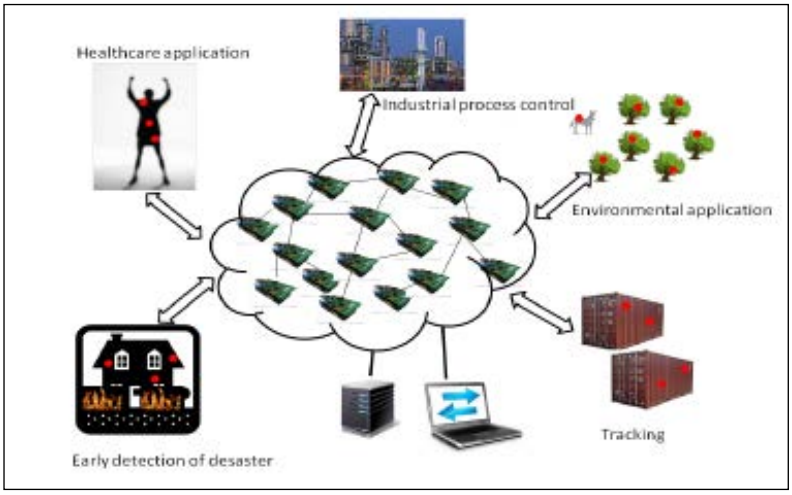
Reliability in wireless sensor networks

Nourhene Maalel

Par **Nourhene MAALEL**

Reliability in wireless sensor networks

Thèse présentée
pour l'obtention du grade
de Docteur de l'UTC



Soutenue le 30 juin 2014
Spécialité : Technologies de l'Information et des Systèmes

D1944

Fiabilisation des transmissions dans les réseaux de capteurs sans fils

THESE DE DOCTORAT

présentée et soutenue le **30 Juin 2014**
pour l'obtention du

Doctorat de l'Université de Technologie de Compiègne

Spécialité: Technologie de l'information et des systèmes

Par **Nourhene Maalel**

Membres du jury:

Président du jury:	Pr. Jacques Carlier	Professeur, UTC
Rapporteurs:	Pr. Dominique Gaiti	Professeur, UTT
	Pr. Hossam Afifi	Professeur, Télécom SudParis
Examinateur:	Pr. Marcelo Amorim	Professeur, UPMC
Encadrants:	Dr. Mounir Kellil	Ingénieur chercheur, CEA
	Mr. Pierre Roux	Ingénieur chercheur, CEA
Directeur de thèse:	Pr. Abdelmadjid Bouabdallah	Professeur, UTC

Quote

"The reward of our work is not what we get, but what we become."

P.C

Abstract

Over the past decades, we have witnessed a proliferation of potential application domains for wireless sensor networks (WSN). A comprehensive number of new services such as environment monitoring, target tracking, military surveillance and healthcare applications have arisen. These networked sensors are usually deployed randomly and left unattended to perform their mission properly and efficiently. Meanwhile, sensors have to operate in a constrained environment with functional and operational challenges mainly related to resource limitations (energy supply, scarce computational abilities...) and to the noisy real world of deployment. This harsh environment can cause packet loss or node failure which hamper the network activity. Thus, continuous delivery of data requires reliable data transmission and adaptability to the dynamic environment. Ensuring network reliability is consequently a key concern in WSNs and it is even more important in emergency application such disaster management application where reliable data delivery is the key success factor.

The main objective of this thesis is to design a reliable end to end solution for data transmission fulfilling the requirements of the constrained WSNs. We tackle two design issues namely recovery from node failure and packet losses and propose solutions to enhance the network reliability. We start by studying WSNs features with a focus on technical challenges and techniques of reliability in order to identify the open issues. Based on this study, we propose a scalable and distributed approach for network recovery from node failures in WSNs called CoMN2. Then, we present a lightweight mechanism for packet loss recovery and route quality awareness in WSNs called AJIA. This protocol exploits the overhearing feature characterizing the wireless channels as an implicit acknowledgment (ACK) mechanism. In addition, the protocol allows for an adaptive selection of the routing path by achieving required retransmissions on the most reliable link. We prove that AJIA outperforms its competitor AODV in term of delivery ratio in different channel conditions. Thereafter, we present ARRP, a variant of AJIA, combining the strengths of retransmissions, node collaboration and Forward Error Correction (FEC) in order to provide a reliable packet loss recovery scheme. We verify the efficiency of ARRP through extensive simulations which proved its high reliability in comparison to its competitor.

Key words: Wireless sensor networks, reliability, packet loss, node failure, retransmission

Résumé

Vu les perspectives qu'ils offrent, les réseaux de capteur sans fil (RCSF) ont perçu un grand engouement de la part de la communauté de recherche ces dernières années. Les RCSF couvrent une large gamme d'applications variant du contrôle d'environnement, le pistage de cible aux applications de santé. Les RCSFs sont souvent déployés aléatoirement. Ce dispersement des capteurs nécessite que les protocoles de transmission utilisés soient résistants aux conditions environnementales (fortes chaleurs ou pluies par exemple) et aux limitations de ressources des nœuds capteurs. En effet, la perte de plusieurs nœuds capteurs peut engendrer la perte de communication entre les différentes entités. Ces limitations peuvent causer la perte des paquets transmis ce qui entrave l'activité du réseau. Par conséquent, il est important d'assurer la fiabilité des transmissions de données dans les RCSF d'autant plus pour les applications critiques comme la détection d'incendies.

Dans cette thèse, nous proposons une solution complète de transmission de données dans les RCSF répondant aux exigences et contraintes de ce type de réseau. Dans un premier temps, nous étudions les contraintes et les challenges liés à la fiabilisation des transmissions dans les RCSFs et nous examinons les travaux proposés dans la littérature. Suite à cette étude nous proposons COMN2, une approche distribuée et scalable permettant de faire face à la défaillance des nœuds. Ensuite, nous proposons un mécanisme de contrôle d'erreur minimisant la perte de paquets et proposant un routage adaptatif en fonction de la qualité du lien. Cette solution est basée sur des acquittements implicites (overhearing) pour la détection des pertes des paquets. Nous proposons ensuite ARRP une variante de AJIA combinat les avantages des retransmissions, de la collaboration des nœuds et des FEC. Enfin, nous simulons ces différentes solutions et vérifions leurs performances par rapport à leurs concurrents de l'état de l'art.

Mots clés: Réseaux de Capteur Sans Fil, Fiabilité, perte de paquets, retransmission

Contents

List of publications

International Conference

- N.Maalel, M.Kellil, P.Roux, A.Bouabdallah; **Fast Restoration of Connectivity for Wireless Sensor Networks**;*The 12th International Conference on Next Generation Wired/Wireless Advanced Networking (NEW2AN)* ; St. Petersburg, Russia, May 2012

- N.Maalel, E.Natalizio, A.Bouabdallah, P.Roux and M.Kellil; **Reliability for Emergency Applications in Internet of Things**; *First IEEE DCOSS workshop on Internet of Things:Idea and Perspectives (IoTIP)*; Boston, USA, May 2013

- N.Maalel, P.Roux, M.Kellil and A.Bouabdallah; **Adaptive Reliable Routing Protocol for Wireless Sensor Networks**;*The Ninth International Conference on Wireless and Mobile Communications(ICWMC)*;***Best Paper Award***; Nice, France, July 2013

International Journal

- N.Maalel, M.Kellil, P.Roux, A.Bouabdallah; **Toward Adaptive Reliable Data Transmission In Wireless Sensor Networks** ;*Under submission to Annals of Telecommunications*

List of Figures

List of Tables

Abbreviations

ACK	ACKnowledgment
ADC	Analog-Digital Converter
ARQ	Automatic Repeat reQuest
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance
CFP	Contention Free Period
CAP	Contention Access Period
ECC	Error Correction Code
ETE	End To End
ETX	Expected Transmission Count
ETT	Expected Transmission Time
FEC	Forward Error Correction
HBH	Hop By Hop
Impl Ack	Implicit Acknowledgment
IoT	Internet Of Things
LQI	Link Quality Indicator
MAC	Media Access Control
NC	Network Coding
QoS	Quality of service
RFID	Radio-Frequency IDentification
RSSI	Received Signal Strength Indicator
SNR	Signal to Noise Ratio
WPAN	Wireless Personal Area Networks
WSN	Wireless Sensor Networks
WSAN	Wireless Sensor and Actuator Networks

Chapter 1

Introduction

1.1 General context

In the past decade, wireless communications [43] [36] have become key technologies, offering mobile and flexible infrastructure for industries, enterprises and individuals. The wireless mobile networks [37] are reaching the leadership as daily voice communications media, surpassing the wired telephone network. On the other side, the widely deployed WiFi technology [19] have also been great success. Beside all these wireless technologies, the Wireless Sensor Networks (WSN) [73] [3] emerges and revolutionizes monitoring applications. The flexibility, effectiveness, low-cost and rapid deployment of sensors, have fueled interest in the possible ubiquitous integration of a massive set of unattended sensors. Such networks enable a whole new class of autonomous control applications and services from environmental control of office buildings to the detection of forest fires. The substantial benefit of sensor networks is that they extend the computation capability to hazardous environment unreachable by human beings. Moreover, it has the potential to provide wealth of data about the environment in which they are deployed and send their results across the network to the end-users.

A natural architecture for such collaborative distributed sensors is a network with wireless links. Sensors have to operate in a complex and noisy real world where link state varies sporadically inducing transmission error, packet collision and interference. These undesirable consequences are the cause of packet loss. Furthermore, due to the short range of sensor nodes, data might have to be transmitted via a multi-hop scheme. This in turn introduces a lot of entry points for errors that can also become the cause of packet loss.

Additionally, uneven energy distribution and consumption or accidental failures due to harsh environments may cause sensors to shut down arbitrarily. These damaged nodes generate constrained region which provokes network partitioning and thus hamper network activity and provoke packet losses. More specifically, due to the convergent nature of traffic in WSNs, the data traffic is typically concentrated at the sensor nodes surrounding the sink. Consequently, those bottleneck nodes around the sink exhaust their batteries faster than other nodes which lead to the isolation of the sink and the drop off of the network reliability.

Thus, to ensure reliability, node failures and packet losses have to be recovered.

Ensuring network reliability is even more important in emergency application where reliable data delivery is the key success factor. For example, in a disaster management application, a large number of sensors can be dropped by a helicopter to report alarm once a dangerous situation is detected. Networking these sensors can assist rescue operations by locating survivors, identifying risky areas and making the rescue crew more aware of the overall situation. Accordingly, routing packets should be essentially an adaptive process, wherein reliable packet delivery can be achieved even though incidents (faulty routing nodes and links) occur, through a specified monitoring and planning operation. In this thesis, we tackle these two design objectives namely recovery from node failure and packet losses and propose solutions to enhance the network reliability.

1.2 Challenges and objectives

The main objective of this thesis is to design a reliable end to end solution for data transmission in wireless sensor networks. This objective encompasses the following challenges, which are to be specifically addressed:

- Proposal of reliable transport protocol which satisfies the constraints and challenges of WSNs: For this purpose, these challenges and their impact are to be investigated and reliability techniques are to be identified and appraised.
- Design of a recovery protocol from node failure which aims to provide availability of the data transmission and extend the network lifetime. Likewise, the proposed solution will have to fit within the energy constraint of WSNs.
- Design of a recovery protocol from packet loss: Based on packets retransmissions to alleviate the packet delivery ratio, the proposed solutions will eventually be exposed to an overhead generated by control messages and retransmitted packets. In this context, our proposed protocol minimizes this extra-overhead by finding a trade-off between reliability and resource consumption.
- Evaluation of the proposed solutions: In order to be satisfactory, the developed protocols must be validated with rigorous simulation and performance evaluation. The simulation environment has to be accurate enough reproducing as close as possible the features of real network (varying wireless channel, limited battery for sensor nodes...)

1.3 Organization of the manuscript

The remainder of this thesis is organized as follows: We start chapter 1 with a presentation of the basics of wireless sensor networks design, architecture and applications then we present sensor networks challenges. The aim of this first part is to help non-specialists in the field to get an overview of wireless sensor networks

and their main research issues in order to pave the way for the following deeper study of specific challenges. The second part of this chapter is dedicated to the reliability on WSNs. We firstly investigate the major causes of packet loss in WSN and then, we provide an overview of the main reliability techniques and we conclude by the necessity of elaborating a reliable transmission protocol fulfilling the challenges of WSNs.

Consequently, we introduce a recovery mechanism from node failure called CoMN2 in chapter 2. We begin by the presentation of the design goal as well as the protocol operation. We also provide a detailed performance evaluation from the points of view of reliability and latency.

With node failure recovery arises the need of providing a recovery from packet loss. For the sake of comprehensiveness and efficiency, we propose AJIA in chapter 3, a packet loss recovery protocol. A detailed overview of the protocol operation and its features are provided to emphasize its benefits. Chapter 4 reports on the performance evaluation of AJIA through simulation experiments. In this chapter, we analyze the performance of our solution in terms of end to end delivery ratio, latency, communication overhead and energy efficiency, and we discuss the results by comparing them to those of a well-known routing protocol. In chapter 5, we propose ARRP, our second packet loss recovery mechanism. With ARRP, we tackle packet loss problem by optimizing packet retransmission, relying on collaboration between nodes. Extensive simulation results are conducted to assess the efficiency of our protocol. The reported results show that ARRP is able to withstand packet losses more efficiently than its existent counterpart in the literature.

Finally, we end up this thesis by a summary of our contributions and an outline of our future work and perspectives.

Chapter 2

Wireless Sensor Networks basics and reliability techniques

Because of the unprecedented prospects they offer and the idea of eliminating human intervention, WSNs have perceived a tremendous attention from the academic and industrial communities over the last decade [98] [73] [3]. They cover a wide area of applications ranging from healthcare to environmental monitoring. In this chapter, we bring in the concept of WSN. To show the breadth of WSNs, we highlight their major applications. Then, we give an overview of their design, challenges and requirements. After that, the major causes of erroneous data forwarding in WSNs are discussed and finally we review the major existing techniques of reliability to tackle it.

2.1 WSN basics

2.1.1 Overview

WSNs are formed by a set of small low cost devices called sensor nodes as showed in figure 2.1. The small size and weight, the low cost of the hardware and the ease of deployment of such platforms enable the sensing of the environment in the least intrusive fashion. By spatially distributing tens or hundreds of such autonomous devices, a WSN can be built to cooperatively monitor physical or environmental conditions at different locations.

After their deployment, sensors deliver their sensed data (reading) to one or several dedicated nodes called sink nodes. Depending on the used topology, the sink is reachable in a single hop or in a multi-hop fashion using wireless transmissions. Once received by the sink, data can then be processed and analyzed permitting to the end user to undertake the adequate actions. Some networks may include special entities called actuators performing special action. Actuators could even be mobile such as robot. In these cases, we refer to these networks as Wireless Sensor and Actuator Networks (WSAN) [90].

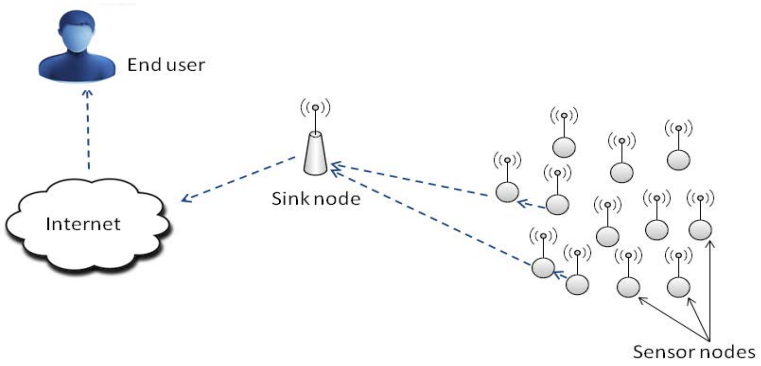


Figure 2.1: WSN architecture

2.1.2 WSN topology and architecture

The size of a WSN varies from a few nodes to several hundreds. The same goes for dedicated applications and their requirements. Accordingly, the topology of a wireless sensor network can vary from a simple star network to an advanced multi-hop wireless mesh network. In this section, we present the different modules forming a sensor node and give an overview of the three basic topologies of WSNs.

2.1.2.1 Sensor node architecture

A typical sensor node is equipped with a sensing device, a processing unit, a radio module, and a power unit, as shown in figure2.2:

The sensing unit senses and converts the signal from analog to digital via the Analog-Digital Converter (ADC).

The processing unit processes and stores the data. It is the core of the sensor node and is responsible for the management of the whole platform.

The radio unit is responsible for wireless data transmissions: it transmits and receives data within the network.

The power unit provides the energy for other units. Batteries are the most common power sources for the sensor platform.

Additional elements such as GPS could be added. However, the cost of nodes has to be kept low and special attention should be paid to energy consumption in order to meet the requirements of most WSNs' applications. Thus, a trade-off must be found between the features and the deployment cost.

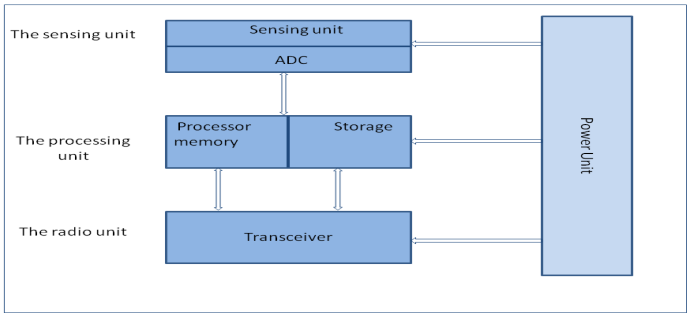


Figure 2.2: Sensor node architecture

2.1.2.2 Wireless sensor Networks topologies

Various topologies exist for wireless sensor networks deployment. In this subsection we present three well known topologies: star topology, tree topology and mesh topology.

- **Star topology** It is used mostly when there are few nodes in the network. The network consists of a central node, to which all other nodes are connected. The central node provides a common connection point for all other nodes and coordinates the traffic. Thus,the central node is the sink node which requires that all nodes in the network are within the radio range of the base station. The nodes directly transmit the gathered data to the sink without needing other nodes to act as a relay. An example of a star network is shown in figure 2.3

- **Tree topology** It is a hierarchical structure whose root is the sink. Some branch nodes (nodes that have child nodes) are denoted as cluster heads. Usually, cluster heads are one level down from the root (they are directly connected to the sink). Leaf nodes (nodes that do not have child nodes) communicate with the base station through their cluster head. Tree networks are particularly useful when the area to monitor consists of several disconnected areas. Their main advantage is their scalability. An example of a tree network is shown in figure 2.3.

- **Mesh topology** Mesh networks are the ad-hoc topology of large WSN. When a node far from the sink (not within its radio range) has data to send, the data has to be forwarded hop by hop toward its final destination. Each node has two major roles: collecting data and cooperating in order to relay data originating from other nodes. Thus, there is no hierarchy like in tree topology and all sensor nodes are supposed identical in term of functionality. The network is self-organized. It establishes routes to the sink and builds its own topology. This type of networks raises routing and optimization challenges: Finding the best routes to the sink and balancing the traffic in an energy efficient fashion are responsibilities of the network. A mesh network whose nodes are all connected to each other is a fully connected network. An example of a

mesh network is shown in figure 2.3. We rely on this kind of networks in our simulation results for all our contributions.

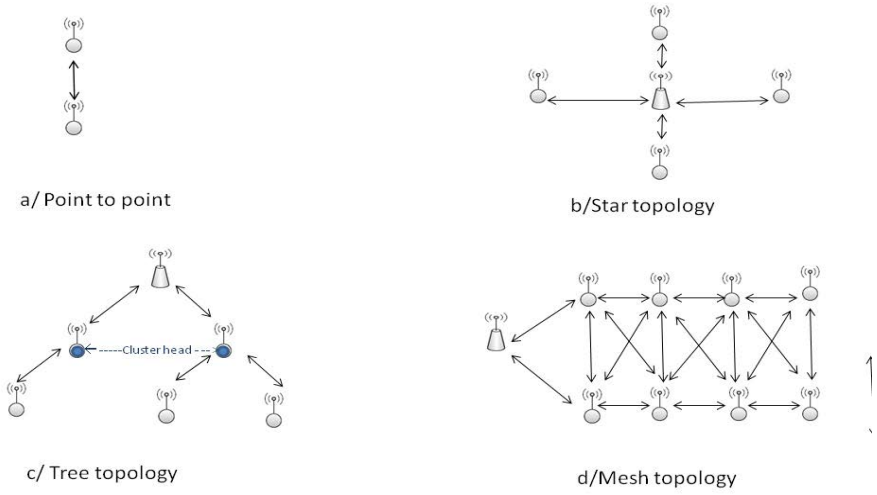


Figure 2.3: Topologies deployment of WSN

2.1.3 Applications

In the last decade, WSNs have experienced an exponential degree of research interests and a growing number of industrial applications like depicted in figure 2.4. They are characterized by easy installation and adaptive self-configuration with no need for maintenance. The application field is so wide that it is difficult to imagine a major application area that cannot benefit from WSN technology. Their great potential permits improving numerous applications of our daily life such as transportation, agriculture, industrial process control, medicine, and the military as well as creating new revolutionary systems in areas such as global-scale environmental monitoring, precision agriculture, home and assisted living medical care, smart buildings and cities. In the following section, we overview the key application field of WSNs:

- **Environmental monitoring:**

  WSNs are able to monitor environmental conditions such as temperature, humidity, chemical emissions or radiation. They enable a continuous and unattended remote surveillance of large areas. Applications for environmental monitoring have become an area of growing interest. More precisely, agriculture can considerably benefit from WSN. Indeed, sensors can optimize the irrigation and save water but also prevent plant disease. For example, water flow studies in the Alps help to enhance the water supply during dry summer periods [78]. Moreover, sensors could be used in biodiversity monitoring applications by tracking species in harsh environments. We can collect periodically and observe wildlife by placing sensors on nodes or using mobile devices.

- **Detection and tracking:**

  Detection and tracking enabled WSNs are employed within different fields of applications. It can be applied on people or objects. Detecting and tracking people is used for social applications such as crowd detection, security surveillance, or for timekeeping systems managing employees' working time, while detecting and tracking of objects is used to localize freight containers, cars or equipment in hospitals. A use-case application of car tracking is to handle traffic flows or to manage parking lots including automated parking fee accounting.

- **Health care systems:**

  WSNs are able to continuously monitor vital values of the human body such as its temperature, blood pressure and glucose level or heart activity. This enables real time patient monitoring. Moreover, it allows to rapidly detect clinical deterioration or to improve the life quality of elderly people by prolonging their time living at their own home and in an unobtrusive manner. Such sensors, placed on the patient's body and communicating with a base station, form a network able to deliver high quality care for patients while allowing autonomy and mobility. Additionally, WSNs are used in medical research to study chronic diseases or the human behavior. Another important healthcare application is the telemedecine allowing patients to be treated at home instead of the hospital. Their vital signs are continuously monitored and transmitted to medical staff which can react to emergency situations.

- **Early detection of disasters:**

  WSNs are able to detect geophysical hazards such as landslides, tsunamis, earthquakes or volcanic eruptions as well as meteorological disasters caused by extreme weather or wildfires. Early warnings are used to reduce the impact of these events on lives and property. Preventing such disasters requires an immediate response to any alarm which involves a continuous supervision of the alarm state. Sensors provide complete visibility of the resources to the administrator of the system. In building monitoring, for example, such visibility enables instant reaction to any event by transferring real-time information about the occurrence of an accident (such as a fire) permitting the evacuation of residential areas or to support firefighters. Additionally, fire detection WSN is used inside buildings to accelerate evacuation time or activate sprinkler systems.

- **Industrial Process Control:**

  WSNs are used in industrial processes for machine condition monitoring, building automation, predictive maintenance, energy management or vendor managed inventory. The ultimate goals of these WSNs are to decrease the manufacturing costs, to improve the operational reliability and to increase the health and safety of the employees. Some concrete examples are logistics applications in the stores like detecting checkpoints and locating items.

- **Military applications:**

  Battlefield surveillance is a major application of WSNs since it contributes to save the life of many soldiers. Such applications enable the surveillance of the battlefield and more precisely of the enemy unit as well as site analysis. This analysis aims to detect the presence of a dangerous entity like nuclear, biological or chemical agent. Moreover, WSNs can be deployed to form a security perimeter in order to forbid enemy intrusion. WSN for such border control application can be used to alert guards in case of invasion detection. For example, in the Vietnam War in 1970s, wireless sensor networks were deployed by the US military in the forest and used for tracking enemies. Recent progresses in these aspects can be found in [49].

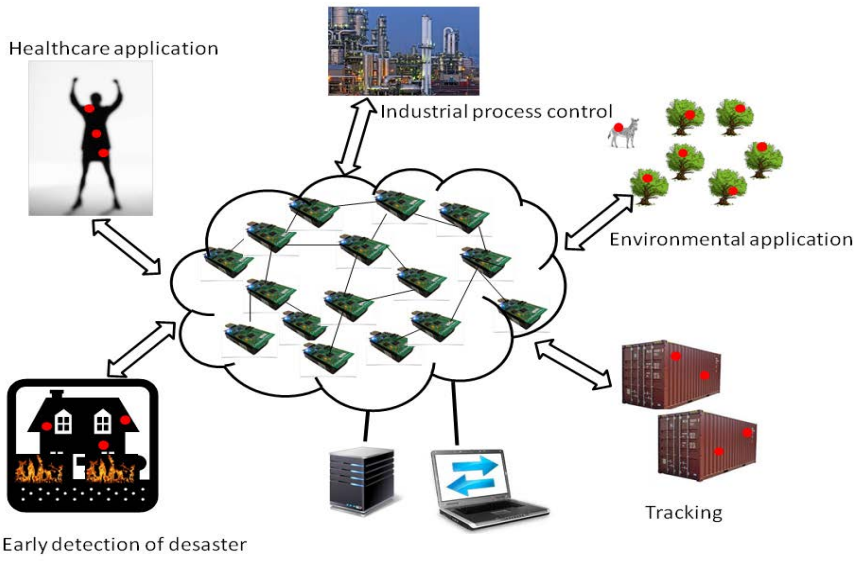


Figure 2.4: Overview of WSN applications

2.1.4 WSN Challenges and requirements

The design of WSNs has to fulfill their intended purposes thus many challenging issues have to be addressed. Despite the diversity of WSNs characteristics, some requirements emerge and are common to most of them. In this section we discuss these challenges. We begin by energy preservation and network lifetime presenting one of the major fields of WSN research. After that, we address the issues of scalability, security and mobility. Finally we discuss some additional challenges which are very important in some application areas.

2.1.4.1 Common challenges

- **Energy and network lifetime**

WSNs are often battery driven so they have to feature an autonomous power supply. Unfortunately, the lifetime of battery powered nodes is limited. Therefore, energy preserving mechanisms [20] are an important field of current WSN research. The required lifetime of a WSN depends on its intended purpose. For example, medical applications for patient monitoring have to be operational for several hours or days only, while a volcano monitoring application has to collect data during years.

Moreover, uneven energy distribution and consumption or accidental failures due to harsh environments may cause these sensors' end of lifetime [81]. These damaged nodes generate constrained region which provoke the network partitioning and loss of connectivity. Thus, the network activity is hampered. This fact is particularly true for sensor nodes surrounding the sink since the data traffic is typically concentrated there [27] which we call the hotspot problem. Consequently, those bottleneck nodes around the sink exhaust their batteries faster than other nodes which lead to the isolation of the sink from the rest of the network. This particular issue constitutes the subject of our first contribution presented in chapter 2. Therefore, the network may get partitioned into disjoint areas due to node failure or energy depletion. If the network gets useless when the first holes occur, the remaining resources are wasted. So, maintaining network connectivity is a crucial concern to ensure application operation.

To sum up, node's energy depletion may cause loss of connectivity in the network leading to the dysfunctioning of the applications. Accordingly, all protocols of the communication architecture have to be energy aware. It was shown that radio operation is more expensive in term of energy expenditure than computation [60]. Indeed the energy amount necessary to transmit 1Kb to a distance of 100 meters is the same as the amount consumed to execute 3 million instructions by a processor with 100MIPS/W. It is then necessary to design protocols with low signaling overhead. Many techniques are used to minimize energy consumption. Further examples will be given in the next section.

- **Scalability**

  Many WSN applications require large number deployment of a of nodes. This is mainly motivated by the large deployment area, the low communication range of nodes and their low cost. Added to that, the deployment of a large number of nodes provides redundancy, which help the network to prevent from node failure.

  On the other side, such deployment triggers the issue of scalability. Scalability refers to the ability of the system to adapt to an increased traffic load, handling and processing large amounts of data. Less scalable protocols may lead to severe performance degradation. Distributed algorithms may be preferred over centralized solutions in this kind of situation.

- **Mobility support and dynamic topology**

In WSNs, nodes are generally static. Nevertheless, we can make use of a mobile entity [87] depending on the application requirements. The movement of such node can be constant or sporadic. Hence, the topology changes introduce the need for dynamic routing protocols design. Nevertheless, the usage of mobile elements introduce several advantages. These include enhancing the coverage and increasing target tracking potential. Node mobility may also be an undesirable consequence of the environmental influences such as wind or animals. It could also be associated to node failure. Indeed, once a node failure occurs, we have to face the topology change and recover from this loss by restoring the network coverage. It brings up the same issues as those of a mobile node. Consequently, WSN should support dynamic topology changes in order to ensure a continuous data delivery.

- **Security**

  Security attacks [29] are problematic for WSN. The characteristics of WSNs, namely the minimal capacity, the physical accessibility, the wireless communication and the openness of the systems make it prone to security attacks. The security problem is further exacerbated by another vulnerability, which is transient and permanent random failures. To meet realistic system requirements, WSN must be able to prevent and to recover from unanticipated security attacks.

2.1.4.2 Additional challenges

- **Quality of service**

  Quality of service (QoS)refers to the level of performance required by the application and translates into network parameters such as delay and reliability. Some applications like multimedia and healthcare applications have high requirements such as high availability of service, stability of service, and low delays. Basically, these user oriented requirements translate into packet latency, throughput, and reliability concerns. If the network is not able to provide an appropriate support to these demanding applications (adequate latency, sufficient throughput and high reliability), the user experience can be degraded, or the application can even become ineffective. In order to provide QoS support [12], protocols have to implement specific, often energy consuming mechanisms. As energy efficiency is a key element in wireless sensor networks design, QoS communication protocols should balance energy efficiency and optimization of the QoS parameters

- **Localization**

  Several WSN applications require the correlation of the sensor readings with physical locations, such as tracking applications. In addition, localization benefit to various networks services such as location-aware routing, data aggregation, etc. The location information may be provided to the nodes in several ways: it may be set during the node configuration, acquired thanks to sensors, or calculated by collaborating with other nodes.

Setting the location of node may raise an issue in large scale WSN since it is possible neither to set it manually nor to have a GPS on every node because of its excessive cost. Thus, it is necessary to design efficient and energy-aware localization algorithms.

2.2 Reliability in WSN

WSNs may face a number of challenges that can hamper their widespread exploitation [44]. A network of sensors has to be self adaptive and resilient to communication errors by providing efficient mechanisms for information distribution especially in the multi-hop scenario. These requirements have to be satisfied in an architecture that can be constrained by limited processing capabilities, scarce energy resources and unreliable communication channels [51]. In particular, in a typical harsh environment, the radio signal is often affected by interferences; medium access conflicts, multipath fading, shadowing etc. These problems may result in significant packet losses.

Moreover, the success of any application (particularly mission-critical ones like life-care data and alarms) requires the delivery of high-priority events to sinks without any loss on the path from the original sources to the final destination [92].These constraints emphasize the need for reliable and robust data transport system in spite of noisy, faulty and non-deterministic underlying physical world realities. Further details about the causes of erroneous data forwarding are given in the next section. We also give an overview about the techniques for reliability used to counteract this concern.

2.2.1 Causes of Erroneous Data Forwarding in WSNs

Erroneous data forwarding in WSNs is mainly due to radio issues. In this section, we give taxonomy of the main causes.

2.2.1.1 Inter-flow and intra-flow interferences

These two types of interferences are generated by transmissions of neighboring nodes sharing the same carrier wave frequency. Packets forwarded on neighboring network paths cause inter-flow interferences, while intra-flow interferences are caused by packets forwarded on the same network path. A good example for intra-flow interferences are acknowledgement messages sent by the sink back to the sender to acknowledge successful reception of a data packet. The authors of [70] evaluated the impact of concurrent transmissions with sensor nodes equipped with CC1000 [24],radio modules. They reported that interferences caused by concurrent transmission show a significant impact on link quality and packet loss. They observed that if the signal to noise ratio (SNR) exceeds a critical threshold, the packet reception rate is over 90%. Moreover, inter-flow and intra-flow interferences are responsible for the hidden node problem [6]. Therefore, handling inter-flow and intra-flow interferences is an important task of WSN stacks supporting multi hop communication.

2.2.1.2 External-interferences

Electronic devices using wireless communication systems and operating on similar carrier wave frequencies may cause significant interferences. Many works report that IEEE 802.11 networks can cause significant interferences in WSNs using IEEE 802.15.4 compliant radio modules [56] [95] For example, the authors of [39] observed that many IEEE 802.11b nodes are not able to detect IEEE 802.15.4 transmissions. Srinivasan et al. [71] discovered that only IEEE 802.15.4 channel 26 is not affected by IEEE 802.11b transmissions.

2.2.1.3 Radio wave propagation effects

Radio wave propagation effects causes interferences generated by the sender itself [69]. Figure 2.5 depicts different radio wave propagation effects such as reflections, diffraction and multi-path fading, which can cause bit errors in the received packet. Radio wave propagation effects are difficult to influence by a link layer protocol. The most promising option is to adapt the transmission power of the radio module. The authors of [77] analyzed radio wave propagation effects in potato field. They report that the radio wave propagation was better at high humidity. This is maybe caused by changes in the reflection of the potato canopy.

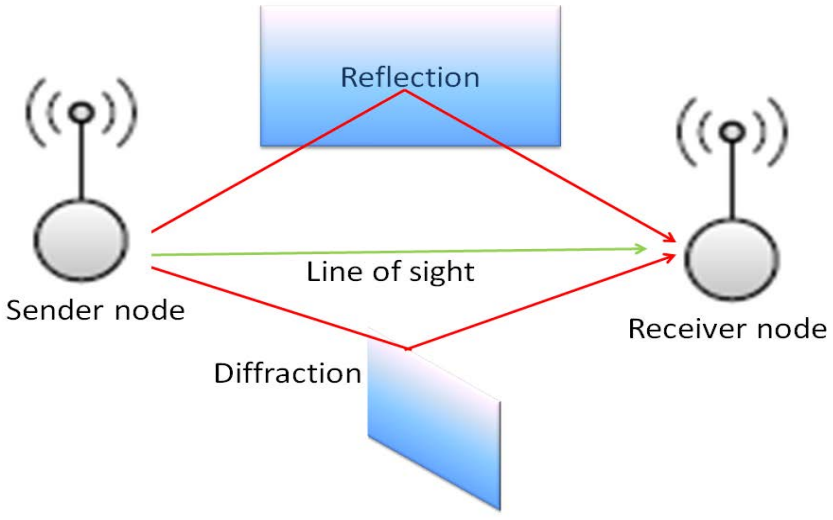


Figure 2.5: radio wave propagation effects

2.2.1.4 Packet buffer overflow

Sensor nodes in WSNs have limited buffering capacity. A buffer overflow [65] at a node occurs if the amount of data it generates and it receives from upstream nodes exceeds its transmission throughput. In this case, the receiver has to drop packets, even if they are error free. This is more likely to occur at the sensor nodes close to the sink, as they usually carry more combined upstream traffic.

2.2.1.5 Collisions

In radio communications, the transmission channel is shared among all nodes within the same radio range but only one transmission may occur at a time. Two or more simultaneous transmissions would cause a collision: both transmissions would get mixed up and not deciphered by the receiver. MAC layer [61] is responsible for handling all access to the physical radio channel. It has to determine who is allowed to access the media while avoiding collisions.

2.2.2 Overview of reliability techniques

This subsection introduces and classifies different reliability techniques [44] for WSNs to recover from packet loss.

2.2.2.1 Automatic Repeat Request Mechanism

Automatic Repeat reQuest (ARQ) mechanisms are used in hop-to-hop and end-to end retransmission based recovery mechanisms. ARQ mechanisms in WSNs make use of three different acknowledgement mechanisms to trigger the retransmission of a lost packet:

- **Explicit acknowledgments:** With explicit acknowledgments, every successfully received data packet is directly acknowledged by the receiver with a short notification message. If a sender is not able to recognize the expected notification message, then the packet is retransmitted. This mechanism offers the highest reliability guarantee. It is used by protocols providing packet reliability on hop-to-hop as well as on end-to-end level.

- **Negative acknowledgments:** Negative acknowledgment mechanisms use sequence numbers to detect packet loss. Figure 2.6 depicts an example for a negative acknowledgment. A node detects the failed transmission of the packet with sequence number n after having received the subsequent packet with sequence number n+1. Now, this node sends a negative acknowledgment to the sender to indicate the loss of packet n. During periods with low packet loss, this mechanism requires a lower amount of individual notification messages than explicit acknowledgments. Negative acknowledgments are used by protocols providing packet reliability on a hop-to-hop as well as an end-to-end level.

- **Implicit acknowledgments:** Figure 2.6 shows how implicit acknowledgments are realized. After transmitting the data packet, the sender overhears the channel to detect the forwarding of the same packet by the next node. This mechanism does not require any additional notification messages. Implicit acknowledgments show some drawbacks in WSNs. For example, overhearing of the forwarded packet requires additional energy and overhearing does not work on last hop. This mechanism is only used by protocols providing hop-to-hop reliability.

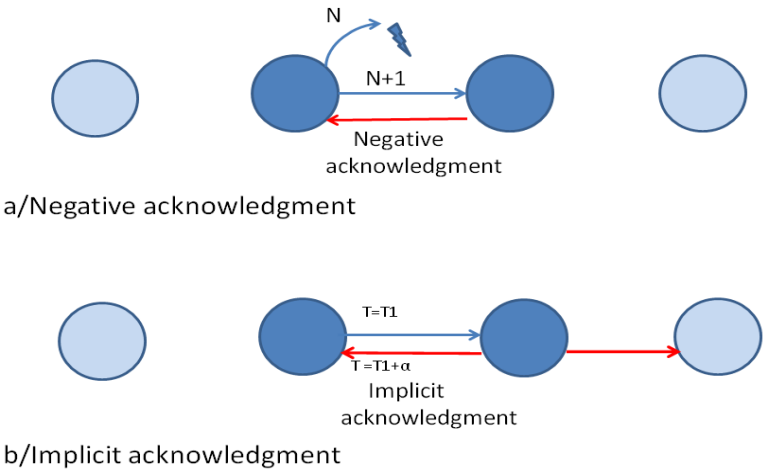


Figure 2.6: Acknowledgments scheme

The next subsection introduces redundancy based packet recovery mechanisms using Forward Error Correction (FEC). In contrast to ARQ, FEC mechanisms do not retransmit lost packets. FEC mechanisms try to recover bit errors in packets which are successfully detected by the physical layer, but incorrectly received, using redundant data.

2.2.2.2 Forward Error Correction Codes

In this subsection we introduce redundancy based packet recovery mechanisms based on FEC. FEC codes use Error Correction Codes (ECCs) [55] to add redundant information to a packet. The redundant information enables a receiver to detect and correct bit errors in the received packet. Note that the amount of bit errors that can be corrected depends on the ECC used by the FEC code. FEC codes show the following advantages and disadvantages concerning reliability and energy consumption :

Advantages	**Disadvantages**
-Reduction of intra-flow and inter-flow interferences -Restoring of corrupted received packets -Reduction of the number of transmission attempts	-Additional latency and energy expenditure for calculations of FEC -Packet size expansion -Conditional recovery(It depends on bit error on the received packets)

Table 2.1: Summary of advantage and disadvantage of FEC code

2.2.3 Overview of former proposed protocols

Reliability is an important performance metric when designing routing protocols. For paths with multiple hops, a bad link in the path can cause significant decline in the path reliability .For example if all links of a given path are reliable except the last one, the end to end reliability of the path will be very low. In order to build reliable packet delivery on top of unreliable communication infrastructures, an error control mechanism is highly required. Existing reliable transmission mechanisms for wireless sensor networks include traditional FEC, ACK, multipath transmission, and some new technologies such as network coding.

2.2.3.1 Multipath routing based protocols

The error control can be implemented as a multipath routing by forwarding packets along several paths in order to improve the overall reliability. This method allows to set up a redundancy and provides each sensor with alternative paths towards the sink enhancing the system delivery rate and reducing control cost. Several protocols were presented in the past [59] [64] combining reliability with energy efficiency or a load balancing scheme.

We distinguish two approaches for multipath routing shown in figure 2.7: disjoint path and braided multipath. A disjoint multipath algorithm is used to build independent paths called main paths while some braided paths called logic paths are built on each main path. For braided multipath model, each intermediate node has a backup node. Performance results [97] show that, for a single node failure, the braided alternative path is 20% more resilient than the node disjoint alternative path. Whereas, for multiple failure the resilience of both approaches is comparable. Besides, the braided alternative path approach is more cost effective.

Although multipath routing can increase reliability of transmission, maintaining multiple paths is usually costly in large scale WSNs and bandwidth-consuming applications like audio/video streaming.

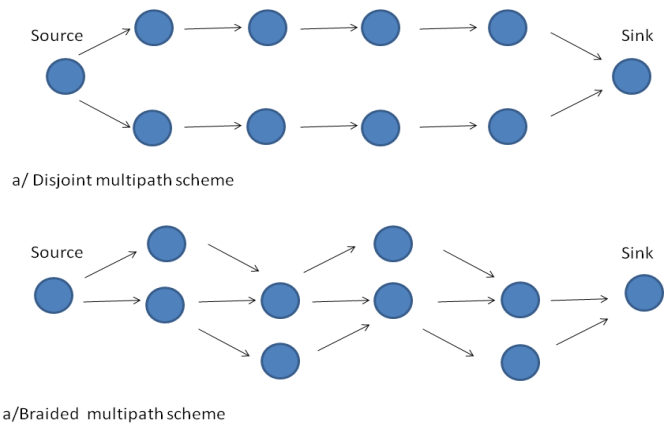


Figure 2.7: Topologies of disjoint and braided multipath schemes

2.2.3.2 FEC based protocols

FEC gives the receiver the ability to recover lost packets and retrieve the original data at the cost of a higher forward channel bandwidth (depending on the amount of redundant data). Authors in [50] propose to code a packet first and then split it into a few smaller packets. These smaller packets are then transmitted over multiple routes to the destination. Finally, the destination reconstructs the original packet from the received packets. Since coding techniques are used, the destination node can tolerate packet loss in some extent. Authors in [93] proposes a lightweight coding algorithm combined with a fault tolerant routing scheme in WSN. The coding-decoding algorithm is based on XOR operation and requires little computation and storage space, which are critical for WSN. In spite of their interest ,the static determination of the FEC code size degrades their performance since the evaluation of the underlying channel state is hardly accurate and even widely varied. In this context, authors [94] proposed an Unequal error protection for WSN.

2.2.3.3 ARQ based protocols

Another traditional way to achieve reliable transmission is the Automatic Repeat reQuest (ARQ) mechanism [74]. However, the number of ARQ should be minimized because sensor nodes are severely resource constrained and data transmission is one of the most costly operations performed by sensors [74]. Moreover, the unreliable radio channel affects the acknowledgment delivery as well. If the sender does not receive any acknowledgment in the specified time interval, it retransmits the message even if the packet was properly delivered. In practice, the sender node makes a delimited number of trials to successfully deliver a message. Therefore, relying on explicit acknowledgment may not be that efficient with regard to the constrained nature of WSNs.

We could identify two categories of transmissions: the Hop By Hop (HBH) and the End to End approach (ETE). According to She and al.[67], HBH is more energy efficient at the cost of large transmission delay compared to ETE. Nevertheless, HBH outperforms ETE on the delay metric for high bit error rate cases. Given that error rates of 10% or above in dense wireless sensor networks may be experienced, HBH is the most suitable candidate for WSNs [100]. Let's notice that the problem with ETE recovery is highly related to the harsh radio environments and to the multihop forwarding techniques to exchange messages which favor exponential error accumulation over multi-hops [100].

Some researches proposed solutions to alleviate the retransmissions cost like Pump Slowly, Fetch Quickly (PSFQ) [11]. PSFQ is based on slowly injecting packets into the network (pump operation) and performing hop-by-hop recovery in case of packet losses (fetch operation). This protocol is efficient for fast recovery but if packet loss occurs in an intermediate node, buffer must be maintained until missing packet re-transmission is done. This causes buffer overflow and increases data transmission delay. Authors in [9] presented a selective acknowledgment mechanism switching between explicit and implicit acknowledgment depending on the current path reliability. For this solution path reliability is determined by measuring the RSSI (Received Signal Strength Indicator) which is proved not to always be a good

indicator to estimate the link state [88]. Furthermore, the algorithm execution generates an additional delay while searching for a reliable link to ask for an ACK. The protocol defined in [47] achieves reliability through cashing and retransmission. As mentioned previously, this solution requires each one hop neighbor to cash the data until the success of its transmission. This leads to both extra buffering and to a risk of memory overflow. In addition, cashing data by neighboring nodes will involve the modification of the MAC layer protocol since caching data is not a feature of the MAC 802.15.4.

2.2.3.4 Network Coding

Considering the topology and the constraints of WSN, the network coding provides lots of benefits for such networks. It will not only reduce energy consumption but also enhance their reliability and their robustness [85]. Network coding relies on the "butterfly network" scheme depicted in figure 2.8.

In traditional retransmission approach, each lost packet is retransmitted separately to the sink. Xiao et al. [91] proposed a wireless broadcast retransmission approach aiming to reduce the total number of transmissions by combining lost packets with network coding.

As stated above link failure highly impact the reliability of the WSN. The sporadic link failure may be temporal or may last for a considerable amount of time. In the latter case, failure cannot be relieved using FEC or ARQ. Authors in [42] provide protection against such link failures using network coding. Moreover, NC can enhance the robustness of WSN by storing coded packets in distributed network storage as proposed in [84].

Despite the opportunities it offers, network coding is not easy to deploy in the random topology of WSN. Thus, special kind of deployment policy needs to be followed in order to benefit from its advantages.

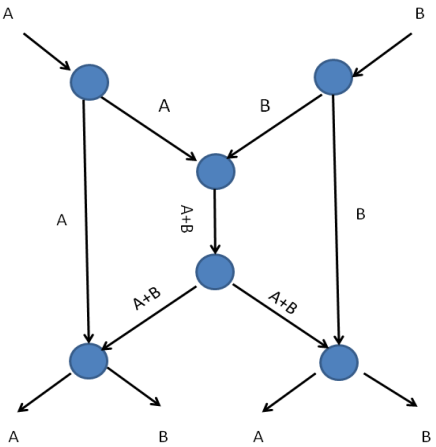


Figure 2.8: Network coding scheme

2.3 Conclusion

In this introductory chapter, we provided a general overview of WSN technology by presenting its applications areas. Based on its applications requirements, we derived and discussed its main challenges. The second part of the chapter was dedicated to the reliability concern of WSN. We highlighted the major causes of packet loss in WSN then we gave an overview about the main techniques providing reliability. To conclude, a reliable transmission protocol fulfilling the challenges of WSNs is highly required especially in specific applications like emergency ones. In the remainder of this thesis, we will focus on the withstanding of two main issues: node failure and packet loss.

Chapter 3

Fast connectivity restoration in WSNs

Node failures represent a fundamental problem in wireless sensor networks. Such failures may result in partitioned networks and loss of sensed information. In this chapter, we propose a scalable and distributed approach for network recovery from node failures in wireless sensor networks called CoMN2 (Connectivity using a Mobile Node in a Mapped Network). It relies on a new concept called network mapping that consists in partitioning the network into several regions of increasing criticality. The criticality is set according to the energy, the traffic distribution and the deployment of nodes. Using this network mapping, our solution CoMN2 seeks into maximizing network lifetime by efficiently swapping nodes from low critical area to highly critical area when required. Also, we compare our protocol to solutions from the state of the art and we evaluate it through extensive simulations.

3.1 Introduction

Uneven energy distribution and consumption or accidental failures due to harsh environments may cause sensors to shut down arbitrarily [89]. These damaged nodes generate constrained region which provokes network partitioning and thus hamper network activity. On the other hand, the data traffic is typically concentrated on the sensor nodes surrounding the sink [26][82]. Consequently, those bottleneck nodes around the sink exhaust their batteries faster than other nodes that lead to the isolation of the sink from the rest of the network. Therefore, the network may get partitioned into disjoint areas (referred to as holes) due to node failure or energy depletion. If the network gets useless when the first holes occur, the remaining resources are wasted. Consequently, maintaining network connectivity is a crucial concern to ensure application operation.

In this chapter, we exploit the non-uniform consumption, traffic distribution and deployment of wireless sensors to elaborate an efficient dynamic recovery approach. We present CoMN2 a protocol restoring WSNs Connectivity using a Mobile Node in a Mapped Network. CoMN2 introduces a new concept called Network Mapping which consists in partitioning the network in several regions of increasing criticality.

For instance, a dense area with low sensing activity will have lower criticality than nodes in a sparse area highly solicited.

Through this network mapping, our solution CoMN2 ensures the continuous network activity by swapping nodes from low critical area to high critical ones when required. The swapping is performed using a mobile node which is aware of the sensor status. Basically, the algorithm is triggered depending on the criticality of the failed node in our mapped network. This solution minimizes the energy consumption by inhibiting the recovery process as long as the damaged node is not vital for the network. Above and beyond that, CoMN2 proceeds to a periodic network mapping update in response to unpredictable network dynamics.

CoMN2 design fulfills the following points that we consider as requirements for our context:

- In view of the autonomous and unsupervised characteristics of WSN, the network recovery from node failure should be performed in a distributed manner.
- The process should be rapid and lightweight to preserve the WSN responsiveness to detected events.
- The overheads should be minimized in order to consider the scarce resources of WSNs.

The remainder of this chapter is organized as follows: the next section gives an overview of the problem statement. The details of the algorithm are given in section 3. Then, section 4 provides the protocol analysis and simulation results. Finally section 5 concludes this chapter.

3.2 Problem Statement

Maintaining inter-node connectivity is a crucial concern in WSNs. Emergency applications like disaster management require highly reliable nodes collaboration to efficiently assess damage and identify safe escape paths. This fact emphasizes the importance of the node connectivity paradigm. Basically, the failure of a sensor node in a critical region may create *holes* 3.1 and induces a partitioned network, and so a reduction of the network operation efficiency. A particular case of this point is the HotSpot problem: Sensor nodes around the sink are extremely solicited to forward the sensed data (from the entire network nodes) toward the sink. In spite of the large deployment of sensors to tolerate possible node failures, we have to face the problem of isolation of the sink node caused by the depletion of the energy of surrounding sensor nodes. This may result in the failure of the whole network, whereas the remaining nodes may be perfectly functional.

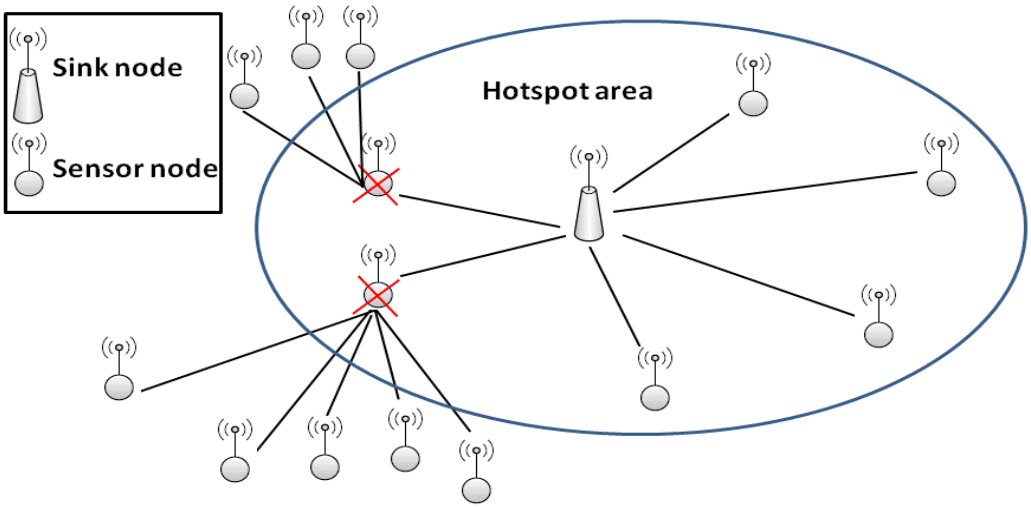


Figure 3.1: Node isolation in the hotspot area

On the other hand, sensor nodes activity may be non-uniform in a hybrid network. For instance, some nodes are expected to monitor a given parameter (e.g. temperature, humidity, etc.) whereas other nodes may be further involved in relaying data toward the sink. This non-uniform activity distribution depletes the battery of some sensors faster than the rest of the network. Nevertheless, in a high-density sensor networks, some areas are monitored by an excessive number of sensors. If two nodes are very close to each other, there is a high probability that they produce similar data [89] Table 1 3.1 presents the relationship between the number of neighbors and the percentage of redundancy. We define a sensor whose sensing area can be covered by its 1-hop neighbors completely as a *completely redundant sensor*. The Percentage of the redundant area is defined as the percentage of a sensor's sensing area covered by its random neighbors. As shown below, with 11 neighbors, the probability of complete redundancy is almost 92.28%, and the percentage of redundant area exceeds 99%.

Number of neighbors	**Probability of redundancy**	**Percentage of the redundant area**
5	31.22-37.01%	91.62%
7	64.29-65.21%	96.89%
9	82.97-83.09%	98.85%
11	92.28%	99.57%

Table 3.1: Number of neighbors Vs Percentage of redundancy extracted from [89].

Thus, the loss of a sensor in a redundant area does not impact the rest of the network because its neighbors are capable to achieve exactly the same role.

Keeping in mind all these aspects, our problem statement can be formulated as follows: given a set of nodes,

- How can we recover from node failure?
- When should we replace the damaged node and which nodes are vital for the functioning of the network?
- How can we handle unpredictable dynamics of the networks?

To deal with the first issue, CoMN2 proposes to recover from node failure by swapping nodes using a mobile node. This solution raises another issue which is the selection of the node to move. This problem is resolved thanks to our new concept: the network Mapping. Upon the detection of a damaged node, CoMN2 considers four characteristics for each node:

- The battery level
- The number of neighbors
- The location of the node (Hotspot area or not)
- The activity of the node

These issues are summarized in the ZoneAffectationTable which will be presented in the next section. Depending on the value of these parameters, the network is mapped in three zones of increasing rank expressing the criticality of sensors of each zone. This mapping is periodically updated to consider the network dynamism. The detailed CoMN2 algorithm is explained in the next section.

3.3 CoMN2 Operation

3.3.1 Overview of the mechanism

In the following scenario, we are considering a hybrid WSN with static sensors, along with a sink and a mobile node. The mobile node is not as energy constained as other nodes (e.g., a mobile robot) equipped with high processing capabilities and longer battery life. We assume that the sensors are randomly scattered over the network and that every node is aware of its location and its remaining energy.

The communication between sensors and the mobile node is performed via the sink. All the control messages (node failure alert) are initially sent to the sink which transfers them to the mobile node as shown in figure3.2. We assume that the mobile node notifies the sink of its new position each time it moves.

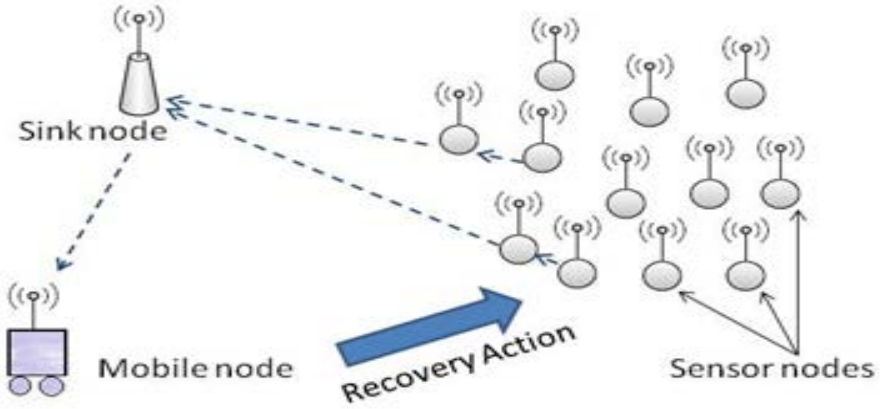


Figure 3.2: Nodes communication

As mentioned above, the crash of some nodes may stir up the division of the network into disjoint segments and leads to the formation of holes. To avoid this, our solution uses a mobile node carrying out the network restoration by switching a failed node with another from a redundant area (less critical). Furthermore, our solution handles simultaneous sparse node failure by classifying nodes according to their importance through an innovative process called network mapping (partitioning). This method consists in organizing the network into three zones of increasing rank depending on the criticality of nodes belonging to each area. The detailed mechanism will be presented in the following part.

3.3.2 Network Model

Our idea of mapping the network depicted in figure3.3 emanates from the observation that sensors have no uniform activity in the network as explained in section II. The key plan is to hierarchically divide the sensor field into zones of increasing importance according to the nodes activity.

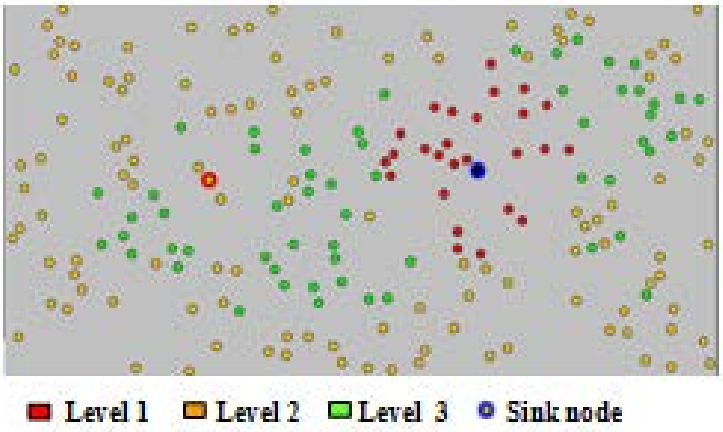


Figure 3.3: The network mapping

In fact, the more the node is solicited, the faster its energy depletes and the sooner it will die. On the other side, some high density zones are monitored by excessive sensors, which means that the failure of a node of that zone does not impact its functioning as stressed in section 3. That is why we decide to give priority to connectivity in specific zones instead of favoring connectivity of the entire network.

We divide the network into three zones of growing rank. Our basic assumption is that the nodes around the sink are the most active [26][82] and that they belong to the first zone. Since we are considering a heterogeneous network with nodes assigned to diverse applications, some nodes may be very solicited for sensing or reporting operations. In view of their importance, the sensors of these specific zones belong to the first zone too. We define the third zone as the less critical one containing the set of nodes with more than K neighbors (section 3). In the simulation phase, the K value is set to 10. The remaining nodes constitute the second zone. Finally we obtain a network's cartography based on how critical is each zone. This network cartography will be dynamic: it will be updated contingent on the activity of nodes.

3.3.3 Protocol Operation

3.3.3.1 Initial assignment of nodes through zones

The first affectation of nodes through zones exploits their position and their neighborhood repartition. The mobile node will maintain a table referred to thereafter as ZoneAffectationTable table 3.4 containing five parameters *{Id, Pos, NN, RZ, Zone}* where:

- *Id* is a unique identifier of each node.
- *POS* designates the local position of the node.
- *NN* is the number of neighbors.
- *RZ* is a binary parameter designating risky zones: 1 for risky one and 0 for no risky. Initially, RZ is set to 0 for all nodes. It will be set to 1 during the update process when the mobile node will notice the high activity of some nodes.
- *Zone* indicates the zone of affection. It can be 1, 2 or 3 with 1 designating the most critical zone.

Id_1	POS_1	NN_1	RZ_1	$Zone_1$
Id_2	POS_2	NN_2	RZ_2	$Zone_2$
Id_i	POS_i	NN_i	RZ_i	$Zone_i$
Id_n	POS_n	NN_n	RZ_n	$Zone_n$

Figure 3.4: ZoneAffectationTable design

To establish ZoneAffectationTable, each node should send NN and Pos to the sink which will relay it to the mobile node as shown in figure 3.5. The whole process is kept local by requiring every node to maintain a list of only its close neighbors: we restrain our list to the 1-hop and 2-hop ones only.

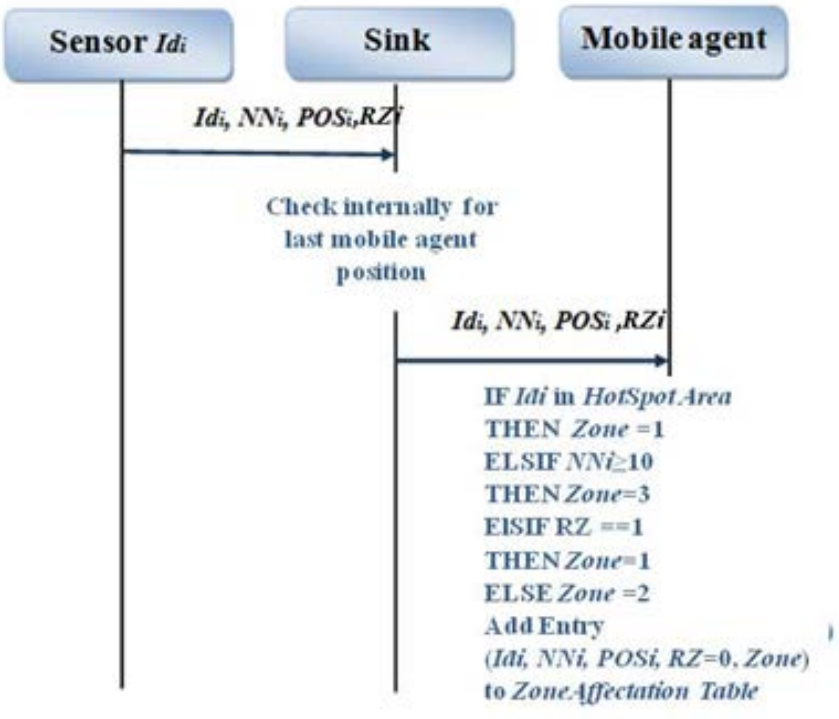


Figure 3.5: Message exchange for zone affectation establishment

After receiving this information from nodes, the mobile proceeds to the zone affectation depending on the number of neighbors around each node:

$$zone = \begin{cases} 3 & \text{if } NN \geq 10\,, \\ 2 & \text{otherwise}\,. \end{cases} \tag{3.1}$$

Nodes around the sink constitute the special case of the *HotSpotArea*. They are automatically affected to zone 1.

3.3.3.2 Update of the network mapping

CoMN2 pursues a dynamic approach by updating the mapping of the network throughout the exchange of specific messages between nodes. This reorganization involves the refresh of the *ZoneAffectationTable* figure3.6 and more precisely of NN and RZ.

NN is kept up to date by sending periodic *HEARTBEAT* messages between neighboring nodes. The absence of *HEARTBEAT* response from a neighboring node F means that F failed then a message is automatically sent to the sink which will

notify the mobile node. Upon receiving this notification, the mobile node modifies the value of the corresponding *NN* in the *ZoneAffectationTable*.

Furthermore, we define *Tobs* as the time that the sink waits before sending a periodic activity report to the mobile node for analysis. This report aims at determining the most solicited nodes by quantifying the number of operations (including sensing and routing) per node: each sensor is equipped with a counter that is incremented after every action like relaying or sensing data. The result of this counter is periodically sent to the sink. Keeping in mind that this report must characterize the network and is used to identify the most active zones, special attention is paid to well choosing Tobs which should be long enough. Once the account is relayed by the sink, the mobile node calculates the average number of operations per node (the mean value) and sets *RZ* to 1 for the most solicited nodes: with a number of operations greater than the mean value. Thanks to this parameter, a node initially belonging to zone 3 can be classified as zone 1 if it's highly active. Additionally, as soon as a node's energy drops below a critical level, it sends a *DISTRESSMESSAGE* containing its ID to the mobile node.

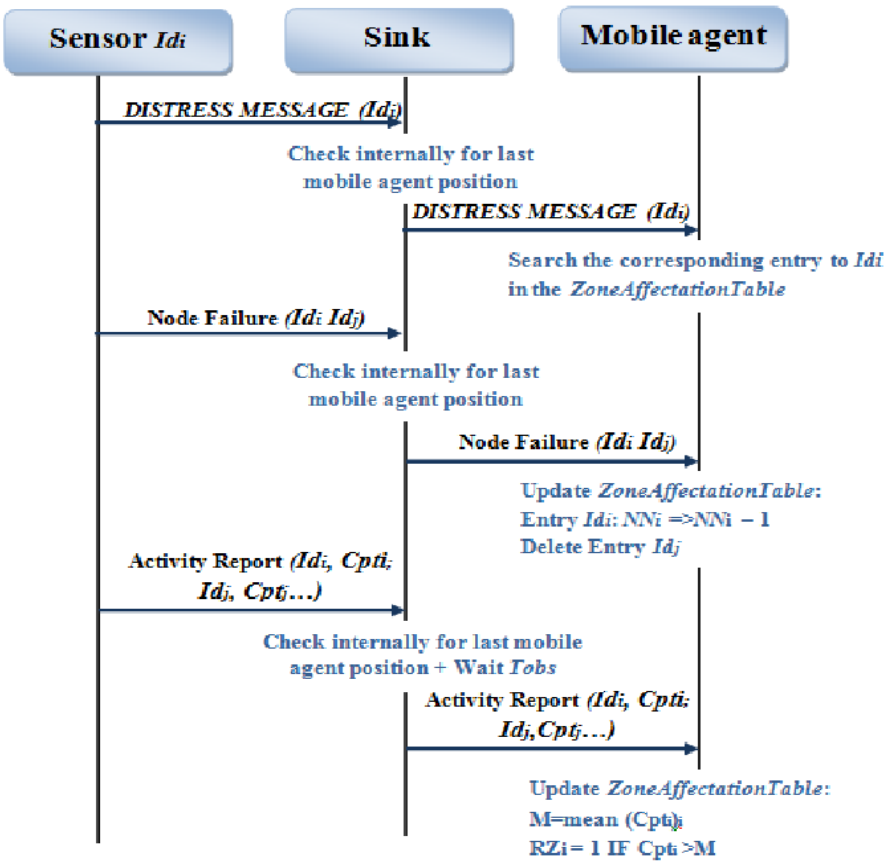


Figure 3.6: Message exchange for th eupdate of the network mapping

3.3.3.3 Initiation of the CoMN2 recovery process

To expound how CoMN2 Algorithm performs, let's consider these two cases with a first classical one of a single node failure alert message and a second case showing how our proposed algorithm handles simultaneous node failure. Before giving rise to the recovery process upon detecting a failure, the mobile node waits for *Thold* seconds to check if there is another alert.

Single Alert:

For the case of single alert, the mobile node should decide whether to initiate the recovery or not. This choice is based on the zone to which the node belongs. Actually, the network mapping aims to avoid restoring connectivity at the cost of additional energy consumption when we can do without this node: The protocol tussles to minimize the movement of the mobile node in order to save its energy. In fact, despite its high battery compared to the sensors, the energy of the mobile node is not unlimited. Execution of restoration is triggered only if the node belongs to zone 1 or 2 which are the most important regions. If the dying node is from zone 3, we suppose that it has a sufficient number of close redundant neighbors to replace it. The mobile node checks at first *ZoneAffectationTable* to determine the Zone of the failing node then decide to rescue it or not.

Simultaneous Alert:

Let's consider the case of two alerts received during *Thold* from node A in zone 1 and from node B in zone 3. The mobile node proceeds to the comparison of the zone of each node using *ZoneAffectationTable* then decides to rescue the node with the most important zone (which is node A in our example). If all the alerts arise from zone 3, no recovery process is triggered because the failure of these nodes does not impact the efficiency of the network application. The case of simultaneous alerts with the same importance is handled by a FIFO mechanism (First In First Out). The mobile node continues its current task then deals with the second affected node.

3.3.3.4 Recovery execution

The recovery solution is based on the swap of nodes and the replacement of the broken ones with the help of the mobile node. For the beginning of the network recovery approach, the mobile node exploits the Network Mapping to find the most suitable node to move. The choice of the best candidate is based on two criteria:

- A minimum change in the network topology. This condition can be achieved by choosing a node from a redundant area (zone3) which will not involve the reorganization of the network.

- The nearest one, given the position of the failed node and the actual position of the mobile node in order to minimize the overhead of the recovery process. The mobile node searches the closer node of zone 3 in the *ZoneAffectationTable* given the position of the failed node and its current position. This way, the inter-node connectivity is re-established without involving a change of the network topology. Once the spare node reaches the failed node location, it becomes responsible for carrying out its tasks including both routing and the

application level aspects. The choice of the optimal path to reach the failed node is out of the scope of this chapter.

3.4 Performance evaluation

3.4.1 Protocol analysis

How to deal with node failure is a challenging concern that has been widely studied in the past years. For instance, [38] proposed a power extension algorithm that consists in the increase of the transmission power to extend the communication range so that a node can reach further nodes when its next hop fails. This algorithm handles the connectivity aspect but does not provide a solution to ensure the sensing function of the failed node. Another proposed approach in [4] uses the neighbor information to construct a new path and take the routing role of the failed nodes. This work assumes that sufficient number of neighbors surround the failed node. Yet, this is not always true because of the random deployment of sensors in some scenarios such as hostile environments and hazardous zones.

Many researches investigate the use of mobile nodes [80] [7] [75] [15] to restore connectivity in mobile networks. In [41], the author suggests to use a mobile base station in order to load balance the charge around the sink by changing the nodes located close to it. Unlike our proposed approach, this algorithm restrains its efficiency on nodes surrounding the sink and involves an extra message overhead to update the location of the mobile base station. Another solution referred to as Coverage Conscious Connectivity Restoration algorithm [75] (C3R) relocates one or multiple neighbors of the failed node to recover from the damage. Each neighbor temporarily moves to substitute the failed node what leads to intermittent connectivity. The assumptions in the above work are strict because there is no guarantee to have multiple neighbors available.

Controlled mobility is a framework operating in context aware devices mobile with intelligent ability permitting to determine their future location on the basis of some conditions like battery level or the amount of data gathered. Authors in [15] exploit controlled mobility to propose an algorithm for sensor nodes relocation. The algorithm assumes WSNs with a cluster topology and allows relocation of sensor nodes between clusters based on predefined utility function. Despite its interest, this approach arises clustering limitations such as cluster head deployment and placement.

Our proposed algorithm CoMN2 is different from the conventional recovery algorithms since it assumes that the sensors are static nodes while the overhead related to the recovery operations (i.e., recovery decision and mobility operations) is limited to the Mobile Agent. Given that the deployment of such entities with special capabilities is expensive, and for the sake of ensuring a low protocol complexity, we opt for using only one mobile agent unlike other protocols [75]. Besides, this implicates fewer constraints on the network topology. To summarize, the table below (Table 2) compares our solution CoMN2 with other recovery approaches. Our study shows that CoMN2 involves less topology change compared to other approaches that also

use sensor relocation to substitute the failed node (i.e., C3R and the cluster solution (CLS) [15]). In addition, the mobile sink approach described in [41] assumes a permanent relocation of the sink which implies a permanent topology change of the network.

Furthermore, since we use only one mobile agent, the deployment cost is lower for CoMN2 because C3R and CLS assume that all nodes are mobile. Besides, in CoMN2 there is no communication overhead due to node mobility since only the sink is notified about the position of the mobile node. On the other hand, given that in C3R the network recovery is localized in the neighborhood of the failed node, the distance travelled by a recovering node is, by essence, optimized. This enables to reduce network recovery delays and minimize battery consumption on the recovering node. Nonetheless, although the battery constraint of the recovering node is not an important concern in our solution, the network recovery delay remains a problem that will be addressed in a future work.

3.4.2 Experimental evaluation

Besides the comparative analysis of our protocol, we conducted experiments to evaluate its performances. This section describes the simulations to assess the effectiveness of the proposed approach. We compared CoMN2 to its competitor the Mobile Sink solution and to a basic topology without restoration.

3.4.2.1 Baseline approach

We use two metrics to evaluate the performance of CoMN2: the network lifetime and the failure rate.

- The network lifetime is a generic term depending on the considered scenario. In our case, we define it as the time for the first node to become unable to reach the sink (no route available because of the failed nodes). Our goal is to increase the network lifetime.

- The failure rate indicates the rate of failed nodes. It serves as a measure of the fault tolerance of the network: The higher the failure rate is at the end of the network life, the more fault-tolerant the network is.

A java simulator has been developed to evaluate CoMN2. The simulation experiments involve randomly generated WSN topologies with varying number of nodes from 100 to 800 in a network field with dimensions of 10000mx20000m. The location of the sink is selected randomly. For each topology set-up (network size, protocol) simulation is run 10 times and the average performance is reported. During each run, packets are generated and sent successively. The sending sensors are randomly selected and the path to the sink is created using least-cost routing in terms of number of hops. We assume the energy distribution is uniform in the network: all nodes have initially the same battery level. Nodes' battery-level decrease at each operation until attaining a critical threshold equivalent to 20% of the initial battery level. Once the threshold reached, CoMN2 is activated if the node belongs to zone 1

or 2 and the node is considered as failed. As explained in section 4, many researches propose solutions with Mobile Sink. Our simulation uses a configuration where the sink moves periodically in the network to load balance the traffic. It chooses a location where nodes in the neighborhood (nodes around the new position of the sink) have a high level of battery available.

3.4.2.2 Performance comparison

We conduct different experiment by varying the network density. To test the scalability of our approach, we increased progressively the number of nodes for a fixed network field dimension and transmission range. As Figure 3.7 shows, our algorithm significantly increases the network lifetime compared to both the basic configuration (without restoration) and to the Mobile Sink approach for all the network sizes. This increase in network lifetime is more significant in the case of dense networks. This is attributed to the fact that the number of the back-up nodes used to replace the failed nodes in CoMN2 is higher in a dense network.

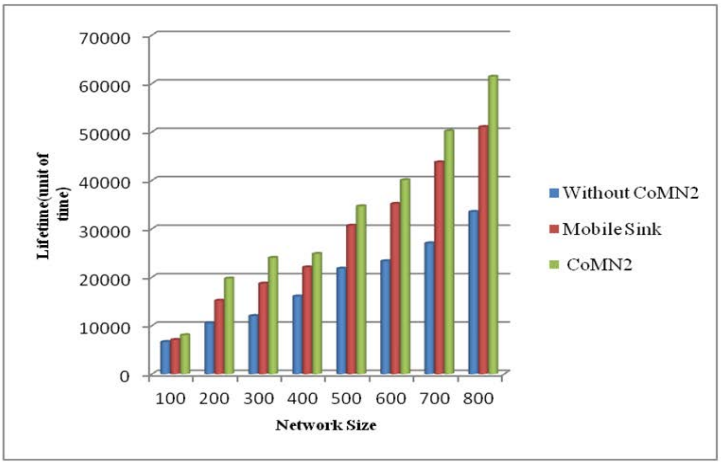


Figure 3.7: Network Lifetime vs. Network size

Table 3.2summarizes the increase of the network lifetime for the different configurations. We observe an average increase of more than 11% compared to the Mobile Sink and 40% compared to the network without CoMN2. This result demonstrates the effectiveness of our approach.

Network Size	% No CoMN2	%Mobile Sink
100	17.41	12.19
200	46.81	23.20
300	50.04	22.15
400	35.43	11.19
500	37.10	11.54
600	41.79	12.23
700	46.15	12.83
800	45.46	16.93

Table 3.2: Percentage of the network lifetime increase with CoMN2

Moreover, we tested the performance of CoMN2 in term of fault tolerance. Figure 3.8 shows that the network fault tolerance is considerably high in CoMN2 compared to the other approaches. Indeed, in CoMN2 the network tolerates a percentage of failed nodes of about 35 % versus approximately 10 % without CoMN2.

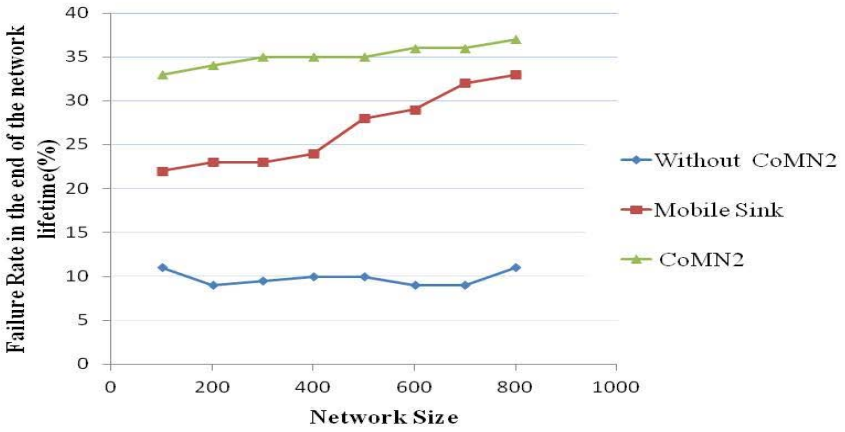


Figure 3.8: Fault tolerance vs. Network Size

This increase is due to the different location of the failed nodes as shown in Figure 3.9. We notice that unlike in CoMN2, failed nodes are scattered around the sink for the first figure with a basic approach which shortens the network lifetime and provokes the hotspot problem.

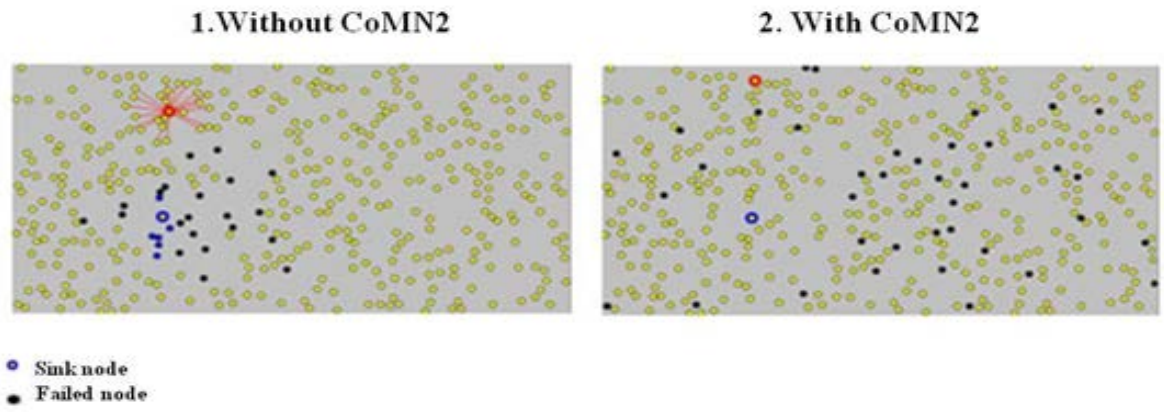


Figure 3.9: Failed node location for the same failure rate

In fact, the curves of failure rate vs. time for a network with and without CoMN2 follow the same slope in the beginning of the simulation as shown in figure 3.10. Nevertheless, CoMN2 maintains nodes around the sink alive by swapping them with nodes far from it. This increases the lifetime of the network. However, the hotspot area around the sink remains vulnerable to node failure (highly solicited) which increases its failure rate hence the higher slope of CoMN2 from a certain delay equals to 500 ms.

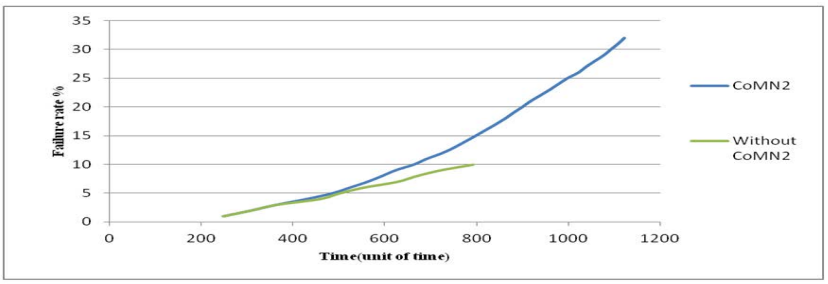


Figure 3.10: Failure rate vs Time

3.5 Conclusion

In this chapter, we dealt with the challenging issue of providing network recovery in wireless sensor networks. To do so, we presented a distributed and dynamic recovery protocol CoMN2 which handles node failure in WSNs. This protocol introduces a new class of Network mapping based on node resources. Such mapping represents an attractive paradigm in the design of wireless sensor networks. In particular, this network mapping is used to prevent network partitioning that may be

caused by a node failure. Concretely, we show that by using CoMN2 in a WSN with 600 nodes spread over 10000m x 20000 m, the enhancements yield a 40% improvement in packet delivery performance over the baseline. For the future work, we plan to reduce the network recovery delay of our approach by investigating the optimal routing path to follow in order to join the sink.

Chapter 4

Adaptive Joint mechanism with Implicit Acknowledgments for WSNs

The success of WSNs applications is contingent upon the reliable delivery of high-priority events from many scattered sensors to one or more sink nodes. In particular, WSNs have to be self-adaptive and resilient to errors by providing efficient mechanisms for information distribution especially in the multi-hop scenario. To meet the stringent requirement of reliable data transmission , we propose a lightweight mechanism for packet loss recovery and route quality awareness in WSNs called AJIA (Adaptive Joint mechanism with Implicit Acknowledgments). In this chapter, we use the overhearing feature characterizing the wireless channels as an implicit acknowledgment (ACK) mechanism. In addition, our protocol allows an adaptive route selection by delivering the packet using the most reliable link. Moreover, we outline the compatibility of our protocol with the IEEE 802.15.4 standard.

4.1 Motivation

WSNs may face a number of challenges that can hamper their widespread exploitation [92]. A WSN has to be self-adaptive and resilient to errors by providing efficient mechanisms for information distribution especially in the multi-hop scenario. These requirements have to be achieved in an architecture that is constrained by limited processing capability, scarce energy resources and unreliable communication channels [58]. In particular, in a typical harsh environment, the radio signal is often affected by interference; medium access conflicts, multipath fading, shadowing etc. These problems may induce significant packet losses in the WSN. Moreover, the success of many applications (particularly mission-critical ones like life-care data and alarms) require the delivery of high-priority events to sinks without any loss from the original sources to the final destination [35] [14].These constraints emphasize the need for an energy-efficient, scalable and reliable data transport system.

Data retransmission has been considered as one of the most common schemes [18] [74] for improving transmission reliability in wireless sensor networks. ACK/-NACK messages are the basic method used to assess the necessity of retransmission. Nevertheless, such a method generates an extra traffic causing an additional over-

head which is not suitable in a highly constrained and error prone environment like WSNs.

In essence, an alternative solution should be found to deal with retransmissions without wasting bandwidth. In this chapter, we present a reliable and energy-efficient joint mechanism for packet loss recovery and route quality evaluation in WSNs. In this protocol, we use the overhearing feature, characterizing the wireless channels [35], as an implicit ACK mechanism. In addition, our protocol allows a dynamic and adaptive route selection based on a link state metric which we will present in the next section.

Our protocol has the following benefits:

- Straightforward with minimum requirement of changes in the MAC layer.
- Minimum signaling thereby reducing the communication cost (energy +overhead).
- Selective recovery allowing responsiveness to high error rate under faulty conditions thanks to the use of the most reliable link to retransmit the data.

The remainder of this chapter is organized as follows: the next section highlights the need for reliable data delivery in WSNs, and reviews solutions aiming at providing it. An overview of the solution and the network model are given in section 3. Section 4 presents the different steps of the protocol design and finally section 5 concludes this paper.

4.2 Background

Since AJIA is a mechanism that combines both reliable packet delivery and routing metric, there are two lines of related works: error control mechanisms and routing metrics. Given that a detailed overview of error control mechanism was given in the first chapter, this section will only deal with routing metrics.

The key issues of designing a routing protocol for wireless sensor networks are energy efficiency, robustness to environmental changes and end-to-end reliability of data delivery [68]. The primary routing metric proposed was the Hop Count. This metric designates the best path as the one with lowest number of hops. In practice, the Hop Count metric does not achieve good performance because it does not consider the intrinsic characteristics of wireless links, as it considers all network links to be similar. Actually, the quality of a wireless link depends on several parameters, such as link length, path loss, shadowing and interference sources. Therefore, different wireless links tend to present different levels of quality in term of reliabiliy.

Many research have been conducted in the literature on this problem. Some existing solutions aim to improve packet delivery ratio performance by dynamically evaluating characteristics of links. The first work,called Expected Transmission Count (ETX) metric [52], characterizes the number of MAC transmissions required to successfully transmit a packet between two nodes. The Expected Transmission Time (ETT) metric [17] aims to the minimization of the end-to-end delay by associating the cost of each link to the product of its ETX and the transmission delay

of a packet (using the links' current transmission rate). The Minimum Loss (ML) metric searches for the minimum end-to-end packet error rate (PER) by choosing the route with the lowest end-to end loss probability.

Although the previous metrics consider different approaches to quantify the quality of wireless links, they are all based on the same paradigm: broadcasting periodically control messages to retrieve statistical data from links. Typically, packet error rate increases with the transmission rate, considering the same signal-to-noise ratio conditions, because lower rates tend to use more robust modulations and code rates. Since control packets are smaller than data packet, they are usually sent at lower rate and thereby at more robust rate than data packet. Consequently, error rates are higher for data packets and these metrics based on control packet broadcasting does not reflect the real link state and may lead to inaccurate statistics.

4.3 Basic Principles of AJIA

4.3.1 Overview

Throughout this chapter, we focus on elaborating an efficient packet loss control mechanism with implicit acknowledgments. Our protocol tackles the link failure and packet loss problem, by proposing a reliable error control protocol in a limited environment in terms of computational resources. Our idea stems from the overhearing characteristic of wireless communications as shown in figure 4.1. When a node transmits a packet, nodes in its neighborhood overhear the packet transmission even if they are not the intended receivers. This is due to the broadcast nature of the wireless channel. Our solution uses this overhearing characteristic instead of sending specific acknowledgment messages to guarantee reliability in the network. Moreover, when a packet loss is detected, retransmission is carried out on the most reliable link between the node originating the lost packet and its one-hop neighbors. We define the reliability of links through a metric based on the link history and the link quality indicator (LQI) which will be detailed in the next subsection. Besides, the device resource heterogeneity is exploited to load balance the traffic and share the current workload. Indeed, nodes with available resources are the most involved in retransmission issues. To achieve these goals, we use a different approach in comparison to traditional end-to-end error recovery mechanisms, where only the final destination node is responsible for detecting loss and requesting retransmission. We propose a hop-by-hop packet loss recovery mechanism, in which intermediate nodes also take responsibility for loss detection and recovery. This approach segments multi-hop forwarding operations into a series of single hop transmissions, so that error accumulation is avoided. Intuitively, the hop-by-hop approach is more scalable and capable to recover from loss.

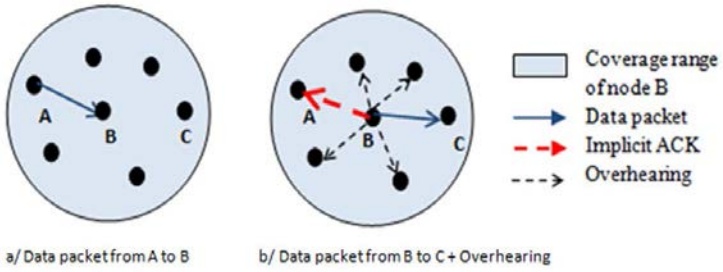


Figure 4.1: Overhearing of the transmission from B to C by node A

4.3.2 Network model and assumptions

4.3.2.1 Assumptions

We consider a dense set of randomly distributed sensors. Our objective is to provide a reliable retransmission scheme that takes into consideration the link consistency without inducing the extra overhead caused by acknowledgement messages. Before discussing the details of our protocol, we need to clarify our assumptions:

- We assume that the network is composed of sensors which are randomly distributed in a limited environment.
- We consider a many-to-one traffic pattern where source nodes send data to the sink.
- Data packets are randomly generated and transmitted to the sink node.
- We assume that nodes are stationary during their lifetime and able to record the performance of the link between themselves and their neighboring nodes in terms of the ratio packet lost/packets sent.
- We adopt a routing scheme, where a packet gets one-hop closer to the sink after each transmission. This is made possible by assigning to each node a level as shown in figure 4.2 corresponding to its hop-distance from the sink.
- Data is transmitted level by level toward the sink. When different nodes are at the same distance from the sink, the next hop is chosen according to the AJIA metric explained in the next section.
- We assume that the collection of data from a node to the sink must be completed within a specified time. If the packet does not reach the sink within this time limit, it is dropped and considered as lost.

- We consider that channels are symmetric, so that both the endpoints of the channel will keep an identical history of link performance. Link state is set to up or down after each packet transmission depending on the reception/loss of the packet at the receiving endpoint. Since the channel state is binary, a simple count of the number of states (up/down) is sufficient to fully describe its history. We define the channel history as a metric to evaluate the reliability of the link based on the number of the recorded state of the channel. The link/channel history will be used to elaborate our AJIA metric used in the recovery mechanism, which will be explained in the next subsection. The history of the link states, we can characterize the link reliability as the probability of link failure.
- In this protocol, we define the rank of a node as a metric corresponding to the hop-distance of this node from the sink. Also, we define a level as a set of nodes with the same rank.

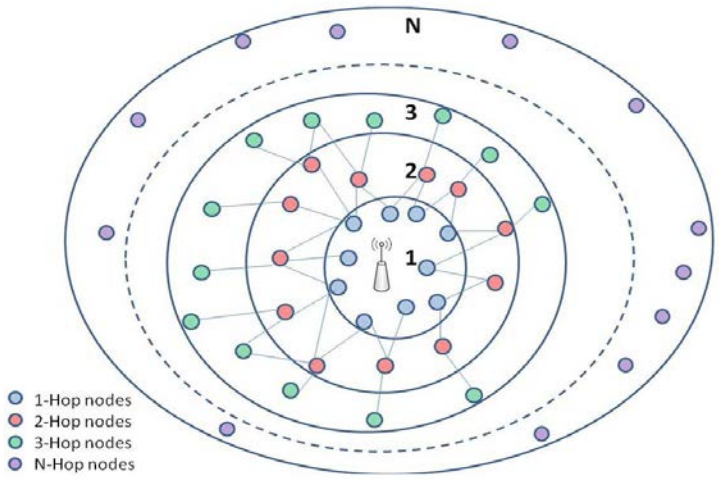


Figure 4.2: Network levels

4.3.2.2 IEEE 802.15.4

Although our protocol improves reliability, it must be supported by lower-layer stack protocols when designing a globally power-efficient architecture. An emerging standard for the WSN MAC/PHY layer is IEEE 802.15.4 [101], that refers to low-rate and low-power wireless personal area networks (LRWPAN). In this section, we give a brief description of this standard with a focus on its adaptability to AJIA to make easy the understanding of our protocol implementation.

Overview and architecture:
As already mentioned, the IEEE 802.15.4 standard specifies the physical layer (PHY) and Medium Access Control (MAC) sub-layer that are designed for low data-rate, short-range Wireless Personal Area Networks (WPANs). Figure 4.3 illustrates the global architecture of the standard. The MAC sublayer provides access to the physical channel and supports two operation modes, namely beacon enabled mode and

beaconless mode. In both modes, the network shall have one central controller device commonly referred to as PAN coordinator. The PAN coordinator is responsible for PAN identification, device address assignment and device synchronization. IEEE 802.15.4 introduces the concept of beaconing to improve the probability of successful delivery and decrease power consumption. A beacon is a control frame that carries information about the current network configuration.

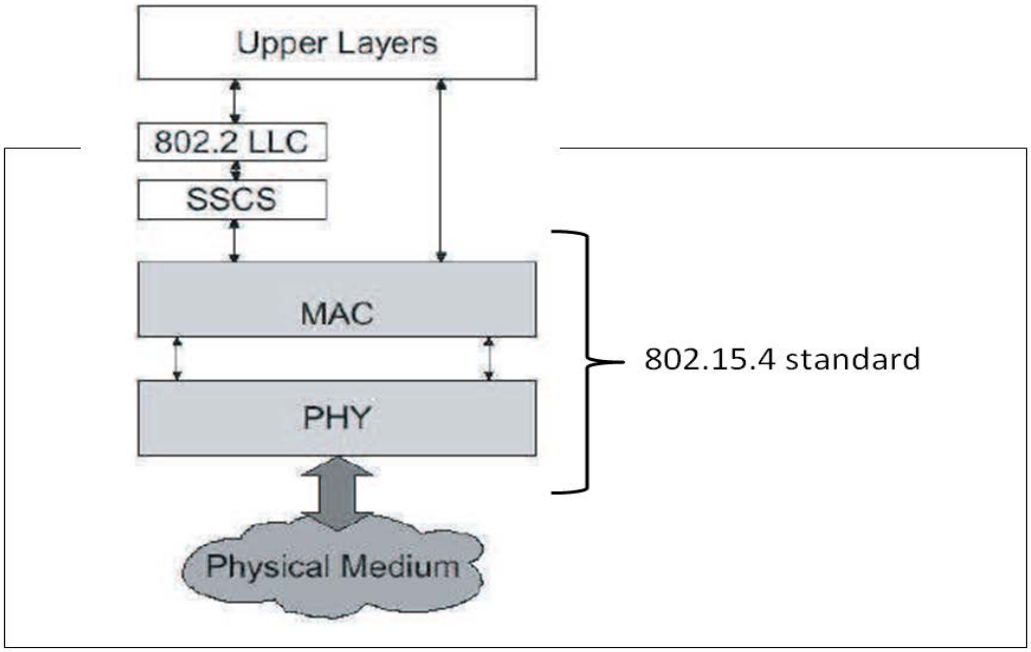


Figure 4.3: WPAN device architecture[10]

The IEEE 802.15.4 can operate in one of the two modes: Beacon mode and Beaconless mode.

- **Beaconless mode**: With the beaconless mode, devices communicate without any synchronization and transmit frames according to an unslotted CSMA-CA algorithm.

- **Beacon enabled mode**: The PAN coordinator periodically sends a beacon frame to provide synchronization and other services to child devices .The time is divided into superframes bounded by beacons frames. Each superframe has an active portion and an optional inactive portion as shown in figure 4.4. Devices communicate only during the active period and enter into a low power mode during the inactive period in order to save energy.

  The active portion of each superframe is divided into 16 equally sized slots and includes three parts: a beacon, a Contention Access Period (CAP) and an optional Contention Free Period (CFP). The beacon is sent at the start of slot 0 without the use of CSMA-CA. It is used to describe the structure of the

current superframe, identify the PAN and synchronize the attached devices. The CAP period will start right after the beacon and all frames sent during this period shall use slotted CSMA-CA channel access mechanism. The PAN coordinator may dedicate slots from the active superframe to devices that require dedicated bandwidth. These slots are called Guaranteed Time Slots (GTS) and together they constitute the CFP period.

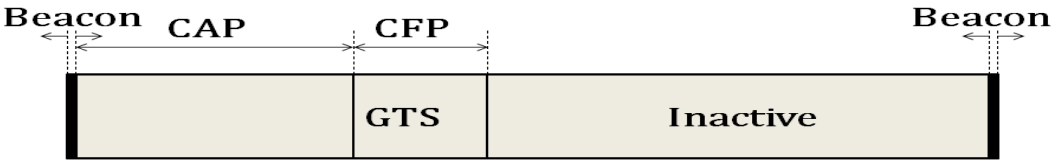


Figure 4.4: Superframe structure in 802.15.4

Synchronization of nodes through 802.15.4
The standard defines two main network topologies: star and peer-to-peer. To meet the requirements of AJIA and fit the adhoc structure of wireless sensor networks, we adopt the peer-to-peer topology of IEEE 802.15.4. We work in beacon-enabled mode with the use of superframes in order to achieve synchronization between levels.

4.3.2.3 Link Quality Indicator

LQI is a metric of the current strength of the received signal. By using the 802.15.4 standard, we have access to the link quality measurement LQI between neighboring nodes in the network. This measurement is in the form of a LQI value which is reported with each received packet in the MAC header and the result is reported as an integer ranging from 0 to 255. The limit LQI values (0 and 255) characterize the lowest and the highest quality IEEE 802.15.4 reception detectable by the receiver. Until now, there is no context of use defined for the LQI in the standard Zigbee. In this paper, we propose the use of this measurement as a part of our metric for the routing decision.

4.3.2.4 Link failure model

In order to provide an adequate measure of network reliability, the use of probabilistic reliability metrics is necessary. In wireless networks, link failures may be

caused by radio fading, radio interference, background noise and other inherent characteristics of the wireless medium as stated in chapter 1. Therefore, the link failure is more frequent than the node failure. Thus, our analysis will focus on link failure only, whereas node failure could be considered as the failure of all its links.

In this section, we present the link failure model adopted in our protocol simulation. It is a well known realistic distance-dependent model (depending on the Euclidian distance between nodes) which is used in [86].

In reality, the received power levels might vary significantly in the transmission area range around nodes over time. Thus, to make the model realistic, one might assume a log normal shadowing radio propagation model [57] where the logarithmic value of the mean received power is normally distributed with standard deviation σ. The link failure probability is then given as a function of the distance r (the length of the link) relatively to r_0 corresponding to the maximum length of a link between two neighbor nodes. This probability is specified as:

$$p(r) = \frac{1}{2} + \left[1 + \text{erf}\left(\frac{10 * \log \frac{r}{r_0}}{\sqrt{2}\log 10\psi}\right)\right], \psi \doteq \frac{\sigma}{\eta} \tag{4.1}$$

In this formula ψ is the ratio between the standard deviation of shadowing σ, and the path-loss exponent η which depends on the environment and may vary between 2 (in free space) up to 6 (in urban area for example). Low values of ψ correspond to small variations of signal power around the transmission range area mean power and high values of ψ correspond to stronger shadowing effects.

4.4 AJIA Operation

The motivation behind our protocol is to minimize the loss recovery cost by using localized data recovery among one hop neighbors, while ensuring low latency. This section gives the details of our protocol. It comprises an initialization phase followed by three steps: primary path establishment, lost message detection and selective recovery.

4.4.1 Initialization phase

Nodes enter a setup phase and perform the following initialization tasks:

- Neighbor discovery.
- Rank assignment.
- Each node of rank k constitutes the list of the one-hop neighbors with rank k-1.
- Computation of primary metric.

The initialization phase is illustrated through the pseudo code of figure 4.5. We start with the construction of a spanning tree for routing operations. To perform this task, we assign ranks to nodes thanks to a discovery neighbor phase. The sink starts by

broadcasting a *HelloMessage* with a *SequenceNumber* equals to 0 corresponding to its rank. Nodes receiving this *HelloMessage* will be considered as rank 1 and will broadcast this message to their neighbors after incrementing the *SequenceNumber*. This step is repeated until all nodes have an assigned rank. In the meantime, this operation allows each node to form the one hop neighbors list with lower rank and to compute the initial AJIA metric.

```
Int init=nnodes+10

for (i=1,i=nnodes,i++)
{
    NODE[i].rank=init;
}

SINK.rank=0;
HelloMessage.SeqNumber=1
SINK.SEND(HelloMessage);

While(T<TimeInit)
{
    if (NODE[k].recHelloMess(NODE[j])== True)
    {
          if (NODE[k].rank>HelloMessage.SeqNumber)
          {
                    NODE [k].rank=HelloMessage.SeqNumber;
                    HelloMessage.SeqNumber=(HelloMessage.SeqNumber)+1;
                    //List of  one-hop neighbors with lower rank
                    NODE [k].candidateList.PUSH(NODE[j]);
                    NODE [k].COMPUTE_INIT_METRIC(LQI,Phist); //Compute metric
          }
   else
          {
                    HelloMessage.SeqNumber=(NODE[k].rank)+1;
          }
          NODE[k].SEND(HelloMessage);
}
```

Figure 4.5: Pseudo code of the initialization phase

Let A and B be two communicating nodes, n_{up} be the number of successful transmissions through the link AB and n_{down} be the number of failed transmissions. We denote the link failure probability, P_{hist}, as it follows:

$$P_{hist}(A,B) = \frac{n_{up}}{n_{up} + n_{down}} \tag{4.2}$$

Moreover, during this phase, we establish the one-hop list of candidate nodes for retransmission tasks. This list does not include all the one-hop nodes but only those with rank lower than the considered node in order to get closer to the sink after each retransmission as shown in figure 4.6.

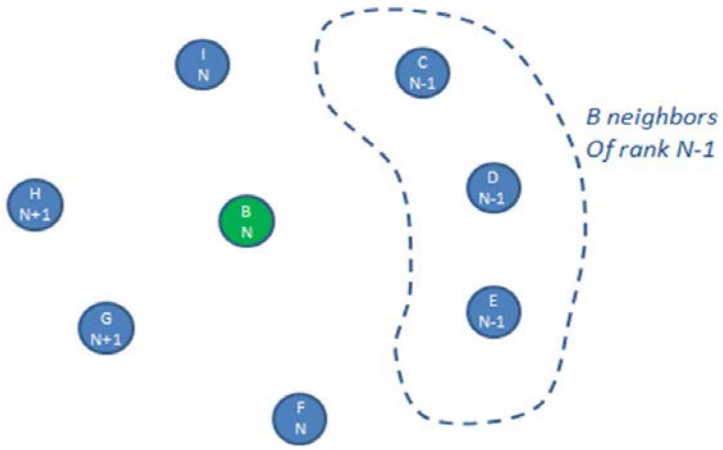


Figure 4.6: Elaboration of the one hop list of candidate

4.4.2 Data relaying

Once the initialization phase completed, data will be then transmitted level by level toward the sink in order to avoid loops. Intermediate nodes keep the *packetId* in their buffer during *Tbuff* which correspond to the maximum delay allowed to transmit a packet from a given node to its neighbor node. Neighbors that receive this packet check their local data cache to discard any replica. If the packet is received for the second time, it is automatically dropped to avoid message duplication. AJIA metric evaluates and assigns costs to network links, and then it determines the most suitable node to act as a relay. The metric component of AJIA evaluates links according to 3 parameters:

- The history of the link presented in the previous subsection (Phist).
- The Link Quality Indicator *LQI* is a metric of the current received signal quality. By using the 802.15.4 standard [15], we have access to the *LQI* of the channel between neighboring nodes in the network. This measurement is included into the MAC header of each received packet. The use of the *LQI* score function ensures a certain level of adaptability to the environmental conditions. In fact, it expresses the real quality of the link. The implementation of our adaptive routing is accomplished by considering the *LQI*, which is considered as a so-called 'stigmergic' variable (i.e. a variable that contains the information used by nodes to communicate indirectly). The *LQI* variables are defined and used by nodes to adaptively change the way they build routing path. Any change in this value may induce a change in the preferred direction towards the sink.
- The resources of the node (E): the heterogeneous nature of the network infers various levels of resources consumption for different nodes as stated in chapter. This is the reason why we pay special attention to this parameter when we

choose the best next hop and we exploit the network diversity to save nodes energy.

Besides, LQI experiences frequent fluctuations in highly interfered environment. Hence, we consider statistics (average number of lost packet per link) as a basis to assess the reliability of links. For this reason, we have decided to weight the AJIA metric by the link failure probability given by our probabilistic history model, $P_{hist}(A, B)$ described in equation4.2. Therefore, even if the last recorded value of LQI does not match the real state of the link, the history will correct it. Let us assume that $P_{hist}(A, B)$ is set to 1 at the beginning, and during a fixed time T_{init}, before obtaining a real history. In fact, at the establishment of the process, we do not have sufficient feedback to assess the reliability of a link. To do so, AJIA assigns a cost to each link, which is given by the following formula:

$$AJIA_{AB} = \alpha * e^{-age} * LQI_{AB} + \beta * P_{histAB} + \delta * E \tag{4.3}$$

Where LQI_{AB} denotes the link state indicator between nodes A and B and age corresponds to the delay since the LQI value has been recorded. The exponential function provides a decreasing function according to the age, which means that more recent values of LQI are considered as more significant. The P_{hist} represents the probability of transmission success between nodes A and B. α,β and δ are constants used to weight the equation. E is related to the resource available at the device. It varies from 1 to 10, where 1 means a device with a limited battery and 10 indicates a device with maximum battery capacity. This metric is calculated periodically in the network to update the route and make the protocol more robust against environmental change. This update period depends on the packet error rate of the network: the faultier the network is, the more frequent the update is.

4.4.3 Path establishment

The main objective of this step is to establish a global path with the most reliable links between the source x and the destination d. To do so, we use our AJIA metric as a criterion to evaluate links and to choose the next hop to which the packet have to be forwarded. In our solution, the next hop B is selected after the evaluation of the following formula :

$$AB = \text{argmin}\{\gamma_{lAB} \backslash \text{rank A= rank B +1 }\} \tag{4.4}$$

Where γ_{lAB} corresponds to the cost of the link AB. It depends on the link quality and can be written as follows:

$$\gamma_{lAB} = \frac{1}{AJIA_{AB}} \tag{4.5}$$

Consequently, the path formation is accomplished hop by hop until reaching the destination node.

With the proposed path selection criterion, we avoid routing loops. In fact, we need to recall here that we assumed that the first parameter to consider for transmitting a packet from a node x to the sink is the number of hops. Consequently, only the one-hop neighbors with a level value lower than that of x (nodes which are closer to the sink) will be candidates to be the next hop.

4.4.4 Packet loss detection

Packet loss detection is achieved thanks to the overhearing mechanism. Let us assume that node A sends a packet *packetId56* to node B. After the transmission, node A awaits during *Twait* overhearing node B's transmission, in order to check whether the packet has been forwarded to the next hop or not. By localizing loss events, this mechanism segments the multi-hop forwarding operations into a series of single hop transmission processes, which is effective in highly error-prone environments. Upon sensing the channel, if node A does not hear node B transmitting the packet *packetId56* to its next hop, it becomes aware that the packet has been lost. . So, it has to retransmit it. So, it has to retransmit it.

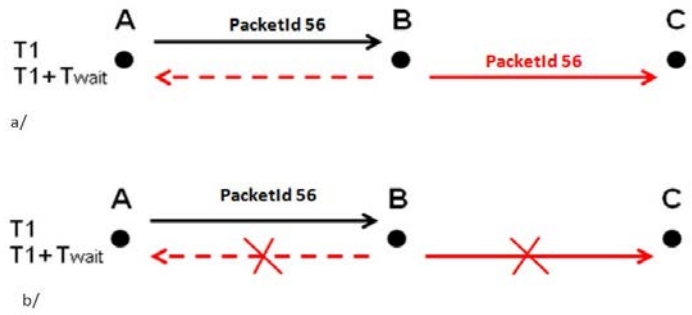


Figure 4.7: Packet loss detection:a/overhearing(implicit ACK) b/packet loss

4.4.5 Selective recovery

Once a packet loss is detected, the AJIA mechanism relies on its routing metric to choose the best next hop B' for the packet retransmission. Once this metric is calculated for all the candidate nodes closer to the sink, node A selects the node with the highest AJIA metric as next hop to which it retransmits the lost packet. Indeed, the reliability of the link is proportional to this AJIA metric: higher the metric is, more reliable the link is. Consequently, the reliability of the link is inversely proportional to $\gamma_{lAB'}$. Thus the link AB is selected with the smallest $\gamma_{lAB'}$ value among all candidate nodes. Therefore, the node carrying on the retransmission task is not necessarily the one which has sent the lost packet. The use of alternative nodes deals with temporal fluctuation of the link quality.

This process is repeated until the packet is received by the next hop (an implicit ACK is received) or until a specific timer expires and ends the packet lifetime. In this last case, the packet is dropped and considered as lost.

4.5 Conclusion

In this chapter, we proposed a new routing protocol with an error recovery support for heterogeneous wireless sensor networks called AJIA. We also explained in details the design of our solution. In order to demonstrate the performance of AJIA, we conducted simulation experiments and compared the results to our best competitor in the literature. The simulation results are presented in Chapter 5.

Chapter 5

Performance Evaluation of AJIA

In this chapter, we study the efficiency of AJIA through simulation experiments. We describe in detail our approach to perform this evaluation through the framework Castalia, and then we analyze the relative performance of AJIA and AODV. The results demonstrate the ability of our solution to provide higher packets delivery ratio while preserving nodes energy and show that our protocol provide lower latency.

5.1 Goal

In the previous chapter, we presented in detail the design of AJIA, a lightweight routing protocol with an error recovery support for heterogeneous wireless sensor networks. With our solution, we aim to provide high packet delivery ratio while ensuring energy efficiency and low end to end delay. In order to assess the performance of our protocol, we carried out extensive simulations and we evaluated the results compared with those of AODV [23], Ad-hoc On-demand Distance Vector routing. We selected AODV as a competitor since it is a well-known routing protocol designed for dynamic application scenarios in wireless Ad-hoc networks and it is the closest protocol in the literature to our protocol offering quick adaptation to dynamic link conditions, . In order to evaluate the abilities of AJIA, we examine the following metrics: *delivery ratio, latency, energy efficiency and control message overhead.*

5.2 Simulation Environment

We implemented AJIA in Castalia [53] which is a framework simulator for Wireless Sensor Networks (WSN). To the best of our knowledge, Castalia WSNs simulator emerges for its feature and completeness. Castalia provides a generic platform to perform "first order validation of an algorithm before moving to an implementation on a specific platform" [53]. It is based on the well-known OMNeT++ [54] simulation environment. OMNeT's basic concepts are modules and messages. A simple module is the basic unit of execution. A high-level language (NED) allows to assemble modules into larger components and to configure simulation scenarios. Modules are programmed in C++.

In the context of this work, we make use of version 3:2 of Castalia, which builds upon version 4:1 of OMNeT++. In this version, Castalia offers exhaustive models for simulating both the radio channel and the physical layer of the radio module. In particular, it provides bundled support for the CC2420 radio controller, which is the transceiver of choice for the TelosB platform [25] and the radio chosen for our simulation model. Using an accurate channel model is an important issue in order to have realistic environment model and meaningful results. To do so, Castalia based its path loss model on two components: the average path loss and the temporal variation model.

The average path loss is given by the lognormal shadowing model given by the equation (4.1). This formula returns path loss in dB as a function of the distance between two nodes. The component of the path loss due to temporal variation is given thanks to a probability density function (pdf) that we draw our new value from. We keep the last "observed" value (in reality the last value we computed) and the time it has passed since then. These two numbers define this probability density function. Obviously, if little time has passed, then the bulk of the probability in this pdf should be in values close to the last observation. If the last observation is a deep fade (e.g. -40 dB) then the pdf should enhance bigger values. A typical channel variation is given in the figure below 5.1.

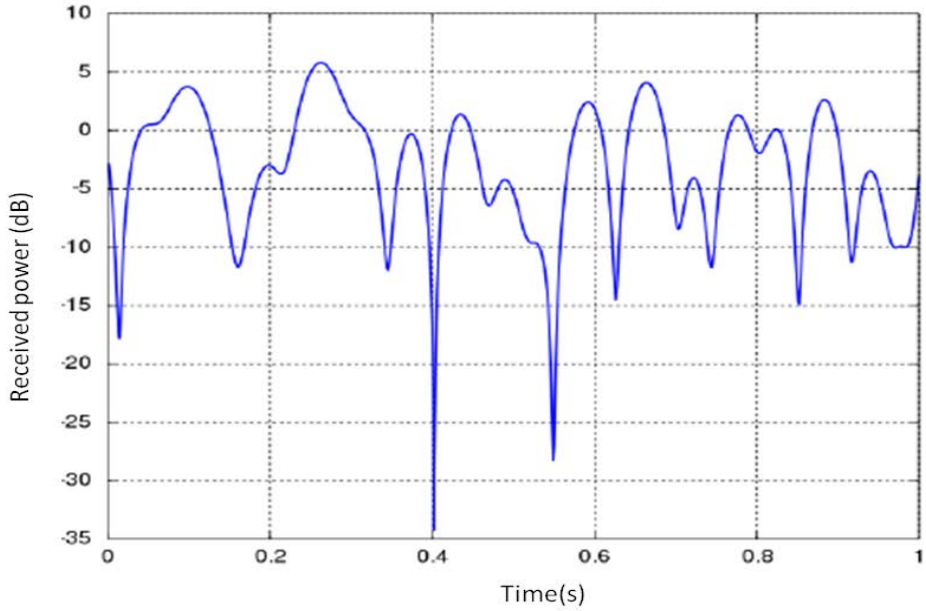


Figure 5.1: Typical temporal fading in Wireless Sensor Networks [53]

Castalia's basic module structure is shown in the diagram below5.2.

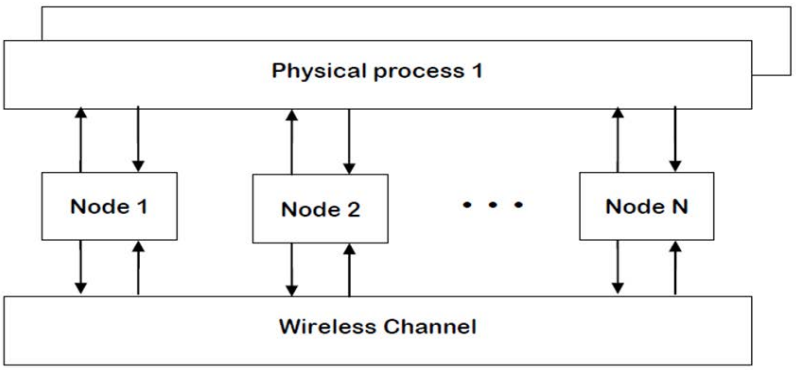


Figure 5.2: The modules and their connections in Castalia [53]

It is worth noting that the nodes do not interact directly with each other but through the wireless channel module(s). The arrows represent message passing from one module to another. When a node has a packet to send, it firstly goes to the wireless channel. Then, the latter decides where (to which nodes) to transmit it. The nodes are also linked through the physical processes, from where they get their sensor readings. The node module is a composite one. Figure5.3 shows the internal structure of the node composite module. The solid arrows signify message passing and the dashed arrows signify simple function calling.

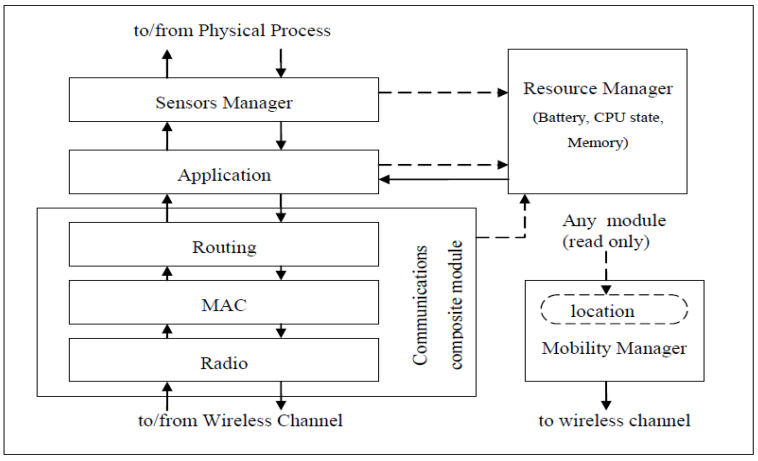


Figure 5.3: The node composite module [53]

Since this environment is free and provides building blocks for writing wireless networks simulations, we selected this solution to develop our simulations. We reused the IEEE 802.15.4 module and a basic application module. We developed our own routing layer to simulate AJIA and we also implemented AODV in the simulator. Finally, we set up identical scenarios to compare the performance of AJIA and AODV. The details of the implementation and the simulation parameters are given in the next section.

5.2.1 Simulation scenario and parameters

In this section, we provide an overview of the implementation parameters of AJIA and of the simulation scenario. We propose an implementation on top of IEEE 802.15.4 in a beacon-enabled mode with the use of superframes in order to achieve synchronization between levels. The different settings parameters are summarized in table 5.1.

Parameter	**Value**
Field dimension (m)	100x100
Nb nodes	40
Physical datarate (kbs)	250
TxPower (dBm)	-1-10
Path Loss Exponent	2.4
Standard deviation σ (db)	1-10

Table 5.1: Simulation parameters for Castalia

We simulated a network of 40 stationary nodes that we position on a square grid of 100mx100 m in a random manner: x and y coordinates are picked up randomly according to a uniform random variable. To evaluate our protocol we set up a value reporting scenario where data is generated randomly by nodes and forwarded to the sink.

Our protocol relies on a Loop free topology for ordinary routing operations, and resorts to exploiting alternative paths only when a malfunctioning is detected. To build this loop free topology, we resort to use Setup packets. These packets do not contain any payload, only the identifier of the sender node and a field characterizing the number of hops from the sink. The broadcast of these packets allows the formation of the different level in the network and the construction of the list of the candidate nodes for retransmission tasks: those closer to the sink in term of number of hops.

To meet the requirements of our protocol, we use three types of packets: *setup packets, data packets and ACK packets.* As explained in the previous chapter, AJIA exploits the overhearing feature as an implicit acknowledgment instead of the usual ACK packet. However, we cannot use implicit ACK for the sink node so we make use of ACK packets only for this node. To sum up, we have two kinds of control packets: *setup packets* and *ACK packets.* For the sake of completeness the figure 5.4below shows the structure of the different packets:

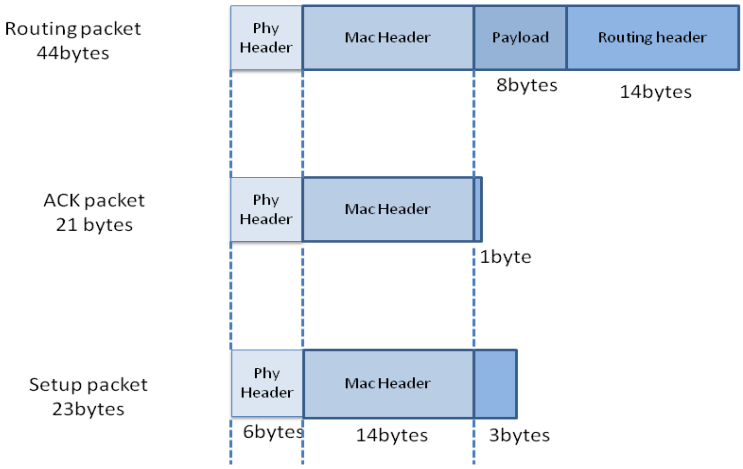


Figure 5.4: Structure of AJIA packets

The transmission power Tx is an input of the user. Castalia offers values that can be selected by the user. An example from the radio CC2420 defines the 8 possible transmission power levels is presented in the table below5.2. The first line corresponds to the output power of the different transmission levels in dBm and the second line gives the energy amount spent when transmitting in a the corresponding power level.

Tx (dBm)	0	-1	-3	-5	-7	-10	-15	-25
Tx (mW)	57.42	55.18	50.69	46.2	42.24	36.3	32.67	29.04

Table 5.2: Transmission power level for CC2420

We also used the IDEAL radio mode provided by Castalia which emulate the unit disk model where transmissions within a certain range from a transmitter are perfectly received, and outside this range not received at all. Doing this, we controlled the coverage range of nodes by controlling the Tx power thanks to the following formula:

$$P_L(d_0) = (\text{Tx} - \max(receiverSensitivity, noisefloor + 5)) - 10 * \beta * \log(coverage) \quad (5.1)$$

where β is the path loss exponent and Tx is transmission power in dBm.

5.2.2 AODV implementation

We used an open source code for AODV implementation [48]. The implementation respects the features of AODV provided in [23]. The three messages Route Requests (RREQs), Route Replies (RREPs), and Route Errors (RERRs) defined by

AODV for route establishment and maintenance are provided. Moreover,the destination sequence number feature insuring a loop free topology is implemented. We also enable the local repair of AODV to recover from link failure in order to compare the efficiency of its recovery process with ours.

5.3 Simulation results

We evaluated the performance of AJIA through extensive simulations using Castalia. We selected AODV as a competitor for our protocol and we evaluated its performance under the same simulation scenarios. We studied the relative performance of AJIA and AODV under various transmission power. As well, since the standard deviation is the main factor in shadowing effects, we have studied the delivery ratio evolution while varying this parameter.

Each scenario is simulated 20 times with different seeds. To measure the performances of AJIA we focused on several parameters namely comparative successful packet delivery ratio, average latency, energy consumption and overhead.

5.3.1 Delivery ratio

The delivery ratio represents the most important metric to estimate the efficiency of our solution in term of reliability. It is defined by the ratio between the total number of packets generated in the network at the source nodes and the total number of data packets successfully delivered to the sink. As explained in the previous chapters, packet loss occurs due to several factors including signal variation. Since signal variation is highly related to the standard variation and the transmission power, we considered these two axes for our evaluation process.

Figure 5.5 illustrates the comparative delivery ratio of AJIA and AODV on a network size of 40 nodes and a Tx equals to -3 dBm while varying the standard deviation Sigma. First of all, we notice that AJIA outperforms AODV for all sigma values. We have an average gap of 40% improvement between the delivery ratio of AJIA and AODV. Moreover, this figure depicts that higher standard deviation of the signal decreases the probability of successful packet reception. For example, the delivery ratio remains above 80% till standard deviation of four for AJIA. Hence we should not worry about the shadowing effects for smaller standard deviation. But the probability of successful packet reception decreases rapidly for higher standard deviations (i.e. 8 and 10). For example, for standard deviation of 10 we have only 69% and 25% of average delivery ratio for AJIA and AODV respectively. That means a decrease of almost 20% with values recorded for a null standard variation.

To investigate the effects of transmission power on the packet delivery ratio, we repeated the previous simulations while varying the transmission power Tx between -1 dBm and -10 dBm for an ideal modulation mode presented in the previous section. We recall that the IDEAL mode emulates the unit disk model where transmissions within a certain range from a transmitter are perfectly received and lost otherwise. The simulation results are depicted in figure 5.6. This figure shows that the delivery ratio improves considerably for high Tx power for both protocols. This result is

predictable since the increase in transmission power enlarge the coverage range and provide a higher number of neighbors for each node.

Moreover, AJIA achieves the best packet delivery ratio in all scenarios particularly for the highest Tx power where it is equal to 97% while it is 58% for AODV. This gap between AODV and AJIA could be explained by the number of dropped packets caused by local repair failure of AODV. Recall that for AODV ha node along an active path may locally repair the route if the destination is closer in number of hops to the node than to the source of the packet. When a node A performs local repair for destination B, an upstream node C may continue forwarding packets to A destined to B. This causes packets destined to B to be at node A. When the route repair fails, these packets at A are simply dropped.

This phenomenon is visible especially for low Tx power for example we have a delivery ratio of less than 30% for AODV while we have almost 50% for AJIA. On the other side, we notice that our recovery process performs well for all TX power and allows reaching a good delivery in all conditions. More precisely AJIA achieves almost 50% of delivery ratio while we have only 29% for AODV in the worst case.

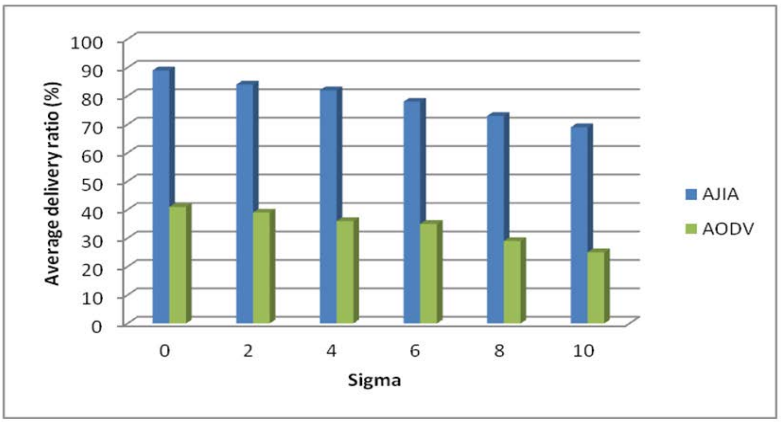


Figure 5.5: Comparative successful packet delivery ratio of AJIA and AODV while varying the standard variation

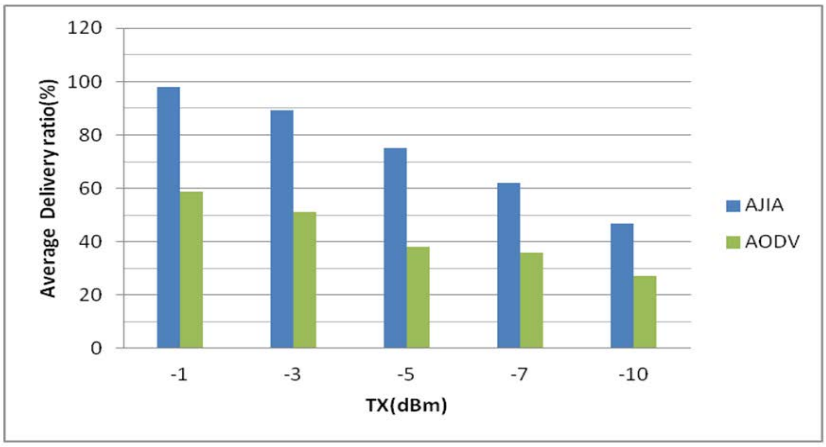


Figure 5.6: Comparative successful packet delivery ratio of AJIA and AODV while varying the transmission power

5.3.2 Average latency

In this experiment, we evaluate the end to end delay and measured the average latency for a packet to be transmitted from a source to the destination. To do so, we run several simulations while varying the Tx power and sigma.

We summarize results obtained while varying Tx with an ideal mode in figure 5.7. We notice that the average delay variation per packet is insignificant for the ideal mode since we assume that the signal level remains constant for a given distance, hence less number of packets needs to be retransmitted. On the other hand, we remark that AJIA has lower latency than AODV for all transmission powers even if the gap is very small.

Moreover, the shadowing effects consider a wide variation of the signal. Hence, more retransmissions are required which involve that a packet has to wait for longer period of time due to unsuccessful packet receptions. Figur 5.8 plots the delay of end to end packet delivery for AJIA and AODV while varying the standard variation Sigma.

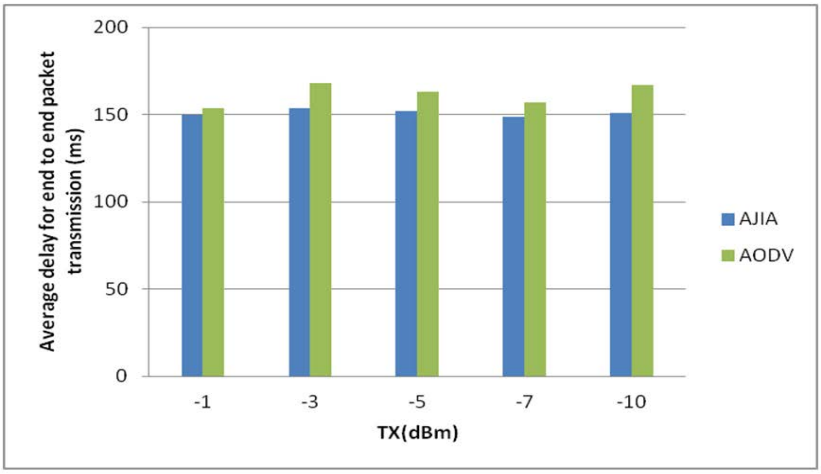


Figure 5.7: Average end to end delay for packet transmission while varying the transmission power

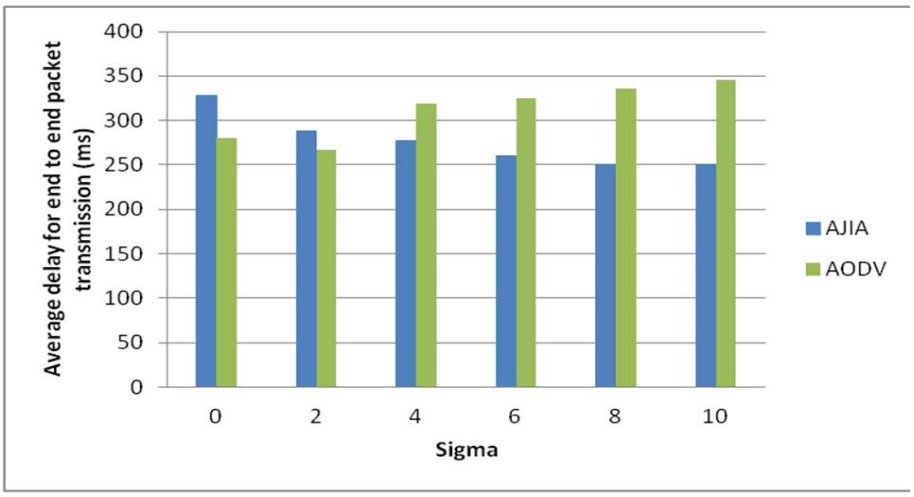


Figure 5.8: Average end to end delay for packet transmission while varying the standard variation

As expected, the delay increases as the Tx power decreases for AODV. Indeed, more packet losses are experienced thus more transmissions are required to deliver packets which extend the delivery delay. Moreover, we notice that AODV has the highest latency for highest sigma values (from sigma=4) which is due to the additional time of the path establishment for each new packet and during local repair. This delay would be decreased if we had another configuration with only one source

node which would not require the establishment of a new path to the destination for each packet. This gap between AJIA and AODV could also be explained by another major factor which is the increased congestion caused by local repair overhead. This additional overhead causes significant network congestion, which in turn causes a significant number of packets to be dropped and end-to-end delay to soar. We Note that AJIA's distributed packet recovery scheme does not experience this problem.

Furthermore, we notice that AJIA has a decreasing latency. That observation is due to the fact that dropped packets are mostly far from the sink in terms of number of hops. Indeed, the farther the source node is the more likely the packet is dropped. Accordingly, the end to end delay is reduced when sigma increases because high standard variation values imply more packet losses and thus more packets are dropped.

5.3.3 Message overhead

In this subsection, we compare the number of control packets exchanged during the simulation for both protocols. As stated in the beginning of this section, we have two different control packets for AJIA: *Setup Packets* and *ACK packets*. On the other side, AODV is based on the exchange of four control packets. Further information are provided in [16]. Figure 5.9shows that AJIA outperforms AODV for all Tx powers. For example, with Tx equals '-5 dBm' we have a gap of 36% increase between the number of control packets used by AJIA and AODV. Moreover, we notice that the amount of control packets used decreases with Tx power. Indeed, for lower values of Tx many nodes become unreachable thus the number of received packets decrease. Added to that, the big number of control packets used by AODV is due to unnecessary discovery of new routes. In fact, wireless links may suffer from temporary bad radio and this spurious link failure detection may trigger needlessly route discovery.

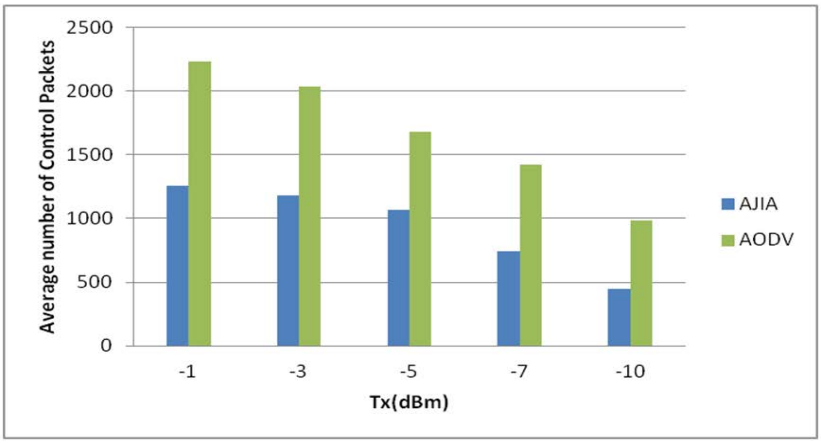


Figure 5.9: Comparative message overhead

Moreover, the number of control packets used by AJIA is mostly due to setup packets of the initialization step like depicted in figure 5.10. The number of ACK packets used for all configurations is below 10%. In fact, thanks to our recovery mechanism based on overhearing instead of the traditional explicit ACK mechanism, we highly reduce the number of control packets used preserving the bandwidth minimizing the number of collisions and enhancing the delivery ratio.

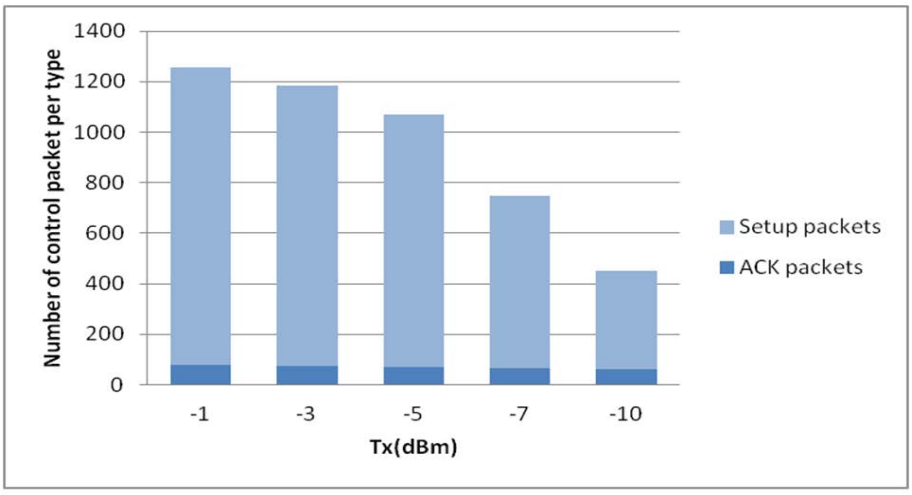


Figure 5.10: Control packets repartition for AJIA

5.3.4 Energy

We recall that we run the simulations for both protocols under the same MAC and radio settings given in table5.1. The average consumed energy per node is computed and depicted in figure 5.11 while varying the Tx power. For the sake of objectivity and accuracy, the energy model used by Castalia incorporates overhearing energy cost in the total energy expenditure due to a transmission between two nodes. Simulation results show that both protocols consume almost the same amount of energy. We notice that the energy spending decreases as the Tx power decreases which is an expectable result since less packets are transmitted.

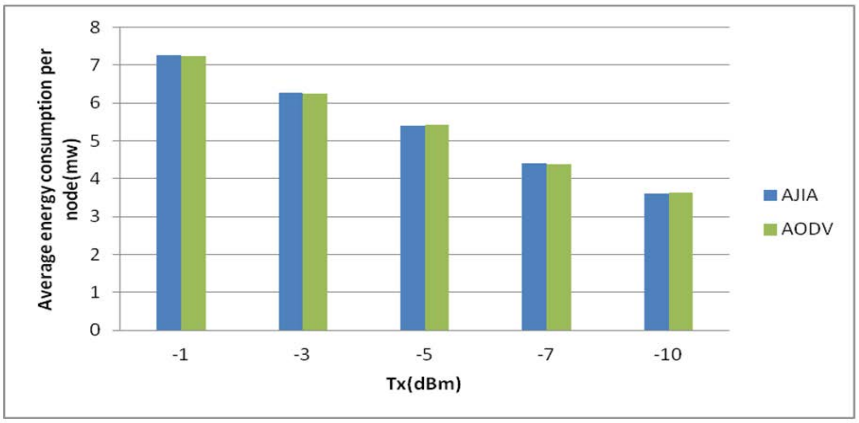


Figure 5.11: Comparative energy consumption

5.4 Conclusion

In this chapter, we performed extensive simulations via Castalia in order to demonstrate the performance of AJIA and compared the obtained results with our competitor AODV.

The results have shown that AJIA outperforms AODV in terms of delivery ratio for all channel variation scenarios: varying the Tx power and the standard variation. AJIA can offer almost 97% delivery ratio but at the cost of high transmission power which increase the energy consumption. On the other side, AJIA experiences limited packet loss in comparison to AODV (30% Vs 70%) for the highest standard variation value thanks its recovery process while preserving an acceptable latency.

Therefore, we showed that our protocol performs well in all conditions in term of delivery ratio, overhead, latency and acceptable energy expenditure.

As a future work, we aim to further analyze the performance of AJIA through the study of energy consumption and to modify the MAC layer in order to achieve more energy saving. In addition, we intend to compare the performance of AJIA

with our other contribution ARRP.

Chapter 6

Adaptive reliable routing protocol for WSNs

In order to provide reliable data delivery in WSNs, we proposed AJIA in the previous chapter which is a packet loss recovery protocol based on implicit acknowledgments and retransmissions on the most reliable link. In this chapter, we propose Adaptive Reliable Routing Protocol for WSNs (ARRP) which is a variant of AJIA. The novelty of this approach relies on the retransmission triggered automatically by the neighborhood upon the detection of a link failure thanks to nodes cooperation. Then, we detail the different steps of our protocol and asses its efficiency through simulations. Indeed, we have conducted intensive simulations considering different scenarios, topologies and demonstrated that our solution grants higher reliability.

6.1 Introduction

In traditional wireless networks, the broadcast nature of channels is considered as a severe challenge and many techniques have been elaborated to alleviate this effect. However, this characteristic opens a wide line of research targeted at exploiting all the potential benefits of those schemes. In fact, if the overheard information is properly forwarded by the surrounding nodes, then the overall performance can be improved. This fact has revealed a new direction of research that endorses node collaboration, namely cooperative communication [10].

Research on cooperative communication has gained momentum in the community at the physical layer but more recently its importance and usability have also been realized at upper layers of the network protocol stack. The improvement induced by exploiting cooperation in wireless networks can be attained in terms of reliability enhancement, reduction of power consumption, or even increased coverage range.

Among these collaborative services that have been proposed in the literature [76] [32] , we chose to focus on collaborative networking services [46] [62], which we define as featuring functions that improve the communication reliability of any two networked nodes. This communication model demonstrated its adaptability to local and time-varying requirements such as energy level of nodes, interconnection

throughput, link states, etc. Additionally, this paradigm allows self organization as well as the robustness to dynamic uncertainties such as node failures or non-stationary neighborhoods.

In the previous chapter, we brought out how overhearing and retransmission can be employed to guarantee reliable communication in WSNs. In this chapter, we propose to take advantage of node collaboration to enhance the design of our previous scheme. We tackle packet loss from a different axis trying to improve packet retransmission by relying on collaboration between nodes.

Indeed, we propose to redesign the retransmission step by relying on neighboring nodes to get over packet loss so that we reduce the communication overhead of the protocol. This step is achieved thanks to the caching of the overheard data by neighboring nodes. Then, once a packet loss is experienced, the neighboring nodes will act on behalf the node which detects packet loss, so that these neighboring nodes forward the data they have previously overheard. Thus no more retransmission is required by the node which firstly sends the packet. These steps are illustrated in figure 6.1 below.

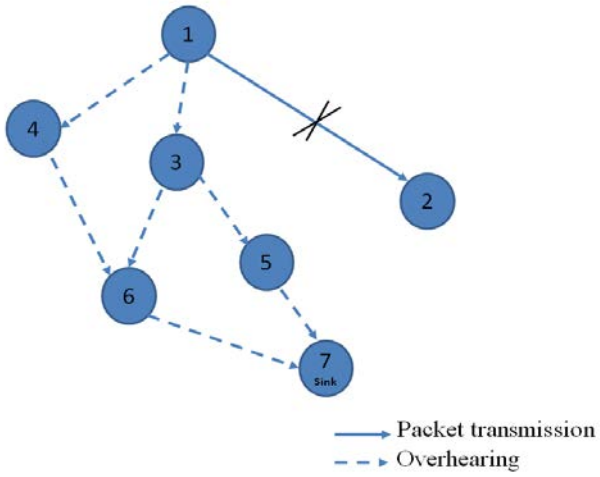


Figure 6.1: Retransmission scheme exploiting node collaboration

The rest of the chapter is organized as follows. In Section 2 we provide an overview of the existing scheme of collaborative communication through the different layers and discuss them in the context of wireless sensor networks. In Section 3 we give the basic idea of ARRP then detail its design specifications in section 4. The protocol operation is described in section 5 through different scenarios. In Section 5 we conduct both protocol analysis and simulations to assess the performance of ARRP in different network configurations. Finally, Section 6 concludes the paper.

6.2 Collaborative Schemes in wireless sensor networks

Collaborative communication fully leverages the broadcast nature of wireless channels and exploits time as well as spatial diversity in a distributed manner, thereby achieving significant improvements in system capacity and transmission reliability. Collaborative diversity has inspired a number of interesting ideas in the physical layer perspective. Thereafter, collaborative MAC design has also drawn much attention recently. In this subsection, we give an overview of some solutions proposed in the literature through the different layers.

6.2.1 Physical layer

The concept of collaboration in radio transmission field has been first introduced by Sendonaris et al. in 1998 [63] for cellular mobile users in order to increase the uplink capacity. In each cell, each user isresponsible not only for its own information but also for the information of its "partner user" that it receives and detects.

Afterward, this concept has been expanded and new approaches for cooperative transmission in wireless networks emerged. Cooperation has been exploited to overcome signal fading problems and to enhance communication quality in wireless communications. Many schemes, dealing with how to relay the information, have been proposed such as store and forward (S&F), amplify-and-forward (A&F), decode and forward (D&F), coded cooperation (CC), and many others [13] [32] [33] [22].

With A&F scheme, a group of relay nodes receive a signal from a source and simply retransmit it to the destination without decoding it. With D&F scheme , relay nodes are more involved. They decode the received message, re-encode it to enhance error protection and retransmit it as a new message. Upon reception of multiple signals from the source and the cooperating nodes, the destination combines them and recovers the original message. The advantages of these cooperative schemes often depend on the availability of reliable inter-node links. The benefit of A&F scheme relies on the quality of the relayed signal since cooperating nodes amplify both the signal and the noise received from the source. Likewise, in D&F scheme, an assisting node can decode and relay the message only if it is able to receive reliably the original message from the source.

6.2.2 MAC layer

Cooperative Automatic Retransmission Request (C-ARQ) schemes exploit the broadcast nature of the radio channel by allowing users which overhear a transmission to act as spontaneous relays when a packet has been received with errors at destination. Transmission is done as follows: the source transmits a frame to the destination and in the case of error,a relay node retransmits the frame, so providing the system with cooperative diversity.

The gains of a cooperative ARQ scheme has been analyzed in terms of improved probability of error in [16], SNR gain and average number of required retransmissions

in [102] . Some other works have been focused on the relay selection criteria within the context of distributed cooperative ARQ schemes. For example, in the works presented in [8] [34], an opportunistic forwarding scheme is presented wherein the best candidate to retransmit is selected whenever a communication has failed. Previous work showed that distributed cooperative ARQ schemes may yield improved performance, lower energy consumption and interference, as well as increased coverage area by allowing communication at lower SNRs.

It is worth mentioning that there exists in the literature a completely different family of cooperative MAC protocols [31] [76] [66] which have not been designed for the execution of distributed cooperative ARQ schemes in wireless networks, but they are aimed at solving other kind of interesting cooperative issues. However, up to our knowledge, there are no network protocols conceived to execute cooperative ARQ schemes in wireless networks and to attain the achievable benefits discussed in the aforementioned research works.This is the main motivation of the development of our protocol ARRP based on C-ARQ in the network layer.

6.2.3 Routing layer

The broadcast property of wireless channel has been explored in the network layer to provide energy efficiency and . Khandani et al. [2] tackles the issue of finding the optimal path in a multi-stage decision making problem, where at each stage a set of nodes may cooperate to relay the information to a chosen node. The aim is to find the tradeoff between spending more energy in each transmission slot to reach a larger set of nodes, and the potential savings in energy in subsequent transmission slots due to cooperation. ExOR [8] is proposed to increase the throughput in a multi-hop wireless networks by taking advantage of the multiple forwarders. It is likely to increase total network capacity as well as individual connection throughput. In [83], a modified version of AODV over specialized IEEE 802.11 MAC protocol is proposed to strengthen the path reliability through selecting the optimal relay node. Combining a MAC protocol capable of channel state based next hop selection [28] with AODV [45], the proposed method could deal with packet loss due to channel error. Srinivasan et al. [72] apply game theory to the problem of cooperation of energy constrained nodes. The authors in [30] work on the cross-layer design in which a set of cooperating nodes are selected to transmit to a set of receiving nodes with the objective to minimize energy consumption. An energy efficient cooperative routing scheme with space diversity using space-time block codes (STBCs) is proposed in [40]. Full diversity from the orthogonal STBC is utilized to overcome multipath fading and to enhance power efficiency. X Huang et al. [21] proposed a Robust Cooperative Routing Protocol catering to mobile WSNs.

Cooperative caching, sharing and coordination of cached data among multiple nodes can improve the delay and reliability of packet delivery in wireless ad hoc networks. Yin and Cao propose three different cooperative caching scheme (data, path and hybrid caching) [99] to reduce the query delay and message complexity. In [79] , the authors employ cooperative packet caching and shortest multipath routing to reduce packet loss. Unlike our protocol ARRP, this solution require to store not only the overheard data but also multiple routes to every active destination which

become unsuitable with the limited memory of WSNs.

The fundamental concept of nodes cooperation has been deeply studied during the last years and it becomes one of the most attractive topics in several engineering fields ranging from information theory to computer science. However, there is still a long way ahead in bringing to life all these theoretical concepts and developing efficient protocols that can exploit the inherent broadcast nature of wireless links to improve the performance of networks operating over the air interface. Among other open issues, the design of en efficient routing protocol exploiting nodes cooperation to reduce packet loss and enhance the network reliability is yet a topic of great interest.

6.3 Overview of ARRP

In our protocol, we keep the same assumptions as the previous chapter by assuming a randomly and densely deployed wireless sensor network where packets are generated randomly and forwarded hop by hop to the sink node. We assume all nodes have the same transmission range and a path has already been established between a source and a destination according to a specific metric presented in the next section. As nodes in the coverage range of each source node are able to take advantage of wireless broadcast characteristic of the channel, they can work cooperatively to deliver packets once a packet loss is detected. Thus, by relying on multiple nodes for recovering from packet loss, the robustness is enhanced at each hop. To put it more concrete let's revisit figure 6.1 If link 1-2 fails due to deep fading, then node 2 will lose the packet. Without waiting for potential multiple retransmissions over the unreliable or lost link 1-2, a substitute link 3-5 or 4-6 could transfer the packet proactively to the sink node 7. As long as at least one link is capable of delivering the packet successfully, the packet can be received and further forwarded towards the destination.

We define the set of eligible nodes as the set of candidate nodes for retransmission task once a packet loss is experienced at one link. This eligibility is acquired after a rank assignment process which will be detailed in the next subsection.

To sum up, when a node fails to receive a packet from its upstream node, eligible nodes from the neighborhood will help to forward the packet proactively to the downstream node without waiting for the routing instruction. This is made possible thanks to coordination between eligible nodes by the exchange of control messages. The schedule of message exchange will be explained in the next subsection. The probability that all eligible links fail simultaneously is much smaller than the probability of the failure of only one link. Therefore, alternate eligible links can improve the reliability and reduce the end-to-end delay. On the other hand, energy savings via avoiding retransmissions over a lost link may potentially offset the energy consumption of overhearing. It is possible that cooperation among eligible nodes lowers the energy consumption while achieving robustness. ARRP's design faces the following key challenges:

- The nodes must agree on which sub-set of them is eligible to retransmit the lost packet and when the retransmission should take place. Since agreement

involves communication, it must have low enough overhead that doesn't overwhelm ARRP's potential throughput gain. The protocol must also be robust enough to minimize duplicate forwarding while retransmitting. This is achieved through the coordination of nodes. That's why we decide to reinforce the protection of the exchanged control packet (representing the essence of the coordination step) by an adequate FEC.

- A metric reflecting the reliability cost of each link have to be defined in order to classify eligible nodes and to define the best node to carry on retransmission task. Our index assignment mechanism fulfills this requirement and permits to fully sort nodes by means of a reliable metric.

6.4 Basic Principles of ARRP

6.4.1 Neighbor index assignment

As explained above, neighbor index assignment is used not only for routing but also to classify eligible nodes hence its importance. The neighbor index assignment is based on the AJIA metric presented on the previous chapter. Recall that, this metric reflects the links quality and is based on the LQI measure and the history of the link. Figure 6.2 represents a node B of rank N and its neighborhood. More particularly, it shows its N-1 neighbors C, D and E of index 1, 0, and 2, respectively. We assign an index with an increasing reliability order which means that more reliable the link is, lower the index is. So that node with index 0 has the best metric. Let us notice that our protocol provides uniqueness of index to avoid collision problem: when different nodes have the same index, a random back off is added to the metric in order to have distinguish index. In our protocol, we consider that each node is aware of its index in regards to its upstream nodes.

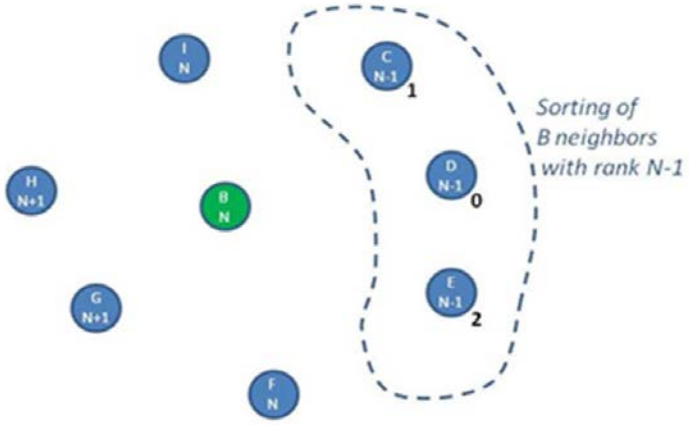


Figure 6.2: Neighbor index assignment

6.4.2 Transient context definition

Once a packet with a Packet IDentifier (*PID*) is received for the first time by a node of rank N from a node of rank N+1, the packet is stored and an associated transient context is created in its memory to manage this packet PID. Each node associates a transient context to every couple (PID, Src), namely, the packet Id received and the sender node. This context is used to manage the control packets and to schedule the retransmission task. Indeed, each context requires a specific action to follow depending on the received message. Use-cases of the transient context will be given in the next section. We distinguish two type of context: *Primary context* and *Secondary Context* . Primary context (*P Ctxt*) is related to the node with the best metric (level 0 in the beginning) in regards to the sender node. Otherwise, the context is considered as 'Secondary' (*S Ctxt*). Basically, a node with a primary context for a given data packet would immediately forward it whereas a node with a secondary context would buffer received packets and wait for a possible retransmission request. The retransmission task is always carried out by the node with a (*P Ctxt*). That's why the type of the context (primary or secondary) is updated each T_{delay} in order to trigger the retransmission by the most appropriate node. The context may be then upgraded from primary to secondary to take care of a retransmission when required. Further details will be given in section 5 through the different scenarios.

6.4.3 Differentiated FEC on ARRP packets

As explained in the previous section, ARRP exchange two types of packets:

- Data packets

  Data packets are generated by sensor nodes then gathered and forwarded to the sink. The aim of our protocol is to prevent the loss of data packets. Retransmission represents the key mechanism for providing the reliability in our protocol . On the other hand, an excess of retransmissions may provoke extra energy consumption. Consequently, we decide to decrease the number of retransmissions by adding redundant data on each message so the receiver can detect and correct errors. On the other side, our analysis of the state of the art in chapter 1 stressed the fact that the use of FEC may induce an extra overhead in dense networks which is not suitable with the limited computational capacities of sensor nodes.

- Control packets

  ARRP exchanges three types of control packets for the coordination of retransmission tasks among nodes namely Explicit Retransmission Request (*ERR*), Explicit Retransmission Cancel (*ERC*) and Taking-Care (*TC*).

  - *ERR* is sent to the eligible nodes once a packet loss is detected. This message is sent after T_{delay} corresponding to the maximum delay for a packet to be sent and acknowledged. Beyond this delay, the packet is considered as lost and the retransmission task is triggered thanks to the

ERR as shown in figure 6.3. Once this message is received by the set of eligible nodes, the lost packet is retransmitted each T_{delay} by one of the candidate nodes (eligible nodes) unless an *ERC* message is received in the meantime. The most reliable node (with index 0) begins the retransmission. If the packet is still lost, node with index 1 will retransmit it after T_{delay} and so on until the packet is received or a timeout expires.

- *TC* is sent by the node carrying the retransmission to the node which sent the *ERR* message in order to trigger the *ERC* message sending. Actually, we first envisage to trigger the *ERC* sending once we overhear that an eligible node sent the lost packet but we notice that this would imply an excessive number of unnecessary retransmission if the *ERC* is sent too late. This case is explained by figure 6.5 and 6.4.

- *ERC* is sent once a lost packet is retransmitted by an eligible node i.e. once a *TC* or an Implicit ACK message is received. This message is generated by the source node to the set of eligible nodes in order to cancel the retransmission request and to free their cache. Otherwise, we would have redundant data transmission which increases the risk of collision and raises the overhead in the network.

The loss of such control packets leads to frequent unnecessary retransmissions and an increased energy consumption of resource-restricted wireless nodes. Consequently, ARRP efficiency requires that control packets benefit from an increased transmission reliability. This can be achieved either by assigning higher transmit power for control packets (as compared to the transmit power assigned to data packets) or by assigning more efficient and redundant FEC to control packets. Of course, a combination of both strategies can also be proposed. It should be noted that because control packets are expected to be much shorter than data packets, the energy cost of such strategies remains affordable.

Based on these elements, we decide to combine the retransmissions and the FEC by applying a differentiated FEC schemes depending on the packet types (data packets or control packet). We rely either on a lightweight FEC for data packet while a much more efficient and redundant FEC will be used for the control packets. Indeed, using a FEC with such kind of packet does not require a considerable extra overhead since we consider control packet of only one byte. The details of implementation of this FEC on the different packets are given in section 6.

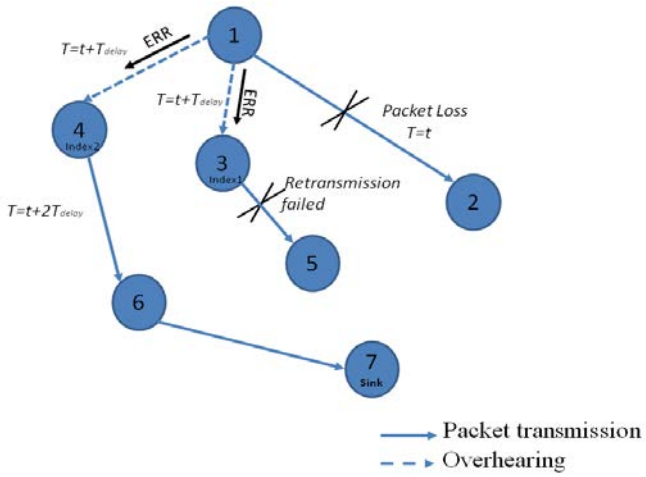


Figure 6.3: ERR packet transmission

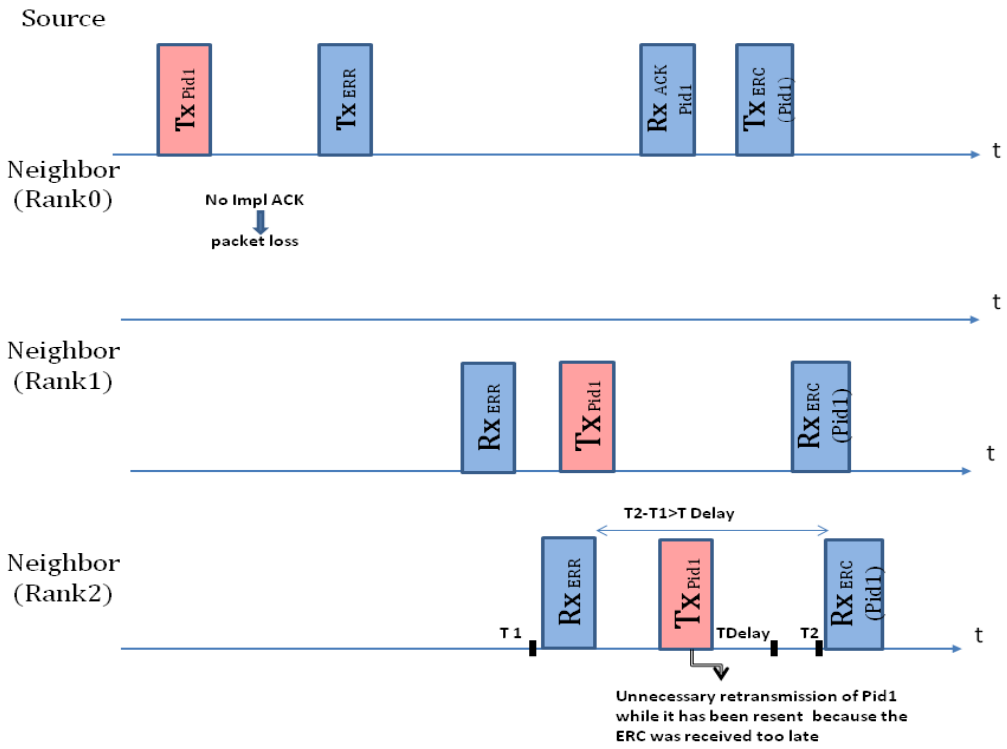


Figure 6.4: Packet duplication without TC message

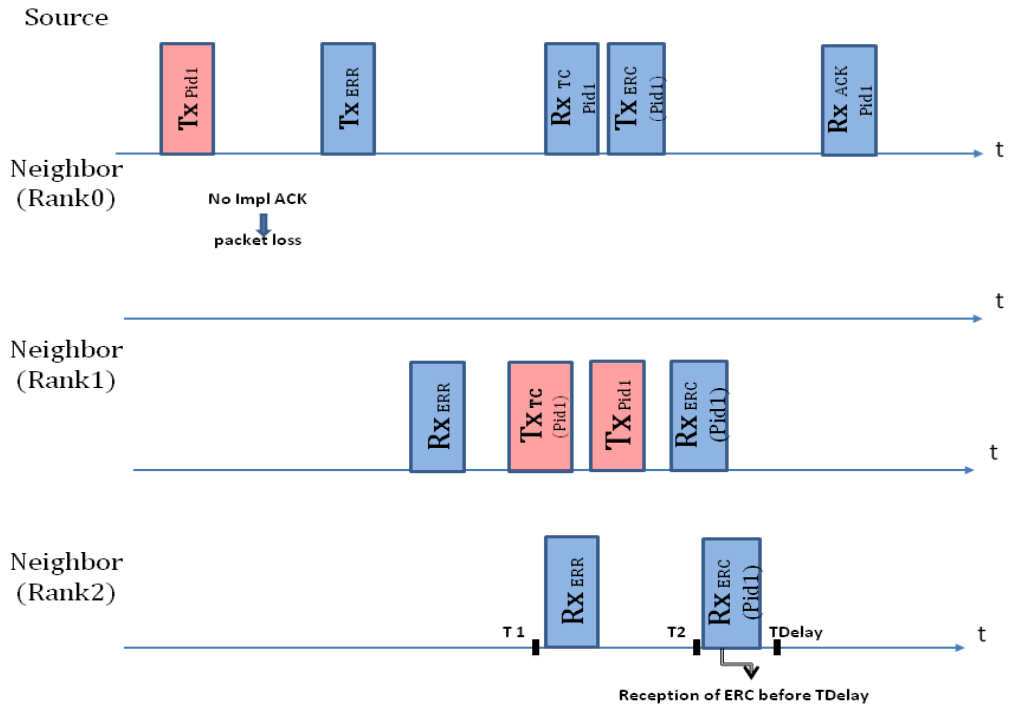


Figure 6.5: Duplication avoidance with TC message

6.5 ARRP operation

The proposed protocol acts as a routing module and provides reliable communication. The novelty of this approach relies on the retransmission triggered automatically by the neighborhood upon the detection of a link failure thanks to cooperation among the neighborhood. Every node maintains a small buffer for caching data packets that it overheard. Once a downstream node encounters a packet loss, an upstream node with the same data in its buffer can retransmit the packet. For this strategy to be effective, nodes must collaborate efficiently and schedule the retransmission task. Having developed the basic concept of our protocol, we now present 3 scenarios that illustrate the functioning of ARRP.

6.5.1 Loss Free case

In this section we consider the case where there is no packet loss. Let's revisit figure 6.2 with nodes B, C, D and E. If all the nodes (C, D and E) receive the packet *PID*, node D (which has index 0 for B), creates a primary context for *PID*. Also, nodes C and E both create a secondary context for packet *PID*. Because node D has created a primary context for PID, it immediately forwards the packets towards its own neighbors. At the same time, the node B overhears the packet forwarded by node D. There is an implicit acknowledgment for packet PID so the node B can release its primary context for *PID*. After T_{delay}, nodes C and E realize that node B didn't send any Explicit Retransmission Request (*ERR*) message. They can safely

get rid of their secondary context for packet PID and free their cache of packet *PID*. In the loss-free case, this process goes on until the packet *PID* reaches the sink, without involving any waiting period in any of the forwarding nodes on the path to the sink. The different messages exchange are given in figure 6.6 below.

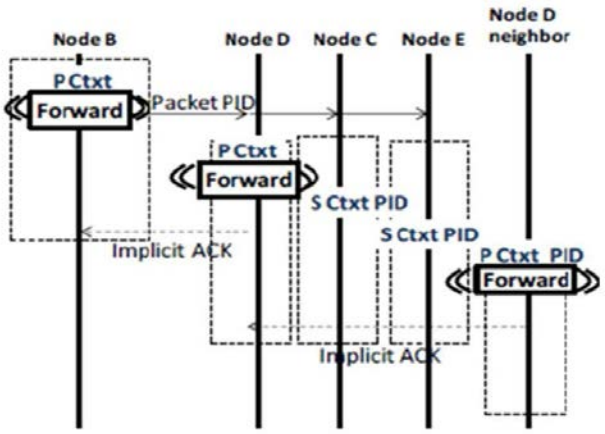


Figure 6.6: Message exchange in a loss free case

6.5.2 Packet Loss case

Now, if we consider a case including packet loss depicted by 6.7, we can come back to the situation where node B has just received packet *PID* and has just created a primary context for this packet. Again, it forwards the packet to its neighbors, but we now assume that the node D doesn't receive the packet, while nodes C and E receive it. The node C has index 1 with respect to node B so it creates a secondary context for packet *PID*. Then the node C waits for a possible Explicit Retransmission Request (*ERR*) from node B with respect to packet *PID*. T_{delay} later, node C with rank 1 turns its (*S Ctxt*) into a (*P Ctxt*). When it receives the *ERR* message for packet *PID*, It immediately sends a Taking Care (*TC*) to B and forwards this packet toward its neighbors because it has a primary context for this PID. Once node B receives *TC* message from node C, it broadcasts an Explicit Retransmission Cancel message (*ERC*) with respect to packet *PID*. The reception of this message by the neighborhood allows nodes to delete this packet *PID* from their cache and to release their context.

We may now consider another case including packet loss 6.8. We come back to the same situation as before, but we now assume that among neighbors of rank N- 1, only node E has received packet *PID* from node B. Once Node B detects the packet *PID* was not forwarded by D, it sends an *ERR* request for *PID* to C (as before), and E. However, node E does not immediately forward packet *PID* (even if it is the only node which is able to retransmit the packet) because it has a secondary context for *PID*. That is why it waits for a supplementary delay T_{delay}. Then, if no (*ERC*) message with respect to *PID* is received from node B, it turns its secondary context

for *PID* into a primary context, and it forwards packet *PID* to its own neighbors. The rule is that once a node having index n with respect to another node receives an *ERR* from this said other node, it waits for a delay equal to (n-1) times T_{delay} for a possible (*ERC*) . If no (*ERC*) is received during this time, then retransmission occurs. This process aims to avoid sending duplicate packets and consequently to reduce bandwidth consumption.

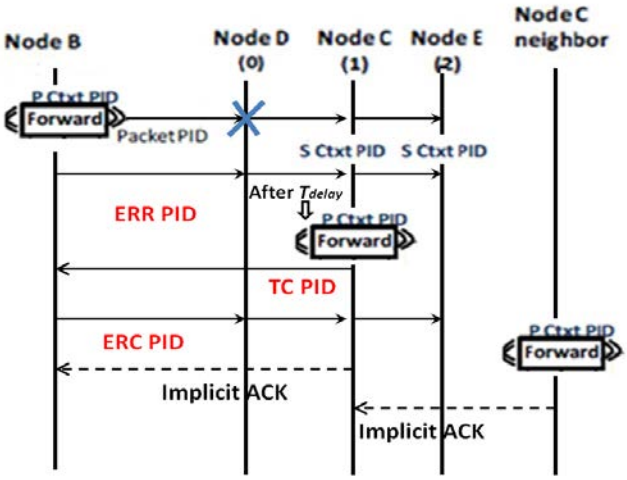


Figure 6.7: Message exchange in a loss case1

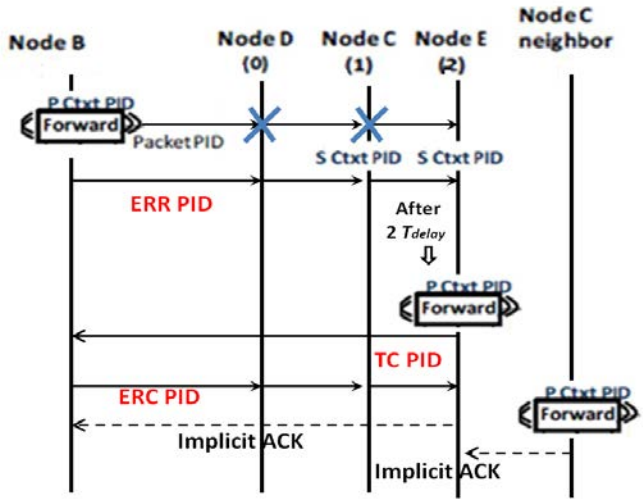


Figure 6.8: Message exchange in a loss case2

6.6 Evaluation

In this section, we study the behaviors of ARRP compared to other solutions through simulations in order to assess the benefits of our solution over different

topologies and conditions particularly in highly lossy networks. The goal of our solution is to provide high reliability against packet loss. Since we use a specific category of acknowledgment namely implicit acknowledgment we believe it is important to compare ARRP with both an explicit ACK and implicit protocols.

6.6.1 Simulation environment

In order to evaluate the gain that we can achieve with our approach, we have studied the behaviors of ARRP compared to the following protocols:

- Unacknowledged transmissions (basic)
- Link layer explicit acknowledgments and retransmissions (ACK).
- Implicit acknowledgments (ImpACK).

These results were obtained through the development of a dedicated simulator in Java which was privileged over the use of an existing networking simulation environment, such as OmNet++ (used for the evaluation of AJIA in the previous chapter) for simplicity reasons. Indeed, our simulator allows the implementation of specific network features (e.g. radio propagation model, routing protocol..), as well as graphical outputs conceived to meet our requirements. Whereas, our simulator does not implement the MAC layer which makes the use of the LQI parameter of the 802.15.4 header infeasible. Hence, we decide to replace this parameter by the distance between nodes which is a reasonable approximation as long as we deal with short distance which is our case. So this approximation does not put into question the effectiveness of our approach. In the rest of the section we will consider the parameters depicted in table 6.1.

6.6.1.1 Network topologies

Network topology has an important impact on the performance of routing protocols. To study this impact on the robustness of the different protocols, we considered randomly generated WSN topologies with varying number of nodes from 250 to 1000 in a network field with dimensions of 10000mx20000m. We consider that a link exists between two nodes if their distance is less or equal to the coverage range which is a parameter of the topology generation layout.

Parameter	Value
Field dimension (m)	20000x10000
Nb nodes	250-1000
Datarate (kbs)	256
TxPower (mW)	0.2-1
noiselevel (nW)	0.2-1
Delay between retransmission	10-30
Path Loss Exponent	3
Standard deviation σ (db)	3

Table 6.1: Simulation parameters

6.6.1.2 Shadowing

To make the model realistic, we simulate a log-normal shadowing radio propagation model where the logarithmic value of the mean signal power at different locations is normally distributed.

Another very important aspect of the wireless channel is the temporal variation called fading which is pronounced in rapidly changing environments as those experienced in a WSN.

Therefore, the total received power (Pr) in dB is given by [57]:

$$P_r(d) = P_t - P_L(d_0) - 10 * \beta * \log \frac{d}{d_0} + \chi \tag{6.1}$$

where P_t is transmission power, d is transmitter-receiver distance, d_0 is reference distance, $P_L(d_0)$ is the power decay for a reference distance d_0, β is the path loss exponent (rate at which signal decays with respect to distance) and χ is a Gaussian random variable that accounts for time-varying multipath fading and shadowing effects. The Signal-to- Noise ratio γ in db at the receiver is:

$$\gamma = P_r(d) - P_n \tag{6.2}$$

where P_n is the noise power in dB.

6.6.1.3 Link performance for data and control packets

In this subsection, we provide simulation assumption in terms of packet probability reception versus the signal to noise ratio. As stated above, we use 2 schemes for the different packet type. Figure 6.9 plots the probability of receiving a packet.

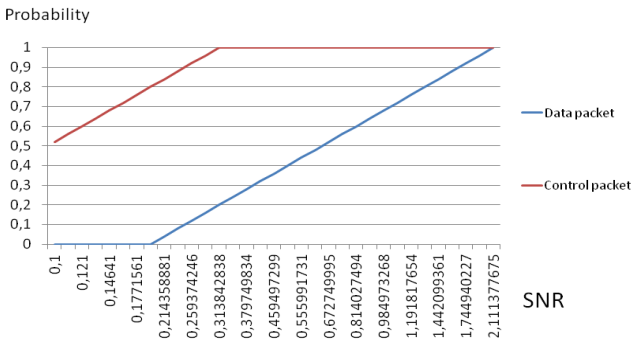


Figure 6.9: Probability of packet reception Vs SNR

Figure 6.9 shows significant performance improvement benefiting to control packet compared to data packets. As explained before, control packet carry very few content (e.g. only one byte) while data packet may carry much more content (up to 128

bytes according e.g. 802.15.4 standard). Then it is natural to propose efficient FEC to protect control packets without worrying too much about the implied additional overhead. Such performance improvement can be brought e.g. by repetition codes or more sophisticated FECs, while still maintaining much shorter control packets as compared to data packets. Though another option was to increase transmission power for control packets (as compared to data packets) .This second option has not been adopted in our simulations. Only differentiated FEC is supposed to be used with the performances described in Figure 6.9. Furthermore, the size of control packets is still assumed to 4 times shorter than the size of data packets.

6.6.2 Simulation Results

In this section, we provide performance results along with discussions and analysis with respect to some important performance criteria namely: the delivery ration, the communication overhead, the delay and the scalability.

6.6.2.1 Packet delivery ratio

As stated in the previous chapter, many applications success especially mission critical ones is contingent upon the reliable delivery of high-priority events. Therefore, the delivery ratio represents an important metric to estimate the contribution of a solution to improve the reliability. It is defined as the quotient between the number of packets received by the soink node and the number of packet sent during the simulation time.

$$\text{Delivery ratio} = \frac{\textit{Number of packets received}}{\textit{Number of packets sent}} \tag{6.3}$$

In figure 6.10, we plotted the packet delivery ratio of the different protocols for a network size of 500 while varying the transmission power level TX and the delay between successive retransmissions. As shown in this figure, ARRP achieves the best packet delivery ratio in all scenarios particularly for the highest Tx power and the lowest delay where it is equal to 98% while it is 41% for the Impl Ack protocol. Such results were predictable since reducing the delay permit to deal with more packet loss and increasing transmission power effectively improves link quality and, therefore, reduces the number of transmissions needed to deliver a packet. However, low delays may engender redundant packets. Indeed, reducing the wait time between retransmission will generate unnecessary *ERR* message leading to redundant transmission of the same packet. Figure 6.11illustrates the message sequence over time when the delay is too low. Although using the highest Tx power provide the best performance, doing so results in interference with more nodes than if reduced power were used. Another undesirable impact of the use of high transmitter power is that it results in increased energy usage. Therefore, there is a trade-off to find between the highest packet delivery ratio and the energy consumption.

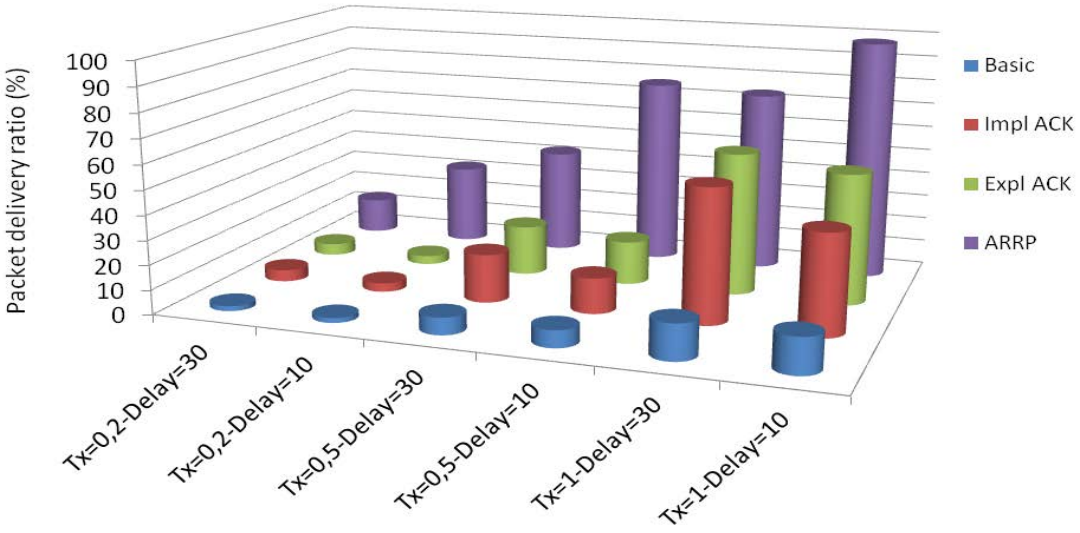


Figure 6.10: Packet delivery ratio for network size=500

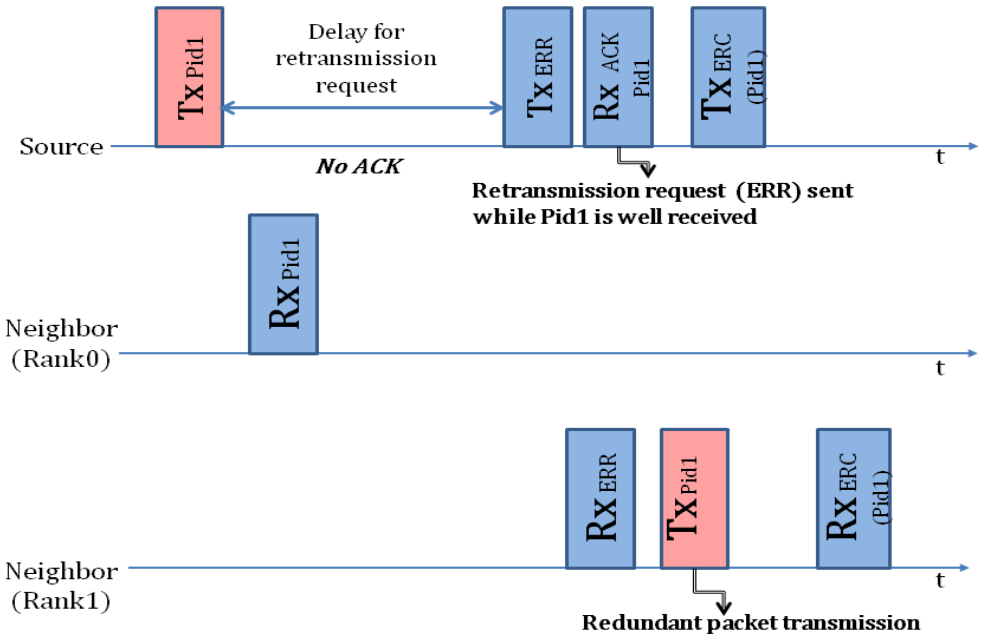


Figure 6.11: Message scheduling for low delay between retransmission

To assess the efficiency of our protocol, we also study its performance in different

loss conditions. To do so, we varied the Noise level parameter. Figure 6.12 illustrates the evolution of the packet delivery ratio for different noise level. We notice that ARRP outperforms the other protocols for all noise level especially high lossy networks (noise level=2). In the worst case scenario the reliability is 80%, and the average reliability is approximately equal to 89%, thus demonstrating that ARRP is very reliable in all network conditions.

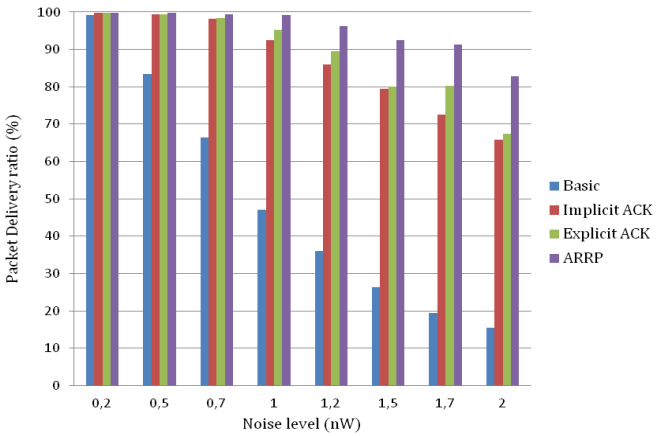


Figure 6.12: Comparative packet delivery Vs Noise level

6.6.2.2 Latency

In this experiment, we fixed the network size and varied the noise level to evaluate the latency of our protocol. We compute the average latency to deliver a packet from a source to the destination for many configurations and summarize the different results obtained after experiments in Figure 6.13. As would be expected, the delay increases as the noise level increases. Indeed, more packet losses are experienced thus more transmissions are required to deliver packets which extend the delivery delay. Moreover, we notice that ARRP has the highest latency which is due to the additional time of the control message exchange (ERC and ERR). Nevertheless, the gap is not significant for low noise level. For noise level under 1.5, the latency is very small thus the gap between the different protocols is low (≈25ms). In the worst case with noise level equals 2, we have a gap of 125 ms between ARRP and Expl ACK due to the scheduling of the multiple retransmissions required to reliably deliver the data. Expl ACK and Impl ACK outperforms ARRP, but at the expense of poor packet delivery ratio. Thereby, for high noise level, the number of retransmissions increases as the packet loss grows which increases the latency. Furthermore, if we look at these results from the temporal point of view, we notice that this supplementary delay for packet delivery is offset over time by a higher delivery ratio. Figure 6.14 plots the delay of packet delivery for the three protocols over time. We notice that bellow the threshold value of 320ms ARRP achieves the lowest number of received packet. Conversely, beyond this threshold, we notice that ARRP reaches the highest number of received packets while the other protocols

do not receive any new ones. This point confirms our hypothesis that the high latency is due to the retransmission scheduling mechanism in order to afford more reliability in the network. Consequently there is a trade-off between reliability and latency. In our solution, we chose to focus on reliability, since many applications like mission critical ones would require high reliability. Moreover, even if our protocol achieves higher latency, the supplementary delay is relatively low (≈125 ms). Also, the latency experienced remains fully acceptable.

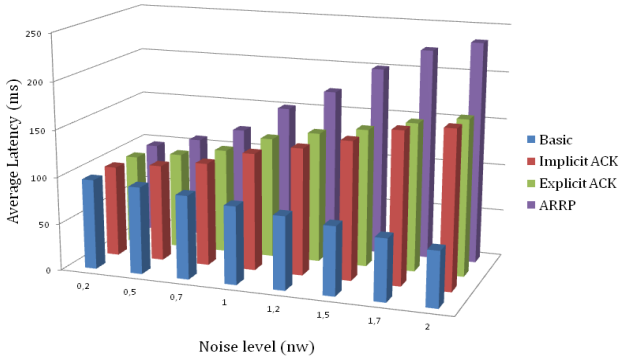


Figure 6.13: Comparative average latency

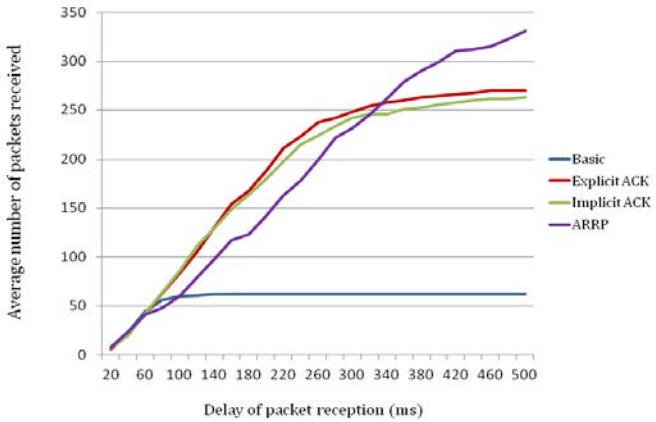


Figure 6.14: Average Number of packet received for noise level=0.7

6.6.2.3 Scalability

To guarantee large scale deployments of WSN, it is important to study the scalability of our proposed solution. We study the evolution of different parameters

in the network namely: the packet delivery, the latency and the communication overhead. The experiments are conducted while increasing the network size from 250 to 1000. Let us start by the delivery ratio which is the most important criterion in our solution. Figure 6.15 plots the evolution of this ratio with network size. Simulation results show that ARRP outperforms the other protocol for all network size. Intuitively we would expect ARRP to have better performance while increasing the network size since the set of eligible nodes will increase as the network size increases. Thus, each node would have more candidates to perform a retransmission task which reduces the probability of packet loss. However, we notice that simulated protocol behave differently and more particularly we notice a big gap between the results of network size equals 250 and the others. This is due to an important factor which is the coverage radius. All experiments were simulated with same coverage radius equals 100 for all network size. However, this parameter is not suitable for network size equals 250.

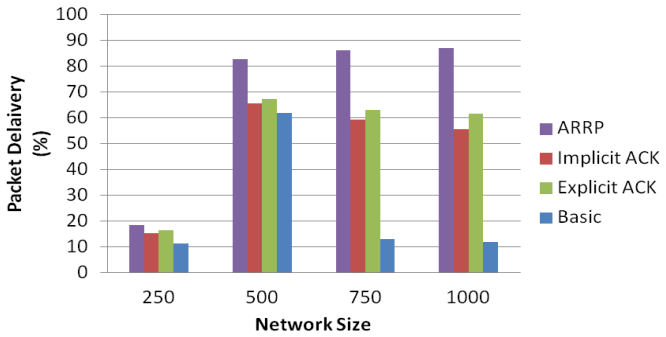


Figure 6.15: Evolution of the delivery ratio with the network size

To clarify this fact, figure 6.16shows a snapshot of our simulation results with a topology of 250 nodes and a radius equals 100. Each node has an associated level giving its hop distance from the sink. We notice that some nodes have a rank of -1 meaning that they are not able to reach the sink because of their coverage range hence all the packets generated by these nodes will be lost automatically. Moreover, we observe that some nodes have high rank even if they are not too far from the sink like the node selected in the figure 6.16 with rank 34. This is due to the same reason: a too low radius. This situation incurs high packet loss hence the results obtained in figure 6.15.

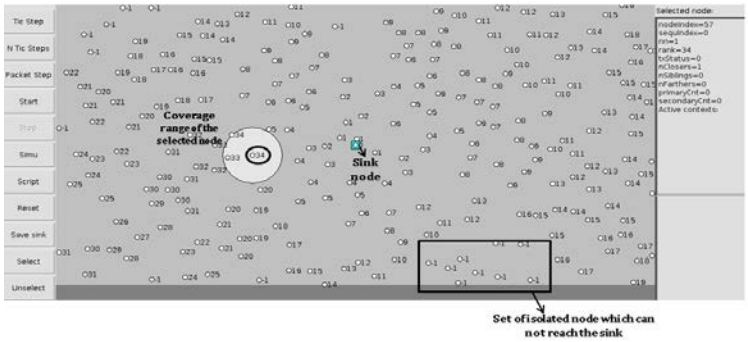


Figure 6.16: Snapshot of the network topology with 250 nodes and radius =100

Another important parameters to focus on is the communication overhead and hence the required number of control messages. Figure 6.17 illustrates the evolution of the number of exchanged messages in the network while increasing the network size. We notice that ARRP scales well because the number of messages required is not proportional to the network size. The increase of the number of control packets exchanged between a topology of 250 nodes and the one with 1000 nodes is only 50%. Therefore, ARRP is adapted to small networks as well as large groups without inducing a high bandwidth overhead.

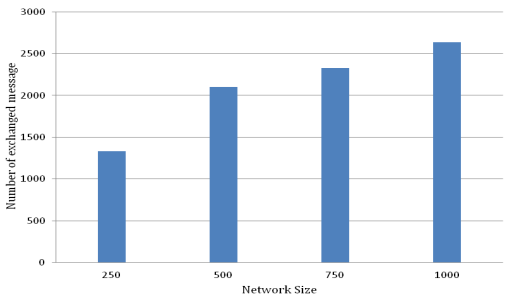


Figure 6.17: Communication overhead for ARRP

In figure 6.18 we studied the evolution of ARRP latency while varying the network size. As stated in the previous section, the highest latency is experienced by ARRP for all network size except for network size equals 250. This is due to an important factor which is the coverage radius. In order to provide the most realistic results, we choose to launch our simulations with the same radius value for all network sizes which is not always suitable. For example for a network of 250, the radius used was too low which created an important number of isolated nodes which cannot reach the sink. Consequently, only close nodes to the sink are considered in the

figure below for the average latency. That's why we experience the lowest latency for ARRP with this network size. Moreover, we notice that increasing the network size does not imply an important extra delay. The gap is about 50 ms between the network size of 250 and 1000 which make it suitable to emergency application and real time ones.

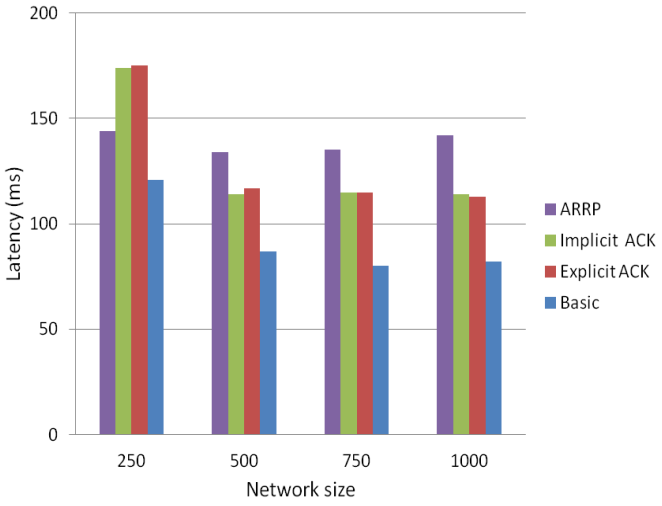


Figure 6.18: Average latency while varying network size

6.7 Conclusion

In this chapter, we have explored version variant of AJIA protocol presented in the previous chapter. Similarly to AJIA, ARRP relies on overhearing and adaptive network retransmissions to provide reliability in the network. However, ARRP exploits the node cooperation to schedule an autonomous retransmission. Through simulation experiments, we were able to verify that our protocol effectively gives much higher reliability than its competitors and scales well in large networks without inducing extra latencies and communication overheads. As a future work, we intend to further analyze the performance of ARRP in terms of energy consumption and to compare it to other solutions of the state of the art.

Chapter 7

Conclusion and Perspectives

7.1 General conclusion

The great progress in wireless communications and Micro-Electro-Mechanical systems coupled with the low power wireless communication technologies have prompted the fast expansion of WSNs. Thanks to their sensing, computation and communication abilities, WSNs covered a large range of applications as depicted in chapter 1. In emergency applications, critical data collected by the sensor nodes needs to be reliably delivered to the sink for successful monitoring of an area. Therefore, given the nature of error prone wireless links and node failure, this delivery can be hindered. Thus, ensuring reliable transfer of data from resource constrained sensor nodes to the sink is one of the major challenges in WSNs.

The research works presented in this thesis were guided by a strong idea: In order to operate efficiently; WSN needs autonomous and reliable network architecture upon its deployment. We have investigated the need for this reliable network architecture through the study of the characteristics, the challenges and the application requirements of WSNs in the first chapter.Then we identified the causes of erroneous packet transmission and review the existents techniques of reliability. Two major factors thwart the reliable deployment of WSNs: the packet loss and node failure.

The first axis of our work focused on recovering from node failures. In chapter 2, we proposed a scalable and distributed approach for network recovery from node failures in WSNs called CoMN2. We first review the existent approaches to deal with node failure. Then, we detail our protocol operation relying on a new concept called network mapping that consists in partitioning the network into several regions of increasing criticality. Using this network mapping, our solution CoMN2 seeks into maximizing network lifetime by efficiently swapping nodes from low critical area to highly critical area when required. In this solution, we make use of a mobile entity which carries on the swap of nodes. Finally, we present the simulation results and show that by using CoMN2 in a WSN with 600 nodes spread over 10000m x 20000 m, the enhancements yield a 40% improvement in packet delivery performance over the baseline.

The second axis of our thesis dealt with recovery from packet loss. In chapter 3, we introduced a lightweight mechanism for packet loss recovery and route quality awareness in WSNs called AJIA. This protocol exploits the overhearing feature

characterizing the wireless channels as an implicit acknowledgment (ACK) mechanism. In addition, the protocol allows for an adaptive selection of the routing path by achieving required retransmissions on the most reliable link. Then, we described our approach for implementing the simulations, and we presented the simulation results in chapter 4. The reported results have shown that AJIA outperforms its competitor AODV in terms of delivery ratio for all channel variation scenarios: varying the Tx power and the standard variation. On the other side, AJIA experiences limited packet loss in comparison to AODV (30% Vs 70%) for the highest standard variation value thanks its recovery process while preserving an acceptable latency. Therefore, we have proven that our protocol performs well in all conditions in term of delivery ratio, overhead, latency and acceptable energy expenditure.

In the chapter5, we proposed our protocol ARRP which is a variant of AJIA. In the design of our protocol, we combined the strengths of retransmissions, node collaboration and FEC in order to provide a reliable packet loss recovery scheme. We started by a review of node collaboration schemes in WSNs. Afterward, we specify our protocol operation. The novelty of this approach relies on the retransmission triggered automatically by the neighborhood upon the detection of a link failure thanks to nodes cooperation. Indeed, we have conducted intensive simulations considering different scenarios, topologies and demonstrated that our solution grants higher reliability.

7.2 Perspectives and future works

The works presented in this thesis are in progress and much still remains to be done. As future work, we plan to reduce the network recovery delay of CoMN2 by investigating the optimal routing path followed by the mobile node. We intend to propose a heuristic for finding good trajectories of the mobile node in large-scale WSNs. On the other side, we aim to further analyze the performance of AJIA through the study of energy consumption and to modify the MAC layer in order to achieve more energy saving. In addition, we intend to implement ARRP in Castalia and to compare its performance with those of AJIA.

Our ultimate goal is to provide a full suite of protocols for emergency application with high requirements. In this perspective, we aim to elaborate an efficient protocol combining loss packet recovery and maintaining connectivity in the network as well as cross-layer optimizations to provide energy efficiency.

Let us conclude this thesis with an open issue which is not specifically related to the presented study but rather to the future evolution of WSN namely the internet of things [5] [96] (IoT). The IoT paradigm is gaining substantial ground in modern wireless telecommunications. Thanks to technology innovation, wireless broadband connectivity is turning out to be affordable and ubiquitous. This advance brings about the proliferation of connected devices through internet. The IoT describes a vision where heterogeneous objects like computers, sensors, Radio-Frequency IDentification (RFID) tags or mobile phones are able to communicate and cooperate efficiently to achieve common goals thanks to a common IP addressing scheme. The major strength of the IoT idea is incontestably the high impact it will have on our

daily life.

Several standards are currently involved in the development of solutions for IoTs fulfilling its technological requirements and acting as a bridge between the physical world and the Internet for the IoT. The most used are ZigBee [101] and 6LowPAN [1]. We believe that ideas introduced in this thesis would be of great interest in the context of Internet of things. Thus, special care should be taken while designing new protocols in order to make them worthwhile in such heterogeneous environment.

Bibliography

[1] 6lowPan. `https://www.6lowpan.net/`.

[2] A.E.Khandani, J.Abounadi, E. Modiano, and L.Zheng. **Cooperative Routing in Static Wireless Networks**. *IEEE TRANSACTIONS ON COMMUNICATIONS*, pages 2837–2842, 2007.

[3] I. F. Akyildiz, W.Su, Y. Sankarasubramaniam, and E. Cayirci. **Wireless sensor networks: a survey** . *Journal Computer Networks:The International Journal of Computer and Telecommunications Networking*, pages 393–422, 2002.

[4] D. An and H. Cam. **Route recovery with one hop broadcast to bypass compromised nodes in wireless sensor networks**. In *IEEE Conference in Wireless Communications and Networking Conference, WCNC*, pages 2495–2500, 2007.

[5] L. Atzori, A. Iera, and G. Morabito. **Robust Cooperative Routing Protocol in Mobile Wireless Sensor Networks**. *Computer Network*, pages 2787–2805, 2010.

[6] A. Bachir, D. Barthel, M. Husse, and A. Duda. **Hidden nodes avoidance in wireless sensor networks**. In *International Conference on Wireless Networks, Communications and Mobile Computing*, pages 612–617, 2005.

[7] Basagni.S and Carosi.A. **A detailed study of mobility model in wireless sensor networks**. *Journal of Wireless networks*, pages 831–858, 2008.

[8] S. Biswas and R. Morris. **ExOR: opportunistic multi-hop routing for wireless networks**. In *Proceedings of the ACMSIGCOMMConference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM)*, pages 133–144, 2005.

[9] M. Blagojevi, M. Nabi, and M. Geilen. **A Probabilistic Acknowledgment Mechanism for Wireless Sensor Networks** . In *In Proceeding of IEEE international conference on networking, architecture and storage (NAS)*, pages 63–72, 2011.

[10] J. Cai, X. Shen, J. W. Mark, and A. S. Alfa. **Semi-distributed user relaying algorithm for amplify-and-forward wireless relay networks**. *IEEE Transactions on Wireless Communications*, pages 1348–1357, 2008.

[11] A. Campbell, Krishnamurthy, and Lakshman. **Pump-slowly, fetch-quickly (PSFQ): a reliable transport protocol for sensor networks** . *IEEE Journal on selected area in communications*, pages 862–872, 2005.

[12] D. Chen and P. K. Varshney. **QoS Support in Wireless Sensor Networks: A Survey** . In *International Conference onWireless Networks (ICWN)*, 2004.

[13] T. M. Cover and A. A. E. Gamal. **Capacity theorems for the relay channel**. *IEEE Transactions on Information Theory*, pages 572–584, 1979.

[14] B. Deb, S. Bhatnagar, and B. Nath. **ReInForM: Reliable information forwarding using multiple paths in sensor networks**. In *Annual IEEE International Conference on Local Computer Networks. (LCN)*, pages 406–415, 2003.

[15] D. Denkovski, A. Mateska, and L. Gavrilovska. **Extension of the WSN Lifetime through Controlled Mobility**. In *The Seventh International Conference on Wireless On-demand Network Systems and Services IEEE/IFIP WONS*, pages 151–156, 2010.

[16] M. Dianati, X. Ling, K. Naik, and X. Shen. **CA node cooperative ARQ scheme for wireless ad hoc networks**. *IEEE Transactions on Vehicular Technology*, pages 1032–1044, 2006.

[17] P. Esposito, M. Campista, I. Moraes, L. Costa, and O. D. ans M.G. Rubinstein. **Implementing the Expected Transmission Time Metric for OLSR Wireless Mesh Networks**. In *Wireless Days*, pages 1–5, 2008.

[18] D. Ganesan, R. Govindan, and S. Shenker. **Highly resilient, energy-efficient multipath routing in wireless sensor networks**. *ACM SIGMOBILE*, pages 11–25, 2001.

[19] A. Ghosh, D. R. Wolter, J. G. Andrews, and R. Chen. **Broadband wireless access with WiMax/802.16: current performance benchmarks and future potential** . *IEEE Communication Magazine*, pages 129–136, 2005.

[20] A. Giuseppe, M. Contib, M. Di Francescoa, and A. Passarella. **Energy conservation in wireless sensor networks: A survey** . *Ad Hoc Networks*, pages 537–568, 2009.

[21] X. Huang, H.Zhai, and Y.Fang. **Robust Cooperative Routing Protocol in Mobile Wireless Sensor Networks**. *IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS*, pages 5278–5286, 2008.

[22] T. E. Hunter and A. Nosratinia. **Cooperation diversity through coding**. In *in Proceedings of IEEE International Symposium on Information Theory (ISIT)*, page 220, 2002.

[23] IETF. `https://www.ietf.org/rfc/rfc3561.txt`.

[24] T. Instrument. http://www.ti.com/lit/ug/swru058/swru058.pdf/.

[25] T. Instrument. http://www.ti.com/product/cc2420.

[26] J. J, D. A. Irudhayara, and J. Emerson. **Effective strategies and optimal solutions for Hot Spot Problem in wireless sensor networks**. In *International Conference on Information Sciences Signal Processing and their Applications(ISSPA)*, pages 389–392, 2010.

[27] R. Jaichandran, A. A. Irudhayaraj, and J. E. Raja. **Effective strategies and optimal solutions for Hot Spot Problem in wireless sensor networks (WSN)**. In *10th International Conference onInformation Sciences Signal Processing and their Applications (ISSPA)*, pages 389–392, 2010.

[28] S. Jain and S. R. Das. **Exploiting path diversity in the link layer in wireless ad hoc networks**. In *6th IEEE WoWMoM Symposium*, pages 22–30, 2005.

[29] T. Kavitha and D. Sridharan. **Security Vulnerabilities in wireless sensor networks: A survey** . *Journal of Information Assurance and Security*, pages 31–34, 2010.

[30] A. Khandani, J. Abounadi, E. Modiano, and L. Zhang. **Cooperative routing in wireless networks**. In *Proc. Allerton Conf. on Comm., Control and Computing*, pages 808–817, 2003.

[31] T. Korakis, S. Natayanan, A. Bagri, and S. Panwar. **Implementing a cooperative MAC protocol for wireless LANs**. In *Proceedings of the IEEE International Conference on Communications (ICC)*, pages 4805–4810, 2006.

[32] J. N. Laneman, D. N. C. Tse, and G.W.Wornell. **Cooperative diversity in wireless networks: efficient protocols and outage behavior**. *IEEE Transactions on Information Theory*, pages 3062–3080, 2004.

[33] J. N. Laneman, G. W. Wornell, and D. N. C. Tse. **An efficient protocol for realizing cooperative diversity in wireless networks**. In *in Proceedings of IEEE International Symposium on Information Theory (ISIT)*, page 294, 2001.

[34] P. Larsson and N. Johansson. **Multiuser diversity forwarding in multihop packet radio networks**. In *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC)*, pages 2188–2194, 2005.

[35] G. N. Lee and E. N. Huh. **Reliable data transfer using overhearing for implicit ack**. In *ICROS-SICE*, pages 1976–1979, 2009.

[36] G. Li, Z. Xu, C. Xiong, C. Yang, S. Zhang, Y. Chen, and S. Shugong Xu. **Energy-efficient wireless communications: tutorial, survey, and open issues**. *IEEE Wireless Communications*, pages 28–35, 2011.

[37] P. Li, X. Huang, and Y. Fang. **Capacity scaling of multihop cellular networks**. In *Proceedings of the IEEE Infocom Conference*, pages 2831 – 2839, 2011.

[38] C. Liang, X. Huang, and J. Deng. **A fault tolerant and energy efficient rou ting protocol for urban sensor networks**. In *Proceedings of the 2nd international conference on Scalable information systems*, page 27, 2007.

[39] C. M. Liang, N. B. Priyantha, J. Liu, and A. Terzis. **Surviving Wi-fi Interference in Low Power ZigBee Networks**. In *ACM Conference on Embedded Networked Sensor Systems (SenSys)*, pages 309–322, 2010.

[40] L.Liu, Z. Wang, and M. Zhou. **Energy-efficient cooperative routing for wireless sensor networks using space time block code**. In *Systems, Man and Cybernetics, ISIC. IEEE International Conference on*, pages 2837–2842, 2007.

[41] J. Luo and J. Hubaux. **Joint mobility and routing for lifetime elongation in wireless sensor networks**. In *INFOCOM*, pages 1735–1746, 2005.

[42] O. M, Al-Kofahi, and A. E. Kamal. **Network Coding-Based Protection of Many-to-One**. *IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS*, pages 797 – 813, 2009.

[43] A. Mahdy and J. S. Deogun. **Wireless optical communications: a survey** . In *Proceedings of the IEEE Wireless Communications and Networking Conference*, pages 2399–2404, 2004.

[44] M. A. Mahmood and W. Seah. **Reliability in Wireless Sensor Networks: Survey and Challenges Ahead**. *Preprint submitted to Elsevier*, pages 1–42, 2012.

[45] M. Marina and S. R. Das. **Cooperative routing in wireless networks**. In *International Conference on Network Protocols(ICNP)*, pages 14–23, 2001.

[46] M.Dehghan, M.Ghaderi, and D.Goeckel. **Minimum-Energy Cooperative Routing in Wireless Networks with Channel Variations**. *IEEE Transactions on Wireless Communications*, pages 3813–3824, 2011.

[47] Messina and Daniele. **Achieving Robustness through Caching and Retransmissions in IEEE 802.15.4-based WSNs**. In *Proceedings of IEEE International Conference on Computer Communications and Networks (ICCCN)*, pages 1117–1122, 2007.

[48] M. Michel. `http://sourceforge.net/projects/aodvcastalia/`.

[49] M.P.Durisic, Z. Tafa, G. Dimic, and V. Milutinovic. **A survey of military applications of wireless sensor networks**. In *Embedded Computing (MECO), Mediterranean Conference on*, pages 196–199, 2012.

[50] S. Mueller, R. P. Tsang, and D. Ghosal. **Multipath routing techniques in wireless sensor networks : A survey** . *Lecture notes in computer science*, pages 209–234, 2004.

[51] M. Nassr, J. Jun, S. Eidenbenz, and A. Hansson. **Scalable and reliable sensor network routing: Performance study from field deployment** . In *The 26th IEEE INFOCOM*, pages 670–678, 2007.

[52] X. Ni, K. Lan, and R. Malaney. **On the performance of expected transmission count (ETX) for wireless mesh networks**. In *Proceedings of the 3rd International Conference on Performance Evaluation Methodologies and Tools*, 2008.

[53] nicta. `https://www.castalia.research.nicta.com.au/`.

[54] omnet. `https://www.omnetpp.org`.

[55] W. W. Peterson and E. J. Weldon. **Error-Correcting Codes** . *MIT Press*, 1998.

[56] M. Petrova, L. Wu, P. Mahonen, and J. Riihijarvi. **Interference measurements on performance degradation between colocated ieee 802.11g/n and ieee 802.15.4 networks**. In *Sixth International Conference on Networking(ICN)*, page 93, 2007.

[57] P.Stuedi and G.Alonso. **JLog-normal shadowing meets SINR: A numerical study of Capacity in Wireless Networks**. In *Sensor, Mesh and Ad Hoc Communications and Networks, 2007. SECON '07. 4th Annual IEEE Communications Society Conference on*, pages 550–559, 2007.

[58] S. Qaisar and H. Radha. **Multipath distributed data reliability for wireless sensor networks**. In *In Proceeding of IEEE International Conference on Communications (ICC)*, pages 330–334, 2009.

[59] M. Radi, B. Dezfouli, K. Abu Bakar, and M. Lee. **Multipath routing in wireless sensor networks : Survey and research challenges** . *Sensors*, 2012.

[60] V. Raghunathan, C. Schurgers, and S. Park. **Energy aware wireless microsensor networks** . *IEEE Signal Processing Magazine*, pages 40–50, 2002.

[61] V. Rajendran, K. Obraczka, and K. K. Garcia-Luna-Aceves. **Energy-efficient collision-free medium access control for wireless sensor networks**. In *Proceedings of the 1st international conference on Embedded networked sensor systems (SenSys)*, pages 181–192, 2003.

[62] R.Madan, N. Mehta, A. Molisch, and J.Zhang. **Energy-Efficient Decentralized Cooperative Routing in Wireless Networks**. *IEEE Transactions on ON AUTOMATIC CONTROL*, pages 512–528, 2009.

[63] A. Sendonaris, E. Erkip, and B. Aazhang. **Increasing uplink capacity via user cooperation diversity**. In *Proc. IEEE Int. Symp. Information Theory (ISIT)*, pages 156–159, 1998.

[64] K. Sha, J. Gehlot, and R. Greve. **Multipath routing techniques in wireless sensor networks : A survey** . *Wireless Personal Communications*, 2012.

[65] Z. D. Shakir, K. Yoshigoe, and R. B. Lenin. **Adaptive buffering scheme to reduce packet loss on densely connected WSN with mobile sink**. In *Proceedings of the IEEE Consumer Communications and Networking Conference (CCNC)*, pages 439–444, 2012.

[66] N. S. Shankar, C.-T. Chou, and M. Ghosh. **Cooperative communication MAC: a new MAC protocol for next generation wireless LANs**. In *in Proceedings of the International Conference on Wireless Networks, Communications and Mobile Computing*, pages 1–6, 2005.

[67] H. She, Z. Lu, , and A. Jantch. **Analytical evaluation of retransmission schemes in wireless sensor network** . In *In Proceeding of IEEE Vehicular Technology Conference (VTC)*, pages 1–5, 2009.

[68] S. Singh, M. P. Singh, and D. K. Singh. **Routing Protocols in Wireless Sensor Networks A Survey**. *International Journal of Computer Science and Engineering Survey (IJCSES)*, pages 63–85, 2010.

[69] H. Sizun. ***Radio Wave Propagation for Telecommunication Applications***. Springer Berlin Heidelberg, 2004.

[70] D. Son, B. Krishnamachari, and J. Heidemann. **Experimental study of concurrent transmission in wireless sensor networks** . In *ACM SenSys*, pages 237–250, 2006.

[71] D. Son, B. Krishnamachari, and J. Heidemann. **Experimental study of concurrent transmission in wireless sensor networks**. In *Proceedings of the 4th international conference on Embedded networked sensor systems, ser. SenSys*, pages 237–250, 2006.

[72] V. Srinivasan, P. Nuggehalli, C. F. Chiasserini, and R. R. Rao. **Cooperation in wireless ad hoc networks**. In *IEEE INFOCOM*, pages 808–817, 2003.

[73] J. A. Stankovic, A. D. Wood, and D. HeGreig. **Realistic Applications for Wireless Sensor Networks**. *Springer Berlin Heidelberg*, pages 835–863, 2011.

[74] I. Stojmenovic and A. Nayak. **Design guidelines for routing protocols in ad hoc and sensor networks with a realistic physical layer** . *IEEE Communications Magazine*, pages 101–106, 2005.

[75] N. Tamboli and M. Younis. **Coverage-Aware Connectivity Restoration in Mobile Sensor Networks**. In *IEEE International Conference on Communications ICC*, pages 1–5, 2009.

[76] Z. Tao, T. Korakis, Y. Slutskiy, S. Panwar, and L. Tassiulas. **Cooperation and directionality: a co-opdirectional MAC for wireless ad hoc networks**. In *in Proceedings of the 5th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt)*, pages 4805–4810, 2007.

[77] J. Thelen and D. Goense. **Radio wave propagation in potato fields**. In *Proceedings of the 1st Workshop on Wireless Network Measurements*, 2005.

[78] T.Igor, H.Andreas, G.Stephan, and T.Christian. Permasense: Investigating permafrost with a wsn in the swiss alps. In *Proceedings of the ACM 4th Workshop on Embedded Networked Sensors(EmNets)*, pages 8–12, 2007.

[79] A. C. Valera, W. K. G. Seah, and S. V. Rao. **Supporting cooperative caching in ad hoc networks**. *IEEE Transaction Mobile Computing*, pages 77–89, 2005.

[80] Vasanthi.v, Kumar.R, Singh.A, and M.Hemalatha. **A detailed study of mobility model in wireless sensor networks**. *Journal of Theoretical and Applied Information Technology*, pages 7–15, 2011.

[81] H. Wang, N. Agoulmine, M. Ma, and Jin.Y. **Network lifetime optimization in wireless sensor networks** . *IEEE Journal on Selected Areas in Communications*, pages 1127–1137, 2010.

[82] J. Wang, I. J. de Dieu, A. De Leon Diego Jose, and L. Sungyoung, L ; Young-Koo. **Prolonging the Lifetime of Wireless Sensor Networks via Hotspot Analysis** . In *Applications and the Internet (SAINT), 2010 10th IEEE/IPSJ International Symposium on*, pages 383–386, 2010.

[83] J. Wang, H. Zhai, W. Liu, and Y. Fang. **Reliable and efficient packet forwarding by utilizing path diversity in wireless ad hoc networks**. In *in Proceedings of the IEEE MILCOM*, pages 258–264, 2004.

[84] J. Y. Wang, G. Yang, X. H. Lin, Z. Q. He, and J. R. Lin. **Continuous Data Collection in Wireless Sensor Networks through PNC and Distributed Storage**. In *International Conference on Wireless Communications, Networking and Mobile Computing*, pages 2568–2571, 2007.

[85] S. Wang, X. Gao, and L. Zhuo. **Survey of Network Coding and its Benefits in Energy Saving over Wireless Sensor Networks** . In *7th International Conference on Information, Communications and Signal Processing, (ICICS)*, pages 1–5, 2009.

[86] W. Wang and M. Zhao. **Joint Effects of Radio Channels and Node Mobility on Link Dynamics in Wireless Networks**. In *Infocom 2008*, pages 1606–1614, 2008.

[87] X. Wang, S. Zhong, and R. Zhou. **A mobility support scheme for 6LoWPAN** . *Computer communications*, pages 1127–1137, 2012.

[88] S. Woo and H. Kim. **Estimating Link Reliability in Wireless Networks**. In *In Proceeding of IEEE Infocom*, pages 1–5, 2010.

[89] K. Wu, Y. Gao, F. Li, and Y. Xiao. **Lightweight deployment aware scheduling for wireless sensor networks**. *Mobile Networks and applications*, pages 837–852, 2005.

[90] F. Xia. **QoS Challenges and opportunities in wireless sensor/actuator networks** . *Sensors*, pages 1099–1110, 2008.

[91] X. Xiao, L. M. Yang, and B. X. Pu. **Error-Correcting Codes** . *Computer Application*, pages 849–852, 2008.

[92] Y. Xiao, X. Li, Y. Li, and S. Chen. **Evaluate reliability of wireless sensor networks with OBDD** . In *IEEE International Conference on Communications (ICC)*, pages 1–5, 2009.

[93] Z. Xiong, Z. Yang, W. Liu, and Z. Feng. **A Lightweight FEC Algorithm for Fault Tolerant Routing in Wireless Sensor Networks** . In *International Conference onWireless Communications, Networking and Mobile Computing (WiCOM)*, pages 1–4, 2006.

[94] G. Xue, L. Zhang, and Y. Liu. **Unequal Error Protection Based on Symmetric Slepian-Wolf Coding in Wireless Sensor Network** . In *5th International Conference on Wireless Communications, Networking and Mobile Computing, (WiCom)*, pages 1–4, 2009.

[95] D. Yang, Y. Xu, and M. Gidlund. **Coexistence of ieee802.15.4 based networks: A survey**. In *36th Annual Conference on IEEE Industrial Electronics Society*, pages 2107–2113, 2010.

[96] L. Yang, S. Yang, and L. Plotnick. **How the internet of things technology enhances emergency response operations**. Technical report, Technological Forecasting and SocialChange, 2012.

[97] Y. Yang, C. Zhong, Y. Sun, and J. Yang. **Network coding based reliable disjoint and braided multipath routing for sensor networks**. *Journal of Network and Computer Applications*, pages 422–432, 2010.

[98] J. Yick, B. Mukherjee, and D. Ghosal. **Wireless sensor network survey**. *Computer Networks*, 52:2292–2330, 2008.

[99] L. Yin and G. Cao. **Supporting cooperative caching in ad hoc networks**. *IEEE Transaction Mobile Computing*, pages 77–89, 2006.

[100] J. Zhao, R. Govindan, and D. Estrin. **Computing Aggregates for Monitoring Wireless Sensor Networks** . In *Proceedings of the First IEEE international Workshop on Sensor Network Protocols and Applications*, pages 139–148, 2003.

[101] zigbee alliance. `https://www.zigbee.org`.

[102] E. Zimmermann, P. Herhold, and G. Fettweis. **he impact of cooperation on diversity-exploiting protocols**. In *Proceedings of the 59th IEEE Vehicular Technology Conference (VTC)*, pages 410–414, 2004.