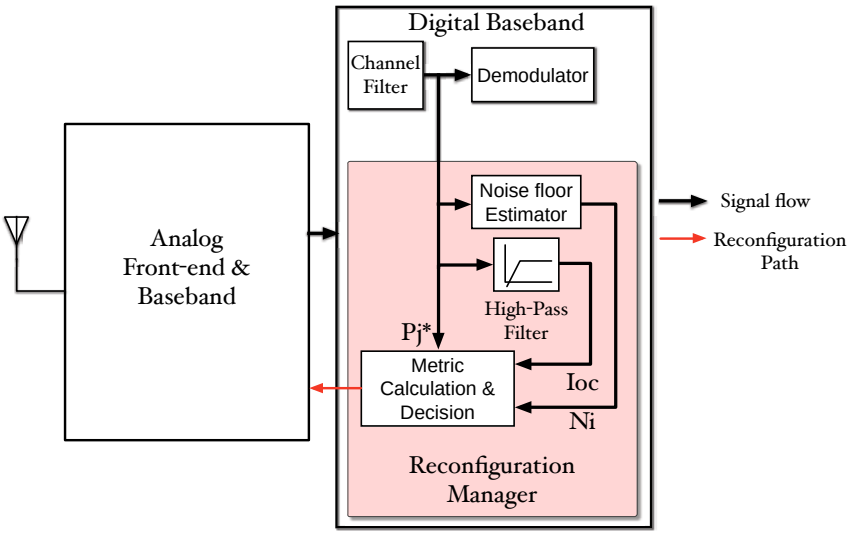


Figure 4.12: Implementation of hardware LQE in the digital baseband.

As can be seen from Figure 4.12, the proposed metric calculation and decision block measures the power of the in-channel signal P_j^* (Equation 4.15) as well as the residual adjacent interference power I_{oc} (Equation 4.16). Indeed, since the sum of the in-channel interference and noise are indistinguishable from the received signal, the total power at the output of the digital channel filter is

$$P_j^* = P_j + N_i + \overbrace{\sum_{k\neq i,j} \alpha_{i,k} \cdot P_k}^{\text{out-of-channel interference}} + \overbrace{\sum_{k\neq l,j} \sum_{k\neq k,j} \beta_{i_{k,l}} \cdot P_k P_l^2}^{\text{Intermodulation interference}} + \overbrace{P_{PN}}^{\text{Phase-noise interference}} \tag{4.15}$$

where P_j is the desired signal power, N_i is the receiver noise figure, $\alpha_{i,k}$ is the rejection factor between channel i and k, $\beta_{i_{k,l}}$ is the intermodulation coefficient, and P_{PN} is the phase noise interference power.

$$I_{oc} = \sum_{k \neq i,j} \alpha_{i,k}|_{\alpha \neq 1} \cdot P_k \tag{4.16}$$

An estimate of the noise-floor N_i, constructed by monitoring and averaging the in-channel signal strength before or after packet reception, is also an input to the metric calculation & decision block. Since N_i and I_oc can be extracted from the in-channel signal, the proposed LQE metric is

$$LQE_{Reconfig} = \frac{P_j^* - (N_i + I_{oc})}{N_i + I_{oc}} \tag{4.17}$$

This hardware LQE block is characterized by a power consumption overhead and a calculation delay. For high reactivity, the shortest possible delay is preferable. Figure 4.13 shows the effect of adjacent and alternate interference on the SINR and the $LQE_{Reconfig}$ in the case where the receiver is in low performance mode and the desired signal power of -85 dBm. In this figure, the effect of intermodulation is illustrated. In the case of intermodulation absence, the $LQE_{Reconfig}$ curve fits with the SINR one, however, when intermodulation interference is present, the $LQE_{Reconfig}$ estimates the SINR only if the interference power is under -50 dBm.

As shown in Figure 4.13, since co-channel interference, intermodulation distortion and reciprocal mixing noise cannot be extracted from the in-channel signal power, the above LQE provides an erroneously optimistic estimation of $SINR_{gen}$ when these interference mechanisms are strong. Since in such case reception quality is severely impaired, the reconfiguration event that may ensue (and further lower the $SINR_{gen}$) should not impair system performance.

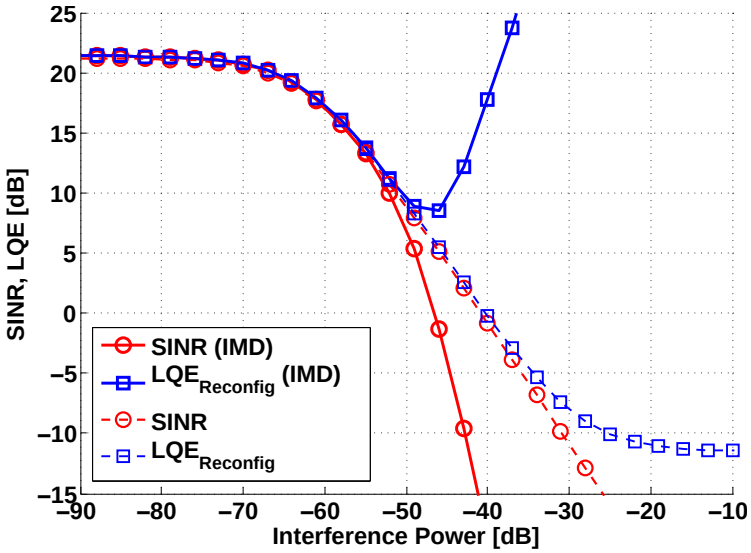


Figure 4.13: Variation of the reconfiguration metric $LQE_{Reconfig}$ and the SINR as function of interference power in the case of intermodulation presence and without intermodulation.

This new hardware LQE could be used in, for example, a simple reconfiguration algorithm based on two predefined thresholds: LQE^{th}_{Low} and LQE^{th}_{High}. At the beginning of each packet reception, the receiver performance is set to high performance mode. After LQE_{Delay}, if $LQE_{Reconfig} > LQE^{th}_{High}$, the performance mode is lowered. During packet reception, $LQE_{Reconfig}$ is continuously monitored and if $LQE_{Reconfig} < LQE^{th}_{Low}$, the performance mode is increased. Of course, a more complex reconfiguration algorithm could also be used. Based on this simple algorithm, assuming $LQE^{th}_{Low} = 3dB$ and $LQE^{th}_{High} = 9dB$ and using the reconfigurable receiver model of Table 4.7, Figure 4.14 shows the receiver power consumption as function of the interference and desired signal power. In this figure, we see clearly that when the interference power I_{oc} is under -60 dBm the receiver operates in low performance mode whatever the value of P^*_j in the interval $[-85\ dBm, -55dBm]$, however, when I_{oc} is larger than -40 dBm the receiver performance will switch to moderate of high performance mode depending on the P^*_j power.

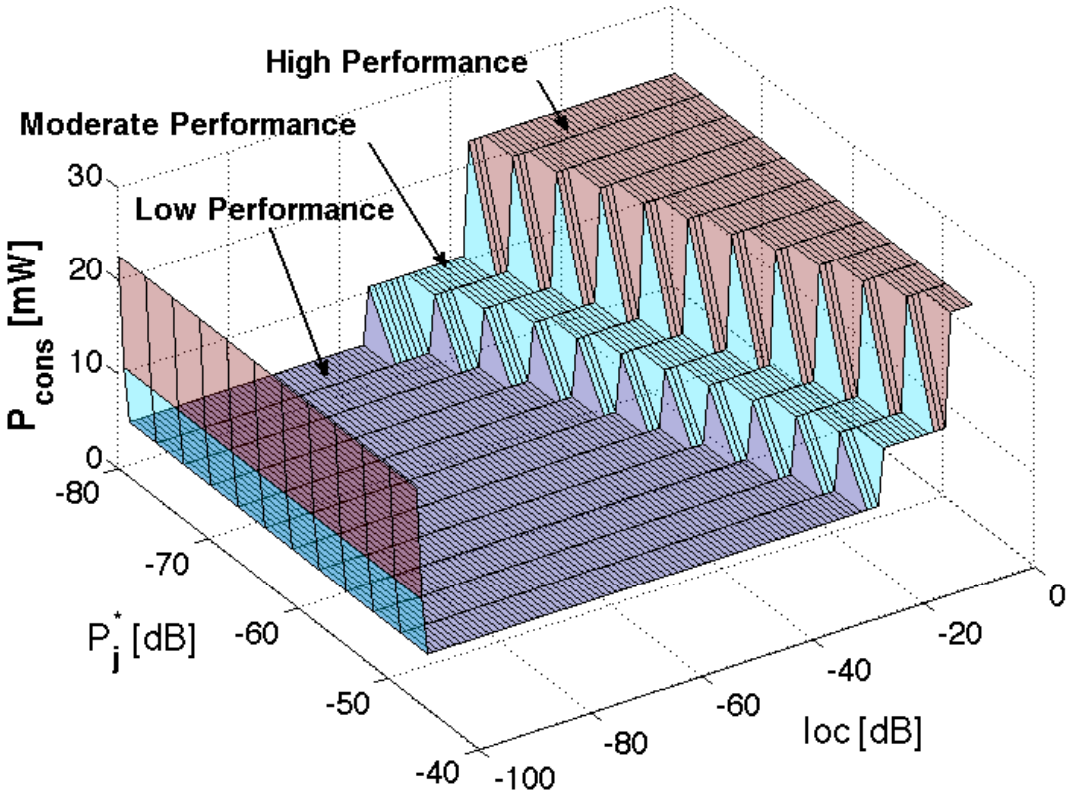


Figure 4.14: Variation of receiver power consumption as function of the desired signal and interference power.

4.6.4.2 *Example Hybrid Link Quality Estimator*

In this section, we present a hybrid LQE which combines the hardware LQE measurement described in the previous section with a PRR measurement. The objective of this LQE is to provide an accurate and highly reactive link quality estima-

tion. In order to limit the power consumption associated to the hardware LQE calculation, the value of $LQE_{Reconfig}$ from Equation 4.17 is calculated only once, for a duration of 32µs after the middle of the packet as shown in Figure 4.15.

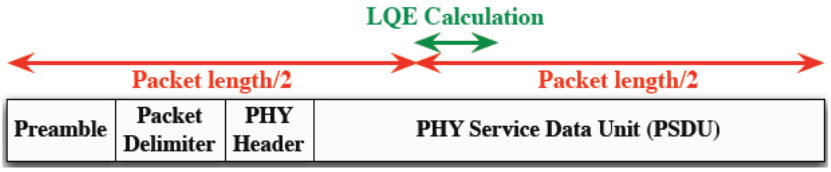


Figure 4.15: $LQE_{Reconfig}$ calculation for new hybrid LQE.

This measurement, performed for each received packet, is used in conjunction with the PRR calculated over an observation window w as shown in Figure 4.16. A possible control algorithm using this hybrid LQE is shown in Figure 4.19.

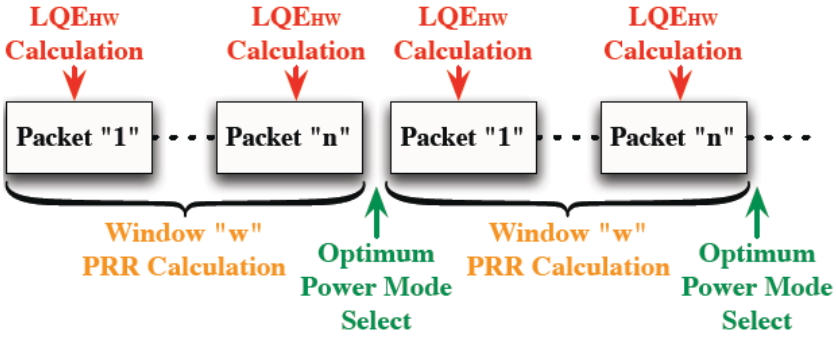


Figure 4.16: Combination of hardware and software LQEs.

4.7 TRANSCEIVER RECONFIGURATION STRATEGIES

In this section, we describe potential reconfiguration strategies for both the transmitter and the receiver.

4.7.1 *Transmitter Reconfiguration Strategies*

As mentioned in Section 4.4.1, the most commonly reconfigurable block of a transmitter is the PA. In this section, we focus on the establishment of a reconfiguration strategy that allows the transmitter to adapt the power amplifier output power as a function of the channel conditions. In the literature, extensive research studies [126] [157] [158] [159] have been conducted in order to reduce the excessive power consumption of a network by adjusting the radio transmission power of individual nodes.

This technique known as Transmission Power Control (TPC) suggests that each sensor node should control its transmission power in order to ensure a minimum power consumption while guaranteeing that the transmission power is sufficient to communicate with other nodes. A good transmitter reconfiguration strategy for WSNs should provide an energy-efficient mechanism to reduce the transmission power thus ensuring a low power consumption and less RF interference to neighbor networks.

The TPC can be accomplished only by a negotiation between the transmitter and the receiver. Most proposed TPC techniques use signal strength metrics such as RSSI or SNR computed over incoming packets and compare the measured value to a predefined threshold to determine the optimum transmission power. In [158], the authors determine the required transmission power for stationary nodes using the RSSI which has a correlation with the packet reception ratio (PRR). In theory, any other LQE technique (Section 4.6.1) can be used. For example, instead of using the RSSI measured by the receiver (Radio B) and fed back to the transmitter (Radio A), an alternative approach could consist in estimating directly in Radio A the SINR of the acknowledgement packet sent by Radio B using the LQE_{HW} from Equation 4.17 and using this information to reconfigure the transmitter (Figure 4.17).

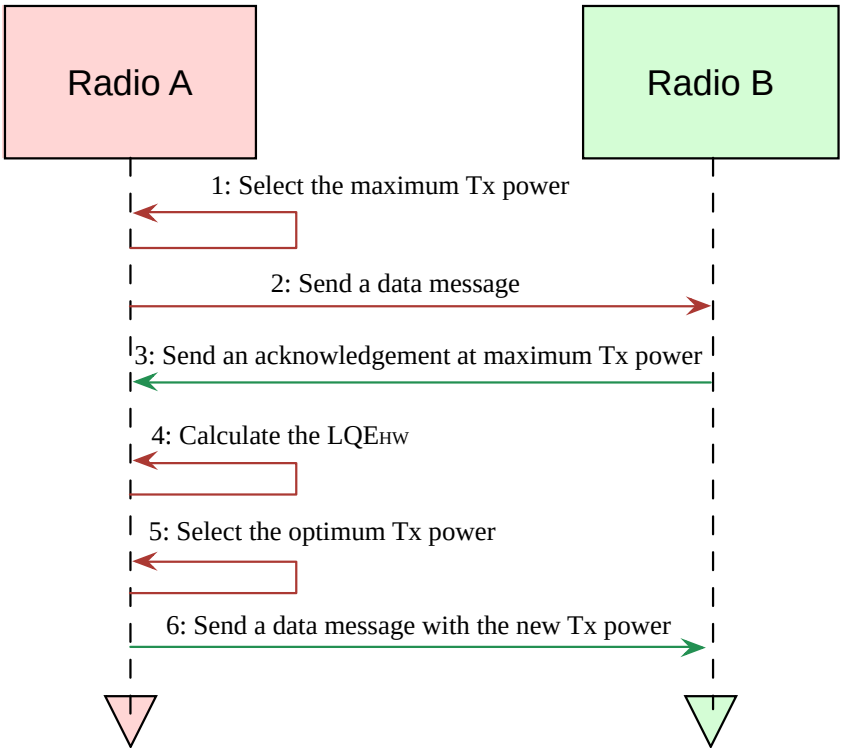


Figure 4.17: Sequence diagram of transmission power adaptation.

We consider that radio A is the reconfigurable transmitter and radio B the receiver. Before the establishment of the link, the radio A initializes its transmitter power at a maximum in order to insure that the transmitted packet will reach the radio B. After radio B receives a data message from radio A, radio B sends an acknowledgment to confirm the reception of the message. During the reception of this acknowledgement packet, Radio A invokes the reconfiguration manager which measures the LQE_{HW} and computes the optimum transmission power. After radio A finishes the configuration of the new settings, it sends a new data message to the radio B.

4.7.2 *Receiver Reconfiguration Strategies*

Receiver reconfiguration strategies are based on an adaptive change of the receiver settings as a function of a chosen LQE. Existing LQE techniques are based either on hardware measurements such as Received Signal Strength Indication (RSSI) or Link Quality Indicator (LQI), or on software measurements such as Packet Reception Rate (PRR), or on a combination of hardware and software measurements [116]. In practice, many other different types of software and hardware LQE's could be used to estimate link quality (packet CRC failure rate, acknowledgement failure, re-routing probability, etc.). However, part of the overhead induced by reconfiguration is due to the LQE measurement which therefore must be correctly modeled.

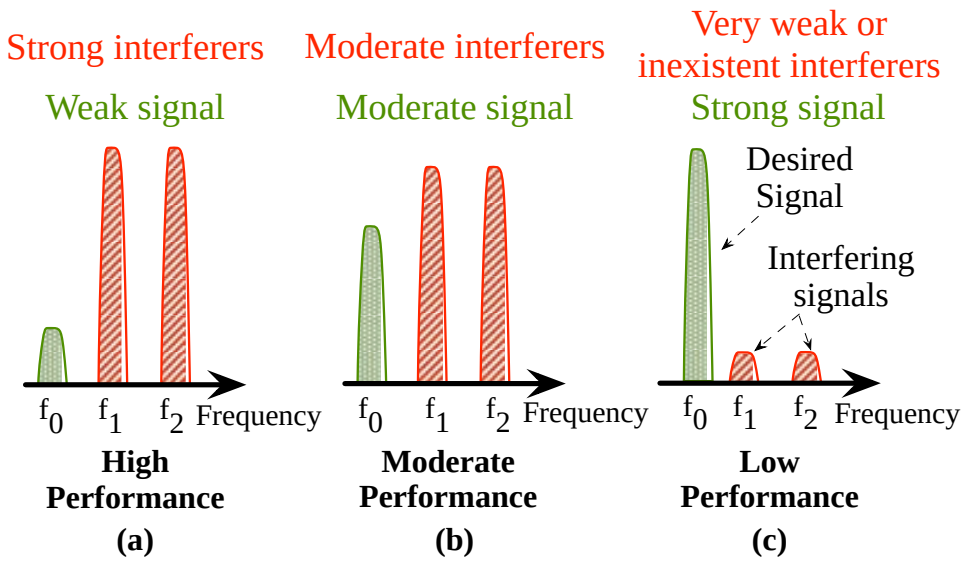


Figure 4.18: Signal and interference scenarios.

Without loss of generality, in the following, we assume that the receiver is supposed to be able to reconfigure itself in three performance modes: high, moderate and low, as a function of the three typical signal/interference propagation scenar-

ios illustrated in Figure 4.18. In (a), the signal is weak and the adjacent channel interferers are strong. The receiver must therefore operate in high performance mode which minimizes noise floor and phase noise while selectivity, linearity, and dynamic range are maximized. In (b), the moderate signal and interferer levels imply that the noise floor, linearity, phase noise and dynamic range requirements can be relaxed (i.e. moderate performance mode), hence resulting in power consumption savings. In (c), the strong signal and weak adjacent channel interferers imply that the receiver specifications can be further relaxed along with selectivity (i.e. low performance mode), thus providing a maximum of power savings. As stated before, the reconfigurable receiver is used in order to save power consumption when the channel conditions are favorable.

We distinguish two main families of reconfiguration strategies: dynamic reconfiguration (or intra-packet reconfiguration) and static reconfiguration (inter-packet). In the dynamic reconfiguration family, we assume that the receiver is able to reconfigure itself during packet reception, whether continuously, periodically, a single time, in case of event detection, etc. Static reconfiguration assumes that the receiver can change its settings only between two packet receptions.

4.7.2.1 *Receiver Static Reconfiguration*

In this reconfiguration strategy, the receiver settings are updated periodically (but not during packet reception) based on a software, hardware or hybrid LQE. For example, the hybrid LQE defined in Section 4.6.4.2 could be used. An important point concerning static reconfiguration is that the receiver performance mode may remain constant for several transmissions (potentially a large number) depending on the desired LQE measurement period. Note that in this type of reconfiguration, the VCO can be reconfigured along with the other blocks if the phased-locked loop (PLL) performance can be guaranteed.

From an RF design perspective, static reconfiguration implies that, before each packet reception, the receiver blocks are configured in a given mode which remains constant during the entire packet reception. This means that controlling the parameters of the transceiver blocks can be performed simply and with low requirements on delay.

However, when the receiver is configured in its lower performance modes, this implies that its sensitivity is momentarily reduced. This has a direct impact on the network which might lose connectivity if mechanisms are not provided to return periodically to high performance made.

For example, a statically reconfigurable receiver might employ the hybrid LQE defined in Section 4.6.4.2 in the control algorithm illustrated in Figure 4.19. If the $LQE_{Reconfig}$ measurement for each packet is greater than a threshold LQEth and if all packets are correctly received (PRR = 100%) the receiver performance mode is lowered. Otherwise, the performance is switched directly to high performance mode. For higher reactivity, the receiver switches directly to a higher performance mode if the $LQE_{Reconfig}$ measurement of a packet is below LQEth or if a packet is lost before the end of the window w.

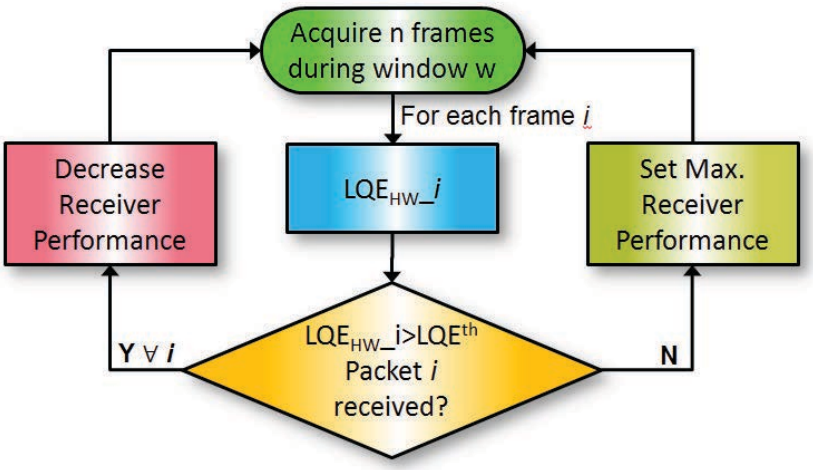


Figure 4.19: Control algorithm using new hybrid LQE for a statically reconfigurable receiver

4.7.2.2 *Receiver Dynamic Reconfiguration*

As stated previously, in the dynamic reconfiguration family, we assume that the receiver is able to reconfigure itself during packet reception, whether continuously, periodically, a single time, etc. For example, assuming that an LQE can be calculated during packet reception (most probably in the digital baseband section of the transceiver), "a continuous dynamic" reconfiguration strategy consists in continuously calculating this LQE, such as the SINR estimation defined in 4.17, and continuously adapting the receiver performance mode during packet reception without service interruption. If the power consumption associated with the continuous LQE measurement is low, dynamic reconfiguration potentially offers the largest power savings and, unlike static reconfiguration, ensures that no packets are lost due to reconfiguration (if the system is sufficiently reactive). Of course, this reconfiguration is prospective since it is still necessary to prove that it is possible to change the receiver's settings during packet reception without impacting the reception quality (see [129] for initial work in this area).

Note that while some form of fast hardware LQE is necessary for the dynamic reconfiguration family, hybrid LQEs can also be used. For example, a reconfiguration algorithm might enable or disable the dynamic reconfiguration of the receiver depending on network statistics observed over a large number of packets.

While modelling a dynamically reconfigurable receiver, it is necessary to account for the hardware LQE calculation delay which might cause an SINR drop at the sudden appearance of blockers. Indeed, this may temporarily increase BER and therefore PER.

A potential continuous dynamic reconfiguration algorithm that employs the hardware LQE presented in Section 4.17 is defined in Figure 4.20 and illustrated in Figure 5.14: The receiver performance is set to high mode at the beginning of each packet reception. After $\mathrm{LQE}_{\mathrm{Delay}}$ which accounts for the processing speed of the LQE_{HW} calculation block, an initial LQE_{HW} is calculated which switches the performance mode (here from high to moderate). Since the receiver settings are modified, the $\mathrm{SINR}_{\mathrm{out}}$ is affected which in turn leads to a new LQE_{HW} calculation. In this example, since the new value is between $\mathrm{LQE}^{\mathrm{th}}_{\mathrm{Low}}$ and $\mathrm{LQE}^{\mathrm{th}}_{\mathrm{High}}$, no new reconfiguration event occurs until the appearance of a strong adjacent channel interferer. The resulting $\mathrm{SINR}_{\mathrm{out}}$ drop leads to a corresponding change in the LQE_{HW} which here falls below $\mathrm{LQE}^{\mathrm{th}}_{\mathrm{Low}}$, hence resulting in a performance mode switched back to the high setting.

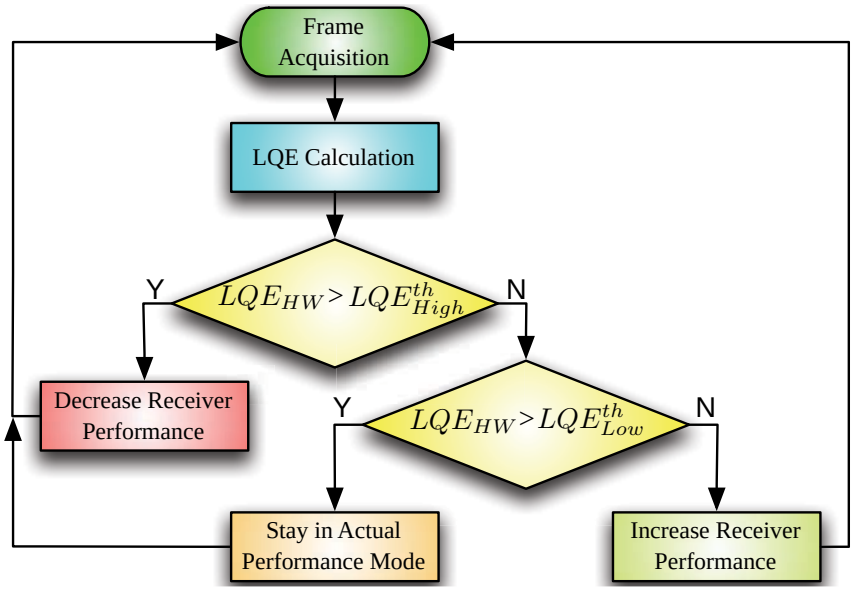


Figure 4.20: Control algorithm for a continuous dynamically reconfigurable receiver

4.7.3 *Combined Transmitter/Receiver Reconfiguration Strategies*

As discussed above, the goal of transmission power control is to minimize the power consumption of the transmitter. However, the decrease of the transmitter power will lower the SNIR of the signal received by the receiver. If this last is a reconfigurable receiver, this will force it to function in a high performance mode. Recall the scenario shown in Figure 4.17. An optimum solution consists in pursuing the negotiation between the two radios until an agreement on the performance of each radio is reached. In other words, the radio B must calculate the optimum reception performance and make a proposal to radio A. If the proposal is accepted, then the radio A changes its transmission power to the proposed value. Ideally, the solution found minimizes the power consumption of both Radio A and B.

Note that TCP is a static type of reconfiguration strategy whereas the receiver can potentially be reconfigured either statically or dynamically. The research opportunities offered by combined transmitter/receiver reconfiguration strategies are immense and are unfortunately beyond the scope of this present work.

4.8 CONCLUSION

In this chapter, we have presented a reconfigurable transceiver model that changes its performance and power consumption according to the channel conditions. The transceiver model proposed is based on Figure of Merits of measured circuits reported in the literature. The receiver performance modes are chosen based on a link quality estimation that provides a good approximation of the real SINR. We showed that considerable power can be saved in the case of good channel conditions. In the next chapter, we show that the models proposed can easily be implemented in a wireless network simulator in order to validate the value of a reconfigurable architecture according protocol models and real-world deployment scenarios.

5

ENVADAPT SIMULATION FRAMEWORK FOR POWER-RECONFIGURABLE TRANSCEIVERS

Contents

5.1 INTRODUCTION

In the previous chapters, we have presented the models, mechanisms, and algorithms that allow a radio transceiver to adapt itself to the channel conditions. We have shown also that adapting the radio transceiver to the channel's changing conditions might provide large gains in power consumption and potentially reduces the interference to neighboring sensor networks.

To evaluate the performance and behavior of the reconfigurable transceiver in large-scale WSNs, we extend the WSNet simulator by integrating the interference and radio components that are required to support the simulation of such adaptive transceivers. The choice of simulation and modeling over an experimental method can be attributed to several reasons. Firstly, reconfigurable RF transceivers are in their initial stages of development and commercial circuits are still not available. Secondly, transceiver modeling and interference simulations allow the analysis of relatively large scale networks. This eases the study of interference due to adjacent networks. Thirdly, the lifetime of hundreds of sensor nodes can thus be simulated. Therefore, this work relies on simulation and modeling of adaptive transceivers in order to evaluate the different control mechanisms and reconfiguration strategies.

5.2 CHALLENGES FOR WSN PHYSICAL LAYER MODELING AND SIMULATION

As wireless communication systems are becoming more and more complex, accurate simulations of the physical layer and channel conditions play a fundamental role in determining the communication performance in a WSNs. The simulations of the PHY layer usually consist in the evaluation of the BER or PER of the transmitted packets that may have been corrupted by the modeled transmission channel. The natural candidate for these simulation is MATLAB. However, as mentioned in Chapter 3, Matlab fails to model the protocol stack of a WSN. Thus, in this chapter we focus on PHY layer modeling within a WSN simulator.

A detailed (spice-level) physical layer model cannot be included in the network simulator because of processing complexity and simulation time limitations. Therefore, an accurate abstraction model of the PHY layer that encapsulates the major impairments of a real transceiver must be proposed. The selected abstract model must account for the radio standard and access technique, the modulation scheme, the channel models used, and other radio parameters. Since relatively simple models are needed to limit simulation time, our idea consists in implementing in a WSN simulator a behavioral model of the transceiver and a corresponding interference model that together approximate the behavior of a real transceiver.

5.2.1 *PHY Layer of Existing WSN simulators*

The PHY layer of a WSN simulator is responsible for packet sending and receiving, channel sounding, and SINR calculation. As discussed in Chapter 3, existing network simulators presently in use such as Castalia [84], Mixim [160], NS3 [81], offer advanced and complete simulation environments to evaluate, test, and debug any kind of network protocol and wireless system. However, the design choices

made during the design of these simulators oriented these toward high level abstract views of the wireless channel and sensor network components, in order to achieve a fast network simulation. This problem has been addressed by *Ben Hamida et al.* in [161]. They highlighted the trade-off between PHY layer modelling accuracy and scalability in WSNs simulators. The same problem was studied by the authors in [162] who compared the results obtained from a selected set of simulators (Castalia, Mixim, Tossim and WSNet) to a series of indoor and outdoor experiments using MICAz sensor nodes. They found that the simulation results vary drastically from one simulator to another. This is mainly due to the lack of accuracy and reliability of the models used to represent the characteristics of the PHY layer. As shown in Table 5.1, the radio channel modeling and interference calculation is the point where current simulators differ largely [21].

Simulation environments	Radio range modeling			Radio link modeling		Interference modeling	
	pathloss	shadowing	fading	link model	modulation	model	SINR computation
NS-2	free space, two-ray	log-normal	rician, rayleigh	threshold	-	limited	strongest signal
GloMoSim [6]	free-space, two-ray	log-normal	rician, rayleigh	threshold, BER	BPSK, QPSK	limited	adaptive
JiST/SWANS [7]	free-space, two-ray	-	rician, rayleigh	threshold, BER	BPSK	limited	cumulative
GTSNetS [8]	free-space, two-ray	-	-	threshold	-	limited	strongest signal
WSNet [23]	free-space, two-ray	log-normal	rician, rayleigh	threshold, BER	BPSK, STEP, OQPSK, FSCK	limited, full	adaptive, cumulative

Table 5.1: PHY layer modeling in common simulation environments: NS, JiST/SWANS, GloMoSim, GTSNetS and WSNet [21].

In the following section, we focus on the description of interference models for WSNs simulators.

5.2.2 *State-of-the-art interference models*

Interference is one of the most important causes of performance degradation in WSN communications. Understanding the sources of wireless interference is essential to the design and management of the reconfigurable transceiver. However, accurate measurement of interference is extremely difficult due to its random occurrence outside and inside the radio transceiver (Figure 5.1) and also to the complex interaction of the radio signals in the environment. Interference is due to the following factors: (1) the first one is due to the increasing number of wireless technologies that use the same spectrum band that may communicate with all devices existing in their range and consequently interfere with other networks. (2) The second important interference phenomenon results from the multipath effect in the propagation environment (i.e. fading, shadowing) where the signal received at the antenna is a combination of the reflected, scattered, diffracted and randomly delayed components of the transmitted signal. (3) The third interference effect is

encountered internally in the radio transceiver due to the physical imperfections of the electronic components.

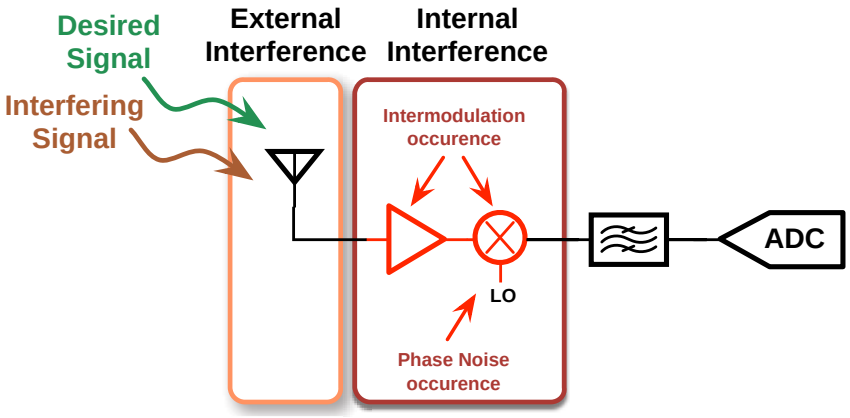


Figure 5.1: Simple block diagram of a wireless receiver.

In this work, interference modeling and characterization are mainly used for two purposes. First, to define the performance of each building block of the transceiver in terms of linearity, noise figure, gain, etc. Second, to build a reconfiguration metric which defines the transceiver parameters required for a successful packet transmission or reception based on an accurate estimation of interference level and desired signal power. Developing an accurate interference model has significant importance in the specification of each operating mode of the adaptive transceiver which is required to operate usually at minimum SINR. The performance mode selected by the reconfiguration manager is mainly articulated around the radio subsystems which is charged to carry out the various processing necessary to describe the interference level and the communication performance in the surrounding channel. Therefore, an accurate modeling and characterization of the interference allows the identification of the performance degradation source but also an important step towards the design, analysis and implementation of a reconfigurable transceiver.

Interference in WSN has been investigated extensively in the research literature using several models. The choice of the interference model is of fundamental importance. Not only has the proposed model to reflect the nature of real communications, but also to facilitate the development of rigorous analysis.

The performance of a wireless communication system is commonly measured using the packet-error-rate (PER) which is usually expressed as a percentage. Within a WSN simulator, the PER is calculated by assuming that the bit error probability (BER or bit-error-rate) of each bit within the packet is independent. In all communication systems, the BER is a function of the demodulation scheme as well as the signal-to-interference-plus-noise ratio (SINR) at the demodulator input.

Therefore in order to analyze the performance of a communication system, it is necessary to model accurately this SINR. One model widely used in the literature [163] [164] [165] is the *standard* SINR (or $SINR^s$).

$$SINR^s = \frac{P_{signal}}{N + P_{interference}} \tag{5.1}$$

where P_{signal} and N are respectively the desired signal and the thermal noise power at the output of the receiver front-end, while $P_{interference}$ is interference power due to co-channel and out-of-channel interference.

In practice, signal reception often takes place in the presence of different types of interference, the most commonly studied being co-channel and adjacent channel interference. Co-channel interference (CCI) is caused by undesired signals using the same channel as the receiver. This type of interference is the most harmful and cannot be rejected or attenuated even with an ideal filter as it overlaps with the desired signal channel.

Adjacent channel interference (ACI) results from signals adjacent in frequency to the desired channel. Even if the transmission mask is ideal, interference coming from adjacent channels may appear at the receiver. This phenomenon is illustrated in Figure 5.2. The effect of ACI is caused by the receiver non-ideal receiver filters (analog and digital) that allow nearby interferers to leak into the channel of interest. Additionally, the effect of ACI depends on the power of signals in the adjacent channels as well as the frequency offset between the desired signal and the interferers.

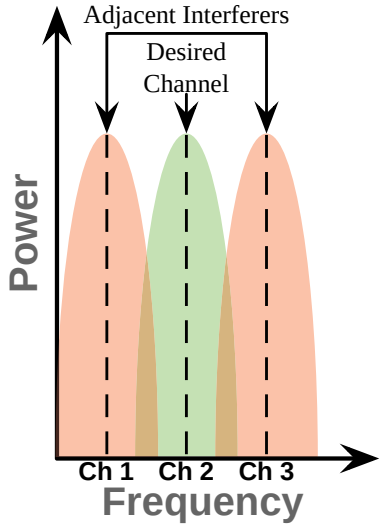


Figure 5.2: Interference of adjacent radio channels.

An *enhanced* SINR (or $SINR^e$) model that takes into account these two types of interference was defined in [21]. The $SINR^e$ is calculated for node i when receiv-

ing the desired signal emitted by node j and while other nodes (denoted k) are simultaneously emitting a signal (Figure 5.2). The SINR^e is given by :

$$\mathrm{SINR}^e_{i,j} = \frac{P_j}{N_i + \sum_{k \neq i,j} \alpha_{i,k} \cdot P_k} \tag{5.2}$$

where P_j is the power of the desired signal, P_k is the power of the k^{th} interferer signal and $\alpha_{i,k}$ is a rejection factor that emulates the channel selectivity of node i and is therefore a function of the frequency offset between the RF channels occupied by nodes i and k.

All signal powers are referred to the antenna of node i. The received signal power P_j and P_k are calculated using an appropriate channel model for the application scenario. This may be a simple pathloss model or a more elaborate statistical model to take into account fading and shadowing effects in a mobile scenario. The results of simulation using the SINR^e model are reported to be in good agreement with real-world experiments [166]. However, neither of these models consider the other imperfections of the receiver, for example the possibility that the inherent non-linearity of the receiver (Figure 5.1) will create interference which may also corrupt the desired signal.

In the following, the emphasis is placed upon integration of transceiver impairments in WSNs simulation by extending the SINR model of (5.2).

5.3 RECEIVER IMPERFECTION-BASED INTERFERENCE MODEL

As stated before, other forms of interference are caused by the impairments of the receiver building-blocks that may corrupt the system performance. In this section, we focus on describing the models of the most essential impairments typically encountered in RF receivers and their impact on the communication performance of WSNs. These impairments include intermodulation, phase noise, and quantization noise. In fact, there exists other impairments which mainly depend on the receiver architecture and are not considered in this work such as IIP2 in a Zero-IF receiver, or image band in Low-IF receiver [144]. Altogether, these impairments play a major role in the definition of the performance modes of the reconfigurable transceiver.

5.3.1 *Intermodulation interference*

Intermodulation products are produced when two or more in-band signals are applied to the input of a nonlinear receiver. To analyze intermodulation, we con-

sider the case of a memoryless nonlinear circuit which is described by the transfer function given in (Equation 5.3). Because the higher order products are of less significance, the transfer function order is limited to 3 in this discussion.

$$y(t) = \sum_{n=1}^{3} k_n x(t)^n \tag{5.3}$$

where k_n describes the coefficients of the different polynominal orders, and $x(t)$ is the input signal, which is assumed to consist of two sinusoids of different frequencies such that

$$x(t) = A_1 \cos(2\pi f_1 t) + A_2 \cos(2\pi f_2 t). \tag{5.4}$$

Amplitudes and corresponding frequencies of the components of the output signal $y(t)$ are summarized in Table 5.2.

Table 5.2: Intermodulation products and their amplitudes.

Order	Frequency	Component Amplitude
1^{st} order	f_1	$k_1 A_1 + \frac{3}{4} k_3 A_1^3 + \frac{3}{2} k_3 A_1 A_2^2$
	f_2	$k_1 A_2 + \frac{3}{4} k_3 A_2^3 + \frac{3}{2} k_3 A_2 A_1^2$
2^{nd} order	$f_1 \pm f_2$	$k_2 A_1 A_2$
3^{rd} order	$2f_1 \pm f_2$	$\frac{3}{4} k_3 A_1^2 A_2$
	$2f_2 \pm f_1$	$\frac{3}{4} k_3 A_1 A_2^2$

As shown in Table 5.2, the output signal $y(t)$ contains second-order intermodulation components at the sum and difference of the two frequencies f_1 and f_2. These will be neglected in the following since they either fall out of the receiver passband or, for the product at frequency $f_1 - f_2$ which falls close to direct-current (DC), are an issue for certain types of Zero-IF or Low-IF receivers only [144]. In this work, we consider only the terms at frequencies $2f_1 - f_2$ and $2f_2 - f_1$ which are commonly referred to as third-order intermodulation distortion (IMD_3) products. As depicted in Figure 5.3, if the distance between the frequencies of two in-band interferers (f_1 and f_2) is similar to the distance between one of the interferers and the useful signal at f_0, one of the resulting intermodulation products may fall close or within the receiver channel and may corrupt the desired signal.

Using Table 5.2, the IMD_3 is given

$$IMD_3 = \frac{3}{4} k_3 A^3 \; ; \; A = A_1 = A_2. \tag{5.5}$$

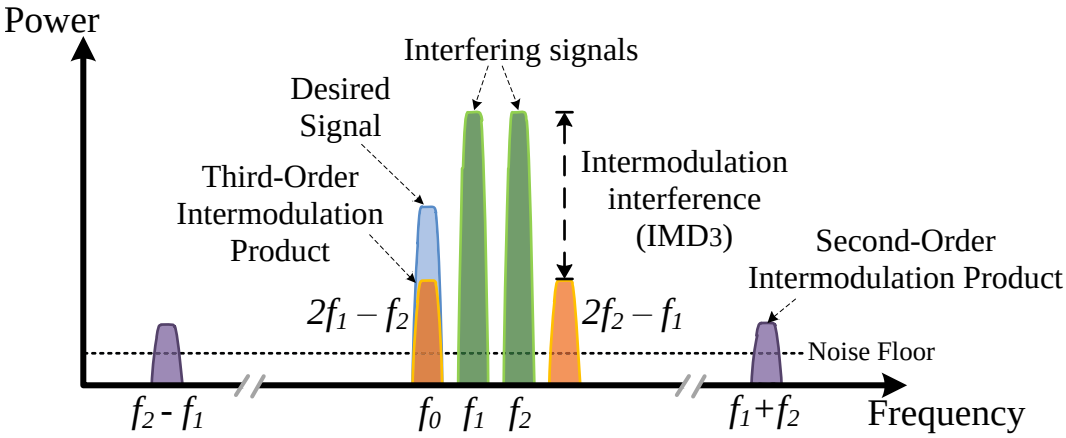


Figure 5.3: Power spectrum of intermodulation product.

Next, we analyze the relationship between the third-order input intercept point IIP_3 and IMD_3. IIP_3 is a commonly used metric for defining the linearity of an RF receiver and it corresponds to the input-referred interferer power which generates intermodulation products of equal strength as the linear components. The IIP_3 is given by [144]

$$IIP_3 = \frac{1}{2}(3A - IMD_3) \tag{5.6}$$

$$-2 \cdot IIP_3 = 10 \log\left(\frac{3}{4}k_3\right) \tag{5.7}$$

Using (5.6) and (5.7), we derive a new formulation of IMD_3 such as

$$IMD_3 = 10^{(-2IIP_3/10)} \cdot A^3 \tag{5.8}$$

In practise, if $A_1 \neq A_2$, it is necessary to define two IMD_3 products

$$IMD_3(f) = \begin{cases} 10^{(-2IIP_3/10)} \cdot A_1^2 A_2\ ; & f = 2f_1 - f_2 \\ 10^{(-2IIP_3/10)} \cdot A_1 A_2^2\ ; & f = 2f_2 - f_1 \end{cases} \tag{5.9}$$

Since the IIP_3 value for a given RF receiver is often published on its datasheet, Equation 5.9 can be used to find the magnitude of the IMD_3 products for the interferer powers A_1 and A_2. To avoid the interference caused by third-order distortion, the receiver usually needs a relatively high IIP_3. However, for an RF receiver, there is generally a link between higher IIP_3 and increased power consumption.

As shown in Figure 5.3, intermodulation products create unwanted components at frequencies which may interfere with the desired signal. In a worst case, the IMD_3 product will fall directly into the desired channel, a phenomenon similar to co-channel interference. As the frequency offset between the IMD_3 product and

the desired signal increases, their contribution to the SINR degradation becomes less harmful (in a similar manner as a adjacent channel interference).

A model for SINR between nodes i and j including the distortion due to 3^{rd} order intermodulation can therefore be derived as

$$SINR_{ij}^{IMD} = \frac{P_j}{N_i + \sum_{k \neq i,j} \alpha_{i,k} \cdot P_k + P_{IMD_3}} \tag{5.10}$$

P_{IMD_3} represents the power of interference caused by third-order distortion products and is expressed as

$$P_{IMD_3} = \sum_{k \neq l,j} \sum_{l \neq k,j} \beta_{i_{(k,l)}} \cdot P_k P_l^2 \tag{5.11}$$

where $\beta_{i_{(k,l)}}$, similarly to $\alpha_{i,j}$ above, is a rejection factor that varies with the frequency offset between the desired signal channel and the resulting third-order intermodulation product at frequency $2f_k \pm f_l$. The sum extends to all interference components resulting from the intermodulation of each couple of interferers.

In order to evaluate the impact of the IMD_3 products, in the following, we will assume that the IMD_3 products that fall within the desired receiver channel are treated as co-channel interference while the ones that fall out of the desired channel are treated as adjacent interference. In an RF receiver, this assumption implies that all of the intermodulation products are created before any channel filtering is applied and that the channel filtering operation is perfectly linear. Thus, the intermodulation rejection factor $\beta_{i_{(k,l)}}$ can be expressed as a function of receiver IIP_3 and $\alpha_{i,j}$ such that

$$\beta_{i_{(k,l)}}\ [mW] = \alpha_{i,2k-l} \cdot 10^{(-2IIP_3/10)}. \tag{5.12}$$

The implementation of $SINR^{IMD}$ model in a network simulator requires the knowledge, at a given time, of all the RF-band signals concurrently received by a given node. In this work, the network simulations are run using the wireless sensor simulator WSNet [167]. Thanks to its native support of adjacent channel interference, WSNet requires only minor modifications to implement the $SINR^{IMD}$ model. In fact, intermodulation calculation, reduces the simulation speed only by 6% for 3 channels and by 20% for 16 channels.

Note that none of WSN simulators reported in the literature (such as OMNet++, NS-3, etc.) support intermodulation interference calculation, few allowing even the calculation of adjacent channel interference.

ILLUSTRATION 1: IMPACT OF THE $SINR^{IMD}$ MODEL ON BER:

In this section, we modify the SINR equation native to WSNet in order to evaluate the impact of the intermodulation distortion. To this end, using MATLAB,

we plot BER curves obtained using the proposed $SINR^{IMD}$ model with that obtained with the $SINR^{e}$. The 2.4 GHz PHY layer of IEEE 802.15.4 adopts the offset quadrature phase-shift keying (O-QPSK) modulation with half-sine pulse shaping (which is equivalent to MSK from an RF spectrum mask perspective). In an additive white Gaussian noise (AWGN) channel, the bit error rate (BER) can be expressed by [168]

$$BER = Q\left(\sqrt{2 \cdot SINR}\right) \tag{5.13}$$

where $Q(x)$ is the Gaussian Q-function

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_{x}^{+\infty} \exp\left(-\frac{u^2}{2}\right) du. \tag{5.14}$$

The simulation scenario is shown in Figure 5.4 where the source node j emits a 32 Bytes/s Constant Bit Rate (CBR) broadcast traffic through channel 1 of IEEE 802.15.4 2.4 GHz radio. Then, two strong interferers (m using channel 2 and n using channel 3) start jamming the network (1). The received powers at the receiver (i) from emitter (j) and interferers (m and n) are respectively -73 dBm and -45 dBm.

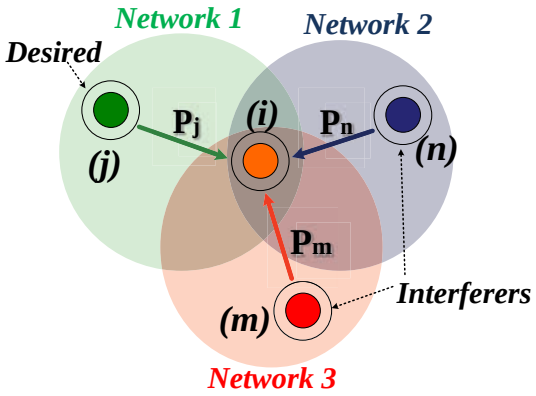


Figure 5.4: Simulation scenario.

The rejection factor $\alpha_{i,k}$ chosen for this simulation is based on the selectivity of the TI CC2520 chip [169], which is given as

$$\alpha_{i,k}[dB] = \begin{cases} 0\,; & i = k \\ -49\,; & 0 \leqslant i-k \leqslant 1 \\ -54\,; & 1 < i-k \leqslant 2 \\ -55\,; & i-k > 2 \end{cases} \tag{5.15}$$

IMD_3 power resulting from the mixing of received signals is for $i = 1$, $m = 2$ and $n = 3$

$$\begin{aligned} P_{IMD_3} = {} & \beta 1_{(1,2)} P_1 P_2^2 + \beta 1_{(2,1)} P_2 P_1^2 \\ & + \beta 1_{(1,3)} P_1 P_3^2 + \beta 1_{(3,1)} P_3 P_1^2 \\ & + \beta 1_{(2,3)} P_2 P_3^2 + \beta 1_{(3,2)} P_3 P_2^2 \end{aligned} \tag{5.16}$$

The IIP_3 of the radio chip CC2520 is -24 dBm. From (5.15), the calculated P_{IMD_3} is -87 dBm. The simulation results are illustrated in Figure 5.5. We observe that the curve obtained with the $SINR^e$ model is very close to that of the SNR, which implies that the interferers in adjacent channels are highly rejected. However, the curve obtained with the new model clearly shows that the intermodulation interference impacts the BER performance and is not negligible.

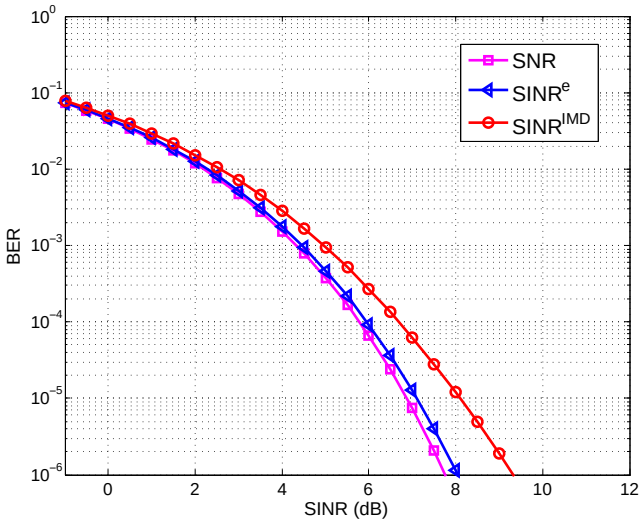


Figure 5.5: Comparison of BER using different SINR models.

ILLUSTRATION 2: IMPACT OF THE $SINR^{IMD}$ MODEL ON PER:

In order to further investigate the impact of intermodulation interference, another set of simulations was run using WSNet to determine the relation between PER and channel intermodulation. In these simulations, the positions of the transmitter and the receiver were fixed, 30 m apart, and the jamming nodes were randomly distributed in a circular area around the receiver. The maximum distance between the jamming nodes and the receiver was 10 m. The simulation scenario is represented in Figure 5.6.

We wish to study the case where the power at the antenna of the receiving node due to interfering nodes exceeds the power of the desired signal. Thus, the transmission power of the desired transmitting node is -10 dBm, while the trans-

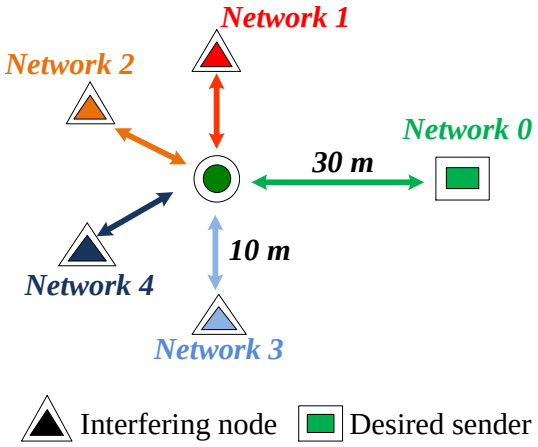


Figure 5.6: Simulation scenario used for analysis of PER under intermodulation.

mission power of the interfering nodes is 0 dBm. The received signal power at a distance d from the transmitter is simply modeled as [170]

$$P_j = P_j(d_0)\left(\frac{d}{d_0}\right)^{-n} \quad ; \quad d > d_0 \tag{5.17}$$

where d_0 is the reference distance ($d_0 = 50$m) and n is the path loss exponent ($n = 2$). In this simulation, the networks use predefined channels and adopt a non-slotted CSMA/CA and IEEE 802.15.4 2.4 GHz compliant radio. The transmission channels employed by each network are described in Table 5.3 where each packet size is 64 Bytes.

Table 5.3: Transmission channels used by each network.

Network number	channel
0	2425 MHz
1	2415 MHz
2	2420 MHz
3	2430 MHz
4	2435 MHz

Simulations are run using either the $SINR^{IMD}$ model or the $SINR^{e}$ model, with and without intermodulation. The results obtained in terms of PER are shown in Figure 5.7, upon varying the node population density per jamming network, in the range 0 through 22. The diagram clearly shows the drastic impact of intermodulation on PER as the density of nodes per adjacent network increases.

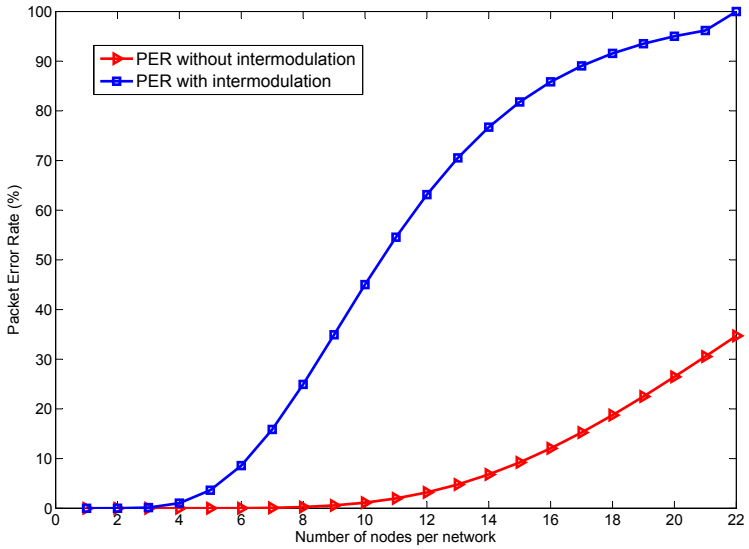


Figure 5.7: Effects of node density on PER with and without intermodulation.

5.3.2 *Phase Noise Interference*

Phase noise is the most important parameter that characterizes the functioning of the local oscillator (LO) and consequently the PLL (phase-locked loop) of the RF transceiver's frequency synthesizer. Indeed, oscillators in transceivers are used to generate a reference signal for frequency translation purposes. Ideally, in the frequency domain, the generated signal would be a Dirac and has energy at only the carrier frequency (f_c). However, in real oscillators, various noise sources lead to phase fluctuations. The generated signal spectrum is affected by a kind of a "skirt" around f_c and can be presented in the time domain as [171]

$$LO(t) = \cos\left[\omega_c t + \Phi_n(t)\right] \tag{5.18}$$

Here ω_c is the oscillator frequency, and $\Phi(t)$ is the random phase fluctuation due to noise in the oscillator and is called *phase noise*. Typically, phase noise is measured in the frequency domain in terms of dBc/Hz at a given offset frequency from the carrier frequency (f_c) and denoted as $L(\Delta f)$. Local oscillator phase noise may introduce phase errors that result in demodulation errors. Another serious problem caused by phase noise that may affect the BER is called reciprocal mixing.

As depicted in Figure 5.8, reciprocal mixing occurs when the phase noise on the receiver LO mixes with a strong interfering signal to produce a noise signal which falls inside the desired signal band of bandwidth (BW). As shown in 5.19 the power of this interference depends on the LO phase noise (L), the adjacent interferer power ($P_{Interferer}$), and the signal bandwidth (BW) [172]:

$$P_{PN}[dB] = P_{Interferer}[dB] + L_{PN}(\Delta f) + 10\log(BW)[dBm] \tag{5.19}$$

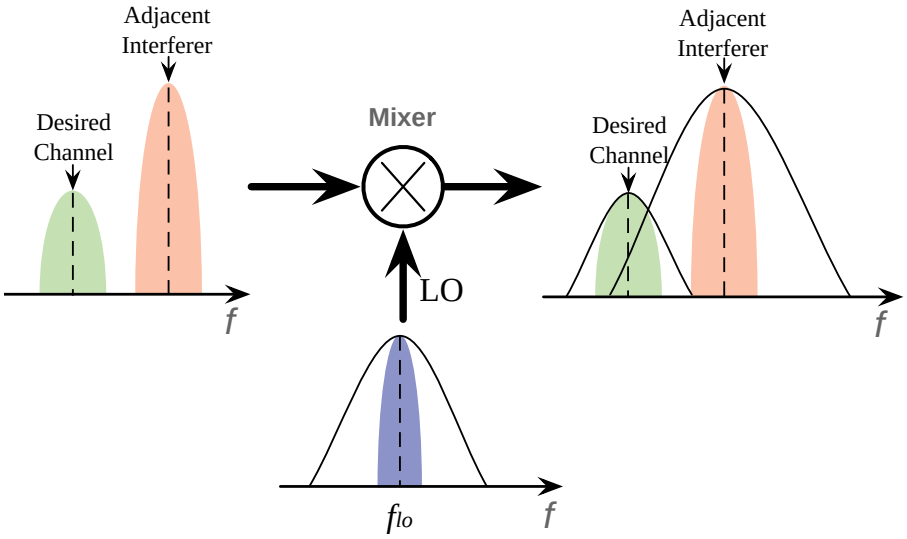


Figure 5.8: Effect of reciprocal mixing due to the local oscillator phase noise.

5.3.3 *Quantization Noise*

An ADC is responsible for the conversion of a signal that is continuous in amplitude and time to a signal that is discrete in amplitude and time. Since the input analog signal has an infinitesimal resolution and the output digital signal has a limited resolution, an error due to the process of analog to digital conversion is introduced. This error is known as quantization noise which depends on the number of bits of the ADC.

The architecture of the reconfigurable receiver shown in Figure 4.10 includes a reconfigurable ADC in terms of power consumption and number of bits, thus, its important to model the impact of its reconfiguration on the global SINR. The quantization noise power P_Q has been found to be [173]

$$P_Q = \frac{Q^2}{12} \tag{5.20}$$

where Q is the quantization step size. An increase of the precision of the ADC by one bit corresponds to a reduction of Q by a factor of 2 and results in a decrease of P_Q by a factor of 4. This quantization noise power is added to the total noise of the receiver N_i.

5.3.4 *Enhanced SINR Model*

Finally, the new general SINR model (or short $SINR^{GEN}$) including the effect of all interference described before is given by

$$SINR_{ij}^{GEN} = \frac{P_j}{N_i + P_{Co/Adj} + P_{IMD_3} + P_{PN}} \tag{5.21}$$

The main challenge is to correctly implement the new SINR model in the simulation framework. The implementation of this model will described in the next section.

5.4 THE ENVADAPT SIMULATION FRAMEWORK

The EnvAdapt simulation framework was designed in order to allow the modelling and simulation of reconfigurable transceivers. One of the fundamental goals of EnvAdapt is enhancing the realism of the PHY layer by providing detailed radio models that are closer in implementation to a real radio system. EnvAdapt is based on the WSNet [167] simulator and takes advantage of its existing features including modularity of the networking layers, detailed channel models, etc. Similarly to the HarvWSNet co-simulation framework presented in chapter 3, the EnvAdapt simulator can be viewed as an enhancement of the original kernel and PHY models of WSNet. Indeed, WSNet does not allow the simulation of several receiver reconfiguration strategies which require that the value of the LQE_{HW} and SINR be calculated during packet reception and not after. While the architecture of WSNet is accurate in terms of PHY layer simulation [161] [162], modeling the reconfigurable transceiver requires a realtime monitoring and adaptation of the transceiver performance. This process was not possible to implement in WSNet because the interference calculation was performed in the simulator kernel and does not give the transceiver the ability to monitor and update the calculation parameters. For this reason, the interference calculation process is implemented directly in PHY layer.

We have implemented in the architecture of EnvAdapt all of the necessary components to enable the study of any transceiver reconfiguration strategy. In addition to the existing WSNet simulation components, each network node possesses its reconfigurable transmitter and/or receiver models, an enhanced SINR calculation block, a hardware LQE and decision block and a reconfiguration manager.

An additional block has been added in the MAC layer in order to support software LQE calculation and decision. Cross-layer function calls are implemented in order to allow the software LQE algorithm to account for any required hardware LQE measurements. In the following sections, we detail the parameters and the core components of the EnvAdapt PHY layer required for the reconfigurable radio simulation.

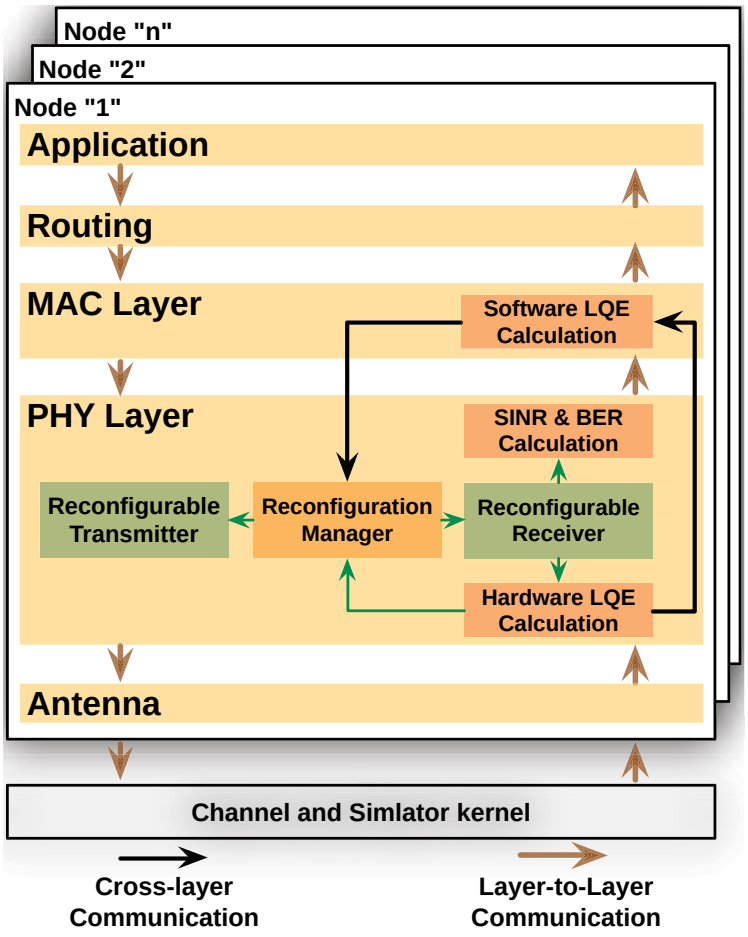


Figure 5.9: EnvAdapt Architecture.

In contrast with existing WSNs simulators, the PHY layer of a sensor node in EnvAdapt can be described with several performance levels. The XML[1] presentation of a reconfigurable radio transceiver is shown in Figure 5.10. Note that the transceiver parameters indicated are based on the reconfigurable transceiver presented in Chapter 4.

5.4.1 *EnvAdapt PHY Transmitter Model*

The PHY transmitter model is composed of four functions namely: *Tx*, *Tx_end*, *Tx_reconf_manager* and *MEDIA_Tx* as shown in Figure 5.11.

In EnvAdapt, when a packet is received by the PHY transmitter layer, the function *"tx"* checks the radio state (idle or active), its activity (transmitting or listening), and the available energy in the battery before transmission. If the energy module (battery or energy harvester) is unable to power the node, the packet is

1 The XML configuration file is used to describe the network and the parameters of each protocol layer of each sensor node.

```
<transceiver name="ReconfigurableRadio">
    <!-- == Upper Layer = -->
    <up name="MAC"/>
        <!-- == Down Layer = -->
        <down name="antenna"/>
        <parameters>
            <!-- == Transceiver Global parameters = -->
            <param key="channel" value="2"/>
            <!-- == Rx Reconfiguration parameters = -->
            <!-- 0 = no Reconfiguration, 1 = Static Reconfiguration, 2 =
                Dynamic Reconfiguration -->
            <param key="Reconfiguration" value="2"/>
            <!-- == LQE reconfiguration Thresholds -->
            <param key="LQEth_High" value="5"/>
            <param key="LQEth_Low" value="0"/>
            <!-- == Rx linearity for each mode-->
            <param key="IIP3_h" value="-26"/>
            <param key="IIP3_m" value="-27"/>
            <param key="IIP3_l" value="-29"/>
            <!-- == Rx Noise Figure for each mode-->
            <param key="NF_h" value="4.6"/>
            <param key="NF_m" value="7"/>
            <param key="NF_l" value="14.7"/>
                <!-- == Rx Phase Noise for each mode-->
                <param key="PN_h" value="-114"/>
                <param key="PN_m" value="-110"/>
                <param key="PN_l" value="-95"/>
            <!-- == Rx current consumption in mA for each mode-->
            <param key="Iconsum_h" value="18.83"/>
            <param key="Iconsum_m" value="8.75"/>
            <param key="Iconsum_l" value="3.75"/>
            <!-- == Tx Reconfiguration parameters-->
            <!-- 0 = no Reconfiguration, 1 = Reconfigurable Tx -->
            <param key="Reconfiguration" value="0"/>
            <!-- == Tx current consumption in mA for each mode-->
            <param key="Iconsum_tx_h" value="20"/>
            <param key="Iconsum_tx_m" value="15"/>
            <param key="Iconsum_tx_l" value="10"/>
            <!-- == Tx transmission power in dBm for each mode-->
            <param key="Tx_power_h" value="5"/>
            <param key="Tx_power_m" value="0"/>
            <param key="Tx_power_l" value="-5"/>
        </parameters>
</transceiver>
```

Figure 5.10: The XML configuration file used in EnvAdapt for a reconfigurable radio.

automatically dropped and a cross-layer notification about energy is created and sent from the energy module directly to the application layer.

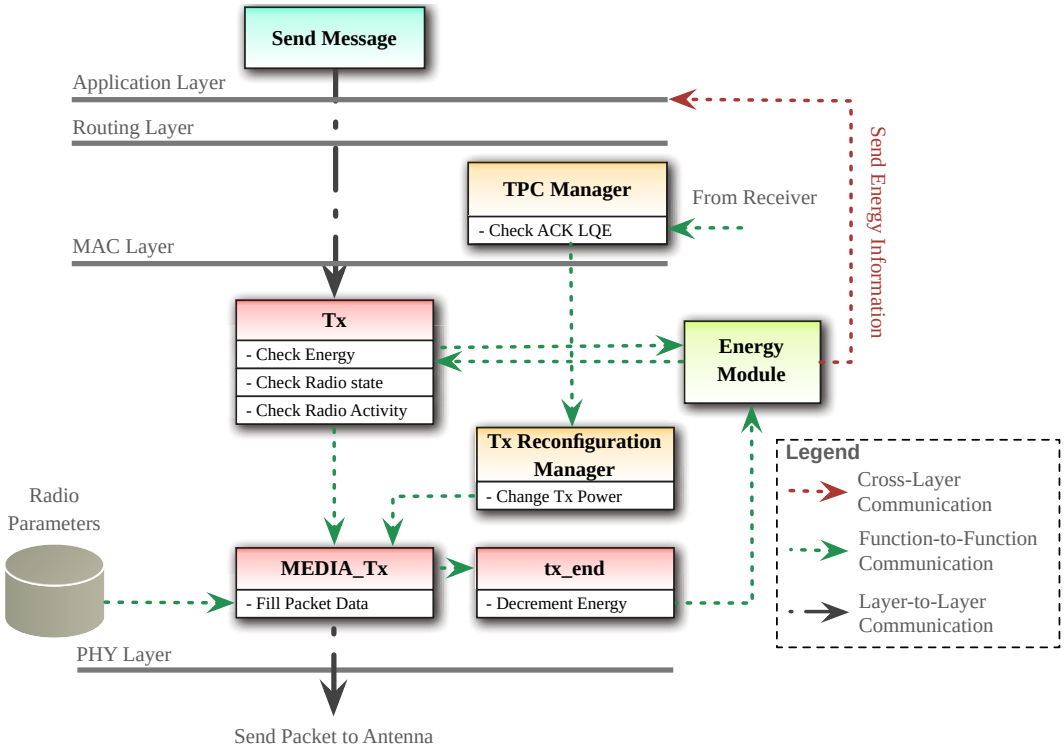


Figure 5.11: Interactions between the reconfigurable transmitter PHY layer and the other EnvAdapt layers.

When the radio is in idle mode, then the packet structure is filled by the transmission parameters at the *"media_tx"* function before being transmitted to the antenna. These parameters include: transmission power, channel frequency, modulation type, and packet duration. The function *tx_end* is responsible for decrementing energy and is called directly when the packet is transmitted by the antenna.

The "*Tx_reconf_manager*" function is called only if the reconfiguration mode is activated and is responsible for transmission power control. It serves as an interface between the MAC layer which determines the optimal transmission power and the PHY layer which updates the radio parameters.

5.4.2 *EnvAdapt PHY Receiver Model*

The software architecture of the PHY layer of EnvAdapt is composed of two main functions namely *"cs"* (carrier sense) and *"rx"*, which are responsible for all receiver operations including interference and SINR calculation, hardware LQE calculation, and receiver parameters reconfiguration. This architecture can be seen as a major enhancement of the WSNet PHY layer architecture which includes the

required functions for receiver reconfiguration. Figure 5.12 gives an overview of the software architecture of the EnvAdapt receiver PHY layer. Note that the reconfiguration mode, either static or dynamic, can be chosen directly in the XML configuration file. In the following, we describe each block and the functions required for the process of receiver reconfiguration.

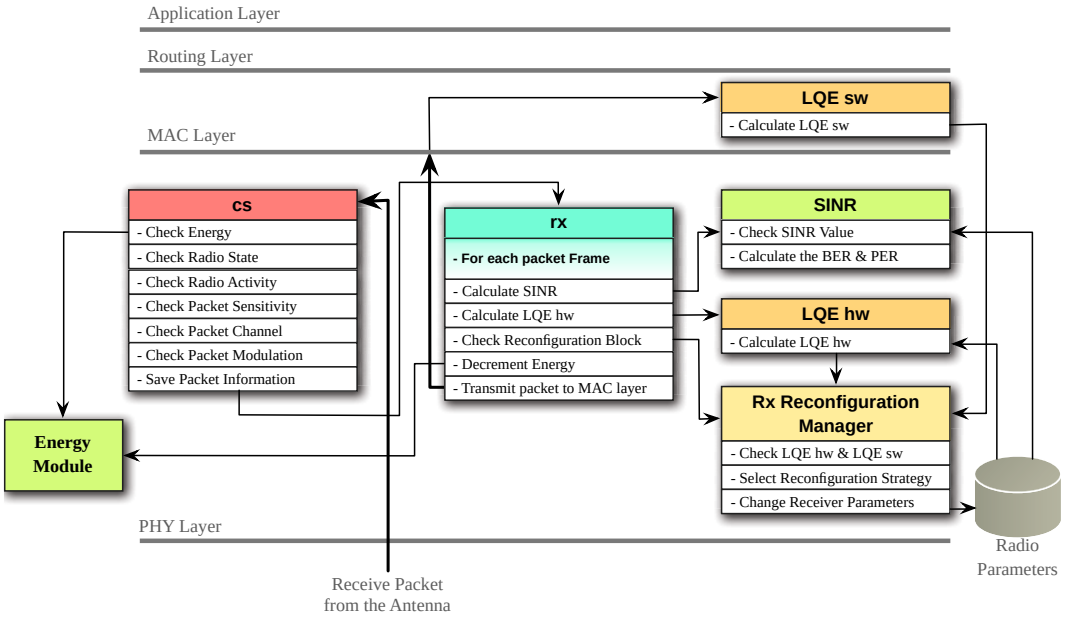


Figure 5.12: Overview of the software architecture for the EnvAdapt receiver PHY layer.

5.4.2.1 *Interference and SINR Calculation Block*

As described in Section 5.2, when modeling interference in a network simulator, different levels of complexity and detail are possible. The original engine of WSNet supports the simulation of an interference model that accounts for adjacent channel effects. This is an important feature since the implementation of the $SINR^{GEN}$ model in a network simulator requires the knowledge, at a given time, of all the RF-band signals concurrently received by a given node. In addition, the IEEE 802.15.4 implementation in WSNet allows the calculation an SINR for each frame composing the packet as shown in Figure 5.13. The desired packet is *Packet 2* while the other packets are interferers.

At the reception of a packet at the antenna, the first function called is *"cs"* which is responsible of checking of energy state and all information about the packet including radio activity, state, sensitivity, etc.

In EnvAdapt, SINR is implemented in the *"rx"* function as shown in Figure 5.12. For each packet arriving at the *"rx"* function, the SINR calculation function has to decide whether to calculate the corresponding SINR or treat the arriving packet

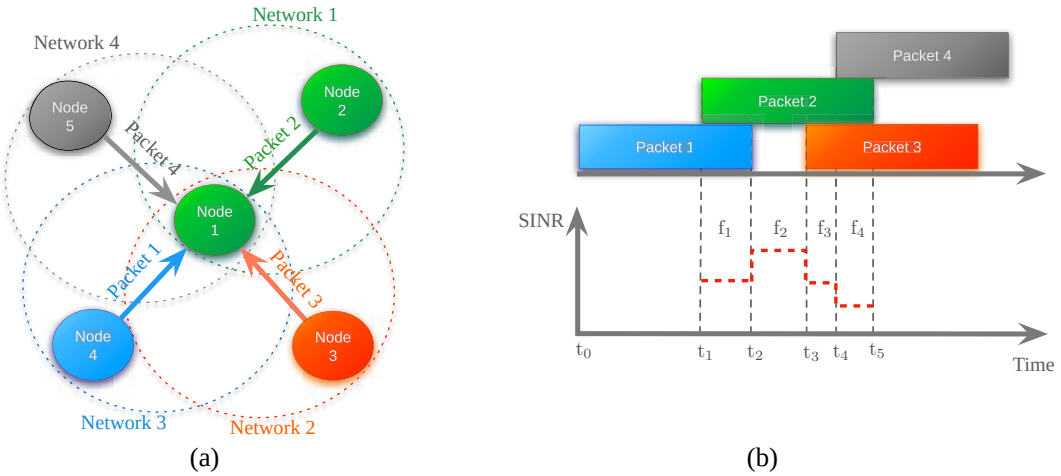


Figure 5.13: (a) Illustration of interference generated at node 1, (b) SINR computation at node 1.

as an interferer. The time moment for the decision making (e.g. denoted as t_0 in Figure 5.13) depends on the packet channel and other receptions currently ongoing on the transceiver.

A packet is considered as an adjacent interferer if its channel is different of the radio channel and consequently will be added to the noise structure. On the other hand, a received packet is treated as a co-channel interferer if its channel is the same as the receiver channel but other radio parameters, such as modulation, differ from the receiver parameters.

The receiver considers a packet as desired if it respects all standard controls such as: correct modulation, packet power level above the receiver's sensitivity, packet channel is the same as the receiver's channel, etc. In this case "packet 2" is the desired packet being evaluated. At the start of receiving "packet 2" there has already been "packet 1" on the channel, which is considered as adjacent interferer.

In WSNet and therefore EnvAdapt, the evaluation of SINR considers all packets that overlap in time with the desired packet. Assuming that each noise source is modeled as additive white gaussian noise (AWGN) and that all noise sources, are uncorrelated, the noise power P_{f_1} of the first frame f_1 is given as

$$P_{f_1} = \alpha_{13} P_1 + PN_1 \tag{5.22}$$

where α_{41} is the rejection factor[1] between channel 1 and channel 4, and PN_1 is the phase noise due to packet 1. Note that when packet 1 disappears, WSNet calculates a new value of SINR corresponding to frame f_2. Then, similarly to frame f_1, the noise due to packet 3 is $P_{f_3} = \alpha_{12} P_3 + PN_3$. Starting from time t_4, packets 3 and 4 both overlap with the desired packet 2. In this case, and in addition to the noise

1 we assume that node 1 and 2, node 3, node 4, and node 5 use the channels 1, 2, 3, and 4 respectively.

calculation due to packet 4, the intermodulation products between the packets 3 and 4 are calculated and then added to the total interference noise of the receiver "1".

Once the SINR is calculated for each frame affecting the desired packet, the bit error rate (BER) is calculated from the SINR of each frame according to the modulation used (e.g. using 5.13). Assuming that the PHY layer cannot correct reception errors and assuming also an independent bit error distribution, in EnvAdapt, the packet error rate (PER) is calculated from the BER of each packet frame as

$$PER = 1 - \prod_{i} (1 - BER_i)^n \tag{5.23}$$

where n is the packet size in bit. After the calculation of the PER, a random number uniformly distributed between 0 and 1 is generated using the function *"get_random_double()"*. If the random number is less than the calculated PER, then the packet is rejected.

5.4.2.2 *Hardware LQE Calculation Block*

As stated in chapter 4, a hardware LQE calculation block ideally provides an accurate estimation of the instantaneous SINR. This link estimation (in combination or not with a software LQE) can be used to select the receiver performance mode for the incoming frame or the next packet. In case of intra-packet dynamic reconfiguration, the link quality estimation should be fast enough in order to support any type of reconfiguration strategy ranging from one-shot or periodic to continuous. When a packet frame is received by the "rx" function, the LQE_{HW} is simultaneously calculated by the LQE hardware block (Figure 5.12) before making a decision on packet acceptance or rejection. This implementation provides considerable realism in the modelling of the behavior of a reconfigurable receiver.

The calculated LQE_{HW} of each packet frame is sent to the reconfiguration manager, which selects using its algorithm, the best performance mode of the receiver. The new performance mode parameters are then saved to the "Radio Parameters" database which holds also the processing delay associated to the calculation of LQE_{HW}.

To meet the requirements of high simulation speed, the elements of the LQE_{HW} metric are calculated in parallel with those of the SINR. The total power P_j^* and the out of channel interference are calculated only if the incoming packet is considered as signal and not as noise. The noise floor of the receiver is constructed by monitoring and averaging the in-channel signal strength after packet reception.

5.4.2.3 *Reconfiguration Manager Block*

The receiver reconfiguration manager is capable to make the receiver reconfigurable or not according to the function mode indicated in the XML file. For example, "0" indicates no reconfigurable, "1" indicates static reconfiguration, and "2" indicates dynamic reconfiguration. The reconfiguration manager takes as input the LQE_{HW} and LQE_{SW} calculated by the software and/or hardware LQE calculators as shown in Figure 5.12.

The receiver appropriate power mode is selected according to a predefined thresholds corresponding to the minimum acceptable SINR in the case of dynamic reconfiguration, or to the number of dropped packets in a time window, in the case of static reconfiguration. The algorithms used in each reconfiguration mode are described in the next section.

5.5 ENVADAPT TRANSCEIVER RECONFIGURATION STRATEGIES

As stated in Section 4.7.2, we distinguish two main families of reconfiguration strategies: dynamic reconfiguration and static reconfiguration. In the dynamic reconfiguration family, we assume that the receiver is able to reconfigure itself during packet reception, whether continuously, periodically, a single time, in case of event detection, etc. Static reconfiguration assumes that the receiver can change its settings only between two packet receptions.

5.5.1 *Receiver Reconfiguration*

The architecture of EnvAdapt was designed to support any transceiver reconfiguration mode. The current version of EnvAdapt supports two reconfiguration strategies namely continuously dynamic and static. In the following, we show the implementation of these reconfiguration strategies in EnvAdapt.

5.5.1.1 *Receiver Dynamic Reconfiguration*

As stated in Chapter 4, the dynamic reconfiguration is processed during the reception of the incoming packet. To allow a receiver to adapt to the real-time channel conditions, in this strategy, LQE_{HW} calculation and performance mode selection are performed continuously during packet reception as illustrated in Figure 5.14.

In EnvAdapt, the each packet received is subdivided into frames. It is possible to select directly (i.e. XML file) the size of each frame that can vary from one bit to several bits. In order to insure that the first frame is correctly received, the receiver is set to high performance mode at the beginning of the packet reception as shown

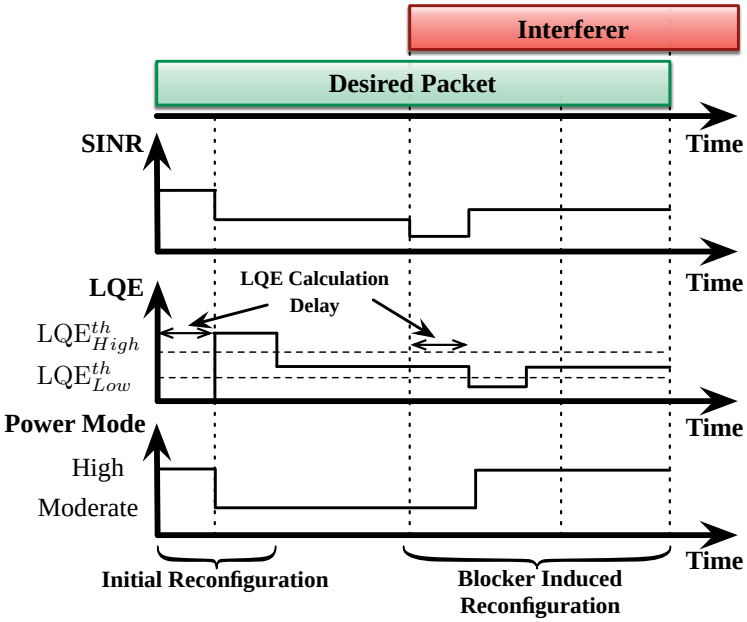


Figure 5.14: Variation of SINR, LQE_{HW} and power consumption for a continuous dynamically reconfigurable receiver

in Figure 5.15. At the end of the first frame acquisition, the LQE_{HW} is calculated and consequently the optimum power mode is selected for the next frame reception. Note that a processing delay to the LQE_{HW} calculation is assigned in order to account for the processing speed of the LQE_{HW} block.

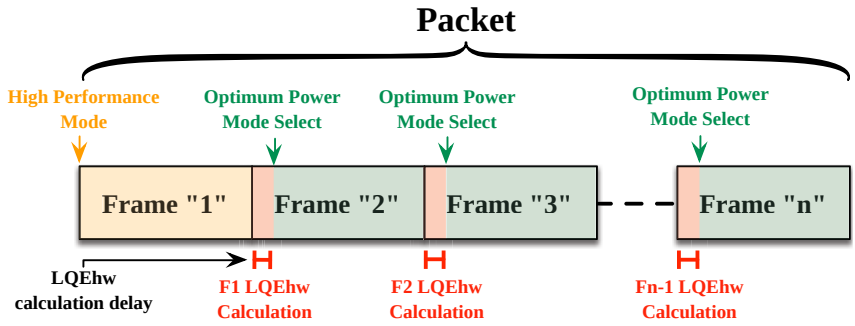


Figure 5.15: Illustration of dynamic reconfiguration during packet reception with LQE_{HW} calculation delay.

The algorithm responsible of dynamic reconfiguration management and the functions called are illustrated in Figure 5.16.

5.5.1.2 *Receiver Static Reconfiguration*

Unlike the aforementioned reconfiguration strategy, static reconfiguration adjust the performance of the receiver only at the end of a packet reception or at the end of the observation time window ω as shown in Figure 5.17. The idea behind this implementation aims to reduce the processing overhead due to LQE_{HW} calculation

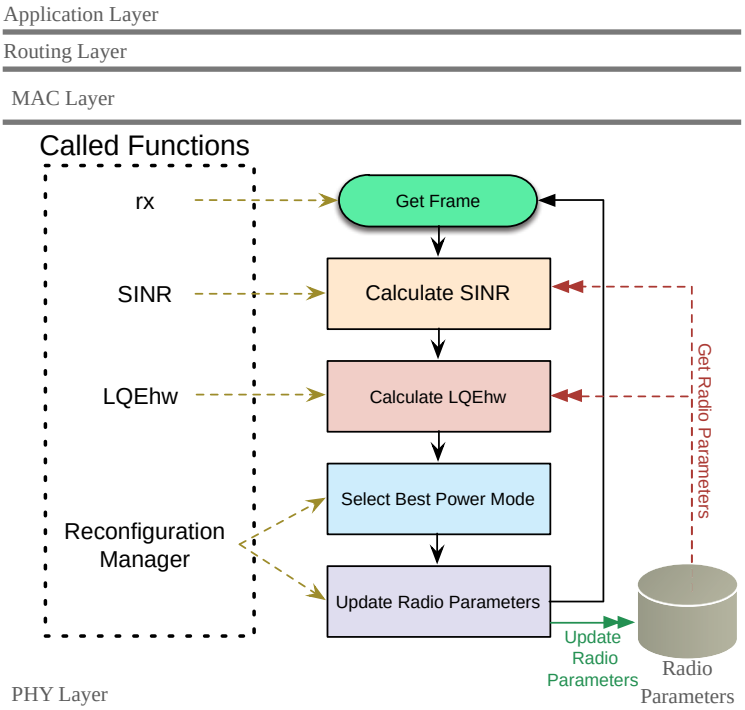


Figure 5.16: Dynamic reconfiguration algorithm.

and receiver performance adjustment during packet reception. At the startup, the first training packets (at least 2 packets) are received in high performance mode in order to accommodate the learning needs of the reconfiguration manager. Based on these packets, the LQE_{SW} is calculated and the optimum power mode is correspondingly selected.

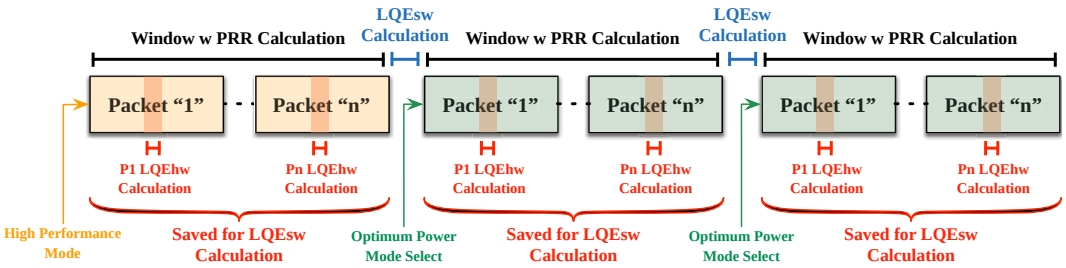


Figure 5.17: Receiver static reconfiguration during packets reception.

The algorithm responsible of static reconfiguration is illustrated in Figure 5.17. For each packet received in EnvAdapt, an SINR and LQE_{HW} is calculated. If the LQE_{HW} of a packet is under the predefined threshold, the reconfiguration manager switches the receiver performance directly to high power mode before the end of the observation window ω. This case ensures that the next packets are correctly received. In the case where the LQE_{HW} calculated is larger than the threshold, this information is saved in structure and sent using a cross-layer call to the reconfiguration manager which updates the performance as function of the LQE_{SW} calculated in the MAC layer.

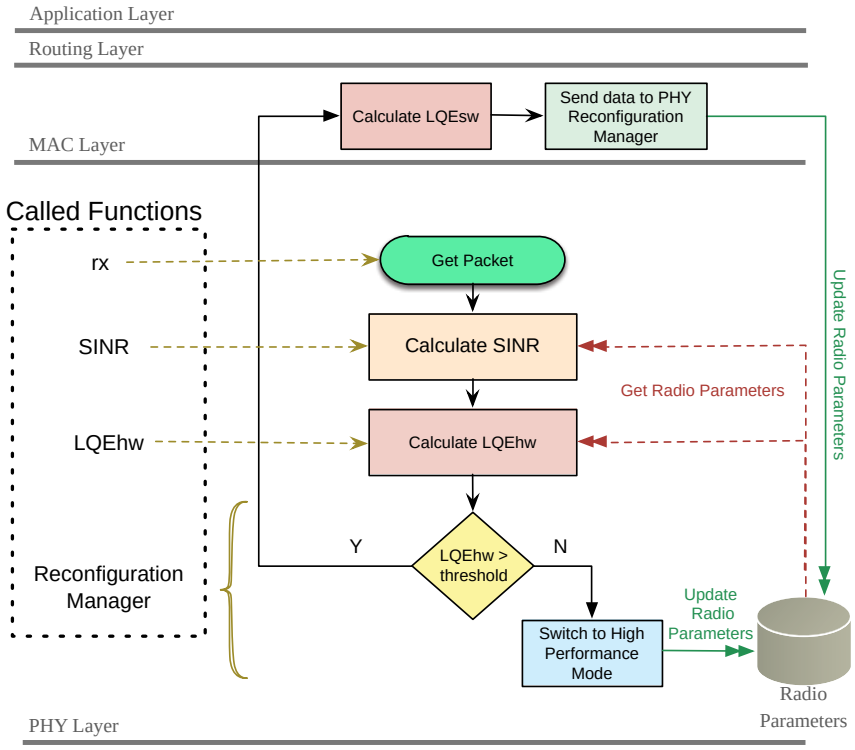


Figure 5.18: Receiver static reconfiguration algorithm.

5.5.2 *Transmitter Reconfiguration*

As described in Chapter 4, the transmitter reconfiguration is based on the measure of RSSI on the acknowledgment packet. For each value of RSSI a corresponding configuration mode is selected for the next packet. The current implementation of EnvAdapt supports only the control of transmitted power, a future version may include also the reconfiguration of other blocks such as the VCO. This will heavily depends on the reconfiguration metric used.

The algorithm responsible of this process is shown in Figure 5.19. Since the PHY layer has not the ability to accept a packet even if it respects all receiving controls, the strategy of transmission power control is implemented only at the MAC which is the only layer that has the ability to distinguish if the acknowledgment packet is intended to the actual receiving node or not.

Assuming a single-hop network, the transmission power control in EnvAdapt works as follows: At the beginning of the simulation, the transmission power of all nodes is set to an initial value. Each node stores a table containing all the transmission power levels of the transmitter used. When an acknowledgment packet is received, the "rx" function calculates its RSSI and sends it to the MAC layer which selects the transmission power as function of the RSSI table. This information is

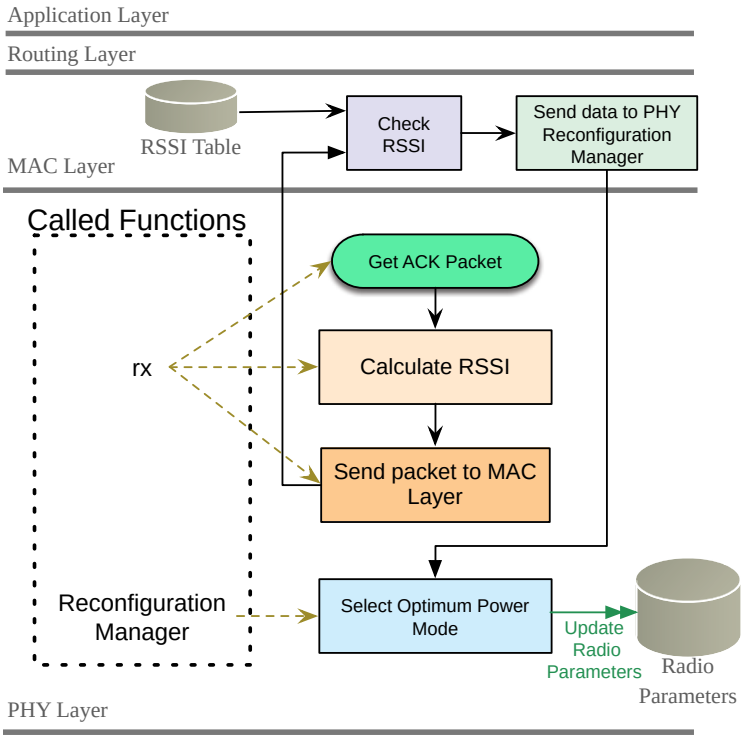


Figure 5.19: Transmitter reconfiguration algorithm.

then sent to the PHY reconfiguration manager which updates the transmission power.

5.6 EXPLORATION OF RECEIVER RECONFIGURATION USING ENVADAPT

In this section, we illustrate the use of EnvAdapt to compare the static versus the continuously dynamic reconfiguration strategies presented in Section 5.5.1.1 and Section 5.5.1.2, respectively, for the scenario represented in Figure 5.20 and using the reconfigurable receiver model defined in 4.5.2.5. Recall from Table 4.7 that this receiver possesses three power/performance modes and thus its power consumption can be reduced from 22.6 mW to 4.5 mW when there is no interference and the desired signal power is high. For the sake of this illustration, we assume that the receiver is also able to calculate the hardware LQE metric defined in 4.17. The power consumption cost associated with the calculation of this metric is assumed to be 1mW and the LQE_{HW} calculation delay is 10 ns.

5.6.1 *Simulation Scenario*

In this scenario, the source node j emits a constant bit rate broadcast traffic using channel 1 of its IEEE 802.15.4 2.4 GHz radio (at a 250 kbit/s raw bit rate). Then,

two strong interferers namely, m (using channel 2) and n (using channel 3) start transmitting which creates interference for network (1).

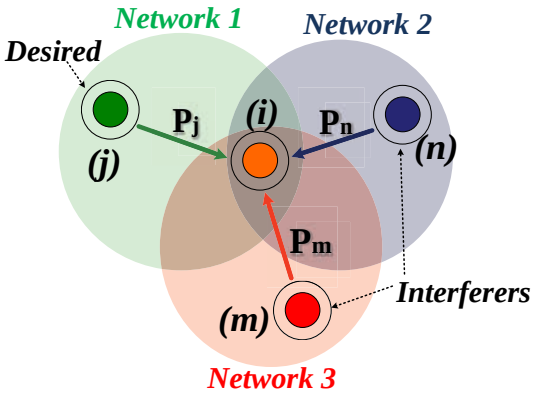


Figure 5.20: Simulation scenario.

We consider that the power of the interfering nodes is such that their coverage regions overlap with network 1. Using a channel model with a pathloss exponent of 2, the nodes are placed such that the received power at receiver (i) from emitter (j) and interferers (m and n) is respectively -85 dBm and -55 dBm. We assume that receiver (i) rejects the first and second adjacent interferers by 39 dB and 44 dB, respectively. Finally, the power consumption of the receiver (i) in idle mode is set to 400 μW.

5.6.2 *Dynamically Reconfigurable Receiver Simulation*

In order to illustrate the reconfigurable receiver behavior, we consider that node j transmits a 64 byte packet to receiver i every 2.5 ms. Interferers n and m broadcast a packet of 32 bytes randomly in time. The continuous dynamic reconfiguration algorithm employed is the one presented in 4.20. In this simulation, we assume that $\text{SINR}_{\text{out}}^{min} = 3$ dB, $\text{LQE}_{\text{Low}}^{\text{th}} = 3$ dB and $\text{LQE}_{\text{High}}^{\text{th}} = 5$ dB. Figure 5.21 shows the hardware LQE and the performance mode of receiver i as a function of time.

In the time interval ranging from 0 ms to 5 ms, we observe that the absence of interference allows the receiver to reach and remain in the low performance mode for a major part of packet reception. The appearance of interferers after 5 ms causes a drop in the LQE and consequently a performance change. As shown in Figure 5.21, the receiver dynamically adapts to changes in channel conditions and continuously chooses the appropriate performance mode. The amount of power consumption saved, in this scenario, will be studied in the next sections.

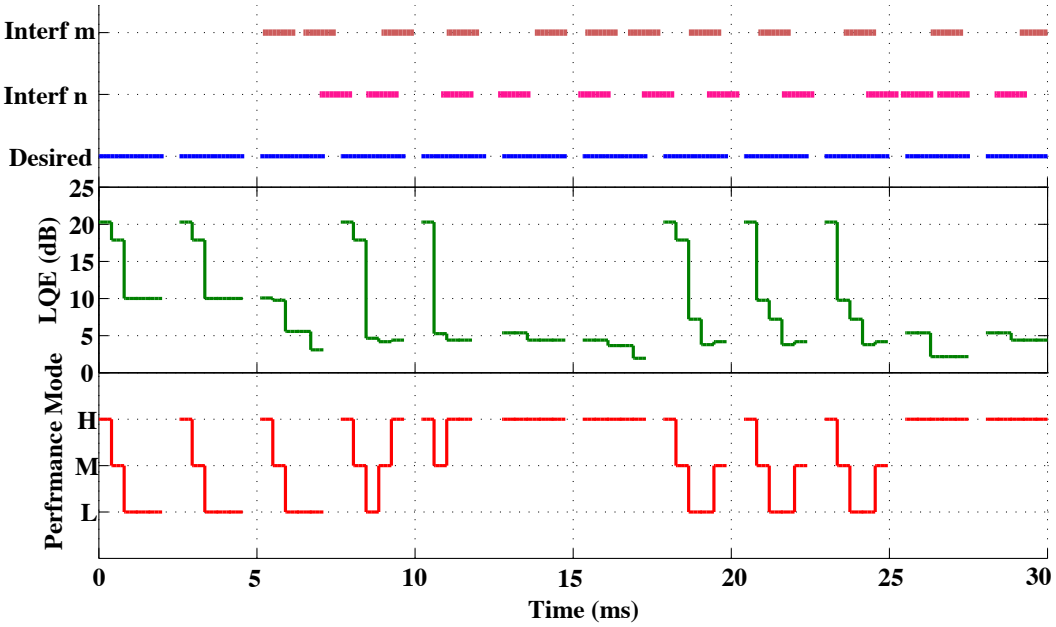


Figure 5.21: Simulation time-trace assuming dynamic reconfiguration.

5.6.3 *Statically Reconfigurable Receiver Simulation*

We reproduce the scenario described in Figure 5.4 but now using the hybrid LQE and static reconfiguration algorithm described in Figure 4.19. In this simulation, node j transmits a 100 byte packet to receiver i every 10 ms, while interferers n and m send a 100 byte packet randomly in time.

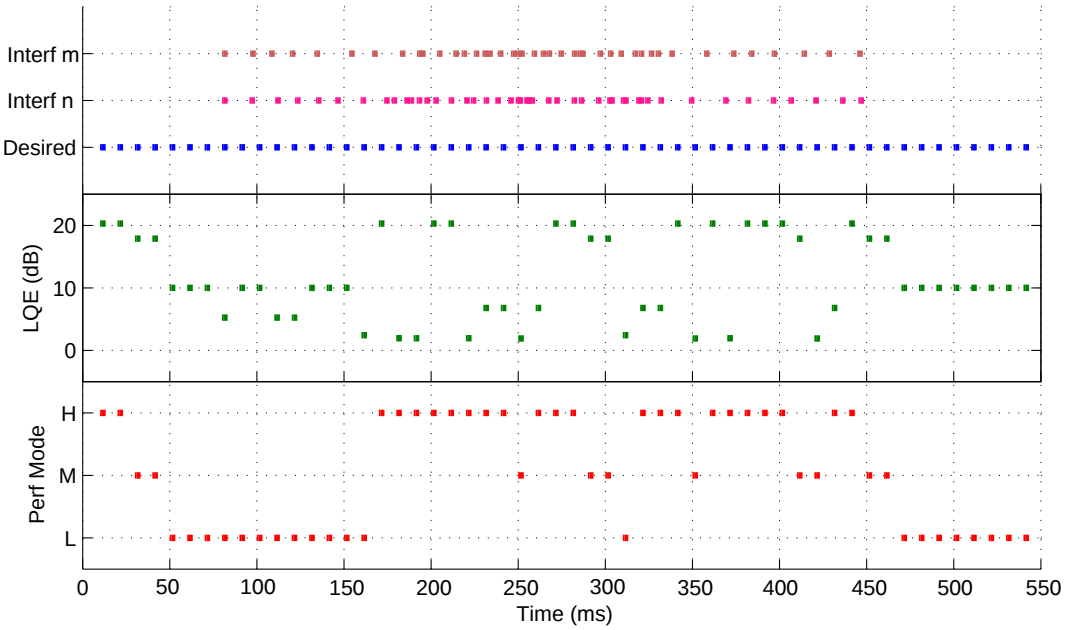


Figure 5.22: Simulation time-trace assuming static reconfiguration.

Here, the chosen $LQE_{th} = 5$ dB and the power cost associated with the hardware LQE measurement is 1 mW during 32 μs. We neglect the power cost associ-

ated with the software LQE measurement. Figure 5.22 shows the hardware LQE and receiver performance mode in time. With a reconfiguration window length of 20 ms, the number of packets per window is 2. In Figure 5.22, we observe as specified that when the LQE_{HW} of the two packets is greater than the LQE^{th}, the receiver switches to a lower performance mode. At 160 ms, the more frequent appearance of interferers induces a drop in the LQE and consequently a general increase of performance. Of course, when static reconfiguration is employed, a drop in the LQE can be detected only at the end of packet reception which increases the probability of packet loss.

5.6.4 *Impact of Dynamic and Static Reconfiguration on PRR and Power consumption*

In order to evaluate the impact of the receiver reconfiguration strategies, we consider again the scenario described in Figure 5.4. The node j transmits 100 Bytes packet every 10 ms while interferers n and m send a 100 Bytes packet randomly in time with variable density. For each reconfiguration strategy, a number of simulations each of 100 s duration are performed. The time density of interfering packets is constant per simulation. The energy source used is based on the battery model of HarvWSNet described in Section 3.5.3.

Figure 5.23 shows the PRR (a) for the desired packets received by node i and the energy consumed (b) by this node as a function of the time density of the interfering packets. Note that the power consumption overhead due to the LQE_{HW} processing is included in the static and dynamic reconfiguration.

In Figure 5.23-(a), we observe that in all channel conditions the receiver dynamic and static reconfiguration give the same PRR as a non-reconfigurable receiver in high performance mode. As expected, Figure 5.23-(b) shows that reconfiguration allows the receiver to consume less energy when the channel conditions are favorable. For the densities of 5, 10, 15 and 20 interfering packets/100 ms, we observe that the energy consumption in the static reconfiguration is higher than the dynamic one due to the frequent appearance of interferers, thereby forcing the receiver to stay most time in high performance mode. However, for the densities 0 and 3, dynamic reconfiguration consumes more energy than the static one due to the overhead of the LQE_{HW} calculation during the entire packet reception.

These preliminary results show the need of correctly modelling the power cost associated with the LQE measurements as well as that associated with packet acknowledgment and retransmission.

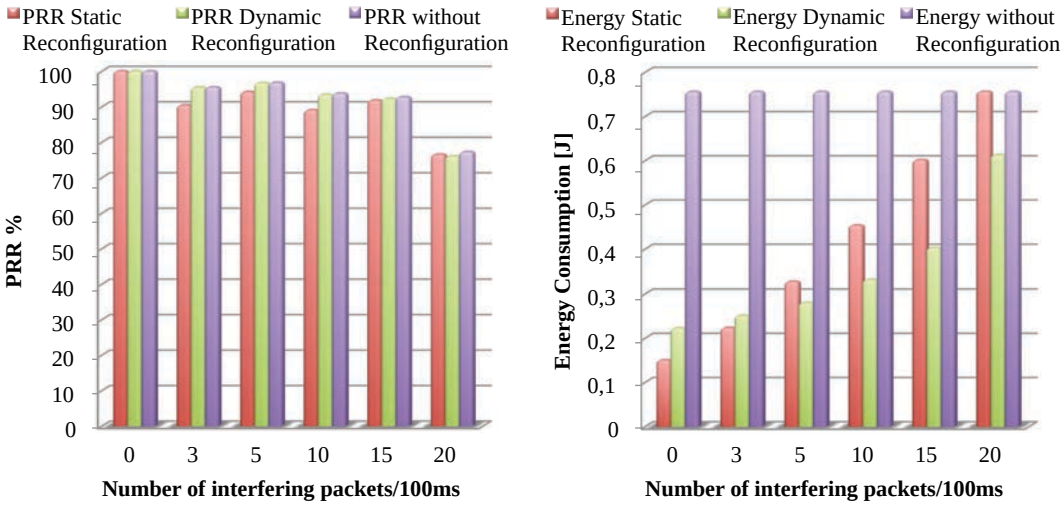


Figure 5.23: Impact of receiver dynamic and static reconfiguration on (a) PRR and (b) energy consumption.

5.7 CONCLUSION

In this chapter, we presented EnvAdapt, the first WSN simulator that fully supports all types of reconfigurable transceivers (RX and TX) models, all types of reconfiguration strategies, as well as all types of link quality estimators (software, hardware, hybrid). We have also implemented in this simulator reconfigurable receiver model that adapts its performance and power consumption to the channel conditions using a link quality estimator and two reconfiguration strategies (i.e. static and dynamic). The proposed approach allows the receiver to operate at minimum power consumption without affecting the required BER and PER. Simulation results for each reconfiguration strategy and the corresponding algorithms were also presented. Finally, an illustrative WSN deployment implemented via EnvAdapt compares statically and dynamically reconfigurable receiver with a non-reconfigurable one and shows the benefits of reconfiguration on energy consumption.

6

GENERAL CONCLUSION AND PERSPECTIVES

In the last decade, the apparition of wireless sensor networks (WSNs) has considerably changed the way to monitor and control many environmental and physical phenomenons. Multiple successful deployments of the tiny sensor nodes, forming the WSN, have already demonstrated their application in many domains including environmental and structural health monitoring, home automation and industrial process surveillance. In order to ensure energy sustainability while being non-intrusive, WSNs usually rely on limited-capacity batteries. However, battery-operated sensor nodes running at full duty cycle may last only several days. Therefore, most existing sensor nodes are set to operate at a very low duty cycle in order to prolong the lifetime of the WSN. On the Medium Access Control (MAC) and routing layers, a plethora of energy efficient protocols have been proposed to minimize the energy dissipation of the radio transceiver, which is typically the most energy-hungry component in a sensor node. More recently, the appealing promises of energy-harvesting techniques has led researchers to investigate the integration of such a technology in WSNs. Each sensor node could be equipped with an energy-harvesting system that converts the environmental energy into electrical energy. While energy harvesting appears to be a very promising technology for the extension of WSN lifetime, it still remains unpredictable and hard to prototype by deploying complete energy harvesting sensor nodes in real environment. Therefore, new evaluation tools must be used in order to validate the design choices and provide a reliable and fast evaluation of energy-harvesting WSNs at a low cost. In this thesis, we addressed the problems of energy-harvesting WSNs evaluation and transceiver over-consumption using computer simulations. This approach is usually preferred over hardware prototyping to limit costs of large scale deployments and also to enable a deep analysis of the internal states of each device or to evaluate the design choices of nonexistent hardware and/or software. To this end, we have proposed in Chapter 3, HarvWSNet, a unified co-simulation framework that fills the gap between network simulations and evaluation of energy-harvesting modules. HarvWSNet combines the strengths of WSNet and Matlab to evaluate energy-harvesting WSNs. WSNet is network simulator which was used to evaluate discrete-event network protocols, while Matlab was used for the simulation of continuous-time energy harvesting systems. These tools are run interactively and synchronized using TCP sockets. We exemplified the use of HarvWSNet in two case studies. The first one investigated the lifetime of solar energy harvesting WSNs while the second one was used to evaluate the feasibility

of a wind-harvesting WSN for structural health monitoring of a subway tunnel. The accurate energy models implemented in HarvWSNet were able to estimate precisely the energy harvested and the energy available in the storage element as function of the environmental data and energy consumed by the sensor node. The use of this framework facilitates research around power management strategies and power aware algorithms and hardware for the emerging class of perpetual energy harvesting wireless sensor applications.

Even if scavenging energy directly from the environment of sensor nodes appears to be an ideal solution that possibly solve the energy sustainability problem, some applications do not have the luxury of having a continuous ambient energy in their environment. Globally, energy harvesting systems still face the challenge of intermittent energy sources that can be hard to harvest or provide a relatively low energy that may be insufficient to power the sensor node. Thus, energy-harvesting systems can be practical only if the powered system operates with relatively low power consumption. Therefore, reducing the power consumption of the radio transceiver was mandatory. In Chapter 4, we have proposed a reconfigurable transceiver model that adapts its performance and power consumption to the surrounding channel conditions. The performance modes are selected according to a link quality estimation that provides a good approximation of the signal-to-noise-plus-interference ratio (SINR). Using radio transceiver reconfiguration strategies, a considerable power was saved when the channel conditions are favorable. In order to evaluate the interaction between the radio transceiver and network protocols in a WSN deployment, we have implemented the radio transceiver models and the corresponding reconfiguration strategies in the EnvAdapt WSN simulator presented in Chapter 5. EnvAdapt is the first WSN simulator that supports the evaluation of several transceiver reconfiguration strategies and radio link quality estimators. For the receiver part, we have implemented two reconfiguration strategies namely, static and dynamic. In order to ease the adoption of our contributions, we have simulated using EnvAdapt a WSN deployment scenario. The results obtained clearly show the benefits of static and dynamic reconfiguration on energy consumption of the WSN.

The approaches and platforms, devoted to the study of energy harvesting and energy-aware radio transceivers for WSNs presented in this manuscript, have shown that the concept of perpetual sensor nodes is possible. Nevertheless, several future research directions must be considered.

- **More realistic energy harvesting models:** The results obtained from HarvWSNet simulations are based on the models of real devices. However, there is always a gap between simulations and real world deployments. We envision

to calibrate the HarvWSNet framework with a real energy-harvesting sensor node such as PowWow.

- **Calibration of EnvAdapt with circuit level simulations:** The overconsumption reduction due to the radio transceiver has been demonstrated using RF system level models. Nevertheless, to evaluate accurately the efficiency of the power saving mechanisms, circuit level simulations should be done in order to calibrate the performance of the transceiver defined in EnvAdapt.

- **Hardware reconfiguration manager:** The proposed reconfiguration manager is based on algorithms which are responsible of adapting the radio transceiver performance to the channel conditions as function of the link quality measured. As this bloc is central to an effective reconfiguration, it would be interesting to study how the algorithms proposed in this thesis could be transformed to a circuit bloc that may be implemented in the radio transceiver digital base-band.

- **Global reconfiguration management protocol:** The reconfiguration strategies presented in Chapter 4 and Chapter 5, allow the radio transceiver to adapt itself to the channel conditions. A future work consists in developing a global reconfiguration management protocol able to simultaneously adapt the transmitter and the receiver performance of each sensor node to the channel conditions and also to the energy supply of adjacent sensor nodes. For example, a sensor node that does not have sufficient energy may switch to low performance mode and ask the adjacent nodes to adapt their performance correspondingly.

LIST OF PUBLICATIONS

- **Impact of RF front-end nonlinearity on WSN communications.** Amine DIDIOUI, Carolynn BERNIER, Dominique MORCHE, and Olivier SENTIEYS. In *IEEE ISWCS '12*: *International Symposium on Wireless Communication Systems*, pages 875-879.
- **HarvWSNet: A co-simulation framework for energy harvesting wireless sensor networks.** Amine DIDIOUI, Carolynn BERNIER, Dominique MORCHE, and Olivier SENTIEYS. In *IEEE ICNC '13*: *International Conference on Computing, Networking and Communications*, pages 808-812.
- **A Systematic Design Approach for Reconfigurable Transceivers.** Amine DIDIOUI, Carolynn BERNIER, and Olivier SENTIEYS. In *Colloque GDR SoC SiP '13*.
- **Power reconfigurable receiver model for energy-aware applications.** Amine DIDIOUI, Carolynn BERNIER, Dominique MORCHE, and Olivier SENTIEYS. In *IEEE MWSCAS '13*: *International Midwest Symposium on Circuits and Systems*, pages 800-803.
- **EnvAdapt: An Energy-Aware Simulation Framework for Power-Scalable Transceivers for Wireless Sensor Networks.** Amine DIDIOUI, Carolynn BERNIER, Le Quang Vinh TRAN, and Olivier SENTIEYS. In *IEEE EW '14*: *European Wireless Conference*, pages 1-6.
- **Prototyping an Energy Harvesting Wireless Sensor Network Application Using HarvWSNet.** Florian BROEKAERT, Amine DIDIOUI, Carolynn BERNIER, and Olivier SENTIEYS. In *ARCS '13*: *International Conference on Architecture of Computing Systems*, pages 1-6.
- **A Co-Simulation Approach to Prototyping Complex Energy Harvesting WSN Applications.** Le Quang Vinh TRAN, Amine DIDIOUI, Gregory VAUMOURIN, Carolynn BERNIER, Florian BROEKAERT, and Agnes FRITSCH. In *IEEE JSAC '14*: *Journal on Selected Areas in Communications*, (under acceptance).

BIBLIOGRAPHY

[1] I. Strategy and P. Unit, "The internet of things," tech. rep., 2005.

[2] B. R. Chalamala, "Portable electronics and the widening energy gap," *Proceedings of the IEEE*, vol. 95, no. 11, pp. 2106–2107, 2007.

[3] ETSI, *Technologies Clusters*. http://www.etsi.org/technologies-clusters/technologies/aeronautical, 2012.

[4] Y. K. Tan, *Energy harvesting autonomous sensor systems: design, analysis, and practical implementation*. CRC Press, 2013.

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

[6] K. Lin, J. Yu, J. Hsu, S. Zahedi, D. Lee, J. Friedman, A. Kansal, V. Raghunathan, and M. Srivastava, "Heliomote: Enabling long-lived sensor networks through solar energy harvesting," in *Proceedings of the 3rd International Conference on Embedded Networked Sensor Systems*, SenSys '05, pp. 309–309, 2005.

[7] F. Simjee and P. Chou, "Everlast: Long-life, supercapacitor-operated wireless sensor node," in *Proceedings of the International Symposium on Low Power Electronics and Design*, pp. 197–202, 2006.

[8] C. Park and P. Chou, "Ambimax: Autonomous energy harvesting platform for multi-supply wireless sensor nodes," in *Annual IEEE Communications Society on Sensor and Ad Hoc Communications and Networks*, vol. 1, pp. 168–177, 2006.

[9] PowWow Sensor Node, http://powwow.gforge.inria.fr/.

[10] S. Priya, C.-T. Chen, D. Fye, and J. Zahnd, "Piezoelectric windmill: A novel solution to remote sensing," *Japanese Journal of Applied Physics*, vol. 44, no. 3, pp. 104–107, 2005.

[11] S. Beeby and N. White, *Energy harvesting for autonomous systems*. Artech House, 2010.

[12] J. Paradiso and T. Starner, "Energy scavenging for mobile and wireless electronics," *IEEE Pervasive Computing*, vol. 4, no. 1, pp. 18 – 27, 2005.

[13] N. Shenck and J. Paradiso, "Energy scavenging with shoe-mounted piezoelectrics," *Micro, IEEE*, vol. 21, pp. 30–42, May 2001.

[14] A. Sample and J. R. Smith, "Experimental results with two wireless power transfer systems," in *Radio and Wireless Symposium*, pp. 16–18, 2009.

[15] J. F. Christmann, E. Beigne, C. Condemine, and J. Willemin, "An innovative and efficient energy harvesting platform architecture for autonomous microsystems," in *IEEE International NEWCAS Conference*, pp. 173–176, 2010.

[16] A. Weddell, G. V. Merrett, T. Kazmierski, and B. Al-Hashimi, "Accurate supercapacitor modeling for energy harvesting wireless sensor nodes," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 58, pp. 911–915, 2011.

[17] Texas Instruments, http://www.ti.com/product/tps61221, *TPS61221*.

[18] A. Bachir, M. Dohler, T. Watteyne, and K. Leung, "MAC essentials for wireless sensor networks," *IEEE Communications Surveys & Tutorials*, vol. 12, no. 2, pp. 222–248, 2010.

[19] M. Raja, X. Chen, Y. D. Lei, Z. Bin, B. C. Yeung, and Y. Xiaojun, "A 18 mW tx, 22 mW rx transceiver for 2.45 GHz IEEE 802.15.4 Wpan in 0.18-μm CMOS," in *IEEE Asian Solid State Circuits Conference*, pp. 1–4, 2010.

[20] A. Do, C. C. Boon, A. Do, K.-S. Yeo, and A. Cabuk, "An energy-aware cmos receiver front end for low-power 2.4-ghz applications," *IEEE Transactions on Circuits and Systems*, vol. 57, no. 10, pp. 2675–2684, 2010.

[21] E. Ben Hamida, G. Chelius, and J. M. Gorce, "Impact of the physical layer modeling on the accuracy and scalability of wireless network simulation," *Simulation*, vol. 85, no. 9, pp. 574–588, 2009.

[22] J. Anderson and R. L., *The Internet of Things Will Thrive by 2025*. http://www.pewinternet.org/2014/05/14/internet-of-things/, 2014.

[23] S. Khan, A.-S. K. Pathan, and N. A. Alrajeh, *Wireless Sensor Networks: Current Status and Future Trends*. CRC Press, 2012.

[24] Texas Instruments, http://www.ti.com/litv/pdf/slaa378d, *Wireless Sensor Monitor Using the eZ430-RF2500*.

[25] J. Hill, R. Szewczyk, A. Woo, S. Hollar, D. Culler, and K. Pister, "System architecture directions for networked sensors," in *ACM SIGOPS operating systems review*, vol. 34, pp. 93–104, 2000.

[26] C.-Y. Chong and S. Kumar, "Sensor networks: evolution, opportunities, and challenges," *Proceedings of the IEEE*, vol. 91, no. 8, pp. 1247–1256, 2003.

[27] R. T. Lacoss, "Distributed mixed sensor aircraft tracking," in *American Control Conference*, pp. 1827–1830, 1987.

[28] J. M. Kahn, R. H. Katz, and K. S. J. Pister, "Next century challenges: Mobile networking for smart dust," in *Proceedings of the 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking*, pp. 271–278, 1999.

[29] J. Rabaey, M. Ammer, J. da Silva, J.L., D. Patel, and S. Roundy, "Picoradio supports ad hoc ultra-low power wireless networking," *Computer*, vol. 33, no. 7, pp. 42–48, 2000.

[30] C. Grimm, P. Neumann, and S. Mahlknecht, *Embedded Systems for Smart Appliances and Energy Management*. Springer, 2013.

[31] Cisco, http://www.cisco.com/web/solutions/trends/iot/industrial_networking.html, *Industrial Networking and IoT*.

[32] R. Conant, "Wireless sensor networks: Driving the new industrial revolution," *Industrial Embedded Systems*, 2006.

[33] A. Tiwari, F. Lewis, and S. S. Ge, "Wireless sensor network for machine condition based maintenance," in *Control, Automation, Robotics and Vision Conference*, vol. 1, pp. 461–467, 2004.

[34] M. Becker, B.-L. Wenning, C. Görg, R. Jedermann, and A. Timm-Giel, "Logistic applications with wireless sensor networks," in *Proceedings of the 6th Workshop on Hot Topics in Embedded Networked Sensors*, pp. 6:1–6:5, 2010.

[35] S. Spieker and C. Rohrig, "Localization of pallets in warehouses using wireless sensor networks," in *Mediterranean Conference on Control and Automation*, pp. 1833–1838, 2008.

[36] H. Ren, M.-H. Meng, and X. Chen, "Physiological information acquisition through wireless biomedical sensor networks," in *IEEE International Conference on Information Acquisition*, p. 6, 2005.

[37] P. S. Pandian, A. K. Whitchurch, J. K. Abraham, H. Bhusan Baskey, J. K. Radhakrishnan, V. K. Varadan, V. C. Padaki, K. U. Bhasker Rao, and R. E. Harbaugh, "Smart vest: wearable multi-parameter remote physiological monitoring system," in *Medical Engineering & Physics*, vol. 6931, pp. 69310Q–69310Q–8, 2008.

[38] U. Anliker, J. A. Ward, P. Lukowicz, G. Troster, F. Dolveck, M. Baer, F. Keita, E. B. Schenker, F. Catarsi, L. Coluccini, A. Belardinelli, D. Shklarski, M. Alon, E. Hirt, R. Schmid, and M. Vuskovic, "Amon: A wearable multiparameter medical monitoring and alert system," *Trans. Info. Tech. Biomed.*, vol. 8, no. 4, pp. 415–427, 2004.

[39] R. Paradiso, G. Loriga, and N. Taccini, "A wearable health care system based on knitted integrated sensors," *IEEE Transactions on Information Technology in Biomedicine*, vol. 9, no. 3, pp. 337–344, 2005.

[40] J. Porter, P. Arzberger, H.-W. Braun, P. Bryant, S. Gage, T. Hansen, P. Hanson, C.-C. Lin, F.-P. Lin, T. Kratz, W. Michener, S. Shapiro, and T. Williams, "Wireless Sensor Networks for Ecology," *BioScience*, vol. 55, no. 7, 2005.

[41] M. Durisic, Z. Tafa, G. Dimic, and V. Milutinovic, "A survey of military applications of wireless sensor networks," in *Mediterranean Conference on Embedded Computing*, pp. 196–199, 2012.

[42] A. Durresi, M. Durresi, and L. Barolli, "Security of mobile and heterogeneous wireless networks in battlefields," in *International Conference on Parallel Processing*, pp. 167–172, 2008.

[43] A.-B. Garcia-Hernando, *Problem Solving for Wireless Sensor Networks*. Springer, 2008.

[44] T. N. Le, A. Pegatoquet, O. Berder, and O. Sentieys, "Multi-source power manager for super-capacitor based energy harvesting wsn," in *Proceedings of the 1st International Workshop on Energy Neutral Sensing Systems*, pp. 19:1–19:2, 2013.

[45] M. T. Penella-Lopez and M. Gasulla-Forner, *Powering Autonomous Sensors: An Integral Approach with Focus on Solar and RF Energy Harvesting*. Springer, 2011.

[46] V. Ç. Güngör and G. P. Hancke, "Industrial wireless sensor networks: Applications, protocols, and standards," CRC Press, Mar. 2013.

[47] T. N. Le, A. Pegatoquet, O. Sentieys, O. Berder, and C. Belleudy, "Duty-cycle power manager for thermal-powered wireless sensor networks," in *International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC)*, pp. 1645–1649, 2013.

[48] A. Holmes, G. Hong, K. Pullen, and K. Buffard, "Axial-flow microturbine with electromagnetic generator: design, cfd simulation, and prototype demonstration," in *IEEE International Conference on Micro Electro Mechanical Systems*, pp. 568–571, 2004.

[49] Y. Qi and M. C. McAlpine, "Nanotechnology-enabled flexible and biocompatible energy harvesting," *Energy Environ. Sci.*, vol. 3, pp. 1275–1285, 2010.

[50] J. Carmo, L. Goncalves, and J. Correia, "Thermoelectric microconverter for energy harvesting systems," *IEEE Transactions on Industrial Electronics*, vol. 57, no. 3, pp. 861–867, 2010.

[51] M. Marzencki, Y. Ammar, and S. Basrour, "Integrated power harvesting system including a MEMS generator and a power management circuit," in *International Solid-State Sensors, Actuators and Microsystems Conference*, pp. 887–890, 2007.

[52] R. Elfrink, T. M. Kamel, M. Goedbloed, S. Matova, D. Hohlfeld, Y. van Andel, and R. van Schaijk, "Vibration energy harvesting with aluminum nitride-based piezoelectric devices," *Journal of Micromechanics and Microengineering*, vol. 19, no. 9, 2009.

[53] A. Almusallam, R. Torah, D. Zhu, M. Tudor, and S. Beeby, "Screen-printed piezoelectric shoe-insole energy harvester using an improved flexible pzt-polymer composites," in *Journal of Physics: Conference Series*, vol. 476, p. 012108, I, 2013.

[54] *Enerbee Harvesting*. http://www.enerbee.fr/.

[55] P. Nintanavongsa, U. Muncuk, D. R. Lewis, and K. R. Chowdhury, "Design optimization and implementation for rf energy harvesting circuits," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 2, no. 1, pp. 24–33, 2012.

[56] *Powercast*. http://www.powercastco.com/.

[57] N. Ferry, S. Ducloyer, N. Julien, and D. Jutel, "Energy estimator for weather forecasts dynamic power management of wireless sensor networks," in *Integrated Circuit and System Design. Power and Timing Modeling, Optimization, and Simulation*, vol. 6951, pp. 122–132, Springer, 2011.

[58] J. Rabaey, J. Ammer, T. Karalar, S. Li, B. Otis, M. Sheets, and T. Tuan, "Picoradios for wireless sensor networks: the next challenge in ultra-low power design," in *IEEE International Solid-State Circuits Conference*, vol. 1, pp. 200–201, 2002.

[59] K. S. Kwak, S. Ullah, and N. Ullah, "An overview of ieee 802.15.6 standard," in *International Symposium on Applied Sciences in Biomedical and Communication Technologies*, pp. 1–6, 2010.

[60] M. Al Ameen, S. Islam, and K. Kwak, "Energy saving mechanisms for mac protocols in wireless sensor networks," *International Journal of Distributed Sensor Networks*, 2010.

[61] S. Du, A. K. Saha, and D. B. Johnson, "Rmac: A routing-enhanced duty-cycle mac protocol for wireless sensor networks," in *INFOCOM IEEE International Conference on Computer Communications*, pp. 1478–1486, 2007.

[62] Y. Sun, O. Gurewitz, and D. B. Johnson, "Ri-mac: a receiver-initiated asynchronous duty cycle mac protocol for dynamic traffic loads in wireless sensor networks," in *Proceedings of the 6th ACM conference on Embedded network sensor systems*, pp. 1–14, 2008.

[63] O. Landsiedel, E. Ghadimi, S. Duquennoy, and M. Johansson, "Low power, low delay: opportunistic routing meets duty cycling," in *Proceedings of the 11th international conference on Information Processing in Sensor Networks*, pp. 185–196, 2012.

[64] S. Nath and P. B. Gibbons, "Communicating via fireflies: geographic routing on duty-cycled sensors," in *Proceedings of the 6th international conference on Information processing in sensor networks*, pp. 440–449, 2007.

[65] K. Wang, L. Wang, C. Ma, L. Shu, and J. Rodrigues, "Geographic routing in random duty-cycled wireless multimedia sensor networks," in *GLOBECOM Workshops*, pp. 230–234, 2010.

[66] E.-Y. Lin, J. Rabaey, S. Wiethoelter, and A. Wolisz, "Receiver initiated rendezvous schemes for sensor networks," in *IEEE GLOBECOM Global Telecommunications Conference,*, vol. 5, pp. 3116–3122, 2005.

[67] C. Buratti, F. Fabbri, and R. Verdone, "Area throughput of an ieee 802.15.4 based wireless sensor network.," in *EWSN Springer*, vol. 5432, pp. 1–16, 2009.

[68] D. Dardari, A. Conti, C. Buratti, and R. Verdone, "Mathematical evaluation of environmental monitoring estimation error through energy-efficient wireless sensor networks," *IEEE Transactions on Mobile Computing*, vol. 6, no. 7, pp. 790–802, 2007.

[69] K. Wehrle, J. Gross, and M. Gunes, *Modeling and Tools for Network Simulation*. Springer, 2010.

[70] Senslab, *Very large scale open wireless sensor network testbed*. http://www.senslab.info/.

[71] www.sensorscope.ch, *Sensorscope*.

[72] *Greenorbs*. http://www.greenorbs.org/.

[73] *WISEBED: Wireless Sensor Network Testbeds*. http://wisebed.eu/.

[74] C. B. des Roziers, G. Chelius, T. Ducrocq, E. Fleury, A. Fraboulet, A. Gallais, N. Mitton, T. Noël, and J. Vandaele, "Using senslab as a first class scientific tool for large scale wireless sensor network experiments," in *Proceedings of the 10th International IFIP TC 6 Conference on Networking - Volume Part I*, NETWORKING'11, (Berlin, Heidelberg), pp. 147–159, Springer-Verlag, 2011.

[75] S. Kurkowski, T. Camp, and M. Colagrosso, "Manet simulation studies: The incredibles," *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 9, no. 4, pp. 50–61, 2005.

[76] I. Stojmenovic, "Simulations in wireless sensor and ad hoc networks: matching and advancing models, metrics, and solutions," *IEEE Communications Magazine*, vol. 46, no. 12, pp. 102–107, 2008.

[77] T. Andel and A. Yasinsac, "On the credibility of manet simulations," *Computer*, vol. 39, no. 7, pp. 48–54, 2006.

[78] M. Kropff, T. Krop, M. Hollick, P. Mogre, and R. Steinmetz, "A survey on real world and emulation testbeds for mobile ad hoc networks," in *International Conference on Testbeds and Research Infrastructures for the Development of Networks and Communities*, pp. 6–453, 2006.

[79] W. Kiess and M. Mauve, "A survey on real-world implementations of mobile ad-hoc networks," *Ad Hoc Netw.*, vol. 5, pp. 324–339, Apr. 2007.

[80] V. Shnayder, M. Hempstead, B.-r. Chen, G. W. Allen, and M. Welsh, "Simulating the power consumption of large-scale sensor network applications," in *International Conference on Embedded Networked Sensor Systems*, SenSys '04, pp. 188–200, 2004.

[81] *Network Simulator 3*. http://www.nsnam.org/.

[82] *OPNET*. http://www.riverbed.com/products/performance-management-control/.

[83] G. Chelius, A. Fraboulet, and E. Benhamida, *WSNet - an Event-Driven Simulator for Wireless Networks*. http://wsnet.gforge.inria.fr/.

[84] A. Boulis, *Castalia: A simulator for wireless sensor networks and body area networks*. http://castalia.npc.nicta.com.au/, 2011.

[85] D. Henriksson, A. Cervin, and K. Arzen, "Truetime: Simulation of control loops under shared computer resources," in *Proceedings of the 15th IFAC World Congress on Automatic Control*, 2002.

[86] M. Andersson, D. Henriksson, A. Cervin, and K. Arzen, "Simulation of wireless networked control systems," in *IEEE Conference on Decision and Control*, pp. 476–481, 2005.

[87] M. Bjorkbom and H. Ohman, "Integration of piccsim and truetime," in *In proceeding of Wireless Automation Tactical Communication Systems*, 2012.

[88] *Prowler: Probabilistic wireless network simulator*. http://www.isis.vanderbilt.edu/Projects/nest/prowler/.

[89] *MicaZ*. http://www.memsic.com/userfiles/files/Datasheets/WSN/micaz_datasheet-t.pdf.

[90] D. Benedetti, C. Petrioli, and D. Spenza, "Greencastalia: An energy-harvesting-enabled framework for the castalia simulator," in *Proceedings of the 1st International Workshop on Energy Neutral Sensing Systems*, ENSSys '13, pp. 7:1–7:6, 2013.

[91] A. Sánchez, S. Climent, S. Blanc, J. V. Capella, and I. Piqueras, "Wsn with energy-harvesting: Modeling and simulation based on a practical architecture using real radiation levels," in *Proceedings of the 6th ACM Workshop on Performance Monitoring and Measurement of Heterogeneous Wireless and Wired Networks*, pp. 17–24, 2011.

[92] W. Du, D. Navarro, F. Mieyeville, and F. Gaffiot, "Towards a taxonomy of simulation tools for wireless sensor networks," in *Proceedings of the 3rd International ICST Conference on Simulation Tools and Techniques*, SIMUTools '10, pp. 52:1–52:7, 2010.

[93] P. Levis, N. Lee, M. Welsh, and D. Culler, "Tossim: Accurate and scalable simulation of entire tinyos applications," in *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems*, SenSys '03, (New York, NY, USA), pp. 126–137, ACM, 2003.

[94] B. Titzer, D. Lee, and J. Palsberg, "Avrora: scalable sensor network simulation with precise timing," in *Information Processing in Sensor Networks, 2005. IPSN 2005. Fourth International Symposium on*, pp. 477–482, April 2005.

[95] Z. Zhang, Z. Lu, Q. Chen, X. Yan, and L.-R. Zheng, "Cosmo: Co-simulation with matlab and omnet++ for indoor wireless networks," in *Proc. IEEE Global Telecommunications Conf. GLOBECOM*, 2010.

[96] T. Kohtamaki, M. Pohjola, J. Brand, and L. M. Eriksson, "Piccsim toolchain - design, simulation and automatic implementation of wireless networked control systems," in *Proc. Int. Conf. Networking, Sensing and Control ICNSC '09*, pp. 49–54, 2009.

[97] M. S. Hasan, H. Yu, A. Carrington, and T. C. Yang, "Co-simulation of wireless networked control systems over mobile ad hoc network using simulink and opnet," *IET Communications*, 2009.

[98] J. F. Christmann, E. Beigne, C. Condemine, J. Willemin, and C. Piguet, "Energy harvesting and power management for autonomous sensor nodes," in *Design Automation Conference (DAC), 2012 49th ACM/EDAC/IEEE*, pp. 1049–1054, June 2012.

[99] A. N. Celik and N. Acikgoz, "Modelling and experimental verification of the operating current of mono-crystalline photovoltaic modules using four- and five-parameter models," *Applied Energy*, vol. 84, no. 1, pp. 1 – 15, 2007.

[100] D. Sera, R. Teodorescu, and P. Rodriguez, "Pv panel model based on datasheet values," in *Proc. IEEE Int. Symp. Industrial Electronics ISIE 2007*, pp. 2392–2396, 2007.

[101] E. Lefeuvre, D. Audigier, C. Richard, and D. Guyomar, "Buck-boost converter for sensorless power optimization of piezoelectric energy harvester," *IEEE Transactions on Power Electronics*, 2007.

[102] A.K.Mukerjee and N. Thakur, *Photovoltaic Systems: Analysis and Design*. Prentice-Hall, 2011.

[103] O. Erdinc, B. Vural, and M. Uzunoglu, "A dynamic lithium-ion battery model considering the effects of temperature and capacity fading," in *Proc. Int Clean Electrical Power Conf*, 2009.

[104] *Solar Energy Services for Professionals*. http://www.soda-is.com/.

[105] D. Porcarelli, D. Spenza, D. Brunelli, A. Cammarano, C. Petriolo, and L. Benini, "Adaptive rectifier driven by power intake predictors for energy harvesting sensor networks," *IEEE Journal on Emerging and Selected Topics in Power Electronics*, 2014.

[106] D. Carli, D. Brunelli, D. Bertozzi, and L. Benini, "A high-efficiency wind-flow energy harvester using micro turbine," in *Proceedings of International Symposium on Power Electronics Electrical Drives Automation and Motion (SPEEDAM)*, pp. 778–783, 2010.

[107] C. Gentile, F. Valoit, and N. Moayeri, "A raytracing model for wireless propagation in tunnels with varying cross section," in *Proceedings of IEEE Global Communications Conference (GLOBECOM)*, pp. 5027–5032, Dec 2012.

[108] F. Broekaert, A. Didioui, C. Bernier, and O. Sentieys, "Prototyping an Energy Harvesting Wireless Sensor Network Application Using HarvWSNet,"

in *Proceedings of 26th International Conference on Architecture of Computing Systems (ARCS)*, pp. 1–6, 2013.

[109] *MSP430 Solar Energy Harvesting Development Tool*. http://www.ti.com/tool/ez430-rf2500-seh.

[110] M. Alioto, "Ultra-low power vlsi circuit design demystified and explained: A tutorial," *IEEE Transactions on Circuits and Systems*, vol. 59, no. 1, pp. 3–29, 2012.

[111] B. Cook, A. Berny, A. Molnar, S. Lanzisera, and K. Pister, "An ultra-low power 2.4ghz rf transceiver for wireless sensor networks in 0.13/spl mu/m cmos with 400mv supply and an integrated passive rx front-end," in *IEEE International Solid-State Circuits Conference*, pp. 1460–1469, 2006.

[112] S. Drago, D. Leenaerts, F. Sebastiano, L. Breems, K. Makinwa, and B. Nauta, "A 2.4ghz 830pj/bit duty-cycled wake-up receiver with −82dbm sensitivity for crystal-less wireless sensor nodes," in *IEEE International Solid-State Circuits Conference*, pp. 224–225, 2010.

[113] A. Chandrakasan, D. Daly, D. Finchelstein, J. Kwong, Y. Ramadass, M. Sinangil, V. Sze, and N. Verma, "Technologies for ultradynamic voltage scaling," *Proceedings of the IEEE*, vol. 98, pp. 191–214, Feb 2010.

[114] A. A. Abidi, G. J. Pottie, and W. J. Kaiser, "Power-conscious design of wireless circuits and systems," *Proceedings of the IEEE*, vol. 88, no. 10, pp. 1528–1545, 2000.

[115] D. Ernst, S. Das, S. Lee, D. Blaauw, T. Austin, T. Mudge, N. S. Kim, and K. Flautner, "Razor: circuit-level correction of timing errors for low-power operation," *IEEE Micro*, vol. 24, no. 6, pp. 10–20, 2004.

[116] N. Baccour, A. Koubâa, L. Mottola, M. A. Zúñiga, H. Youssef, C. A. Boano, and M. Alves, "Radio link quality estimation in wireless sensor networks: A survey," *ACM Transactions on Sensor Networks*, 2012.

[117] *SDR Forum*. http://www.wirelessinnovation.org/.

[118] *GNU Radio*. http://www.gnuradio.org.

[119] K. Iniewski, *Wireless Technologies: Circuits, Systems, and Devices*. CRC Press, 2007.

[120] B. Murmann, *ADC Performance Survey*. http://www.stanford.edu/ murmann/adcsurvey.html.

[121] M. Ismail, *Radio Design in Nanometer Technologies*. Springer, 2007.

[122] J. Mitola, "Cognitive radio for flexible mobile multimedia communications," in *IEEE International Workshop on Mobile Multimedia Communications*, pp. 3–10, 1999.

[123] S. Haykin, "Cognitive radio: brain-empowered wireless communications," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 2, pp. 201–220, 2005.

[124] *IEEE 802 LAN/MAN Standards Committee*. http://www.ieee802.org/.

[125] Texas Instruments, http://www.ti.com/product/cc2420, *CC2420*.

[126] J. Gomez and A. Campbell, "Variable-range transmission power control in wireless ad hoc networks," *IEEE Transactions on Mobile Computing*, vol. 6, no. 1, pp. 87–99, 2007.

[127] I. Khemapech, A. Miller, and I. Duncan, "A survey of transmission power control in wireless sensor networks," in *Proceedings of the 8th Annual Postgraduate Symposium on the Convergence of Telecommunications, Networking and Broadcasting*, pp. 15–20, 2007.

[128] S. Panichpapiboon, G. Ferrari, and O. Tonguz, "Optimal transmit power in wireless sensor networks," *IEEE Transactions on Mobile Computing*, vol. 5, no. 10, pp. 1432–1447, 2006.

[129] S. Dwivedi, B. Amrutur, and N. Bhat, "Power scalable radio receiver design based on signal and interference condition," *Journal of Low Power Electronics and Applications*, 2012.

[130] S. Sen, D. Banerjee, M. Verhelst, and A. Chatterjee, "A power-scalable channel-adaptive wireless receiver based on built-in orthogonally tunable lna," *IEEE Transactions on Circuits and Systems*, 2012.

[131] X. Huang, P. Harpe, G. Dolmans, H. de Groot, and J. Long, "A 780–950 MHz, 64–146 μW power-scalable synchronized-switching ook receiver for wireless event-driven applications," vol. 49, no. 5, pp. 1135–1147, 2014.

[132] T. Taris, A. Mabrouki, H. Kraimia, Y. Deval, and J. B. Begueret, "Reconfigurable ultra low power LNA for 2.4GHz wireless sensor networks," in *IEEE International Conference on Electronics, Circuits, and Systems (ICECS)*, 2010.

[133] K. Ghosal, T. Anand, V. Chaturvedi, and B. Amrutur, "A power-scalable RF CMOS receiver for 2.4 GHz wireless sensor network applications," in *19th IEEE International Conference on Electronics, Circuits and Systems*, 2012.

[134] A. Oguz, D. Morche, and C. Dehollaini, "Adaptive power reconfigurability for decreasing power dissipation of wireless personal area network receivers," in *International Symposium on Circuits and Systems*, 2011.

[135] P. A. Nuyts, P. Reynaert, and W. Dehaene, *Continuous-Time Digital Front-Ends for Multistandard Wireless Transmission*. Springer, 2014.

[136] B. Sahu and G. Rincon-Mora, "A high-efficiency linear RF power amplifier with a power-tracking dynamically adaptive buck-boost supply," *IEEE Transactions on Microwave Theory and Techniques*, vol. 52, no. 1, pp. 112–120, 2004.

[137] R. Brederlow, W. Weber, J. Sauerer, S. Donnay, P. Wambacq, and M. Vertregt, "A mixed-signal design roadmap," *IEEE Des Test Comput*, vol. 18, no. 6, pp. 34–46, 2001.

[138] L. Leyssenne, E. Kerhervé, and Y. Deval, *Reconfigurable RF Power Amplifiers on Silicon for Wireless Handsets*. Springer, 2011.

[139] ITRS, *System Drivers*. http://www.itrs.net, 2011.

[140] T. Taris, J. Begueret, and Y. Deval, "A 60µw LNA for 2.4 GHz wireless sensors network applications," in *IEEE Radio Frequency Integrated Circuits*, pp. 1–4, 2011.

[141] H. Kraimia, T. Taris, J. Begueret, and Y. Deval, "A 2.4 ghz ultra-low power current-reuse bleeding mixer with resistive feedback," in *IEEE International Conference on Electronics, Circuits and Systems*, pp. 488–491, 2011.

[142] M. Krishna, J. Xie, M. Do, C. Boon, K. S. Yeo, and A. Do, "A 1.8-v 3.6-mw 2.4-ghz fully integrated cmos frequency synthesizer for ieee 802.15.4," in *Proc. IEEE VLSI System on Chip Conference*, pp. 387–391, 2010.

[143] Z. Jun, F. Xiangning, L. Bin, and Z. Weiwei, "An improved low phase noise and low power consumption VCO for WSN applications," in *Proc. Int Conf Wireless Communications and Signal Processing*, 2010.

[144] Q. Gu, *RF System Design of Transceivers for Wireless Communications*. Springer, 2005.

[145] A. Woo, T. Tong, and D. Culler, "Taming the underlying challenges of reliable multihop routing in sensor networks," in *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems*, SenSys '03, pp. 14–27, 2003.

[146] K. Srinivasan and P. Levis, "Rssi is under appreciated," in *Proceedings of the Third Workshop on Embedded Networked Sensors (EmNets 2006)*, 2006.

[147] K. S. Prabh and J.-H. Hauer, "Opportunistic packet scheduling in body area networks," in *Proceedings of the 8th European Conference on Wireless Sensor Networks*, EWSN'11, pp. 114–129, Springer, 2011.

[148] Atmel, *AT86RF230*. http://www.atmel.com/devices/at86rf230.aspx.

[149] Y. Chen and A. Terzis, "On the mechanisms and effects of calibrating rssi measurements for 802.15.4 radios," in *Proceedings of the 7th European Conference on Wireless Sensor Networks*, EWSN'10, (Berlin, Heidelberg), pp. 256–271, Springer-Verlag, 2010.

[150] C. Renner, S. Ernst, C. Weyer, and V. Turau, "Prediction accuracy of link-quality estimators," in *Proceedings of the 8th European Conference on Wireless Sensor Networks*, EWSN'11, pp. 1–16, Springer, 2011.

[151] P. Jiang, Q. Huang, J. Wang, X. Dai, and R. Lin, "Research on wireless sensor networks routing protocol for wetland water environment monitoring," in *First International Conference on Innovative Computing, Information and Control*, vol. 3, pp. 251–254, 2006.

[152] D. S. J. De Couto, D. Aguayo, J. Bicket, and R. Morris, "A high-throughput path metric for multi-hop wireless routing," in *Proceedings of the 9th Annual International Conference on Mobile Computing and Networking*, MobiCom '03, pp. 134–146, 2003.

[153] A. Cerpa, J. Wong, L. Kuang, M. Potkonjak, and D. Estrin, "Statistical model of lossy links in wireless sensor networks," in *Fourth International Symposium on Information Processing in Sensor Networks*, pp. 81–88, 2005.

[154] N. Baccour, A. Koubâa, H. Youssef, M. Ben Jamâa, D. do Rosário, M. Alves, and L. B. Becker, "F-lqe: a fuzzy link quality estimator for wireless sensor networks," in *Proceedings of the 7th European conference on Wireless Sensor Networks*, pp. 240–255, 2010.

[155] R. Fonseca, O. Gnawali, K. Jamieson, and P. Levis, "Four-bit wireless link estimation.," in *Proceedings of the Sixth Workshop on Hot Topics in Networks*, pp. 240–255, 2007.

[156] C. Boano, M. Zuniga, T. Voigt, A. Willig, and K. Romer, "The triangle metric: Fast link quality estimation for mobile wireless sensor networks," in *Proceedings of 19th International Conference on Computer Communications and Networks*, pp. 1–7, 2010.

[157] M. Krunz, A. Muqattash, and S.-J. Lee, "Transmission power control in wireless ad hoc networks: challenges, solutions and open issues," *IEEE Network*, vol. 18, pp. 8–14, Sept 2004.

[158] S. Lin, J. Zhang, G. Zhou, L. Gu, J. A. Stankovic, and T. He, "Atpc: Adaptive transmission power control for wireless sensor networks," in *Proceedings of the 4th International Conference on Embedded Networked Sensor Systems*, SenSys '06, pp. 223–236, 2006.

[159] D. Son, B. Krishnamachari, and J. Heidemann, "Experimental study of the effects of transmission power control and blacklisting in wireless sensor networks," in *IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks*, pp. 289–298, 2004.

[160] *Mixim Simulator*. http://mixim.sourceforge.net/.

[161] E. B. Hamida, G. Chelius, and J.-M. Gorce, "Scalable versus accurate physical layer modeling in wireless network simulations," in *22nd Workshop on Principles of Advanced and Distributed Simulation*, pp. 127 – 134, 2008.

[162] A. Stetsko, T. Smolka, V. Matyas, and F. Jurnečka, "On the credibility of wireless sensor network simulations: Evaluation of intrusion detection system," in *Proceedings of the 5th International ICST Conference on Simulation Tools and Techniques*, SIMUTOOLS '12, pp. 75–84, 2012.

[163] D. G. Yoon, S. Y. Shin, W. H. Kwon, and H. S. Park, "Packet error rate analysis of ieee 802.11b under ieee 802.15.4 interference," *Vehicular Technology Conference*, vol. 3, pp. 1186–1190, 2006.

[164] L. Angrisani, M. Bertocco, G. Gamba, and A. Sona, "Modeling the performance of csma-ca based wireless networks versus interference level," in *Proc. IEEE Instrumentation and Measurement Technology IMTC 2008*, pp. 376–381, 2008.

[165] A. V. Kini, N. Singhal, and S. Weber, "Performance of a wsn in the presence of channel variations and interference," in *Proc. IEEE Wireless Communications and Networking Conf. WCNC 2008*, pp. 2899–2904, 2008.

[166] H. Huo, Y. Xu, C. C. Bilen, and H. Zhang, "Coexistence issues of 2.4ghz sensor networks with other rf devices at home," *Third International Conference on Sensor Technologies and Applications*, pp. 200–205, 2009.

[167] G. Chelius, A. Fraboulet, and E. Benhamida, *WSNet - an Event-Driven Simulator for Wireless Networks*. http://wsnet.gforge.inria.fr/, 2008.

[168] J. Proakis, *Digital Communications*. McGraw Hill, 2000.

[169] TI, *CC2520 Datasheet 2.4GHz IEEE 802.15.4/ZigBee RF Transceiver*. Texas Instruments, december 2007.

[170] T. S. Rappaport, *Wireless Communications: Principles and Practice*. 1996.

[171] B. De Muer and M. Steyaert, *CMOS fractional-N synthesizers: design for high spectral purity and monolithic integration*. Springer, 2003.

[172] A. Aktas and M. Ismail, *CMOS PLLs and VCOs for 4G wireless*. Springer, 2004.

[173] D. A. Johns and K. Martin, *Analog Integrated Circuit Design*. John Wiley and Sons, 1997.