APLICAÇÃO DOS CONCEITOS MUNDO PEQUENO NO PROJETO DE TOPOLOGIAS PARA REDES DE SENSORES SEM FIO HETEROGÊNEAS

DANIEL LUDOVICO GUIDONI

APLICAÇÃO DOS CONCEITOS MUNDO PEQUENO NO PROJETO DE TOPOLOGIAS PARA REDES DE SENSORES SEM FIO HETEROGÊNEAS

Tese apresentada ao Programa de Pós--Graduação em Ciência da Computação do Instituto de Ciências Exatas da Universidade Federal de Minas Gerais como requisito parcial para a obtenção do grau de Doutor em Ciência da Computação.

Orientador: Antonio Alfredo Ferreira Loureiro
Co-Orientador: Raquel Aparecida de Freitas Mini

Belo Horizonte
Junho de 2012

DANIEL LUDOVICO GUIDONI

APPLYING THE SMALL WORLD CONCEPTS ON THE DESIGN OF HETEROGENEOUS WIRELESS SENSOR NETWORKS TOPOLOGIES

Dissertation presented to the Graduate Program in Computer Science of the Federal University of Minas Gerais in partial fulfillment of the requirements for the degree of Doctor in Computer Science.

Advisor: Antonio Alfredo Ferreira Loureiro
Co-Advisor: Raquel Aparecida de Freitas Mini

Belo Horizonte
June 2012

© 2011, Daniel Ludovico Guidoni.
Todos os direitos reservados.

Guidoni, Daniel Ludovico.

G948m Aplicação dos conceitos mundo pequeno no projeto de topologias para redes de sensores sem fio heterogêneas

Daniel Ludovico Guidoni. — Belo Horizonte, 2011.
xvii, 97f. : il. ; 29cm

Tese (doutorado) — Universidade Federal de Minas Gerais. Departamento de Ciência da Computação.

Orientador: Antônio Alfredo Ferreira Loureiro
Coorientadora: Raquel Aparecida de Freitas Mini

1. Computação - Teses. 2. Redes de computação – Teses. 3. Redes de sensores sem fio – Teses. I. Orientador. II. Coorientador. III. Título.

CDU 519.6*22 (043)

UNIVERSIDADE FEDERAL DE MINAS GERAIS
INSTITUTO DE CIÊNCIAS EXATAS
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

FOLHA DE APROVAÇÃO

Projeto de redes de sensores sem fio heterogêneas utilizando os conceitos de redes small word

DANIEL LUDOVICO GUIDONI

Tese defendida e aprovada pela banca examinadora constituída pelos Senhores:

PROF. ANTONIO ALFREDO FERREIRA LOUREIRO - Orientador
Departamento de Ciência da Computação - UFMG

PROFA. RAQUEL APARECIDA DE FREITAS MINI - Co-orientadora
Departamento de Ciência da Computação - PUC-MG

PROF. LUIZ FILIPE MENEZES VIEIRA
Departamento de Ciência da Computação - UFMG

PROF. GERALDO ROBSON MATEUS
Departamento de Ciência da Computação - UFMG

PROF. EDUARDO FREIRE NAKAMURA
Departamento de Ciência da Computação - UFAM

PROFA. REGINA BORGES DE ARAÚJO
Departamento de Computação - UFSCAR

Belo Horizonte, 23 de setembro de 2011.

Dedico essa tese a todos que de alguma forma me ajudaram nessa conquista.

Resumo

Uma Rede de Sensor Sem Fio (RSSF) considera um conjunto de nós homogêneos em termos de *hardware*. Entretanto, esse tipo de rede possui baixos limites de desempenho em relação à latência durante a comunicação de dados. Outro modelo de RSSF, chamado de Redes de Sensores Sem Fio Heterogêneas (RSSFH), considera um conjunto de nós sensores heterogêneos em termos de *hardware*, especialmente em relação ao raio de comunicação e reservas de energia. Neste trabalho é proposto modelos baseados na teoria de *Small World* no projeto de Redes de Sensores Sem Fio Heterogêneas. O primeiro modelo proposto considera o padrão de comunicação em RSSF na criação de atalhos direcionados ao nó monitor da rede com o objetivo de reduzir a latência na comunicação de dados. Os pontos finais desses atalhos são nós com maior capacidade de comunicação e reservas de energia para suportar a comunicação de longo alcance. O modelo proposto foi avaliado e foi verificado que o mesmo apresenta as mesmas características de redes *Small World* (caminho médio mínimo e coeficiente de agrupamento) observadas nos modelos da literatura. Além disso, quando os atalhos são criados na direção ao nó monitor, depositando uma pequena quantidade de nós com maior capacidade de *hardware*, o modelo proposto apresenta melhores características de redes *Small World* e melhores *tradeoffs* entre latência e energia consumida durante a comunicação de dados quando comparado aos modelos da literatura. O modelo proposto também foi avaliado com relação à resiliência considerando falhas gerais e específicas e, em ambos os casos, o modelo proposto se mostrou mais robusto e apresenta uma baixa degradação da comunicação de dados na presença de falhas nos nós. Entretanto, a comunicação de longo alcance entre os nós com maiores capacidade de comunicação causa uma alta interferência no canal sem fio. Para isso, nós apresentamos um modelo para criação de RSSFH que utiliza múltiplas interfaces sem fio e a capacidade de utilização de múltiplos canais de comunicação da camada MAC para reduzir as colisões durante a comunicação de dados. Resultados de simulação mostraram que quando os atalhos são direcionados ao nó monitor e assinalados a diferentes canais de comunicação sem fio, as colisões são reduzidas e, por conseguinte, a latência na comunicação de dados é reduzida. Finalmente, nós apresentamos um *framework* baseados nos conceitos de redes *Small World* no projeto de RSSFH com qualidade de serviço. O *framework* utiliza três topologias para prover qualidade de serviço em RSSFH. Cada topologia possui o seu objetivo em relação à latência e energia consumida durante a comunicação de dados. Resultados de simulação mostraram que a utilização das topologias do *framework* proposto reduz o consumo de energia e a latência quando comparado as topologias utilizadas na literatura para prover qualidade de serviço em redes de sensores sem fio heterogêneas.

Palavras-chave: Redes de Sensores Sem Fio, Redes de Sensores Sem Fio Heterogêneas, Comunicação de Dados, Redes Small World, Qualidade de Serviço.

Abstract

A typical Wireless Sensor Network (WSN) assumes a homogeneous set of nodes in terms of capabilities. However, this kind of network suffers from poor fundamental limitations of latency during the data communication. Another model of WSN assumes a heterogeneous set of nodes with different capabilities (especially in terms of communication range and energy reserves) called Heterogeneous Sensor Networks (HSN). In this work, we propose small world models to design Heterogeneous Sensor Networks (HSNs). The first model takes into account the communication pattern of this network to create shortcuts directed to the monitoring node, decreasing the data communication latency. The endpoints of these shortcuts are nodes with more powerful communication range and energy reserves to support the long communication range. We evaluate the proposed model and show that they present the same small world features (average path length and clustering coefficient) observed in the literature models. When the shortcuts are created toward the sink node, with a few number of powerful sensors, the network presents better small world features and interesting tradeoffs between energy and latency in the data communication when compared to the literature model. Also, we evaluate the resilience of the proposed model considering general and specific failures and, in both cases, the proposed model is more robust and presents a graceful degradation of the network latency, which shows the resilience of those models. However, the long range communication used to create a shortcut causes a high interference in the wireless channel. For this, we present a model that uses multi-interface and multi-channel capability of the MAC layer to reduce collisions during the data communication. Simulation results showed that when the shortcuts are directed to the sink node and assigned to a different wireless channel, collisions and latency are reduced. We also proposed a framework based on the small world concepts to design heterogeneous sensor networks with QoS. The framework uses three different topologies to provide QoS in sensor networks. Each topology has its own objectives related to latency during communication and energy consumption. Simulation results showed that the proposed framework can reduce latency and energy consumption compared to

the topology used in the literature to provide QoS in sensor networks.

Keywords: Wireless Sensor Networks, Heterogeneous Sensor Networks, Data Communication, Small World Networks, Quality of Service.

List of Figures

1.1 Data Communication in WSN. 2

2.1 Example of graphs using the random rewiring technique 13
2.2 Example of graphs using the random addition model 14
2.3 Example of L-sensor and H-sensor 19

3.1 Creation of shortcuts when the value of p changes in the random addition model 23
3.2 Directed angulation toward the sink model 24
3.3 Creation of shortcuts when the value of p changes in the TDASM model ($\phi = 30°$) 25
3.4 Small world features of the theoretical models 31
3.5 Small world features of on-line models 33
3.6 Latency in the data communication 36
3.7 Latency in the data communication with five general failures. Figs (a)–(c) show the shortcuts in ORAM. Figs (d)–(f) show the shortcuts in ODASM 40
3.8 Latency in the data communication with general failures 41
3.9 Latency in the data communication with specific failures (50% of H-sensors). Figs (a)–(c) show the shortcuts in ORAM. Figs (d)–(f) show the shortcuts in ODASM 42
3.10 Latency in the data communication with specific failures 43

4.1 Directed region to the sink node 48
4.2 Directed Angulation toward the sink Model 49
4.3 Small world features 52
4.4 Number of collisions 54
4.5 Data communication latency 55

5.1 Framework topologies 60

5.2 Adjustable search space to the sink node 66
5.3 Examples topologies using the DASM Tree model 70
5.4 Small world features 77
5.5 Comparison among the three topologies 77
5.6 Number of components among the H-sensor network 79
5.7 Latency evaluation 80

List of Tables

3.1 Number of added links for all models when we change the probability of shortcut addition 32
3.2 Energy consumption and transmitted packets 35

4.1 Number of collisions points 53

5.1 Energy consumption in Joules. L (L-sensors) and H (H-sensors) 82

List of Algorithms

1 ODASM algorithm 29
2 Algorithm to create the homogeneous topology's routing 62
3 Algorithm to create the DASM topology's routing 64
4 Shortcut Creation of the DASM Tree model 68
5 Algorithm to create the DASM Tree topology's routing 71
6 QoS Routing Algorithm 73

Contents

Resumo **vii**

Abstract **ix**

List of Figures **xi**

List of Tables **xiii**

1 Introduction **1**

1.1 Motivation 1

1.2 Objectives 3

1.3 Contributions of this Thesis 5

1.4 Other Contributions 7

1.5 Work Organization 8

2 Background and Related Work **9**

2.1 Background 9

2.1.1 Complex Networks 9

2.1.2 Small World Networks 11

2.2 Related Work 15

2.2.1 Small World in Wireless Networks 15

2.2.2 Heterogeneous Sensor Networks 17

2.3 Final Remarks 20

3 Directed Angulation Toward the Sink Node **21**

3.1 Shortcut Creation in Wireless Sensor Networks 21

3.2 Theoretical Directed Angulation Toward the Sink Node 23

3.3 On-line Model to Design HSNs 26

3.4 Protocol Design of the On-line Model 27

3.5 Complexity Analysis of the ODASM Algorithm 29
3.6 Simulation Results 30
3.6.1 Scenarios 30
3.6.2 Small World Features Evaluation 30
3.6.3 Experimental Evaluation 34
3.6.4 Experimental Evaluation in Potential Failure Scenarios 37
3.7 Final Remarks 41

4 Multi-Channel Multi-Interface Directed Angulation Toward the Sink Node 45
4.1 Introduction 46
4.2 Shortcut Creation 46
4.3 Channel Assignment 48
4.4 Simulation Results 50
4.4.1 Scenarios 50
4.4.2 Small World Features Evaluation 51
4.4.3 Collision Evaluation 52
4.4.4 Latency Evaluation 53
4.5 Related Work 55
4.6 Final Remarks 56

5 A Framework based on Small World Features to Design HSN Topologies with QoS 57
5.1 Introduction 57
5.2 Framework Design 59
5.2.1 Introduction 59
5.2.2 Homogeneous Topology 61
5.2.3 DASM Topology 62
5.2.4 DASM Tree Topology 64
5.3 Framework Instantiation 72
5.4 Simulation Results 73
5.4.1 Scenarios 73
5.4.2 Methodology 74
5.4.3 Small World Features 75
5.4.4 Number of Connected Components 78
5.4.5 Latency Evaluation 78
5.4.6 Energy Evaluation 81

5.5 Final Remarks 82

6 Conclusion and Future Work **84**

6.1 Summary of this Work 84

6.2 Future Work 85

Bibliography **87**

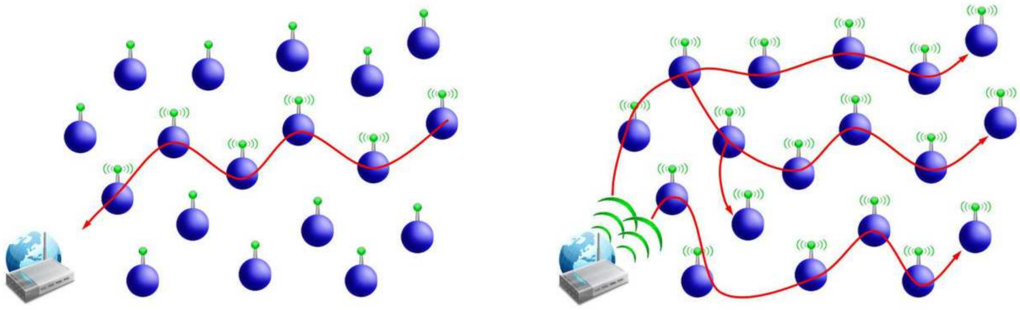
Chapter 1

Introduction

1.1 Motivation

Wireless sensor networks (WSNs) [Estrin et al., 1999; Intanagonwiwat et al., 2002; Pottie and Kaiser, 2000; Akyildiz et al., 2002] are composed of sensors nodes that collect data from the environment, where they are embedded and proceed to be processed and relayed to a monitoring node. This node often does not have resource restrictions and is responsible for collecting information about the network and sending control data to sensor nodes. Sensor nodes interconnected through wireless communication networks with the goal of performing any sensing task will improve how information is collected and processed. Wireless sensor networks need to carefully integrate functionalities traditionally considered to be separate to achieve maximum efficiency, with respect to energy consumption and management. Additionally, WSNs are evolving from simple data transportation networks to functionally rich distributed systems. Wireless sensor networks can be employed in the monitoring, tracking, coordination and processing of different applications. For instance, sensors can be interconnected to monitor and control environmental conditions in a forest, ocean or planet.

Data communication in WSNs tends to be different from "traditional" data networks in such a way that the sink node is either the origin or the destination of a message, whereas in other networks data communication happens between arbitrary communicating entities. In this case, the data communication from the point of view of the communicating entities can be divided into two types, as depicted in Figure 1.1: from sensors to a monitoring node, and from a monitoring node to sensors. The first one, called data collection (Figure 1.1(a)), is used to send the sensed data to a monitoring application. It is also used when sensor nodes have to send important control information to a monitoring node, such as their energy consumption. The second one, called data dissemination (Figure 1.1(b)), is normally used when a monitoring node has to disseminate some information to sensor nodes. Reliable data dissemination is crucial when a monitoring node has to perform some specific activities, such as to disseminate an information that is important to all sensor nodes, change the operational mode of part or the entire network, activate/deactivate one or more sensors, and send queries or new interests to the network. The sink node may also wants to change the node duty cycle. Furthermore, data dissemination is fundamental to the basic operation of many protocols in WSNs such as identification of multiple paths between sensor nodes and establishment and maintenance of routes.



(a) Data Collection (b) Data Dissemination

Figure 1.1. Data Communication in WSN.

A usual model of a WSN assumes a homogeneous set of nodes, i.e. all nodes have the same capabilities of memory, processing and communication range. However, Gupta and Kumar [2000] and Yarvis et al. [2005] show that a homogeneous ad hoc network suffers from poor fundamental limits and network performance related to latency, throughput, etc. For instance, in a multi-hop homogeneous WSN, nodes close to the sink tend to spend a significant amount of energy routing packets from other nodes, which can eventually lead to a disconnected topology.

Another model of a sensor network, named Heterogeneous Sensor Network (HSN) [Du and Lin, 2005; Mhatre et al., 2005; Du et al., 2006; Zhang et al., 2008], assumes the existence of a large number of low-end sensors (L-sensors) and a small set of powerful high-end sensors (H-sensors). The L-sensors have limited capacity of memory, energy, processing and communication range, while the H-sensors have powerful capabilities. In a HSN, due to hardware differences and mainly to greater communication capacity of H-sensors, the network exhibits a non-trivial topology structure because of the high variability in neighbor's connections, i.e. H-sensors have a high number of neighbors while L-sensors have just a few.

Complex Networks [Newman, 2003; Watts, 2007; Kleinberg, 2008] are becoming more important to model a network that has certain non-trivial topological features [da F. Costa et al., 2007] as found in small world or scale free graphs. An important model of a complex network is the small world model [Watts and Strogatz, 1998; Amaral et al., 2005; Strogatz, 2001a], which presents desired characteristics for a computer network such as small average path length between pairs of nodes and high cluster coefficient[1]. The small average path length leads to a small data communication latency and high values of clustering coefficient can improve the network resilience. In the context of a computer network, resilience means the capacity of a network to provide and maintain an acceptable quality of service (specified by the user and/or network designer) related to latency, energy consumption and others in the presence of faults [Menth and Martin, 2005]. Resilience is recognized as a fundamental attribute for truly effective use of wireless sensor networks.

In order to create a network with small world features, the designer should add a small number of long-range links, called shortcuts. In traditional networks, the goal of the shortcuts is to optimize the communication between pairs of nodes. However, in WSNs, data communication typically happens when the sink node wants to send a message to the sensor nodes (data dissemination), or when the sensor nodes send the sensed data to the sink node (data collection). Due to the specific communication pattern of wireless sensor networks, the goal of the shortcuts in such a network should be to optimize the communication between sink node and sensor nodes, instead optimizing the communication for any pairs of nodes. Thus, to improve the data communication in wireless sensor networks, the shortcuts should be created toward the sink node.

Also, when a WSN is modeled according to the small world concepts, the network becomes more robust to the presence of node failures due to the shortcut addition, i.e. the network becomes more resilient. In this case, the shortcuts may interconnect the remaining nodes, keeping alive the data communication in the network [Albert et al., 2000].

1.2 Objectives

The main goal of this proposal is to design heterogeneous wireless sensor networks based on the small world models to improve data communication in wireless sensor

[1]Clustering coefficient is a topological metric that measure the degree in which nodes in a graph tend to cluster together. The terms small average path length and clustering coefficient will be defined later in Chapter 2.

network. In this way, when a network exhibits small world features, the following issues can be improved:

- Data communication latency: since small world networks have small average path length compared to regular networks, if wireless sensor networks have small world features, data communication latency can be improved.
- Network resilience: a small world network has high values of clustering coefficient. High values of clustering coefficient indicates that the network is resilient in presence of node failures.
- Transmitted messages: the shortcuts reduce the number of messages during the routing. Since the shortcuts are long range connections, the total number of transmitted messages during the routing can be reduced compared to the network without long range connections.
- Energy Consumption: By reducing the number of transmitted messages during data communication, we also reduce the amount of energy spent during this process.

Based on these issues, we are investigating the best way to create a heterogeneous sensor network with small world features to improve the data communication. The specific objectives of this thesis are:

- Propose a theoretical model to design heterogeneous wireless sensor networks that can be used to improve the data communication in wireless sensor networks. The created shortcuts in the network toward the sink node improve data communication in this kind of network. Also, the theoretical model lead us to a planned deployment of the shortcuts.
- Propose an on-line version of the theoretical model. The on-line model will create the small world topology during the network start up. For this, a planned deployment of the wireless shortcuts is not necessary, as observed in the theoretical model.
- Propose an on-line model that creates one shortcut per H-sensor and because of this feature, we can apply the channel assignment problem to assign a different wireless channel for each shortcut in the network. The shortcuts are long-range wireless communication. In this way, when a message is transmitted by a shortcut, the interference in the wireless channel will be high. Based on this

observation, each shortcut can use a different wireless channel to avoid wireless interference during the long-range communication.

- Propose a framework to design heterogeneous sensor networks with QoS. The proposed framework is able to create more than one topology using the same H-sensor infrastructure. With this feature, we can create topologies with different objectives and each topology provides different levels of QoS in terms of latency and energy consumption.

1.3 Contributions of this Thesis

The student has been working on his thesis and some papers were published/accept for publication. Each paper introduces some concept/model described in this text. The first work related to this thesis was published in *IEEE 19th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC '08)* [Guidoni et al., 2008a]. In this paper, the student investigated the small world concepts in homogeneous wireless sensor networks. This work does not investigate the design of heterogeneous sensor network and the shortcuts are created increasing the communication range of the L-sensor. Because of this, this work is not used anymore in this thesis. However, the ideas behind that model is used in the proposed models in this thesis, such as the directed angulation toward the sink node and the unicast shortcuts.

In the second paper, published in *ACM/IEEE 11th International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM '08)* [Guidoni et al., 2008b]. The student proposed a theoretical small world model to design heterogeneous sensor networks. In this scenario, the endpoints of the shortcuts are powerful nodes. The student also proposed another model to design heterogeneous sensor network called SSD (Sink as Source or Destination). In this model, the shortcuts are created between the H-sensors and the sink node. However, this model is not used anymore in this thesis, since the H-sensors have limitations related to a single hop communication between them and the sink node.

The model proposed in [Guidoni et al., 2008b] is theoretical. In this case, it is not applicable to create a heterogeneous sensor network with small world concepts during the network startup. To solve this problem, we proposed the on-line model to create a network with small world features. In this model, the powerful nodes are deployed in the network and during the network startup, they create shortcuts among them to introduce small world features in the network topology. This work was published in *IEEE Symposium on Computers and Communications (ISCC '09)* [Guidoni et al., 2009].

After published these three works, the student studied the resilience of the proposed models. In this case, some nodes or a group of nodes fails during the network execution and we evaluate the network metrics to verify if the networks can keep the small world features. This work was published in *Elsevier Computer Networks* [Guidoni et al., 2010d] and we verified that even under nodes' failures, the created shortcuts can keep small values of average shortest path and high values of clustering coefficient.

The major drawback of the proposed models until now is the use of the same wireless channel in all created shortcuts (H-sensor interface). In this case, when a H-sensor sends a message through its shortcuts, the neighboring shortcuts can not be used at the same time. So, multiples transmissions can not take place at the same time and this can increase the network latency. For this, we published a paper in *IEEE Global Communications Conference (GLOBECOM '10)* [Guidoni et al., 2010b] that studies the channel assignment problem over a heterogeneous sensor network with small world features. The authors proposed a new version of the previous model to design heterogeneous sensor networks. The new model was designed in order to apply the channel assignment problem over a network with shortcuts. In this case, each shortcut work in a different wireless channel.

All the previous models create random directed shortcuts. In this way, the topology considering just the H-sensors is not connected. So, during data communication, both L-sensors and H-sensors are used to route the message. Based on this consideration, we propose a Tree-based model. In this model, the shortcuts among the H-sensors are created in order to create a connected (tree) topology among them. In this case, when a message arrives at any H-sensor, it will be forwarded using just shortcuts. The proposed model was published in *P2MNET - proceedings of IEEE Local Computer Networks* [Guidoni et al., 2011]. After, we put together the homogeneous, DASM and DASM-tree topologies to propose a framework to design heterogeneous wireless sensor networks with QoS. This work was accepted for publication in *IEEE Symposium on Computers and Communications (ISCC '12)* [Guidoni et al., 2010c]. Finally, we studied the Kleinberg model to create a HSN with small world features. This work was accepted for publication in IEEE Communications Letters.

1.4 Other Contributions

In parallel with his research topic, the student has been participating as a co-author in other works related to his major research topic: Wireless Sensor Networks. As a result of this interaction some works were published/accept for publication/submitted and are listed below.

- do Val Machado et al. [2007], published in *10th ACM/IEEE Symposium on Modeling, analysis, and simulation of wireless and mobile systems (MSWiM '07).* This work proposes a gossiping-based data dissemination protocol in Wireless Sensor Networks.

- Souza et al. [2008], published in *XL Simpósio Brasileiro de Pesquisa Operacional (SBPO '08).* The work proposes a GRASP based algorithm to generate small world topologies.

- Maia et al. [2009], published in *12th ACM/IEEE International conference on Modeling, analysis and simulation of wireless and mobile systems (MSWiM '09).* This work uses the DASM model proposed in [Guidoni et al., 2008b] to propose an over-the-air programming protocol for wireless sensor networks.

- Villas et al. [2010], published in *13th ACM/IEEE International conference on Modeling, analysis and simulation of wireless and mobile systems (MSWiM '10).* The work proposes a scalable and dynamic data aggregation aware routing protocol for wireless sensor networks.

- Guidoni et al. [2010a], published in *IEEE Symposium on Computers and Communications (ISCC '10).* This work studies the synchronization problem under a heterogeneous sensor network with small world features.

- Oliveira et al. [2011], submitted to *Elsevier Computer Networks.* This work proposes a location-free greedy forward algorithm with hole bypass capability in Wireless Sensor Networks.

- Villas et al. [2011d], published in *13th ACM/IEEE International conference on Modeling, analysis and simulation of wireless and mobile systems (MSWiM '11).* The work proposes a timespace correlation for realtime, accurate, and energy-aware data reporting in Wireless Sensor Networks.

- Villas et al. [2011a], published in *International Conference on Communications (ICC' 11)*. The work proposes a dynamic and scalable routing to perform efficient data aggregation in Wireless Sensor Networks.

- Villas et al. [2011c], published in *Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (SBRC' 11)*. The work explores the spacial correlation in wireless sensor networks.

- Villas et al. [2011b], published in *WLN - proceedings of IEEE Local Computer Networks*. This work proposes an energyaware spatial correlation mechanism to perform efficient data collection in WSNs

1.5 Work Organization

This thesis is organized as follows. Chapter 2 presents the background and related work. This chapter is divided as follows. Firstly, we present the Complex Networks theory followed by a Small World Networks section that describes the small world features and models. Next we present the small world models in wireless networks. Finally, we present a Heterogeneous Sensor Network section to briefly describe the heterogeneous sensor networks. Chapter 3 presents the Directed Angulation Toward the Sink Node model and Chapter 4 presents the Multi-Channel Multi-Interface Directed Angulation Toward the Sink Node model. Chapter 5 presents the framework based on the small world concepts to design Heterogeneous Sensor Network with QoS. Finally, Chapter 6 presents the conclusion and future work.

Chapter 2

Background and Related Work

This work aims to use the small world concepts to create Heterogeneous Sensor Networks. Based on this, Section 2.1 presents the background to better understand the small world concepts and features. Section 2.2 presents the literature small world models to design wireless networks and Heterogeneous Sensor Networks. Finally, Section 2.3 presents the conclusions.

2.1 Background

2.1.1 Complex Networks

In the context of network theory, a complex network is a network (graph) with non-trivial topological features [Strogatz, 2001b] – features that do not occur in simple networks such as lattices or random graphs. Complex Networks are found in the real world in different areas of science, such as social, biological, technological and information. Some examples include the Internet, WWW, neural networks, metabolic, literature cross reference, friendship networks, among others [Newman, 2003; Watts and Strogatz, 1998; Egerstedt, 2011; Boasa et al., 2011]. After these works, networks have been used to model and simulate complex interactions between elements of a system, providing support for a better understanding and analysis. It is important to emphasize that complex networks differ from complicated systems/networks. Large systems/networks can be considered complicated, although their components and behavior are well known. However, complex networks show diverse behaviors, not always known or predictable.

It is well known that networks have been studied in graph theory since the 18th century, when Leonard Euler formulated the Königsberg Bridge Problem as a graph [Newman et al., 2006]. The Complex network novelty lies in the fact that formerly, these networks could be directly inspected not only for the small size of the instances, but also their static structure. However, the networks' large size and scope have changed these paradigms, together with the assumption that networks evolve and change in time.

Primarily, the complex networks have been investigated by the Physics and Sociology communities, but recently, it has gained the attention of other fields, including Computer Science. This fact has led to the development of a new branch of science called "Network Science" [Barabási, 2003]. In this direction, many challenges still exist, one of which aims to uncover which factors naturally lead to complex network development and why their characteristics may be considered efficient. Therefore, the understanding of the relationship between the structure and function of complex networks is certainly of great importance.

Two network types widely studied in Computer Science are known as small world networks [Watts and Strogatz, 1998; Reka and Barabási, 2002; Watts, 2004; Kleinberg, 2000a] and scale-free networks [Newman, 2003; Reka and Barabási, 2002; Newman et al., 2006]. The small world concept, first introduced for social networks from Milgram's experiment [Milgran, 1967], showed that the world is small because a person knows all other people in the world, directly or indirectly through few intermediaries. In [Watts and Strogatz, 1998; Newman and Watts, 1999], the authors formalized the small world concept and defined small world networks. Small world graphs share features of regular graphs (high clustering coefficient) and characteristics of random graphs (small average shortest path length). In a scale-free network, the degree distribution follows a power law, at least asymptotically. In other words, for large values of k, the fraction $P(k)$ of nodes in the network having k connections to other nodes is defined as $P(k) \sim k^{-\gamma}$ where γ is a constant whose value is typically in the range $2 < \gamma < 3$. Moreover, the degree distribution certainly will deviate from a Poisson distribution, even when $P(k)$ shows an exponential tail [Reka and Barabási, 2002]. Based on all these features, the Small World networks have appropriate features in the design of sensor networks, such as small average path length between a pair of nodes and high clustering coefficient. In this work, we will explore the small world concepts in the design of heterogeneous wireless sensor networks, since this model exhibits important features in the design of sensor networks, such as small average path length and high clustering coefficient.

2.1.2 Small World Networks

In this section we discuss small world networks. Firstly, we discuss the small world history and features. Following this, we present the small world models to design small world networks. Finally, we show the literature works that apply the small world concept in the design of wireless networks.

2.1.2.1 Small World Features

A small world network receives its name by analogy with the small world phenomenon, which is also known as the six degrees of separation. We can trace the origins of the small world hypothesis to the work of Hungarian writer Frigyes Karinthy, who in 1929 published a short story titled "Chains" (part of the volume "Everything is Different"), which describes the idea of degrees of separation. The small world phenomenon is the hypothesis that the chain of social acquaintances required to connect one arbitrary person to another arbitrary person anywhere in the world is, in general, short. In 1967, Stanley Milgram tested experimentally this hypothesis, which led to the famous phrase "six degrees of separation" [Milgran, 1967].

In 1998, Watts and Strogatz [1998] published the first small world network model that uses a single parameter to smoothly interpolate between a regular graph and a random graph. In a regular graph, each vertex has the same number of neighbors. This graph has a high clustering coefficient and a high average path length. In a random graph, the probability of connecting any pair of vertices on the graph is the same. This graph has small clustering coefficient and small average path length. They show that adding only a small number of long-range links to a regular graph, in which the diameter is proportional to the size of the network, it can be transformed into a small world model that has a very small average number of edges between any two vertices and a large value for the clustering coefficient.

The average path length can be defined as follows. Let $G(V, E)$ be a graph where V is the set of nodes and E is the set of edges. The length of a path connecting the vertices $i, j \in V$ is given by the number of edges along that path. The shortest path between vertices i and j is any of the paths connecting these two nodes whose length is minimal. The average shortest path length (L) for all nodes in a network can be obtained by equation 2.1, where SP_{ij} is the shortest path between i and j, $i \neq j$, n is the total number of nodes and the pairs of nodes that are not related in the same component are disregarded.

$$L = \frac{1}{n(n-1)} \sum_{i=1}^{n} \sum_{j=1}^{n} SP_{ij} \tag{2.1}$$

This coefficient represents the density of triangles in the network, which can be illustrated as follows: suppose we have vertices $i, j, m \in V$ and edges (i, j) and (j, m). The clustering coefficient defines the probability of having the edge (i, m) in the network forming a "triangle" among the three vertices. Formally, this coefficient can be describes as follows. The cluster coefficient of a node i (with degree $k_i > 1$) is given by the ratio between e_i, that defines the number of edges among the neighbors of i and the maximum number of edges among theses vertices, that is $\frac{k_i(k_i-1)}{2}$. Thus,

$$CC_i = \frac{2e_i}{k_i(k_i-1)}. \tag{2.2}$$

The global clustering coefficient of the network is obtained by equation 2.3, where n is the number of nodes.

$$CC = \frac{1}{n} \sum_{i=1}^{n} CC_i \tag{2.3}$$

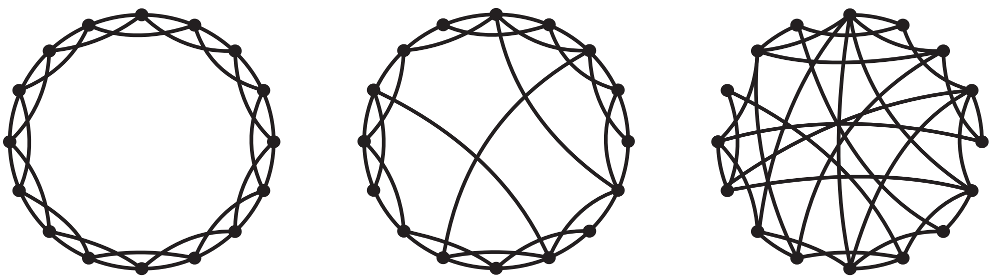
Many different abstract graphs exhibit the small world property, such as networks of scientific collaboration and the World Wide Web [Kleinberg, 2000b; Watts, 2004; Watts et al., 2002]. Given a directed Web graph consisting of N vertices and the shortest path d between two documents (vertices), Albert et al. [1999] showed that the Web is a highly connected graph that has an average diameter of 19 links. Furthermore, they show that there is a logarithmic dependence of d on N that is important to understand the growth of the Web. For instance, if the Web increases by 1,000%, the value of d will increase from 19 to just 21.

2.1.2.2 Small World Models

In [Watts and Strogatz, 1998], the authors proposed a model to construct small world networks. This model consists of a network of vertices whose topology is that of a regular graph, with the addition of a low density of connections between randomly-chosen pairs of vertices. This regular network is rewired to introduce an increasing amount of disorder that can be highly clustered, like regular graphs, yet having characteristics of short path lengths, like random graphs.

The random rewiring technique, proposed by Watts and Strogatz [1998], starts by choosing a vertex from the regular lattice and an edge that connects it to its nearest neighbor. Next, with probability p, this edge is reconnected to a vertex chosen uniformly at random over the entire ring, with duplicate edges forbidden. This process is repeated by moving clockwise around the ring, considering each vertex in turn until one lap is completed. Next, the edges that connect vertices to their second-nearest neighbors are considered clockwise. As before, each of these edges are rewired with probability p. This process continues circulating around the ring and processing outward to more distant neighbors after each lap, until each edge in the original lattice has been considered once. In this process, the number of edges and vertices remains constant, no two vertices can have more than one bound running between them, and no vertex can be connected by a bound to itself. By rewiring bounds, this construction allows us to tune the graph between regularity ($p = 0$) and disorder ($p = 1$), and thereby to probe the intermediate region $0 < p < 1$. The rewired bounds will be referred to as "shortcuts".

As an example of using the random rewiring technique to create small world networks, consider $p = 0$ in which the original ring is unchanged as illustrated in Figure 2.1(a). As p increases, the graph becomes increasingly disordered having small world characteristics as illustrated in Figure 2.1(b). When $p = 1$, all edges are rewired randomly as illustrated in Figure 2.1(c). The main result of this rewiring process is that for intermediate values of p, i.e., $0 < p < 1$, the graph is a small world network: highly clustered like a regular graph, yet with a short path length, like a random graph. Watts and Strogatz [1998] showed that graphs of this type can indeed possess well-defined locales while at the same time possessing average vertex-vertex distances that are comparable with those found on random graphs, even for quite small values of p.



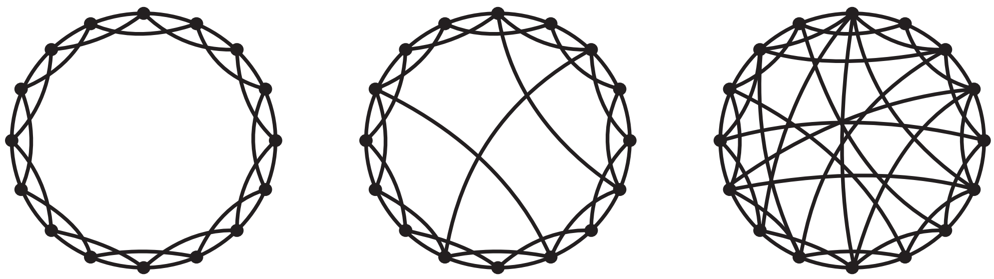
(a) Regular graph (b) Small world graph (c) Random graph

Figure 2.1. Example of graphs using the random rewiring technique

The random rewiring technique presented above has two main drawbacks. Firstly, from the practical point of view, it is not easy to remove a link in wireless networks.

Secondly, the average distance between pairs of vertices on the graph is poorly defined. The reason is that there is a probability different from zero in such a way that a portion of the graph might become detached from the rest in this model by removing links. Formally, we can represent this by saying that the distance from such a portion to a vertex elsewhere on the graph is infinite. Thus, the rewiring technique is not suitable to be used in wireless networks.

In [Newman and Watts, 1999], it is proposed a Random Addition Model (RAM) that is a new model for creating small world networks. In this model, we start again with a regular lattice, but now instead of rewiring each shortcut with probability p, we add shortcuts between pairs of vertices chosen uniformly at random, but we do not remove any shortcuts from the regular lattice. Figure 2.2 illustrates the creation of a small world graph based on the random addition technique. When $p = 0$, we have a regular graph as illustrated in Figure 2.2(a). As we increase p, the regular graph becomes a small world graph as showed in Figure 2.2(b). Finally, when $p = 1$, we have a random graph as illustrated in Figure 2.2(c).



(a) Regular graph (b) Small world graph (c) Random graph

Figure 2.2. Example of graphs using the random addition model

Notice that both models build small world networks that preserve the high cluster coefficient, as in a regular graph, and the characteristics of a short path length, as in a random graph.

The models proposed by Watts and Strogatz [1998] and Newman and Watts [1999] use a fixed probability p for rewiring or adding a shortcut in the regular graph. This probability does not consider the geographic location of the shortcut. Because of this, a shortcut may be created between two nodes that are close or two nodes that are far from each other. Based on this observation, Kleinberg [2000a] proposed a small world model where the shortcuts are created considering the geographic positions of their endpoints. Instead of using a fixed probability p in the shortcut addition, the Kleinberg model uses $p \approx \frac{1}{d_{i,j}^r}$, where $d_{i,j}$ is the distance in hops between the nodes i

and j and r is a basic structural parameter that measures how clustered the network is. In this case, the placement of the shortcuts in the network follows a long-tailed distribution [Jespersen and Blumen, 2000; Kozma et al., 2005]. In other words, the basic structural parameter defines the distance importance on the shortcut addition probability. In the Kleinberg model, when $r = 0$, we have a uniform distribution in generating shortcuts, making their creation independent of their position on the network. As r increases, the long-range shortcuts become more clustered around the node i.

2.2 Related Work

2.2.1 Small World in Wireless Networks

Wireless networks are spatial graphs in which the links between nodes depend on the communication radius, which in turn is a function of the distance between nodes. The graph of a wireless network tends to be much more clustered than random networks, and has much higher path length characteristics. As stated in [Helmy, 2003], it is possible to reduce drastically the path length of wireless networks by adding a few random shortcut links. Furthermore, Helmy [2003] showed that these random links need not be totally random, but in fact may be confined to a small fraction of the network diameter, thus reducing the overhead of creating such a network. However, Helmy [2003] did not investigate an appropriate way to create the shortcuts in wireless networks.

Cavalcanti et al. [2004] evolved the work proposed by Helmy [2003] by exploiting the small world concept to increase the connectivity in wireless ad hoc networks. They proposed using a fraction of nodes in the network equipped with two wireless interface with different transmission range (H-sensors) in order to add the long-range shortcuts. They showed that a small fraction of these special nodes can improve the network connectivity. That work only examined the average shortest path length of small world networks. However, to classify a network as a small world, it is necessary to evaluate both the shortest path length and the clustering coefficient. Furthermore, Cavalcanti et al. [2004] did not investigate the best way to add shortcuts to the network. In a wireless sensor network, this is an important issue because of its typical communication pattern among nodes (nodes–sink and vice-versa), as discussed above.

In [Hebden and Pearce, 2006, 2007], the authors presented a hierarchical clustering protocol designed for a large scale Wireless Sensor Network (WSN) called Distributed Asynchronous Clustering (DAC). The goal of the protocol is to provide an effective, low cost solution to generate a near optimal number of well separated cluster heads in a wireless deployment. The protocol solves this problem by using local knowledge of the network. As a result, the algorithm finds the best nodes that can work as cluster heads. Also, the authors show that the cluster heads may be replaced by powerful nodes and as a result the topology starts presenting small world features. However, the proposed solution is not practical in real sensor networks, since some nodes have to be replaced by others to create the desired topology.

Jiang et al. [2008] studied variable length shortcuts to construct a wireless network with small world features. The shortcuts are created using mobile nodes called data mules or data ferries. These nodes are equipped with a powerful device to support long range communication and the data mules are used to transfer data between nodes and the destination. The data mule shortcuts are generated when data needs to be forwarded and a shortcut length is determined by data demand. The authors evaluate their approach on connected and disconnected topology. In a connected topology, the communication among data mules serve as shortcuts, thus, reducing the average path length of the network. In a disconnected topology, the shortcuts serve as bridges to connect sub-networks. Simulation results show that with a few number of data mules, the average path length is significantly reduced during the data communication and the small world phenomenon can also be observed when the network topology is disconnected. However, the use of mobile nodes to collect sensor data may introduce a large delay in data collection.

Chitradurga and Helmy [2004] and Sharma and Mazumdar [2006] studied the use of wire shortcuts in wireless sensor networks. It is assumed that wireless transceivers attached to wire ends have no energy constraints. In this way, it is used a heterogeneous sensor network and the endpoints of the shortcuts are H-sensors. They showed that wires can be used as shortcuts to reduce the average hop count of the network, resulting in reduced energy dissipation per node. It is also shown that the addition of wires can significantly reduce the non-uniformity in the energy dissipation across the sensor nodes. The reduction in the per-node energy dissipation, coupled with more uniform energy dissipation across the sensor nodes can significantly increase the network lifetime. However, in many wireless sensor network applications, like tracking enemy movements and remote surveillance, it might not be possible to augment the network with wires. For these applications, adding wires to the network might be unfeasible because of the cost involved in wiring. In this work, we consider wireless shortcuts.

In [Verma C. K. and R., 2011] the authors propose the use of the small world concepts to create a wireless mesh network. The authors claimed that wireless mesh networks suffer from many disadvantages, such as rapid throughput degradation with the path length, poor capacity scaling with large networks and wireless channel related performance issues. To create a network with small world features, it was proposed three different strategies: Random Addition Strategy, Gateway Aware Addition Strategy and Gateway Aware Greedy Addition Strategy. In all strategies, to create shortcuts in the network the models add router/gateway nodes with more than one long-range wireless interfaces. These interface are used to create shortcuts among them. Also, each shortcut is assigned to a different wireless interface. In this way, to add more than one shortcut per router/gateway, it is necessary more than one wireless interface. The authors observed than adding 10% of routers/gateways, the average path length of a wireless mesh network decreases 43%. However, to create shortcuts, the proposed models use routers/gateways without constraints and this assumption is not practical in sensor network. Even considering a heterogeneous sensor network, the powerful nodes are not without any restrictions.

In the literature of wireless sensor networks, there are many other works that use the small world concepts to create a sensor network. Some of them deploy multiple sink nodes to create shortcuts among them in order to decrease the average path length during communication [Chinnappen-Rimer and Hancke, 2009, 2010; Xuejun Liu and Lu, 2010; Chen et al., 2006a; Hawick and James, 2010]. Others use the concepts of small world to create routing protocols [Xi and Liu, 2009; Ye et al., 2009; Latvakoski, 2010, 2009]. In these cases, the small world concepts were introduced in the routing in order to find the list of nodes to participate in the data communication. However, none of these works employ the small world concepts to design a heterogeneous sensor network topology with small world features.

2.2.2 Heterogeneous Sensor Networks

This Section shows the concepts os Heterogeneous Sensor Networks as well as some solutions to create general-purpose heterogeneous topologies. Heterogeneous Sensor Networks (or Heterogeneous Wireless Sensor Networks) can be understood in two ways. The first way is when we have sensor nodes with multiple sensing modalities. In this kind of heterogeneity, the nodes are equipped with different sensor devices. Some topics of interest in this field are how to place different nodes in the same sensor field or how to collect and use different type of data from diferent sensing devices. In [Kushwaha et al., 2008a,b], the authors describe an approach for target tracking in urban environments utilizing a wireless HSN, where some nodes are equipped with an acoustic sensor and others with web cameras. The work also describes briefly the components of the system for audio processing, video processing, and multi-modal sensor[1] fusion

[1]Multi-modal sensor means that a sensor node has more than one and different sensing devices.

for target tracking in urban environments. Zhou et al. [2010] present a multimodal detection and tracking algorithm for sensors composed of a camera mounted between two microphones. The algorithm uses the time difference of arrival estimation between the two microphones in the audio sensors when the camera detects an object. Lazos and Poovendran [2006] studied the coverage in this kind of network. The proposed formulation considers a heterogeneous sensing model, where sensors need not have an identical sensing capability. Moreover, the proposed approach is applicable to scenarios where the sensing area of a sensor node is not an ideal circle but instead has any arbitrary shape.

The other type of Heterogeneous Sensor Networks assume the existence of nodes with different capabilities, especially in terms of processing, memory, communication range and energy reserves. This type of network is also known as Wireless Sensor and Actor Networks (WSAN) [Akyildiz and Kasimoglu, 2004]. As previously described, in this kind of network we have at least two sets of nodes, one called L-sensors and another called H-sensors. The L-sensors represent the nodes with low hardware capabilities and the H-sensors with high hardware capabilities. Real examples of L-sensors are Mica2 [Mica2, 2005], MicaZ [Mica, 2006], TelosB [TelosB, 2006]. For instance, the MicaZ sensor node is equipped with a 8-bit Microcontroller ATmega128L (up to 16 MIPS throughput at 16 MHz) and 128 KBytes of internal memory. The transmitter has provisions for both ON-OFF keyed and amplitude-shift keyed modulation. Also, the communication range (outdoor) is up to 100 m and the bandwith is 250 kbps. On the other hand, a real example of an H-sensor is the Stargate platform [Stargate, 2006]. The stargate node is equipped with a 32-bit Intel PXA255 RISC processor (up to 400 MHz) and has 32 MBytes of flash memory and 64 MBytes of SDRAM memory. Also, the device is equipped with a 802.11a/b wireless communication, allowing long distance communication with high bandwidth. Figure 2.3 illustrates both types of nodes. In this work, we are interested in Heterogeneous Sensor Networks when the nodes have different hardware capabilities.

In [Yarvis et al., 2005], the authors defined three common types of hardware heterogeneity: *computational heterogeneity*, where some nodes have added computational power (e.g., Intel's Stargate), *link heterogeneity*, where some nodes have highly reliable long-distance communication links (e.g., 802.11 connectivity) and *energy heterogeneity*, where nodes have unlimited energy resources (e.g., connection to a wall outlet). Also, the work explores questions of where, how many, and what types of heterogeneous resources to deploy to decrease data communication latency, energy consumption and network throughput. In the energy heterogeneity, the authors claim that the nodes

(a) MicaZ node (b) Stargate node

Figure 2.3. Example of L-sensor and H-sensor

that are close to the sink node expend more energy during data communication. In this case, these nodes may be connected to a wall outlet. In the link heterogeneity, the authors show that it reduces the average number of hops that data packets take from each sensor to the sink. Since sensor network links tend to have low reliability, each hop significantly lowers the end-to-end delivery rate and the nodes with long range communication provide a highway bypass across the network, simultaneously increasing the end-to-end delivery rate. In this case, compared to homogeneous sensor networks, the heterogeneous sensor networks can improve (i) communication latency; (ii) energy consumption; (iii) end-to-end success rate and (iv) network connectivity.

The majority of Heterogeneous Sensor Network solutions are focused on algorithm solutions [Mhatre et al., 2005; Du et al., 2007a, 2006, 2007b; Cardei et al., 2008; Han et al., 2010; Yingshu Li; Chunyu Ai; Chinh Vu; Yi Pan; Beyah, 2011; Barsi et al., 2011] instead of the topology design. In this case, the solutions deploy a few number of powerful sensors (H-sensors), especially in terms of long range communication, to create a two tier routing protocol [Du et al., 2007b; Cardei et al., 2008; Mhatre et al., 2005]. These solutions create a backbone among the H-sensors and thus the communications between the sink and the sensor nodes, or vice-versa, is done via this backbone. Du et al. [2007b] present a secure and efficient routing protocol for HSNs. The protocol takes advantage of the H-sensors to create a Two Tier Secure Routing (TTSR). The protocol uses a key management scheme among the H-sensors to create a secure data communication. In [Cardei et al., 2008], the authors address fault-tolerant topology control in a heterogeneous wireless sensor network consisting of several H-sensors, used for data relaying and a large number of energy-constrained wireless sensor nodes. To introduce the fault-tolerant, a k-degree anycast topology control was proposed. Also, the authors showed two ways to solve this problem: a centralized greedy algorithm and a distributed and localized algorithm. It is important to emphasize that the literature works regarding heterogeneous sensor networks are focus on the design of routing solutions that use powerful nodes to increase/decrease some network metric, such as latency, energy consumption, throughput and others.

2.3 Final Remarks

This Chapter presents the background and related work necessary to understand the algorithms and models proposed in the thesis. Also, the related works indicated that new solutions should be proposed to fill the gap of the design of heterogeneous sensor network considering specific features of this kind of network. In Section 2.1.1, we introduced the Complex Networks concepts and their applicability in communication networks. After, we presented the Small World concepts as well as the literature models to create a network with small world features. In Section 2.2.1 we presented the small world models to design wireless networks as well as some models to design wireless sensor network. We also showed the problems and disadvantage of the existing models in the literature. Finally, Section 2.2.2 presented the heterogeneous sensor network concepts, hardware and general-purpose solutions to design heterogeneous sensor networks.

Chapter 3

Directed Angulation Toward the Sink Node

In this Chapter, we present a theoretical model and an on-line model that can be used to create a heterogeneous wireless sensor networks with small world features. First, we will introduce how to generate shortcut in wireless sensor networks. Second, we study a theoretical model in which the shortcuts are created using an angulation toward the sink node (Section 3.2). After, we present an on-line version that can use both models (the proposed model and the RAM – literature model) to build a heterogeneous sensor network with small world features (Sections 3.3 and 3.4). By on-line we mean a distributed solution that is executed when the sensor network starts operating. This is an important feature of a resilient WSN since upon the deployment of the sensor nodes, some of them may become unavailable or their physical locations may not correspond to the original planning. This is opposed to an off-line model specified beforehand, i.e., before the network starts operating, the exact number and the positions of H-sensors (by using GPS) are known. The main goal of the on-line model is to optimize and lead to a resilient communication between the sink node and sensor nodes (L-sensors and H-sensors are randomly deployed in the monitoring area). Finally, Section 3.6 presents the simulation results and Section 3.7 presents the final remarks.

3.1 Shortcut Creation in Wireless Sensor Networks

Wireless sensor networks are typically regular graphs[1]. Therefore, it is possible to decrease significantly the average minimum path length between pairs of nodes by adding

[1] If the deployment of nodes in the monitoring area is done using a uniform distribution, we can consider that the connectivity graph of the network is similar to the regular graph.

some shortcuts, i.e., introducing a set of powerful communication sensors (H-sensors). However, as wireless sensor networks are spatial networks and the communication is done by broadcasting, all neighbors in the communication range receive the packets transmitted by a given node. To introduce the small world property to a wireless sensor network, we have to use unicast communication to create the shortcuts. If the number of shortcuts per H-sensor is more than one, the H-sensors may use parallel multicast messages in order to create shortcuts among H-sensors. The endpoint nodes of these shortcuts should operate in two distinct frequencies, one for the communication among H-sensors and another one for the communication with the L-sensors. In this way, long distance transmissions will not interfere in the communication of these nodes.

In a wireless sensor network, the best way to create a long range unicast link is to increase the communication radius. However, the communication radius is the most important factor that affects the energy consumption. The minimum energy spent to transmit a signal is proportional to R^α, where R is the communication radius of the node and α is the path loss exponent ($2 \leq \alpha < 4$) [Shelby et al., 2005]. In order to support the energy consumption overhead of the shortcuts, the H-sensors (endpoints of the shortcuts) should be equipped with a more powerful hardware, with higher energy reserves and long range transmission via multiple frequencies. This configuration leads to a heterogeneous wireless sensor network. It is important to note that the H-sensor have powerful hardware but limited, i.e., they have more energy reserves but still have battery to provide the energy. In this way, it is not feasible to create one long range link to connect each H-sensor directly to the sink node.

As an example of inserting shortcuts in a wireless sensor network, consider Figure 3.1 that illustrates the shortcuts created in a WSN using the random addition model described in Section 2.1.2.2. In that network, 1000 nodes were deployed randomly, forming a flat topology in a $1000 \times 1000\,\text{m}^2$ sensor field. The communication range is 50 m and each node has an average of 8 neighbors. The endpoint nodes of the shortcuts, as presented before, are equipped with a powerful hardware allowing them to communicate in a long distance. The communication range of these sensors is 500 m, i.e., an order of magnitude higher than the other sensors. The shortcut generation is done by adding unicast communication with a probability p. When $p = 0.0001$ (Figure 3.1(a)), the network is similar to a regular graph. As we increase the value of the probability p, the original network starts showing the small world characteristics (Figure 3.1(b), Figure 3.1(c)), and later, random graph characteristics (Figure 3.1(d)).

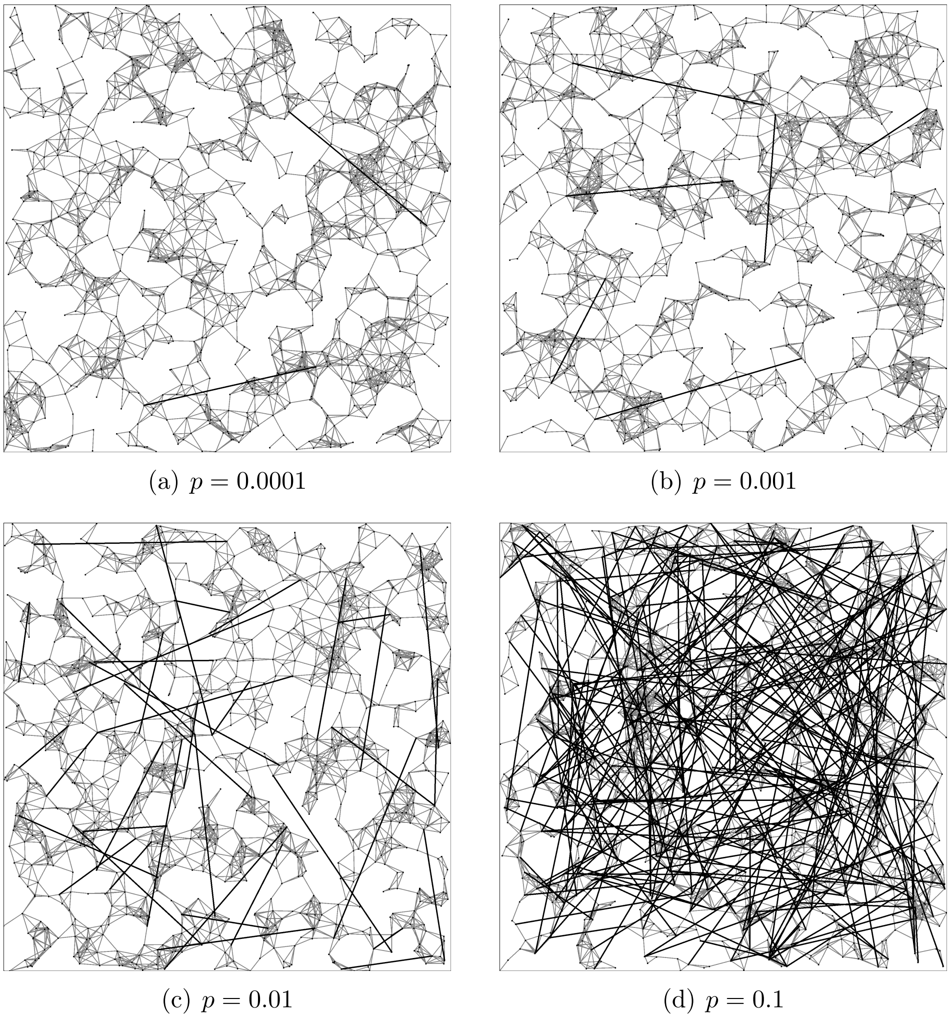


Figure 3.1. Creation of shortcuts when the value of p changes in the random addition model

3.2 Theoretical Directed Angulation Toward the Sink Node

In this work, we consider that the data communication in a WSN is either from the sink to sensor nodes or vice-versa. Thus, the shortcuts to be added should be devised considering this communication pattern to optimize the data communication between sink and sensor nodes. In this section, we present a model in which all shortcuts are generated using an angulation directed to the sink.

In the theoretical model of Directed Angulation toward the Sink node Model (TDASM), each node randomly chooses another node to create a shortcut between them. This shortcut is introduced according to a probability p and its endpoints are

H-sensors[2]. The shortcuts are created considering the straight line $y = mx + b$ that passes by the geographic position of each node and the geographic position of the sink node. This straight line is the bisectrix of an angle ϕ that defines an angle region directed to the sink node or opposite to it. Figure 3.2 shows the search space of a sensor node i. Nodes that are in the direction of the data flow between the sensor node i and the sink node, represented as black nodes, can be chosen in the shortcut creation. The search space includes the direction opposite to the sink node since this direction will also help in the data communication process.

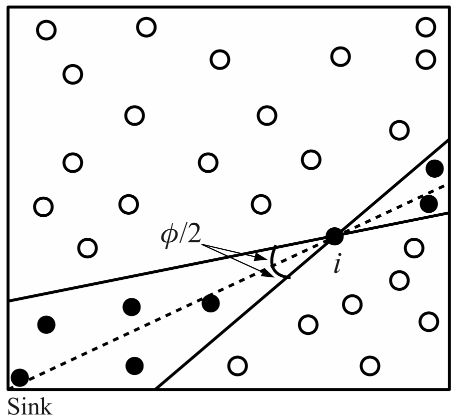


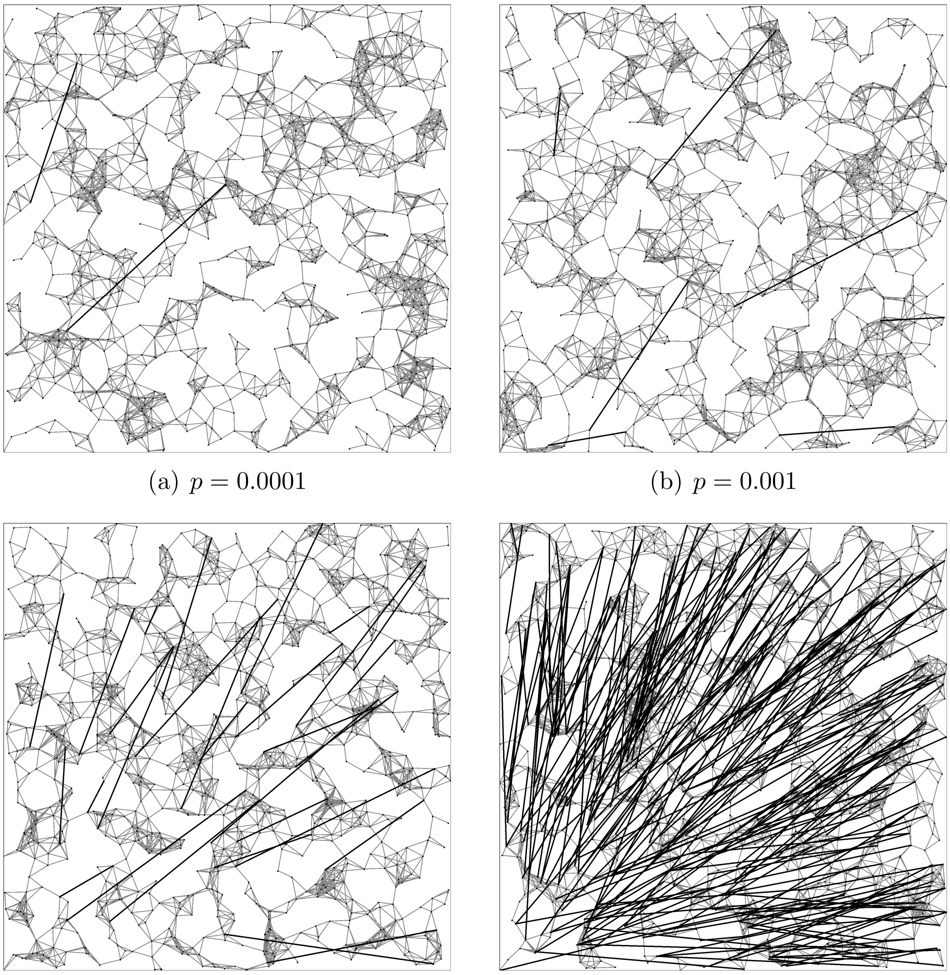
Figure 3.2. Directed angulation toward the sink model

To verify whether another node is inside its angulation region, a sensor node performs the following steps. Firstly, the node i calculates the straight line equation $y_1 = m_1 x + b_1$ that passes by its geographic location and the sink node geographic location. Secondly, the node i calculates the equation of the straight line $y_2 = m_2 x + b_2$ that passes by its geographic location and the geographic location of node k, where $k \neq i$ and node k is the node that we want to know whether it is in the angulation directed of node i. The tangent of the angle formed by the two straight lines is given by the equation: $\tan\theta = \frac{m_1 - m_2}{1 + m_1 m_2}$. Thus, the node i can determine if a given node is inside the angle ϕ directed angulation making $\tan\theta < \tan\frac{\phi}{2}$.

Using these shortcuts, we create the small world model as follows. Consider a regular graph $G(V, E)$. Given an edge $e \in E \mid e = (v_i, v_j)$, where $v_i, v_j \in V$, it is necessary to add an edge e' between nodes v_i and $v_k \in V$ in such a way that $e' = (v_i, v_k)$, where $\tan\theta = \frac{m_i - m_k}{1 + m_i m_k} < \tan\frac{\phi}{2}$, $v_i, v_k \in V$ and m_i and m_k are the angular coefficients of the straight line that passes through node i and the sink node and through node i

[2]The H-sensors nodes are equipped with GPS [Hofmann-Wellenhof et al., 1993]

and node k. The addition of this edge is done using the same probability p determined for each edge of G. These steps are repeated for all edges of G.



(a) $p = 0.0001$ (b) $p = 0.001$

(c) $p = 0.01$ (d) $p = 0.1$

Figure 3.3. Creation of shortcuts when the value of p changes in the TDASM model ($\phi = 30°$)

Figure 3.3 illustrates the shortcut addition for the model of adding directed shortcuts varying the probability, using the same topology of Figure 3.1. In these figures, the sink node is located at the lower-left corner. It is worth noticing that for all probability values, all shortcuts are created in the direction of the sink node. When the probability of adding a shortcut is 0, the sensor network is a regular graph. As the probability p increases, the sensor network starts exhibiting the small world characteristics and, after this, the network will be a random graph. To evaluate this model, in the next section, we present the values of the path length and the clustering coefficient for different values of the probability p.

3.3 On-line Model to Design HSNs

In a WSN, the random addition and the directed angulation toward the sink node models cannot be directly applied to the HSN creation due to two reasons. Firstly, in these models, to choose the endpoints of a shortcut, each node has to have the localization of all nodes in the network. In WSNs, due to the hardware and mainly energy restrictions, this assumption is infeasible because of the possibly large number of packets transmitted in the network. Secondly, in these models, a node chooses randomly another node to create a unicast link (shortcut) between them. The direct application of these models would lead to a planned deployment of H-sensors nodes, since they are endpoints of the shortcuts, which might be the case for an off-line model as discussed earlier. However, there are many WSN applications that assume a random deployment of sensor nodes, and, thus, need an "on-line" model. The random addition and the directed angulation toward the sink node are theoretical models to create networks with small world features, which we identify as TRAM (Theoretical Random Addition Model) and TDASM (Theoretical Directed Angulation toward the Sink node Model), respectively. These theoretical models can be executed before the deployment of the sensor nodes, i.e., they are off-line models.

When we consider the restrictions of the theoretical models to design an on-line solution, i.e., using a random deployment of sensor nodes, we have to adapt them to create a HSN. The on-line model is a way to implement the theoretical model in a distributed fashion for real applications and it was designed to preserve the small world features of the theoretical models. To create the on-line model, we evaluate the theoretical models and, based on their probability p of shortcut addition, we obtain the following parameters: (i) the number of endpoints (H-sensors); (ii) the number of shortcuts created per H-sensor; and (iii) the neighboring H-sensors that are the number of H-sensors inside the communication radius of a specific H-sensor for TRAM, or the number of H-sensors inside the directed angulation area for TDASM.

As a result of evaluating the theoretical models, we obtain the exact number of H-sensors to be randomly deployed in the network. In this case, both (H-sensors and L-sensors) are randomly deployed. The idea behind this process is to introduce the small world features to the wireless sensor network when the network starts operating. In the on-line model, we define p' as the ratio between the number of shortcuts created per each H-sensor and the number of H-sensors inside the communication radius of each H-sensor (values obtained upon evaluating the theoretical models). The value of p' defines the probability of creating a shortcut between two H-sensors. In this way, for both the On-line Random Addition Model (ORAM) and the On-line Directed Angulation toward the Sink node Model (ODASM), each deployed H-sensor in the network uses the probability p' for adding a shortcut between them. Using this scheme, for each value of probability p, we have the number of H-sensors created and, consequently, the value of p' that represents the probability of shortcut addition between two H-sensors in the on-line models.

It is important to point out that, while the theoretical models require each node to know the position of all other sensors, in the on-line model, the H-sensors only need to know the geographic position of other H-sensors that are within their communication range (neighboring H-sensors) and the Sink's position. This is due to the fact that each shortcut is created only between neighboring H-sensors. To find the neighboring H-sensors, all H-sensor nodes broadcast a *hello* message when the network starts operating.

If the network topology changes over time, due to the addition of new nodes, we can find the neighboring H-sensors using well-known neighbor discovery protocols [Ramanathan and Steenstrup, 1998; Borbash et al., 2007]. Another possibility to obtain the neighboring list is integrating different layers of the protocol stack [Demirkol et al., 2006]. Several MAC protocols were proposed for wireless sensor networks [Ye et al., 2002; Enz et al., 2004; Ye et al., 2004] and most of them use a list of neighbors to synchronize the medium access. In this case, the routing protocol can be integrated with the MAC protocol and use the list of neighbors the MAC protocol has. In this case, the routing protocol has always an updated list of neighbors without additional cost. The next section presents the protocol design of the on-line version for both models.

3.4 Protocol Design of the On-line Model

Algorithm 1 describes the operation of the shortcut creation of the on-line directed angulation toward the sink node model. This algorithm is executed during the start-up time of each H-sensor. The variables used in the algorithm are:

- p': probability of shortcut addition between a pair of neighboring H-sensors;
- v_i: geographic location of node i;
- L_i: list of the geographic location of node i's neighbors (H-sensors);

- v_k: geographic location of node i's neighbors (H-sensors);
- v_{sink}: geographic location of sink node;
- m_i: angular coefficients of the straight line that passes through node i and the sink node;
- m_k: angular coefficient of the straight line that passes through node i and node k;
- Ls_i: shortcut list of node i.

The algorithm works as follows. At the beginning, node i sends a *hello* message containing its geographic location to all neighboring H-sensors and, after that, it receives their locations (Lines 2 and 3). Initially, the list of shortcuts is empty (Line 4). In Line 5, node i calculates the angular coefficient of the straight line that passes through its geographic location and the sink ones. In Lines 6 to 13, the list of shortcuts for node i is created as follows. In Line 7, node i calculates the angular coefficient of the straight line that passes through its geographic location and the location of each neighboring H-sensor. In Line 8, node i verifies whether this angular coefficient is inside the angle region. If it is the case, the node chooses a probability at random and verifies if it is smaller than p' when the node i sets its list of shortcuts (Lines 9 to 11). In Line 14, node i sends a message containing its list of shortcuts for all neighboring H-sensors that are endpoints of the created shortcut. When an H-sensor receives this message, it checks whether itself is inside the list. If this is the case, it updates its list of shortcuts. Otherwise, the H-sensor discards this message. In Line 15, the algorithm returns the list of shortcuts to be used for data communication among H-sensors. Algorithm 1 can be converted to the algorithm of ORAM removing Lines 5, 7, 8 and 12.

In the ODASM, all H-sensors must know their geographic position. This information allows a node to make a decision if another H-sensor is inside its angle region to the sink node. Moreover, the cost of discovering each node's neighbors is the same for both models (one broadcast at the beginning and another one at the end of the algorithm). The difference is that the *hello* message transmitted to the node's neighbors in ODASM contains its geographic position. The on-line version of both models is evaluated in Section 3.6.2.

Algorithm 1 ODASM algorithm

1: **procedure** CreateShortcut(p', v_i)
2: broadcastHelloPacket(v_i)
3: $L_i \leftarrow$ receiveHelloPacket( )
4: $Ls_i \leftarrow \emptyset$
5: $m_i \leftarrow$ getAngularCoefficient(v_i, v_{sink})
6: **for all** $v_k \in L_i$ **do**
7: $m_k \leftarrow$ getAngularCoefficient(v_i, v_k)
8: **if** $\frac{m_i - m_k}{1 + m_i m_k} < \tan \frac{\phi}{2}$ **then**
9: **if** random( ) $\leq p'$ **then**
10: $Ls_i \leftarrow Ls_i \cup v_k$
11: **end if**
12: **end if**
13: **end for**
14: broadcast(Ls_i)
15: **return** Ls_i
16: **end procedure**

3.5 Complexity Analysis of the ODASM Algorithm

In this section, we will briefly analyze the complexity of the ODASM algorithm in terms of communication requirements, processor resources, and time consumption:

- *Communication complexity.* In the ODASM algorithm, the H-sensor nodes start the shortcut creation by broadcasting their geographic location (Algorithm 1, Line 2). Each neighboring H-sensor that receives all broadcasts messages choses one or more neighboring H-sensors to create a shortcut between them. For this, the node sends a message to the other endpoints of the created shortcuts (Algorithm 1, Line 14). Thus, the communication complexity of our algorithm is $O(H)$, where H is the number H-sensors deployed in the network, since each H-sensor sends two broadcast messages in the algorithm.

- *Computational complexity.* As mentioned before, the ODASM algorithm creates the directed search space by calculating the angular coefficient between lines and by generating uniform random numbers. These procedures are quite simple and can be done using a small number of float point operations.

- *Time complexity.* The execution time of the ODASM algorithm is constant, since the algorithm sends two broadcast messages during its execution. After these two broadcasts, each H-sensors creates its shortcuts. However, the required time is proportional to the time to send two broadcast messages, which does not depend of the number of H-sensors or network nodes.

3.6 Simulation Results

3.6.1 Scenarios

In the simulation, we use a network with a total of 1000 nodes (L-sensors and H-sensors) randomly deployed in a sensor field of $1000 \times 1000\,\text{m}^2$. There is only one sink node located at the lower-left corner of the network. The communication radius of the L-sensors and H-sensors are 50 m and at most 500 m, respectively. For each generated topology, each L-sensor has an average of 8 neighbors.

We evaluated different angles ϕ to define the search space (10, 20, 30, 40 and 50 degrees). However, when the angle ϕ is too small (10 or 20 degrees), the search space does not include enough nodes to create shortcuts. On the other side, when the angle ϕ is too large (40 and 50 degrees), the search space is large enough to include nodes that are not directed to the sink node. Because of this behavior, the angle ϕ equal to 30 degrees showed to be more appropriate, since this value is large enough to include a number of sensor nodes but not too large to include nodes that are not directed to the sink node. We used ϕ equal to 30 degrees to evaluate our model.

All results correspond to the arithmetic average of n simulations, with n different network topologies, where n is defined according to the confidence interval desired in the simulation [Jain, 1991]. Let $n = \left(\frac{100zs}{ew}\right)^2$, where s is the standard deviation, w is the average of the initial sample of 10 simulations[3], and z is the normal variate of the desired confidence level. In all simulations, it was used a confidence interval of 95% ($e = 0.05$).

3.6.2 Small World Features Evaluation

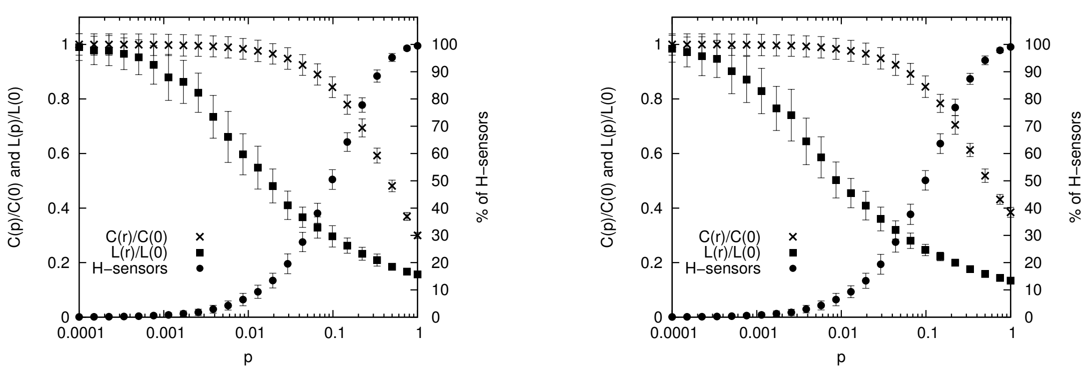
In this section, we evaluate the average minimum path length[4] (L) and the clustering coefficient (C) of TRAM, TDASM, ORAM, and ODASM. In each generated topology, we evaluated the average minimum path length and the average clustering coefficient of the network. After n simulations, we obtain the average of these n results. The values $C(0)$ and $L(0)$ define the clustering coefficient and the average minimum path length of the WSN, respectively, with no shortcut addition (regular graph). The values

[3] This value can be defined arbitrarily. The larger the initial sample, the smaller the value of n.

[4] The average minimum path length is obtained by averaging the minimum number of hops between the sink node and each sensor node.

$C(p)$ and $L(p)$ define the same parameters but for a probability p of shortcut addition probability. In all simulation results, it is calculated the ratio between $C(p)/C(0)$ and $L(p)/L(0)$. This ratio clearly shows the influence of adding a fraction of shortcuts in the original regular graph.

Figure 3.4(a) illustrates the average minimum path length and the clustering coefficient of the theoretical random adding model when we change the value of the probability p. We observe that for small values of the probability ($p < 0.001$), the average minimum path length is reduced only 1.13 times. This is due to the fact that the shortcuts are randomly generated and they do not contribute too much to reduce the average minimum path length between the sink and other nodes of the network. It is worth noting that for $p < 0.001$, the clustering coefficient of the network is close to the one of a regular network. Then, for this value of probability, the network still presents characteristics of a regular graph. For values of p close to 0.01, this model generates a network with small world characteristics. For $p = 0.01$, the average minimum path length is reduced 1.67 times and the clustering coefficient is close to the clustering coefficient of the regular graph. For values of probabilities close to 0.1, the average minimum path length is reduced 3.37 times and the clustering coefficient is reduced only 1.18 times, which lead to a network with small world characteristics. For values of $p > 0.1$, the average minimum path length does not reduce significantly. However, the clustering coefficient reduces fast, characterizing a random network.



(a) TRAM (b) TDASM ($\phi = 30°$)

Figure 3.4. Small world features of the theoretical models

Figure 3.4(b) shows the average minimum path length and the clustering coefficient for TDASM. For $p < 0.001$, the average minimum path length is reduced 1.20 times and the clustering coefficient is the same as the regular graph. When $p = 0.01$, the average minimum path length decreases 2.21 times and the clustering coefficient only 1.02 times. Again, the network exhibits small world characteristics. When the value of the probability is 0.1, the average minimum path length is reduced 4.06 times and the clustering coefficient only 1.17 times, and the network still keeps its small world characteristics. When the value of p is close to 1, the network presents random graph characteristics, once the average minimum path length and the clustering coefficient are smaller than the one of a regular graph.

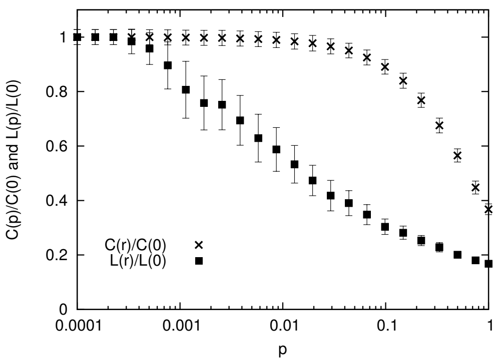
Table 3.1 shows the number of H-sensors deployed in the network, the number of edges created per H-sensor, and the number of neighboring H-sensors inside its communication radius. As discussed in Section 3.3, the on-line model receives the probability of adding a shortcut between a particular pair of H-sensors. In this case, when the probability of adding a shortcut between two nodes in the theoretical model is p, the probability of adding a shortcut in the on-line model is $p' = \frac{\#edges/H-sensor}{\#neighbors/H-sensor}$. As an example, for $p = 0.01$ in the theoretical model, the corresponding probabilities p' in ORAM and ODASM are $p' = 1.03/33.71 = 0.03$ and $p' = 0.91/7.01 = 0.12$, respectively. ORAM has a smaller probability than ODASM because in the former an H-sensor has more neighbors than in the latter due to the angulation toward the sink. Using these values of probability, the number of shortcuts added per H-sensor in the on-line models is preserved, which allows us to maintain the desired average minimum path length (L) and clustering coefficient (C), as depicted in Figure 3.5. Besides, we can have #edges/H-sensor < 1 due to the fact that each edge is counted only once.

p	#H-sensors		#edges/H-sensor		#neighbors/H-sensor	
	TRAM	TDASM	TRAM	TDASM	TRAM	TDASM
0.001	8.15	8.21	0.98	0.85	4.21	1.14
0.01	64.41	64.43	1.03	0.91	33.71	7.01
0.1	504.72	501.66	1.44	1.27	254.94	48.57
1	1000.00	1000.00	7.49	6.56	472.73	85.29

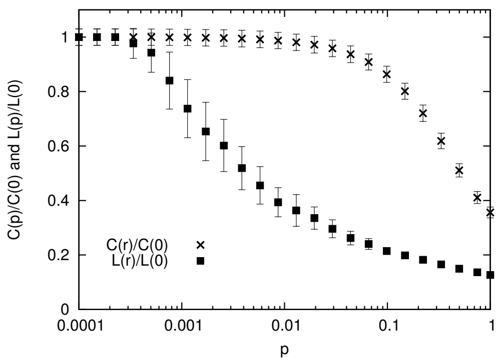
Table 3.1. Number of added links for all models when we change the probability of shortcut addition

Figure 3.5 illustrates the average minimum path length and the clustering coefficient of ORAM and ODASM. Recall that, for example, for $p = 0.001$, the real probability of adding a shortcut between the H-sensors is $p' = \frac{\#edges/H-sensor}{\#neighbors/H-sensor}$. Those values are given in Table 3.1. We present the results in this way so it is possible to have a better and fair comparison between the theoretical and on-line models. Figure 3.5(a) illustrates the average minimum path length and the clustering coefficient of ORAM. We observe that for $p < 0.001$, the average minimum path length does not reduce significantly, because the small number of deployed H-sensors in the networks

(Figure 3.4) lead to a disconnected topology of H-sensors. For values of p close to 0.01, the average minimum path length is reduced 1.70 times and the clustering coefficient is close to the clustering coefficient of a regular graph, which leads to a network with the small world characteristic. For values of p close to 0.1, the average minimum path length is reduced 3.39 times and the clustering coefficient is reduced only 1.12 times, and the network still keeps its small world characteristic. For values of $p > 0.1$, the average minimum path length does not reduce significantly. However, the clustering coefficient reduces fast, characterizing a random network.



(a) ORAM model



(b) ODASM model ($\phi = 30°$)

Figure 3.5. Small world features of on-line models

Figure 3.5(b) shows the average minimum path length and the clustering coefficient in ODASM. For $p < 0.001$, the average minimum path length does not reduce significantly and the clustering coefficient is the same as the regular graph. When $p = 0.01$, the average minimum path length decreases 2.75 times and the clustering coefficient only 1.02 times. In this way, the network exhibits the small world characteristic. When $p = 0.1$, the average minimum path length is reduced 4.65 times, whereas the clustering coefficient only 1.15 times, and the network still keeps its small world characteristics. When the value of p is close to 1, the network presents random graph characteristics, once the average minimum path length and the clustering coefficient are smaller than the one of a regular graph. We can observe that ODASM has the best average minimum path length for all values of p compared with ORAM. This is due to the fact that the shortcuts in ODASM are generated toward the sink node, contributing to the reduction of the average minimum path length between the sink and other nodes in the network. It is important to notice that the theoretical models can be used as an upper limit of the small world characteristics for the on-line models since they present better values for the average minimum path length and the clustering coefficient for the same value of p.

3.6.3 Experimental Evaluation

We evaluated both ODASM and ORAM models to generate HSN topologies through simulations using the Network Simulator (NS-2) [Ns2, 2007]. We study the data dissemination in the network, i.e, when the sink sends a packet to all nodes. In the network layer, we use a simple flooding protocol to perform this dissemination, where each node retransmits a message just once. In this case we can analyze the impact of the resulting network topology on the data communication without considering any particular routing feature. However, other dissemination protocols could be used. If the network uses a hierarchical routing protocol, the H-sensors of our model can be used as cluster-heads. Many hierarchical-based routing protocols proposed in the literature, such as [Du and Lin, 2005], can use the model proposed in this work to achieve a better performance of the routing process. In this way, the H-sensors can form an overlay network connecting the L-sensors with the sink node efficiently.

In our experiments, the physical communication is performed using the free space propagation model and the MAC layer is IEEE 802.11 with a bandwidth of 250 kbps. The cost of transmitting and receiving a message from a distance of 50 m (L-sensors) is 0.0744 W and 0.0648 W, respectively[5]. The cost of transmitting or receiving a message by the H-sensors is proportional to the distance between them with path loss $\alpha = 2$. When an H-sensor sends a packet to an L-sensor or vice versa, the energy consumption is based on a distance of 50 m. The packet size is 64 bytes, which is enough for all message exchanges in the network.

In each simulation, the sink node broadcasts data periodically to all nodes in the network and we evaluate three important network parameters in wireless sensor networks: transmitted packets (Tx), energy ($Energy$), and latency in data communication ($Latency$) for ORAM and ODASM. In all results, as in the previous section, we calculate the ratios $Tx(p)/Tx(0)$, $Energy(p)/Energy(0)$ and $Latency(p)/Latency(0)$ to better understand the influence of adding a fraction of shortcuts to the original regular graph and, for each simulation, we evaluated the average of Tx, $Energy$ and $Latency$ of all nodes (L-sensors and H-sensors). In the experiments discussed in the following, the curves are plotted with dotted lines to help the reader better visualize the results. In practice, we just obtained the individual points on each curve.

[5] These values were obtained from MICAz nodes [MicaZ, 2009].

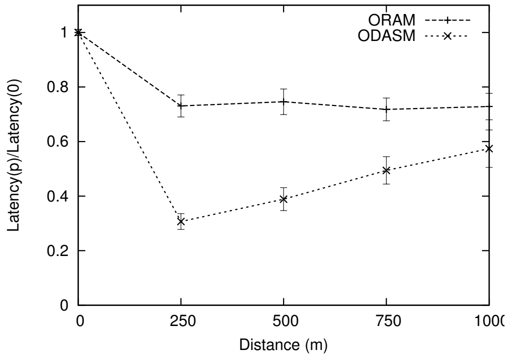
Table 3.2 shows the ratio of transmitted packets and consumed energy for the evaluated models. The transmitted messages among H-sensors and their respective energy consumption were accounted into the total number of messages and amount of energy in the network. These metrics were computed in this way to better understand the tradeoff between using long range communication among H-sensors and their impact on the energy consumption. For all values of the probability p, the two models send the same number of packets and have the same consumed energy (overlap of the confidence interval). When $p = 0.001$, ORAM and ODASM send approximately 1.002 and 1.003, respectively, more packets than the network without shortcuts (H-sensor) and the consumed energy is 1.04 and 1.03 for ORAM e ODASM, respectively. For $p = 0.01$, the number of transmitted packets is, respectively, 1.026 and 1.028 for ORAM and ODASM times greater than the network without shortcuts. For this value of probability, we can see the impact of the long range transmission in the energy consumption. The consumed energy of ORAM and ODASM is, respectively, 1.34 and 1.35 times greater than the network without shortcuts. It is important to point out that the consumed energy with the long range transmission will happen only in H-sensors, which are equipped with a powerful battery.

For $p = 0.1$, the number of transmitted packets in ORAM and ODASM is, respectively, 1.39 and 1.51 times greater than the number of transmitted packets in the regular graph. The energy consumption of both ORAM and ODASM for this value of probability is, respectively, 5.08 and 5.24 times greater than the network without shortcuts. This can be explained by the increase of the transmitted packets as discussed above. Furthermore, there is a cost associated with the long distance shortcut addition to guarantee the small world features in the network. The results of consumed energy, number of transmitted packets and latency for $p > 0.1$ are not shown, because for these values of p, the network presents random graph characteristics.

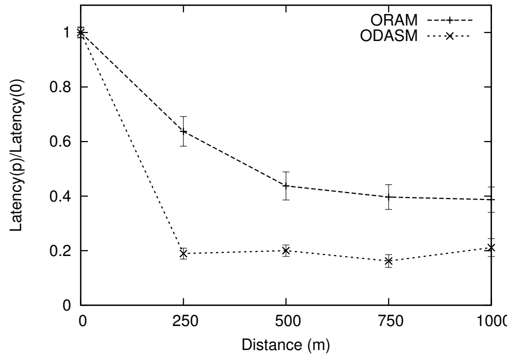
p	$Tx(p)/Tx(0)(\pm\sigma)$		$Energy(p)/Energy(0)(\pm\sigma)$	
	ORAM	ODASM	ORAM	ODASM
0.001	1.003 $\pm 10^{-3}$	1.002 $\pm 10^{-3}$	1.04 $\pm$0.01	1.03 $\pm$0.01
0.01	1.026 $\pm 10^{-3}$	1.028 $\pm 10^{-3}$	1.34 $\pm$0.03	1.35 $\pm$0.02
0.1	1.39 $\pm$0.03	1.41 $\pm$0.04	5.08 $\pm$0.11	5.24 $\pm$0.09

Table 3.2. Energy consumption and transmitted packets

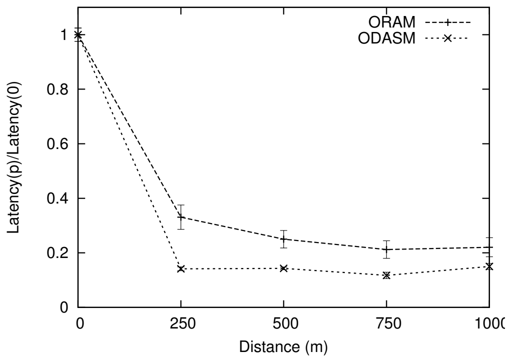
Figure 3.6 shows the latency in the data communication for the evaluated models considering three different probabilities of p. For each probability, we calculated the latency in four different regions of the network: nodes up to 250, 500, 750 and 1000 m



(a) $p = 0.001$



(b) $p = 0.01$



(c) $p = 0.1$

Figure 3.6. Latency in the data communication

far from the sink. For $p = 0.001$ (Figure 3.6(a)) and for a distance of 250 m far from the sink, ODASM is about 2.38 times smaller than the one of the ORAM model and both are, respectively, 3.17 and 1.51 times smaller than the regular network (without H-sensors). Thus, we can see the impact of the added shortcuts in the direction of the sink with a low number of H-sensors. As we increase the distance from the sink node, i.e., 1000 m, the latency of ODASM is just 1.26 times smaller than the one of ORAM and both are, respectively, 1.74 and 1.37 times smaller than the regular network. Also, the number of deployed H-sensors in the network for ODASM and ORAM is 0.8% for both models as shown in Figures 3.5(a) and 3.5(b), respectively[6].

When $p = 0.01$ (Figure 3.6(b)) and for a distance of 250 m, ODASM is about 3.36 times smaller than ORAM and both are, respectively, 5.28 and 1.56 times smaller than the regular network. For a distance of 750 m, ODASM is about 2.45 times smaller

[6]Each probability leads to a different number of H-sensors deployed in the network. To find the number of H-sensors of a specific shortcut creation probability, the reader can match the circle dots with right y-axis.

than the one of ORAM. For this probability, the number of deployed H-sensors in the networks is 6.4% for both models as shown in Figures 3.5(a) and 3.5(b), respectively. When $p = 0.1$ (Figure 3.6(c)), several shortcuts are created in the network and the direction of the shortcuts has low impact on reducing the network latency (compared with ORAM). In this case, the number of deployed H-sensors in the network is 50.4% for ORAM and 50.1% for ODASM. For this probability, ODASM is about 1.81 times smaller than ORAM for all distances. ODASM presents better results for all values of p and for all distances far from the sink and, when $p = 0.01$, ODASM presents the better tradeoff in terms of latency and number of H-sensors. Also, the energy consumption and the number of transmitted packets for both models are equal, due to the same number of shortcuts created and deployed H-sensors in the network (overlap of the confidence interval).

The results above explore the tradeoff of using different numbers of H-sensors (shortcuts) in the network for different values of p, and their impact on the energy consumption, transmitted packets, and latency. We can have an overall impact on all these metrics when we use a certain number of H-sensors. When $p = 0.001$, we deploy only 0.8% of H-sensors and the network presents regular graph characteristics. In this case, the latency does not decrease significantly, and we observe a low overhead in terms of energy consumption and transmitted packets. When $p = 0.1$, the network presents random graph characteristics. In this case, the network latency diminishes substantially. However, the cost of this solution is unfeasible in terms of energy consumption and transmitted packet for a WSN since we need to deploy 50,1% of H-sensors. The most useful scenario happens when $p = 0.01$ and the network presents small world characteristics using only 6,4% of H-sensors in the network. This solution is very applicable to the HSN design since it diminishes significantly the latency with a small overhead in terms of energy consumption and transmitted packets.

3.6.4 Experimental Evaluation in Potential Failure Scenarios

In this section, we evaluate the resilience of ODASM and ORAM through simulations using the Network Simulator (NS-2). The sink node broadcasts data periodically to all nodes in the network and we evaluate the network latency. We calculate the latency of nodes up to 1000 m far from the sink and the simulation time considered is 100 seconds. The failure generation in the network happens in two ways: general and specific. In the general failure generation, at each interval of 10 seconds of simulation, a location is generated randomly in the network and all nodes (L-sensors and H-sensors) in a radius of 100 m around this location are destroyed (area failure). In the specific failure generation, at each interval of 10 seconds of simulation, 10% of randomly chosen H-sensors lose their shortcuts, i.e., they stop working. In the evaluated failure scenarios, we had no disconnected topologies. In this way, all active nodes in the network will receive the messages and forward them once. An active node will receive the dissemination message through an alternative path, which is probably longer as can be seen in the impact on the network latency shown below.

As an example of a general failure generation, Figure 3.7 illustrates this failure for three different values of probability for both models. The gray edges represent the communication among the L-sensors (short communication) and the black edges represent the communication among the H-sensors (long range communication). In this figure, five failures are generated randomly in the network for the network topology described in Section 3.6.1. In this case, all nodes (H-sensors and L-sensors) inside the failure area stop working, i.e., these nodes stop participating in any further data communication. The purpose of generating this type of failure is to analyze the resilience of the network in terms of latency in general failure scenarios.

Figure 3.8 shows the latency of data communication for the evaluated models considering three different probabilities of p (0.001, 0.01, and 0.1). At the end of the simulation (100 seconds) we generate 10 general failures in the network (one failure each 10 seconds). For $p = 0.001$ (Figure 3.8(a)) we have just 0.8% H-sensors deployed in the network and, at the beginning of the simulation, the latency of ODASM is about 1.27 times smaller than ORAM and both are, respectively, 1.75 and 1.37 times smaller than the regular graph (without H-sensors). When we have a low number of H-sensors in the network, the loss of few H-sensors causes a considerable impact on the network. For the instant $t = 100\,\text{s}$ and 10 failures in the network, the latency of ODASM is about 1.22 times smaller than the one of ORAM and both are, respectively, 1.37 and 1.11 times smaller than the regular graph.

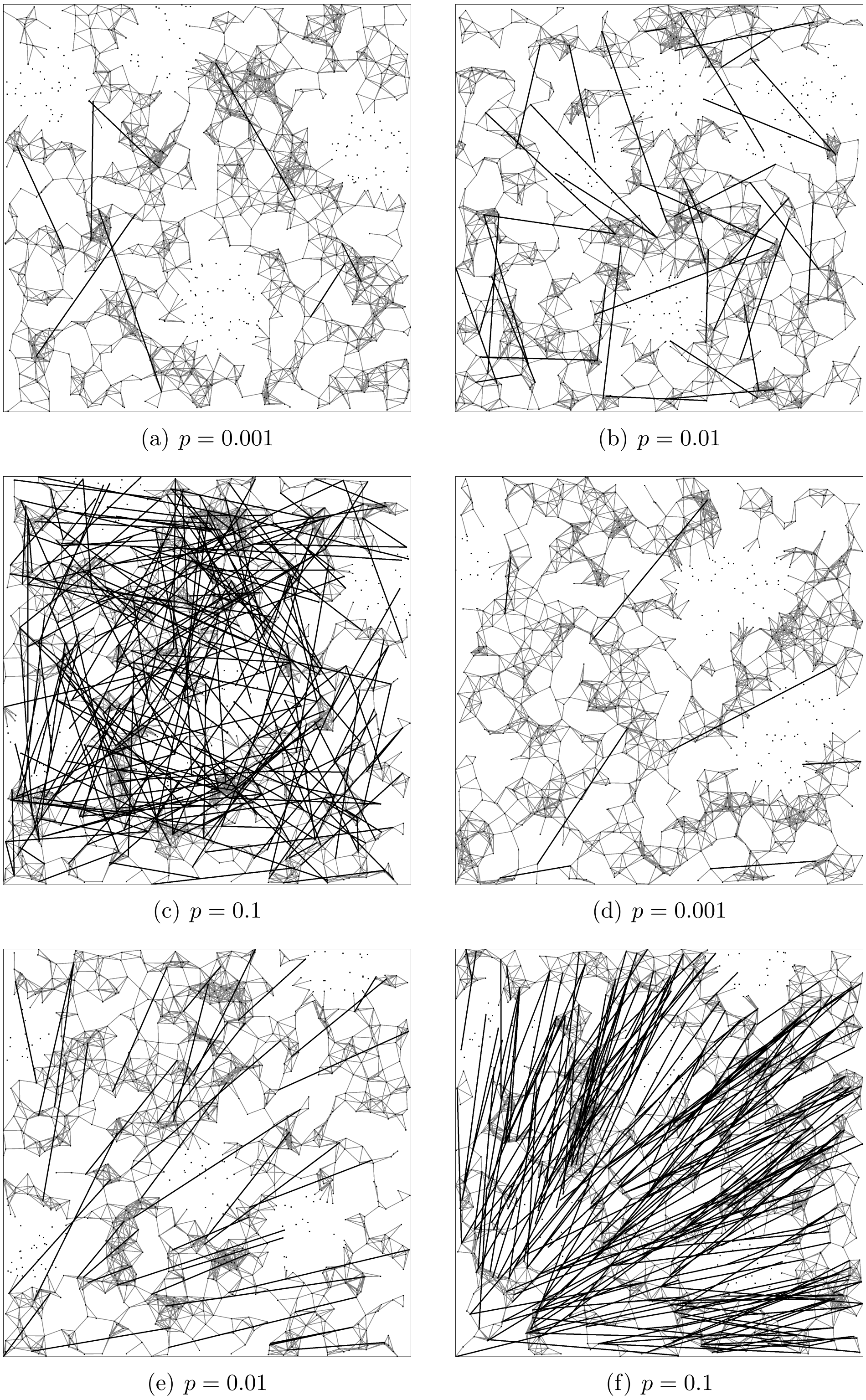
When $p = 0.01$ (Figure 3.8(b)) and $p = 0.1$ (Figure 3.8(c)) each network has, respectively, 6.4% and 50.1% H-sensors deployed in the network. In this case, these values of H-sensors are suitable for the generation of general failures and have almost no effect on the network latency. When $p = 0.01$, and $t = 0$ the latency of ODASM is 1.91 times smaller than ORAM and both are, respectively, 4.76 and 2.49 times smaller than the regular graph. At the end of simulation, ODASM and ORAM have a latency 3.84 and 2.11 times smaller than the regular graph, respectively, and ODASM is about 1.81 times smaller than the one of ORAM. When $p = 0.1$, due to the fact that the network has several H-sensors, the general failures have a very low impact on the network latency. In this case, with just 6.4% H-sensors in the network ($p = 0.01$) and 10 failures in the network ($t = 100\,\text{s}$), the latency of the ODASM is just 1.57

times higher than having $p = 0.1$, and the latency of ORAM is 1.89 times higher than having $p = 0.1$. Thus, we can see the impact of the directions of the shortcuts in its performance.

Figure 3.9 shows the generation of specific failures for different values of probability for both models. A dashed edge represents an H-sensor failure. Thus, when an H-sensor fails, it loses the connectivity with other H-sensors. In this case, 50% of the H-sensors fails, and we can see the impact of losing the connectivity among H-sensors in data communication.

Figure 3.10 shows the latency in the data communication for the evaluated models considering three different probabilities of p. At the end of simulation ($t = 100\,\text{s}$), all H-sensors in the network lose their communication among H-sensors. When $p = 0.001$ (Figure 3.10(a)) and $t = 20\,\text{s}$ (20% of H-sensors are destroyed), the latency of ODASM and ORAM are, respectively, 1.28 and 1.13 times higher than when we have no communication links destroyed among H-sensors. When a network has a low number of H-sensors, if just one H-sensor fails, there is a considerable impact on the data communication. In this way, when the shortcuts are created in the direction of the sink node, the loss of a few H-sensors causes a higher impact than when the shortcuts are created randomly. However, the latency of ODASM is 1.12 times smaller than ORAM for this number of destroyed H-sensors. When 50% of H-sensors are destroyed ($t = 50\,\text{s}$), the latency of ODASM is only 1.03 times smaller than the one of ORAM and both are, respectively, 1.15 and 1.03 smaller than the regular graph.

For $p = 0.01$ (Figure 3.10(b)) and $t = 30\,\text{s}$ (30% of H-sensors are destroyed), the latency of ODASM and ORAM are, respectively, 1.42 and 1.21 times higher than when we have no communication links destroyed among H-sensors. Therefore, the latency of ODASM is 1.62 times smaller than ORAM and both are, respectively, 3.31 and 2.14 times smaller than the regular graph. When $t = 60\,\text{s}$, it is interesting that the latency of ODASM is 1.31 times smaller than ORAM and both are, respectively, 1.83 and 1.39 times smaller than the regular graph for 60% of H-sensors destroyed in the network. For $p = 0.1$ and $t = 50\,\text{s}$, the latency of ODASM is 1.71 smaller than ORAM and both are, respectively, 5.15 and 3.01 higher than when we have no communication links destroyed among H-sensors. For $t > 60\,\text{s}$, the latency in the network grows quickly for both models.

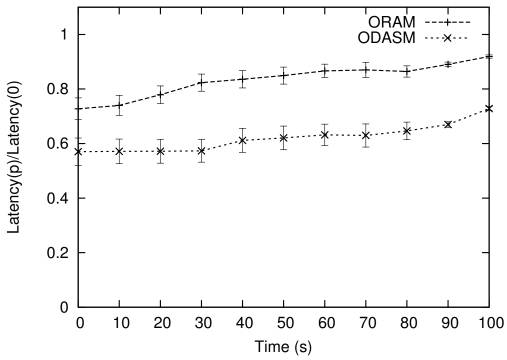


(a) $p = 0.001$ (b) $p = 0.01$

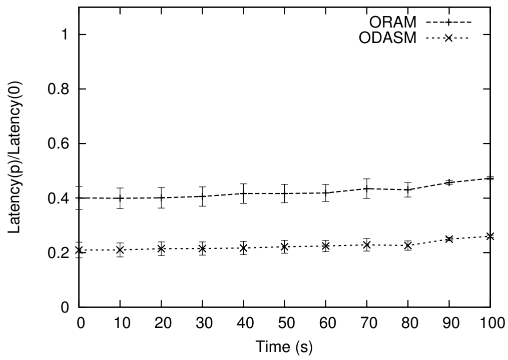
(c) $p = 0.1$ (d) $p = 0.001$

(e) $p = 0.01$ (f) $p = 0.1$

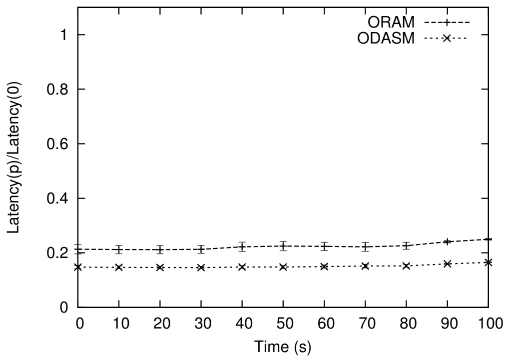
Figure 3.7. Latency in the data communication with five general failures. Figs (a)–(c) show the shortcuts in ORAM. Figs (d)–(f) show the shortcuts in ODASM



(a) $p = 0.001$



(b) $p = 0.01$



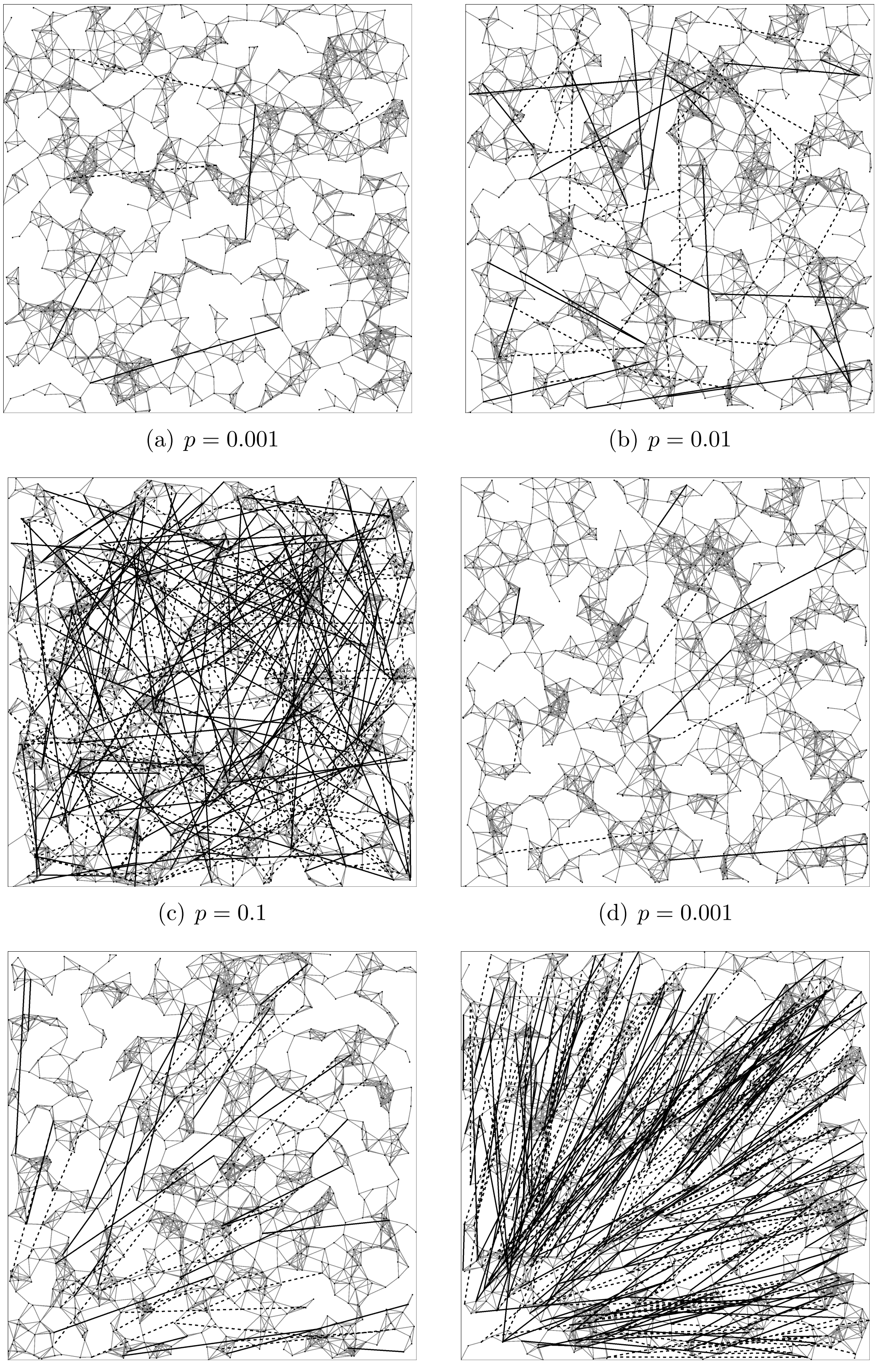
(c) $p = 0.1$

Figure 3.8. Latency in the data communication with general failures

3.7 Final Remarks

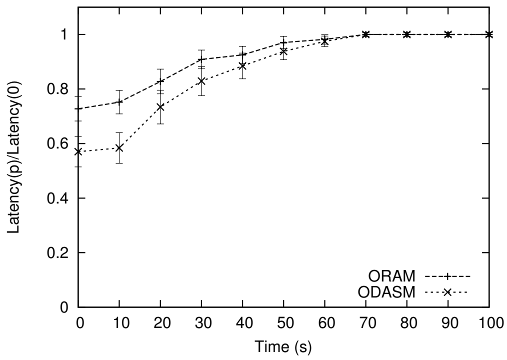
In this chapter, we presented the design of topology for heterogeneous sensor networks using the small world concept, which takes into account the communication pattern of a WSN. We studied theoretical models (TRAM and TDASM) in which the shortcuts are added and also presented their distributed on-line versions (ORAM and ODASM) to build a heterogeneous sensor network with small world features during its start-up time. In the proposed model, the added shortcuts reduce the latency and increase the network resilience.

We evaluated the on-line models and showed that they present the same small world features observed in the theoretical models. In particular, the ODASM exhibits the most interesting tradeoffs between energy and latency. Besides, if an intermediate value of probability is used for the shortcut creation, it is possible to diminish significantly the latency with a small overhead in the terms of energy consumption and transmitted packets. Moreover, the nodes that will spend more energy are equipped

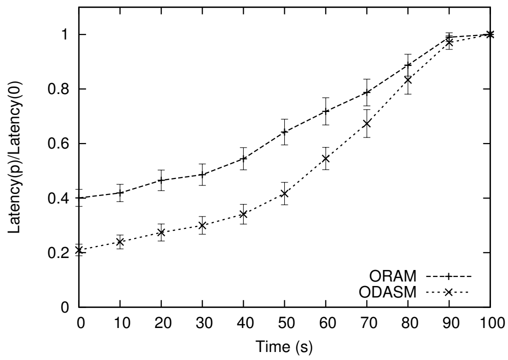


(a) $p = 0.001$ (b) $p = 0.01$ (c) $p = 0.1$ (d) $p = 0.001$ (e) $p = 0.01$ (f) $p = 0.1$

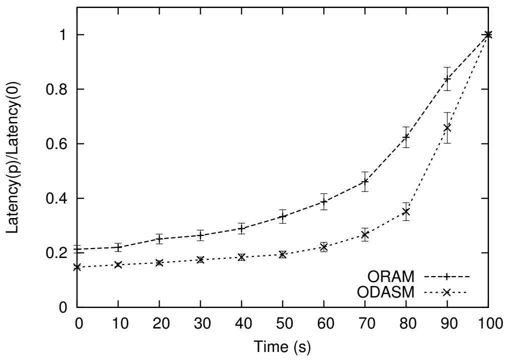
Figure 3.9. Latency in the data communication with specific failures (50% of H-sensors). Figs (a)–(c) show the shortcuts in ORAM. Figs (d)–(f) show the shortcuts in ODASM



(a) $p = 0.001$



(b) $p = 0.01$



(c) $p = 0.1$

Figure 3.10. Latency in the data communication with specific failures

with a greater hardware capacity (H-sensors).

Next, we evaluated the resilience of the on-line models (ORAM and ODASM) for two cases: general and specific failure generation. In both cases, the failures were introduced progressively and also the latency of ODASM was smaller than the one of ORAM for all evaluated scenarios. Furthermore, results showed that general failures on the nodes (L-sensors and H-sensors) do not have a significant impact on the network latency. On the other hand, the specific failures in shortcuts (H-sensors) can decrease the latency when multiple shortcuts are destroyed. However, the destruction of a few shortcuts does not cause a great impact on network latency. Thus, the proposed model can be used to reduce network latency even if the network is exposed to general and specific network failures.

As stated above, the ODASM exhibits the most interesting tradeoffs between energy and latency, which allows us to design applications with desired QoS features [Choe et al., 2010; De and Sang, 2009]. Some applications in wireless sensor networks require a low latency in data communication, others must be robust to node failures. The on-line model proposed in this work can be used for both types of applications. For instance, fire monitoring applications require a low latency whenever a fire spot is detected and also a high resilience to node failures so the network can continue sensing data and sending it to the sink node.

Chapter 4

Multi-Channel Multi-Interface Directed Angulation Toward the Sink Node

As we discussed in the previous Chapter, the shortcut creation in an HSN should be done using the communication range of the H-sensors, since these nodes have more energy reserve to support this kind of communication. However, traditional wireless sensor networks are usually composed by single radio nodes. The main drawback of such networks is the drop in end-to-end throughput as the number of hops on a route increases, especially in HSNs, because of the higher communication range of the H-sensors. In order to overcome this problem, i.e., increase the network capacity and wireless channel utilization, the usage of multiple radio interfaces and multiple channels for communication should be explored, since multiple and long range transmissions can take place without interfering. However, increasing the number of wireless interface we are increasing the H-sensor cost. But as we are using multiple interfaces only in the H-sensors and the expected number of H-sensors in the network is small, this cost does not become a problem in heterogeneous sensor networks.

The goal of this chapter is to design HSN topologies using the small world concept, considering multi-interfaces and multi-channel communication. H-sensors deployed in the network should have two network interfaces, one to communicate with L-sensors and another one to communicate with H-sensors. As in the previous Chapter, the WSN communication pattern is taken into account when defining the shortcuts to be added in the network (among H-sensors). The model was designed in such a way that each shortcut in the network can use a different wireless channel (from the available set of channels), avoiding collisions and increasing the network performance. Simulation results showed that the use of shortcuts with multi-channels can significantly reduce the number of collisions, and also, the latency on data dissemination.

4.1 Introduction

As stated before, a wireless sensor network can be tuned into a small world network adding a few random links. In this way, a few shortcuts can reduce the average path length keeping high values of clustering coefficient. However, in a WSN we cannot choose two random nodes to create a shortcut between them, due to hardware constraints. In this way, to add shortcuts in WSNs we have to deploy H-sensors in the network and the communication among H-sensors will work as shortcuts. These nodes have a higher communication range and energy reserve to support this long communication.

Due to the powerful communication range of H-sensors, a great interference in the wireless channel can occur. For this, H-sensors have to be equipped with two wireless interfaces, one to communicate with L-sensors and another one to communicate with H-sensors. Furthermore, to use shortcuts efficiently, each shortcut created in the network must use a different wireless communication channel from all nearby shortcuts (i.e., each wireless channel is assigned to a different frequency from the available set of frequency). With this assumption, when an H-sensor sends a message using the shortcut, we can avoid the wireless channel interference in the H-sensor interface. In the following, we present the Directed Angulation toward the Sink Model – Multi-Channel especially designed to create shortcuts in a WSN.

4.2 Shortcut Creation

In the Directed Angulation toward the Sink node Model – Multi-Channel (DASM–MC), each H-sensor randomly chooses another H-sensor directed to the sink node to create a shortcut between them. Considering that each H-sensors have just one wireless interface to communicate with each other, each H-sensor must have just one shortcut. If this is true, each shortcut may use a different wireless channel from all neighboring shortcuts, avoiding collisions during the communication. If an H-sensor has more than one shortcut, all of them have to use the same channel. In this way, the goal of this phase is to create directed shortcuts to the sink node and just one (if possible) per H-sensor. The shortcut creation is described as follows.

Firstly, each H-sensor in the network sends a *hello* message that contains its ID and geographic position using the H-sensor interface. Also, each H-sensor stores the received *hello* messages in a neighbor table. To discover all H-sensors that are inside its communication range and directed to the sink node, each H-sensor performs the following steps (basically, the same steps are observed in the previous chapter):

- It calculates the straight line $y_1 = m_1 x + b_1$ that passes by its geographic position and the geographic position of the sink node. This line is the bisectrix of an angle ϕ, that defines the directed region to the sink node or opposite to it, since this direction will also contain the directed region to the sink node. Figure 4.1 illustrates the directed region of a H-sensor i.
- For each H-sensor k in the neighbor table, it creates a straight line $y_2 = m_2 x + b_2$ that passes by its geographic position and the geographic position of H-sensor k.
- It calculates the tangent of the angle formed by these two lines. This tangent can be calculated making $\tan\theta = \frac{m_1 - m_2}{1 + m_1 m_2}$.
- It verifies whether the H-sensor k is inside its directed region to the sink node. For this, the H-sensor can make $\tan\theta < \tan\phi/2$. If the H-sensor k is inside its directed region to the sink node, the H-sensor stores k in the directed neighbor table.

After executing all these steps, each H-sensor has the directed neighbor table that contains only H-sensors that are inside its directed region to the sink node. After that, each H-sensor randomly chooses another H-sensor in the directed neighbor table to create a shortcut between them. Thus, the H-sensor sends a message *create-shortcut* to the target node using the H-sensor interface. If the target node has not created a shortcut yet, it sends a message *ack-confirmation*, creating a shortcut between them. If the target node has already created a shortcut, it sends a *nack-confirmation*. In this case, the H-sensor randomly chooses a different H-sensor in the directed neighbor table and sends again a message *create-shortcut*. These steps are repeated until the H-sensor receives an *ack-confirmation* or while there are unvisited H-sensors in the directed neighbor table.

When an H-sensor sends a message *create-shortcut* to all H-sensors in the directed neighbor table and all responses are *nack-confirmation*, the node enters in a recovery

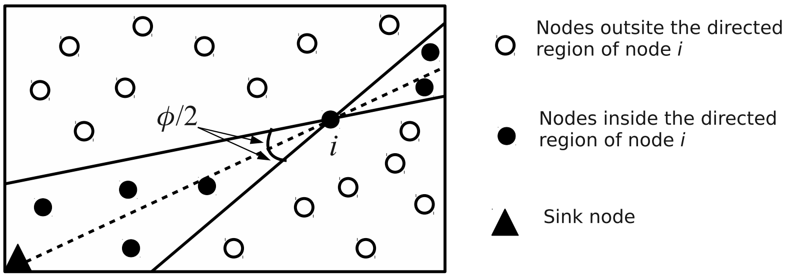


Figure 4.1. Directed region to the sink node

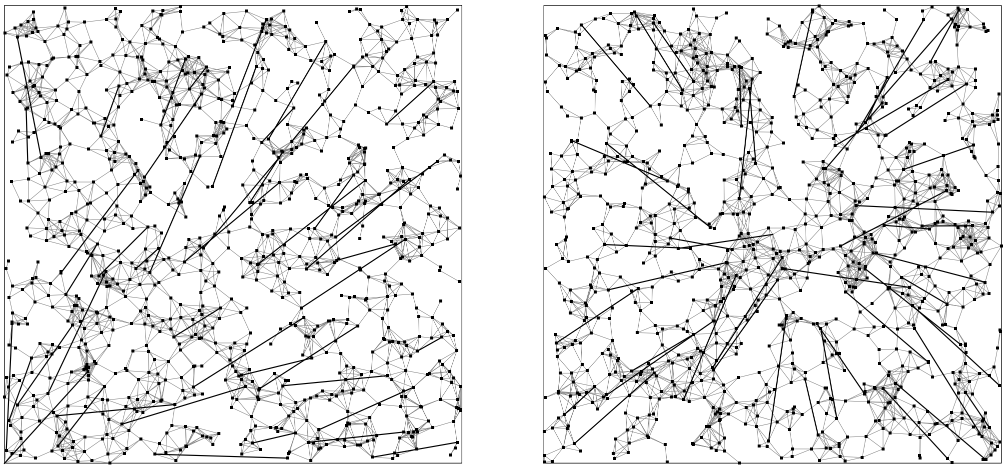
mode. In this mode, the H-sensor randomly chooses an H-sensor in the directed neighbor table (considering all nodes) and sends a message *force-create-shortcut*. When an H-sensor receives this message, it knows that the previous H-sensor did not create a shortcut. For this, the node answers with an *ack-confirmation*.

It is important to point out that the proposed model tries to create just one shortcut per H-sensor. However, in special cases, we should force some H-sensors to have more than one shortcut, otherwise, there will be H-sensors without a shortcut. This drawback happens when the search space of H-sensors does not consider the entire communication range, i.e., H-sensors that are on the board of the monitoring area. Furthermore, it is important to note that all exchanged messages to define the directed region and the shortcuts are done using only the H-sensor interface. In this way, the L-sensor will not participate in this process.

Figure 4.2 illustrates the shortcut creation in the proposed model. In both figures, the network has 1000 nodes including L-sensors and H-sensors. The number of H-sensors deployed is 80. The L-sensor and H-sensor communication ranges are respectively 50 m and 300 m. The light gray edges represent the connectivity between L-sensors and the black edges represent the connectivity between H-sensors after the shortcut creation phase. We can observe that all shortcuts are created in the direction of the sink node, independently on the sink node position.

4.3 Channel Assignment

After the shortcut creation phase, we have to solve the channel assignment problem, i.e., choose a channel from the available set of channels and assign it to the shortcut. This problem can be tackled using a simple interference graph, whose vertices correspond to



(a) Sink node at the bottom left corner (b) Sink node at the center

Figure 4.2. Directed Angulation toward the sink Model

the shortcuts and edges correspond to possible communication interference among the shortcuts. Thus, an edge in this graph means that its endpoints (shortcuts) interfere with each other. A channel can be reused every time a shortcut does not interfere with shortcuts previously allocated to this channel. In this way, we can perform an assignment algorithm, whose output corresponds to the minimization of interference points among the shortcuts.

The channel assignment can be done in a distributed or centralized way. In a distributed way, H-sensors can exchange messages using the H-sensor interface identifying the channel to be used, but minimizing the interference with their neighbors. In a centralized way, the sink node is responsible for solving the assignment and informing the nodes the channel they should use. It is important to note that using a centralized fashion, a better solution can be reached since the sink node has a global view of the shortcuts instead of the H-sensors' local view.

The solution adopted in the proposed model is centralized since the energy cost to execute a distributed channel assignment algorithm is high compared to centralized solutions. For this, all H-sensors in the network execute the shortcut creation described in the previous section. After this, H-sensors send their shortcuts to the sink node using the H-sensor interface in a multi hop way. When the sink node receives the shortcuts, it uses the approximation centralized algorithm described in Das et al. [2006] to solve the channel assignment problem. This algorithm, has an approximation rate of $1-\frac{1}{k}$, where k is the number of available channels, which is 0.91 using 12 channels. After solving the channel assignment problem, the sink node sends to the H-sensors the problem solution using the H-sensor interface. When an H-sensor receives this message, for now on it will use the specific channel to communicate via its shortcut. It is important to point out that if an H-sensor has more than one shortcut, all of them use the same channel, since the H-sensors are equipped with one wireless interface to communicate among them. Also, if the number of deployed H-sensors is enough to create a connected topology among the H-sensor interfaces (considering all H-sensor neighbors), it is not necessary that a L-sensor participates in the routing problem to solve the assignment. However, if the H-sensor density in the network is low, some L-sensor may participate in this process.

4.4 Simulation Results

4.4.1 Scenarios

In all simulations we use a network with 1000 nodes (including L-sensors and H-sensors). All nodes are randomly deployed in a sensor field of 1000x1000 m^2. The communication ranges of the L-sensors and H-sensors are 50 m and 300 m respectively. In such network, each L-sensor has, on average, 8 neighbors. The sink node is located at the bottom left corner. As in the previous chapter, all results correspond to the arithmetic average of n simulations, with n different network topologies, where n is defined according to the confidence interval desired in the simulation [Jain, 1991]. Let $n = \left(\frac{100zs}{ew}\right)^2$, where s is the standard deviation, w is the average of the initial sample of 10 simulations[1], and z is the normal variate of the desired confidence level. In all simulations, it was used a confidence interval of 95% ($e = 0.05$).

We evaluated both the Random Addition Model (RAM) and the Directed Angulation toward the sink Model (DASM) using the Network Simulator (NS-2). In the RAM, each H-sensor chooses at random another H-sensor inside its communication range to create a shortcut between them. To perform the collision and latency evaluations, the sink node periodically sends a message to the entire network (data dissemination). The results correspond to the average of all data disseminations. Since there is not a version of IEEE 802.15.4 MAC protocol (ZigBee) that uses multi-interface and multi-channel available in the NS, we used the IEEE 802.11 MAC layer with a bandwidth of 250 kbps to simulate a sensor node with a limited bandwidth. The implementation of multi-interface and multi-channel in the H-sensors is based on the architecture proposed in [Raniwala and cker Chiueh, 2005]. In the simulation, we used the 13 channels available in the IEEE 802.11g MAC protocol.

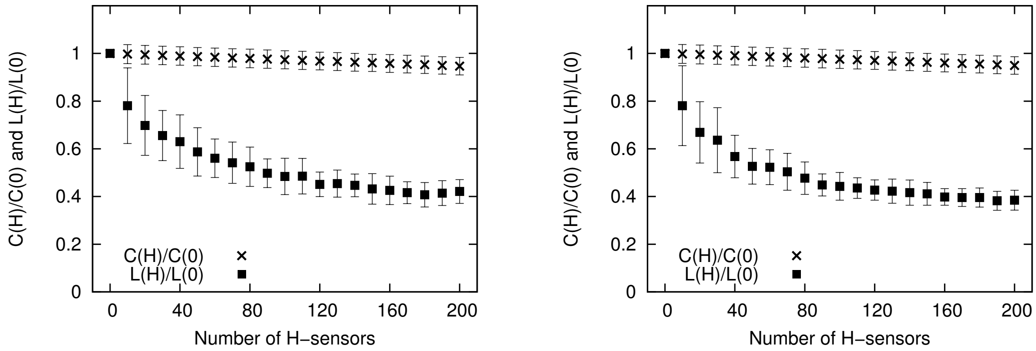
[1]This value can be defined arbitrarily. The larger the initial sample, the smaller the value of n.

4.4.2 Small World Features Evaluation

To check whether a network has small world features, we have to evaluate the average path length[2] (L) and the clustering coefficient (C). If the network with shortcuts presents smaller average path length and similar values of clustering coefficient compared to a network without shortcuts, we can define such network as a small world network. Values $C(0)$ and $L(0)$ represent the clustering coefficient and average path length, respectively, when the network does not have shortcuts (network composed by 1000 L-sensors). Values $C(H)$ and $L(H)$ represent the same metrics when the network has H H-sensors. In all results, it is calculated the ratio between $C(H)/C(0)$ and $L(H)/L(0)$. This ratio clearly shows the influence of adding a fraction of H H-sensors to the original regular network.

Figure 4.3(a) illustrates the metrics L and C of the RAM. When $H < 40$ the average path length is reduced in 36%. For this number of H-sensors, the network does not present small world features yet. However, when $H = 80$, the average path length is reduced in 47% keeping high values of clustering coefficient, which leads to a network with small world features. For $H > 80$, the average path length does not reduce significantly. When $H = 200$, the average path length is reduced in 56%, just 9% smaller compared to $H = 80$. In this case, it is not worth to deploy more than 80 H-sensors.

Figure 4.3(b) shows the same metrics using the DASM. When $H = 40$, the average path length is reduced in 43% keeping high values of clustering coefficient. In this case, the network presents small world features. It is important to point out that due to the fact that in the DASM all the shortcuts are directed to the sink node, the network starts presenting small world features with less H-sensors compared to the RAM. When $H = 80$ the path length is reduced in 52% keeping similar values of clustering coefficient. Again, the network presents small world features. For $H > 80$ the average path length does not reduce significantly, and when $H = 200$ the path length is reduced in just 8% compared to $H = 80$. After evaluating all these parameters for both models, the following is apparent: (i) when the network has 80 H-sensors, the average path length is significantly reduced compared to the network without H-sensors; (ii) for all values of H, the DASM presents smaller values of path length keeping high values of clustering coefficient than the RAM; and (iii) when the network has more than 80 H-sensors, the path length does not reduce significantly. In this way, it is not necessary more than 80 H-sensors to transform a regular network into a network with small world features.



(a) Random Addition Model (b) Directed Angulation toward the sink Model

Figure 4.3. Small world features

4.4.3 Collision Evaluation

Table 4.1 shows the number of possible collision points in the network for different values of H-sensors. RAM and DASM represent, respectively, the Random Addition Model and the Directed Model without multi-channel. RAM-MC and DASM-MC represent the same models with multi-channel (12 channels). The number of collision points is calculated as follows. For each shortcut in the network, we calculate the number of shortcuts that are affected by its transmission. We can see that when the network has 40 H-sensors, the number of possible collision points is small for both models without channel assignment. However, while the number of H-sensors increases, the number of possible collision points increases as well. Moreover, in the RAM model, as the shortcuts do not have any specific direction, the number of collision points is greater compared to the DASM, where all shortcuts are directed to the sink node.

The same behavior is observed when both algorithms use the channel assignment. It is important to point out that for 40 and 80 H-sensors deployed in the network, the average number of possible collision points is smaller than 1. This means that in some topologies, some H-sensors entered in the recovery mode during the shortcut creation phase. Because of this, if an H-sensor has more than one shortcut, the same channel is assigned for all of them. Furthermore, it is important to observe that using multi-channel in the shortcuts, the number of possible collisions reduces dramatically. In this

Table 4.1. Number of collisions points

H-sensors	RAM	DASM	RAM-MC	DASM-MC
40	63.56	45.78	0.49	0.47
80	270.38	225.50	0.93	0.64
120	631.38	535.68	9.18	6.87
160	1151.18	991.74	27.36	22.48
200	1812.62	1622.06	54.42	46.42

way, the number of collisions during the data dissemination process will be reduced.

Figure 4.4 shows the number of collisions during the data dissemination process. The continuous lines represent RAM and the dashed lines represent DASM. The version of both models with multi-channel and without multi-channel are plotted with a different point style. We can see that the number of collisions is proportional to the number of possible collisions observed in Table 4.1. Also, as the number of possible collisions is smaller for all values of H-sensors, the DASM has less collisions during the data dissemination phase. For instance, the number of collisions in DASM is 10% smaller compared to the RAM for all values of H-sensors. As the number of collisions in the RAM-MC and DASM-MC is close to zero, we plotted the results of both models in a different view. We can see that when multi-channel is used, the number of collisions in the network reduces dramatically. For instance, when $H \leq 100$, the number of collisions is close to zero. However, even using multi-channel, if the number of H-sensors increases, the number of collisions increases as well.

4.4.4 Latency Evaluation

The data dissemination latency is a reflection of the average path length between the sink node and the sensor nodes and the number of collisions during the data dissemination. Figure 4.5 shows the data dissemination latency for different numbers of H-sensors. In all results, as in Section 4.4.2, we calculate the ratio $Latency(H)/Latency(0)$. Figure 4.5(a) shows the latency when the network has 40 H-sensors. We can see that because of the low number of deployed H-sensors, the

[2] The average path length is calculated between the sink node and sensor nodes instead of between all pairs of nodes.

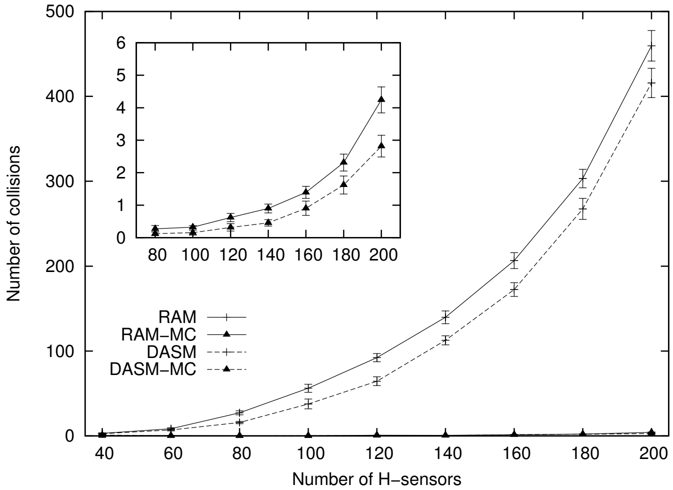
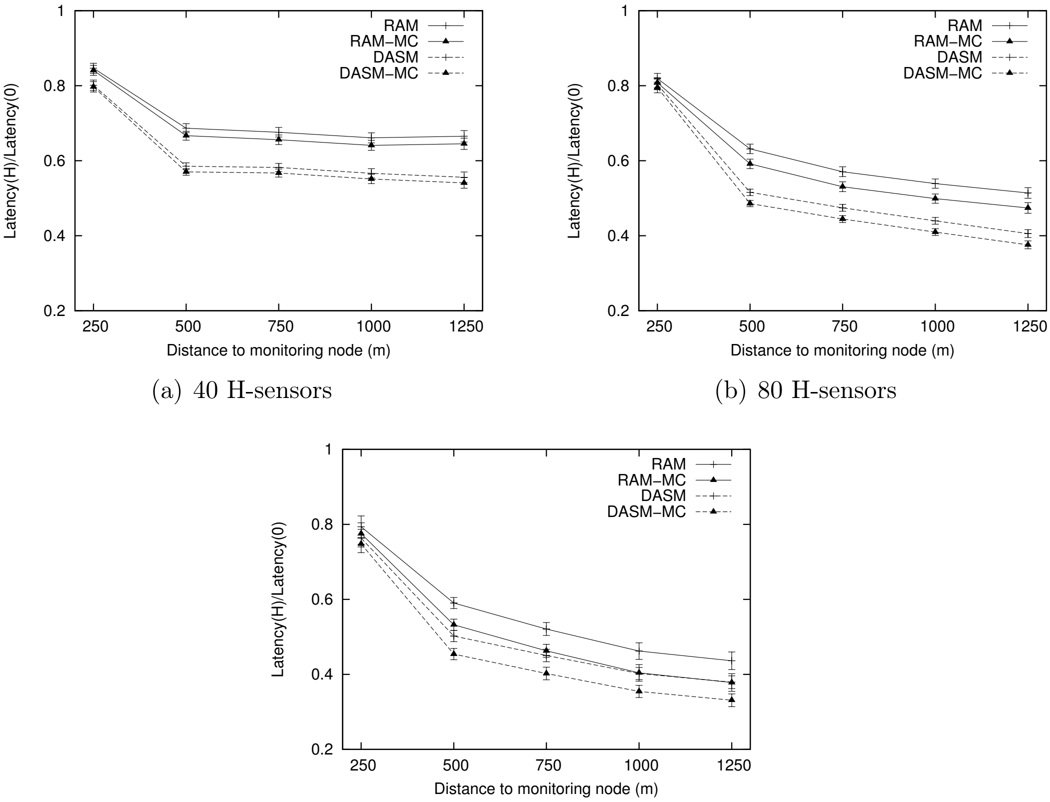


Figure 4.4. Number of collisions

number of collisions during the communication is small and the multi-channel support does not reduce significantly the latency for both models. However, as in DASM all shortcuts are directed to the sink node, the latency is 12% smaller compared to the RAM when the distance from the sink node is greater than 500 m. Also, the RAM and DASM latency is, respectively, 36% and 43% smaller compared to the network without H-sensors.

When the number of H-sensors increases (Figure 4.5(b)), the number of collisions during the data dissemination increases as well. In this way, when multi-channel is used, the latency starts to reduce. The DASM-MC latency is 5% smaller compared to DASM, and the RAM-MC latency is 7% smaller compared to RAM. We can see that the difference between RAM-MC and RAM is greater than the difference between DASM-MC and DASM. This happens because in RAM the number of collisions is greater than in DASM. Furthermore, DASM-MC latency is 17% smaller compared to the RAM-MC latency. Compared to the network without H-sensors, the RAM-MC and DASM-MC latencies are, respectively, 53% and 62% smaller.

When the number of H-sensors is 120 (Figure 4.5(c)), the difference between RAM and DASM decreases. This happens because the higher number of H-sensors induces the creation of directed shortcuts to the sink node in RAM. On the other hand, the difference between RAM and RAM-MC and between DASM and DASM-MC increases because of the higher number of collisions during the data dissemination



(a) 40 H-sensors

(b) 80 H-sensors

(c) 120 H-sensors

Figure 4.5. Data communication latency

process. It is important to note that when the network has more than 80 H-sensors, the data dissemination latency does not reduce significantly for both models. For instance, when the distance from the sink node is 1250 m and the network has 120 H-sensors, the latency for RAM-MC and DASM-MC are, respectively, 6.5% and 4% smaller compared to the network with 80 H-sensors. In this way, it is not necessary more than 80 H-sensors to transform a regular network into a small world network, in order to significantly reduce the latency using multi-channel.

4.5 Related Work

In recent years, multi-channel networks have been the subject of several studies in literature. Early work proposed by Cidon and Sidi [1989] consists of a distributed dynamic channel assignment algorithm, which is conflict-free and suitable for shared channel multi-hop networks. MAC protocols based on modification of the IEEE 802.11 standard were proposed for employing multiple channels in wireless networks and wireless mesh networks [So and Vaidya, 2004; Raniwala and cker Chiueh, 2005; Subramanian et al., 2008].

Since WSNs have a limited bandwidth and data packet size is very small, multi-channel protocols designed for general wireless networks are not suitable for WSNs. Thus, new multi-channel MAC protocols, such as MMSN [Zhou et al., 2006] and MC-MAC [Chen et al., 2006b] were designed especially for WSNs. Both are able to increase network energy efficiency, network lifetime, and data throughput. Wu et al. [2008] proposed a novel tree-based multi-channel scheme for data collection applications in WSNs. The main idea is the allocation of channels to disjoint trees, exploiting parallel transmissions among trees. The proposed protocol, called TMCP, can significantly improve the network throughput and reduce packet losses. Also, the authors claim that TMCP better accommodates multi-channel realities found in WSNs than other multi-channel protocols such as MMSN and MCMAC.

Multiple channels require us to address the channel assignment problem, i.e., the decision problem of which channel should be used by a particular communication link. The goal is to minimize the total interference among all links. This problem is closely related to the MAX k-CUT problem, which is known to be NP-hard [Sahni and Gonzalez, 1976]. Thus, there is no polynomial time exact algorithm, which can always find the optimal assignment, suggesting the use of approximation algorithms and heuristics.

4.6 Final Remarks

In this chapter we proposed the design of HSNs based on the small world concept, which takes into account the communication pattern of a WSN to create shortcuts directed to the sink node. Also, to use the shortcuts efficiently, the H-sensors should have two wireless interfaces, and the H-sensor interface should use the multi-channel capability of the MAC layer to avoid wireless interference during the data dissemination. Simulation results showed that a WSN can be tuned into an HSN with small world features adding only 8% of H-sensors. Furthermore, the use of directed shortcuts with multi-channel support reduces significantly the latency and the number of collisions during the data dissemination. As a future work, we plan to study the data communication when the sensor nodes send their data to the sink node, i.e., data collection.

Chapter 5

A Framework based on Small World Features to Design HSN Topologies with QoS

5.1 Introduction

Based on the past few years, a WSN will become pervasive in our daily lives, for example, in our cars, offices and homes. WSNs promise to revolutionize the way we interact and manage the physical world, in such a way that a WSN needs to provide QoS to satisfy some application requirements [Xia, 2008; Choe et al., 2010]. For instance, in a fire detection system, the sensor nodes have to report the occurrence of fire to the sink node in a short period of time and then, the sink node will react by a certain deadline so that the situation does not become a disaster.

Different applications require different QoS requirements. For example, in an air-conditioning system that controls the temperature inside a house, a large delay during the communications among the sensor nodes and the sink node is acceptable. On the other hand, considering a safety-critical control system, a large delay might not be acceptable. Although QoS is an overused concept and it is easy to understand by examples, different technical communities may perceive and interpret QoS in different ways [Chen and Varshney, 2004]. For instance, RFC 2386 characterizes QoS as a set of service requirements to be met when transporting a packet stream from the source to its destination. However, considering the Internet Community, QoS could be defined as "the ability to provide better than best effort services" or "the ability to provide different priority to different applications, users, or data flows, or to guarantee a certain level of performance to a data flow" [Wang, 2001]. Considering different WSN applications, QoS in WSN can be characterized by reliability, robustness, security, timeliness, availability and others. The quality of satisfaction of these features can be quantified measuring the packet loss rate, jitter, throughput, delay, energy consumption and other network metrics. These network metrics are described as follows:

- *Packet loss rate:* packet loss occurs when one or more packets during data communication fail to reach their destination. The packet loss rate is the percentage of these packets that are lost during this process.
- *Jitter:* is the variation of the delay to delivery packets in a communication network.
- *Throughput:* is the effective number of data flow transported within a certain period of time in a communication network.
- *Delay:* is the travel time of a packet from the source to the destination node.
- *Energy consumption*: is the amount of energy spent during data communication.

The wireless sensor network applications can be divided into different classes, where each class requires different levels of the above network metrics. For instance, solutions for the class of applications that is delay-tolerant need to provide different levels of end-to-end delay based on the application requirements. Some critical applications requires a very low communication delay while others may accept a certain level of delay. It is important to point out that a routing structure to provide low communication delay may expend more energy. In this case, just critical applications should use this kind of structure.

As we described before, a WSN is composed by sensor nodes that are small devices with limited hardware capabilities. In this scenario, QoS solutions may not be applied in WSNs because of the limited memory size, energy constraints and communication radius [Xia, 2008]. To bypass this problem, heterogeneous nodes with powerful capabilities can be used to improve network metrics and then, providing QoS in sensor networks.

In this Chapter we describe a framework based on small world features to design HSN topologies with QoS. The proposed framework was designed to improve the delay and energy consumption metrics during data communication in HSN. The next sections describe in details the framework.

5.2 Framework Design

5.2.1 Introduction

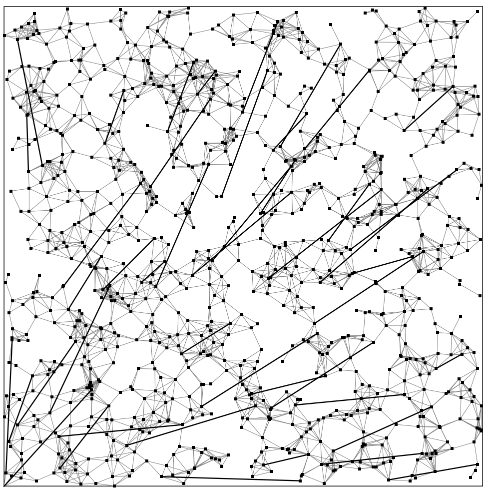
This section describes the proposed framework to design HSN topologies with QoS. However, instead of using one network topology to provide quality of services, the proposed framework is based on the small world concepts to create different network topologies using the same network infrastructure (L-sensors and H-sensors). The framework creates three topologies and each topology has different goals, based on the QoS metrics (Section 5.1), especially in terms of end-to-end delay and energy consumption. Thus, the framework provide a topology for critical applications and also provide a topology for applications that do not need a small delay during data communication. The three topologies are: (i) Homogeneous topology (Homogeneous); (ii) DASM topology (DASM) and (iii) DASM Tree topology (DASM-T).

The homogeneous topology creates a topology that connects the sink node and the sensor nodes using only the L-sensors or the L-sensor interface of the H-sensors. The goal of this topology is to decrease the energy consumption during the data communication. Since the topology uses only the L-sensors (small communication range) or the L-sensor interface of the H-sensors to route packets, the expected energy consumption of the communication task using this topology is low. This observation is intuitive in a multi-hop data communication network, since one of the energy consumption factors is the communication range.

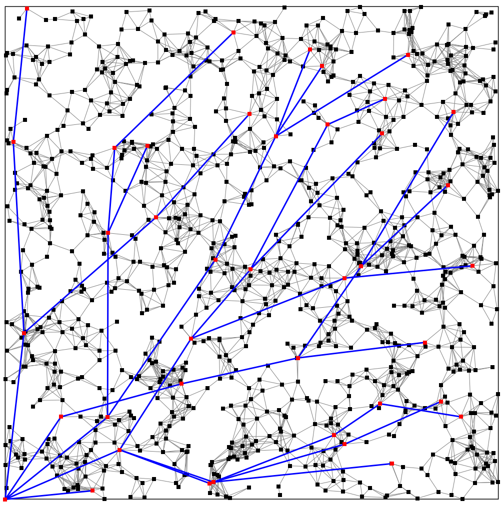
The second topology is the DASM topology and this topology was described before in details in Chapters 3 and 4. The DASM topology creates random-directed shortcuts using the H-sensor interface (long-range communication) of the H-sensors. An important feature of this topology is the disconnection of the shortcuts among the H-sensors. The DASM creates shortcuts in such way that the shortcuts do not create a connected topology among them and this is an important feature for our framework. Based on this, the data communication using this topology uses the shortcuts among the H-sensors and the communication among the L-sensors, mixing both types of nodes. Because of this, the energy consumption of the DASM topology is greater compared to the homogeneous topology, since the communication among the H-sensors consumes a high amount of energy because of their greater communication range.

An important feature of the DASM topology is the reduced latency during data communication compared to the homogeneous topology. This happens because of the created shortcuts. Moreover, as the communication among the H-sensors does not create a connected topology among them, some L-sensors will certainly be used during the communication and this will increase the latency during the communication. However, some network applications as critical systems require a very small latency. Based on this, the framework is able to create in a distributed way a connected topology that connects all the H-sensors with the sink node without using the L-sensors. This topology is called DASM Tree and it was designed in order to provide a smaller latency during the communication. To create a connected topology among the H-sensors we will redefine the directed region to the sink node used in the DASM model.

Figure 5.1 illustrates the topologies of the proposed framework. In these networks, 1000 nodes were deployed randomly, forming a flat topology in a $1000 \times 1000\,\text{m}^2$ sensor field. In these figures, the sink node is located at the lower-left corner. The communication range of the L-sensors is 50 m and each node has an average of 8 neighbors. Also, each network has 80 H-sensors and the endpoint nodes of the shortcuts, as presented before, are H-sensors allowing them to communicate in a long distance. The communication range of the H-sensors is 400 m. The homogeneous topology can be found considering the short communications among the L-sensors and removing the shortcuts (long range communication) in both figures. Figure 5.1(a) shows the DASM topology and Figure 5.1(b) shows the DASM Tree topology. We can see that in the DASM topology the connectivity among the H-sensors does not create a connected topology while the connectivity among the H-sensors in the DASM-T topology creates a connected topology among them.



(a) DASM topology



(b) DASM Tree topology

Figure 5.1. Framework topologies

The proposed framework works as follows. During the network startup, the framework creates the three topologies using three different algorithms. First, the Framework creates the homogeneous topology. Then, the DASM topology is created and finally, the DASM Tree topology is created. The algorithms to create each topology are described in the next sections.

5.2.2 Homogeneous Topology

The Homogeneous topology is created considering just the L-sensors or the L-interface of the H-sensors in the network. The goal of this topology is to improve the energy consumption during data communication. However, to decrease the energy consumption we have to use small communication range during data communication process, since the communication range is one of the factors in the energy consumption. By doing this, we will increase the communication delay because of the large number of hops to perform the routing process.

Algorithm 2 describes the creation of the homogeneous topology's routing. The algorithm creates a topology to connect the L-sensors to the sink node. The algorithm was design in such a way to be simple and efficient in terms of overhead (control packets). For this, the homogeneous topology to connect the L-sensors to the sink node is not a minimum spanning tree, since the cost to create this kind of topology is high related to the number of messages [Khan et al., 2009]. The variables used in the algorithm are:

- id_i: Node's id;
- fH_i: Father's id in the homogeneous topology of a node i;
- msg: Variable to receive a message from other nodes;
- *L-interface*: Indicate the id of the L-sensor wireless interface;

The algorithm is started by the Framework and works as follows. The sink node starts a broadcast in the network and updates its father in the topology. As the sink node does not have father, it sets the fH_i as *localhost* (Lines 5 to 9). When a L-sensor does not have father yet, it waits to receive a packet from the sink node or other L-sensor. In both cases, the node updates its fH_i in the routing and forwards the broadcast message (Lines 11 to 15).

As a result of Algorithm 2 we have a homogeneous topology's routing (composed just by L-sensors) that connects the L-sensors to the sink node. It is important to point out that the fH_i of a L-sensor is determined by the first package received, i.e., fH_i is the node who sent that message. The algorithm was designed in this way to reduce the number of control messages to create the topology. However, as we expected, the created topology is not the best one in terms of number of hops to perform the communication between the sink and the L-sensors. As the goal of this topology is to reduce the energy consumption during communication, the algorithm devises a routing with this purpose.

Algorithm 2 Algorithm to create the homogeneous topology's routing

```
procedure HomogeneousTopologyRouting( )
    id_i;
    fH_i ← null;
    msg ← null;

    if id_i == sink then
        broadcast(id_i, L-interface);        ▷ Using the L-sensor interface
        fH_i ← localhost;                    ▷ Sink does not have father
        return
    end if

    if fH_i == null then
        msg ← receiveBroadcast(L-interface); ▷ Using the L-sensor interface
        fH_i ← msg.id
        broadcast(id_i, L-interface);
    end if
end procedure
```

5.2.3 DASM Topology

The DASM topology is defined using the shortcut creation algorithm proposed in Chapter 4, Section 4.2. The basic idea of this algorithm is to create directed shortcuts among the H-sensors deployed in the network. For this, we introduced the calculus of the directed region to the sink node. The directed region to the sink node also includes the direction opposite to it, since the shortcuts in this region will also be directed to the sink node. The algorithm was designed in such a way to create one shortcut per H-sensor. Because of this, the created topology among the H-sensors is not connect. With this feature we can apply a channel assignment algorithm to assign a different wireless channel for each shortcut, when possible. However, the DASM model used in the proposed framework will not consider the channel assignment problem, but it will continue to create one shortcut per H-sensor.

To create the DASM topology the sink node sends a broadcast message to the entire network announcing the DASM topology creation. However, just the H-sensors will execute the algorithm. We need to send the message to the entire network (considering L-sensors and H-sensors) because some H-sensors (a set of H-sensors) can be outside of the communication range of other set of H-sensors. In this way, the L-sensors will work as a bridge to interconnect them. When a H-sensor node receives this message, it introduces a random delay before executing the DASM algorithm. We use a random delay because some H-sensor can receive the broadcast message before others. It is important to note that the DASM model does not require a synchronization to start the topology creation.

After the broadcast to create the DASM topology, the sink node sends another broadcast to create the DASM topology's routing to connect all nodes (L-sensors and H-sensors) to the sink node. Algorithm 3 describes the algorithm to create the DASM topology's routing. The variables used in the algorithm are:

- id_i: Node's *id*;
- fD_i: Father's id and the associate interface in the DASM topology of a node i;
- *msg*: Variable to receive a message from other nodes;
- *L-interface*: Indicate the *id* of the L-sensor wireless interface;
- *H-interface*: Indicate the *id* of the H-sensor wireless interface;
- *shortcut*: Endpoint of the shortcut.

The algorithm starts when the sink node sends a broadcast message to create the routing infrastructure (Lines 6 to 13). In the case that the sink node has a shortcut to another H-sensor, it sends a unicast message using the shortcut communication (Line 9). When a node does not have a father, it receives the message and update its father's *id* and its father's interface (Lines 17 and 18). These variables will be used during the routing process. It is important to point out that a L-sensor receives the message only by the L-sensor interface and a H-sensor can receives the message by both L-sensor and H-sensor interfaces. After, the node broadcasts the received message using the L-sensor interface (Line 19) allowing both types of node to receive that message. The L-sensors do not have shortcuts, however, the H-sensors must have. For this, when a node has a shortcut, it means that the node is an H-sensor. In this case, the H-sensor sends the received message to the other endpoint of the shortcut using a unicast communication (Line 21).

Algorithm 3 Algorithm to create the DASM topology's routing

```
1: procedure DASMTopologyRouting( )
2:     id_i;
3:     fD_i ← null;                                          ▷ Father id in the topology
4:     shortcut;                                             ▷ Endpoint of the shortcut
5:
6:     if id_i == sink then
7:         broadcast(id_i, L-interface);                     ▷ Using the L-sensor interface
8:         if shortcut ≠ ∅ then
9:             unicast(id_i, shortcut, H-interface);         ▷ Using the H-sensor interface
10:        end if
11:        fD_i ← localhost;                                 ▷ Sink does not have father
12:        return
13:    end if
14:
15:    if fD_i == null then
16:        msg ← receiveBroadcast();
17:        fD_i.id ← msg.id
18:        fD_i.interface ← msg.interface
19:        broadcast(id_i, L-interface);                     ▷ Using the L-sensor interface
20:        if shortcut ≠ ∅ then
21:            unicast(id_i, shortcut, H-interface);         ▷ Using the H-sensor interface
22:        end if
23:    end if
24: end procedure
```

After the execution of the algorithm to create the DASM topology's routing, each node has its father and the associate interface to communicate with it. It is important to point out that a L-sensor can have as father another L-sensor or an H-sensor. In both cases, the communication between them is performed by the L-sensor interface. Using the same idea, an H-sensor can have as father another H-sensor or a L-sensor. However, to communicate with another H-sensor, an H-sensor uses the H-sensor interface.

5.2.4 DASM Tree Topology

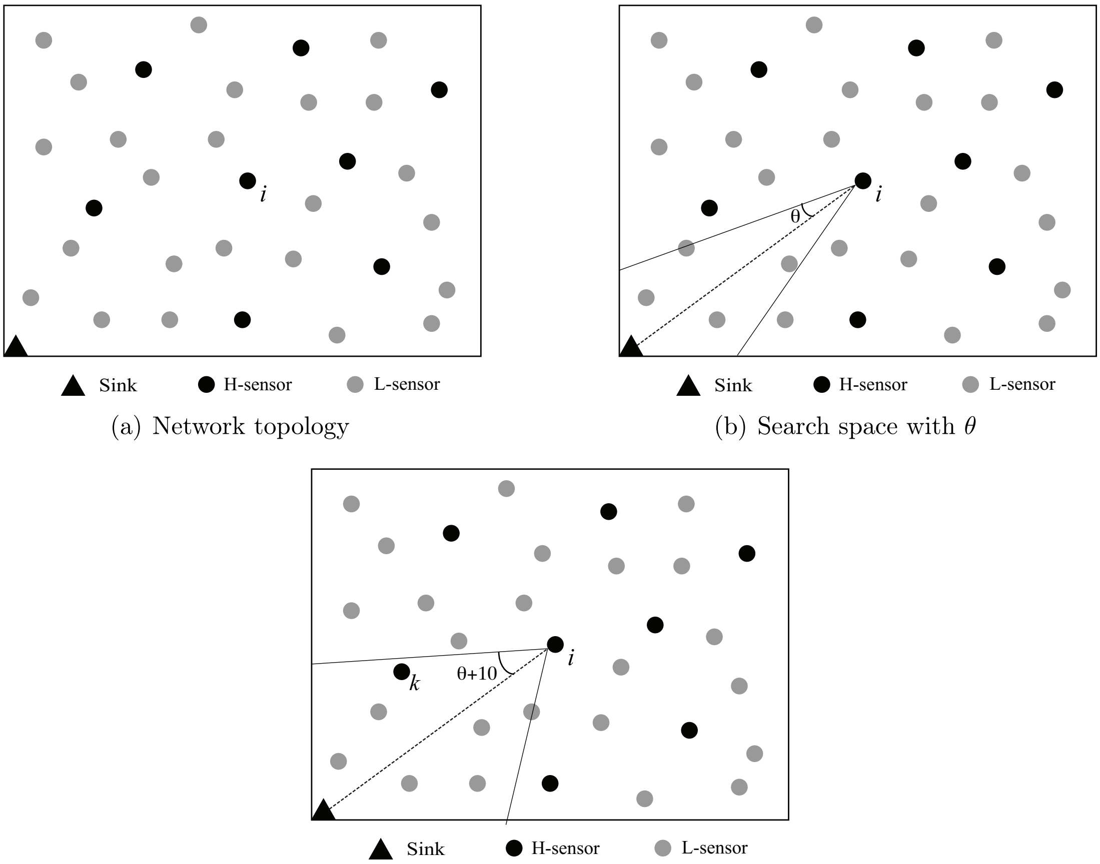
In this section we present the DASM Tree model (DASM-T) to design HSN topologies and how the proposed model creates a tree topology among the H-sensors in a distributed fashion connecting the H-sensors to the sink node. The goal of the model is to create a tree topology to improve the data communication in HSN. Section 5.2.4.1 introduces the proposed model as well as the adjustable search space. Section 5.2.4.2 describes the DASM Tree topology's routing algorithm.

5.2.4.1 Shortcut Creation Algorithm

As stated previously, a wireless sensor network can be tuned into a small world network adding a few shortcut links. In this way, a few shortcuts can reduce the average path length, keeping high values of clustering coefficient. However, in a WSN we can not choose two random nodes to create a shortcut between them due to hardware constraints. In this way, to add shortcuts in WSNs, we have to deploy H-sensors in the network and the communication among H-sensors will work as shortcuts. These nodes have a higher communication range and energy reserve to support this long range communication. Also, these nodes are equipped with two wireless interfaces, one to communicate with the H-sensors and another to communicate with the L-sensors. Based on this, a long transmission performed by an H-sensor will not interfere in the communication performed by the L-sensors.

To understand how the proposed model creates a tree-based topology, we need to introduce the adjustable search space. For this, each deployed H-sensor defines its search space to discover other H-sensors. The shortcuts will be created between two nodes that are inside the search space. As the search space is limited, directed to the sink node and does not consider the direction opposite to it, the shortcuts will create a connected topology among the H-sensors. To define the search space, each H-sensor uses an angle θ. The goal of this model is to start with a small angle, and when necessary, each H-sensor will increase the angle to find H-sensors. As a result of this process, the model creates a connected topology among the H-sensors.

Figure 5.2 illustrates the adjustable search space of a node i. Figure 5.2(a) shows the nodes deployment (L-sensors and H-sensors) highlighting the H-sensor i. Figure 5.2(b) shows the search space of the H-sensor i. We can see that because of the small angle θ used and because of the low density of H-sensors in the network, there is no H-sensor inside its search space. When this happens, the node gradually increases the search space increasing the angle θ. Figure 5.2(c) shows the search space using $\theta + 10$. We can see that using a bigger angle, the H-sensor i can find the H-sensor k. After this procedure, the H-sensor i can create a shortcut between them. Using this process for all H-sensors, the model creates a connected topology among them. In the following we describe in details how the DASM Tree model creates the shortcuts among the H-sensors.



(a) Network topology

(b) Search space with θ

(c) Search space with $\theta + 10$

Figure 5.2. Adjustable search space to the sink node

Algorithm 4 describes the operation of the shortcut creation of the DASM Tree model. The variables used in the algorithm are:

- x_i, y_i: Node's position;
- $sink_x$, $sink_y$: Sink's position;
- id_i: Node's id;
- NT_i: Neighbor table;
- SNT_I: Search neighbor table;
- ANG: Hash table, key: id, content: angle;
- θ: Angle to define the search space;

- m_i: The angular coefficient of the straight line that passes by the geographic position of the node that is executing the procedure and the geographic position of the sink node;

- m_k: The angular coefficient of the straight line that passes by the geographic position of a node k the geographic position of the sink node;

- $tangentOfAngle$: Defines the tangent of the angle between two lines;

This algorithm is executed during the start up time of each H-sensor. The algorithm works as follows. Lines 1 to 7 define all the variables used in the algorithm. Firstly, each H-sensor in the network sends a *hello* message that contains its ID and geographic position using the H-sensor interface (Line 9). Each H-sensor stores the received *hello* messages in a H-sensor neighbor table (Line 10).

In order to discover all H-sensors that are inside its communication range and inside its search space, each H-sensor performs the following steps. Each H-sensor calculates the angular coefficient m_i of the straight line that passes by its geographic position and the geographic position of the sink (Line 12). For each H-sensor v_k with ID k in the neighbor table, it calculates the angular coefficient m_k of the straight line that passes by its geographic position and the geographic position of H-sensor v_k (Line 14). With the angular coefficient of these lines, we can calculate the angle formed by them. After this step, the node calculates the tangent of the angle formed by these two angular coefficients. This tangent can be calculated making $\frac{m_i - m_k}{1 + m_i m_k}$ (Line 15). The node also stores in a hash table the ID of the node k and the tangent of the angle formed by the two angular coefficients (Line 16).

In Lines 19 to 27, for each entry in the hash table, the node i verifies which one of the H-sensors are inside its search space considering the angle θ. Figure 5.2 illustrates the search space of a H-sensor i. For this, the node makes $tangentOfAngle < \tan\theta$ (Line 22). If in this case, node i stores in the search neighbor table the verified H-sensor (Line 23). After evaluating all entries in the hash map, the node increases the search space[1] and verifies if the search neighbor table is not empty, which means that the node has at least one H-sensor inside the search space, or if the whole search space was already verified ($\theta < 360°$). If the node has at least one H-sensor in the search neighbor table, one of them is chosen at random and after this, function *createShortcut* is called to send a message to the other endpoint creating a shortcut between the two

[1]The gap to increase the search space was defined to be 10 degrees. However, this gap could be defined as 1, 2, 5 or any other angle. Small values of this gap, more steps would be necessary to verify the entire search space.

H-sensors. In Line 32, the algorithm returns the created shortcut to be used for data communication among H-sensors.

Algorithm 4 Shortcut Creation of the DASM Tree model

```
1: procedure CreateShortcut( )
2:     x_i, y_i;
3:     id_i;
4:     NT_i ← ∅;
5:     SNT_i ← ∅;
6:     ANG_i ← ∅;
7:     θ ← 10°;
8:
9:     broadcastHelloPacket(x_i, y_i, id_i)
10:    NT_i ← receiveHelloPacket( )
11:
12:    m_i ← getAngularCoefficient(x_i, y_i, sink_x, sink_y)
13:    for all v_k ∈ NT_i do
14:        m_k ← getAngularCoefficient(x_i, y_i, x_k, y_k)
15:        tangentOfAngle ← (m_i − m_k) / (1 + m_i m_k)
16:        ANG_i.insert(k, tangentOfAngle)
17:    end for
18:
19:    while θ < 360° and SNT_i is empty do
20:        for all key ∈ ANG_i do
21:            tangentOfAngle ← ANG_i.get(key)
22:            if tangentOfAngle < tan θ then
23:                DNT_i.insert(key)
24:            end if
25:        end for
26:        θ ← θ + 10
27:    end while
28:
29:    if DNT_i is not empty then
30:        endpoint ← choseAtRandom(DTN_i)
31:        createShortcut(endpoint)
32:        return endopint
33:    end if
34: end procedure
```

It is important to point out that the proposed model starts with a small angle to define the search space. Because of this, the model tries to create a straightforward shortcut. However, due to the low density of H-sensors in the network or the small angle θ that defines the search space, the region of some H-sensors may not contain other H-sensors. In this case, the model increases monotonically the θ angle in order to find H-sensors to create a shortcut between them. Also, when $\theta = 360°$ and a H-sensor does not find other H-sensors, it means that the H-sensor does not have other H-sensors inside its communication range. On the other hand, if all H-sensors have at least another H-sensor inside their search space, the model creates a connected topology among the H-sensors since all shortcuts are created using the directed search space.

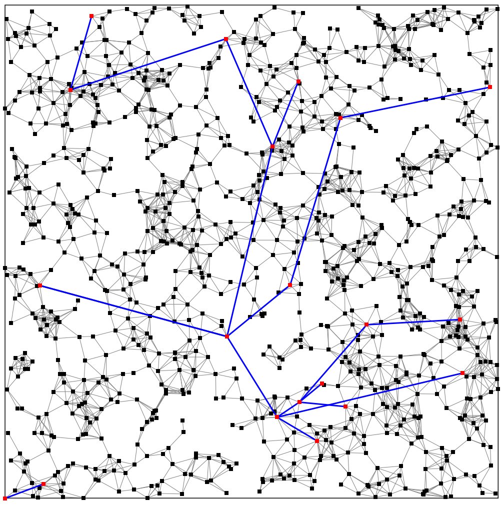
Figure 5.3 illustrates the shortcut creation in the proposed model. In both figures, the network has 1000 nodes including L-sensors and H-sensors. The L-sensor and H-sensor communication ranges are respectively 50 m and 400 m, and the sensor field is $1000 \times 1000\,m^2$. The light gray edges represent the connectivity between L-sensors and the black edges represent the connectivity between H-sensors after the Algorithm 4. When the density of H-sensors in the network is very low (Figure 5.3(a)) , there are some H-sensors that do not have other ones inside their communication range. Because of this, the created topology among the H-sensors is not connected. When we have at least one H-sensor inside the communication range of each H-sensor, we can see that the proposed model creates a connected topology among them (Figure 5.3(b)). However, because of the low density of H-sensors, some shortcuts are not totally directed to the sink node. In this case, the important aspect is to connect a H-sensor with the remaining H-sensor network. Figure 5.3(c) illustrates such a network when the number of H-sensors is 80. We can see that because of the high density of H-sensors, the network topology among them is connected.

5.2.4.2 DASM Tree Topology's Routing Algorithm

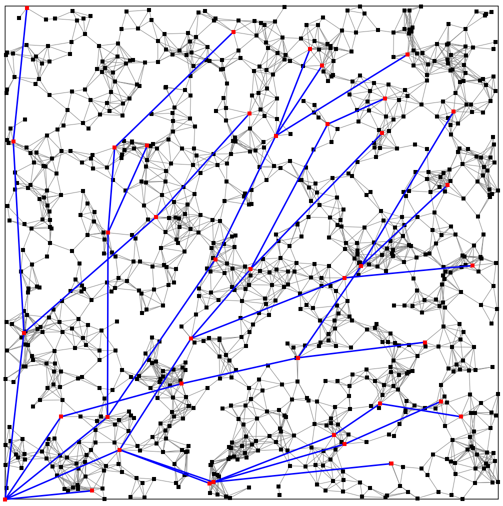
The previous section described the shortcut creation algorithm to create shortcuts in such a way that the connectivity among the H-sensors creates a tree topology. Back to Algorithm 4, the function *createShortcut*(*endpoint*), at Line 31, sends a message notifying the other endpoint of the created shortcut and consequently, both endpoints know the existence of each other. Also, it is important to note that an H-sensor can be chosen by more than one H-sensor. In this way, a specific H-sensor has one shortcut chosen by it and others chosen by different H-sensors.

However, even in the case that the H-sensors know their father's *id* in the tree topology, the L-sensors do not know. In this way, we will describe the algorithm to create the DASM Tree topology's routing. The variables used in the algorithm are:

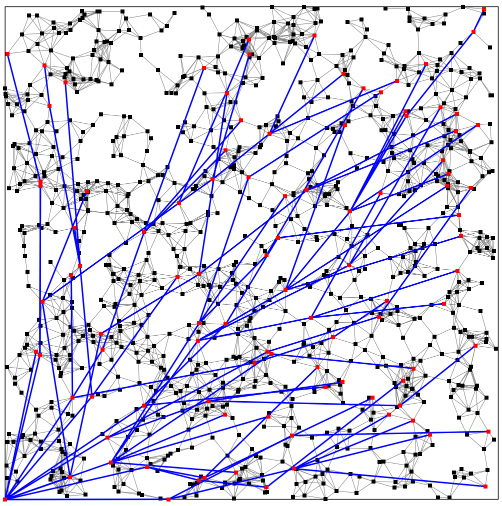
- id_i: Node's *id*;



(a) H = 20



(b) H = 40



(c) H = 80

Figure 5.3. Examples topologies using the DASM Tree model

- fDT_i: Father's id and the associate interface in the DASM Tree topology of a node i;
- *msg*: Variable to receive a message from other nodes;
- *L-interface*: Indicate the *id* of the L-sensor wireless interface;
- *H-interface*: Indicate the *id* of the H-sensor wireless interface;
- *shortcuts*: List of shortcut's endpoints;

The main difference from Algorithm 3 is when an H-sensor forward the sink's message to create the routing structure. Since the H-sensors may have more than one shortcut, the algorithm uses a multicast message to notify all the endpoints using one message[2]. The variable *shortcuts* contains all the endpoints received when the function $createShortcut$ was called.

Algorithm 5 Algorithm to create the DASM Tree topology's routing

```
procedure DASMTreeTopologyRouting( )
    id_i;
    fDT_i ← null;
    shortcuts;

    if  id_i == sink then
        broadcast(id_i, L-interface);                    ▷ Using the L-sensor interface
        multicast(id_i, shortcuts, H-interface);         ▷ Using the H-sensor interface
        return
    end if

    if fDT_i == null then
        msg ← receivePacket();
        fDT_i.id ← msg.id
        fDT_i.interface ← msg.interface
        broadcast(id_i, L-interface);                    ▷ Using the L-sensor interface
        multicast(id_i, shortcuts, H-interface);         ▷ Using the H-sensor interface
    end if
end procedure
```

After the execution of the DASM Tree Topology's routing algorithm, each node (L-sensor or H-sensor) knows its father's *id* in the tree structure. It is important to

[2]In this work we consider that the multicast function to transmit a message to a group of nodes one hop away is provided by the protocol stack and we will not go deeper in this routing function.

point out that the father's *id* of an H-sensor will always be other H-sensor because of the multicast communication used. In this way, during the route creation an H-sensor will first receive a package from an H-sensor and consequently, the node drops other messages from H-sensors or L-sensors.

5.3 Framework Instantiation

After the creation of the three topologies, we need to describe the QoS routing algorithm that uses the three topologies to perform the data collection. As we showed before, each topology has its own objectives. The homogeneous topology was designed in such away to reduce the energy consumption during data communication. Because of that, the number of hops to deliver a packet is higher than in the other topologies as expected. To reduce the communication latency, we introduced the DASM topology in which random and directed shortcuts are created. Because of the greater communication range of the H-sensor to provide the shortcuts, the energy consumption of these nodes is high. However, as the shortcuts among the H-sensors do not create a connected topology among them, some L-sensors are used to forward routing messages and consequently, the energy consumption by the H-sensor is reduced. To decrease even more the communication latency, we proposed the DASM Tree topology. In this topology, when a packet arrives at any H-sensor, the packet is forwarded to the destination (sink node) using only the shortcuts, i.e., connection among H-sensors. The disadvantage of this kind of communication is the energy consumption of the H-sensors that is high because of their greater communication range.

Based on these observations, we can consider that each topology provide different quality of services in terms of energy consumption and data communication latency. In this way, the framework can provide three different levels of QoS: (i) low energy consumption and high latency – homogeneous topology; (ii) average energy consumption and average latency – DASM topology and (ii) high energy consumption and small latency – DASM Tree topology. The terms low, average and high were used comparing the features of the proposed topologies.

The QoS routing algorithm works as follow. When an event is detected by a node or by a group of nodes, the node verifies the QoS level of the detected event. For example, if the node wants to notify the sink node about the humidity or atmospheric pressure, the packet may be routed using the homogeneous topology, since this kind of information is not urgent compared (low priority) to other ones. If the sensed information is important but not critical, the DASM topology should be used (average priority). Finally, if the sensed information is critical (fire detection system) the node should use the tree topology to deliver the packet quickly (high priority). Algorithm 6 describes the QoS routing algorithm. When a node receives a packet from the application (Line 1) or from other node, it verifies the priority of the sensed information (Lines 6, 10 and 14) and based on the priority the node chooses the correct path to send the packet (Lines 7, 11 and 15). The routing stops when the packet arrives at the sink node.

Algorithm 6 QoS Routing Algorithm

```
1: procedure QoSRouting( Packet p from application or other node)
2:     fH_i;                      ▷ Node's father in the homogeneous topology
3:     fD_i;                      ▷ Node's father in the DASM topology
4:     fDT_i;                     ▷ Node's father in the DASM Tree topology
5:
6:     if p.priority = low then
7:         p.next_hop ← fH_i
8:     end if
9:
10:    if p.priority = average then
11:        p.next_hop ← fD_i
12:    end if
13:
14:    if p.priority = high then
15:        p.next_hop ← fDT_i
16:    end if
17:
18:    sendPacket(p)
19: end procedure
```

The complexity analysis to create the three topology's routing algorithms can be found at Section 3.5, since the number of exchange messages is similar for all algorithms.

5.4 Simulation Results

5.4.1 Scenarios

In all simulations we use a network with 1000 nodes (including L-sensors and H-sensors). All nodes are randomly deployed in a sensor field of 1000x1000 m^2. The communication ranges of the L-sensors and H-sensors are 50 m and 400 m respectively. In such a network, each L-sensor has, on average, 8 neighbors. All results correspond to the arithmetic average of n simulations, with n different network topologies, where n is defined according to the confidence interval desired in the simulation [Jain, 1991]. Let $n = \left(\frac{100zs}{ew}\right)^2$, where s is the standard deviation, w is the average of the initial sample of 10 simulations[3], and z is the normal variate of the desired confidence level. In all simulations, a confidence interval of 95% ($e = 0.05$) was used.

We evaluated the Homogeneous, the Directed Angulation Towards the Sink Node Model (DASM) and the DASM Tree model (DASM-T) using the Network Simulator (NS-2) [Ns2, 2007]. The L-sensors were configured to work as a MicaZ node and the H-sensors were configured to work as a Stargate node.The initial angle ϕ used in the DASM-T was 10 degrees and used in DASM was 30 degrees. We consider that the H-sensors know their positions. The node's positions are used to compute the search space. To perform all evaluations, each sensor node sends one message to the sink node during the simulation time. The latency evaluation corresponds to the average of all messages.

5.4.2 Methodology

We compare our framework (Homogeneous, DASM and DASM-T topologies) to the heterogeneous topology used by the literature work to create heterogeneous sensor networks [Du et al., 2008; Samundiswary et al., 2009; Bagchi et al., 2011; Xu et al., 2010; Cardei et al., 2008; Du et al., 2007b; Barsi et al., 2011; Li et al., 2010]. Most of the literature works to create a heterogeneous topology deploy a few number of H-sensors in the network and each H-sensor uses all available links that connect it to its neighbors. The difference among these works is how the routing algorithm uses the infrastructure provided by the H-sensors in the network. Therefore, these works aim to propose different strategies regarding the routing problem and do not consider the design of heterogeneous topologies. The literature of QoS in heterogeneous sensor networks is based on the same concepts. Most of the proposed works [De and Sang, 2009; Nabi et al., 2011; Choe et al., 2010, 2009] uses a heterogeneous topology to improve some network metrics and then, providing QoS in sensor networks. However, these works do not consider the topological features behind the design of heterogeneous sensor network topologies to provide QoS. Based on these observation, we compare our solution to the solution that deploys H-sensors in the network considering all possible connections among them. We named this topology as Generic topology.

To evaluate the topologies, we will analyze the following topological metrics and network metrics:

[3]This value can be defined arbitrarily. The larger the initial sample, the smaller the value of n.

- Small world features: we evaluate the average path length and the clustering coefficient to verify if the created topology has small world features.

- Number of connected components: we evaluate the number of connected components considering the H-sensors. If the H-sensor infrastructure has one connected component, it means that we have a path among any pair of H-sensor in the network. In the case that we have more than one component, these components are connected using the L-sensor infrastructure.

- Latency: we evaluate the latency during data collection considering different number of H-sensors deployed in the network. We also show the relationship between the average path length and latency. Also, the latency is calculated considering the data collection communication.

- Energy consumption: we show the energy consumption of all evaluated topologies considering the overhead to create the topology, routing and the data collection process.

5.4.3 Small World Features

To check if a network has small world features, we need to evaluate the average shortest path length[4] (L) and the clustering coefficient (C) of the network without shortcuts and the network with shortcuts. If the network with shortcuts presents a smaller value of average shortest path length and similar value of clustering coefficient compared to a network without shortcuts, we can define such a network as a small world network [Newman and Watts, 1999]. Value $C(0)$ represents the clustering coefficient when the network does not have shortcuts (network composed only by L-sensors). Value $L(0)$ represents the average shortest path length when the network does not have shortcuts. Values $L(H)$ and $C(H)$ represent the same metrics when the network has H H-sensors. To show the influence of adding H H-sensors in the original network, we compute the ratio between $C(H)/C(0)$ and $L(H)/L(0)$.

Figure 5.4(a) shows the metrics L and C of the DASM topology. When $H < 40$, the average shortest path length is reduced by 47%. In this case, the network does not have small world features yet. However, when $H = 80$, the average shortest path length is reduced by 59%, and presenting high values of clustering coefficient which leads to a network with small world features. For $H > 80$, the average shortest path length does not reduce significantly compared to $H = 80$.

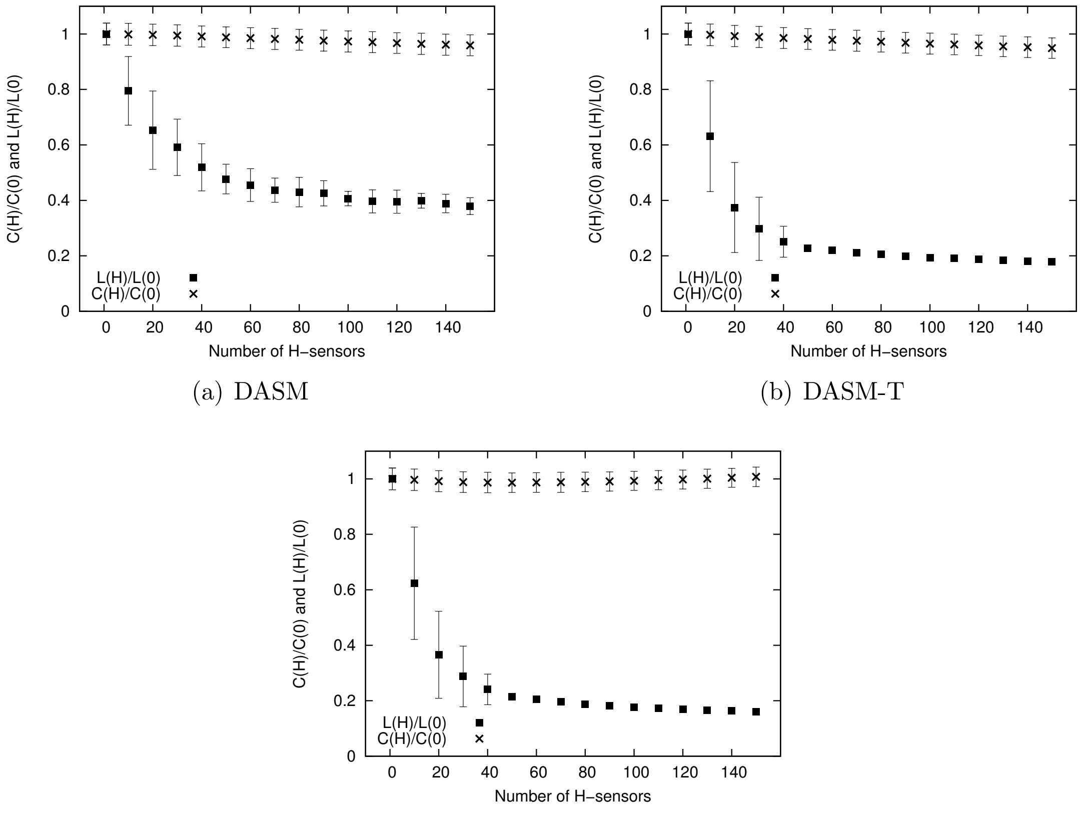
[4]The average shortest path length is calculated between the sink node and sensor nodes instead of between all pairs of nodes.

Figure 5.4(b) shows the metrics L and C of the DASM-T topology. Because of its strategy to create the shortcuts building a connected topology among the H-sensors, by deploying a few number of H-sensors the average path length between the sink node and the network nodes is significantly reduced. For instance, when $H = 40$, the average path length is reduced by 74%, keeping high values of clustering coefficient. In this way, the network presents small world features. Also, we can see that for values of $H > 40$, the path length does not reduce significantly.

Figure 5.4(c) illustrates the same metrics using the Generic topology to create a HSN. When $H = 40$, the average path length is reduced by 75%, keeping high values of clustering coefficient. For this number of H-sensors, the network presents small world features as well. It is important to note that because of the connection among the H-sensors, when we increase the number of H-sensors in the network the value of clustering coefficient increases as well. This happens because the Generic model to create a HSN considers all possible connections among the H-sensors to create the topology. Also, we can see that for values of $H > 40$, the path length does not reduce significantly.

Figure 5.5 shows the comparison among the three topologies considering just the average path length metric. We can see that the DASM-T topology has the same values of path length compared to the Generic topology. It is important to point out two issues: (i) the Generic topology considers all possible connections among the H-sensors and (ii) the DASM-T topology considers just the created shortcuts among the H-sensors. Based on these observations we can see that it is not necessary to have all possible connections among the H-sensors to reduce significantly the average path length between the sink node and sensors nodes. Considering just a few connections we can have the same effect in the network. Also, the average path length of the DASM-T and Generic topologies is 56% lower compared to the DASM topology.

After evaluating all these parameters for all topologies, the following is apparent: (i) to significantly reduce the average path length compared to the network without H-sensors, the DASM topology needs 80 H-sensors while the DASM-T and Generic topologies need just 40 H-sensors; (ii) for all values of H, the DASM-T and Generic



(a) DASM

(b) DASM-T

(c) Generic

Figure 5.4. Small world features

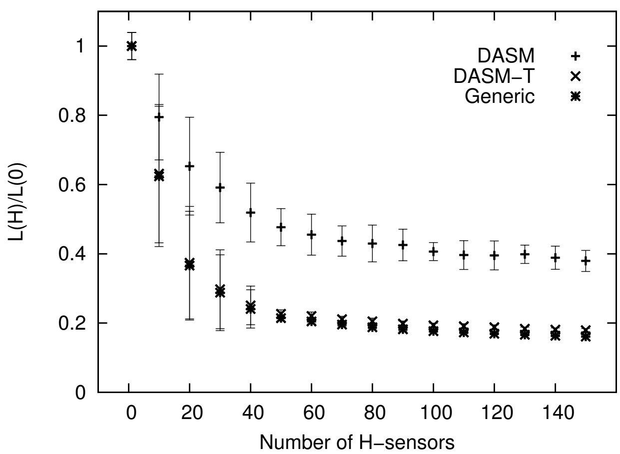


Figure 5.5. Comparison among the three topologies

topologies present smaller values of path length keeping high values of clustering coefficient compared to the DASM; (iii) the DASM-T and Generic topologies present similar values of path length; and (iv) when the network has more than 40 H-sensors, the path length does not reduce significantly for the DASM-T and Generic topologies. In this way, for the network configuration used in this work, it is not necessary more than 40 H-sensors to convert a regular network into a network with small world features.

5.4.4 Number of Connected Components

Figure 5.6 shows the number of connected components among the H-sensor network for the evaluated topologies. A connected component means that we have at least one path between any pair of nodes (H-sensors and sink) inside the component. In this case, we are evaluating the number of components using just the H-sensor wireless interface, used by the H-sensors to communicate with them. When we increase the number of H-sensors deployed in the network, the number of connected components of the DASM topology increases as well. This happens because the DASM model tries to create one shortcut per H-sensor. Because of the number of connected components, some L-sensors work as bridges to interconnect two or more components (set of H-sensors and their shortcuts) that are spatially separated.

As we presented in Section 5.2.4.1, the DASM-T model increases monotonically the search space in order to find other H-sensors. In this way, the model tries to create a connected topology among the H-sensors. Moreover, when the number of H-sensors in the network is greater than 40, the proposed model creates a connected network among the H-sensor network. With this feature, when the sink node sends a message to entire network or when the sensor nodes send their sensed information to the sink node, the shortcuts are fully used to better improve the data communication latency. Also, the L-sensors do not work as bridges to connect two components that are spatially separated considering this number of H-sensors deployed in the network. We can observe the same behavior in the Generic topology. When the network has more than 40 H-sensors the topology among the H-sensors is connected.

5.4.5 Latency Evaluation

The data communication latency is a reflection of the average shortest path length between the sink node and the sensor nodes. Figure 5.7 shows the data communication

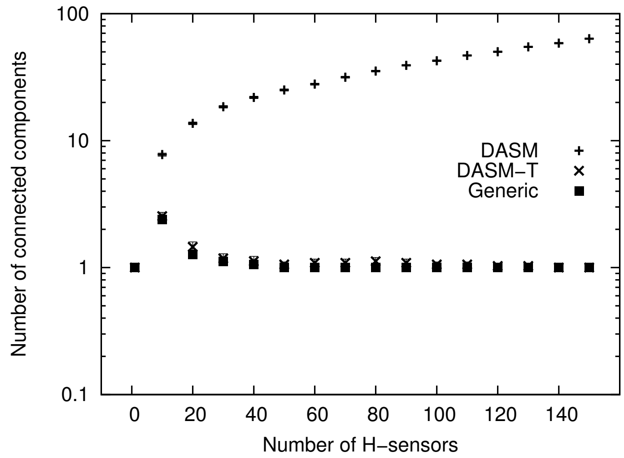
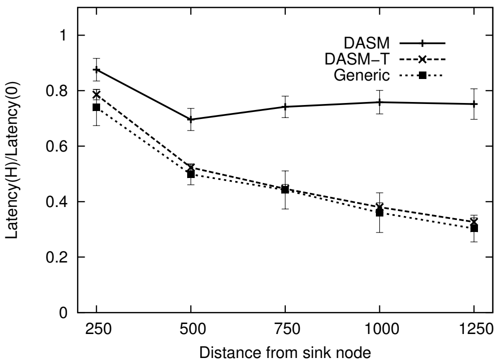


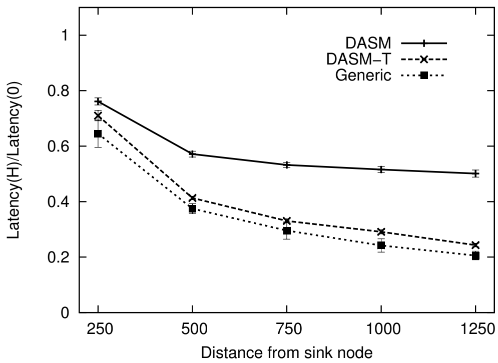
Figure 5.6. Number of components among the H-sensor network

latency for different numbers of H-sensors and for five different distances from the sink node (250, 500, 750, 1000 and 1250 m). In all results, as in Section 5.4.3, we calculate the ratio $Latency(H)/Latency(0)$. Figure 5.7(a) shows the latency when the network has 20 H-sensors. We can see that because of the low number of deployed H-sensors, the latency during data communication does not reduce significantly in the DASM topology. For instance, when the distance from the sink node is up to 750 m, the DASM topology's latency is reduced by 25% compared to the topology without H-sensors (homogeneous topology). Back to Figure 5.4, we can see that for this number of H-sensors, the network topology does not have small world features. Considering the DASM-T and Generic topologies and when the distance from the sink node is up to 750 m, the latency during data communication is reduced by 55% compared to the homogeneous topology and reduced by 40% compared to the DASM topology.

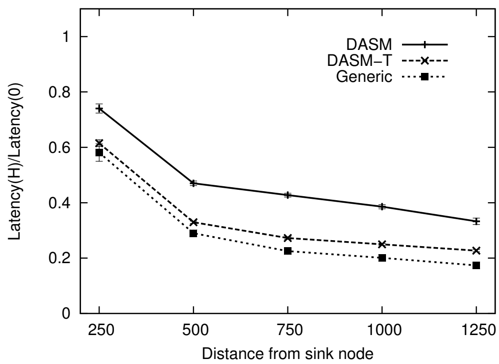
When the number of H-sensors increases (Figure 5.7(b)), the latency in all topologies decreases as expected. For instance, when the distance from the sink node is up to 1250 [m], the DASM topology's latency is reduced by 53% compared to the topology without H-sensors. As the evidence of the small world features happens for $H > 20$ in the DASM-T and Generic topologies, the latency in both topologies does not decrease by introducing more H-sensors as the latency decreases in the DASM topology. For instance, when the distance from the sink node is up to 750 m and $H = 40$, the latency in both topologies is reduced by 25% compared to when the topologies have



(a) 20 H-sensors



(b) 40 H-sensors



(c) 80 H-sensors

Figure 5.7. Latency evaluation

20 H-sensors. Also, the latency is reduced by 66% compared to the network without H-sensors and is reduced by 38% compared to the DASM topology.

Figure 5.7(c) shows the data communication latency when the number of H-sensors deployed in the network is 80. As the evidence of the small world features in the DASM-T and Generic topologies already exists in the network, the latency during data communication does not reduce proportional to the number of H-sensors deployed. For instance, when the distance from the sink node is up to 1250 m, both DASM-T and Generic topologies reduce the latency by 7% compared to 40 H-sensors in the network. On the other hand, as the evidence of the small world in the DASM model happens when $H > 80$, the latency when the distance from the sink is up to 1250 m is reduced by 63% compared to the network without H-sensors.

Evaluating the data communication latency of all topologies for different numbers of H-sensors deployed in the network, the following is apparent. When the network has 40 H-sensors, the DASM topology do not reduce the data communication latency significantly. On the other hand, for the same value of H-sensors, the DASM Tree and

Generic topologies reduce the latency by more than 60%. Increasing the number of H-sensors to 80 and comparing to 40 H-sensors, the data communication latency of both topologies reduce by just 7%. In this way, it is not necessary more than 40 H-sensors to convert a wireless sensor network into a heterogeneous sensor network with small world features in order to reduce the data communication latency. However, as the goal of the framework is to design topologies considering different QoS requirements, the DASM topology shows to be suitable for application that are not critical in terms of delay considering 40 H-sensors. Considering critical applications, the DASM-T topology created by the framework is applicable. Also, the DASM-T topology has the same results, on average, compared to the Generic topology.

5.4.6 Energy Evaluation

In this section we present the results regarding the energy consumption during the routing process. In this case, each node sends one reading message to the sink node during the simulation time. We also consider the energy spend during the topology creation of the framework and the Generic topology. To measure the energy consumption during data communication, we considered just the energy to transmit or receive a packet. We did not consider the energy consumption in the idle mode, since this mode will be the same in all topologies. Based on the MicaZ node, the energy to transmit a packet is 0.0744 W and the energy to receive a packet is 0.0648 W. We used these values to measure the energy consumption in our simulations. These energy consumption values were used considering a communication range of 50 m. To calculate the energy consumption of the H-sensors with a communication range of 400 m, we considered the function R^α to calculate the energy dissipation where R is the communication range and α is the path loss exponent. We used $\alpha = 2$.

We evaluated the Homogeneous, DASM, DASM-T and Generic topologies and the energy consumption was calculated routing all the packets considering each topology separately. We also evaluate the Homogeneous, DASM and DASM-T topologies together (our framework solution). In this case, a fraction of 1/3 of the messages were routed using the Homogeneous topology, a fraction of 1/3 of the messages were routed using the DASM topology and a fraction of 1/3 of the messages were routed using the DASM-T topology. We divided the routing in this scheme to evaluate the three topologies considering a load balanced scenario. Once a single packet is routed using one topology, the packet will be forwarded by that topology until its reaches the sink node. The energy consumption of the framework solution was calculated routing all the packets considering the three topologies.

H-sensors	Homogeneous	DASM		DASM-T		Generic		Framework	
		H	L	H	L	H	L	H	L
10	3.19	55.90	1.10	130.26	1.47	128.33	0.37	63.06	1.10
20	3.19	48.93	1.01	93.05	0.24	92.47	0.19	48.89	1.23
40	3.19	36.01	0.34	48.71	0.17	48.23	0.11	30.26	1.17
60	3.19	31.25	0.29	33.63	0.15	33.83	0.08	23.32	0.92

Table 5.1. Energy consumption in Joules. L (L-sensors) and H (H-sensors)

Table 5.1 shows the average energy consumption of all evaluated topologies considering different number of H-sensors deployed in the network. Our baseline to compare the energy consumption is the energy consumption of the Homogeneous topology. As we can see, the energy spent by the homogeneous topology during data communication is 3.19 J. It is important to note that the homogeneous topology uses just the L-sensors to perform the routing. The energy consumption of the Generic and DASM-T topologies considering the H-sensors is higher compared to the DASM one in all cases. This is because once a packet reaches the infrastructure of H-sensors, the packet will be forwarded in the H-sensor topology until it reaches the sink node without using any L-sensors. Because of that, the energy consumption of the L-sensors are lower compared to the DASM one. As the framework uses the Homogeneous, DASM and DASM-T topologies to perform the data collection, considering the H-sensors, its energy consumption is greater than the DASM one and lower compared to the DASM-T and Generic topologies. Based on these observations the following is apparent. Our framework provides different network topologies where each topology has its energy consumption that is proportional to the latency during data communication. To provide a topology with a low latency during data communication, more H-sensors will be used to forward the packets and then, more will be the energy consumption of these nodes. In some applications, the latency is not critical. In these cases, the framework provides the DASM and Homogeneous topologies, that could be used to reduce the energy consumption of the H-sensors and L-sensors and then, improving the lifetime of both H-sensors and L-sensors.

5.5 Final Remarks

In this Chapter we presented a new Framework based on the Small World concepts to design Heterogeneous Sensor Networks with QoS. The framework provides three different topologies and each topology has its own objectives in terms of latency and energy consumption during data communication. Simulation results showed that when the network has 40 H-sensors, the Framework provides three topologies in which three different levels of QoS are archived. With this number of H-sensors, the DASM topology showed to be appropriate for applications that require average latency and energy consumption during data communication while the DASM-T provides smaller latency showing to be suitable for critical applications. However, the DASM-T spends more energy compared to the DASM one. Also, the homogeneous topology showed to be suitable for applications that do not require a small latency during data communication. As a future work we are planning to design a new routing algorithm to better use the Framework's topologies. The algorithm will be designed using the concept of *deadline* to deliver a sensed information and during the routing, the algorithm will change the routing topology among the three topologies to deliver the package on time.

Chapter 6

Conclusion and Future Work

This Chapter is organized as follows. Section 6.1 presents the summary of this thesis and Section 6.2 presents the future work.

6.1 Summary of this Work

This document presented the dissertation entitled Applying the Small World Concepts on the Design of Heterogeneous Wireless Sensor Networks. In this work, we proposed the TDASM, ODASM, DASM-MC and a Framework to design HSNs with QoS. The TDASM is a theoretical model to design heterogeneous sensor networks with small world features. The ODASM is the on-line version of the TDASM, where the shortcuts are created during the network lifetime. Simulation results showed that both models can create a heterogeneous wireless sensor network with small world features, and when the shortcuts are created directed to the sink node, the network presents better small world features compared to the original small world model (TRAM and ORAM). Also, we evaluated the resilience of those models and the following is apparent. When the shortcuts are directed to the sink, the network is more resilient in the presence of node's failures.

The shortcut creation in the ODASM model is done using the larger communication range of the H-sensors. The main drawback of the ODASM and ORAM is the medium access performed by a transmission in the H-sensor interface. To overcome this problem, we proposed the DASM-MC. This model was designed in such a way that each shortcut may use a different wireless communication channel. Based on this, when a H-sensor sends a message using the H-sensor interface (using the shortcut), the wireless channels of its neighbors are not affected. However, to assign a different wireless channel for each shortcut, each H-sensor should have just one shortcut. Simulation results showed that the DASM-MC is able to create heterogeneous sensor networks with small world features and when a shortcut is assigned in a different channel, the number of collisions during the data communication is reduced.

To improve even more data communication latency, we proposed the DASM-T model. The goal of this model is to create a connected topology among the H-sensors in order to decrease the latency. We also proposed a framework that creates three different network topologies to provide QoS: Homogeneous topology, DASM topology and DASM-T topology. Each topology has its own goals related to energy consumption and latency during communication. The homogeneous topology is applicable when the sensed information does not require low latency. In this way, the framework uses this topology to decrease the energy consumption to deliver that information. The DASM topology is applicable when the sensed information needs to be delivered as soon as possible but does not have critical deadline. When the sensed information needs to be delivered on time, the DASM-T topology should be used. The main drawback of the DASM-T is its energy consumption by the H-sensors. Because of that, this topology should be used carefully.

6.2 Future Work

As a future work, we plan to investigate the proposed topologies in this thesis in other network tasks, such as:

- *Data Fusion:* one of the major problems in data fusion protocols is to discover the aggregating nodes, i.e., nodes that will perform data fusion. These nodes usually receive data from different sources and then perform data fusion considering all received data. In our case, the H-sensors could work as aggregating nodes, since the majority of the messages will be routed by them. Since the H-sensors do not have hardware constraints like the L-sensors, we can apply a robust and efficient data fusion algorithm.

- *Position Estimation:* as the H-sensors have a powerful hardware, we can equip each H-sensor with a GPS receiver. In this case, the H-sensors can broadcast their position using their powerful communication radius. In this scenario, the L-sensors may receive positions from different H-sensors in order to accurately estimate their position.

- *Object Tracking:* propose solutions for the tracking of mobile objects. In this scenario, we can create random shortcuts in such a way to improve the tracking of objects in the networks. The shortcuts may also change during the network lifetime to better tracking the objects.

We are also planning to investigate other QoS metrics of our framework. Once we have three topologies, we need to investigate deeper the use of these topologies during the routing process. Also, we can evolve the proposed framework in such way to use more than three topologies. At one extreme we have a Homogeneous topology and at another extreme we have a Tree topology in terms of energy consumption and latency during data communication. Based on this, we can design more topologies between them to provide different levels of quality of service, instead of the three levels of quality of services provided by the proposed framework.

Bibliography

Akyildiz, I. F. and Kasimoglu, I. H. (2004). Wireless sensor and actor networks: research challenges. *Elsevier Ad Hoc Networks*, 2(4):351--367.

Akyildiz, I. F., Su, W., Sankarasubramaniam, Y., and Cayirci, E. (2002). Wireless sensor networks: A survey. *Elsevier Computer Networks*, 38(4):393--422.

Albert, R., Jeong, H., and Barabási, A.-L. (1999). Diameter of the world wide web. *Nature*, 401:130--131.

Albert, R., Jeong, H., and Barabási, A.-L. (2000). Error and attack tolerance of complex networks. *Nature*, 406:378--382.

Amaral, L. A. N., Scala, A., Barthélémy, M., and Stanley, H. E. (2005). Classes of small-world networks. *Proceedings of the National Academy of Sciences*, 97(21):11149--11152.

Bagchi, A., Madan, A., and Premi, A. (2011). A computationally inexpensive and power efficient fully distributed topology for data collection in heterogeneous wireless sensor networks. In *IEEE International Conference on Distributed Computing in Sensor Systems and Workshops*, pages 1--8.

Barabási, A.-L. (2003). *Linked: How Everything is Connected to Everything Else and What It Means.* Penguin Books Ltd.

Barsi, F., Bertossi, A., Lavault, C., Navarra, A., Olariu, S., Pinotti, M., and Ravelomanana, V. (2011). Efficient location training protocols for heterogeneous sensor and actor networks. *IEEE Transactions on Mobile Computing*, 10(3):377--391.

Boasa, P. R. V., Antiqueira, L., Viana, L. M. P., and da Rocha, L. E. C. (2011). Analyzing and modeling real-world phenomena with complex networks: A survey of applications. *Advances in Physics*, 60(3):329--412.

Borbash, S. A., Ephremides, A., and McGlynn, M. J. (2007). An asynchronous neighbor discovery algorithm for wireless sensor networks. *Elsevier Ad Hoc Networks*, 5(7):998--1016.

Cardei, M., Yang, S., and Wu, J. (2008). Algorithms for fault-tolerant topology in heterogeneous wireless sensor networks. *IEEE Transactions on Parallel and Distributed Systems*, 19(4):545--558.

Cavalcanti, D., Agrawal, D., Kelner, J., and Sadok, D. (2004). Exploiting the small-world effect to increase connectivity in wireless ad hoc networks. In *IEEE International Conference on Telecommunications*, pages 388--393.

Chen, D. and Varshney, P. K. (2004). Qos support in wireless sensor networks: A survey. In *International Conference on Wireless Networks*, pages 1--7.

Chen, P., Wang, C., Li, L., and Liu, M. (2006a). Heterogeneous sensor networks with effect of small world. In *IEEE International Conference on Wireless, Mobile and Multimedia Networks*, pages 1--4.

Chen, X., Han, P., He, Q.-S., Tu, S.-l., and Chen, Z.-L. (2006b). A multi-channel mac protocol for wireless sensor networks. In *WASET International Conference on Computer and Information Technology*, pages 224--229.

Chinnappen-Rimer, S. and Hancke, G. (2009). Modelling a wireless sensor network as a small world network. In *International Conference on Wireless Networks and Information Systems*, pages 7--10.

Chinnappen-Rimer, S. and Hancke, G. (2010). Energy efficient message routing in a small world wireless sensor network. In *Advances in Wireless Networks and Information Systems*, pages 183--192.

Chitradurga, R. and Helmy, A. (2004). Analysis of wired short cuts in wireless sensor networks. In *IEEE International Conference on Pervasive Services*, pages 167--176.

Choe, H. J., Ghosh, P., Basu, K., and Das, S. K. (2009). Class-based data reporting scheme in heterogeneous wireless sensor networks. In *IEEE International Conference on Computer Communications and Networks*, pages 1--6.

Choe, H. J., Ghosh, P., and Das, S. K. (2010). Qos-aware data reporting control in cluster-based wireless sensor networks. *Elsevier Computer Communications*, 33(11):1244--1254.

Cidon, I. and Sidi, M. (1989). Distributed assignment algorithms for multihop packet radio networks. *IEEE Transactions on Computers*, 38(10):1353--1361.

da F. Costa, L., Rodrigues, F. A., Travieso, G., and Boasa, P. R. V. (2007). Characterization of complex networks: A survey of measurements. *Advances in Physics*, 56(1):167--242.

Das, A. K., Vijayakumar, R., and Roy, S. (2006). Static channel assignment in multi-radio multi-channel 802.11 wireless mesh networks: Issues, metrics and algorithms. In *IEEE Global Communications Conference*, pages 49--55.

De, D. and Sang, L. (2009). Qos supported efficient clustered query processing in large collaboration of heterogeneous sensor networks. In *IEEE International Symposium on Collaborative Technologies and Systems*, pages 242--249.

Demirkol, I., Ersoy, C., and Alagoz, F. (2006). Mac protocols for wireless sensor networks: A survey. *IEEE Communications Magazine*, 44(4):115--121.

do Val Machado, M., Mini, R. A., Loureiro, A. A. F., Guidoni, D. L., and de Melo, P. O. S. V. (2007). Gossiping using the energy map in wireless sensor networks. In *ACM Symposium on modeling, analysis, and simulation of wireless and mobile systems*, pages 368--372.

Du, X., Guizani, M., Xiao, Y., and Chen, H.-H. (2006). A secure routing protocol for heterogeneous sensor networks. In *IEEE Global Telecommunications Conference*, pages 1--5.

Du, X., Guizani, M., Xiao, Y., and Chen, H.-H. (2007a). A secure time synchronization scheme for heterogeneous sensor networks. In *IEEE Global Telecommunications Conference*, pages 5143--5147.

Du, X., Guizani, M., Xiao, Y., and Chen, H.-H. (2007b). Two tier secure routing protocol for heterogeneous sensor networks. *IEEE Transactions on Wireless Communications*, 6(9):3395--3401.

Du, X., Guizani, M., Xiao, Y., and Chen, H.-H. (2008). Secure and efficient time synchronization in heterogeneous sensor networks. *IEEE Transactions on Vehicular Technology*, 57(4):2387--2394.

Du, X. and Lin, F. (2005). Improving routing in sensor networks with heterogeneous sensor nodes. In *IEEE Vehicular Technology Conference*, volume 4, pages 2528--2532.

Egerstedt, M. (2011). Complex networks: Degrees of control. *Nature*, 473(7346):158--159.

Enz, C., El-Hoiydi, A., Decotignie, J.-D., and Peiris, V. (2004). Wisenet: An ultralow-power wireless sensor network solution. *IEEE Computer*, 37(8):62--70.

Estrin, D., Govindan, R., Heidemann, J., and Kumar, S. (1999). Next century challenges: Scalable coordination in sensor networks. In *ACM International Conference on Mobile Computing and Networking*, pages 263--270.

Guidoni, D. L., Boukerche, A., Mini, R. A., and Loureiro, A. A. (2010a). A small world model to improve synchronization algorithms for wireless sensor networks. In *IEEE International Symposium on Computers and Communications*, pages 229--234.

Guidoni, D. L., Boukerche, A., Souza, F. S., Mini, R. A., and Loureiro, A. A. (2010b). A small world model based on multi-interface and multi-channel to design heterogeneous wireless sensor networks. In *IEEE Global Communications Conference*, pages 1--6.

Guidoni, D. L., Boukerche, A., Souza, F. S., Mini, R. A., and Loureiro, A. A. (2011). A tree-based approach to design heterogeneous sensor networks based on small world concepts. In *IEEE P2MNET Local Computer Networks*, page Accepted for publication.

Guidoni, D. L., Boukerche, A., Villas, L. A., Souza, F. S., Mini, R. A., and Loureiro, A. A. (2010c). A framework based on small world features to design hsns topologies with qos. In *IEEE International Symposium on Computers and Communications*, pages 229--234.

Guidoni, D. L., Mini, R. A., and Loureiro, A. A. (2008a). Creating small-world models in wireless sensor networks. In *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, pages 1--6.

Guidoni, D. L., Mini, R. A., and Loureiro, A. A. (2008b). On the desing of heterogeneous sensor networks based on small world concepts. In *ACM International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, pages 309--314.

Guidoni, D. L., Mini, R. A., and Loureiro, A. A. (2009). A on-line model to design heterogeneous wireless sensor networks based on small world concepts. In *IEEE International Symposium on Computers and Communications*, pages 513--519.

Guidoni, D. L., Mini, R. A., and Loureiro, A. A. (2010d). On the design of resilient heterogeneous wireless sensor networks based on small world concepts. *Elsevier Computer Networks*, 54(8):1266--1281.

Gupta, P. and Kumar, P. (2000). The capacity of wireless networks. *IEEE Transactions on Information Theory*, 46(2):388--404.

Han, X., Cao, X., Lloyd, E. E., and Shen, C.-C. (2010). Fault-tolerant relay node placement in heterogeneous wireless sensor networks. *IEEE Transactions on Mobile Computing*, 9(5):643--656.

Hawick, K. A. and James, H. A. (2010). Small world effects in wireless agent sensor networks. *INDERSCIENCE International Journal on Wireless Mobile Computing*, 4:155--164.

Hebden, P. and Pearce, A. (2006). Distributed asynchronous clustering for self-organisation of wireless sensor networks. In *IEEE International Conference on Intelligent Sensing and Information Processing*, pages 37--42.

Hebden, P. and Pearce, A. (2007). Distributed asynchronous clustering for self-organisation of wireless sensor networks. *International Journal on Information Processing*, 1(2):1--10.

Helmy, A. (2003). Small worlds in wireless networks. *IEEE Communications Letters*, 6(10):490--492.

Hofmann-Wellenhof, B., Lichtenegger, H., and Collins, J. (1993). *Global Positioning System. Theory and practice.* Springer, Wien (Austria).

Intanagonwiwat, C., Estrin, D., Govindan, R., and Heidemann, J. (2002). Directed diffusion: A scalable and robust communication paradigm for sensor networks. In *IEEE International Conference on Distributed Computing Systems*, pages 263--270.

Jain, R. (1991). *The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation and Modeling.* John Wiley.

Jespersen, S. and Blumen, A. (2000). Small-world networks: Links with long-tailed distribution. *Physics Review E*, 62(5):6270--6374.

Jiang, C.-J., Chen, C., Chang, J.-W., Jan, R.-H., and Chiang, T. C. (2008). Construct small worlds in wireless networks using data mules. In *IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing*, pages 28--35.

Khan, M., Pandurangan, G., and Anil Kumar, V. S. (2009). Distributed algorithms for constructing approximate minimum spanning trees in wireless sensor networks. *IEEE Transactions on Parallel Distributed Systems*, 20(1):124--139.

Kleinberg, J. (2000a). The small-world phenomenon: An algorithmic perspective. In *ACM Symposium on Theory of Computing*, pages 163--170.

Kleinberg, J. (2008). The convergence of social and technological networks. *Communications of the ACM*, 51(11):66--72.

Kleinberg, J. M. (2000b). Navigation in a small world. *Nature*, 406:845--845.

Kozma, B., Hastings, M. B., and Korniss, G. (2005). Diffusion processes on power-law small-world networks. *Physics Review Letters*, 95(1):018701–1--018701–4.

Kushwaha, M., Oh, S., Amundson, I., Koutsoukos, X., and Ledeczi, A. (2008a). Target tracking in heterogeneous sensor networks using audio and video sensor fusion. In *IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, pages 14--19.

Kushwaha, M., Oh, S., Amundson, I., Koutsoukos, X., and Ledeczi, A. (2008b). Target tracking in urban environments using audio-video signal processing in heterogeneous wireless sensor networks. In *Asilomar Conference on Signals, Systems and Computers*, pages 1606--1610.

Latvakoski, J. (2009). Towards hierarchical routing in small world wireless networks. In *IARIA International Conference on Wireless and Mobile Communications*, pages 322--327.

Latvakoski, J. (2010). Hierarchical routing concept for small world wireless networks. In *IARIA International Conference on Wireless and Mobile Communications*, pages 325--330.

Lazos, L. and Poovendran, R. (2006). Stochastic coverage in heterogeneous sensor networks. *ACM Transactions on Sensor Networks*, 2(3):325--358.

Li, Y., Ai, C., Vu, C., Pan, Y., and Beyah, R. (2010). Delay-bounded and energy-efficient composite event monitoring in heterogeneous wireless sensor networks. *IEEE Transactions on Parallel and Distributed Systems*, 21(9):1373--1385.

Maia, G., Guidoni, D. L., Aquino, A. L., and Loureiro, A. A. (2009). Improving an over-the-air programming protocol for wireless sensor networks based on small world concepts. In *ACM International conference on modeling, analysis and simulation of wireless and mobile systems*, pages 261--267.

Menth, M. and Martin, R. (2005). Network resilience through multi-topology routing. In *IEEE International Workshop on Design of Reliable Communication Networks*, pages 271--277.

Mhatre, V. P., Rosenberg, C., Kofman, D., Mazumdar, R., and Shroff, N. (2005). A minimum cost heterogeneous sensor network with a lifetime constraint. *IEEE Transactions on Mobile Computing*, 4(1):4--15.

Mica (2006). Mica2 crossbow technology. `http://www.xbow.com/Products/productdetails.aspx?sid=164`.

Mica2 (2005). Mica2 crossbow technology. `http://www.xbow.com/Products/productdetails.aspx?sid=174`.

MicaZ (2009). Micaz crossbow technology. `http://www.xbow.com/Products/productdetails.aspx?sid=164`.

Milgran, S. (1967). The small world problem. *Psycology Today*, pages 60--67.

Nabi, M., Blagojevic, M., Geilen, M., and Basten, T. (2011). Dynamic data prioritization for quality-of-service differentiation in heterogeneous wireless sensor networks. In *IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks*, pages 296--304.

Newman, M., Barabasi, A.-L., and Watts, D. J. (2006). *The Structure and Dynamics of Networks*. Princeton Studies in Complexity.

Newman, M. E. J. (2003). The structure and function of complex networks. *SIAM Review*, 45:167--256.

Newman, M. E. J. and Watts, D. J. (1999). Scaling and percolation in the small-world network model. *Physics Review E*, 60(6):7332--7342.

Ns2 (2007). The network simulator ns-2. `http://www.isi.edu/nsnam/ns/`.

Oliveira, H. A. B. F., Boukerche, A., Guidoni, D. L., Nakamura, E. F., Mini, R. A. F., and Loureiro, A. A. F. (2011). An enhanced location-free greedy forward algorithm with hole bypass capability in wireless sensor networks. *Elsevier Computer Networks*, pages --.

Pottie, G. and Kaiser, W. (2000). Embedding the internet wireless integrated network sensors. *Communications of the ACM*, 43:51--58.

Ramanathan, R. and Steenstrup, M. (1998). Hierarchically-organized, multihop mobile wireless networks for quality-of-service support. *Springer Mobile Network Applications*, 3(1):101--119.

Raniwala, A. and cker Chiueh, T. (2005). Architecture and algorithms for an ieee 802.11-based multi-channel wireless mesh network. In *IEEE International Conference on Computer Communications*, pages 2223--2234.

Reka, A. and Barabási (2002). Statistical mechanics of complex networks. *Reviews of Modern Physics*, 74:47--97.

Sahni, S. and Gonzalez, T. (1976). P-complete approximation problems. *Journal of the ACM*, 23(3):555--565.

Samundiswary, P., Priyadarshini, P., and Dananjayan, P. (2009). Performance evaluation of heterogeneous sensor networks. In *International Conference on Future Computer and Communication*, pages 264--267.

Sharma, G. and Mazumdar, R. (2006). A case for hybrid sensor networks. In *ACM International Symposium on Mobile Ad Hoc Networking and Computing*, pages 366--377.

Shelby, Z., Pomalaza-Ráez, C., Karvonen, H., and Haapola, J. (2005). Energy optimization in multihop wireless embedded and sensor networks. *Springer International Journal of Wireless Information Networks*, 12(1):11--21.

So, J. and Vaidya, N. H. (2004). Multi-channel mac for ad hoc networks: handling multi-channel hidden terminals using a single transceiver. In *ACM international symposium on Mobile ad hoc networking and computing*, pages 222--233.

Souza, F. S., Guidoni, D. L., and Loureiro, A. A. F. (2008). Uma abordagem baseada em grasp para geração de topologias small world. In *Simpósio Brasileiro de Pesquisa Operacional*, pages 1--12.

Stargate (2006). Stargate crossbow technology. `www.xbow.com/Support/Support_pdf_files/Stargate_Manual.pdf`.

Strogatz, S. H. (2001a). Exploring complex networks. *Nature*, 410(6865):268--276.

Strogatz, S. H. (2001b). Exploring complex networks. *Nature*, 410:268--276.

Subramanian, A. P., Gupta, H., Das, S. R., and Cao, J. (2008). Minimum interference channel assignment in multiradio wireless mesh networks. *IEEE Transactions on Mobile Computing*, 7(12):1459--1473.

TelosB (2006). Telosb crossbow technology. `http://www.willow.co.uk/html/telosb_mote_platform.html`.

Verma C. K., Tamma B. R., M. B. and R., R. (2011). A realistic small-world model for wireless mesh networks. *IEEE Communications Letters*, 15(4):455--457.

Villas, L. A., Guidoni, D. L., Araujo, R. B., Boukerche, A., and Loureiro, A. A. F. (2010). A scalable and dynamic data aggregation aware routing protocol for wireless sensor networks. In *ACM international conference on modeling, analysis and simulation of wireless and mobile systems*, pages 110--117.

Villas, L. A., Guidoni, D. L., Araujo, R. B., Boukerche, A., and Loureiro, A. A. F. (2011a). Dynamic and scalable routing to perform efficient data aggregation in wireless sensor networks. In *IEEE International Conference on Communications*, page Accepted for publication.

Villas, L. A., Guidoni, D. L., Araujo, R. B., Boukerche, A., and Loureiro, A. A. F. (2011b). An energy-aware spatial correlation mechanism to perform efficient data collection in wsns. In *IEEE WLN Local Computer Networks*, page Accepted for publication.

Villas, L. A., Guidoni, D. L., Araujo, R. B., Boukerche, A., and Loureiro, A. A. F. (2011c). Explorando a correlacao espacial na coleta de dados em redes de sensores sem fio. In *Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*, pages 425--438.

Villas, L. A., Guidoni, D. L., Araujo, R. B., Boukerche, A., and Loureiro, A. A. F. (2011d). Time-space correlation for real-time, accurate, and energy-aware data reporting in wireless sensor networks. In *ACM international conference on modeling, analysis and simulation of wireless and mobile systems*, page Accepted for publication.

Wang, Z. (2001). *Internet QoS: Architectures and Mechanisms for Quality of Service*. Morgan Kaufmann Publishers.

Watts, D. J. (2004). *Six Degrees: The Science of a Connected Age*. W. W. Norton & Company.

Watts, D. J. (2007). A twenty-first century science. *Nature*, 445:489.

Watts, D. J., Dodds, P. S., and Newman, M. E. J. (2002). Identity and search in social networks. *Science*, 296:1302--1305.

Watts, D. J. and Strogatz, S. H. (1998). Collective dynamics of 'small-world' networks. *Nature*, 393:440--442.

Wu, Y., Stankovic, J. A., He, T., and Lin, S. (2008). Realistic and efficient multi-channel communications in wireless sensor networks. In *IEEE International Conference on Computer Communications*, pages 1193--1201.

Xi, F. and Liu, Z. (2009). Small world topology-aware geographic routing in wireless sensor networks. In *WRI International Conference on Communications and Mobile Computing*, pages 116--120.

Xia, F. (2008). Qos challenges and opportunities in wireless sensor/actuator networks. *Sensors*, 8:1099--1110.

Xu, K., Hassanein, H., Takahara, G., and Wang, Q. (2010). Relay node deployment strategies in heterogeneous wireless sensor networks. *IEEE Transactions on Mobile Computing,*, 9(2):145 –159.

Xuejun Liu, Jihong Guan, G. B. and Lu, H. (2010). Swer: small world-based efficient routing for wireless sensor networks with mobile sinks. *Frontiers of Computer Science in China*, 3(4):427--434.

Yarvis, M. D., Kushalnagar, N., Singh, H., Angarajan, A., Liu, Y., and Singh, S. (2005). Exploiting heterogeneity in sensor networks. In *IEEE International Conference on Computer and Communications*, pages 878--890.

Ye, W., Heidemann, J., and Estrin, D. (2002). An energy-efficient mac protocol for wireless sensor networks. In *IEEE International Conference on Computer Comunication*, pages 1567--1576.

Ye, W., Heidemann, J., and Estrin, D. (2004). An adaptive energy-efficient and low-latency mac for data gathering in wireless sensor networks. In *IEEE International Parallel and Distributed Processing Symposium*, pages 224--232.

Ye, X., Xu, L., and Lin, L. (2009). Small-world model based clustering division scheme in wireless sensor network. In *WRI International Conference on Communications and Mobile Computing*, pages 87--91.

Yingshu Li; Chunyu Ai; Chinh Vu; Yi Pan; Beyah, R. (2011). Delay-bounded and energy-efficient composite event monitoring in heterogeneous wireless sensor networks. *IEEE Transactions on Parallel and Distributed Systems*, 21(9):1373--1385.

Zhang, Z., Ma, M., and Yang, Y. (2008). Energy efficient multi-hop polling in clusters of two-layered heterogeneous sensor networks. *IEEE Transactions On Computers*, 57(2):231--245.

Zhou, G., Huang, C., Yan, T., He, T., Stankovic, J. A., and Abdelzaher, T. F. (2006). Mmsn: Multi-frequency media access control for wireless sensor networks. In *IEEE International Conference on Computer Communications*, pages 1--13.

Zhou, H., Taj, M., and Cavallaro, A. (2010). Target detection and tracking with heterogeneous sensors. *IEEE Journal of Selected Topics in Signal Processing*, 2(4):503--514.