N° d'ordre : 12/2014-D/INF

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université des Sciences et de la Technologie Houari Boumedienne

Faculté d'Electronique & Informatique

THESE

Présentée pour l'obtention du grade de DOCTORAT
en INFORMATIQUE
Spécialité : Programmation et Systèmes

Par

Mr Mohamed GUERROUMI

Thème

Data Management in Wireless Sensor Networks

Soutenue publiquement le 08 /05. /2014, devant le jury composé de:

Mr.	A. BELKHIR	Professeur, USTHB	Président
Mr.	N. BADACHE	Professeur, CERIST/USTHB	Directeur de thèse
Mr.	H. HAFFAF	Professeur, Univ Oran	Examinateur
Mr.	M. KOUDIL	Professeur, ESI	Examinateur
Mr.	M. BENCHAIBA	MCA, USTHB	Examinateur
Mme.	L. BOUALLOUCHE	MCA, Univ Béjaia	Examinatrice

Abstract

A wireless sensor network is typically composed of many tiny computers that feature a low frequency processor, some flash memory for storage, a radio for short-range wireless communication and limited energy source. Applications of wireless sensor networks have emerged in many domains ranging from environmental monitoring to structural monitoring as well as industry manufacturing. In all these applications, the primary task of a wireless sensor networks is to collect useful information by monitoring phenomena in the surrounding environment. Typically, in a wireless sensor network, sensor nodes generate data about a phenomenon and disseminate streams of data to a special device, namely a data sink, for analysis and processing.

Data management in wireless sensor networks has been an area of significant research in recent years. Many existing sensor data management systems view sensor data as a continuous stream that is sensed, filtered, processed, aggregated and disseminated from sensor node to users.

This thesis focuses on the efficient data extraction and dissemination in wireless sensor networks with energy awareness to extend the lifetime of the network. Energy is a critical factor as sensor nodes once deployed cannot be recharged. These sensor nodes majorly depend on batteries for energy, which get depleted at a faster rate because of the computation and communication operations they have to perform. Communication protocols can be designed to make efficient utilization of energy resources of a sensor node. As first part of our research, we start with studying and analyzing the main sources of energy consumption in wireless sensor networks. We present an overview of some well known existing solutions. We propose a set of algorithms to organize the Medium Access Control and minimize energy consumption. In this first part, we propose two MAC protocols both for asynchronous and synchronous communication mode. We evaluate the parameters of performance and we investigate the problems of energy consumption with extensive simulations.

We study in the second part, the problem of data dissemination in wireless sensor networks. We propose a mechanism to disseminate the sensed data form sensor node to the sink with minimum resource requirements. We extend this mechanism to support sink mobility, as in large sensor network, the communication between the sensors and the stationary sink can lead to high energy consumption, and consequently reduce the lifetime of the network. In our sink mobility scheme, the mobile sink selects its next destination based on the data dissemination frequent. We evaluate with extensive simulations and we compare the communication and the traffic cost of the proposed data dissemination with other approaches.

Keywords: Wireless sensor networks, Energy consumption, MAC layer, Data dissemination, Mobile sink.

Acknowledgements

First and foremost, I would like to express my gratitude to my supervisor, Professor Nadjib BADACHE, who gave me the opportunity to do this PhD research. He gave me the freedom to try out new ideas and gave me continuous support during the research.

I feel very lucky to have daily support and guidance from Professor Samira MOUSSAOUI. She is such a friendly and patient person. I have really learned a lot from her; how to present an idea properly and to construct precise statements.

I particularly appreciate Doctor Al-Sakib Khan Pathan from IIUM for his help and guidance. He provided refreshing insight, critical questions and valuable comments on my papers. I am grateful for his steady encouragement and infinite attention to details that ensured the quality of my work.

I would like to thank Professor Abdelkader BELKHIR, Professor Hafid HAFFAF, Professor Mouloud KOUDIL, Doctor Mahfoud BENCHAIBA and Doctor Louiza BOUALLOUCHE for agreeing to serve on my thesis committee.

I am also grateful to Professor Abdelmadjid BOUABDALLAH from UTC France and Doctor Mahfoud BENCHAIBA for giving their expertise and approved my PHD work.

Many thanks to my parents for their love, committed support, and prayers in all that I have done till now are the keys of all my achievements. Many thanks to my brothers and my sisters for their always encouragement. Many thanks to my wife for her patient and fully support.

Finally, my most heartfelt thanks...

Table of contents

List of Figures

List of Tables

Chapter 1

Introduction and Thesis Research Plan

1.1 Introduction

Recent advances in wireless communications and sensor technologies have emerged Wireless Sensor Networks (WSNs) as an interested research area with a great effect on practical application developments. They permit to monitor and observe an ambient environment with low economical cost. In inaccessible environments where human involvement may be too dangerous sensor networks may provide a robust service. These networks are widely used in commercial and industrial areas such as environmental monitoring, habitat monitoring, healthcare, process monitoring and surveillance.

WSNs may consist of many tiny sensor nodes of many different types such as seismic, magnetic, thermal, visual, infrared and radar, capable to monitor a wide variety of ambient conditions. These tiny sensor nodes are mainly composed of three components: sensing unit for data sensing, processing unit (processor, and memory) for data processing, and transceiver for wireless communication. All these components require a source of energy, which is usually a battery. For most wireless sensor network's applications, it is impractical to recharge or replace batteries. Future sensor nodes are envisioned as disposable, and it is important to use technologies that will maximize battery lifetime.

Sensor networks are mainly designed to disseminate data from sensor nodes to a data processing center for treatment and analysis. Moreover, minimize power consumption while ensuring that certain quality of service requirements such as delay, throughput, packet drop rate, and fairness is considered as primary network design goal. WSN has potential to design many new applications for handling emergency, military and disaster relief operations that requires real time information for efficient coordination and planning. Sensors are devices that produce a measurable response to a change in a physical condition like temperature, humidity, pressure …etc.

The issues of data management and network lifetime are extremely important in wireless sensor networks due to their deployment in hostile environment. The system should provide fault tolerant energy efficient, real-time communication as well as automatic and effective action in crises' situations.

All the research works therefore have a common concern of minimizing energy consumption and it is a significant issue at all layers of the wireless sensor networks. The major part of energy consumption, which is produced by Medium Access Control (MAC) layer, is related to the collision, the idle listening and the overhearing. Therefore, the main focus in this study concerns the network layer and MAC layer. Network protocols deal with how to disseminate data from a source to a sink efficiently. MAC layer provides efficient medium access control for the sensor networks with low energy consumption.

1.2 Communication model of wireless sensor networks

A WSN consists of a number of sensors spread across a geographical area. Each sensor has wireless communication capability and sufficient intelligence for signal processing and networking of the data. A wireless sensor network can be deployed in remote geographical locations and requires minimal setup and administration costs. Moreover, the integration of a wireless sensor network with a bigger network such as the Internet or a wireless infrastructure network increases the coverage area and potential application domain of the ad hoc network. Sensed information is relayed to a sink node by using multi hop communication [61]. The sink node is a sensor node with gateway functions to link to external networks such as the Internet and sensed information is normally distributed via the sink node.

Figure1.1 shows typical communication model of wireless sensor networks. Each sensor node bases its decisions on its mission, the information it currently has, and its knowledge of its computing, communication, and energy resources. Each of these scattered sensor nodes has the capability to collect and disseminate data either to the sink. A sink may be a fixed node or a mobile node capable of connecting the sensor network to an existing communications infrastructure or to the Internet where a user can have access to the reported data.

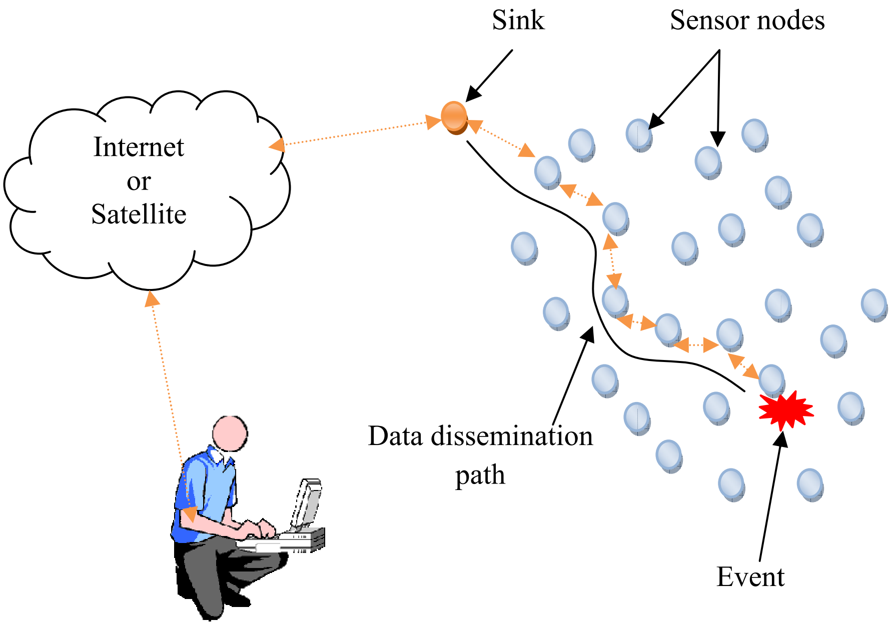


Figure 1.1: Communication model of wireless sensor networks.

Instead of sending the raw data to the nodes responsible for the fusion, sensor nodes use their processing abilities to locally carry out simple computations and transmit only the required and partially processed data. The sensor nodes not only collect useful information such as sound, temperature, light …etc, they also play a role of the router by communicating through wireless channels under battery-constraints [214]. Sensor network nodes are limited with respect to energy supply, restricted computational capacity and communication bandwidth. The ideal wireless sensor is networked and scaleable, fault tolerance, consume very little power, smart and software programmable, efficient, capable of fast data acquisition, reliable and accurate over long term, cost little to purchase and required no real maintenance.

WSNs differ from MANETs (Mobile Adhoc NETworks) in many fundamental ways. Viewing a wireless sensor network as a large-scale multi-hop ad hoc network may not be appropriate for many real-world applications. The communication overhead for configuring the network into an operational state is too large. The number of nodes in a wireless sensor network, which can be of several orders of magnitude higher than the nodes in an ad hoc network and sensor nodes that are prone to failure, are densely deployed. Sensor nodes mainly use broadcast, while most MANETs are based on the Peer-to-Peer (P2P) communication paradigm. Information exchange between end-to-end nodes will be rare in wireless sensor networks. They are limited in power, computational capacity and memory, and may not have global IDs. Wireless sensor networks have a wide range of applications ranging from monitoring environments, sensitive installations, to remote data collection and analysis. In wireless sensor networks the nodes act both as hosts and as routers. They operate in a self organizing and adapting manner.

1.3 Usage of wireless sensor networks

Wireless sensor network provides an intelligent platform to gather and analyze data without human intervention. As a result, sensor networks are applied in a wide range of areas, such as military applications, public safety, medical, surveillances, environmental monitoring, commercial applications, habitat and tracking. In general, sensor networks will be ubiquitous in the near future, since they support new opportunities for the interaction between humans and their physical world. In addition, sensor networks are expected to contribute significantly to pervasive computing and space exploration in the next decade. Deploying sensor nodes in an unattended environment will give much more possibilities for the exploration of new applications in the real world. Consequently, wireless sensor networks have emerged as a promising technology with various applications:

- **Environmental observations:** Wireless sensor networks can be used to monitor environmental changes. An example is water pollution detection in a lake that is located near a factory that uses chemical substances. Sensor nodes are randomly deployed in unknown and hostile areas and relay the exact origin of a pollutant to a centralized authority which takes appropriate measures to limit the spreading of pollution. Other examples include forest fire detection, air pollution [161], and irrigation management in agriculture and landscaping by monitoring soil moisture [28].

- **Building monitoring:** Wireless sensor networks can also be used in large buildings or factories to monitor climate changes. Thermostats and temperature sensor nodes are deployed all over the building's area. In addition, sensors can be used to monitor vibration that can damage the structure of a building [167].

- **Health care:** Sensors can be used in biomedical applications to improve the quality of provided care. Sensors are implanted in the human body to monitor medical problems such as cancer and help patients maintain their health. Examples of this class include personal health care monitoring with body area networks [15], and tracking doctors and patients inside hospitals [118]. Since the sensors in health care application are physically grouped to form a body area network, it has a different mobility model, called group mobility [156].

- **Military monitoring:** The military uses sensor networks for battlefield surveillance [164]. Sensors can monitor vehicular traffic, track the position of the enemy [64], or even safeguard the equipment of side deploying sensors.

- **Animal monitoring:** Typical applications of animal tracking are monitoring animals for studying their behaviors, and locating or confining them in pastoral regions or in the wild. Monitoring of a typical farm environment [127], in particular cattle monitoring, and wildlife tracking [74] in the ecological system are the major examples of such applications. Different from the environmental monitoring applications, animal monitoring introduces mobility within the WSNs.

1.4 Thesis research plan

Data dissemination and energy efficiency are the main purpose of our research. Wireless sensor nodes, which are battery powered, are used for detecting and collecting information from the areas where there is very little scope for manual handling to recharge or change batteries. These sensing nodes collect and send data towards the sink for further actions. For a better functioning and a longer lifetime for a sensing node within the network, energy consumption has to be considered as a major factor of concern.

Many contributions have been done in this field restricting their work towards finding out suitable data dissemination protocols that are used for a specific surveillance application. Sensor nodes in this kind of application are used to detect and collect information regarding any object that is moving or any triggered event. A simple way to disseminate this information is to let the network carrying it using an ordinary protocol stack which carries out the general process of transmission without any concerns for energy efficiency factor.

1.4.1 Data dissemination in wireless sensor networks

Data dissemination is the process by which queries or data are routed in the sensor network. The data collected by sensor nodes have to communicate to the sink or to any other interested node. A data dissemination protocol is required to provide effective data transmission from sensor nodes to the sink. Data dissemination protocols have a certain relation to the routing protocols. The routing protocols are general and are designed to find a path between the source and destination nodes. On the other hand, data dissemination protocols should guarantee successful transmission from nodes to the sink. Data dissemination in WSN depends on specific needs of the application and also on time sensitivity of the data collected [70]. As explained below, data dissemination models can be categorized as periodic-based, query-based or event-based models [178] [209].

- **Time-based:** This method is required for applications that need periodic data monitoring. In this model, to save energy, sensors can be sleeping or turned-off most of the time and periodically they wake up, switch on their sensors, sense the environment and transmit the sensed data to the sink in periodic intervals. Periodic data do not need to be transferred reliably since it has generally the same content as the previous reading.
- **Query-based:** This method is often initiated by the sink, by sending out a query to sensor nodes, asking them to send data which have the properties specified in the queries. It is a typical way of extracting data from a sensor. In this model, sensors only transmit data when it is explicitly requested by the sink. The sink may also send a query for some other purposes such as to specify or change the operation mode of a group of sensors or to update the system software running on the sensor nodes.
- **Event-based :** In this approach, whenever an event of interest occurs in the environment, the sensor nodes have to report and disseminate the data associated with that event to the sink. Usually events are rare. However, when an event occurs, a burst of packet is generated that needs to be delivered to the sink as quick and reliable as possible.

- **Hybrid-based:** For different applications, a combination of different data reporting models is also possible. In networks, where different data dissemination models coexist, the data dissemination protocol should change its operation depending on the importance of the sensed data.

Data dissemination protocols need to indicate whether the data are to be transmitted in broadcast or unicast mode. Routing protocols and other techniques, such as data aggregation, may also be used for performance optimization [21].

1.4.2 Energy saving in wireless sensor network

These wireless sensor networks have severe resource constrains and energy conservation is very essential. The sensor node's radio in the wireless sensor networks consumes a significant amount of energy. Substantial research has been done on the design of low power electronic devices in order to reduce energy consumption of these sensor nodes. Because of hardware limitations further energy efficiency can be achieved through the design of energy efficient communication protocols. Major sources of energy waste in wireless sensor network are basically of four types [183].

- **Collision:** The first one is the collision. When a transmitted packet is corrupted due to interference, it has to be discarded and the follow on retransmissions increases energy consumption. Collision increases also latency.
- **Overhearing:** The second is overhearing, meaning that a node picks up packets that are destined to other nodes.
- **Packet Overhead:** The third source is control packet overhead. Sending and receiving control packets consumes energy too and less useful data packets can be transmitted.
- **Idle listening:** The last major source of inefficiency is idle listening i.e., listening to receive possible traffic that is not sent. This is especially true in many sensor network applications. If nothing is sensed, the sensor node will be in idle state for most of the time.

The main goal of any MAC protocol for sensor network is to minimize the energy waste due to collision, idle listening, overhearing and network overhead.

1.5 Research motivation and challenges

The major sources of energy consumption sited above are located at the level of MAC layer. Therefore, designing data dissemination protocol with efficient energy still insufficient while the main sources of energy dissipation remain existing. Consequently, our researches are divided into two parts and focused on the network and MAC layers.

The figure 1.2 shows the relationship between the sources of energy consumption and the wireless sensor network layers.

The data link layer is responsible for multiplexing data streams, data frame detection, medium access, and error control. It ensures reliable point-to-point and point-to-multipoint connections in a communication network. MAC protocol plays an important role in energy saving, throughput, QoS and minimum delay. Moreover, in a wireless multi-hop self-organizing sensor network, The MAC protocol should achieve two goals.

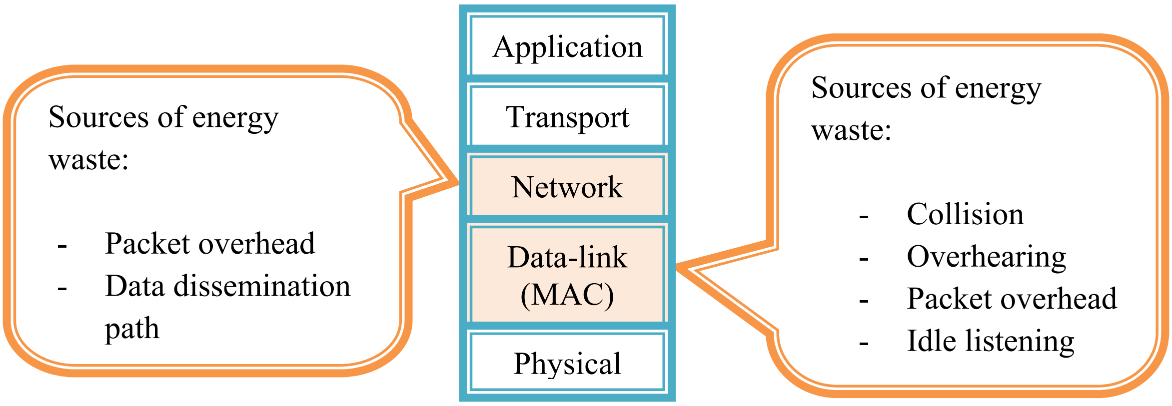


Figure 1.2: Wireless sensor network layers and energy consumption.

The first is the creation of the network infrastructure. Since thousands of sensor nodes are densely scattered in a sensor field, the protocol MAC must establish communication links for data transfer. This forms the basic infrastructure needed for wireless communication hop to hop and gives the sensor network self-organizing ability.

The second objective is to fairly and efficiently share communication resources between sensor nodes. Since the environment is noisy and sensor nodes can be mobile, the MAC protocol must be power-aware and be able to minimize collision, overhearing and idle listening.

Regardless of which type of MAC scheme is used for sensor networks, it certainly must have built-in power-saving mechanisms and strategies for proper management of node mobility or failure. The most obvious means of power conservation is to turn the transceiver off when it is not required. Although this power-saving method seemingly provides significant energy gains, an important point that must not be overlooked is that sensor nodes communicate by using short data packets.

The main features of sensor MAC are periodic listen and sleep, collision and overhearing avoidance, and message passing. The duration of the sleep and awake cycles are application dependent and they are set the same for all nodes.

The network layer takes care of routing the data supplied by the transport layer. The network layer of a sensor network is usually designed according to the following principles:

- ✓ Power efficiency is always an important consideration.

- ✓ Sensor networks are mostly data-centric.
- ✓ Data aggregation is useful only when it does not hinder the collaborative effort of the sensor nodes.

An ideal sensor network has attribute-based addressing and location awareness. Energy-efficient routes can be found based on the available power in the nodes or energy required for transmission in the links along the routes. An efficient energy route is selected by one of the following approaches:

- ✓ The maximum power available route is the route that has maximum end to end energy.
- ✓ The minimum energy route is the route that consumes minimum energy to transmit the data packets between the sink and sensor nodes.
- ✓ The minimum hop route is the route that makes the minimum hop to reach the sink.

The minimum power available node route is the route along which the minimum power available is larger than the minimum power available of the other routes.

Data aggregation combines data from many sensor nodes into a more compact form before forwarding to a location for processing. Data aggregation is needed to handle the large amount of data generated in sensor networks.

1.6 Thesis contribution

The ultimate goal of a wireless sensor network is to detect specified events of interest in a sensor field and to deliver them to end user.

The consumed electric power depends substantially on how the sensed data is handled and communicated. Because the capacity of the battery of the sensor node is very limited, it is necessary to consider the extent to which the demands of applications can be met. End-to-end data transfer schemes that fit the characteristics of wireless sensor networks are needed and power aware data communication protocols are actively being researched. In this context and with regard to the previously mentioned research plan and focus we describe the main contributions of the thesis:

- ➢ ***First part: MAC layer and energy efficiency***

  In this part we focus on reducing the transceiver’s energy consumption because the transceiver often uses more power than any other hardware resource. Thus we work on the MAC layer and we attempt to limit transceiver energy consumption by preventing or reducing collisions, overhearing, idle listening, and overhead. Collisions within sensor networks cause the same problems as other wireless networks: performance limitation and energy waste. While many sensor network applications can cope with a slight performance decrease because they have low data rate requirements and high delay tolerances, energy waste due to frequent collisions can significantly decrease a sensor node’s lifetime. Retransmitting a message requires the sensor node to operate its transceiver at the highest power levels and consumes multiple times the minimum energy required for that message. For wireless sensor networks that do not require a reliable link layer, and thus do not retransmit messages, collisions have a smaller impact, but the loss of data may decrease the application’s accuracy. Several sensor nodes may receive the same transmission, possibly multiple times with retransmissions, even though the source intended it for only one recipient. In these cases the unintended receivers overhear the message and waste energy on reception and processing.

MAC protocols may limit, but cannot prevent overhearing from occurring in some fashion. Fortunately, MAC protocols can leverage overhearing to infer information about the wireless channel, such as sensor node availability or link status, and decrease the effective energy loss. A MAC protocol may also end a reception early and enter the sleep state to limit the energy losses associated with overhearing messages once it determines the message which belongs to another node. Energy waste also occurs when no sensor node transmits a message, but nearby sensor nodes attempt to receive a message. In this case the receiving sensor nodes perform idle listening and waste the energy consumed by the transceiver during this time. Reception does not consume as much energy as transmission in most designs, but it does consume more power many times than if the sensor node placed the transceiver in the sleep state. Idle listening can account for a significant portion of the energy a sensor node consumes in some cases.

A typical solution to limit idle listening uses a timer to end reception if the sensor node does not detect any activity on the channel. Note that idle listening does not include carrier sensing, which many MAC protocols require for proper operation. So to improve energy consumption, we provide the following contributions:

- **Contribution1: MAC protocol with synchronous mode**

  MAC protocols may use synchronization messages to organize sensor nodes together or allow sensor nodes to estimate distances based on the received signal strength. Synchronous mode may also use to schedule the neighboring sensor nodes around common active/sleep program. We review the state of the art synchronous methods in wireless sensor networks. We present a mechanism for organizing the sensor nodes under pair and odd clusters that are active in load balancing mode which permit to avoid collision and decrease energy waste. We evaluate the performance’s parameters and we investigate the problems of collision and latency with extensive simulations. This work appeared in the following paper [109]:

Clustering Pair and Odd to Optimize the Energy Consumption in Wireless Sensor Networks, with N. Badache and S. Moussaoui, in Communications of SIWN, Vol. 6, pp. 127-131, April 2009.
http://fatech.org.uk/press/tai/cosiwn0006.htm

- **Contribution2: MAC protocol with asynchronous mode**

  Synchronous MAC protocols provide the capability to reduce collisions and message retransmissions. These kinds of protocols are more required for many applications. Unfortunately, these protocols consume more resources to share information and maintain common state. They pose big problem for the scalability and real time applications. Therefore, we propose MAC protocol in which sensor node schedules its own active/sleep program and adapts its listening according to the medium activity. We define new mechanism based on the two neighboring node information to avoid collision and the hidden host problem. We evaluate the performance of our protocol with extensive simulations and we compare its performance with another well known MAC protocols. The work appeared in the following paper [110]:

  MAC Protocol with Low Energy Consumption in Wireless Sensors Network , with N.Badache, S. Moussaoui in Second International Conference on Advances in Future Internet, Venice, Italy, ISBN: 978-0-7695-4091-7.18-25July2010.
  http://doi.ieeecomputersociety.org/10.1109/AFIN.2010.14

➢ ***Second part: Network layer and an efficient energy data dissemination***

Data management might vary depending on the sensor network architecture and the application. A daunting challenge in the design of a reliable wireless sensor network is to augment its lifetime in terms of energy and information efficiency. After enhancing the energy consumption in the layer consuming the major part of this resource by proposing the above solutions. Our purpose in this part is to design an energy-aware data dissemination solution. An optimal solution has to:

- Minimize energy requirements at each node to transfer individual packets and,
- Maximize the operational lifetime of scalable networks.

It is primarily important to save energy of the sensor nodes while disseminating the sensed data to the sink node. This may either be accomplished by involving as minimum as possible the number of nodes participating in the data dissemination process and selecting the best data dissemination path. In this purpose we present a detailed study of the existing data dissemination protocols in [115] and we offer the following contributions:

- **Contribution1: Data dissemination with static sink**

  To enable sensor nodes to efficiently disseminate their sensed data with minimum energy requirement, we exploit the advantages of clustering technique such as more scalability, less load, less energy consumption and more robustness. We propose set of algorithms where sensor nodes are divided into clusters with same assignment level. The cluster head is responsible for data aggregation, information dissemination and network management. We propose new cluster head management scheme based on a dynamic energy threshold. With this approach, we aim at decreasing energy consumption whereas meeting the requirements such as high data delivery ratio and low delivery delay. The performance of our protocol has been evaluated with extensive simulations and compared with another protocols. The work related to this contribution appeared in the following papers [113], [114]:

  1. *Clustering data dissemination in wireless sensors network* , with N. Badache, S. Moussaoui in International Conference on Communications, Computing and Control Applications (CCCA), 2011 , P 1 - 6, ISBN: 978-1-4244-9795-9.
     http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=6031226

  2. *Data Dissemination and Power Management in Wireless Sensor Networks*, with N. Badache, S. Moussaoui in Advances in Computing and Communications, Communications in Computer and Information Science, 2011, Volume 193, Part 6, pp 593-607, DOI: 10.1007/978-3-642-22726-4_61.
     http://www.springerlink.com/content/w4k2021372877730/.

- **Contribution2: Data dissemination with mobile sink**

The concept of a mobile base station is probably the most intuitive solution to the hot spot problem. Since hot spots arise in the vicinity of the sink, making this node mobile distributes the problem across the network. To achieve reliable data dissemination of events with efficient energy, we extend our previous work [114] by handling efficient sink mobility. In our mobility approach, the sink node moves periodically and randomly toward a chosen destination. The sink selects its destination based on the sensed data frequency of each cluster recorded during the last period. We compare the performance of our protocol with other approaches using simulation. This work appeared in the following paper [112], [111]:

1. *Sink Mobile for Efficient Data Dissemination in Wireless Sensor Networks*, with N. Badache, S. Moussaoui in Communications in Computer and Information Science, 2012, Volume 253, Part 1, pp 635-645, ISBN 978-3-642-30506-1.
   http://www.springerlink.com/content/p2v21718l2403735/
2. *Mobile Sink and Power Management for Efficient Data Dissemination in Wireless Sensor Networks,* with N. Badache, S. Moussaoui in International journal of Telecommunication Systems, accepted Jun 2013.
   http://link.springer.com/journal/11235

1.7 Thesis outline

Chapter 1 has presented an overview about wireless sensor networks communication model, their main applications, the scope of this thesis, our research plan and the main contributions. The remainder of this thesis is organized into seven additional chapters. This thesis begins by providing review of the important existing works related to our research scope. In subsequent chapters the various aspects pertaining to our research plan are discussed.

- ✓ Chapter 2 presents detailed review of some existing MAC protocols and the comparison of their characteristics. This chapter classifies these protocols according to the synchronization parameter into three classes, synchronous, asynchronous and hybrid MAC protocol.
- ✓ Chapter 3 provides an overview of data dissemination protocols in wireless sensor networks. It describes their design and classifies them according to number of sink and its nature, stationary or mobile.
- ✓ Chapter 4 describes and evaluates our first contribution, namely RSMAC protocol and compares its performance with two other MAC protocols.
- ✓ Chapter 5 presents our second contribution and evaluates the performance of the proposed asynchronous MAC protocol by comparing with other well known protocols.
- ✓ Chapter 6 begins the contributions of the second part of our research. This chapter describes and simulates our data dissemination protocol namely DDPM designed for static sink wireless sensor networks.
- ✓ Chapter 7 describes the mobility concept in wireless sensor networks and presents an extended version of DDPM by handling sink mobility. The performances of this new protocol, namely EDDP are evaluated using extensive simulations.

We conclude this thesis in Chapter 8 summarizing the key results and highlighting the possible future research directions for the problems and solutions presented in the thesis.

Chapter 2

Medium Access Control in Wireless Sensor Networks

2.1 Introduction

In order to prolong lifetime of wireless sensor networks, several solutions of energy conservation have been proposed. However, the choice of energy conservation protocol still difficult and very related to the nature and the goal of the sensor networks application. The principal sources of energy consumption, as presented in the previous chapter, are Collision, Control packet overhead, idle listening and the overhearing. All these paramount parameters are related directly to the operating mode of MAC protocols, which pushed us to study the various protocols proposed for this layer.

Designing power efficient MAC protocol is one of the ways to prolong the lifetime of the network. To find out the advancements, achievements, challenges, and issues in this topic, here, we present a study of the energy efficient MAC protocols for wireless sensor network. We present the basic concepts, the operating modes, and the characteristics of each protocol by scrutinizing the strong and the weak points of each one of them.

The rest of the chapter is organized as follows: Section 2.2 presents the different functionalities provided by sensor MAC protocols, the parameters that have to be considered to design a good MAC protocol and shows some common metrics that need to be considered to evaluate its performances. Section 2.3 discusses the related surveys presented in this area. Based on the need of synchronization between neighboring nodes, this Section presents a taxonomy that is used to categorize the existing sensor MAC protocols. Sections 2.4, 2.5 and 2.6 present synchronous, asynchronous and hybrid sensor MAC protocols, respectively. The main points of medium access scheme of all the reviewed protocols are then summarized in Section 2.7; this section discusses and compares these protocols based on the evaluation metrics presented in Section 2.2. Finally, Section 2.8 draws the conclusions with some open research directions.

2.2 Mac protocol functionality, design and metrics

At the end of network deployment, communication links between sensor nodes have to be established. Moreover, communication medium needs to be shared fairly and efficiently. These main points constitute the objectives that any medium access protocol has to achieve.

2.2.1 Mac protocol functionality

Depending on the network requirements and device capability, MAC protocol provides different functionalities. As discussed in [91] and [71], these functions can be noted as below:

- Control medium access by determining the winner of the medium at any time. Medium access represents the main function of wireless MAC protocols since broadcasts easily cause data corruption through collisions.
- Define the frame format, the time frame, and perform data encapsulation and decapsulation for communications between devices.
- Ensure successful and reliable transmission between devices using acknowledgement (ACK) messages and retransmissions when necessary.
- Prevent frame loss through overloaded recipient buffers.
- Use error detection or error correction codes to control the amount of errors present in frames delivered to upper layers.

2.2.2 Mac protocol design

In traditional wireless ad hoc network, MAC protocols attempt to provide high throughput, low latency, fairness, and mobility management, but often have little or no consideration for energy conservation. In wireless sensor networks, where sensor nodes are characterized by their limited resources, multi-hop operation mode, and different application requirements, MAC protocols however, must provide the best performance at the smallest amount of energy consumption due to the limited energy resources available to each sensor node. Nevertheless, energy efficiency and throughput are the major aspects that need to be considered in MAC protocol design for wireless networks. According to [158], [198], for designing a good MAC protocol for these networks, the following parameters have to be considered:

- **Energy Efficiency:** sensor nodes are battery powered and it is often very difficult to change or recharge batteries for these sensor nodes. Sometimes it is beneficial to replace the sensor node rather than recharging them.
- **Latency:** this parameter basically depends on the application requirements. In some sensor network applications, the detected events must be reported to the sink node in real time so that the appropriate action could be taken immediately.
- **Throughput:** depends on the application requirements. Some sensor network applications require sampling the information with fine temporal resolution. In such sensor applications it is better that sink node receives more data.
- **Fairness:** related to the limited bandwidth, it is necessary to ensure that the sink node receives information from all sensor nodes fairly.

2.2.3 Mac protocol metrics

To evaluate the performance of MAC protocols, the research community considers some common metrics that need to be considered [158]. However, each protocol has some other specific metrics related to its design that also need to be evaluated. The common metrics are:

- **Energy consumption per bit (joules/bit):** can be defined as the total energy consumed divided per the total bits transmitted. Energy consumption is affected by all the major sources of energy waste in wireless sensor network such as idle listening, collisions, control packet overhead and overhearing.
- **Average delivery ratio:** is the number of packets received by the sink to the number of packets sensed by each node and sent over the network towards the sink node.
- **Average Packet Latency:** is the average time taken by the packets to reach the sink node.
- **Network Throughput:** is defined as the total number of packets delivered at the sink node per time unit.

2.2.4 Medium access methods

In wireless sensor networks, controlling access to the channel, generally known as multiple access control, plays a key role in determining channel capacity utilization, network delays and more important, power consumption. It also influences congestion and fairness in channel usage. CSMA (Carrier Sense Multiple Access) and TDMA (Time-Division Multiple Access) are the most controlling channel access methods in wireless sensor networks.

- CSMA (Carrier Sense Multiple Access) is the simplest form of medium access control in which nodes can transmit at any time as long as there is no contention [98]. CSMA can be non-persistent or *p*-persistent. In non-persistent CSMA, a wireless channel has to sample before any data transmission to determine if another device has already started transmitting. If the channel is busy, a backoff operation has to perform before attempting to transmit again. When the channel is free, sensor node transmits its data immediately. In *p*-persistent CSMA, sensor node continues to sense the channel when the channel is busy instead of delaying and checking again later. When the channel becomes free, sensor node transmits its data with probability *p* and delays the transmission with probability (1-*p*). An extended version of CSMA, called CSMA with collision avoidance (CSMA/CA) attempts to avoid collisions by using a control message exchange to reserve the wireless channel before each data message transmission using the RTS/CTS (Request to

Send / Clear to Send) mechanism. This method is usually more used. It does not require clock synchronization and global topology information. Dynamic node joining and leaving are handled gracefully without extra operations. However, RTS/CTS mechanism incurs high overhead of the channel capacity in sensor networks [18], [79] because, data packets are typically very small in sensor networks.

- TDMA (Time-Division Multiple Access) is a common scheduling method which schedules transmission times of neighboring nodes to occur at different times. Each sensor node transmits data during its own time slot [214]. Thus, it can solve the hidden terminal problem without extra message overhead. However, TDMA has many disadvantages [199] like clock synchronization and scalability problem.

2.3 Sensor MAC protocols classification

MAC protocols for wireless sensor networks can be classified into several categories based on the medium access mechanism. In [20], two classes have been provided: contention-based protocols and schedule-based protocols. In [91], the authors classify the MAC protocols with the same manner as was presented in [20] and also they provide one more sub-class under the two broad categories. Based on how neighboring nodes organize access to the shared medium, the MAC protocols are classified in [92] into random access, slotted access, frame-based, and hybrid protocols. Another classification is given in [158], where the authors provide a thematic taxonomy and classify MAC protocols according to the problems dealt with: scheduled protocols, protocols with a common active period, preamble sampling protocols, and hybrid protocols. In [212], the authors broadly classify the MAC protocols for wireless sensor networks into contention-based protocols, contention-free (scheduled-based) protocols, hybrid protocols and preamble sampling protocols.

Another comprehensive state-of-the-art study of WSN MAC protocols is provided in [1]. In this study, the authors provide a thematic taxonomy in which sensor MAC protocols are classified according to the dealt problems. The studied protocols are classified into three categories: scheduled protocols, protocols with common active periods, and hybrid protocols. The survey presented in [149] explores the extent to which existing MAC protocols for WSNs can serve for mission-critical applications. The analyzed protocols are classified according to data transport performance and suitability for mission-critical applications. Therefore, the following two main categories are used: delay-aware with four sub-classes (node-to-node decrease, node-to-node guarantee, end-to-end decrease and end-to-end guarantee) and reliability-aware with node-to-node increase, node-to-node guarantee, end-to-end increase and end-to-end guarantee. In [142], the authors detail the evolution of WSN MAC protocols with four categories: asynchronous, synchronous, frame-slotted, and multichannel. These protocols have been evaluated in terms of energy efficiency, data delivery performance, and overhead needed to maintain a protocol's mechanisms. MAC strategies for cognitive radio networks have also been surveyed in [3]. This survey shows the fundamental role of the MAC layer and identifies its functionalities in a cognitive radio network. Classification of the cognitive MAC protocols is proposed with two main categories: Direct Access Based (DAB) and Dynamic Spectrum Allocation (DSA). This work also discusses the advantages, drawbacks, and further design challenges of cognitive MAC protocols.

In [106], the authors focused their study on timeliness issues of slotted contention-based MAC protocols and provide a comprehensive review and taxonomy of synchronous MAC protocols. Based on the delay efficiency, the authors classify these protocols into two main categories: static schedule and adaptive schedule. Dealing with mobility can pose many challenges in protocol design; especially, at the MAC layer. These barriers require mobility adaptation algorithms to localize mobile nodes and predict the quality of link that can be established with them. In this context, the authors in [150] survey the current state-of-art in handling mobility. They describe the existing mobility models and patterns; and analyze the challenges caused by mobility at the MAC layer. In [60], the authors outline the sensor network properties that are crucial for the design of MAC layer protocols and study some MAC protocols without giving any classification.

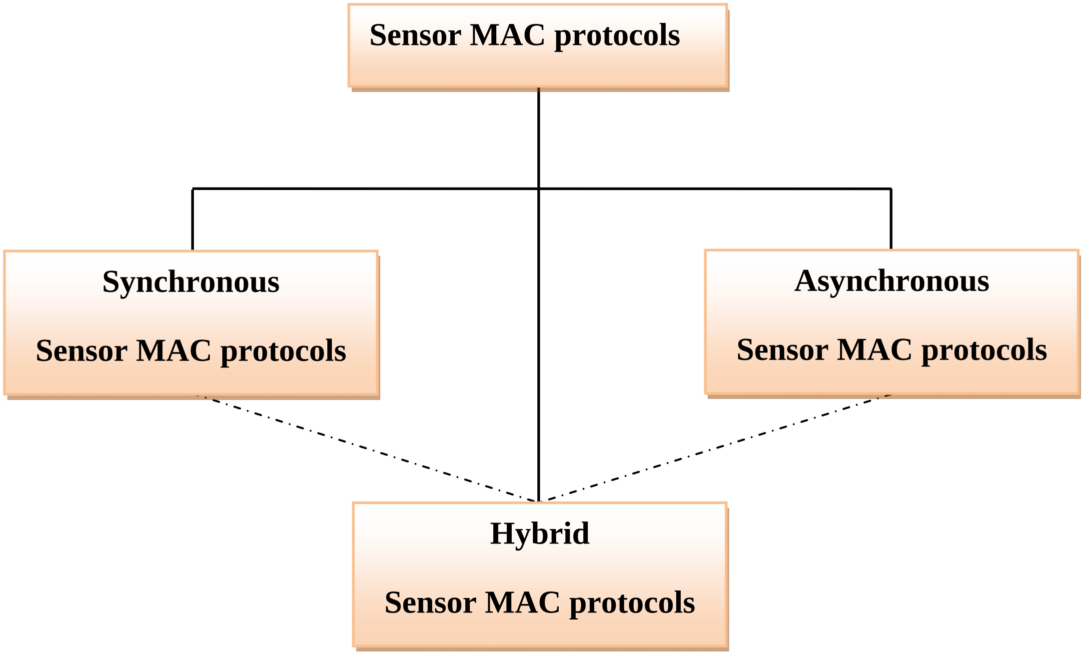


Figure 2.1: Sensor MAC protocols classification.

Now, after having some background, here we present a discussion on several representative MAC protocols proposed in the previous works. As shown in Figure 2.1, three general classes for sensor network MAC protocols can exist. These classes are principally based on one key parameter. Some MAC protocols require that sensor nodes have to be synchronized to perform their functions. However, some other protocols do not need this requirement and let sensor nodes have medium access without synchronization. Moreover, other MAC protocols allow sensor nodes to switch between these two previous modes according to the traffic behavior and the sent packet type.

Synchronous MAC protocols attempt to organize nearby sensor nodes so their communications occur in an ordered fashion. The most common synchronization method organizes sensor nodes using time TDMA, where a single sensor node utilizes a time slot. Organizing sensor nodes provides the capability to reduce collisions and message retransmissions at the cost of synchronization and state distribution.

Asynchronous protocols attempt to conserve energy by allowing sensor nodes to operate independently with a minimum of complexity and without clock synchronization. While collisions and idle listening may occur and cause energy loss, these kinds of MAC protocols typically do not share information or maintain state.

The hybrid MAC protocols combine the two previous classes by allowing sensor nodes to use both synchronous and asynchronous mode.

2.4 Synchronous sensor MAC protocols

Synchronous MAC protocols attempt to reduce energy consumption by coordinating sensor nodes with a common program. This can be done by establishing transmission schedules statically or dynamically to allow nodes to transmit data packets without collisions [190]. Most of the synchronous sensor MAC protocols use some form of TDMA because the other forms of multiple access, such as frequency or code division, would increase the cost and power requirements of the sensor nodes [206].

By using a common program, the MAC protocol specifies which sensor nodes should utilize the channel at any time and thus, limits or eliminates collisions, idle listening, and overhearing. Nodes not participating in communication with its neighbors may enter in the sleep mode until they have a message to transmit or to receive and thus, can optimize energy consumption.

In the existing literature, several synchronous MAC protocols for WSNs have been proposed like [95], [198], [183], [69], [190], [203], [148], [98], [160], [30], [29], [162], [205], [181], [189], [184]. Some of these solutions have been broadly surveyed in [91] and are divided into four subclasses, Priority-based, Traffic-based, Clustering-based, and Slotted TDMA. Synchronous sensor MAC protocols have one common aspect. Before any data packet transmission, the peer neighboring sender and receiver nodes have to be synchronized. Based on this aspect, we regard to these protocols as only one category and we survey in this section, the most significant of these protocols.

2.4.1 Channel Access protocols

In [95], the authors proposed three protocols using the priority of nodes or links calculated from a random function to permit the channel access. The random function uses sensor node IDs and time slot numbers as input values to establish the priority within a two-hop neighborhood. The sensor nodes share their neighbor information and each sensor node maintains information about its two-hop neighborhood.

The first protocol proposed is called Node Activation Multiple Access (NAMA). NAMA uses distributed time division, time is divided into blocks of S_b sections and each section is divided into P_S parts. The parts contain T_P time slots. A node i chooses only one part p_i, during which to contend for a time slot to transmit data packets. The choice of a part is dependent on the density of neighbors already using that part, usually decided when the node joins a network. In this protocol, the last section of each block is reserved for signaling messages that allow sensor nodes to join the network. Each sensor node calculates its priority, compares it with the priority of its neighbors and determines who has access to the current time slot within the sensor node's chosen part. If a sensor node has the highest priority among its two hop neighbors for the given time slot, then the sensor node may transmit.

The second protocol called Link Activation Multiple Access (LAMA) is a time-slotted code division medium access scheme using Direct Sequence Spread Spectrum (DSSS) [72], [119] code assigned to the receiver and the priority of the transmitter. Each sensor node gets a code assigned from a finite set of pseudo-noise codes. During each time slot, the sensor node with the highest priority in a two-hop neighborhood, calculated based on sensor node ID as in NAMA protocol, may activate a link by using the code assigned to the receiver. Using orthogonal codes allows sensor nodes to communicate when they would normally interfere and using the neighborhood information prevents collisions at the receiver.

The third protocol called Pairwise-link Activation Multiple Access (PAMA) is also a time-slotted link activation protocol based on a code division multiplexing scheme using DSSS code [72], [119]. The links between sensor nodes can be activated by assigning priorities to the links and by varying the codes and priorities of links based on the current time slot. A communication link between two sensor nodes can be established if the link between the source (s) and the destination (d) node has the highest priority among all links of nodes s and d, and node source has the highest priority of its two-hop neighbors using the code assigned to link (s, d). Using DSSS allows nodes to communicate on different codes without interruption and the protocol algorithm prevents collisions on the same code by using the neighborhood information.

The main advantage of these protocols is the collision avoidance and the sensor nodes need only local information for channel access decision. But, the major drawback of these protocols is in the resources required. All the protocols require a sensor node to calculate the priorities of each neighboring sensor node and for each time slot; which consume more energy resources and decrease the network lifetime. Also, in LAMA and PAMA, the sensor nodes need to have radios with spread spectrum capabilities, which increases sensor node cost.

2.4.2 Sensor MAC protocols (S-MAC)

This protocol is specifically designed for wireless sensor networks. The protocol S-MAC [198], [199] aims to reduce energy consumption, while supporting good scalability and collision avoidance. S-MAC tries to reduce energy consumption from all the sources that cause energy waste, like idle listening, collision, overhearing and control overhead. S-MAC consists of three major components: periodic listen and sleep, collision and overhearing avoidance, and message passing.

The basic scheme of S-MAC is shown in Figure 2.2. Each node goes to *sleep* for some time, and then wakes up and listens to find if any other node wants to talk to it. During sleep, the node turns off its radio, and sets a timer to awake itself later. The listening and sleeping time duration can be selected according to different application scenarios. For simplicity, S-MAC uses the same values for all the nodes.

Before starting its periodic listen and sleep, the sensor node has to choose a schedule and exchange it with its neighbors. First, the sensor node listens for a certain amount of time. If it does not hear a schedule from another node, it randomly chooses a time to go to sleep and immediately broadcasts its schedule in a SYNC message, indicating that it will go to sleep after *t* seconds.

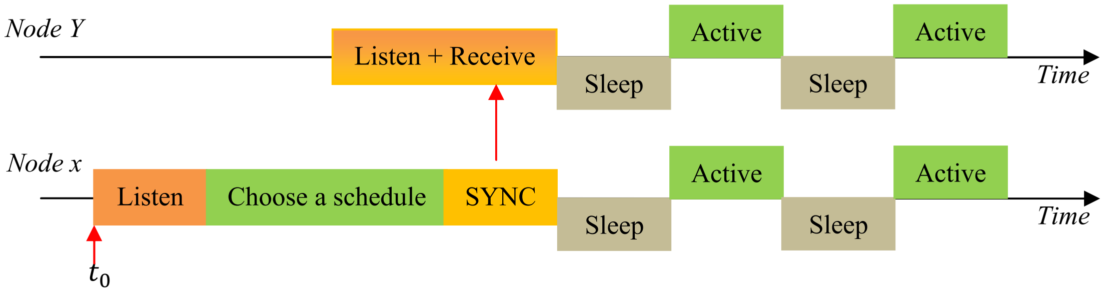


Figure 2.2: S-MAC basic scheme.

If the node receives a schedule from a neighbor before choosing its own schedule, it follows that schedule by setting its schedule to be the same. It then waits for a random delay t_d and rebroadcasts this schedule, indicating that it will sleep in $(t - t_d)$ seconds. Neighboring nodes form virtual clusters to set up a common sleep/active schedule. If two neighboring nodes reside in two different virtual clusters, they wake up at listen periods of both clusters.

To maintain synchronization among neighboring nodes, the sensor nodes periodically transmit SYNC messages at the beginning of the active period. The SYNC messages allow sensor nodes to learn their neighbors' schedules so they can wake up at the proper time to transmit a message. To improve performance, however, sensor nodes adopt the schedule of their neighbors in several cases. If a node currently does not have a schedule and hears a SYNC message, it adopts the schedule and joins the virtual cluster. If a sensor node hears multiple, sufficiently different schedules, it adopts them all so as to allow communications between different virtual clusters. A sensor node that does not hear any SYNC messages from neighbors chooses its own schedule. In order to detect new schedules, sensor nodes periodically listen for a longer time period that enables them to detect neighboring schedules with high probability. Each sensor node performs a simple contention avoidance algorithm based on a random backoff to limit the number of SYNC message collisions.

To receive both SYNC packets and data packets, the listen period is divided into two parts. The first part is reserved to send or receive SYNC packets, and the second one for sending or receiving RTS/CTS packets, as shown in Figure 2.3. If a sensor node wants to send a SYNC packet, it starts carrier sense (CS) [199] when the receiver begins listening. It randomly selects a time slot to finish its carrier sense. If it has not detected any transmission by the end of the time slot, it sends its SYNC packet. The sensor node follows the same procedure when sending RTS, CTS, DATA and ACK packets.

The RTS and CTS packets contain the message transmission time, including time for the ACK packet, which permits the other neighboring nodes that are not concerned with this communication to sleep until the end of the transmission. S-MAC has been improved by the same authors in [199]. The authors introduce the adaptive listening technique, where nodes, in the same virtual cluster, that overhear a CTS, can wake up at the end of the data transmission to possibly act as the next hop. By doing this, the sensor nodes may transfer a message across two hops per frame time and decrease the latency. The authors also introduce a message fragmentation option, called message passing that allows sensor nodes to transmit relatively larger messages as smaller fragments using a single RTS/CTS exchange. Thus, if one fragment becomes corrupt due to collision or channel error, only the small fragment needs to be retransmitted instead of the entire data message.

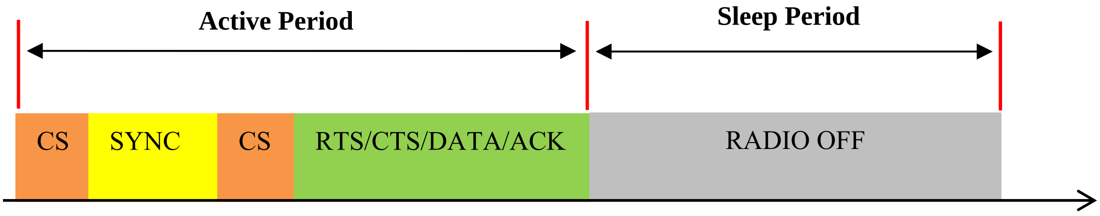


Figure 2.3: S-MAC frame format.

S-MAC offers several advantages like introducing the active /sleep period allows optimization of energy consumption. The concept of message-passing, where long messages are divided into small frames also decreases the energy consumption. The free synchronization method minimizes the problem of coordinating sensor nodes for communication and may provide adequate synchronization and clustering functionality for other protocols. We find also that S-MAC algorithm requires modest resources, such as memory for schedule offsets and timers for wakeup. Moreover, S-MAC can scale easily since the sensor nodes do not require any scalability coordination. S-MAC only coordinates neighbors using beacon messages, so sensor nodes do not have to forward or share large amount of state information.

However, S-MAC has some disadvantages. Sensor node can follow multiple schedules, which results in more energy consumption via idle listening and overhearing; the border nodes may die faster and cause segmentation along the borders of the virtual clusters. The static duty cycle of S-MAC can consume more energy and limit the protocol's performance. The duty cycle can be set based on expected application requirements, but S-MAC does not adapt to environment changing. Also, S-MAC does not expect to control virtual cluster size throughout the network. Varying cluster sizes have several impacts on the protocol's performance and large clusters can increase the message latency.

2.4.3 Timeout MAC protocol (T-MAC)

T-MAC protocol [183] extends the protocol S-MAC. T-MAC tries to reduce the idle listening by using a variable active period instead of using a fixed duty cycle schedule. To maintain an optimal active time under variable load, T-MAC dynamically determines its duration. Every node periodically wakes up to communicate with its neighbors and then, goes to sleep again until the next frame. Meanwhile, new messages are queued. Nodes communicate with each other using a RTS/CTS/DATA/ACK mechanism, which provides both collision avoidance and reliable transmission. In T-MAC, A node will keep listening and potentially transmitting, as long as it is in an active period. An active period ends when no activation event has occurred for an additional period or timeout (TA).

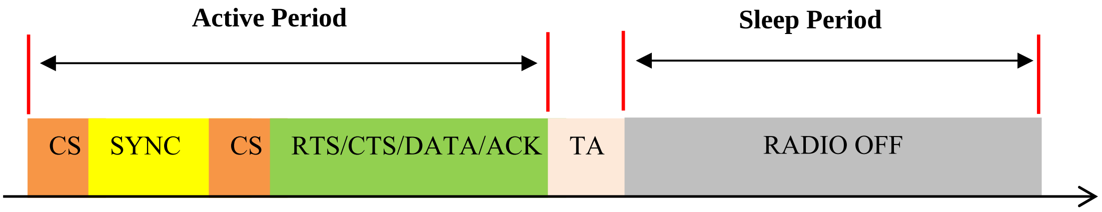


Figure 2.4: T-MAC frame format.

Figure 2.4 shows a T-MAC frame in which each node starts its frame by waiting and listening. If it hears nothing for a certain amount of time (CS), it chooses a frame schedule and transmits a SYNC packet, which contains the time until the next frame starts. If the node, during the CS time, hears a SYNC packet from another node, it follows the schedule in that SYNC packet and transmits its own SYNC accordingly. Nodes retransmit their SYNC once in a while. This allows new and mobile nodes to adopt an existing schedule.

If a node has a schedule and hears a SYNC packet with a different schedule from another node, it must adopt both schedules. It must also transmit a SYNC with its own schedule to the other node, to let the other node know about the presence of another schedule. After synchronization, sensor node starts its data transmission if the channel is still free during CS time. The active period ends in each case if any event occurs during the TA time.

To improve message latency, T-MAC introduces a new term called Future Request To Send (FRTS) message to solve the same problem addressed by the adaptive listening technique of S-MAC. Sensor nodes can use an FRTS packet to inform the next hop that it has a future message to transfer. If a node receives a CTS packet destined for another node, it sends immediately an FRTS packet. The FRTS packet contains the length of the data that will be sent. A node must not send an FRTS packet if it senses communication right after the CTS. A node that receives an FRTS packet knows it will be the future destination of an RTS packet and must be awake by that time. The node can determine this from the timing information included previously in the FRTS packet.

To avoid collision between the FRTS and the data packet that follows the CTS, the data packet must be postponed for the duration of the FRTS packet. The initial sender of RTS should send a small Data-Send (DS) packet to preserve the channel during the FRTS duration. After the DS packet, it must immediately send the normal data packet. T-MAC also considers the buffer size priority of the sensor node and gives the possibility to control the channel to the sensor node that has a full buffer. This sensor may immediately send an RTS message after receiving an RTS message from another sensor node, which allows it to can limit buffer overflow.

2.4.4 Adaptive Coordinated Medium Access Control (AC-MAC)

AC-MAC [69] introduces the adaptive duty cycle scheme within the framework of S-MAC. This protocol tries to improve latency and throughput in high traffic loads situation while remaining as energy-efficient as S-MAC. As shown in Figure 2.5, AC-MAC based on the number of packets queued at the MAC layer, allows sensor nodes that have queued packets to introduce multiple data exchange periods using one SYNC frame.

In the beginning of each duty cycle, each sensor node calculates the number of the message queued in its MAC layer and announces this value in the first RTS packet sent within the SYNC frame. Sensor nodes that receive this RTS message can then calculate the duty cycle to use within the virtual cluster for the current SYNC period.

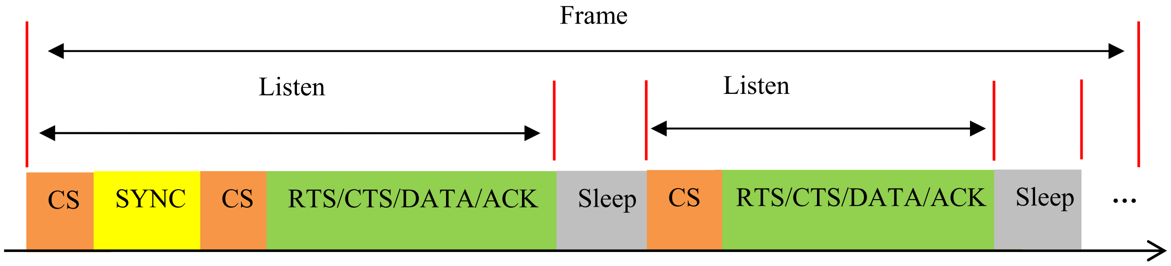


Figure 2.5: AC-MAC frame format.

To optimized latency and throughput, AC-MAC provides sensor nodes with many buffered messages a priority; each sensor node calculates its random backoff value from a contention window whose size varies inversely proportional to the amount of traffic it has buffered. To simplify the protocol, sensor nodes only adopt one schedule per SYNC period.

2.4.5 Pattern MAC (P-MAC)

P-MAC [186] is a '*time slotted*' protocol like S-MAC. P-MAC adjusts its duty cycle based on traffic conditions allowing sensor nodes with more data to utilize more slots than sensor nodes that have no data to transmit. In S-MAC, a node can stay awake for certain duration of a time slot, and go to sleep in the remaining duration. In P-MAC, a node can either be awake or asleep during a time slot.

In this protocol, sensor nodes share their proposed sleep and awake times for the next frame through a pattern sharing procedure. A sensor node gets information about the activity in its neighborhood beforehand through patterns. Based on these patterns, a sensor node can put itself into a long sleep for several time frames when there is no traffic in the network. If there is any activity in the neighborhood, a node will know this through the patterns and will wake up when required. Thus, P-MAC tries to save more power than that of SMAC and TMAC, without compromising the throughput.

A *sleep-wakeup* pattern is a string of bits (zero or one) indicating the tentative sleep-wakeup schedule for a sensor node over several slot times. Bit 1 in the string indicates that the node intends to stay awake during a slot time, while 0 indicates that the node intends to sleep. For example, a pattern of 0010 for a node indicates that the sensor node tentatively plans to be asleep for two consecutive slot times, stay awake in the third and go to sleep in the forth. This pattern is only a tentative plan and it can be changed according to the patterns of its neighboring nodes. Consequently, sleep-wakeup schedule for a node is derived from its own pattern and, the patterns of its neighboring nodes. Also, the pattern is defined as string of N bits indicating the tentative sleep-wakeup schedule during the M time slots of the upcoming frame. If $(N < M)$ then the pattern has to expand to fill the entire frame. For example, if N= 0010 and M= 10 time slots, the tentative pattern will be then, 0010001000.

In order to adapt to the traffic load, a node’s pattern is updated during each period using the local traffic information available at the node and exchanged at the end of each period. P-MAC uses pattern technical update similar to TCP (Transmission Control Protocol) window growth and each node generates its update pattern according to the following sequence:

1, 01, 001, 0001, ……..$0^{ß}$1, $0^{ß}$ 01, $0^{ß}$001,………..0^{N-1}1

In the first period, the working pattern of each node is 1, which expects that the traffic load is high at the beginning and each node should be awake. If there is no data to send during the first time slot of bit 1, then it indicates that the traffic load is potentially low and sensor node should update its pattern to 01. With the same manner, if the sensor node has no data to send during the second time slot, it updates its pattern to 001. This update continues by increasing the number of 0 until the number of 0 bits in the updated pattern reaches a predefined threshold ß. After this threshold, the number of 0 bits could increase if there is no data to send during period 1 until the number of 0 bits reaches N time slots. A sensor node’s pattern immediately increases to 1 whenever it has messages to send. Sensor nodes constantly update their pattern based on current conditions, but remain in operation according to the previously shared schedule. The sensor node shares its current pattern in the pattern exchange slots at the end of a frame using CSMA.

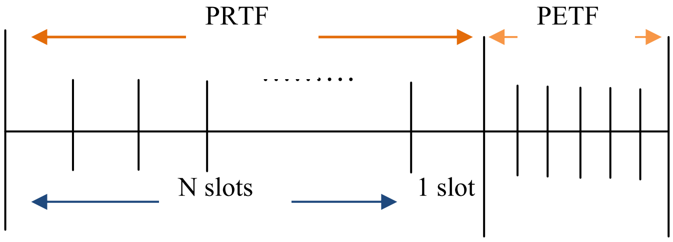


Figure 2.6: P-MAC frame format.

Node’s pattern is performed and exchanged according to P-MAC frame presented in Figure 2.6. P-MAC frame consists of two sub-frames. The first is called Pattern Repeat Time Frame (PRTF), during which each node repeats its current pattern and during N time slots, these time slots are reserved to send data. At the end of these N slots, PRTF has one additional time slot during which all the sensor nodes stay awake. This special time slot is used to speed up communication and to broadcast messages, which occurs after the regular data slots.

The second sub-frame is called Pattern Exchange Time Frame (PETF), during which new patterns are exchanged between neighbors. PETF is also divided into various time slots reserved to exchange the new patterns generated during PRTF at each node to reflect the latest traffic information. The last generated pattern during a particular PRTF becomes the pattern for the next PRTF, and will be advertised to the neighbors during the PETF. The pattern is cyclically repeated during PRTF such that each time slot has one pattern bit assigned. Patterns received from its neighbors during the preceding PETF are also repeated in the same way. If a node receives no new patterns from some of its neighbors during the preceding PETF (probably due to collisions), it then repeats its old patterns.

P-MAC offers a simple way to advertise messages and form schedules between sensor nodes in a neighborhood. The capability to quickly adapt to changing traffic conditions may also make P-MAC an attractive choice for a sensor network deployment. However, the schedule generation algorithm has several possible disadvantages. First, some sensor nodes may not receive an updated pattern due to channel errors while others correctly receive the update. This may lead to different schedules present in the same neighborhood and cause collisions, idle listening, and wasted transmissions. Also, the functionality of the protocol relates directly to the traffic intensity. Each time the sensor node operates in an active time slot, it performs the pattern update algorithm. During times of high traffic intensity, the processing requirements may become large as the sensor node operates in many active time slots.

2.5 Asynchronous sensor MAC protocols

Unscheduled MAC protocols offer the advantage of simplicity, without having to maintain and share state of neighboring nodes. Protocols in this category wake up the next hop node by continuously sending preambles or packets, and thus eliminate the synchronization overhead. A good number of asynchronous protocols are proposed as MAC layer solutions for WSNs. Some of these protocols have been surveyed in [107] and discussed under six categories. Static wake-up preamble [184], [79],[34], [101], [54], [146], [172], [45], [155], [96], [97], [88], [27], [37], [63], Adaptive wake-up preamble [22], [143], [75], [202], [80], [145], [200], [126], Collaborative schedule setting [152], [62], [193], Collision resolution [89], [59], Receiver initiated [45], [204], [140], [141], and Anticipation based [171], [173], [77], [76]. Earlier, in [91], these protocols were classified into four categories: Multiple transceiver [176], Multiple path [58], Event-centered [102], and Encounter-based [45]. Irrespective of the subclass to which a protocol belongs, synchronous MAC protocols use the same techniques, such as channel sensing and channel reservation messages to mitigate the effects of the common problems like a higher rate of collision, idle listening, and overhearing. For this reason, we prefer to simplify the classification and consider these protocols under only one main category. Now, let us know about the most import asynchronous MAC protocols.

2.5.1 Berkeley MAC protocol (B-MAC)

B-MAC [79] is a contention based MAC protocol. Like [34] and [5], B-MAC uses a preamble to wakeup sleeping neighbors. Sensor nodes, in this protocol, independently follow a sleeping schedule based on the target duty cycle for the sensor network. Since the sensor nodes operate on independent schedules, B-MAC uses very long preambles for message transmission. The preamble length is provided as parameter to the upper layer. The source node transmits a long enough preamble causing the destination node wake up and sensing it. Sensor nodes that sense this signal remain awake to receive the data following the preamble or return to sleep if they do not detect activity on the channel. Before transmitting, sensor nodes wait a random period of time to prevent any collision.

Figure 2.7 shows the communication mechanism of B-MAC. If a source node wishes to transmit, it precedes the data packet with a preamble that is slightly longer than the sleep period of the destination node. So, the destination node, at some point during the transmission of the preamble, will wake up and detect the preamble; it has to remain awake to receive the data packet.

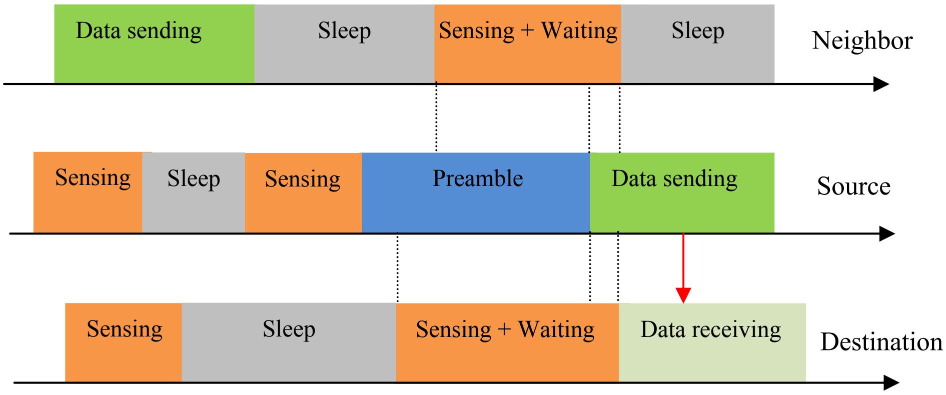


Figure 2.7: B-MAC communication mechanism.

B-MAC utilizes software automatic gain control as a method of Clear Channel Assessment (CCA), which accurately determines if the channel is clear, thus effectively avoiding collisions. This is a necessity so that the node can differentiate between a noise and a signal, due to the fact that ambient noise is prone to environmental changes. This is achieved by taking signal strength samples when the channel is assumed to be free, such as immediately after transmitting a packet. These samples are stored in a FIFO queue and the median of the queue is added to an exponentially weighted moving average with decay. This value gives a fairly accurate estimate of the noise floor of the channel. Effectively, a node, before transmission, takes a sample of the channel. If the noise is below the noise floor, the channel is clear and it can send immediately. This mechanism permits to increase the reliability of channel assessment and provides a great deal of flexibility through a protocol interface that allows the sensor node to change many operating variables in the protocol, such as delay and backoff values.

A key challenge of B-MAC is implementing check intervals that are very short, which ensures a reasonable length for the preamble. Carrier sense duration also has to be very short so that receiver does not have to spend too much energy listening to the communication channel. A carrier sense must be accurate to reduce latency of transmission and energy consumption at sender.

The Low Power Listening (LPL) approach used by B-MAC which employs a long preamble is suboptimal in terms of energy consumption, is subject to overhearing, as well as it introduces excess latency at each hop [101]. This issue is threefold. First, the receiver typically has to wait the full period until the preamble is finished before the DATA/ ACK exchange can begin, even if the receiver has woken up at the start of the preamble. Second, LPL suffers from the overhearing problem, where receivers which are not the target of the sender also wake up during the long preamble and have to stay awake until the end of the preamble to find out if the packet is destined for them. This wastes energy at all non-target receivers within transmission range of the sender. Third, because the target receiver has to wait for the full preamble before receiving the data packet, the per-hop latency is lower bounded by the preamble length. Over a multi-hop path, latency can accumulate to become substantial.

2.5.2 WiseMAC protocol

WiseMAC [22], [7] is CSMA-based medium access control protocol. It uses the preamble sampling technique [6] to minimize power consumption when listening to an idle medium. In this technique, a preamble precedes each data packet for alerting the receiving node. All nodes in a network sample the medium with a common period, but their relative schedule offsets are independent. If a node finds the medium busy after it wakes up and samples the medium, it continues to listen until it receives a data packet or the medium becomes idle again. The preamble transmission time (T_p) is initially set to be equal to the sampling period (S_p).

However, the receiver may not be ready at the end of the preamble, due to reason like interference, which causes the possibility of over-emitting type energy waste. Moreover, over-emitting is increased with the length of the preamble and the data packet, since no handshake is done with the intended receiver. To reduce the power consumption incurred by the predetermined fixed-length preamble, WiseMAC offers a method to dynamically determine the length of the preamble. This method uses the knowledge of the sleep schedules of the neighboring nodes. The nodes learn and refresh their neighbor's sleep schedule during every data exchange as part of the acknowledgement message. So, each node keeps a table of sleep schedules of its neighbors. Based on this table's information, the sender node schedules transmissions by choosing the minimum requirement preamble. To decrease the possibility of collisions caused by that specific start time of wake-up preamble, a random wake-up preamble is advised. To avoid the clock drift between the source and the destination, a lower bound for the preamble length (T_p) is calculated as the minimum of destination's sampling period (S_p) and the potential clock drift with the destination which is a multiple of the time since the last ACK packet arrival. Figure 2.8 shows the communication mechanism of this protocol. WiseMAC has been extended more bit in [144]. This improvement allows a common destination node to automatically stay awake, at the end of the wake-up period, when more traffic has to be handled, which improves the delay.

According to the simulation results [22], WiseMAC performs better than one of the S-MAC variants. Moreover, its dynamic preamble length adjustment results in better performance under variable traffic conditions. In addition, clock drifts are handled in the protocol definition which mitigates the external time synchronization requirement.

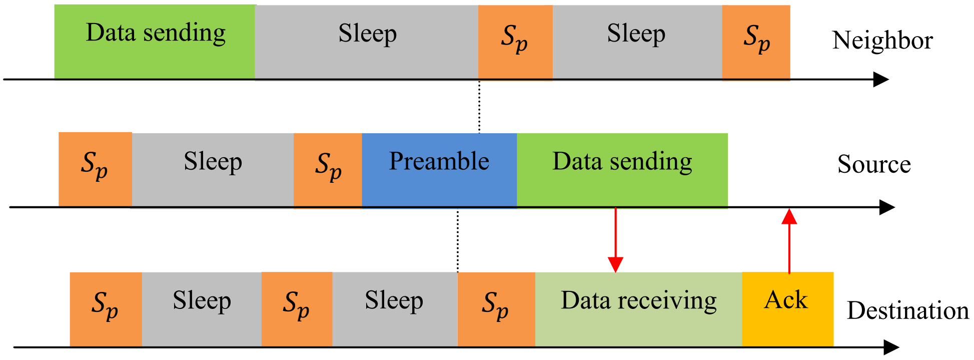


Figure 2.8: WiseMAC communication mechanism.

However, the decentralized sleep-listen scheduling which results in different sleep and wake-up times for each neighboring node represents the main drawback of WiseMAC. This is especially an important problem for broadcast type of communication, since broadcasted packet will be buffered for neighbors in sleep mode and delivered many times as each neighbor wakes up. However, this redundant transmission will result in higher latency and more power consumption. In addition, WiseMAC may suffer more from the hidden terminal problem. That is because WiseMAC is based on non-persistent CSMA. This problem will result in collisions when one node starts to transmit the preamble to a node that is already receiving another node's transmission where the preamble sender is not within the same communication range.

2.5.3 A Short Preamble MAC (X-MAC)

X-MAC [101] is a low-power MAC protocol that strives to overcome the shortcomings of the long preamble used by B-MAC [79]. X-MAC uses a shortened preamble approach and includes the ID of the target sensor node in the preamble. So, non-target receivers can realize that they are not concerned by this transmission and quickly go back to sleep. This solution addresses the overhearing problem. However, X-MAC introduces the strobed preamble. This approach allows the target receiver to interrupt the long preamble as soon as it wakes up and determines that it is the target receiver. This is accomplished by dividing the one long preamble into a series of short preamble (S_p) packets, each containing the ID of the target node (Figure 2.9). Accordingly, instead of sending a constant stream of preamble packets, the protocol inserts small pauses between the series of short preamble packets, during which time the transmitting node pauses to listen to the medium.

These gaps enable the receiver to send an early ACK packet back to the sender by transmitting the ACK during the short pause between preamble packets. When a sender receives an ACK from the intended receiver, it stops sending preambles and sends the data packet. This allows the receiver to cut the short excessive preamble, which reduces per-hop latency and energy spent unnecessarily waiting and transmitting. However, the non-target receivers, after going back to sleep, may wake up and sense the medium for several periods while the data transmission is not yet achieved which wastes energy for these nodes.

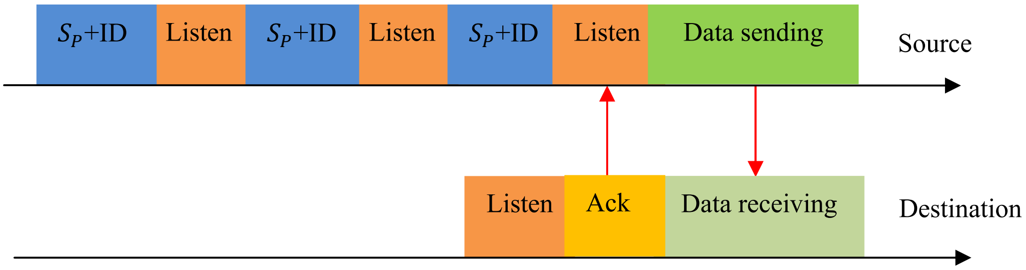


Figure 2.9: X-MAC communication mechanism.

2.5.4 Spatial Correlation-based Collaborative MAC protocol (CC-MAC)

CC-MAC [102] attempts to conserve energy, while fulfilling application requirements, by utilizing the knowledge that sensor nodes located near each other generate correlated measurements. To achieve energy savings, CC-MAC filters measurements from highly correlated sensor nodes in an effort to reduce the number of messages the sensor network must handle.

The authors introduce an analytical framework to investigate the relation between the positions of sensor nodes in the event area and the event estimation reliability. Based on analysis within the framework, the authors introduce the Iterative Node Selection (INS) algorithm that creates a sample topology for the sensor network to exploit spatial correlation and filter correlation between the nodes. Thus, INS creates a correlation region defined by its correlation radius, based on statistical information about the sensor network deployment. Sensor nodes closer than the correlation radius produce correlated data. Therefore, if a node transmits data, the nodes in its correlation region are not required to send data. This algorithm is executed by the sink during the network initialization to calculate and distribute the correlation radius throughout the network.

CC-MAC consists of two components: the Event MAC (E-MAC), which filters sensor node measurements to reduce traffic and the Network MAC (N-MAC), which forwards the filtered measurements to the sensor network sink. More specifically, E-MAC is executed when sensor node wants to transmit its sensed data to the sink, while N-MAC is performed when a node receives data from another node and tries to forward it to the next hop.

E-MAC reduces the traffic generated in an area by having only sensor nodes separated by at least the correlation distance measurements. Other nodes periodically sleep to save energy and awake to forward messages. Correlated sensor nodes rotate the role of generating measurements to balance energy consumption throughout the network. Sensor nodes get elected as active nodes and to represent the correlated sensor nodes by winning contention for the wireless medium. E-MAC modifies the standard RTS/CTS/DATA/ACK scheme by introducing a First Hop (FH) bit into the control packet headers. The sensor node actively reporting measurements sets the FH bit when it transmits messages so that other nodes can decide to generate measurements or not. If a sensor node does not belong to any correlation region, it will then begin to also generate measurements. Once the originating sensor node has transmitted its sensed data, the FH bit gets cleared and the message will be forwarded using N-MAC protocol.

After removing the redundant data present in multiple measurements by E-MAC, N-MAC forwards it from source nodes to the sink. However, the forwarded traffic may become more important. To compensate for this, N-MAC protocol transmissions take preference over E-MAC transmissions through the use of smaller backoff windows and inter-packet times in same way that the PCF (Point Coordination Function) in IEEE 802.11 receives preferential access to the wireless channel over the DCF (Distributed Coordination Function).

The simulation results show that CC-MAC can achieve a good balance of low energy consumption and favorable traffic performance compared to the other protocols. Additionally, the analytical framework proposed in this protocol allows users to apply the CC-MAC protocol to applications with various data fidelity requirements. CC-MAC, however, requires that sensor nodes possess or obtain ranging information about their neighbors in order for N-MAC to filter data from correlated sensor nodes.

The computational resources required by the INS algorithm may also limit the application of the protocol. For example, if the number of sensing events increases, the overhead associated with computing the correlation radius and distributing throughout the network increases. For large networks, this overhead may become significant.

2.5.5 Convergent MAC protocol (CMAC)

CMAC [171] uses unsynchronized wakeup scheduling with a predefined idle duty-cycle. In this wake-up scheduling scheme, the sleep period is fixed according to the duty cycle and active period. Instead of a long preamble to activate the receiver, CMAC uses aggressive RTS. To detect an RTS, nodes periodically wake up and double check the channel for activities. CMAC initially uses anycast to transmit packets to a potential forwarder that wakes up first. *Awake* candidate receivers will contend to be the anycast receiver by prioritizing their CTS transmissions according to their routing metrics to the sink. After receiving CTS, the data packet will be sent to the sender of the CTS immediately. To overcome the disadvantages of anycast, such as higher RTS/CTS overhead and route stretch, CMAC converges from anycast to unicast once it establishes contact with a receiver having a sufficiently good routing metric.

As discussed above, CMAC has three main components: Aggressive RTS equipped with double channel check for channel assessment, anycast to quickly discover a forwarder, and convergent packet forwarding to reduce the anycast overhead.

In the aggressive RTS, CMAC uses multiple RTS packets separated by fixed short gaps instead of a long preamble. The short gap allows receivers to send back CTS packets. CMAC sends all RTS packets without clear channel assessment (CCA) except the first one. In very low duty cycle, nodes must assess the channel very quickly each time they wake up. However, if the receiver wakes up during the gap between two RTS transmissions, it may miss the RTS burst. So the authors propose to use double channel check which works by assessing the channel twice with a fix short separation between them each time a node wakes up.

The anycast mechanism is used to send the RTS burst, where more than one node in the forwarding set may try to reply to the same RTS, and the one closest to the destination should be elected to receive the data packet. The CTS transmissions are prioritized according to the routing metrics of contending nodes. Nodes with better routing metrics sends CTS packets earlier, while other overhearing nodes cancel their CTS transmissions accordingly, and nodes that can make little progress are excluded.

To overcome the shortcomings of anycast like overhead of anycast RTS/CTS exchange, the authors propose convergent packet forwarding. In such mechanism, the node will remain awake for a short duration after receiving a data packet. If the latest anycast receiver has a routing metric close to the best, CMAC will use unicast and send the data directly to this node without using RTS/CTS packets.

The experiment and simulation results show at low duty cycles that CMAC achieves the throughput and latency performance and outperforms other energy efficient protocols like

BMAC [79], SMAC [198] and GeRaF [138],[130],[131]. The issue here is that a lower duty cycle MAC protocol can save energy, but low activity levels place a limit on the protocol's complexity, the possible network capacity, and the message latency.

2.6 Hybrid sensor MAC protocols

Hybrid MAC protocols aim to leverage advantages and mitigate the disadvantages of the synchronous and asynchronous protocols two. These protocols try to adapt their behaviors according to the traffic loads and patterns. Several protocols of this category [211], [210], [48], [124] can be found in [212].

2.6.1 Zebra MAC protocol (Z-MAC)

Z-MAC [67] combines TDMA and CSMA. It adapts to the level of contention in the network. Under low contention, it behaves like CSMA, and under high contention, like TDMA. Z-MAC uses CSMA as the baseline MAC scheme and uses a TDMA schedule to enhance contention resolution and assign time slot during the network deployment phase. The authors adopt a centralized channel reuse scheduling algorithm [66]. After the time slot assignment, each node reuses its assigned slot periodically in every predetermined period, called *frame*. The owner node is a node that is assigned to a time slot. The others are non-owners of that slot. In the adopted algorithm [66], more than one owner per slot may exist and two nodes further than two-hop neighborhoods can own the same time slot.

Z-MAC uses priority scheme to switch between CSMA and TDMA depending on the level of contention. A sensor node may transmit during any time slot. It samples the channel and transmits a packet when the channel is clear. However, an owner of that slot always has higher priority over its non-owners in accessing the channel. The owners are given earlier chances to transmit and their slots are scheduled a priori to avoid collision, but when a slot is not in use by its owners, non-owners can steal the slot.

During the deployment phase, each sensor node learns its two-hop neighbors, assigned to a time slot, chooses its time frame and forwards its frame size and slot number to its two-hop neighbors. In transmission phase, sensor node can be in one of two modes: Low Contention Level (LCL) or High Contention Level (HCL). A node is in HCL only when it receives an Explicit Contention Notification (ECN) message from a two-hop neighbor within the last period. Otherwise, the node is in LCL. A node sends an ECN when it experiences high contention based on the packet loss rate. ECN permits to avoid the problem of hidden host. Z-MAC uses backoff, CCA and LPL interfaces of B-MAC [79] to perform its LCL and HCL.

Z-MAC requires local clock synchronization only among neighboring senders and when they are under high contention (HLC). Z-MAC adopts a technique from RTP/RTCP (Real-Time Transport Protocol) [53] to implement this synchronization. The performance results show that Z-MAC outperform B-MAC under medium to high contention while it shows competitive, but slightly less performance than B-MAC under low contention (especially, in terms of energy efficiency). According to the authors, Z-MAC finds its utility in applications where expected data rates and two-hop contention are medium to high.

2.6.2 CSMA/TDMA hybrid MAC protocol (hybrid-MAC)

In this protocol [117], the authors propose a hybrid MAC protocol based on IEEE 802.15.4 standard [68]. Their main idea consists of adding a dynamic TDMA period into contention access period of this standard. IEEE 802.15.4 standard has two operational modes, non-beacon mode and beacon-enabled mode. In non-beacon mode, channel access is only based on CSMA/CA. But in beacon-enabled mode, a beacon frame has to transmit in specified time intervals by the coordinator. The beacon frame is divided into an active and an inactive period. Devices are in sleep mode during the second period for energy saving purposes. The active period also consists of a Contention Access Period (CAP) and a Contention Free Period (CFP).This beacon permits the devices synchronize themselves for accessing the channel. The authors use the beacon-enabled mode and propose a modified beacon frame by assigning TDMA slots to sensor nodes during the CAP period (Figure 2.10).

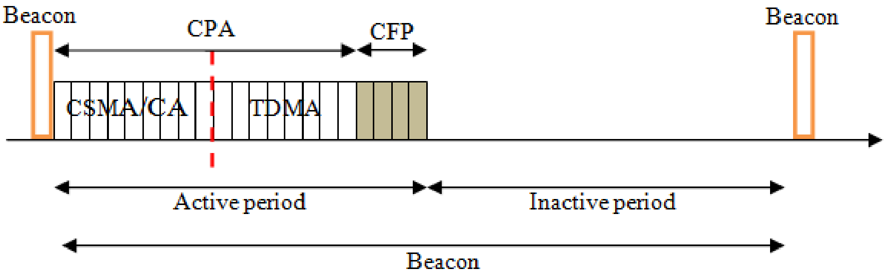


Figure 2.10: Hybrid-MAC communication mechanism.

To determine the limit between TDMA and CSMA in CAP period, the authors consider two parameters: channel utilization level in CAP and the amount of pending data in nodes' queues. To determine the queue state, eight different values have been utilized by using three reserved bits of the standard data packet header. These values define the queue state level meter which indicates the fraction of queue being occupied. Thus, each node gives the coordinator a more accurate description of its queue state. The coordinator maintains the queue state of network nodes in local array structure. Each array cell belongs to one network node and has the initial value of 0. Each time a data packet is received the coordinator checks the queue state of the sending node and updates its corresponding array cell. The channel utilization is evaluated as a simple function of number of used slots, number of unused slots and number of slots having collision.

Two cases explain the decreasing of channel utilization, increasing collisions and reduction of used slots. In the first case, the coordinator checks the queue state array values and assigns TDMA slots to the nodes in descending order of their queue state values. In the second case, when the low channel utilization is caused by the decreasing of the used slots number and low collisions, the TDMA period length should be shortened.

In this protocol, when a TDMA slot is assigned to a node, it will not be authorized to send data in the CSMA/CA period in the same beacon frame. Thus, number of nodes participating in the contention is decreased and fewer collisions occur. Moreover, TDMA slots are only assigned to nodes having queued data, thus permitting to avoid the common problem of under-utilization of slots in TDMA methods. However, the main advantage of this protocol resides in the use of coordinator node.

2.7 Summarization

In this study, we have noticed that each MAC protocol proposed for sensor networks tries to minimize energy consumption by reducing collisions, limiting idle listening, and overhearing. Synchronous MAC protocols require that sensor nodes expend some considerable part of their energy to share state and maintain synchronization.

Table 2.1: Synchronous MAC protocols.

Protocol	Medium access scheme	Energy consumption	Latency	Throughput
NAMA [95]	- Time is divided into blocks. - Block is divided into sections of several time slots. - Sensor node calculates its priority using its ID and the chosen time slot. - Time slot affected according to the highest priority among the two hop neighboring nodes	Decreased by: - Collision avoidance Increased by: - Resources required to calculate the priority	Decreased by: - Collision avoidance Increased by: - Time required to calculate the priority	Increased by: - Collision avoidance Decreased by: - Time required to calculate the priority
LAMA [95]	- Use the DSSS scheme. - Use priority node among two hop neighboring nodes.	Same as NAMA + DSSS scheme increases node cost and energy consumption	Same as NAMA + DSSS scheme increases latency	Same as NAMA + DSSS scheme decreases throughput
PAMA [95]	- Use the DSSS scheme. - Use node priority among two hop neighboring nodes. - Use link priority	Same as LAMA	Same as LAMA	Same as LAMA

(Continued)

Table 2.1: Continued.

Protocol	**Medium access scheme**	**Energy consumption**	**Latency**	**Throughput**
S-MAC [198]	In each active period sensor nodes: - Sample the channel during a random time slot. - Send its SYNC packet if channel is free. - Send RTS, CTS, DATA and ACK packets	Decreased by: - Periodic sleep - Message passing Increased by: - Idle Listening - Overhearing	Decreased by: - Periodic sleep - Message passing Increased by: - Idle Listening	Increased by: - Message passing Decreased by: - Periodic sleep
T-MAC [183]	In each active period sensor nodes: - Do the same like S-MAC. - Adapt the active period to the traffic behavior.	Decreased by: - Adaptive Duty Cycle	Increased by: - Adaptive Duty Cycle	Decreased by: - Adaptive Duty Cycle
AC-MAC [69]	In each active period sensor nodes: - Do the same like S-MAC. - Announce the number the queued message in the beginning of the active period. - Adapt the active/listen period according to the queued message.	- Same as S-MAC	Decreased by: - Adaptive active period. Increased by: - Idle Listening	Increased by: - Adaptive active period. Decreased by: - Idle Listening
P-MAC [186]	- Use an adaptive active/listen pattern. - Repeat the active/listen pattern during the PRTF period. - Adapt the active/listen pattern according to the traffic load. - Update and exchange the active/listen pattern at the end of each PRTF period.	Decreased by: - Adaptive active/listen pattern. Increased by: - Pattern update algorithm in case of high traffic	Decreased by: - Adaptive active/listen pattern. Increased by: - Pattern update algorithm in case of high traffic	Increased by: - Adaptive active/listen pattern. Decreased by: - Pattern update algorithm in case of high traffic

Additionally, the extent and frequency to which the sensor network undergoes organization and reorganization can greatly affect its performance. However, the protocols of this class may allow sensor nodes to remain asleep for longer periods of time and forward messages with less effort than those using unscheduled MAC protocols since the sensor node has some indication of its neighbor's active/sleep schedules. Nevertheless, Synchronous MAC protocols have obvious disadvantage in scalability and adaptability. Because, sensor nodes need to strictly follow the assigned communication time slots in case of TDMA scheme and require frame synchronization which involves the complicated tasks of slot allocation and schedule maintenance. Table 2.1 provides a summary of the studied synchronous MAC protocols.

Table 2.2: Asynchronous MAC protocols.

Protocol	Medium access scheme	Energy consumption	Latency	Throughput
B-MAC [79]	- Use long preamble before sending data to wakeup sleeping neighbors. - Independently follow a sleeping schedule.	Decreased by: - Independent sleep schedule. Increased by: - Overhearing.	Increased by: - Overhearing.	Decreased by: - Long preamble.
WiseMAC [22], [7]	- Use dynamic length preamble. - Use the neighbor's sleep schedule to adjust the preamble length.	Decreased by: - Independent sleep schedule. Increased by: - Hidden terminal problem. - Cost of neighboring sleep schedules table	Increased by: - Hidden terminal problem. - Cost of neighboring sleep schedules table	Decreased by: - Hidden terminal problem. - Cost of neighboring sleep schedules table
X-MAC [101]	- Set up sensors into cluster nodes and elect periodically a cluster head. - Cluster head elected based on its weight calculated based on its energy available and the signal strength for the signal broadcasted by the base station. - Cluster heads constitute the data dissemination path, next cluster head selected based on its weight.	Decreased by: - Short preamble Increased by: - Hidden terminal problem. - Several wakeups during data transmission (data transmission time does not known)	Increased by: - Hidden terminal problem.	Decreased by: - Hidden terminal problem.
CC-MAC [102]	- Create correlation regions using INS algorithm. - Only one node sends data from correlation region.	Decreased by: - Load balance used among the sensor nodes in correlation region. Increased by: - Resources required by the INS algorithm	Increased by: - Backoff windows and inter-packet times	Decreased by: - Backoff windows and inter-packet times
CMAC [171]	- Send multiple RTS packets separated to wake up the receiver. - Receiver double checks the medium and sends back CTS packets. - Send data to the receiver that has the best routing metric.	Decreased by: - Adaptive active /sleep period. Increased by: - RTS/CTS mechanism. - Low activity in low duty cycle	Increased by: - RTS/CTS mechanism.	Decreased by: - RTS/CTS mechanism.

(Continued)

Table 2.2: Continued.

Protocol	**Medium access scheme**	**Energy consumption**	**Latency**	**Throughput**
Z-MAC [67]	- Each node has its time slot according to TDMA scheme and it has the highest priority to use it. - Use backoff, CCA and LPL interfaces of B-MAC to use the time slot. - Time slot can be used by any neighboring node. - Use clock synchronization only among neighboring nodes.	Decreased by: - The use of any free time slot. Increased by: - LPL scheme that use long preamble. - Synchronization algorithm cost.	Increased by: - Long preamble.	decreased by: - Long preamble
hybrid-MAC [117]	- Coordinator node sends periodic beacon frame. - Beacon frame has two periods active and inactive period, sensor nodes sleep in the inactive period. - The active period has two contention types CAP and CFP. - In the CAP period, sensor node can use CSMA/CA or TDMA to send its data. - The limit between CSMA/CA and TDMA determined by the coordinator based on queue stat level of each node and the channel utilization.	Decreased by: - The adaptation of the limit between CSMA/CA and TDMA to the queue stat level of each node and the channel utilization. Increased by: - The cost of the centralized communication at coordinator node level.	Increased by: - Fixed inactive period.	decreased by: - Fixed inactive period.

Asynchronous MAC protocols are characterized by their simplicity. Neighboring sensors nodes do not need to schedule common active/sleep plan; thus, minimizing resource utilization within a sensor node and eliminating the protocol overhead. However, coordinating neighboring sensor nodes for communication becomes a primary function of these protocols. Using the preamble to establish the communication between neighboring nodes eliminates the synchronization need but may increase the overhearing and thus energy waste due to the long preamble. For some protocol, this energy waste can be mitigated using short preamble. Nevertheless the short preamble is not a complete solution to make the protocol adaptive to the traffic load, especially in case of low duty cycle where the number of wasted short preambles can be increased. Table 2.2 shows a comparative summary of the asynchronous MAC protocols discussed in this work.

Hybrid MAC protocols combine the qualities of the two previous classes. The fist class presents collision-free, high channel utilization and throughput, which are suitable for high traffic load situations. The second one possesses simplicity, flexibility and robustness, hence enabling nodes to be adaptable to network topology change. Hybrid MAC protocol may switch its process between the synchronous and asynchronous mode according to the traffic loads. According to the experiment done in [153], hybrid protocol can help the monitoring system to give an almost real-time answer to the emergencies. Table 2.3 recapitulates the main Key design points and performance of the hybrid MAC protocols presented above.

2.8 Conclusion

In this study we have covered many MAC protocols proposed thus far for sensor networks, but many more exist. Based on synchronization mode, the studied protocols have been classified on three classes, synchronous, asynchronous and hybrid MAC protocols. In the first class, sensor nodes change the radio state periodically and need clock synchronization to maintain common active/sleep schedule. Moreover, node may have pre-assigned time slots to transmit the data but each node has to listen to the time slots of its neighbors in order to synchronize. In the second class, sensor node has its own active/sleep schedule that can be established based on the medium activity. Sensor node may adjust its active/sleep time according to the medium state. Hybrid protocols combine between synchronous and asynchronous mode, sensor node may switches its communication mode according to the traffic behavior.

Throughout this work, Energy consumption and latency have been considered as main studied parameters. Each protocol provides benefits for certain applications or under certain conditions according to the chosen design. But there are still many more challenges that need to be solved MAC protocols. it is still need to find out the suitable solution for real time support and energy efficiency, it remains also an open issue, and one of great interest, if a general, flexible MAC protocol exists that supports various applications and operating environments while consuming minimal power and offering acceptable traffic characteristics.

Chapter 3

Data Dissemination in Wireless Sensor Networks

3.1 Introduction

Wireless sensor networks can be applied in several application domains. This kind of network is constructed with a large number of tiny and smart sensor nodes deployed in an ad-hoc manner over an area of interest for collecting the expected information. Among various roles and objectives of WSN, the most crucial objective is data dissemination [157], [46] which is also one of the key problems faced by the sensor nodes. In this environment, the network supervisor (or, administrator) may need to interrogate the sensors by spreading his interests over the whole network, whereas a sensor node needs to notify the supervisor when interested event occurs. During data dissemination processes, sensor nodes communicate with each other to deliver the sensed data to the supervisor via the sink node. Each node in the network acts as a router and may sense the data directly or receive it through other intermediate nodes.

One of the major barriers of a Wireless Sensor Network is that the sensor nodes have limited transmission range. Also their processing and storage capabilities as well as their energy resources are limited [61]. Hence, the limitations of resources are often noted as the key challenge to tackle for designing any operational protocol. Data dissemination within WSN is not an exception to this. In practice, data dissemination protocol for WSNs is responsible for delivering the sensed data using a valid path between source and destination node and has to ensure reliable multi-hop communications. Because of the relentless efforts of hundreds of researchers, several data dissemination protocols have been proposed for wireless sensor networks by this time. Considering all the inherent challenges in WSN as noted above, it is an interesting issue to investigate how the data disseminations are modeled for such networks. This is the core intent of this chapter to analyze various aspects of the design methodologies of data dissemination of the most significant protocols. We describe the achievements so far in this area and highlight the relative strengths and weaknesses of the data dissemination models of various protocols for WSNs. The study presented in this chapter has been published in [115].

The rest of this chapter is organized as follows: Following the Introduction, Section 3.2 describes the WSN data dissemination mechanism, notes the previous related surveys and provides new taxonomy of WSN data dissemination. Based on the classification of data dissemination protocols presented in Section 3.2, Section 3.3 and Section 3.4 present an overview of the major data dissemination strategies with static and mobile sinks respectively. The issues of single and multiple sinks are investigated in detail in separate subsequent sections. Section 3.5 resumes and compares the studied data dissemination protocols. Finally, Section 3.6 presents concluding remarks with directions on open research issues.

3.2 Data dissemination protocols

To design a data dissemination protocol for wireless sensor networks, it has to consider several parameters related to this environment [70]. Limited energy resources of sensor nodes, quality of the wireless channel, packet loss and latency constitute the main important issues that are needed to be considered. Energy consumption effectiveness represents the most significant performance metric which influences directly on the lifetime of network. This is why, several data dissemination protocols have ignored other performance metrics such as the data transmission time, latency, and have put more emphasis on energy consumption [16], [135], [129], [4], [103], [31], [116], [73], [90], [17], [99], [201], [191]. The goal of a data dissemination protocol is to find and maintain a valid path towards sink or base station by which data forwarding process would consume minimum of energy.

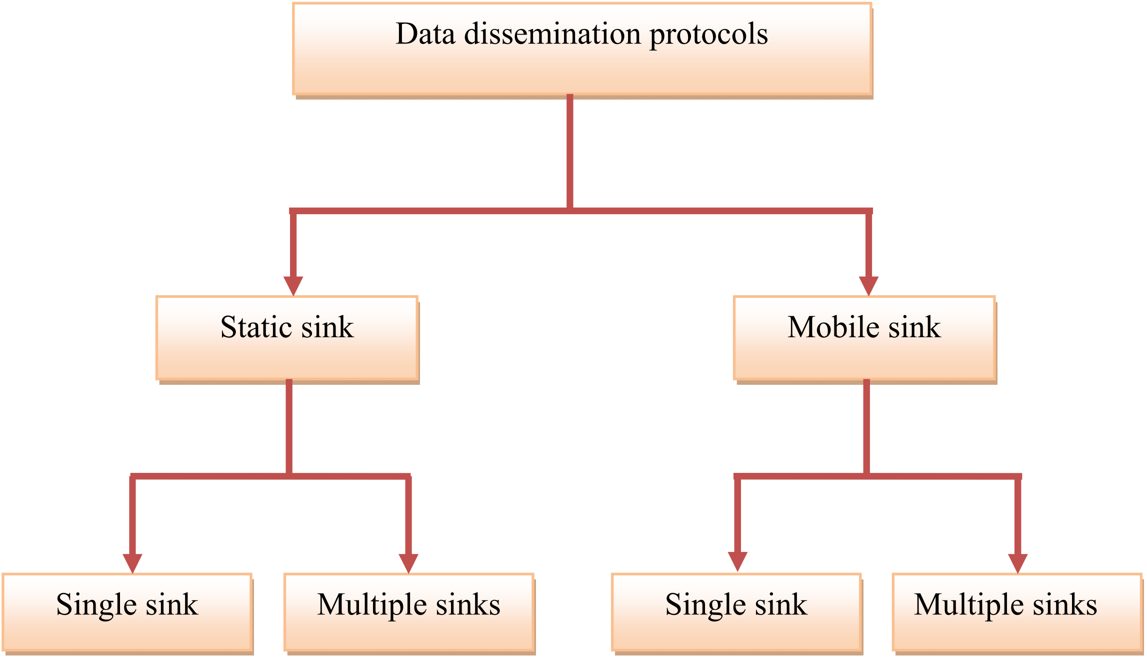


Figure 3.1: Taxonomy of data dissemination protocols

Several data dissemination strategies have been proposed for wireless sensor networks. Their principal ideas mainly are related to the class to which they belong. In literature, these approaches have been classified [104], [83], [177], [70], according to the network architecture, the initiator of communication, the path establishment, and so on. In [46], the authors highlight the special features of sensor data collection in WSNs, by comparing with both wired sensor data collection network and other WSN applications. The authors describe a basic taxonomy and propose to break down the networked wireless sensor data collection into three major stages: namely, the deployment stage, the control message dissemination stage, and the data delivery stage. A literature survey on data collection in WSNs with mobile elements has been presented in [105]. In this work, the data collection issue has been studied through three separate phases. Discovery phase allows nodes to detect the presence of the mobile elements; data transfer phase defines the communication process between a mobile element (ME) and its one-hop neighbors. In the last phase of routing to mobile elements, the authors present and discuss some data dissemination protocols with mobile elements into flat routing and proxy-based routing classes.

Our classification [115] is mainly based on the number of sink(s) (single or multiple) and their nature (static or mobile). Some protocols require that the sink node has to be static and sensor nodes cannot achieve their requirements without this assumption. Other protocols support the mobile sink concept and try to exploit this possibility to provide a good performance. Moreover, in these kinds of protocols, some use more than one sink which requires additional management and coordination operations.

Considering these key parameters, we suggest classifying the existing data dissemination protocols according to the taxonomy shown in Figure 3.1. Two great classes can be found; (i) Static sink data dissemination protocols and (ii) Mobile sink data dissemination protocols. Each class is again divided into two subclasses related to the number of sink(s).

3.3 Static sink data dissemination protocols

Sink node in a WSN is the most important entity. The collected sensor readings from sensory field have to be disseminated to a predefined sink for analysis and processing. The data dissemination strategies with single static sink try to prolong the network lifetime by unbalancing the traffic load using multiple data dissemination paths. Nevertheless, sensor nodes near the sink still deplete their energy faster than that of other nodes due to their heavy overhead of relaying messages. This uneven energy depletion phenomenon causes degraded network performance and limits network lifetime. If all sensors around the sink consume their energy, the sink will be isolated from the network, and then the entire network would fail. Using multiple static sinks can significantly improve the network performance in terms of latency and energy consumption. Having multiple sinks in the network reduces the distance between sensor nodes and a sink, thus can improve both energy consumption and latency [65], [50], [139], [47], [32], [213].

A WSN with multiple sinks can be regarded as set of sub-networks, each of which is composed of a single sink. The number and the locations of sinks should be thoroughly studied as they could directly affect the network lifetime [84], [38], [197], [43].

3.3.1 Single static sink data dissemination protocols

3.3.1.1 Low Energy Adaptive Clustering Hierarchy (LEACH)

LEACH [13] is the first and most commonly known energy-efficient hierarchical clustering algorithm for wireless sensor networks. LEACH is a cluster-based protocol, which includes distributed cluster formation (Figure 3.2). LEACH randomly selects a few sensor nodes as cluster-heads and rotates this role to evenly distribute the energy load among the sensors in the network. In LEACH, the cluster-head node aggregates the sensed data arriving from nodes that belong to its cluster, and sends an aggregated data to the base station in order to reduce the amount of information that must be transmitted to the base station.

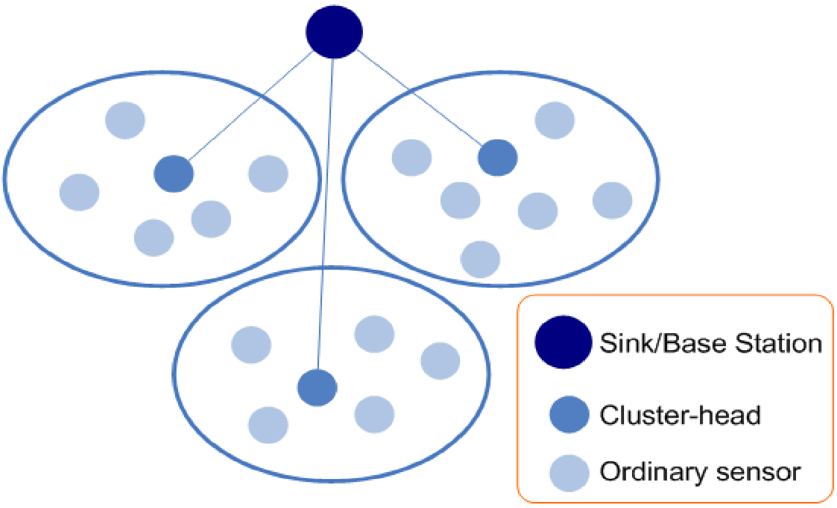


Figure 3.2: LEACH architecture.

LEACH process starts with the entire nodes organizing themselves through the clustering algorithm to form a cluster where one node will be elected as a head node or cluster head. Energy will be depleted more if the cluster head is fixed into one node, thus LEACH has the ability to rotate the cluster head among the nodes in the local cluster. LEACH protocol uses aggregation method to gather all information from the sensor nodes in the local cluster where the cluster head will collect the information for sending to the base station. LEACH protocol design can be divided into three different parts:

- Nodes clustering,
- Data gathering for aggregation,
- Cluster head rotation.

In node clustering setup, each sensor node will select in which cluster it belongs based on the distance between the node and cluster head. The process needs the cluster head to broadcast a message to all its neighboring nodes which alerts them that it is a cluster head. After receiving all the messages from the nodes that would like to be included in the cluster and based on the number of nodes in the cluster, the cluster-head node creates a TDMA (Time Division Multiple Access) schedule and assigns each node a time slot to transmit its data. This schedule is broadcast to all the nodes in the cluster. This schedule permits the nodes to turn off their transmitters if there is no activity in the cluster. Hence, this mechanism reduces inter-cluster collision and energy consumption.

Data fusion or aggregation is to compact the data in cluster head for sending to the base station when all the information is being gathered from the sensor nodes in local cluster. The most important part in LEACH cluster head is the way it handles the rotation among the nodes for cluster head elects. Nodes have to elect the head by themselves based on the energy remaining in the nodes and some given probability calculated individually by each node.

Although LEACH is able to increase the network lifetime, there are still a number of issues about the assumptions used in this protocol. LEACH assumes that all nodes can transmit with enough power to reach the base station (BS) if needed and that each node has computational power to support different MAC (Medium Access Control) protocols. Therefore, it is not applicable to networks deployed in large regions. It also assumes that nodes always have data to send, and nodes located close to each other have correlated data. It is not obvious how the number of the predetermined cluster-head is going to be uniformly distributed throughout the network. Therefore, there is the possibility that the elected cluster heads will be concentrated in one part of the network. Hence, some nodes will not have any cluster-head. Furthermore, the idea of dynamic clustering brings an extra overhead, which may increase the energy consumption.

In [191], a centralized cluster formation version of LEACH has been proposed, where the base station organizes and controls the network. This protocol provides a centralized cluster formation, local processing for aggregation of sensed data and the rotation of cluster heads for every round. These activities are aimed at achieving uniform energy consumption among sensor nodes and maximizing network lifetime. Since, the base station does not have energy constraint, centralized cluster formation methods can be attractive alternatives. In this protocol [191], the cluster formation is formulated as a p-median problem [137], which is one of the well-known facility location problems (Just to clarify a bit here, the p-median problem can be stated very simply, like this: given a set of customers with known amounts of demand, a set of candidate locations for warehouses, and the distance between each pair of customer-warehouse, choose p warehouses to open that minimize the demand-weighted distance of serving all customers from those p warehouses [132]). This algorithm produces better clusters by dispersing the cluster head nodes throughout the network.

3.3.1.2 Threshold-sensitive Energy Efficient Protocols (TEEN and APTEEN)

Threshold-sensitive Energy Efficient sensor Network protocol (TEEN) [14], and

(Adaptive Periodic Threshold-sensitive Energy Efficient sensor Network protocol (APTEEN) [13] are two hierarchical dissemination protocols proposed for real-time application. In TEEN, the authors assume that base station and sensor nodes have same initial energy and base station can communicate directly with each sensor nodes in the network. In this protocol, the sensor nodes sense their environment continuously, but the transmission is done less frequently. As shown in Figure 3.3, the network consists of three communication levels: simple nodes communicate directly with their cluster head and constitute the first communication level; then, cluster head can communicate directly with the base station, or via another intermediate cluster head.

Cluster head sends two parameters to its neighbors, hardware threshold and software threshold - the hardware threshold being the minimum value of an attribute permitting a sensor node to power-on its transmitter and transmit to its cluster head. It permits reducing the number of transmissions by allowing a sensor node to transmit its data if the sensed attribute is in the range of interest. The software threshold reduces the number of transmissions which could have differently occurred when there is little or no change of the sensed attribute.

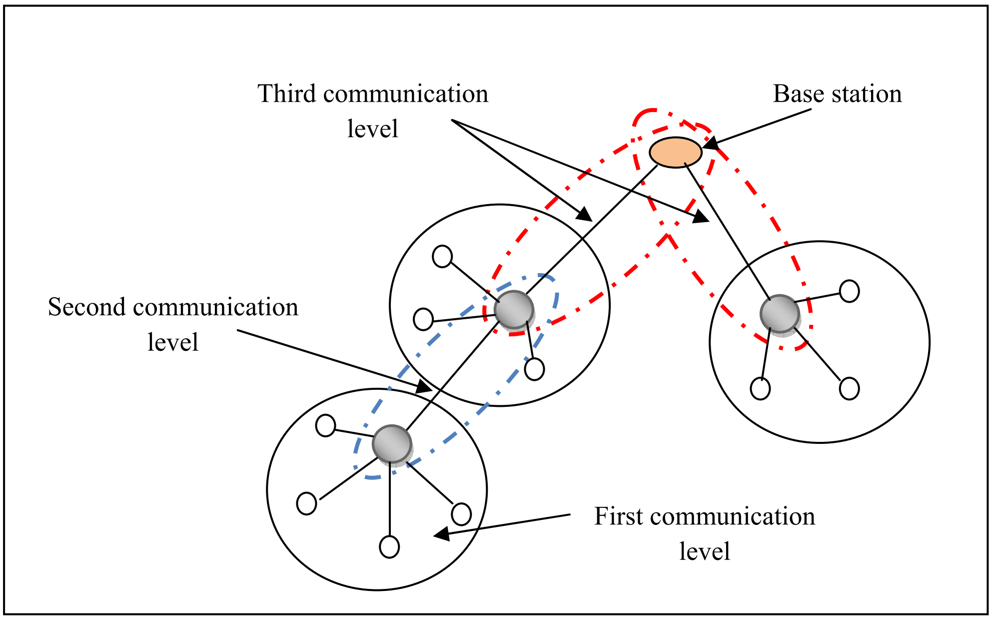


Figure 3.3: TEEN and APTEEN architecture

Based on the two thresholds, data transmission can be controlled and reduced which decreases the energy consumption and improves the effectiveness and usefulness of the receiving data. However, in TEEN, a sensor node may waste its time slot if it does not have any data to transmit. Also, cluster head has to keep its transmitter power “on” to receive data from its members; thus, more energy would be consumed.

APTEEN [13] is an extended version of TEEN. It uses the same network model as of TEEN (Figure 3.3). APTEEN supports both periodic data collection and time-critical situations. After creating the clusters and selection of the cluster heads by the base station in each round, the cluster head sends to its member nodes some parameters concerning the physical parameters; the hard threshold and soft threshold values, the time slot to each node using TDMA and the maximum time period between two successive reports sent by a node. In APTEEN, the cluster head aggregates all the data received from its member nodes and sends it to its higher level cluster head or to the base station which allows reducing the network overhead and the overall energy consumption. Moreover, APTEEN is suitable in both proactive and reactive applications. However, this protocol generates an additional cost and more overhead to organize the sensor nodes in complex multiple levels of clusters.

3.3.1.3 Power Efficient Gathering in Sensor Information Systems (PEGASIS)

In LEACH [194], each node sends the collected data to its cluster-head, which unnecessarily implies a transmission of a great mass of information and it consumes much energy especially if these data are redundant. PEGASIS [170] is proposed to solve this problem and improve the LEACH protocol. In this protocol, each node communicates only with its nearest neighbor and only one node is selected to transmit to the sink which creates a chain communication shape. The chain is constructed in greedy way by assuming that all nodes have global knowledge of the network. The chain construction is started by the farthest node from the sink (node N3) to ensure that nodes farther from the sink have close neighbors. Figure 3.4 shows a chain example (N3, N1, N5, N2, N4) after executing the greedy algorithm. When a node dies, the chain is reconstructed in the same manner to bypass the dead node. Node N5 presents the chain-head and it is responsible to transmit the gathered data to the sink. The chain-head is equitably rotated among the nodes of the chain. Chain-head is selected randomly, and each node has the chance to be the leader once every N round (N is the number of nodes).

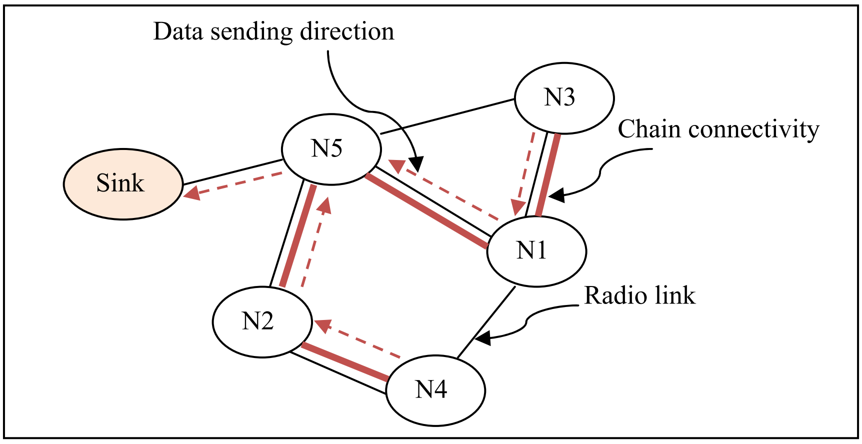


Figure 3.4: Chain construction and data passing approach.

For gathering data in each round, the chain-head sends a control packet to its neighbor to start the data transmission from the end of the chain. In Figure 3.4, node N5 sends this control packet along the chain to node N3. Node N3 will send its data towards node N5. After node N5 receives data from node N1, it will pass the control packet to node N4, and node N4 will pass its data towards node N5 via node N2. Each intermediate node has to aggregate the received data with its local data before sending it to its next neighbor node in the chain. Thus, chain-head sends only one message to the sink by round.

This technique of communication used by PEGASIS allows saving more energy compared to that of LEACH [194] to increase the lifetime of the network and to reduce the bandwidth consumed by using local collaboration between the nodes and by tolerating the failure of the sensor nodes. However, the direct communication between chain-head and sink consumes more energy especially when the distance between them is longer. Moreover, the latency is more important; thus, this protocol cannot be used for real-time applications.

3.3.1.4 Simple Energy-Efficient Routing (SEER)

SEER's [182] data dissemination is *source-based* which eliminates the need for the sink to flood an interest for data through the network. Nodes only transmit data when new data are observed. Data are routed along a single path, which is dynamically established. Every time when a node needs to send data, it selects one neighbor to send the message based on the neighbor's hop count and available energy.

Once the network has been deployed, the sink transmits a broadcast packet with header field of 64 bits. 16 bits are reserved for node, each node in the network is assumed to have a unique address within the network and the header field contains source and destination node addresses. 8 bits are reserved to identify new broadcast messages using a sequence number. The sink increments the sequence number every time it sends a new broadcast message. Nodes store the sequence number locally and forward the broadcast messages only if the sequence number of the message is different from the stored one. The sequence number permit to avoid redundant forwarding of old broadcast messages. An 8 bit hop count ensures that nodes can be up to 255 hops away from the sink.

When a node receives this initial broadcast message, it checks whether it has an entry in its neighboring table for the node that transmitted the message. If not, it adds an entry that consists of the neighbor's address, hop count, and energy level. The node then increments the hop count stored in the message and stores this hop count as its own hop count. It then retransmits the broadcast message, but changes the source address field to its address and the energy level field to its remaining energy level. Every node in the network retransmits the broadcast message once, to all of its neighbors. If a node receives a broadcast message with a lower hop count than the hop count it currently has, it updates its hop count. When this initial broadcast has been flooded throughout the network, each node knows its hop count and has the address, hop count, and energy level of each of its neighbors.

When a node observes new data, it initiates the data dissemination process and specifies in the header of the new message the type of new data (normal or critical), source, destination and creator addresses. A critical message has to transmit to two neighbors instead of only one. The neighbor is selected based on the hop count and the remaining energy. A node searches the neighbor with smaller than or at least the same hop count that it has. If there is only one, this neighbor is selected as the destination for the message. If there is more than one neighbor with a smaller hop count, the node selects the neighbor who has the highest remaining energy. If a node does not have any neighbor with hop count smaller or equal to its own hop count, the message is discarded.

Before the message is sent, the remaining energy entry for the selected neighbor is decreased in the neighboring table. If the message is a critical one, a second neighbor will be selected using the same process. Here, using hop count as the routing metric ensures that the message is always sent to the direction of the sink.

When nodes receive a data message, they update the remaining energy value of the sending node in their neighboring table and forward the data using the same dissemination process; the sending node has to be excluded from the list of the neighboring nodes to avoid any routing loop in the network. When a node's remaining energy decreases than a certain threshold, it transmits an energy message to all of its neighbors to inform them about its energy level.

3.3.1.5 Energy Aware routing Protocol (EAP)

In [124], authors proposed a novel energy-aware routing protocol (EAP) to prolong the lifetime of sensor networks. EAP introduces a new clustering parameter for cluster head election. As LEACH, EAP is divided into rounds, each round begins by a set-up phase in which clusters are organized and routing tree is constructed, followed by a working phase to collect and send data to the sink node. In EAP protocol, each node needs to maintain a neighborhood table to store the information about its neighbors. Each node located in the cluster range is seen as neighbor. At the beginning of each round, each node broadcasts its residual energy (E_r) to its neighbors and setup its state as cluster head candidate. Each node receives the residual energy from all the neighbors in its cluster range. Then, accordingly it updates its neighborhood table and calculates the average residual energy (E_a) of the cluster range and the broadcasting delay time T using the following equations:

$$E_a = \frac{\sum_{i=1}^{n} E_{ri}}{\mathrm{n}}$$

where n is the number of neighbors in the cluster range.

$$T = K \times P \times \frac{E_a}{E_r} ,$$

where K is a real value uniformly distributed between 0 and 1, and P is the time duration for cluster heads election.

During the T time, a sensor waits to receive any proposed cluster head message from its neighbors. If it does not receive any proposition, it proposes itself and broadcasts its proposition to be cluster head to its neighbor nodes. After broadcasting its cluster head proposition, it has to wait $2 \times \Delta t$, where Δt is the time interval which can ensure that all neighbor nodes can receive the cluster head proposition message, to make sure whether there exists another cluster head proposition broadcasted by other nodes in its cluster range. If it does not receive any proposition from its neighbors over Δt, it sets its state as "Head", or else, it compares its weight with the weights of other broadcasting neighbors. If its weight is the largest one, it sets its state as Head and other broadcasting neighbors give up the competition. Otherwise, the node sets its state as member sensor of this cluster.

To reduce energy consumption, EAP adopts the same intra-cluster coverage scheme introduced in [123]. This scheme permits the cluster head to choose randomly m active nodes to ensure a certain required coverage limit. The remaining nodes perform as redundant nodes and turn their radios *off* to minimize energy consumption.

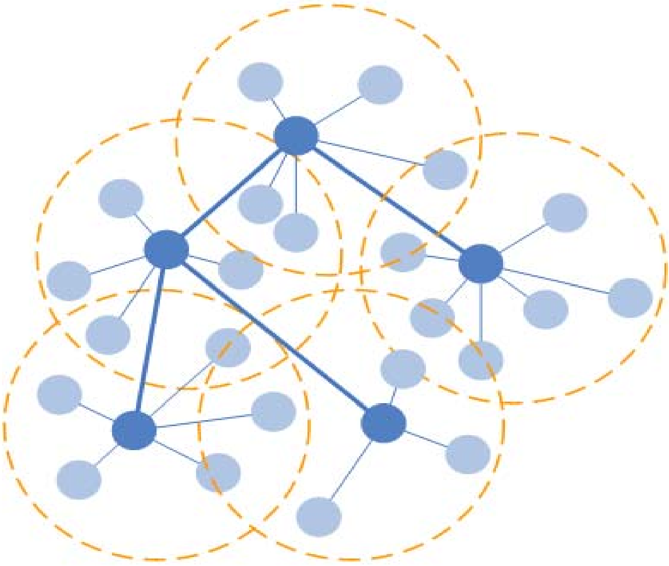


Figure 3.5: EAP architecture.

To define the routing tree after clustering (Figure 3.5), each cluster head broadcasts within a cluster range a weight message, which contains node ID and its weight W defined as below:

$$W_i = \mathrm{D}\ (\mathrm{RSS_i}) \times \frac{\mathrm{E_a}}{\mathrm{D}\ (\mathrm{RSS_{max}}) \times \mathrm{E_r}}$$

where $\mathrm{RSS_i}$ is the node i's received signal strength for the signal broadcasted by the base station, $\mathrm{RSS_{max}}$ is a constant which is determined by the location of the base station, and the function D is used to estimate the distance between node i and the base station. After the deployment of sensors, the base station broadcasts probing message to all sensors and sensors acquire the RSS according to the received signal strength. RSS remains constant during the network lifetime unless base station varies its location or sensor nodes are mobile.

The cluster head compares its own weight and the received weight of the other neighbor cluster heads. If it has smaller weight, it selects the node that has the largest weight as its parent and sends a message to notify the parent node. After a specified time, a routing tree is constructed. The root node has the largest weight among all cluster heads in the same independently connected component. The node that is closer to the base station and located in a sub-region with full energy would be the root node of routing tree due to its higher weight. After routing tree construction, cluster heads broadcast a TDMA schedule to their active member nodes to be ready for data gathering.

The intra-cluster coverage scheme presents the main advantage of EAP – in fact, this technique improves energy consumption and reduces the TDMA schedule overhead. According to the authors and compared with the previous works [191], [136], the selection cluster head technique prolongs the lifetime of the nodes that have low residual energy within the cluster range. Nevertheless, the drawback of this protocol dwells in the high control packet overhead provided during each round which incurs an extra energy and delay cost.

3.3.1.6 Directed Diffusion dissemination protocol (DD)

Directed Diffusion data dissemination protocol [25], [26] is the first proposed data-centric communication protocol for wireless sensor scenarios. The data generated by the producer is named using attribute-value pairs. Figure 3.6 shows the operation of data-centric communication protocol. Directed diffusion is based on query, where sink queries the sensors in an on-demand fashion by disseminating an interest. As shown in Figure 3.6, DD consists of three phases: Interest propagation, initial gradient setup, and data delivery along reinforced path.

In directed diffusion, the data generated by the nodes are named by attribute-value pairs and the data dissemination process is a destination-initiated reactive routing technique in which routes are established when requested.

It creates one if there is no matching interest and a single gradient field is created towards the neighbor from which the interest is received and forwards the requested interest message to its neighbors. A gradient is removed from its interest entry when it expires. A gradient specifies both the data rate as well as the direction in which the events are to be sent. If the interest exists, the timestamp and the duration fields are updated in the entry and the second step starts.

In the initial gradient setup, the sensor node which has a matching interest entry generates event samples and sends an event to all its neighboring nodes for which it has gradients. The last phase begins when the sink starts receiving this event, possibly along multiple paths. The sink then sends a reinforced packet to the neighbor node which is the first one receiving the target data. The neighbor node which receives the reinforced packet can also reinforce and select the neighbor node which can receive the new data first. Consequently, a path with maximum gradient is formed; hence, in future, received data packets can be transmitted along the best reinforced path. Finally, the real data are sent from the source to the sink using the selected path.

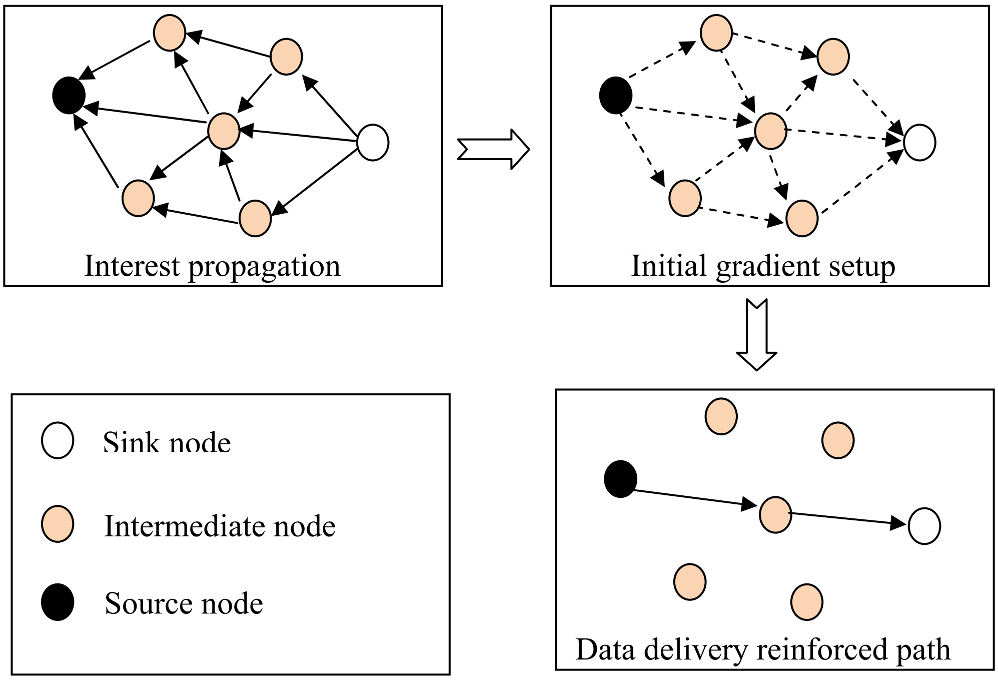


Figure 3.6: Directed diffusion phases.

3.3.2 Multiple static sinks data dissemination protocols

3.3.2.1 A Stateless Protocol for Real-Time Communication in Sensor Networks (SPEED)

SPEED [182] is a real-time communication protocol designed for sensor networks. Speed provides three types of real-time communication services namely: real-time unicast, real-time area-multicast, and real-time area-anycast. These communication types are defined as follows:

- **Real-time unicast:** This is "more to one" communication mode which occurs when one part of a network detects some activity that needs to be reported to a remote base station.
- **Real-time area-multicast:** Contrary to the first communication type, real-time area-multicast is "one to more" communication mode. This type of communication occurs when the base station initiates the communication by sending its query to an area in the sensor network.
- **Real-time area-anycast:** This communication mode can be used when the response of any sensor node is sufficient

SPEED is specifically customized to be a stateless protocol. That means, it only maintains immediate neighbor information and does not require a routing table. SPEED provides a uniform delivery speed across the sensor network to meet the requirement of real-time applications such as disaster and emergency surveillance in sensor networks. To avoid congestion, SPEED uses a novel backpressure re-routing scheme to re-route packets around large-delay links with minimum control overhead. It also uses non-deterministic forwarding to balance each flow among multiple concurrent routes.

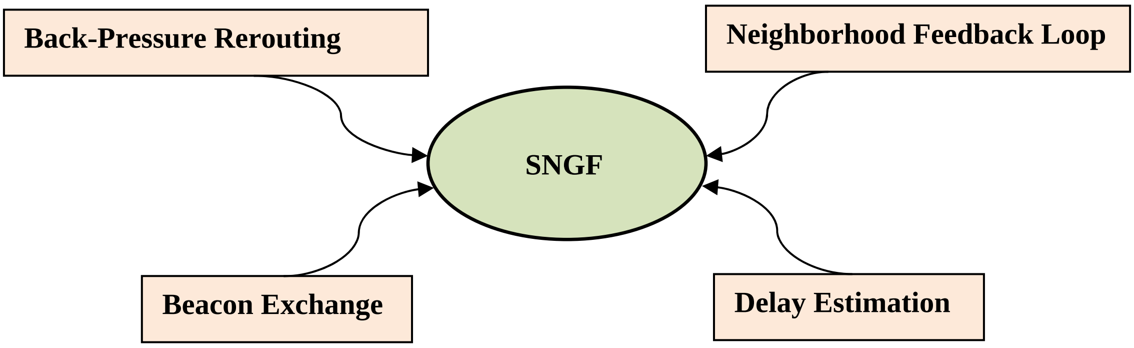


Figure 3.7: SPEED modules

The routing module in SPEED is called Stateless Non-deterministic Geographic Forwarding (SNGF) and it works with four other modules at the network layer [175]. Figure 3.7 shows these different modules.

The beacon exchange mechanism is used to collect information about nodes and their locations. Delay estimation at each node is made by calculating the elapsed time when an ACK is received from a neighbor as the response to a transmitted data packet. SNGF scheme selects nodes that would meet the speed requirement by estimating delay values.

In case no such node is found, relay ratio of the nodes is calculated. Neighborhood Feedback Loop (NFL) module is responsible for providing relay ratio of a node, which is fed to the SNGF module. The relay ratio of a node is calculated by looking at the miss ratios of its neighbors that could not provide the desired speed. The packet is dropped if the relay ratio is less than a randomly generated number between 0 and 1. When a node fails to find a next hop node, the backpressure-rerouting module is finally used to prevent voids and to eliminate congestion by sending messages back to the source nodes so that they would pursue new routes. In comparison to Dynamic Source Routing (DSR) [35] and Ad-hoc on-demand vector routing (AODV) [23], SPEED performs better in terms of end-to-end delay and miss ratio. SPEED reduces transmission energy consumption, control packet overhead, and traffic distribution. It is also able to achieve load balancing in the network to a great extent. SPEED, although is a successful real-time WSN routing protocol based on simple routing algorithm, it is not really energy efficient. SPEED uses only one delay threshold overall to manage transmission of data packets at the highest transmission velocity. As a result, it cannot satisfy different requirements for transmission delay and causes huge energy consumption. The protocol indeed results in energy exhaustion of nodes quickly because it selects nodes having high transmission velocity without considering the remaining energy of nodes. Therefore, for a more realistic understanding of SPEED’s energy consumption, there is a need for comparing it to a routing protocol, which is energy-aware.

In addition to these issues, the idea of per-flow reservation appears to be non-scalable in SPEED due to the highly dynamic links and route characteristics. So, SPEED might not be scalable well for large WSNs. There is an extension of SPEED, called FT-SPEED [100], which is proposed to handle the void problem caused by high sensor failure probability in WSN. In FT-SPEED, a “void announce” scheme is designed to prevent packets from reaching the void through other routing paths. It also introduces a void bypass scheme to route the packets around two sides of a void to guarantee that the packets are delivered rather than just being dropped.

3.3.2.2 Multi path Multi SPEED (MMSPEED)

Multi-path and Multi-SPEED Routing Protocol (MMSPEED) [42], an extension of SPEED is designed to support multiple communication speeds, which provides differentiated reliability. A key feature of MMSPEED is that it addresses both real-time issue and reliability separately. The main goals of MMSPEED design are:

- Localized packet routing decision without global network state update or a priori path setup.
- Providing differentiated QoS (Quality of Service) options in isolated timeliness and reliability domains.

For the first goal, geographic routing mechanism based on location awareness is used.

Each sensor node is assumed to be aware of its geographical location. This location information can be exchanged with immediate neighbors with "periodic location update packets". Thus, each node is aware of its immediate neighbors within its radio range and their locations.

For the second goal, MMSPEED provides multiple delivery speed options that are guaranteed network-widely. For this, the idea of SPEED protocol [182] which can guarantee a single network-wide speed is used. MMSPEED assumes a few important assumptions:

1. All nodes know their geographical location.
2. Location of the packet destination is known.
3. The underlying MAC protocol allows prioritizing between different classes at least stochastically.
4. Each speed level is mapped onto a MAC layer priority class.

Associating messages with deadlines focuses on the problem of providing timeliness guarantees for multi-hop transmissions in a real-time sensor application. In such application, each message is associated with a deadline and may need to traverse multiple hops from the source to the destination. Message deadlines are derived from validity of the accompanying sensor data and start time of the consuming task at the destination. The protocol reduces deadline misses by scheduling message based on their per-hop timeliness constraints. It supports a probabilistic QoS guarantee by provisioning QoS in two domains -Timeliness and Reliability. QoS differentiation in timeliness is provided through multiple network-wide packet delivery speed guarantees. The scheme employs localized geographic packet forwarding augmented with dynamic compensation, which compensates for local decision inaccuracies as a packet travels towards its destination. The intermediate nodes can lift speed level if they find that the packet may miss the delay deadline with current speed but may meet it at a higher level. To reduce the number of collisions, the QoS has been enhanced in [192] by adapting the Contention Window Adapter (CWA) mechanism in which a dynamic contention window has been used.

In supporting service reliability, probabilistic multi-path forwarding is used to control number of delivery paths based on the required end-to-end reaching probability. In this scheme, each node in the network calculates the possible reliable forwarding probability value of each of its neighbors to a destination by using the packet loss rate at the MAC layer. According to the required reliable probability of a packet, each node can forward multiple copies of it to a group of selected neighbors from the forwarding neighbor set to achieve the desired level of reliability. These mechanisms for QoS provisioning are realized in a localized way without global network information, which is desirable for scalability and adaptability to large scale dynamic sensor networks.

Though, MMSPEED [42] does some improvements over SPEED and differentiates among different real-time levels, it does not dynamically adjust routing paths according to the node's energy state. Both SPEED and MMSPEED have a common deficiency that is they do not take into account the energy consumption metric. This metric has been considered by EAMMSPEED protocol [174] which tries to balance the load and energy consumption of individual nodes in the network and improve the overall network lifetime. Therefore, each node makes routing decisions based on the following four parameters: geographic progress towards the destination sink, required end-to-end total reaching probability, delay, and residual energy at the candidate forwarding node. The performance evaluation shows that EAMMSPEED protocol provides stable service in the sensor network and maximizes the lifetime of the entire network while maintaining the QoS guarantees provided by MMSPEED.

3.3.2.3 Sequential Assignment Routing (SAR)

SAR [93] is the first protocol for WSN-oriented QoS. SAR calculates multiple paths from the source nodes to the sink, by building trees rooted from a 1-hop neighbor from the sink (Figure 3.8) and growing outward until it reaches leaf nodes while avoiding paths with low energy or low QoS guarantees. At the end of this process, each leaf node would belong to multiple trees and thus, would have multiple paths to reach the sink.

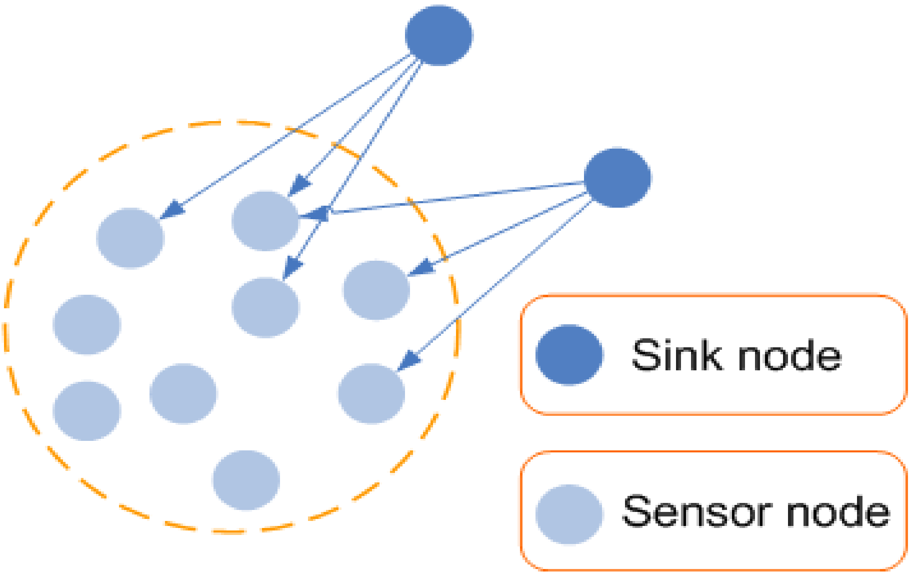


Figure 3.8: SAR Architecture.

For each node, two parameters are associated with each path:

1) Energy resource estimated by the maximum number of packets that can be routed before all of the energy is depleted.

2) Additive QoS metric, where higher metric implies lower QoS.

Each node generating packets makes a decision about which path to choose. This decision is based on the energy resource and a weighted QoS metric which is the additive QoS metric multiplied by a weight coefficient associated with the priority level of the packet.

SAR shows an optimized performance focusing on lowering of the energy consumption of each packet without considering its priority. A routing table update revolves around the network so as to update all the routing tables of the network in order to find out the depleted nodes in the network and ignore any further communication through the ruined path.

The objective of the SAR algorithm is to maximize the lifetime of the network while minimizing the average weighted QoS metric. One of the drawbacks of this protocol is the high overhead due to the large number of tables being kept on each node, especially when the number of nodes becomes huge.

3.3.2.4 Hierarchy-Based Multipath Routing Protocol for Wireless Sensor Networks (HMPR)

In HMRP [206], sensor nodes are assumed to be fixed for their lifetimes, and the identifier of sensor nodes is determined a priori. Additionally, these sensor nodes have limited processing power, storage and energy, while sink nodes have powerful resources to perform any task or communicate with the sensor nodes.

HMRP is based on the hierarchical tree architecture, in which the sink nodes serve as root nodes. Each sensor node must be a member of the architecture. The protocol has two phases: Layer Construction Phase (LCP) and Data Dissemination Phase (DDP).

In the first phase, HMRP forms hierarchical relations by broadcasting a network construction packet (NCP) to all its neighbors. This packet contains *Seq_Number*, *Hop_Count*, *Source_ID*, *Sink_ID*, *Packet_Type*. The sink node initiates the Hop_Count by one, updates the other fields and broadcasts the packet with Layer Construction Request type to discover the one hop nodes. Each sensor node that receives this packet compares the *Hop_Count* field with its hop value in its routing path formation table. If *Hop_Count* field is smaller than its own hop value, then it keeps the packet during some period and updates its routing path formation table, else it drops the packet. If the time of the period duration is finished, the node selects the *Source_ID* of the received packets with the lowest *Hop_Count* values as its candidate parents. If the node receives more packets with the same lowest *Hop_Count*, it saves all *Source_ID* of the received packets as its candidate parents. This node then updates the *Hop_Count* and the *Source_ID* fields in the packet and rebroadcasts it again. Every node continues flooding the Layer Construction Request type packet until the network level is constructed.

In the second phase, Sensor nodes can start disseminating the sensed data to the sink via the parent node. A Received Data Acknowledge (RDACK) packet is sent when the data packet is successfully transmitted to the parent node. The parent node then replies with this packet to notify the source node, and forwards the data packet to next hop. In case of several parents, the source node chooses the parent node as next hop using Round Robin Scheduling when it wishes to send a data packet to a sink. When the source node receives an ACK from the selected parent, it moves this record of this parent in its routing path formation table to the last position and transfers the data packet to the next parent. If no ACK reply is obtained from the parent during some period of time, the source node deletes the record of the concerned parent from its routing path formation table.

The main advantage of HMRP is that the sensor node needs only to know to which parent node to transfer, without maintaining the whole path information. This can reduce the overhead of sensor node. Furthermore, HMRP supports multipath data forwarding path which distributes the energy and prolongs the lifetime of network. However, this protocol has some weaknesses like using an ACK to notify the reception of each data packet increases the network overhead and consumes energy too. This information can be recorded from MAC layer. Moreover, HMRP supports multiple sink nodes scenario, but it does not specify any sink node management procedure - in fact, sink nodes work without any coordination among them, and thus, it has an impact on the overall network performance.

3.3.2.5 Sinks Accessing data From Environment (SAFE)

SAFE [169], [168] is a data dissemination protocol for wireless sensor networks. In this protocol, sensor node can disseminate its sensed data to sinks that explicitly present their interests by sending data requests. Each data sink is allowed to specify its own desired data update rate. SAFE has two major phases: query transfer and dissemination path setup.

In query transfer phase, user sends -via sink node- its query specifying the location, the sensor data type, the desired data update rate, and the service duration. Every node maintains a recent query table and a data management table. The query table records the most recent queries that have been received, and the data table keeps the status of sensor data being or to be distributed by the node. Each node that receives the query performs the tasks as noted below:

- Check the query table if the same query has recently been dealt with. If so,
- Ignore the new query, Otherwise,
- Save the query into its query table

When the node is the data source, it sends a *PathSetup* message to the inquiring node via unicast. If the node is not the source but on a dissemination path, which is called a junction node, it sends a *JunctionInfo* message to the sink via unicast. When the node is neither the data source nor a junction, it forwards the query to the next hop, as long as it is not farther away from the queried location than the previous hop node. The hop sender information might be extracted from the packet header filled by the routing protocol in use, or injected by this data dissemination protocol before forwarding a query.

In the second phase of dissemination path setup, each intermediate node has already inserted the necessary information in its data management table while receiving the *PathSetup* message during the last phase. The intermediate node sets a timer for waiting an *Ack* message from its descendant, which confirms the path is activated. When the sink node receives the *PathSetup* and the *JunctionInfo* messages, it waits for a certain amount of time and then subscribes to the node that sent the best feedback until then. If the best one is a junction node, the sink sends a Subscribe message to this node. When junction node receives this message, it sends a *TrailSetup* message to that sink and establishes the dissemination path. Otherwise, when the source is eventually the best subscription point, the sink sends an *Ack* to its progenitor and every progenitor acknowledges its progenitor in turn until the source gets an *Ack* message and establishes the dissemination path.

3.4 Mobile sink data dissemination protocols

Mobile sink wireless sensor network has recently attracted a lot of attention from the research community. Recent works [159], [10], [8], [108], [78], [196], [39] have shown that the use of mobile sink can enhance connectivity and lifetime of WSNs. Mobility has been proposed as an alternative way in the literature for reducing the communication distance between sensor nodes and sinks. Network lifetime can be improved with mobile sinks by reducing multi-hop communication and avoiding the bottleneck problem, which appears on the nodes close to the static sink.

In wireless sensor networks, mobility can appear in three main forms [49]: mobility of the sensor nodes that sense the environment and transmit the sensing data, mobility of sinks that gather the information from the network and forward data to the applications, and mobility of the observed event.

Sinks can adopt mobility schemes according to the nature of WSN application and its requirements. This mobility can be classified into three categories [159]: (i) Uncontrolled or Random Mobility [51], [33], [86], [52] (ii) Predictable Mobility [39], [87], [2] and Controlled Mobility [159], [10], [108], [40].

3.4.1 Single mobile sink data dissemination protocols

3.4.1.1 Congestion Avoidance Energy Efficient routing protocol CAEE)

In [120], the authors tried to solve the problems of data loss due to congestion around the static sink, and the high energy consumption of the sensor nodes located in the vicinity of the static sinks. Therefore, the authors present a routing protocol that is based on an *in-network* storage model [121], and a mobile sink. In an environment where the sensor nodes are uniformly but randomly deployed, the authors assume the following points:

- Sensor nodes are grouped into clusters, where cluster head is designated for each cluster.

- Cluster forwards the received data from neighboring head nodes and the nodes of its cluster towards the sink.
- Depending on the node density and sensor field coverage requirements, the cluster head manages the nodes of its cluster by assigning them “awake” or “sleep” status.
- Each node has a list of all its neighboring nodes.

CAEE is based on discrete sink mobility along a fixed trajectory. In CAEE, Mini-sinks (MS) are created utilizing the in-network storage model [121] along the mobility trajectory of the sink. Each MS (Figure 3.9) is considered as a cluster of sensor nodes managed by cluster head called a data collector node (DC). The DC node receives the collected data from the sensor field and stores it in the MS nodes. The mobile sink periodically visits each MS and retrieves the stored data.

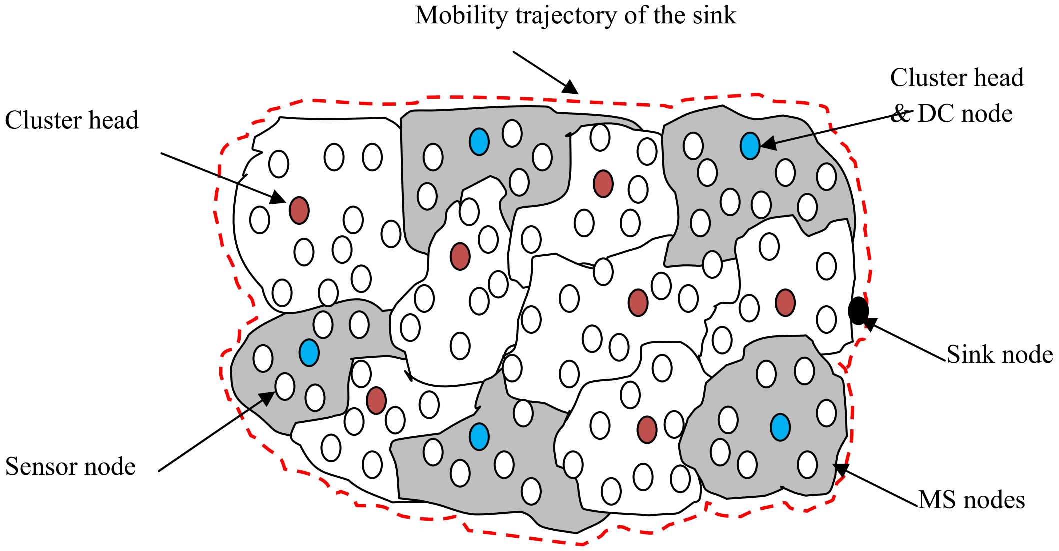


Figure 3.9: CAEE Architecture.

The CAEE protocol does not impose any restriction on the shape of the mobility trajectory of the sink. The mobility path of the sink is along the periphery of the sensor field. During its first trip along the periphery of the sensor field, the sink selects a subset of sensors as DC nodes. Each DC node sets up its MS, and broadcasts this information to the sensor nodes. The sink node starts its first mobility round along the periphery of sensor field to select the DC nodes. It chooses the first or the starting node as DC1 if the last one is cluster head. Otherwise, the sink queries the *start node* about its cluster head node. On retrieval of the required information, the sink assigns the status of DC1 to the obtained cluster head node. Thus, the sink starts its mobility along the periphery of the sensor field, and selects the second data collector node DC2 that is located at least H hops away from DC1. Similarly, the third data collector node DC3 is located at least H hops away from DC2, and so on. In this way, a set of DC nodes are created along the periphery of the sensor field.

To create the MS nodes, each DC node broadcasts a message to invite the sensor nodes to joint its K-hop cluster. The message contains the ID of the DC node and the hop count that is initialized with 1. Each sensor node receiving this message does the following tasks:

- Compares the available routing path to a data collector node with the newly reported route and keeps the shortest one.
- Increments the hop count by 1 in the received message and forwards it.

After a certain period of time, each node knows a shortest possible route to one of the data collector nodes as shown in Figure 3.9.

Each sensor node sends the collected data to the nearest DC node which stores it in one of the buffer nodes of its MS nodes. The mobile sink stops at each MS and requests data transfer from the DC node. The DC node reports the total number of bytes that it wants to transfer to the mobile sink and then the data transfer starts.

In this protocol, the collected data from the sensor node to the MS node are transmitted over the shortest path which increases the lifetime of the network. Also, congestion and data delivery delay have been improved because of the mobility of sink and the multiple MSs. However, this protocol may suffer from latency which can be increased as the number of MSs in the network increases. Hence, this protocol is not recommended for a large scale sensor network and real-time applications.

3.4.1.2 Sink Mobility Protocols for Data Collection (SMPDC)

Sink mobility has been investigated as a method for efficient and robust data delivery in wireless sensor networks [57], [56]. The authors proposed four mobility patterns for the sink, mostly randomized (such as the simple random walk, biased random walks, and walks on spanning sub graphs), as well as predictable mobility (moving on a straight line or cycle). These patterns assume and exploit different degrees of freedom, simplicity and network knowledge. To get data from sensors, the sink movement is combined with three data collection strategies: a passive, a multi-hop, and a limited multi-hop.

The authors consider an environment composed of a huge number of small homogeneous sensor devices with limited capabilities. They suppose that the sensor devices are randomly deployed in a flat square area. The sink does not have any resource limitation, it can calculate accurately its position using Global Positioning System (GPS) and it is aware of the dimension of the network area.

The first proposition called Random Wall and Passive Data Collection (RWPDC) is a simple mobility pattern in which the mobile sink moves randomly towards a chosen direction with constant speed. The mobile sink selects a random uniform angle in $[-\pi, \pi]$ radians. This angle defines the deviation of the mobile sink's current direction. To determine the new position, the mobile sink selects a uniform random distance $d \in (0, dmax)$ which is the distance to travel along the newly defined direction. If the new position is outside the network area, the sink decreases the distance to the network area border. The data are collected in passive manner. The sink broadcasts periodically a beacon message. Each sensor node that receives this message replays by transmitting its data to the sink. RWPDC presents the simplest possible movement, guarantees visiting all sensors in the network, and thus, collects data even from disconnected areas in case of few/faulty sensors or in the presence of obstacles. However, the latency is the biggest problem of this method.

The second proposition is called Partial Random Walk with Limited Multi-hop Data Propagation (PRWLMDP). In this proposition, the authors assume that the network area is partitioned as equal square regions. The center of each region is connected to the center of the adjacent regions. Initially, the mobile sink is positioned on or near one of the center nodes. Then, it calculates the next position by selecting randomly one of the neighbors of the current center. Thus, the sink moves toward this new position with predefined constant speed.

The data collection protocol forms propagation trees. The sink periodically broadcasts a beacon message and indicates the depth of the propagation trees by setting a TTL (Time To Live) parameter. This process creates a number of propagation trees within the network with the roots of these trees being one hop away from the sink. Sensor nodes that belong to propagation tree may begin immediately forwarding their data to the sink.

As the sink moves on, the propagation trees may become disconnected. When the root node loses the communication with the sink, it simply caches all data both generated and relayed, and waits to hear another beacon message to begin the propagation process again.

In this scheme, the distance traveled by the sink is reduced and the time to visit network nodes is accelerated. However, PRWLMDP uses more knowledge of the network which is more expensive in terms of communication and computational costs on the sensor devices.

The third proposition called Biased Random Walk with Passive Data Collection (BRWPDC) extends the previous one (PRWLMDP). The authors use the same assumptions and the same data collection protocol. In this proposition, the sink calculates its next position based on two parameters: the visiting frequency of the region and the number of sensor nodes in the region. The center of the region that has the low visiting frequency and the high number of sensor nodes will be selected as the next position to move toward it. The low visiting frequency is preferred to speed up the coverage of new areas. To increase data delivery in areas with many nodes, the region of high number of sensor nodes is preferred.

The last proposition is called Deterministic Walk with Multihop Data Propagation (DRWMDP), in which the sink's movement is predefined. The trajectory is characterized by its length (L). The authors use a particular trajectory (line or circle trajectory). The linear trajectory consists of a horizontal or vertical line segment passing through the center of the network. The sink moves from one edge of the line to other and returns along the same path. The circle is centered at the center of the network and its radius is defined as $R = \frac{L}{2\pi}$. Initially, the sink is positioned on the perimeter of the circle and continues along this path. In this kind of sink mobility, the authors use a data collection protocol similar to the one presented in the second proposition (PRWLMDP), without the timeout and TTL mechanism, thus paths are created according to minimum hop distance and span throughout the whole network area. The deployment of this protocol imposes a high cost on the sensor devices that perform tree formation and multi-hop propagation. However, it seems that the delivery latency is lower than any of the three previous propositions. Furthermore, the selection of the trajectory length introduces a trade-off between the cost at the sink and the cost at the sensors.

The simulation results show that for applications where time efficiency is not critical and the energy saving is important, it is better to let the sink traverse the whole network area, as given in the first and the third propositions. When the latency is important, the second proposition is more appropriate. For the applications where the mobility capabilities of the sink are limited but can tolerate some loss of information and increased energy consumption, the last proposition is more suitable.

3.4.1.3 Density-based proactive data dissemination Protocol (DEEP)

In [128], the sensing data are proactively distributed and stored throughout the network. The mobile sink is free to choose its own trajectory in any way and at any time. The only condition imposed on the mobile sink in order to retrieve a representative view of the monitored data is on the total number of nodes the mobile sink needs to visit. In DEEP, data dissemination strategy uses a combination of density sensitive probabilistic forwarding with deterministic corrective measures, as given in [188].This technique permits to ensure a predefined average number of transmissions and retransmissions of each message. Based on calculated probability, each node can decide to broadcast any message after receiving it for the first time. If the node does not decide to retransmit the message, it should wait for a given delay and if it does not receive this message from any of its neighbors, then this node retransmits the message. Moreover, in this protocol, the node can store the received message based on another calculated probability.

The simulation results show that DEEP [128] is the more viable solution, especially for sparse networks, when the frequency of sending messages is low, and when the amount of sensed data reported in each message is large. However, the simulation results illustrate that the data storage is well distributed in the network.

3.4.1.4 Data collection with adaptive stopping times (DCAST)

In this protocol, the authors propose biased sink mobility with adaptive stop times, as a method for data collection in wireless sensor networks [11], [12]. The system model contains a single mobile sink and a vast number of ultra-small homogeneous static sensor devices. Each sensor is a fully-autonomous computing and communication device, characterized mainly by its limited power supply. The sensors are deployed randomly in flat square area. The authors assume the existence of some regions, called *pockets*, in the network with high sensor node density. Each *pocket* represents a circular area and does not overlap with another pocket. Moreover, the authors suppose that the mobile sink is not resource constrained. This sink is assumed to be powerful enough in terms of computing, memory, and energy supplies. The sink can accurately calculate its position using GPS and it is aware of the dimensions and the boundaries of the network area. Also, it moves with constant speed according to a given mobility function. The mobility function can be invoked at anytime even before reaching the designated point.

As shown in Figure 3.10, the network area is partitioned, during the network initialization, as equal square regions, called cells. The center of each cell is connected with unidirectional edges only to the four centers corresponding to adjacent cells. Thus, when the sink is located at the center of a cell, it can communicate with every sensor node within the cell area. The sink collects the data in a passive manner and broadcasts beacon messages within the cell. Nodes that receive a beacon start transmitting the data stored in their memory to the sink. Initially, the mobile sink is positioned on one of the central nodes. In the Figure 3.10, two sink mobility schemes proposed by the authors: deterministic walk and biased random walk are represented by the blue-thin and the red-thick arrow lines respectively.

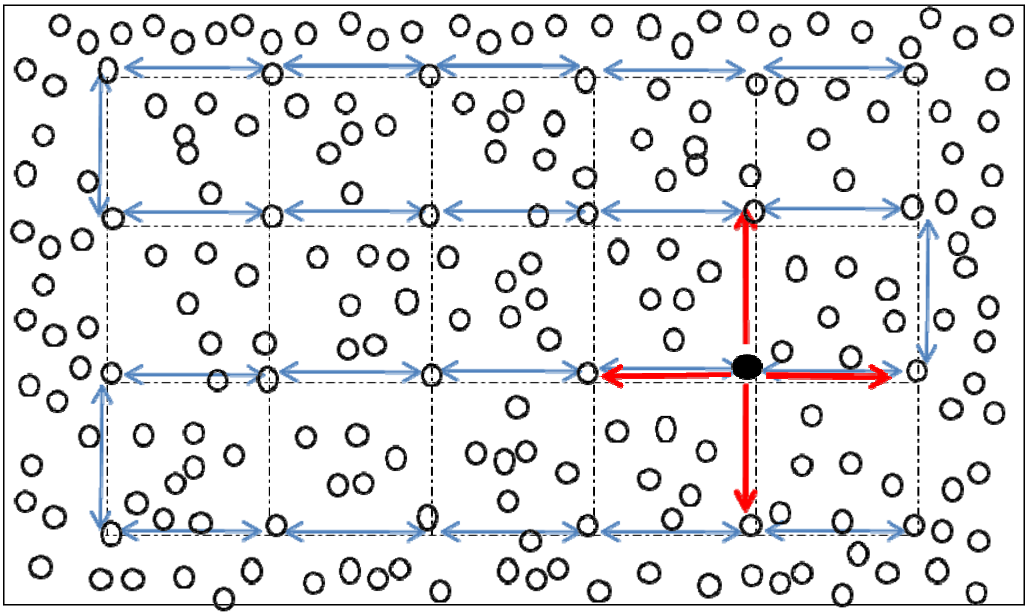


Figure 3.10: CAEE Architecture.

In the deterministic walk, the sink visits cells from left to right and vice versa according to the blue-thin arrow. By moving on this trajectory, the sink can communicate with each node in the network. This walk assumes some global network knowledge to know the boundaries of the cells and the network. It avoids visit overlaps and multi-hop communication, which optimizes the time needed to cover the network. However, in this kind of walk, it may not be feasible to traverse the network with the presence of obstacles that may hinder the movement of the sink. Also, the network topology may not be known to the sink or may change dynamically. To avoid these inconveniences, the authors proposed the second sink mobility scheme (biased random walk) represented by the red-thick arrows in the Figure 3.10.

In this walk, the next position of the sink is determined by selecting the center of one of the neighboring cells. A frequency is associated with each cell - the sink increases this frequency for each visit to the corresponding cell. The selection of the next area to visit is done in a biased random manner depending on this frequency and the less frequently visited regions are favored.

For each mobility scheme, the authors proposed two different types of stopping for the sink. In the Constant stop time, the mobile sink takes a constant and an equal pause time at each cell. In the Adaptive stop time, the sink can leave the cell before the end of the stopping time to avoid spending a lot of time in a cell without collecting data, when the sensors empty their memory before stopping time expires. Also, at each cell, the sink waits for some time after the end of the stopping time to receive the eventual pending data.

3.4.2 Multiple mobile sinks Data dissemination protocols

3.4.2.1 Coordination-based data Dissemination protocol for wireless sensor network (CODE)

CODE [55] considers energy efficiency and network lifetime, especially for sensor networks with high node density. In this protocol, all sensor nodes are stationary except the sinks nodes. The authors assume that each sensor is aware of its residual energy and its location using the Global Positioning System (GPS) [187], [133]. As shown in the Figure 3.11, the sensor network field in CODE is divided into grids. Each grid is indexed based on its geographical location. During the data dissemination process, each grid participates by only one coordinator node. The other sensors remain in the sleeping mode using GAF (Geographical Adaptive Fidelity) protocol [208]. The coordinator acts as an intermediate node to cache and relay data.

CODE has two major phases: query transfer phase and data dissemination phase. For example in Figure 3.11, if an event is detected (grid [1, 0]) the source generates a data-announcement message and sends the message to all coordinators using a simple flooding mechanism. Then, the interested sink sends a query (query transfer phase) to the source node along the path [2, 2] → [1, 1] → [1, 0] which will be used to transport the sensing data during the data dissemination phase. However, the sink checks its geographical location periodically. If the sink moves out of the grid (from [2, 2] to [2, 1]), it has to send a message to remove the previous data dissemination path and then re-sends a query to set up a new one ([2, 1] → [1, 0]).

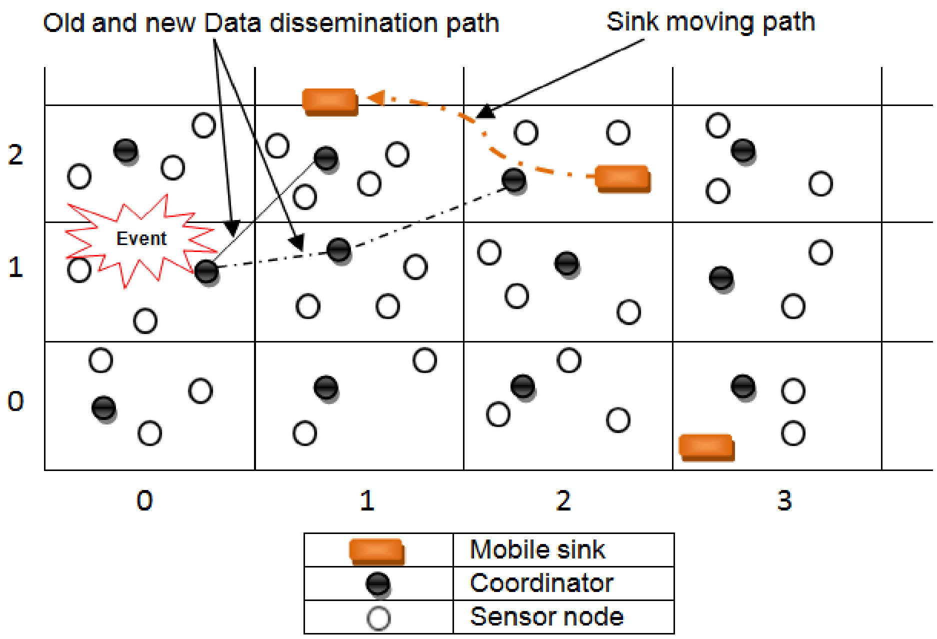


Figure 3.11: CODE Architecture.

CODE establishes a better data dissemination path based on the grid ID without flooding and additional phase. The sinks do not need to periodically propagate their geographical location to the sources. Moreover, CODE takes into account query and data aggregation to reduce the amount of data transmitted from multiple sensor nodes to sinks. However, the random sink mobility presents the major inconvenience of this protocol. The mobile sink can move at anytime, goes away from the source node. Thus, it may increase the latency and energy consumption.

3.4.2.2 Two Tier Data Dissemination protocol (TTDD)

This protocol [51] provides data delivery to multiple mobile sinks based on a decentralized architecture. TTDD uses homogeneous sensors and assumes that each sensor is aware of its own location using GPS [187], [133]. TTDD is grid-based structure. A virtual grid should be created at any new sensed data by the source node. As shown in Figure 3.12, when the source (S) senses new data, it calculates the locations of its four forecasted neighboring dissemination points (D) based on its geographical position and cell size. The source node then sends a data-announcement message to these neighbors to select the real four dissemination nodes (D). Each dissemination node resends the data-announcement message with the same manner until the construction of the virtual grid as shown in Figure 3.12.

Instead of broadcasting the location information of mobile sinks to all sensor nodes, TTDD uses a two-tier data dissemination model to deal with the sink mobility problem and reduces energy consumption.

Only sensors located at a cell boundary need to forward the data. The sink proactively builds the two-tier grid structure throughout the network and sets up forwarding points in the sensors closest to the dissemination nodes. The lower tier is the cell at the sink's current location and the higher tier contains the dissemination nodes at cell boundaries. The sink broadcasts its query within its own cell. When the nearest dissemination node in the cell receives the query, it forwards it to its adjacent dissemination node in another cell. This process continues until the query reaches the source node or one of the dissemination nodes that have the corresponding data. During the query propagation, the network establishes the reverse path towards the sink so that the data could be forwarded on the same path as that of the query propagation. TTDD exploits local flood within a local cell of a virtual grid which sources build proactively. However, it does not optimize the path from the source to the sinks. When a source communicates with a sink, the restriction of grid structure may increase the length of a straight-line path. Also, TTDD creates new virtual grid for each new data source. It therefore, increases energy consumption and connection loss ratio. Moreover, sink mobility in this protocol has random scheduler like CODE [55] which affects negatively on the network performance.

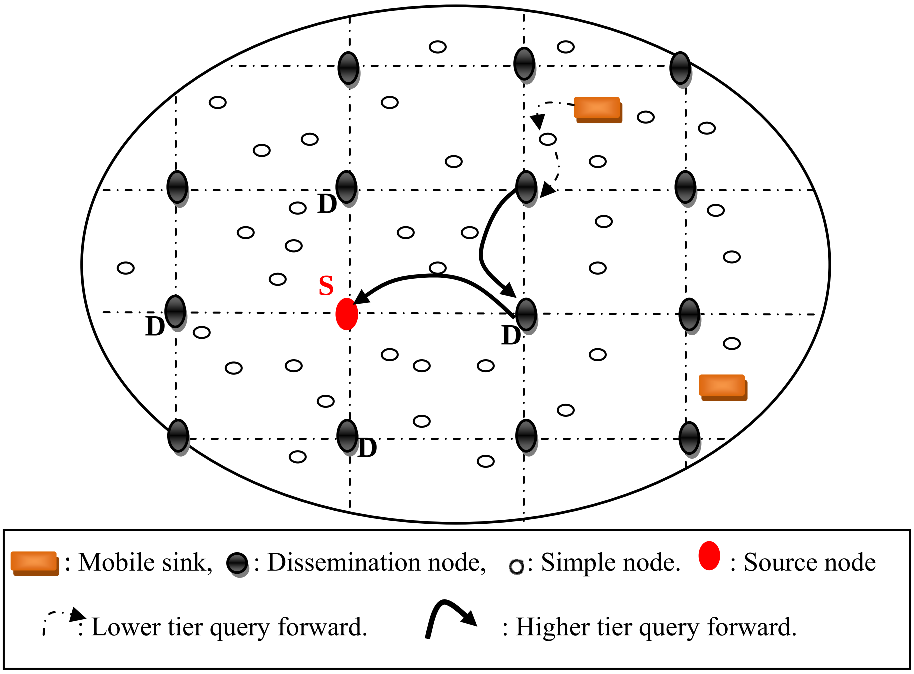
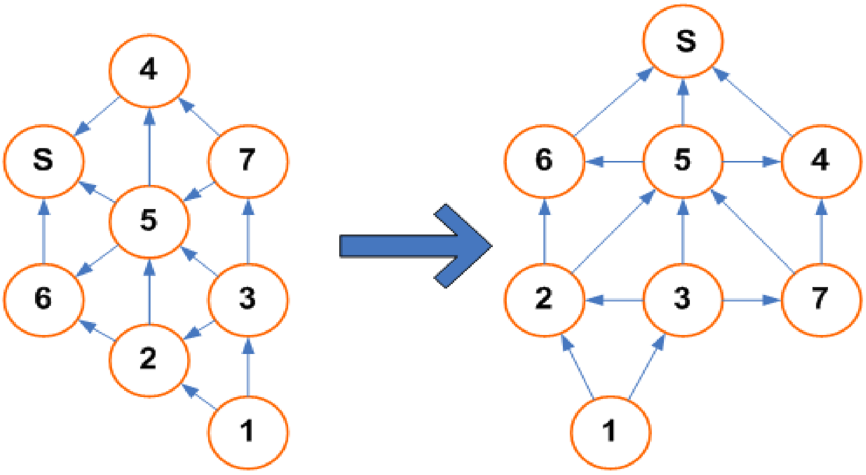


Figure 3.12: TTDD structure.

In the protocol, EGDD (Energy-Aware Grid-based Data) [166], the author tries to overcome the above TTDD’s drawback. In EGDD, the grid is constructed only when no valid grid is present in the sensor field. Also, the dissemination node is selected based on its residual energy. Hence, it can be replaced by another one when its energy becomes equal to the predefined energy threshold. Moreover, EGDD network model ensures query and data forwarding through the shortest path between source and sink. However, sink mobility in this protocol is uncontrolled which brings other challenges for this protocol.

3.4.2.3 Pseudo-Distance Data Dissemination protocol (PDDD)

In PDDD [122], network partitioning is not considered and mobile sink nodes are assumed to have unlimited battery power. Also, it is assumed that the links between sensor nodes are bidirectional and no control messages are lost. The main idea of this protocol is to create and maintain a Totally Ordered Graph (TOG) using pseudo-distance. As depicted in the Figure 3.13(a), when a sink node (S) wants to collect data from sensor nodes, it broadcasts an interest message. By receiving this message, each node can set its pseudo-distance and corresponding level from the sink, and then it broadcasts the received interest message to its neighbor nodes with its own level metric. At the end of this operation, the hierarchical levels of communication are created (figure 3.13 (b)). Thus, each sensor node uses this TOG to disseminate the requested data.



(a) (b)

Figure 3.13: PDDD architecture.

Mobile sink nodes generate periodical heartbeat messages to their direct neighbors. Therefore, if the mobile sink nodes move, the direct sink's neighbors can detect the sink mobility by losing the heartbeat messages. For the other sensor nodes, PDDD uses ACK packet. Each sensor node that transmits data packets to its next hop should receive an ACK from the latter one. If it is not the case, then the link is considered as *failed* link. Therefore, the sensor node has to choose another redundant path. However, if a node loses all of its parent nodes, then it has to update its own level locally to find new parent.

This protocol can achieve an acceptable data dissemination level in stable sensor network. However, PDDD needs many control messages (interest, heartbeat and ACK messages) to create and maintain the TOG graph. The number of control messages increases when the number of mobile sinks increases. Thus, this fact affects directly and negatively on the network performances like latency and energy consumption. Also, PDDD did not consider the energy parameter in its data dissemination process. Some nodes may be more soliciting than others which would accelerate the death of these nodes. Moreover, like CODE [55] and TTDD [51], PDDD did not give any strategy of sink mobility which is still random and uncontrolled.

3.4.2.4 Topology-based rendezvous data dissemination (TRDD)

This protocol [165] assumes a network model of three tiers. Sensor nodes are deployed randomly in the lower tier, each sensor is assumed to be aware of its geographical position using GPS [187], [133]. Sink nodes are placed randomly on the periphery and they constitute the middle tier. The higher tier represents the administration site. TTDD consists of two phases: the events propagation phase and query propagation phase. The dissemination structure of TRDD is based on a simple geometric idea. It considers the network perimeter as a polygon that provides a closed region for the interior nodes called query region (QR). TRDD adapts a modified version of the algorithm presented in [151] to identify dynamically the outer boundary nodes. As shown in Figure 3.14, the construction of the QR starts once the sinks receive the query from the management site and after discovering the boundary nodes. Thus, QR contains the sensor nodes of the network perimeter and their one-hop neighbors.

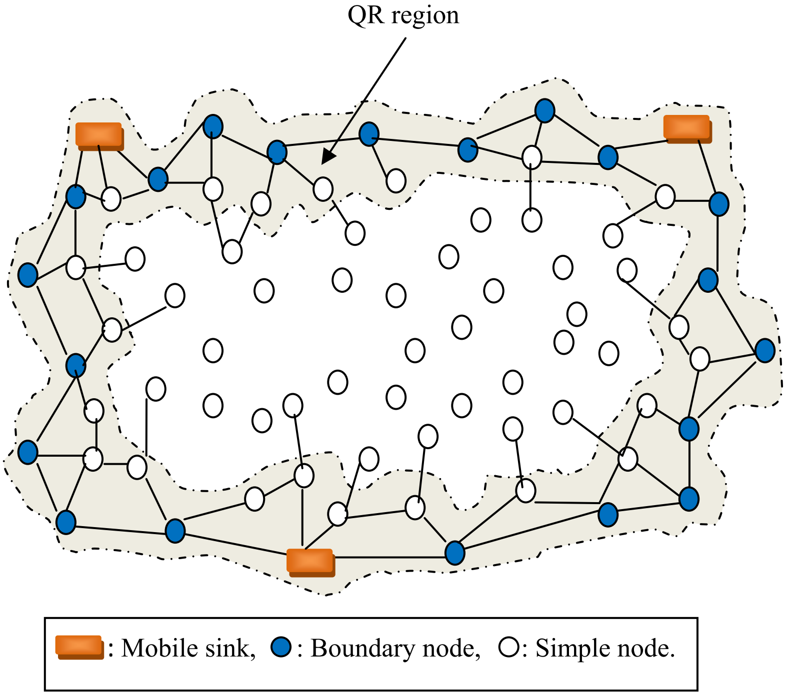


Figure 3.14: TRDD architecture.

When the sensor nodes of one-hop neighbors of the boundary nodes in the QR region receive the query packet, a beacon packet has to be transmitted by those nodes. This beacon allows interior nodes located outside QR to update their neighbor table, to be informed about the creation of the QR region, and to start sending their sensing data toward the QR. After selecting a direction's path (TRDD proposed eight possible directions) based on three policies: Random-based, Round Robin-based or centroid-based, sensor node selects its closest next hop in this direction. In the first policy, sensor node randomly selects one direction. In the second policy, sensor node selects its direction in order after selecting the first one randomly.

In the last policy, sensor node calculates its position relative to the virtual gravity center of the network and selects the contrary direction relative to this center. Intermediate nodes use the same direction chosen by the initiator node except in failure case, where the intermediate nodes should change the direction. Thus, the sensed data will intersect the QR region and will be transmitted in the reverse path to the root sink.

In TRDD, the authors consider and evaluate two sinks mobility patterns, random and controlled mobility. In the controlled mobility, sinks move along the network diagonal or along the network peripheral.

In TRDD, the authors consider and evaluate two sink mobility patterns: random and controlled mobility. In the controlled mobility, sinks move along the network diagonal or along the network periphery.

According to the simulation results presented in [165], the diagonal and the random mobility produce extra costs than that of the peripheral mobility. Moreover, TRDD generates low communication overhead in comparison with TTDD [51]. In TRDD, the sink mobility does not need to transmit any additional control packet contrary to TTDD. However, detecting the boundary nodes and creating the QR may add an additional cost. Also, in TRDD when sensor node detects an event, it creates and stores the new data locally until it receives the beacon packet of the query. Thus, this may influence negatively on the latency and the storage capability of the nodes, especially if the query comes with more delay.

3.5 Summarization

In this study, several previous works on data dissemination issues have been analyzed. Most of these works have proven that their methods improve the data dissemination process in terms of energy usage, network lifetime, and reliability. Some protocols also address the issue of being real-time. Our study has been principally based on two main parameters: the number of sink and the type of sink (stationary or mobile). We have investigated how these parameters affect the data dissemination process. Table 3.1 and Table 3.2 show general overviews of the data dissemination and sink mobility methods of the previously proposed protocols. The character '*slash*' (/) in the table means no information is available or not applicable. These tables present respectively the data dissemination protocols with single sink and multiple sinks.

Table 3.1: Data dissemination protocol with single sink.

Protocol	**Dissemination scheme**	**Number of sink Single / Multiple**	**Sink type Static / Mobile**	**Sink mobility scheme**
LEACH [194], [191]	- Set up sensors into cluster nodes. - Elect periodically a cluster head for each cluster nodes. - Sensor nodes send their data directly to their cluster head using TDMA schedule. - Cluster head aggregates and sends the data to the sink directly or via a super cluster head using multi hop communication	One	Static	/
CAEE [121]	- Set up sensors into cluster nodes. - Choose some cluster heads as data collector nodes (DC), each Dc node creates its mini sinks (MS) nodes group. - Sensor node sends the collected data to the nearest DC node which stores it in one of the buffer nodes of its MS nodes	One	Mobile	- The sink moves along the periphery of the sensor field. - In the first trip, the sink selects the data collector nodes - The mobile sink periodically visits each MS and retrieves the stored data.
TEEN [14]	- Three communication levels, sensor node sends its data to its cluster head, cluster head sends the data to the sink via a second cluster head.	One	Static	/
SMPDC [57], [56]	- Sink broadcasts periodically a beacon message. Each sensor node receives this message replays by transmitting its data to the sink. - Sink periodically broadcasts a beacon message and indicates the depth of the trees or the data dissemination path.	One	Mobile	- Three mobility schemes: - Sink moves randomly towards a chosen direction with constant speed. - Sink visits only the centre nodes, the next centre node chosen randomly or, - Based on the visiting frequency of the region and the number of sensor nodes in the region.
PEGASIS [170]	- Create a communication chain. - Sensor node sends its data to its nearest neighbor. - Only one node (chain head) sends data to the sink. - Chain head selected randomly and equitably rotated among the nodes of the chain.	One	Static	/

(Continued)

Table 3.1: Continued.

Protocol	Dissemination scheme	Number of sink Single / Multiple	Sink type Static / Mobile	Sink mobility scheme
SEER [182]	- Event based. - Sensor node sends its data to one next neighbor. - Next neighbor is selected based on the hop count (number of hop for this neighbor to the sink) and the available energy. - Hop count and energy of each neighbor are learned during the initialization phase when the sink broadcast the path construction packet.	One	Static	/
DEEP [128]	- Sensed data is disseminated based on density sensitive probabilistic forwarding with deterministic corrective measures	One	Mobile	- Sink moves towards any destination at any time and in any way.
EAP [124], [123]	- Set up sensors into cluster nodes and elect periodically a cluster head. - Cluster head elected based on its weight calculated based on its energy available and the signal strength for the signal broadcasted by the base station. - Cluster heads constitute the data dissemination path, next cluster head selected based on its weight.	One	Static	/
APTEEN [13]	- Use the same model as TEEN. - Support both periodic data collection and time-critical situations.	One	Static	/
DCAST [11], [12]	- Network area is partitioned in equal square regions. - Sensor node sends its storage data in passive manner.	One	Mobile	- Two mobility schemes: - Sink visits regions from left to right and vice versa in deterministic walk. - Sink selects the less frequently visited regions among the adjacent regions
DD [25], [26]	- Sink broadcasts its interest to establish the data dissemination path. - The node that has the interest sends back an event to the sink. - The sink reinforces the path from which the event is received - The interested data is disseminated using the reinforced path.	One	Static	/

Table 3.2: Data dissemination protocol with multiple sinks.

Protocol	Dissemination scheme	Number of sink Single / Multiple	Sink type Static / Mobile	Sink mobility scheme
SPEED [182]	- Paths are built using least cost algorithms. - Next hop is selected based on the data transmission speed and miss ratio. - If the required does not found, the message has to send back to the source nodes	Multiple	Static	
MMSPEED [42]	- Each message is associated with delivery deadline. - provides multiple delivery speed options - Uses multiple paths to transmit data.	Multiple	Static	
CODE [55]	- The sensor network field is divided into grids. - Source node sends data announcement message. - The interested sink sends query to the source node and creates the data dissemination path. - Source node sends the data to the sink using the data dissemination path.	Multiple	Mobile	- Sinks move randomly . - Sink check periodically its position. - If sink changes its position the previous data dissemination path has to remove and new one has to set up.
SAR [93]	- Multi Hop, Trees are constructed either from node to sink or sink to node. - Data dissemination path is chosen based on the energy resource and a weighted QoS metric	Multiple	Static	
TTDD [51]	- A virtual grid has to create at any new sensed data by the source node. - Source node sends a data announcement message to four selected neighbors. - The interested sink builds the two-tier grid structure and sink broadcasts its query. - When the query reaches the source, the data will be forwarded to the sink using the data dissemination path created during the query forward.	Multiple	Mobile	- Sinks move randomly.
HMPR [206]	- Muli hop hierarchical tree is constructed, - Sensor node keeps only the nearest next hop to its sink. - Multipath data forwarding path is used in case of several next hop nodes.	Multiple	Static	

(Continued)

Table 3.2: Continued

Protocol	Dissemination scheme	Number of sink Single / Multiple	Sink type Static / Mobile	Sink mobility scheme
PDDD [122]	- When sink sends its query, a Totally Ordered Graph (TOG) using pseudo-distance is created. - Sensor node uses this TOG to disseminate the requested data.	Multiple	Mobile	- Sink sends periodical heartbeat messages to permit its direct neighbors detect its mobility
SAFE [169], [168]	- Sink node initial the data dissemination path on demand by sending its data query. - Each sensor node has the requested data replays the sink. - Sink node chooses the best data dissemination path.	Multiple	Static	
TRDD [165]	- A query region has to create on the network perimeter, it contain the border nodes and their one hop neighbors. - The interior nodes send their sensed data toward the query region.	Multiple	Mobile	- Sinks move within the query region.

3.6 Conclusion

In this chapter, we have studied the recent significant research results on data dissemination in wireless sensor networks and classified these protocols into two main categories based on the number of sink (single or multiple) and the nature of its movement (static or mobile). Whatever the category in which any data dissemination protocol belongs, network resource like energy consumption still remains the major concern while designing protocols for wireless sensor networks.

In a static sink approach, sensor nodes do not need to know the geographical position of the sink at each time. Usually, sink broadcasts its location information in the network only once, just after the network deployment. Moreover, sensor node keeps no more than one valid path to forward its data toward the sink. Thus, such stability can help improve the network performance and reduce the network overhead. However, in static sink approach, sensor nodes relatively closer to the sink can be loaded with relaying a large amount of traffic from other nodes. This situation results in energy exhaustion at the nodes near the sink too soon, leading to the separation of the sink from the rest of the nodes that still have plenty of energy.

As presented in the second category, many protocols try to exploit sink mobility to improve the lifetime of the network. However, there is still some kind of skepticism in the research community about the practicality of deploying moving sinks in WSN scenarios. One of the major concerns behind this skepticism is that mobility inevitably incurs additional overhead in data communication protocols and the overhead can potentially offset the benefit brought by mobility. Further research works may investigate this particular issue in-depth and analyze how effective a mobile sink could be in comparison with static sink in the network for data dissemination.

Chapter 4

Synchronous MAC Communication and Energy Consumption in Wireless Sensor Networks

4.1 Introduction

Considering the importance of energy parameter in wireless sensor networks, an optimal management of this resource is indispensable, especially when we talk about the MAC layer, the layer that is responsible on the significant part of energy consumption caused by the collision, the idle listening and the overhearing. However, the target of our proposal that will be described during this chapter is to minimize, as soon as possible, the idle listening by making the sensor nodes follow an optimal program of listen/sleep permitting the avoidance of energy loss and keeping the correct functionality of network. To completely avoid the problem of the collision and the hidden host, a specific configuration has been introduced which organizes the sensor nodes under different clusters during the deployment phase of sensor network. The proposed solution has been evaluated by simulation, its performances evaluation, comparing with other solutions, shown that the energy consumption has been improved due to the mode of send/receive/sleep adopted during the operation phase.

The remainder of this chapter is organized as follows. We summarize the recent work on energy consumption and collision avoidance in WSN in addition to existing MAC protocols in section 4.2. In section 4.3, we present the parameters of the using environment. In section 4.4, we develop our solution. Performance analysis and simulation results are presented in Section 4.5 and the chapter is concluded

4.2 Related works

One of the principal problems of the wireless sensors networks is the energy consumption, the battery life must support the maximum time. The principal source of this consumption is the radio operator transmission. Several solutions have been proposed in order to save energy in the wireless sensor networks.

Protocol S-MAC [198], more used currently, made an improvement of consumption of energy, grace to the period listen/sleep which is introduced. The sensor node returns in sleep periodically, which reduces idle listening and minimizes the rate of collision. Protocol T-MAC [183] is carried out over two periods of listen/sleep like protocol S-MAC, the difference is the use of an additional screen FRTS (Future Request To Send) making the sensor node able to remain wake up in the end of data transmission, to receive another data information sent by an another nodes.

Other protocols like B-MAC [79], uses a tone to wakeup sleeping neighbours. In this protocol, sensor nodes independently follow a sleeping schedule based on the target duty cycle for the sensor network. Since the sensor nodes operate on independent schedules, B-MAC uses very long preambles for message transmission. The source node transmits a preamble with enough long, the destination, which periodically senses the channel, has enough time to wakeup and senses activity. Sensor nodes that sense activity on the channel remain awake to receive the message following the preamble or take their sleep if there is no activity on the channel. Before transmitting, sensor nodes delay a random time to prevent synchronization, and sense the channel to prevent a possible collision for an ongoing transmission. This protocol is extended others previous works like [18], [34] and provides a great deal of flexibility through a protocol interface that allows the sensor nodes to change many operating variables in the protocol, such as delay and backoff values. WISEMAC [7] tried to eliminate the most assumptions possible to approach the real environment with the disadvantages which are known in the sensor networks.

4.3 System model and hypothesis

A sensor network is comprised of a large number of limited power sensor nodes which collect and process data from a target domain and transmit information back to specific sites (headquarters, disaster control centre). We consider wireless sensor networks which share the same wireless communication channel. We assume also that the sensor nodes have the same clock time and one sensor is a sink, the other nodes send their event information to this sensor.

4.4 Ring Sensor MAC Protocol (RSMAC)

This protocol that appeared in [109] is based on two principal phases, the first is carried out during the deployment of the sensor networks and it aims to gather sensor nodes under ring group form and to define an operating mode for each node. The second phase starts after the end of the previous phase, so the sensors can start its functionality and follow the send, receive and sleep mode defined below.

4.4.1 Setup phase

It is a preparative phase which held during the network deployment, it consists to set sensor nodes under groups with **K** hop, this phase must be preceded by preliminary discovery procedure making sensor nodes able to know its neighbours. The construction clustering process starts by a specific node, chosen by the administrator, generally is the sink node, we call him a beginner or lancer node, the specific sensor sends a configuration message containing the group ID and the hop parameter H initialised with zero value (H=0).

The **Init ()** function below, must executed by each sensor, but only the Beginner node sends the Create group (Creat_grp) message.

Init ()

```
{ MyId_grp=0 ;
  Mygrp_Nbr=0;
  Nextgrp_Nbr=0;
  H=0;
If (node= = beginner_node)
{ Dest = node_neighbor; //Select the first neighbor
   MyId_grp = 1;
   My_turn=1;
   Mygrp_Nbr++;
   Send Creat_grp (MyId_grp, node, dest,
      My_turn, Mygrp_nbr, Nextgrp_Nbr, H);
}
}
```

The **Creat_grp** message is sent to all neighbours one by one. When the neighbour node receives this message increments the parameter H (H++) of one and proceeds to send it to its neighbours.

The message can be comeback to the predecessor node, by sending a new message **End_creat_grp,** in cases below: when the K hops have been reached (H=K), in the second reception of the same create group message and when the sensor has only one neighbour.

Creat_grp (Id_grp, source, dest, send_turn, grp_nbr, Nextgrp_Nbrm, H)

```
{
 If (node_ID== dest)
 {My_predecesor= source;
  if (MyId_grp= =0)
  { H++;
    MyId_grp= Id_grp ;
    If (Id_grp= =1)
```

```
      {grp_Nbr = grp_Nbr+1;
         } else nextgrp_Nbr = nextgrp_Nbr+1;
      My_turn= send_turn+2;
        send_turn= My_turn;
If (H< K)
  {if (exist neighbor ≠ source)
      {dest= neighbor;
         Send Creat_grp(MyId_grp,node, dest,
        My_turn, Mygrp_nbr, Nextgrp_Nbr, H);
         }
         Else send End_Creat_grp(Id_grp,
         My_predecesor, send_turn, Mygrp_nbr,
         Nextgrp_Nbr, H);
         }
If (H= = K)
 {send End_Creat_grp(My_predecesor,
       send_turn, Mygrp_nbr, Nextgrp_Nbr, H);
      }
  } else
    {if (exist neighbor ≠ source)
        {dest= neighbor;
            Send Creat_grp(Id_grp, node, dest,
            send_turn, grp_nbr, Nextgrp_Nbr, H);
         }
         Else send End_Creat_grp (Id_grp,
          My_predecesor, send_turn, grp_nbr,
           Nextgrp_Nbr, H+1);
 if (MyId_grp= =Id_grp-1)
       Mygrp_Nbr = grp_Nbr;
    }
   }
}
```

After the construction of the first group, the beginner node starts, with the same way, the construction of the next group.

This process of creating group allows for each sensor to know the ID and the number of its group. This process stopped by the same beginner sensor after the last reception of the **End_Creat_grp** message with **Nextgrp_Nbr =0.**

End_Creat_grp(Id_grp, dest, send_turn, grp_nbr, nextgrp_Nbr, H)

```
{
If ( node_ID= =dest)
{
    if (exist neighbor ≠ source)
        {dest= neighbor;
          Send Creat_grp (Id_grp, node, dest,
          send_turn, grp_nbr, nextgrp_Nbr, H-1);
        }
        Else if (node == lancer_node)
        {       Id_grp++;
                   dest= first neighbor;
                    nextgrp_Nbr =0;
                   H=0;
            if ( Id_grp==1)
            {
                  Send Creat_grp (Id_grp, node, dest,
                send_turn, grp_nbr, nextgrp_Nbr, H);

              }else
             If (nextgrp_Nbr ≠0)
             {
               grp_nbr= nextgrp_Nbr;
               Send Creat_grp (Id_grp, node, dest,
               send_turn, grp_nbr, nextgrp_Nbr, H);
             } else // all group are created.
          } else send End_Creat_group (Id_grp,
              My_predecesor, send_turn, grp_nbr,
              Nextgrp_Nbr, H);

    }
}
```

4.4.1.1 Time distribution

It is obviously clear that the hidden station remains always a critical problem causing the collision and the energy loss. The existing solution currently for all the protocols MAC (S-MAC,T-MAC) bases the CTS/RTS mechanism, nevertheless the collision between these frames is strongly probable, when a node receives two frames CTS or RTS at the same time.

Our objective is to avoid collision between the sensor nodes in the same group and between the adjacent groups, using a distribution time between the sensors and the groups. This distribution enabled us to avoid the utilisation of RTS/CTS frames, to avoid the hidden host problem and to overcome the collision problem which will minimize the consumption of energy.

In order to avoid collisions in the group, it is indispensable to authorize, in a given time, only one sender node that can send its data. The idea is to find a distributed time function which distributes the transmission time period between the sensor nodes. This function is defined as following:

$\boldsymbol{Turn}\ (\boldsymbol{T})\ = \boldsymbol{T} \times\ \boldsymbol{Mod}\ (\boldsymbol{N} \times \boldsymbol{2})$ such as:

T: is the current time value.
N: is the number of sensor nodes in the group.
This function executed periodically by each sensor, this node can be authorized to send its data during this current period, if the value of the calculated function is equal to its sending turn.

However, to avoid collisions between the groups, we need to define a new mode pair and odd using the same above function. So, two adjacent groups can not have the same sending period, and must follow a different mode (pair or odd).

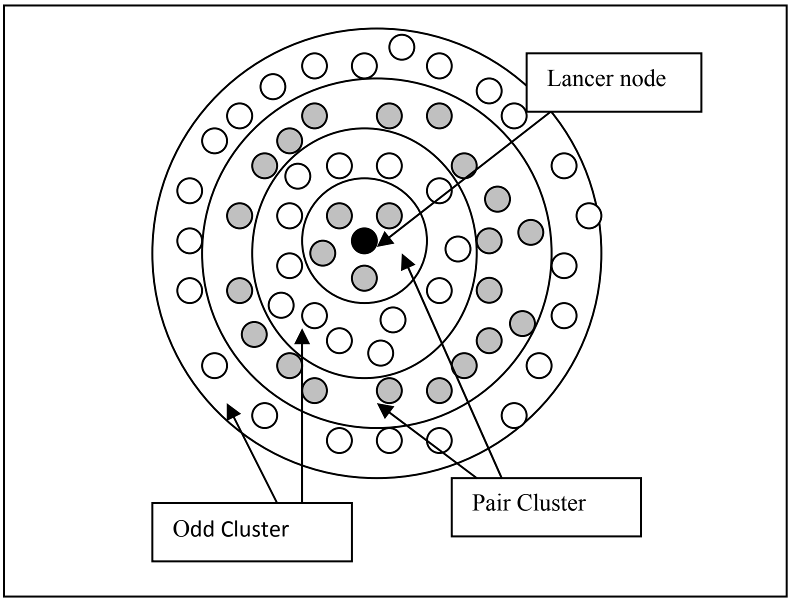


Figure 4.1: Cluster with completely ring.

4.4.1.2 Cluster form and pair and odd function

The objective of this function is to define two sending periods, pair and odd like shown bellow in Figure 4.1 and Figure 4.2 two adjacent groups should not follow the same mode. The shape of the clusters depends to the position of the beginner node.

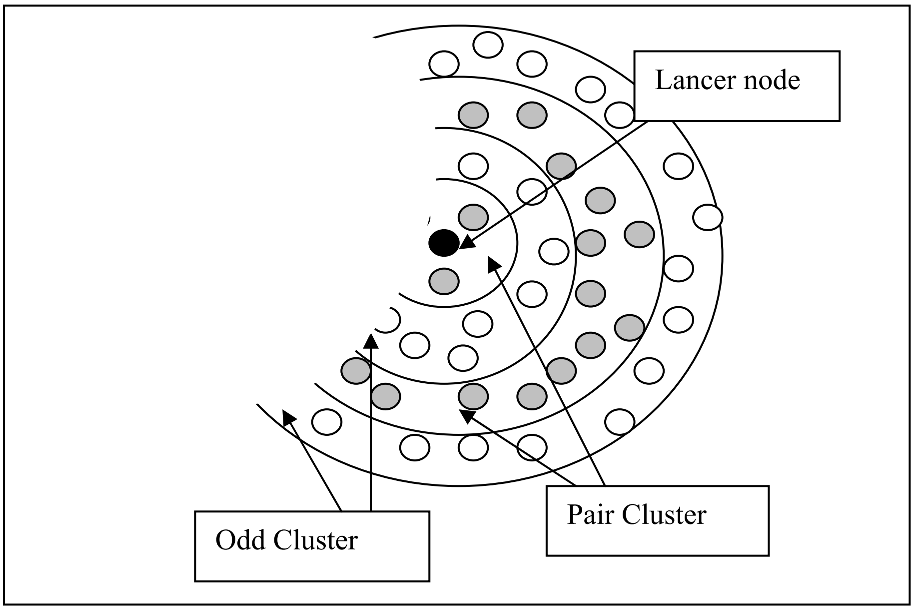


Figure 4.2: Cluster with incompletely ring

According to the geographical position of the beginner node, the cluster topology can be a completely or incompletely circle.

4.4.2 Operation phase

After the configuration phase each sensor node will be know its transmission turn, its group identifier and its group members, so the sensor nodes will be ready to operate using the above function.

4.4.2.1 The problem of sleep /active mode

The target is to find an effectiveness mechanism permitting sensor nodes to turn off and keeping the continuity of services in the sensing environment.

Basing on the distributed time given by the previous phase, when sensor node has got its sending time, using the Turn function, all its neighbours in the same group and in the adjacent group must be in the receive mode.

However, when sensor node reaches its transmission turn and it has not any information to send, it returns in sleep mode for one period of time to save energy. Also, if it was in receiving mode and all its neighbours are also in receiving mode, it must sleep during the current period.

The following diagram (Figure4.3) shows the send and receive mode of different nodes, in which two groups pair and odd are presented with two nodes in each group.

Group1 { - group_ID=1,
- group_type = pair,
- has to sensor nodes respectively with turn 0 and 2

}

Group2 { - group_ID=2,
- group_type = odd,
- has to sensor nodes respectively with turn 1 and 3

}

These nodes are designed by their sending turn, 0 and 2 for the first group, 1 and 3 for the second. These nodes execute periodically the **send_receive_sleep** function given below.

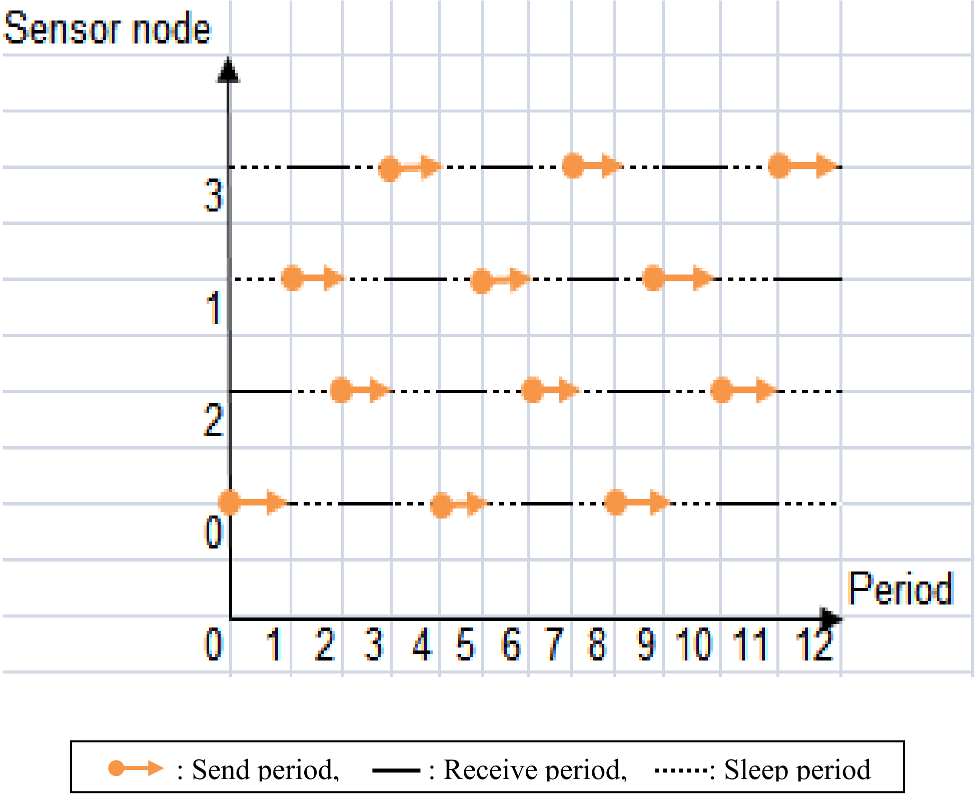


Figure 4.3: Send, receive and sleep mode.

At the end of each period, sensor executes this function. If the returned value is equal to its turn, it has to remain active during the next period and swaps itself to the send mode. Its neighbouring nodes have to remain in receive mode to receive the eventual data of sender node. If the active node belongs the to a pair cluster, the nodes of the odd one have to remain sleep during the next period and vice versa.

Send_receive_sleep()

```
{
If (Turn (T) = = myturn)
{
  If exist buffed data
  {send data;
   } Else sleep;
 } Else if all neighbor_Mode == receive
       {sleep;
       }
}
```

4.5 Evaluation of performances

Our protocol was evaluated through simulation using the Tossim simulator [147]. Some of the simulation tool's code was modified in order to implement the customizations necessary for our simulation. Each simulation of the application consisted of 100 sensor nodes like default number that varied between 100 and 600 nodes, placed randomly in area of 300 x 300 meters, as default value, which varied between 50 x 50 meters and 300 x 300 meters in order to evaluate the density parameter. The total simulated time for each simulation is 15 minutes and the radio range of each sensor is 30 meters. We define the below parameters and metrics:

4.5.1 Simulation parameters

To evaluate the performance of our protocol, we define three parameters:

- **Period time:** is the needed time during which sensor node remains active or sleep. At the end of this period, each sensor node executes the **Turn (T)** to decide its state (active or sleep) during the next period.

- **Cluster length:** this parameter defines the number of hop within the cluster. It is more important to study the impact of this parameter on the below metrics.

- **Density:** The effective range of the sensors defines the coverage area of a sensor node. The density of the nodes indicates the degree of coverage of an area of interest by sensor nodes. The network size affects reliability, accuracy, and data processing algorithms. The density can range from a few sensor nodes to a hundred in a region, which can be less than 10m in diameter. The density μ is calculated as in [134]:

$$\mu(\mathrm{R}) = \frac{(\mathrm{N}\pi\mathrm{R}^2)}{A}$$

  Where N is the scattered sensor nodes in region A, and R is the radio transmission range. Generally, μ(R) gives the number of nodes within the transmission radius of each node in region A

4.5.2 Simulation metrics

The most important metric that has to evaluate according to the above defined parameters is energy consumption, we define this metric as below:

- **Energy consumption**

  The energy E, consumed during the simulation due to communication overhead. E is the sum of the energy expended to transmit (Etx), the energy expended to receive (Erx), the energy expended while sensing the carrier (Es), and the energy spent in sleep mode (Ep).
  E= Etx + Erx + Es + Ep.

  In radio communications, the energy expended to transmit a message over a distance r is proportional to r^e where e is the path loss exponent (we take e = 2) [194] while the receive/sense energy is proportional only to the time the radio is on. The Etx and Erx energy consumption have been calculated using the model given in [194]

  $\mathrm{Etx}(k, r) = \mathrm{Eelec} \times k + \mathrm{Eamp} \times k \times r^2$
  $\mathrm{Erx}(k) = \mathrm{Eelec} \times k$

  Where k is the size of the message in bits, Eelec is the cost for just operating the radio and Eamp captures the amplifier that adjusts the transmission power (range).

Table 4.1 below, presents the parameters values of our environment. As used in [195], we assume that, the radio dissipates Eelec = 50 nJ/bit to run the transmitter or receiver circuitry, Eamp = 100 pJ/bit/m2 for the transmit amplifier and r^2 energy loss due to channel transmission. Thus, to transmit a k-bit message a distance r. We also assume that, the power expended while sensing the carrier Es =13mW, the power spent in sleep mode Ep=15µW and all the sensor nodes have the same channel bandwidth.

Table 4.1: Simulation parameter values.

Parameter	Default value	Variation interval
Number of nodes	100	100 - 600
Radio communication Range	50m	50 – 250m
Packet size	1024 Bytes	
Interval of event detection	60s	1 – 60s
Channel bandwidth	250 Kbps	
Idle power	13mW	
Sleep power	15µW	
Eelec	50nJ/bit	
Eamp	100pJ/bit/m2	
Simulation time	15x60 s	

4.5.3 Simulation results

The below Figure (Figure 4.4) below shown the evolution of energy consumption according to the period time, we want by this experience to find the ideal value of period time in order to fix it in the remainder experiences.

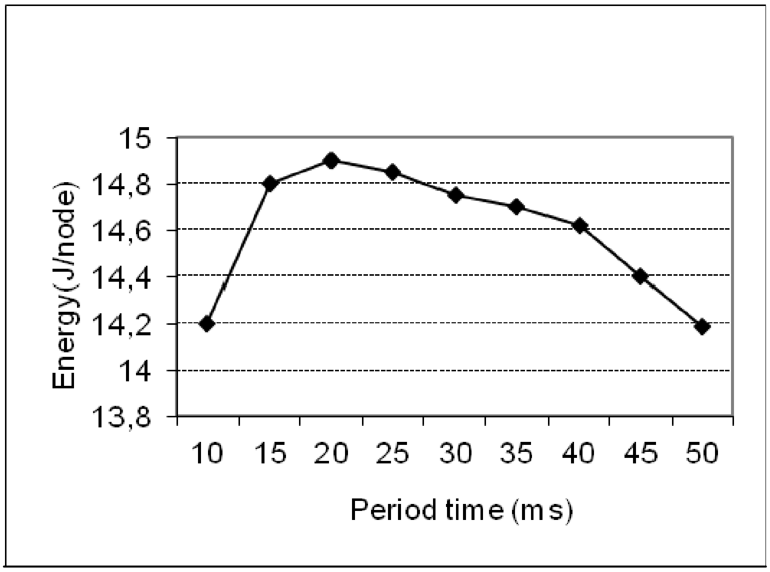


Figure 4.4: Energy consumption and time period

We observe, in this experience, that the energy consumption decrease when the period time takes its high and low value. Generally, when the period time increases, the energy consumption decreases also, because the sensor, in this case, has long period of send and receive, therefore, it can take more sleeping time. This period is also depends to the number nodes in the same group. So, when this number increases the sleeping period increases also, and the energy consumption decreases.

The below experience shown by the figure 4.5 displays the influence of the length of group, given by the number of hop, on the energy consumption. The energy consumption remains high with short group length, because the number of sensors in group decreases. Therefore, the sleep and receive periods decrease too. Contrary, the energy consumption decreases while the group length decreases, because the number of sensors in group increases and the sleep/ receive period becomes long.

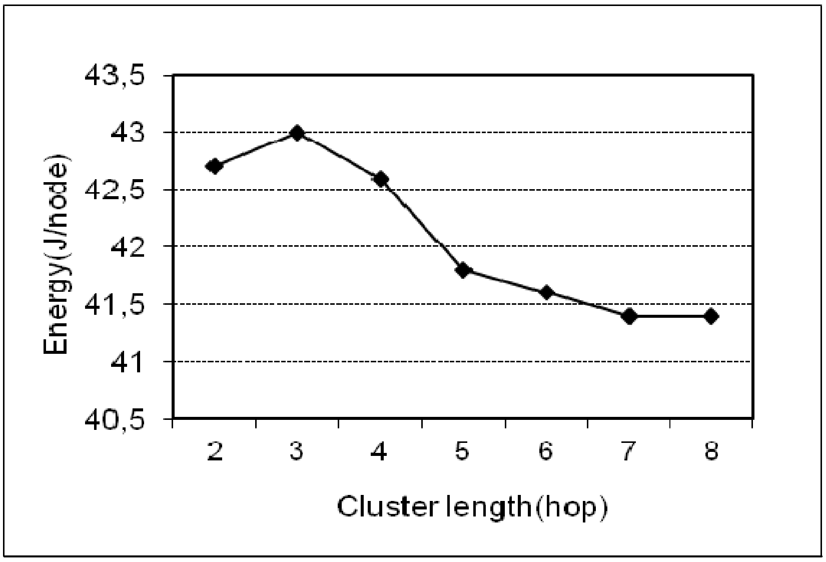


Figure 4.5: Energy consumption and cluster length.

In the below experience (Figure 4.6), we present the evolution of energy consumption according the density parameter. We observe clearly that the energy consumption is always higher in case of S-MAC protocol comparing with the tow others, the most of this consumption is produced essentially by the transmission of synchronisation message, RTS/CTS mechanism and the possible collisions that can be occurred during the network operation. The energy consumption is also high in case of B-MAC, but it remains lower than the S-MAC, because the B-MAC is not based on the synchronisation mechanism.

For the RSMAC, the energy consumption is lowest and becomes stable after some density value, because in RSMAC the RTS/CTS mechanism is not used and the distributed time done by the Turn (T) function allows the collision avoidance.

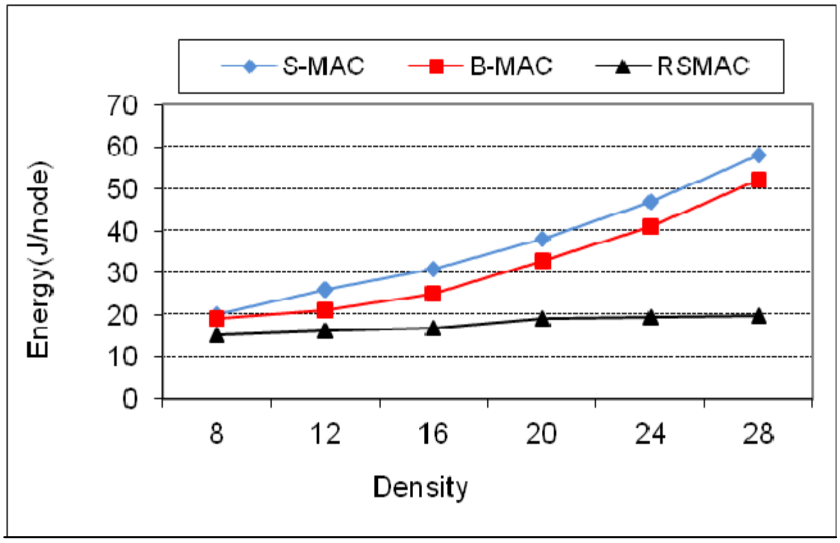


Figure 4.6: Power consumption and density.

4.6 Conclusion

In this chapter we have presented our solution designed for wireless sensor networks. Collision avoidance and energy efficiency are the primary goal in the protocol design. The proposed solution was principally based on the study of the two protocols S-MAC, B-MAC and other protocols such as the WISEMAC. In our design, we have introduced new mode of send/receive/sleep using clustering ring and new function to distribute transmission time between sensor nodes. The most advantageous of RSMAC is the reducing of energy consumption and collision avoidance. However, as the synchronous MAC protocols, RSMAC has more complexity especially during its initialization phase. Moreover, its operational mode makes it more suitable for the no real time applications. In the next chapter we try to overcome these disadvantageous by presenting new asynchronous MAC protocol.

Chapter 5

Asynchronous MAC Communication and Energy Consumption in Wireless Sensor Networks

5.1 Introduction

Energy parameter in wireless sensor networks has a great impact on the lifetime of sensor node. The greater part of this consumption occurs in the MAC layer [183]. In this layer all neighbouring share the same wireless communication channel, so MAC protocol specifies how nodes share the channel, and hence plays a central role in the performance of a sensor network. Packet collisions waste energy by forcing packets to be retransmitted, idle listening is the cost of actively listening for potential packets, overhearing is the cost of receiving packets intended for other destinations, and control traffic represents MAC-level maintenance overhead. Since many sensor networks are quiescent between sensor readings, idle listening can easily become the largest energy cost in a sensor network.

Energy conservation is one of the key technical challenges in sensor networks and ad hoc networks. It is necessary to devise communications and networking schemes which make judicious use of the limited energy resources without compromising the network connectivity and the ability to deliver data to the intended destination. In addition, sensor nodes are often subject to further constraints in terms of CPU power, memory space, etc., which call for simple algorithms and schemes whose memory needs are modest. One of the main mechanisms to save energy is the use of sleep modes at the MAC layer, by which nodes are put to sleep as often as possible.

In the previous chapter we have presented an energy aware MAC protocol, in which sensor nodes are organized under virtual clusters. This protocol reduces energy consumption, but its synchronous mode may require more resources. Also, its active/sleep and send/receive mode may increase the latency. This protocol is more likely for no real time applications.

In this chapter we give more detail about our solution presented in [110]. We propose this solution in order to overcome the inconvenient sited above and support real time traffic. We present an algorithm solution for MAC layer by considering the primary keys of energy consumption like collision, overhearing and idle listening. We consider also the problem of the hidden terminal. We summarize the most important points of MAC layer communication modes. We present the parameters of our environment and we develop our solution. We present the performance analysis and simulation results

5.2 Mac layer communication mode

As given in chapter 2, three communication modes could be exist at MAC layer, synchronous mode, asynchronous mode and hybrid mode. Several researches have been done in each mode, and all of them attempt to reduce the energy consumption by reducing the number of collision that can be happened during the data transmission.

- Synchronous MAC protocol improves the energy consumption, but generally the synchronisation process requires more resources and increases the latency. In S-MAC [198] protocol, a sensor node life is divided into two times. Each node sleeps for some time, and then wakes up and listens to see if any other node wants to talk with it. During sleeping period, the node turns off its radio, and sets a timer to awake itself later. This mechanism must be done in such a way that connectivity is preserved, since if too many nodes are sleeping at the same time, the network may end up being disconnected. In the recent literature, several solutions have been proposed which address this problem. S-MAC has been modified in [163], this modification did not change the mechanism of data transmission and reception mode. It only modified the procedure which determinate the sleep schedules and wake-up of node. In the original S-MAC, the nodes form virtual clusters around shared schedules. In [19], the technique introduced tries to coordinate the sleeping activity of the nodes so that a connecting backbone is always present. In [207], the Authors utilize geographic location information, and self-configure redundant nodes into small groups based on their locations and uses localised, distributed algorithms to control node duty cycle to extend network operational lifetime. However the properties of the grid topology in [110] have not been fully studied. In [183], the authors provide a means to communicate with a node currently asleep, by implementing a *rendezvous* mechanism based on beacon transmissions. As to the MAC itself, most works in the literature assume either TDMA-based schemes [44], or multi-channel setups in which parallel transmissions can be performed without interference [94], [24], or variants of classic contention-based schemes, usually based on RTS/CTS handshake in order to mitigate the hidden terminal problem [183], [207].

- Asynchronous mode characterised by its simplicity and its access channel free. B-MAC [79] and WISEMAC [7] tried to eliminate the most possible assumptions used in [198] in order to approach the real environment with the disadvantages which are known in wireless sensor networks. B-MAC, which extends previous works [18], [34], uses a tone to wakeup sleeping neighbours.

In this protocol, sensor nodes independently follow a sleeping schedule based on the target duty cycle for the sensor network. Since the sensor nodes operate on independent schedules, B-MAC uses very long preambles for message transmission. The source sensor node transmits a preamble long enough that the destination, which periodically senses the channel, has enough time to wakeup and sense activity. Sensor nodes that sense activity on the channel remain awake to receive the message following the preamble or return to sleep if they do not detect activity on the channel. Before transmitting, sensor nodes delayed by a random time to prevent synchronization, and sense the channel to prevent corrupting an ongoing transmission. Additionally, the B-MAC authors provide a great deal of flexibility through a protocol interface that allows the sensor node to change many operating variables in the protocol, such as delay and bakeoff values. Typical of an unscheduled MAC protocol, B-MAC relies on a version of CSMA suited for a sensor network platform. As such, B-MAC provides no implicit protection against traditional wireless problems, such as the hidden terminal problem sensor nodes using B-MAC have instant access to the network once deployed or moved since the protocol requires no setup or prior communication. Furthermore, the long preambles in B-MAC do introduce an additional latency. WiseMAC [7] protocol attempts to reduce energy consumption by having sensor nodes remember the sampling offsets of their neighbours. An extra field in ACK packets allows sensor nodes to notify their neighbours of the time until their next channel sampling. By learning the sampling times of its neighbours, a sensor node can delay transmitting the preamble until just before the receiver wakes up to sense the channel. WiseMAC can thus decrease the amount of time a sensor node transmits preambles and the number of sensor nodes that overhear each message at the cost of an extra field in the ACK messages and the memory required to store neighbour's sampling offsets. The protocol proposed in [102] attempts to conserve energy by utilizing the knowledge that sensor nodes located near each other generate correlated measurements. To achieve energy savings, this protocol filters measurements from highly correlated sensor nodes in an effort to reduce the number of messages the sensor network must handle.

5.3 MAC Protocol design (P-MAC)

To overcome the inconvenient of synchronous communication mode, we adopt in our protocol asynchronous communication and we design our solution accordingly. To reduce collision and thus minimize energy consumption and latency, we propose solution to avoid the problem of hidden host.

5.3.1 System model and hypothesis

We define the same environment like presented in the previous chapter (section 4.3) where a large number of wireless sensor nodes are deployed in large area. Sensor nodes collect from its communication range and transmit it toward the sink node. In this environment we assume the following assumptions.

- Wireless sensor nodes which share the same wireless communication channel.
- Each sensor node has an autonomous and limited energy source.
- Each sensor node acts as source to detect and sense data, and as router to participate in data dissemination process. We assume also the existing of only one sink which is responsible to forward the sensed data to the end user or administrator directly via wired link or remotely via internet or satellite.

5.3.2 Asynchronous communication mode

Because the synchronous mode used in [198] caused many problems such as:

- The collision in synchronization frames themselves;
- The required time of synchronization process.
- The need to share a common program listens/sleep between the nodes.

Our protocol is based on the asynchronous mode that offers the advantage of simplicity. Without having to maintain and share state, an unscheduled MAC protocol may consume fewer processing resources, have a smaller memory footprint, and decrease the number of messages that a sensor node must transmit in the asynchronous mode [79], each data transmission must be preceded by a sending of a preamble permitting the receiver node, when it wakes up, to remain wake up to receive the some eventual data. However, the use of the preamble poses the following problems:

- A loss of energy from the transmitter side.
- A long transmission time in each hop for the reason that the receiver must remain wake up waiting for the end of the preamble, because the transmitter can not know that the receiver detects the preamble.
- The overhearing problem results, because all nodes that detect the preamble stay wake up during the transmission time, and hence, it is impossible to know the destination node.

Our first idea is to provide the address of the destination node in the preamble, so that the undesired nodes can quickly turn to the sleep mode. Thus permit to eliminate the overhearing problem.

The second idea is to fragment the long preamble into several small preambles including the receiver address. Therefore, the sender node needs to make a break between each two small preambles which allowed him to receive an ACK from the target node. In this case the transmission of the preamble should be stopped and the data transmission will be started. In this way, the latency per hop is reduced and the energy consumption is decreased.

The second idea can also resolve the below problem, when several nodes want to send data and sense, in the same time, other data transmission from other node to the same destination. These nodes remain wake up and wait until the channel becomes free, then they begin the transmission of their preambles. Consequently, the sending of the preamble in this case is useless.

However, when sensor node wants to transmit data and detects a preamble, it must still wake-up, and if it senses an ACK from the destination node, it will wait a random time (*Backoff*) and sends its data without using the preamble. Random time is necessary to avoid collision.

This *Backoff* time should be sufficiently long to permit to the first transmitter finishes its data transmission. Moreover, the destination node remains wake-up for a small period at the end of the data reception to receive some eventual data from another transmitter. This period must be equal to the maximum random duration of the transmitters.

The transmission time of the preamble can be more reduced if the next listen period of the destination node has been added in the ACK at the end of data reception. However the transmitter node, in the next communication, can start its data transmission by sending a small preamble.

5.3.3 Problem of the hidden host

The problem of the hidden station arises when two nodes of two different ranges transmit to the same destination node. Figure 5.1 below, illustrates the hidden terminal problem. Suppose that node *A* wants to transmit to node *B* located in its communication range. By only sensing the medium, node *C* will not be able to detect the transmission started by node A (A and C are not in the same communication range) and will start transmitting, leading to collisions at node *B*. This is the well known hidden host problem, where the hidden nodes are not located in the same communication area and transmit to a node that can communicate with them directly and simultaneously.

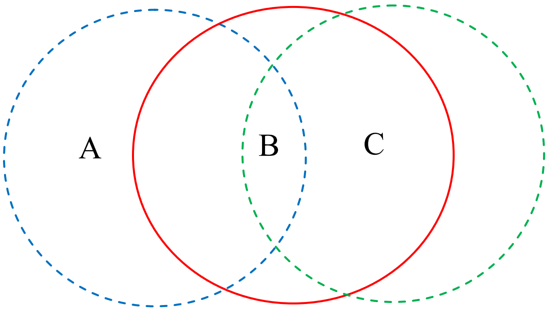


Figure 5.1: Hidden host problem.

This problem was already solved in [198], [79], [7] by using the RTS/CTS mechanism, however this solution presents several disadvantages such as:

- Collision between RTS and CTS packets is possible.
- The traffic can be increase so the energy consumption result is higher. Also this mechanism does not solve all the hidden terminal problems.

For this reason, we have modified this mechanism by using the following method:
We suppose that at the level of each node, there is a neighbouring table of one hop (H1), and then each node sends its table to its neighbours in order to get a neighbouring table of two hops (H2). In this way, if two nodes of different ranges want to transmit to the same destination, each one of them can know if there is another node which can communicate with the same destination node, see the following example (Figure 5.2).

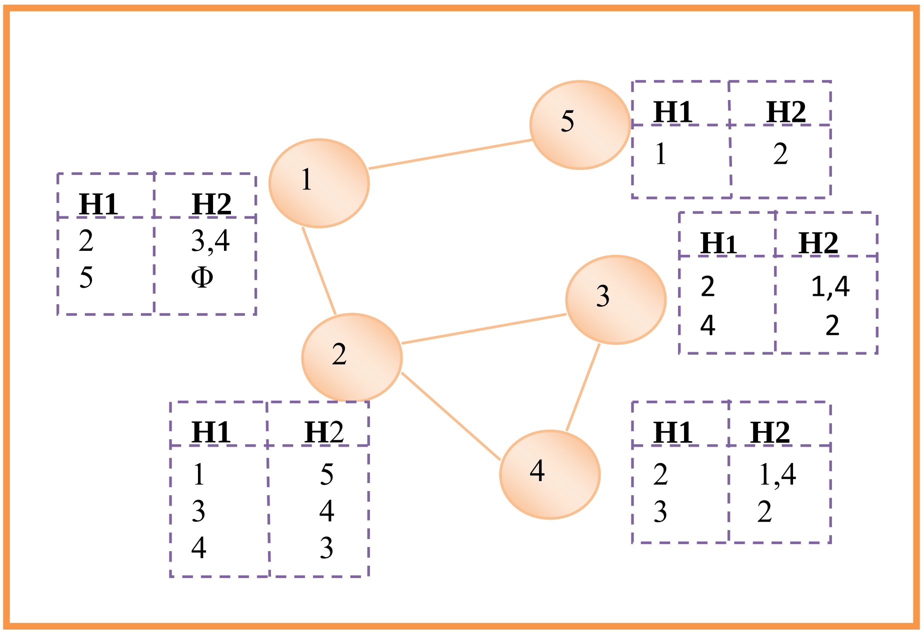


Figure 5.2: Two hops neighbours.

Therefore, before node begins its transmission, it must initially check the existing of the hidden host by consulting its two hops neighbouring table (H1, H2). If the hidden host exits ($H2\neq\Phi$), the node must firstly send a TST frame (Transmission at the Same Time) specifies the destination node and the data transmission time. When the destination node receives the TST frame it performs to transmit it to its neighbours and sends a TSTC (TST Clear) to the sender host which will start the data transmission after receiving the TSTC. The other hosts not concerning by this TST will proceed as below:

1- For the sensor node that receives a different TST from each neighbour, will turn off its radio during the time of minimum transmission.

2- Other ways, the sensor node must only sample the medium and it can not transmit during the time of maximum transmission. The advantage of TST frame is that it is used only when we are sure that there is a hidden node and thus a collision risk, contrary with the use of the RTS frame which is need to send before each transmission.

Send_Data()

```
{
if (T(H2) <> Φ) then Send TST
                   else {
                          Random Since
                          Send Data
                         }
}
```

TST_Received()

```
{
  If (Dest_Add=Node_Add)
      {
        Random Time
        Forward TST to all neighbours
        Send TSTC
      } else
            {
              Receive only or sleep
             }
}
```

In the above example (Figure 5.2), when node 1 wants to send a data to the node2, must firstly send TST frame which will be received by the node 5 that will turn off its radio during the transmission time specified by the TST frame. Sensor node2 sends a TSTC to the node1 that also will be received by node3 and node4. Basing on the neighbouring information (H1 & H2), node3 and node4 will also turn off their radios during the transmission period.

5.4 Performances evolution

Our protocol was evaluated and compared with S-MAC and B-MAC protocols through simulation using the Glomosim simulator [154]. Some of the simulation tool's code was modified in order to implement the customizations necessary for our simulation.

The sensors nodes are randomly chosen to produce and generate new sensing data. The process of data detection follows the model of POISSON with an average interval of 60 seconds. To evaluate the effect of network load, this interval has been varied between 1 and 60 seconds. In our simulation, we have used the CBR application to produce and generate data. This application simulates a constant bit rate generator. In order to use CBR, the following format is needed:

CBR <src> <dest> <items to send> <item size> <interval> <start time> <end time>

where

<src> is the client node.
<dest> is the server node.
<items to send> is how many application layer items to send.
<item size> is size of each application layer item.
<interval> is the inter-departure time between the application layer items.
<start time> is when to start CBR during the simulation.
<end time> is when to terminate CBR during the simulation.

For example:

If we set CBR 0 1 10 1024 1S 0S 600S, node 0 sends node 1 ten items of 1024B each at the start of the simulation up to 600 seconds into the simulation. The inter-departure time for each item is 1 second. If the ten items are sent before 600 seconds elapsed, no other items are sent.

In our simulation we have used 100 sensor nodes as default number that randomly placed in area of 1000x1000 m^2. The number of sensors has been varied between 100 and 600 nodes in order to evaluate the density parameter. The density is defined by to parameters, the number of sensor nodes N and the size of physical terrain or area in which the N sensor nodes deployed. For example: if N= 300 nodes and TERRAIN-DIMENSIONS (1000m, 1000m) the density D= 300 node/Km^2.

5.4.1 Simulation parameters

To evaluate our protocol and compare its performances with other well known protocols, we define the density parameter as given in the previous chapter (section 4.5.1). We use this parameter to evaluate the below metrics:

- **Energy consumption:** we calculate the energy consumption using the same formula presented in the previous section 4.5.2.

- **Collision:** is the average of collided packets of each node, we calculate the collision as below:

$$\text{Collision} = \sum_{i=1}^{n} \frac{\text{collision\_node i}}{\text{i}}$$

Where: collision_node i is collision number of node i occurred during the simulation time. N is the total number of sensor nodes.

5.4.2 Simulation results

Figure5.3 below represents the evolution of energy according to the density. Concerning protocol S-MAC, we notice that the energy consumption does not influenced by the density parameter. However, this consumption is still higher than the other protocols. For B-MAC, we observe, that the energy consumption increases with density increasing.

The protocol B-MAC is effective than the S-MAC in term of energy consumption, because it introduced the sleep period without synchronisation. We observe also that the protocol P-MAC is more effective and consumes low energy than the B-MAC, because it uses the principle of the sleep period and preserves the energy lost by using the concept of short preamble.

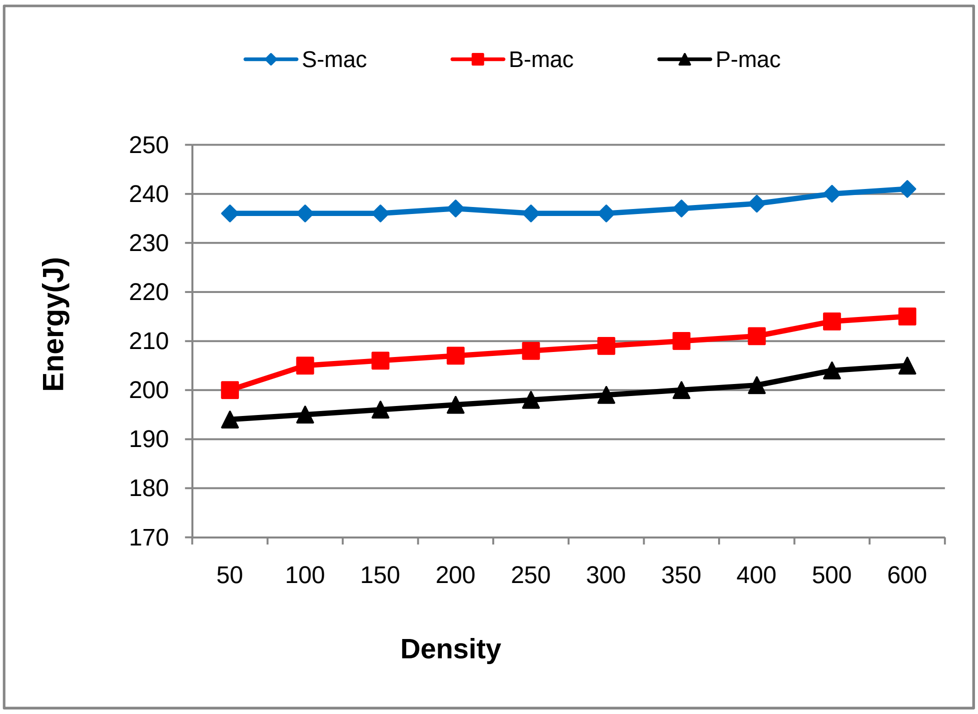


Figure 5.3: Energy consumption and density.

Figure5.4 below represents the variation of the collision rate according to the density variation. We notice that, for the three protocols, the number of collision increases when the density increases. Indeed, more the number of nodes increases, more the number of messages exchanged between them increases and thus more the probability of collision between these messages increases.

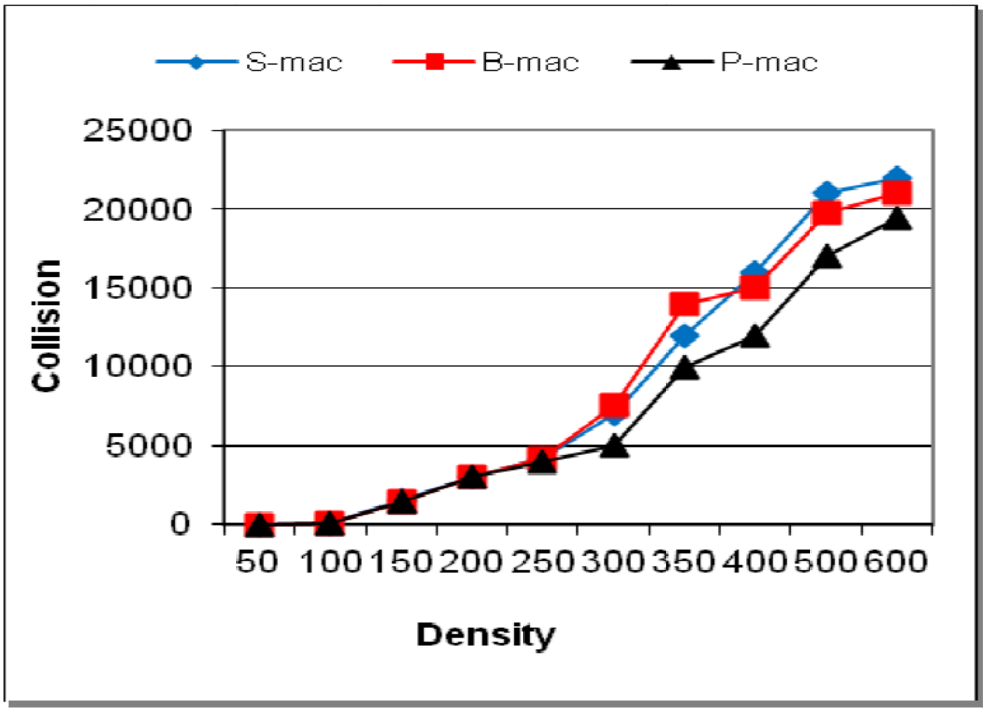


Figure 5.4: Collision and density parameter.

Comparing the three protocols, we notice that the S-MAC and B-MAC are almost superimposed. This is due to the fact that both use mechanism RTS/CTS. Nevertheless, this number is a more less in the S-MAC because there is a collision risk between SYNC frames.

Concerning our protocol, the number of collisions is lower because it uses the TST/CTST frame, which is sent only when we are sure that there is a hidden host. Contrary to the RTS/CTS mechanism, that is used before each data transmission.

5.5 Conclusion

In this chapter we have presented a new MAC protocol, specifically designed for wireless sensor networks. Collision avoidance and energy efficiency have been the primary goal in our protocol design. This solution was principally based on the study of the two protocols S-MAC, B-MAC and other protocols such as the WISEMAC protocol. We have kept the same listen/sleep period introduced by protocol S-MAC, using asynchronous mode and short preamble to wake up the destination node. The protocol presented in this chapter is principally motive by the limitations of our MAC protocol presented in the previous chapter such as latency and complexity. Simulations show that the short preamble and the TST/CTST mechanism of P-MAC reduce the energy consumption, decrease the collision and thus improve the latency.

In next chapter, as a starting point of our next part of this thesis, we present our first data dissemination protocol for wireless sensor networks with static sink.

Chapter 6
Data Dissemination in Static Sink Wireless Sensor Networks

6.1 Introduction

The main task of a wireless sensor network is the monitoring of a larger area. Usually, the end user wants to extract information from the sensor field, this information is gathered by the sensor nodes, and disseminated to the sensor sink. Moreover, this information can be sensed and disseminated to the sink node and after forwarded to the end user without requested by the last one. A possible solution, to disseminate the interested data is letting the sensor nodes use the flooding technique. Nevertheless, this technique produces a high traffic load and consumes many energy resources.

Data dissemination is a process by which the sensed data will be transmitted from the source sensor node to the sink. It consists to determine the optimal path on which the information will be disseminated. The characteristics of the networks of sensors, like the significant density and the limited energy require specific data dissemination protocol. However, the aim of our research consists to conceive and validate a protocol of data of use the concept of aggregation to minimize energy consumption.

Data dissemination in wireless sensor networks has been considered as an important mechanism and very interested issue. In such environment, data dissemination is generally performed from sensor nodes to the sink. Thus, each sensor node can be implicitly provided the direction to forward sensed data towards the sink. The inherent characteristics of sensor nodes such as limited battery and limited computing capability, make the forwarding mechanism is not more suitable for this kind of network.

In this chapter, we present new data dissemination protocol based energy-efficient called Data Dissemination and Power Management Protocol (DDPM) [114]. In this protocol, we propose new energy management scheme using a dynamic power threshold. Firstly, in the initialization phase, the sensor nodes organized under clusters and cluster head should be selected for each cluster. Secondly, in the data dissemination phase, the cluster head collects and transmits the sensed data based on the data dissemination process. The simulation result shows that the proposal protocol permits to reduce the energy consumption and prolong the network life.

The remaining parts of this chapter are organized in the following way. Section 6.2 reviews a set of related data dissemination protocols and summarizes some recent works. Section 6.3 presents the parameters and the assumptions of our environment. Section 6.4

presents our proposal and gives the necessary description of the different concepts used in our design. Performance analysis and simulation results are presented in Section 6.5. Section 6.6 concludes this chapter.

6.2 Related works

Several data dissemination protocols for sensor networks have been proposed in the literature to address the data communication problem in these networks. Protocol LEACH proposed in [194] is one of the first approaches of the hierarchical data dissemination sensors networks. LEACH has been considered as an effective protocol in energy consumption this protocol can extend the lifetime of the network [194], compared with the other protocols. Moreover, this protocol organizes the sensor nodes in clusters form, the elected cluster heads collect the data from its sensor nodes, aggregate and transmit them directly to the sink node, these cluster heads changed and elected periodically. TEEN is a protocol of data dissemination based on the clustering technique proposed by anjeshwar & Al [14]. TEEN uses the same strategy as LEACH to create the clusters node, but adopts a different approach during the data transmission phase. In this phase, TEEN uses two parameters called hardware threshold and software threshold to determine the need of collected data transmission. PEGAGIS [170] is another data dissemination protocol designed for sensor networks which improves the previous LEACH. In this protocol, a sensor node communicates only with the closest neighbors, it should wait its turn to transmit its data to the sink node.

CODE [55] is a protocol based on a virtual grid structure, where each cell of the grid contains a node called coordinator playing the role of an intermediate node. Only these nodes coordinators take part in the process of data dissemination. This protocol is principally inspired from some previous works like GAF [208], [82]. TTDD [51] considers sink mobility, by constructing grid networks for each data source and selecting a grid node as the communication portal of mobile data sinks.

Other protocol SPIN [195] considers the end-to-end communications in sensor networks. It supposes that two sensor nodes can communicate between them without any interference with other nodes. This protocol supposes also that the energy consumption does not constitute a constraint, and the data are never lost.

In the directed diffusion protocol [26], data are inherently dispersed with the physical object and retrieved via queries transferred to the object through the network. It also envisions that the querying and monitoring the physical space may rely on multicast mechanisms. Protocol PDDD [122] tries to surmount the disadvantage of the multicast mechanisms used in the directed diffusion protocol. It eliminates the gradient algorithm of directed diffusion and exploits the information of neighbor nodes.

According to user importance, SAFE [169] considers service differentiation between data sinks, allowing each data sink itself to specify the desired data update rate. This aspect entails multiple level provision of data freshness.

Other protocol MMSPEED [41] presents an evolution in the protocols oriented quality of service. MMSPEED offers several transmissions speed and establishes more than one route form the source node to the destination. Therefore, each offered speed defines a level of temporal QoS and each additional route helps to improve the quality of traffic. These two mechanisms respectively make it possible to respect the degree of criticality of each application, to transmit the data within the required times and to avoid the problems frequently encountered like the congestion and packets loss.

6.3 System model and assumptions

We consider a large scale sensor network with a large number of sensor nodes scattered randomly. Each node acts as either a source to sense information from the environment or a router to forward data through the sensor field to the interest users. We consider only the detected events occurred in the sensor network, the end users receive frequently and randomly the new detected events. Therefore, sensor nodes should be preconfigured to send a notification if the new event verifies some parameters, example if the temperature exceeds or decreases under predefined degree. In this environment we assume that:

- Each sensor node should be aware of its own geographic location using location services such as GPS [187], [133]
- After having been deployed, sensor nodes remain stationary at their initial locations.
- The sensor nodes are stationary, homogeneous and each sensor node has constrained battery energy.
- Sensor nodes communicate with sinks via a multiple hops path.

6.4 Data Dissemination and Power Management Protocol (DDPM)

This protocol extends our previous work namely Clustering Data Dissemination Protocol (CDDP) presented in [113], it bases on a structure of indexed virtual grid like the same used in [55], where each cell represents a cluster nodes that contains a selected cluster head (figure 6.). The selected cluster head collects the data from all the nodes of its group, aggregates and transmits them to the basic station or sink using multi-hop communication. The main difference between CDDP and DDPM is explained in the Section 6.4.2.2, in CDDP, the cluster head role is rotated periodically among the sensor nodes of the same cluster. DDPM uses an enhanced mechanism based on the dynamic energy threshold and the available energy of sensor node.

Like CDDP, the protocol DDPM starts with an initialization phase, during which, the virtual grid will be constructed and a cluster header will be selected for each group. Thus, the process of data dissemination will start in the second phase.

During the second phase, the cluster head receives the data collected by the nodes of its group, then, it transmits them to the interested sink hop by hop, the next hop is defined by the indices of next cell in the grid.

During the second phase, the cluster head receives the collected data sent from the sensor nodes of its cluster. It performs the needed data aggregation to reduce network overhead and thus energy consumption, Section 6.4.2.1 explains this operation. To transmit the aggregated data to the sink, the cluster head has to select one neighboring cluster head as next-hop, the next-hop is determined based on the sink geographical position and the indices of next cell in the grid. Section 6.4.2 presents the next-hop selection procedure.

6.4.1 Initialization phase

The initialization phase begins just after the deployment of network. It aims to prepare the sensor nodes for the second phase of data dissemination process. During this first phase, the virtual grid will be constructed using the geographical position of the sensor nodes and a head will be selected for each cell.

- **Geographical position:** each sensor node calculates its geographical position using GPS [187], [133]. This information is necessary for the virtual grid construction and the data dissemination procedure.
- **Grid virtual:** In virtual grid each cell is known by its coordinates (C_x, C_y). And each sensor node calculates the coordinates of its cell using the following formulas [55]:

$$C_x = \left\lfloor \frac{x}{r} \right\rfloor \text{ and } C_y = \left\lfloor \frac{y}{r} \right\rfloor$$

Where: r is the cell size.

x, y are the geographical coordinates of sensor node.

To ensure the communication between the neighboring cluster heads, the distance between the two most distant cluster heads of neighboring cells should be less or equal to the communication range of cluster head node.

For example in Figure 6.1, the most distant neighboring cluster heads are nodes 1 and node 2 respectively located in cell C2 and cell C3, the distance between these cluster heads should be less or equal to the communication range R in order to be able to communicate between them. This translated mathematically by the following formula:

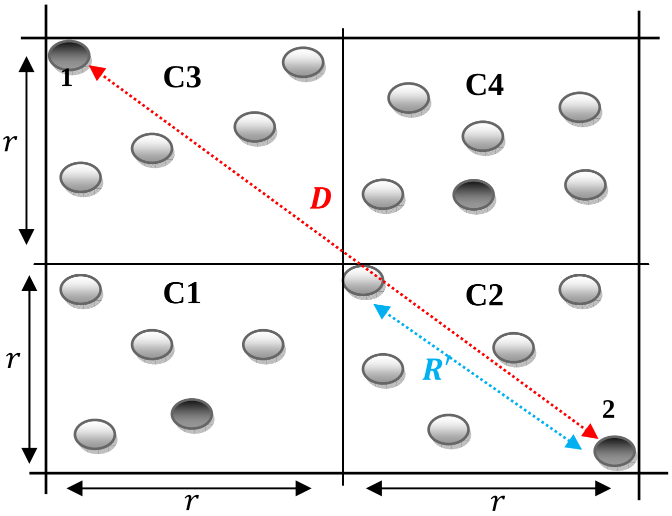


Figure 6.1: Cell size (r).

Let D is the distance between the most distant neighboring cluster heads and R is the communication range of the cluster head nodes.

$$D \leq R \quad \textbf{(1)}$$

Referring to the Figure 6.1 and using Pythagorean theorem the distance D can be calculated as below:

$$D^2 = (r + r)^2 + (r + r)^2 \quad \textbf{(2)}$$

From (1) and (2), cell size (r) can be defined as shown below:

$$(r + r)^2 + (r + r)^2 \leq R^2$$

$$(2r)^2 + (2r)^2 \leq R^2$$

$$4r^2 + 4r^2 \leq R^2$$

$$r \leq \frac{R}{\sqrt{8}}$$

$$\boldsymbol{r \leq \frac{R}{2\sqrt{2}}}$$

It is known that the long communication range increases the collision domain and sensor node consumes more energy when transmitting data. To avoid this problem, we define another communication range for the simple sensor nodes shorter than the communication range of cluster head. Cluster head has to communicate with its neighboring cluster head, its communication range R is equal to the maximum distanceD. According to fig.1 and using Pythagorean theorem, R can be calculated as below:

$$R = D = \sqrt{(r+r)^2 + (r+r)^2}$$
$$R = \sqrt{4r^2 + 4r^2}$$
$$R = 2r\sqrt{2}$$

Sensor node communicates only with its cluster head. Its communication range R′ is calculated as below:

$$R' = \sqrt{r^2 + r^2}$$
$$R' = r\sqrt{2} \qquad \textbf{(3)}$$

Because the cluster head role is rotated among the sensors in the cluster nodes. Sensor node should adapt its communication range (R or R') according to its function

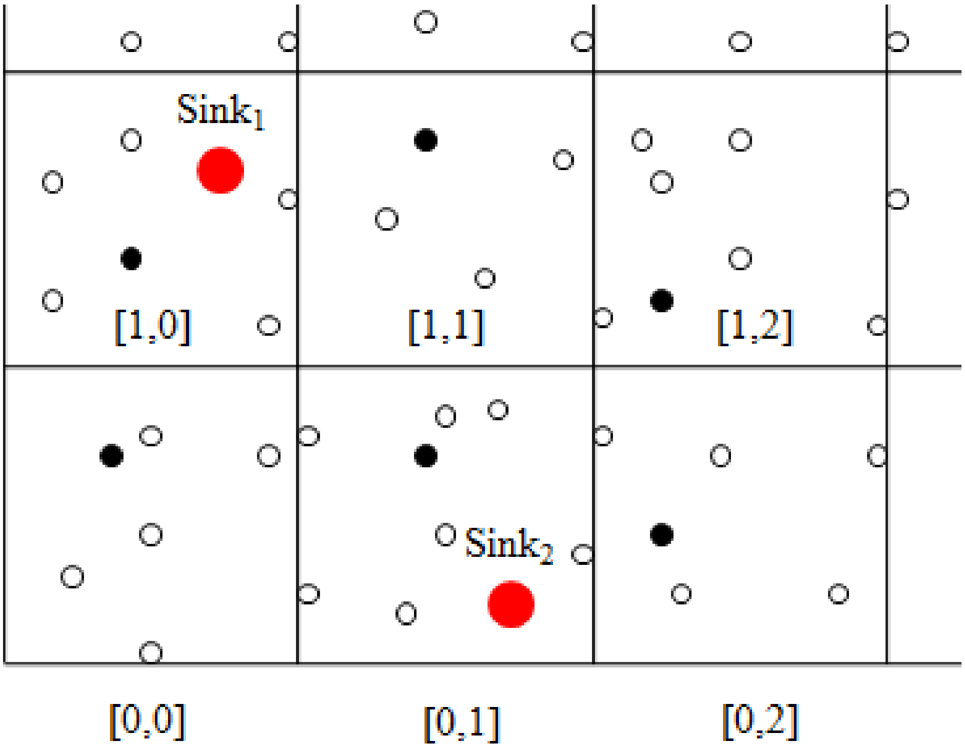


Figure 6.2: Virtual grid.

- **Selection of cluster head:** Selected cluster head depends to the energy availability of sensor node. In each cluster, only one sensor node which has the highest energy availability superior than a predetermined threshold can be selected as head. Initially, all the nodes have the same probability to be a cluster head.

- **Sink table:** each cluster head creates and maintains in its cache a Sink_table that contains *SinkId, SinkPos and Data_type* where:
  SinkId: is the sink identifier.
  SinkPos: the geographical position of the sink.
  Data_type: type of data in which the sink is interested.

After the construction of virtual grid (Figure 6.2) and the selection of the cluster head, each sink advertises the above information which permits to the cluster heads to update their Sink_table.

The sink_table correspondent to head of the cluster [0, 0] is:

Table 6.1: Sink table.

$Sink_{id}$	**Pos**	**Data_type**
$Sink_1$	[1,0]	T1
$Sink_1$	[1,0]	T2
$Sink_2$	[0,1]	T1

DDPM supports multiple sinks, but in the remainder of this chapter we use only one static sink.

6.4.2 Data dissemination process

The data disseminations process is based on the virtual grid built in the first phase. The cluster head collects and transmits the observed data using the virtual grid. Therefore, the data dissemination will be done through two levels.

- Locally, between source sensor node and its cluster head in the same cluster.
- Externally, between the clusters heads from the first head of the cluster in which the event has been sensed to the sink.

Local level:

When a sensor node observes an event, it sends a message of data information contained the type of the observed phenomenon. This message will be received locally by its cluster head, the last will be responsible transmit it to the interested sink.

External level:

In this level the data dissemination is based on the virtual grid, such as only the cluster heads will participate and collaborate to deliver the sensed data to the interested sink. Therefore, when a head receives the sensing data, will define the next destination head, so this next hope will be selected as below:

(X,Y) : Cluster coordinates of the current node.
(Xsink,Ysink) : Cluster coordinates of the sink node.
(Nexthop.X,Nexthop.Y) : Cluster coordinates of next head.

Select_Nexthop()

```
{ If (Xsink=X)   then
  { nexthop.X=X ;
     Selecte_y() ; }
else
   if (Xsink  > X )  then
        { nexthop.X=X++;
           Selecte_y() ;  }
     else
        { nexthop.X=X-- ;
           Selecte_y() ;  }
}
```

Select_y()

```
{  if (Ysink=Y  )  then   nexthop.Y=Y ;
                   Else if (Ysink>Y  )
                  then  nexthop.Y=Y++ ;
                     Else nexthop.Y=Y-- ;
  }
```

6.4.2.1 Data aggregation

During the dissemination phase, each cluster head executes the necessary data aggregation to decrease the number of data transmission packets. When the cluster head receives data from one of its member nodes, it will ignore all the same data received during the next T second, we assume that the sensed data detected during this time are correlated and have the same semantics. T is the necessary data transmission time between two farthest nodes in the cluster. T value is calculated according to the Figure 6.3 and referring to Pythagorean theorem.

Let V is the radio speed, R' is the simple sensor node communication range and r is the cell size.

Using the speed linear function $R' = VT$ from the function (3), T can be found as below:

$$T = \frac{R'}{V}$$

$$T = \frac{r\sqrt{2}}{V}$$

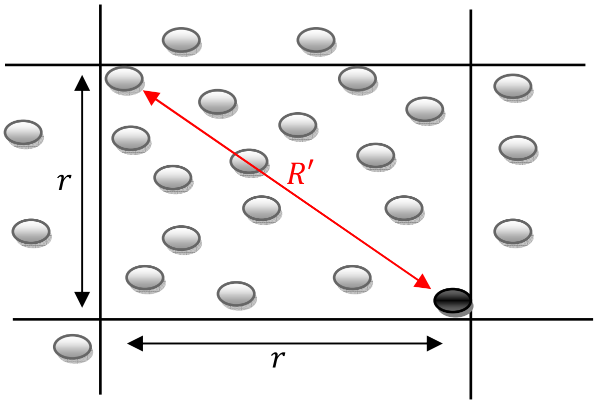


Figure 6.3: Grid cell (cluster).

This operation permits to reduce the number of the transmitted packets. Moreover, when an event is observed by several nodes, only one message will be transmitted to the sink and the other messages will be ignored.

6.4.2.3 Dynamic power threshold and cluster head management

As motioned above, the cluster head is responsible to disseminate the sensed data from the source sensor node to the sink witch consume more energy. Using a fixed energy threshold to determinate the cluster head permits for the selected node to acts as head only one time in its life and it will be the first node that will be died and like this one by one all the node will be died, as result the network performances decrease.

In order to make the proposal protocol more efficient in energy consumption and prolong the lifetime of the sensor nodes, we define the below formula:

$$New\_threshold = Old\_threshold - \frac{Old\_threshold}{K}$$

K is positive integer.

We define also two fixed threshold, $threshold\_max$ and $threshold\_min$. The $threshold\_max$ is the initial threshold, and the $threshold\_min$ is the lowest sufficient energy that permits the cluster head to advertise its energy exhaustion in order to select another one.

Initially the energy available of the first selected cluster head should be more than the $threshold\_max$, we put $Old\_threshold = threshold\_max$, after certain time, its energy decreases and when becomes inferior to the $threshold\_max$. The cluster head sends a message of *Select_newHeader*($threshold\_max$) in order to select new cluster head. The sensor nodes which have a residual energy higher than the threshold sent by the cluster head will replay by sending their energy available $Ack\_newLeader$ ($Residual\_ Energie$).

The sensor node which has the highest residual energy available will be selected as new cluster head. In the worst case, where no sensor node replies on the selected new header message, which means that all the sensor nodes have residual energy less than the specified threshold ($Old\_threshold$). In this case, the cluster head defines new threshold using the above formula and select new cluster head according to it.

However, if the residual energy of the current cluster head is less than or equal to the $threshold\_max$, it sends a *Death_Leader*($Old\_threshold$)) message to advertise its energy exhaustion, and the nodes will cooperate between them with the same manner to choose a new cluster head the node which has the highest residual energy.

6.4.2.4 An empty cluster

Generally in wireless sensor networks the nodes are deployed randomly. Therefore, the nodes density in the sensing field is variable.

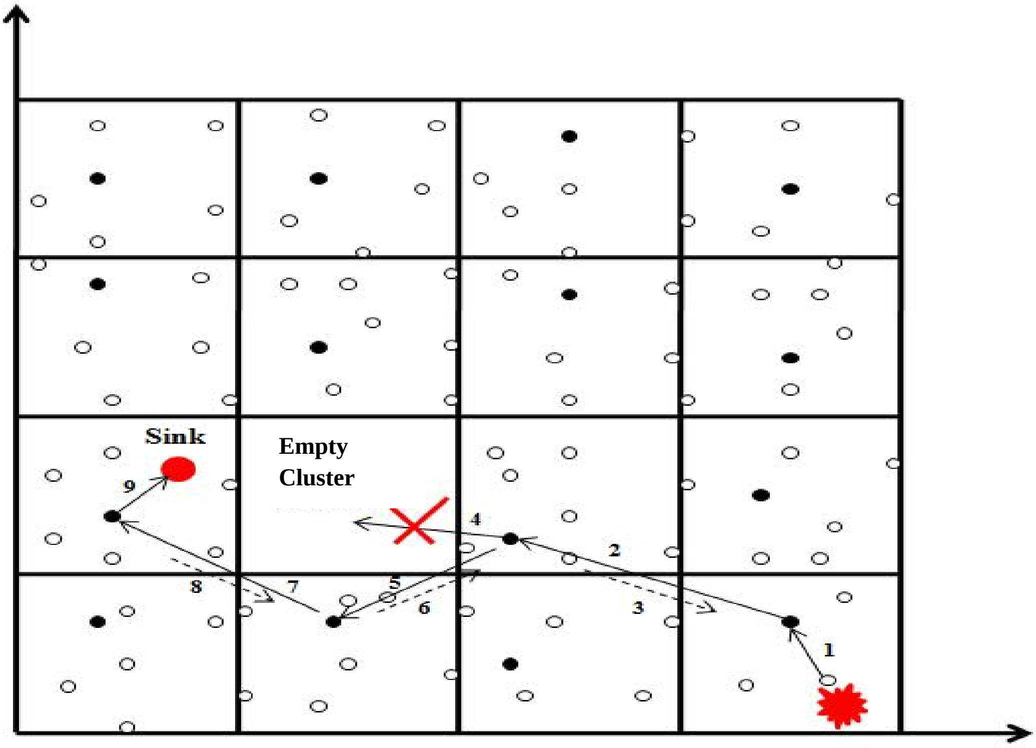


Figure 6.4: An empty cluster.

In our proposal, it is more possible, after the construction of the virtual grid, to find some cluster empty (Figure 6.4). Moreover, after a certain time all the sensors nodes of the same cluster can be died.

During the data dissemination process, the next hope can be carried out on an empty cluster. In this case the data dissemination cannot be done.

To solve this problem, we suggest that the cluster head of the next hope must delivery acknowledgement. Therefore, the source cluster head must await an Ack_recept message from the next selected hope. If any Ack_recept message has been received during the next T period, it will select another next head to which it will disseminate the sensing event.

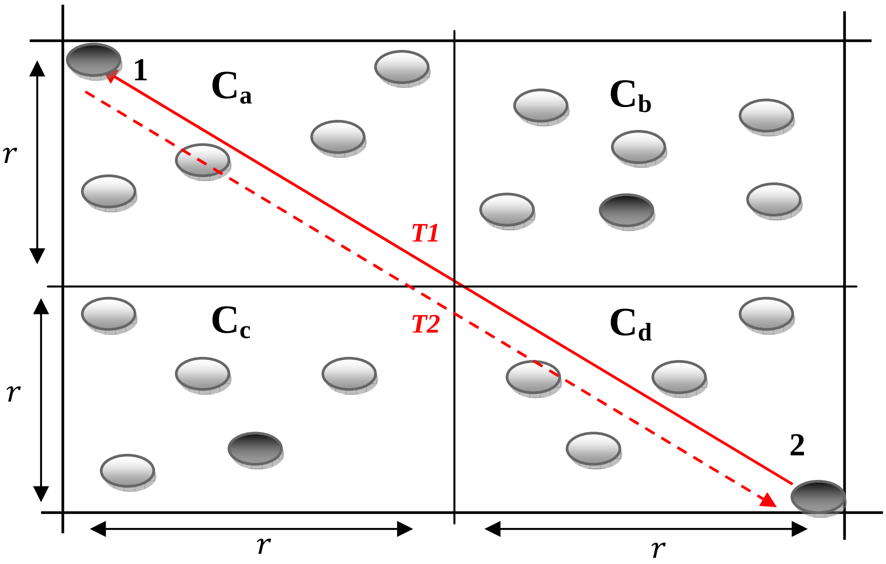


Figure 6.5: Needed waiting time.

The period T is the twice necessary time that a message traverses the distance between two most distant nodes in two neighboring clusters (Figure 6.5).
In the Figure 6.5 the far distant nodes are the node1 in the cluster C_a and the node 2 in the cluster C_d.

T is calculated mathematically as below:

$$T = T1 + T2 \qquad \textbf{(3)}$$

Using the speed linear function

$$T1 = T2 = \frac{R}{V} \qquad \textbf{(4)}$$

From (3) and (4), t can be calculated as following:

$$T = \frac{2r\sqrt{2}}{V} + \frac{2r\sqrt{2}}{V}$$

$$T = \frac{4r\sqrt{2}}{V}$$

6.5 Performances evaluation

This section is reserved to discuss the performance evaluation results of our proposal protocol. The evaluation has been carried out by simulation using Glomosim simulator [154]. To simulate the sensing data, the sensor nodes are randomly chosen to detect and send new sensing event during the simulation time. The sensing process follows the model of POISSON where 60 seconds value fixed as an interval average of this model.

Table 6.2: Simulation parameters.

Parameter	Default value	Variation interval
Number of nodes	100	100 - 700
Initial energy available(KW)	100	
Threshold_max(KW)	3	60
Threshold_min (KW)	1	
Bandwidth(Mbps)	2	
New event detection period (S)	60	1 - 60
Simulation time duration(M)	15	

This interval has been varied between 1 second and 60 seconds in order to simulate the network load and the energy consumption in case of many detected event occurred. In this simulation, energy consumption, response time or latency from the source node which detect the event to the sink node and traffic parameter have been evaluated according to different metrics. Moreover, the proposal protocol has been compared with Leach and Code using the same parameters in the same simulator.

The table 6.2 shows the parameters of our environment. The default dimension of the network is 1000x1000 m^2. The number of nodes can reach 700 nodes. We augment the number of nodes to observe the effect of the density on our solution.

6.5.1 Energy consumption

The energy consumption calculated using the same formula presented in the previous section 4.5.2.

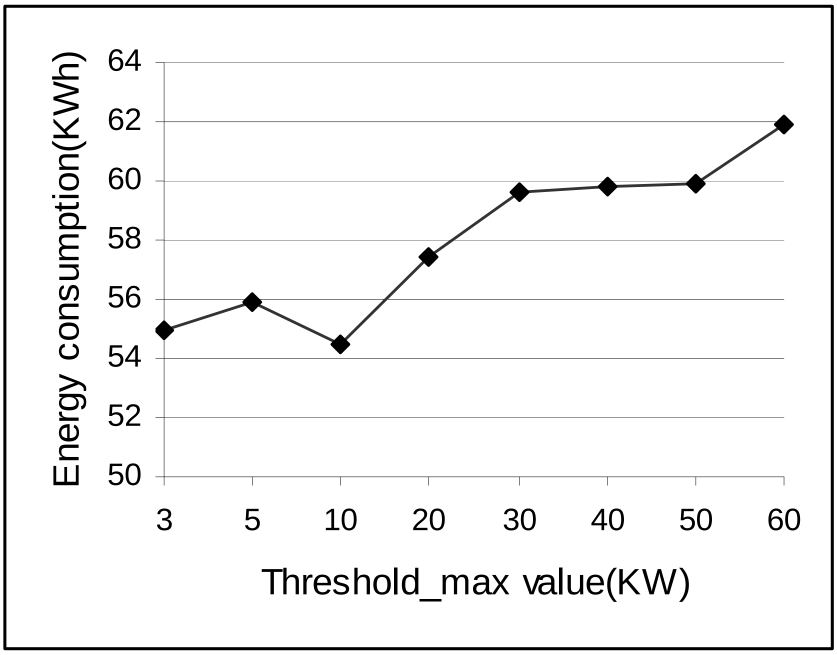


Figure 6.6: Energy consumption and power threshold.

Figure 6.6 represents the evolution of energy consumption according to the dynamic power threshold (threshold_max) related to our proposal. The aim of this experience is to determine the most optimal value of the power threshold_max that we can use it for the remaining simulation tests.

This experience shown clearly, that the most optimal value threshold is 10 KW. This value represents 10 % of an initial power100 KW. When the threshold_max increases the energy consumption increases also because the execution number of cluster head swaps procedure is very high and many control packets will be generated, therefore consumption will increased.

Figure 6.7 represents the evolution of energy consumption according to the density of sensor nodes. It permits also to evaluate the impact of scalability on energy consumption.
For LEACH the energy consumption is very high comparing with two others protocols. It remains stable for density of 100 and 500 nodes/Km2, and then increases when the density reaches 600 and 700 nodes/Km2. This is explained by the dynamic creation group's procedure which needs to send high number of control messages. Concerning protocol CODE and our proposal, the energy consumption increases in parallel with the density. This is caused by the flooding technique used by some nodes in network to communicate their co-ordinates.

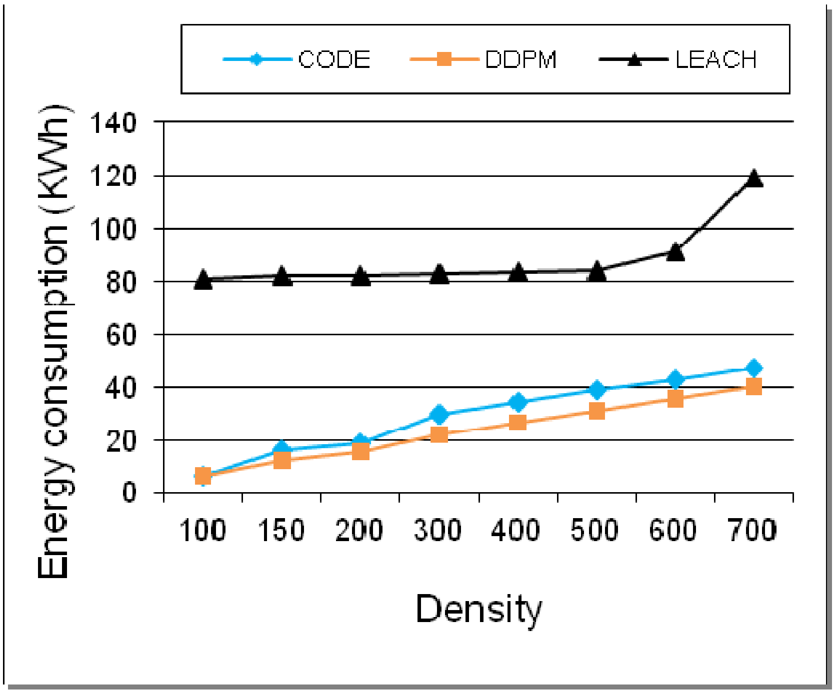


Figure 6.7: Energy consumption and density.

Comparing these protocols, CODE is more effective than LEACH, because the last one consumes more energy for the creation and the re-creation of the dynamic groups, whereas CODE uses a static virtual grid. However, more our protocol appeared more effective than CODE, because the initialization and transferred request phases used in CODE have been eliminated.

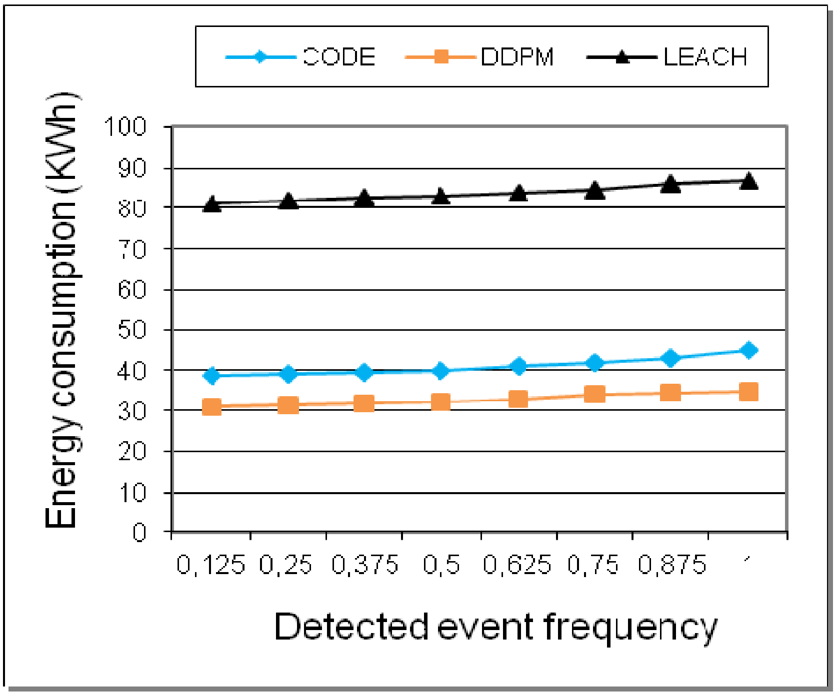


Figure 6.8: Energy consumption and detected event frequency.

The above figure (Figure 6.8) shows the evolution of energy consumption according to the detection event frequency.

For the three protocols, the energy consumption is not very affected by the number of detected event, which means that the most energy consumption provided from the protocol design.

6.5.2 Response time

The response time is the needed average duration to disseminate the detected data from the source to the sink node. The average time ($TpsAcc$) is calculated as below:

$$TpsAcc = \frac{\sum_{K=1}^{N} Resp\_time}{K}$$

$$Resp\_time = Sent\_time - Received\_time$$

Where:

- $Sent\_time$: is the sending data time.
- $Received\_time$: is the data receiving time by the sink node.
- $Resp\_time$: is the needed time for successful delivered data.
- N : is the number of successful delivered data.

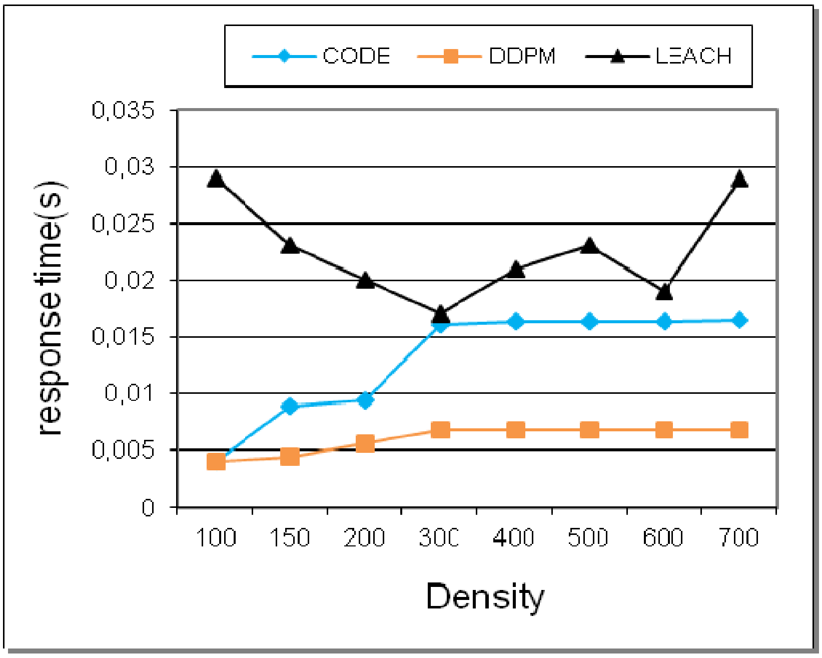


Figure 6.9: Response time and density.

Figure 6.9 represents the evolution of the response time according to the density of sensor nodes. The response time for LEACH protocol is unstable. This permits to say that the density does not have a direct influence on the response time and this instability is related to the protocol design.

Concerning the two other protocols, the response time increases slowly in the beginning and becomes stable. The highest response time is usually corresponds to LEACH protocol, this last uses the concept of period transmission which impacts data dissemination process. Our protocol appears more effective than CODE and gives the best response time

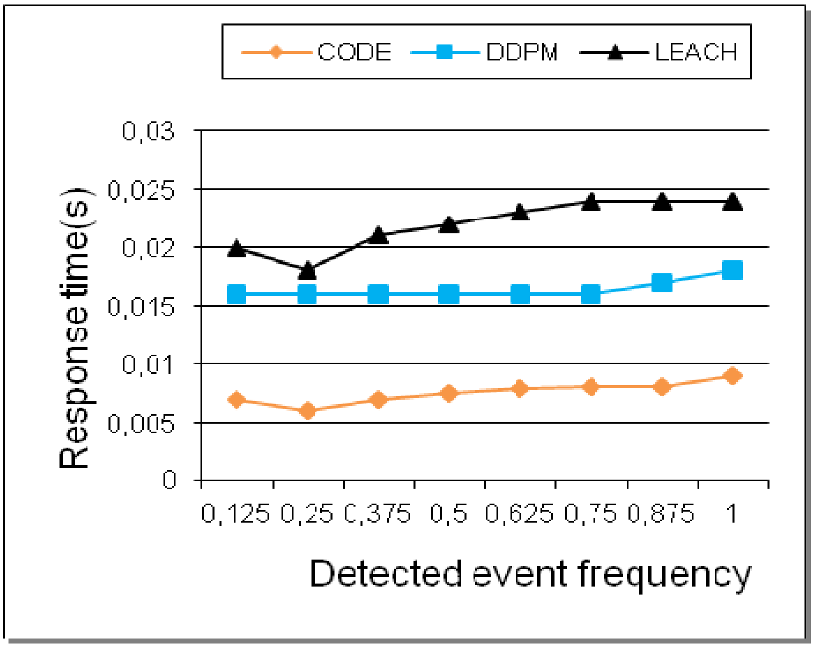


Figure 6.10: Response time and detected event frequency.

In Figure 6.10, we evaluate the behavior of the response time according to the detected event frequency. When the detection frequency increases the network overhead decreases also, therefore the response time or the latency will be increased. Consequently, the real time application will be influenced.

During this experience we notice that the protocol Leach gives the highest response time which increases in parallel with the detected event frequency. In the other hand, the response time in Code is not more influenced and remains stable until the value 0,75 request/second where the response time starts to increase. Moreover, the best response time is that given by our protocol, where it remains low and not influenced.

6.6 Conclusion

In this chapter we saw that the particular nature of the sensor networks, such as the limited lifetime of the sensors in consequence of their limited size, the multiplicity of the components and their performances require a specific mode of communication and represents considerable constraints.

According to the studied related works we noticed that each protocol has advantages and disadvantages, this study allowed us to understand the mechanism of data dissemination for wireless sensor networks, which helped us to propose a new solution that considers the requirements of the sensor networks.

The proposal protocol is based on a virtual grid structure, where each cell in the grid contains a sensor head responsible on the dissemination and the aggregation of the sensed data. This cluster head is selected periodically according to its energy available using dynamic power threshold. In this protocol we have considered only the detected events. Thus, sensed data are disseminated from the source sensor node to the sink.

In this chapter, we have considered statically deployed sink and sensors as a starting point to better understand the problems and challenges of data management in static sink wireless sensor networks. Top-level sensor nodes, especially, one-hop neighbors of static sink may suffer from congestion and quick energy depletion since they process more packets than lower-level nodes. Therefore, power management and clustering data dissemination are more critical in static environments because sink is always connected to same one-hop neighbors. Generally speaking, giving sink the ability to move or using mobile sensors as relay nodes can help avoid this bottleneck problem. In the next chapter, we focus on the mobility aspects in wireless sensor networks and study on data dissemination protocol supporting sink mobility.

Chapter 7

Data Dissemination in Mobile Sink Wireless Sensor Networks

7.1 Introduction

The well-known problem of a static-sink wireless sensor network is the uneven energy depletion phenomenon. Sensors near the sink deplete their battery power faster than those far apart due to their heavy overhead of relaying messages. Non-uniform energy consumption causes degraded network performance and shortens network lifetime. If all sensors around the sink run out of energy, the sink will be isolated from the network, then the entire network fails. There are significant advantages of having mobile sink in the network in terms of latency and energy consumption of information acquisition.
First of all, mobile sink alleviates the unbalanced energy consumption among sensor nodes since the data transmission load is shared among all the sensor nodes. In addition, mobile sink can remarkably reduce the mean distance (and also hop count) between sensor nodes and sink, basically resulting in energy saving and lower latency.

In this chapter, we present new sink mobility-based data dissemination protocol to minimize energy consumption. We define new sink mobility scheme in which sensor sink periodically moves towards any destination based on the data dissemination frequency calculated during the last period. The simulation result shows that the proposal protocol permits to reduce the energy consumption and prolong the network life.

The rest of this chapter is organized in the following way. Section 7.2 reviews a set of related works, presents the wanted advantageous of sink mobility, the different mobility forms and the possible sink mobility types. Section 7.3 presents the concept of our proposal. Section 7.4 evaluates the performance of our protocol. Section 7.5 concludes this chapter.

7.2 Mobility in wireless sensor networks

Recent researches [159], [10], [8], [9], [108], [37] have showed that the use of mobile elements can enhance connectivity and lifetime of wireless sensor networks. In many deployment scenarios, mobile entities already exist in the deployment area, such as firemen in an emergency response application, and buses in a traffic monitoring application.

The mobile nodes, which are capable of communicating with other nodes in the network, can address the connectivity problem by carrying information between isolated (disconnected) parts of wireless sensor networks. Since mobility has been proposed as another way for reducing the communication distance between sensors and sink in the literature, network lifetime can be improved with mobile devices by reducing multi-hop communication. Another important problem of static deployments is the bottleneck problem, which appears on the nodes close to the sink. As all the data is forwarded towards the sink, the average load on a sensor node increases with decreasing distance between the node and the sink [78]. Mobility also helps to solve this problem by deploying mobile sensor nodes and sinks in the network. In [196] the authors discuss how mobile elements improve the network lifetime. By adding only one mobile relay node which moves periodically within a two-hop radius of the sink, the lifetime of energy conserving routing can be doubled. Using mobile sink gives a lifetime improvement which is even better than a mobile relay node.

7.2.1 Mobility form

In wireless sensor networks, mobility can appear in three main forms [49]: mobility of the sensor nodes that sense the environment and transmit the related information, mobility of data sinks that gather the information from the network and forward data to the application, and mobility of the observed event.

- **Sensor Node Mobility:** this type of mobility is generally used in a livestock surveillance application (e.g. sensor nodes attached to cattle). The mobility of sensors can influence protocols at the networking layer. The network has to reorganize itself frequently enough to be able to function correctly. There are trade-offs between the frequency and speed of node movement on the one hand and the energy required to maintain a desired level of functionality in the network on the other hand [49].

- **Sink Mobility:** Since the sink is requester of the data, the mobility of sink may occur due to the mobility of end-users carrying sink nodes in the network. For example, in a disaster response application, a fireman requests event related data from sensor nodes while he is moving inside the network. Generally speaking, sink mobility is offered as a solution for uneven energy depletion of the sensor nodes deployed around the static sink. The sink mobility also has a great influence on protocols at the network layer. The network, possibly with the assistance of the mobile sink, must make provisions that the dissemination data actually follows and reaches the sink despite its movements.

- **Event Mobility:** The cause of the events or the objects to be tracked can be mobile in applications like event detection and in particular tracking scenarios. Battlefield monitoring explains well the mobile event scenario, where the task is to detect a moving tank in a border protection application and to observe it as it moves around. Since the location of the event changes over time, the sensor nodes sensing and reporting the event (source nodes) also change over time. The mobility of event [185] is highly application dependent, meaning that it can not be controlled (mostly also unpredictable) by the sensor network. Generally, event mobility appears in tracking applications such as animal tracking or military applications for tracking enemies.

Sensor node mobility and event mobility are not under consideration for this thesis. In this chapter, we focus only on sink mobility of sensor and we .consider the existing of only one sink. Several sinks and sensor nodes mobility will be the objective of our future works.

7.2.2 Sink mobility type

Mobility of a sink node can be classified as below:

- **Uncontrolled Mobility:** A sink node moves in a random fashion. The mobility of sink can be considered as uncontrollable (or random) if the direction and trajectory of the sink is not related with or determined by the data routing requirements. It is regulated according to the primary purpose of the user (i.e. people, vehicle) carrying the sink node in the sensor field. For example, in fire field monitoring, sink node can carried by fireman that moves according to his primary task. Therefore, the sink mobility is considered as uncontrollable in such a scenario. The main problem in this kind of scenarios is how to deliver data from a source node to sink when moves and changes its location.

- **Predictable Mobility:** Predictable mobility refers to the case when the motion is known but cannot be changed. However, this knowledge can be exploited to route data. Predictable sink mobility can improve energy efficiency of data transmission [125] by combining data relaying with predictable mobility. Sensors can predict the time of data transfer by utilizing the trajectory of the mobile node. Based on the predicted data transfer time, the sensors become active at the time of data transfer, otherwise they sleep until the time of data transfer comes to save energy. A representative example of predictable mobility is the vehicular mobility such as public transportation (i.e. train, bus). Such vehicles can act as mobile sink in wide area wireless sensor networks for applications such as safety road and pollution monitoring.

- **Controllable Mobility:** The mobility of the sink can be controlled by a user. Controllable mobility, like sink mounted on a robot [85], can be used as means for improving network connectivity and data dissemination tasks. Sink mobility is controlled usually for the purposes of avoiding hot spots around the sink and distributing energy consumption throughout the network evenly or enabling single-hop communication between the sink and the source nodes to avoid long-range communication. In these approaches of exploiting controlled mobility of sink, the problem is mostly about finding the optimal motion of the mobile sink to balance the energy consumption in the network or scheduling the mobile sink in real time to visit source nodes such that no sensor buffer overflow occurs and the data loss is avoided [40].

7.3 Energy-based Data Dissemination Protocol (EDDP) with mobile sink

In this section we give more detail about our solutions proposed in [112] and [111]. This solution enhances our previous solution [114] presented in the previous chapter where we consider the mobility of sink. Sink mobility is offered as a solution for uneven energy depletion phenomenon described in Section 7.2. Like [114], this protocol is based on an indexed virtual grid structure. Each cell represents cluster nodes with an elected cluster head. This protocol has two main phases:

- Setup phase: is considered as prerequisite for the second phase. In this primary phase the virtual grid has to build with selected cluster head for each cluster nodes that is represented by the grid cell.
- Dissemination phase: each sensor node sends its sensed data to its cluster head. The cluster head performs the necessary data aggregation with the same manner as presented in the previous chapter and disseminates the aggregated data toward the sink based on the next hop forwarding algorithm presented in Section 6.4.2 in the previous chapter. The cluster head is re-elected periodically and based on the energy available. The period does not fix and depends on the residual energy of current cluster head and the value of dynamic threshold (Section 6.4.2.1)

7.2.3 Environment model and hypothesis

We consider the same environment defined in Section 6.3, where large number of sensor nodes scattered randomly over the monitoring area. Sensor nodes are organized in cluster node and only one node acts as cluster head. All sensor nodes are stationary except the sink that can be mobile. Below we summarize our assumptions:

- Each sensor node has the possibility to calculate its own geographic location using location services such as GPS [187], [133], or other techniques such as triangulation.
- After deployment sensor nodes remain stationary.
- Sink has the capabilities to move toward any destination, the motion characteristics will be described in the below Section.
- The sensor nodes are stationary and have limited energy source.
- Sensor nodes send its sensed data toward the sink using a multiple hop communication.

7.2.4 Sink mobility scheme

Exploiting mobility of the sink could be considered as an interesting concept to enhance the network lifetime by avoiding excessive relaying overhead at nodes close to the static sinks. Our sink mobility model bases on three main points: 1) at what time the sink has to move, 2) in which manner the sink will move and 3) towards which position.

Table 7.1: Data dissemination frequency.

Cell ID	Data dissemination frequency
[2,1]	10
[1,1]	3
[2,3]	0
[0,1]	1

In our protocol, the sink mobility occurs periodically. In each period MTP (mobility time period) sink calculates its new destination and moves towards it randomly. The new destination is defined by the cell ID or the cell coordinates based on the data dissemination frequency. As shows in the table 7.1 below, the sink maintains in its cache the number of data dissemination by cell ID. The data dissemination frequency is number of data dissemination received during the previous period MTP. The new destination of sink is the cell that has the highest data dissemination frequency. When the sink reaches its destination it initializes its table of data dissemination frequency and stays in the new cell till next period. To inform the sensor nodes by the new destination, the mobile sink sends its new positions over the network. This mobility scheme permits to balance traffic load between the different cells. The sink moves always toward the cell that has more traffic.

Thus avoid transmitting big amount of data over the network, reduce energy consumption and improve the latency.

7.4 Performances evaluation

We study and discuss the performance of our protocol using simulation [154]. Sensor nodes are randomly chosen for sensing their data. We simulate several detect during the simulation time. Data detection process follows the model of POISSON where 60 seconds value fixed as an interval average of this model. We modify this interval between 1 second and 60 seconds in order to simulate the network overhead and the energy consumption in case of many detected event occurred. The below table (table 7.2) resumes the simulation parameters and their values. The default dimension of the network is $1000x1000m^2$. The number of nodes can reach 700 nodes. We increase this number to evaluate the impact of the density on the performances of our solution.

Table 7.2: Simulation parameters.

Parameter	Default value	Variation interval
Number of nodes	100	100 - 700
Initial energy available(KW)	100	
Threshold_max(KW)	3	60
Threshold_min (KW)	1	
Bandwidth(Kbps)	250	
New event detection period (S)	60	1 - 60
MTP (S)	60	
Speed Min(M/S)	0	
Speed Max(M/S)	1	
Simulation time duration(M)	15	

To move between the actual and the new calculated position, the mobile sink follows random waypoint mobility model. It moves randomly in the direction of the destination in a speed uniformly chosen between MOBILITY-MIN-SPEED and MOBILITY-MAX-SPEED (meter/sec). After it reaches its destination, the sink node stays there for mobility time period (MTP).

7.2.5 Evaluation parameters

In this simulation we evaluate the below defined parameters:

- **Density:** It is more important to evaluate the impact of this parameter on the network performance. This parameter is already defined in Section 4.5.1 of the fourth chapter.

- **Detection Event Frequency (DEF):** This parameter concerns the mobile sink. It represents the number of detection event per cluster during the mobility time period (MTP). In our mobility scheme, the mobile sink moves toward the cluster which has the maximum detection events during the actual MTP. Calculate the detection event frequency with this manner do not reflect the real detection event during the time. For example, if high detection event recorded from a given cluster node for each MTP. However, at the end of each MTP, the detection event frequency related to this cluster is not the maximum. Thus, mobile sink will never move toward this cluster. For this reason, we opt for the Moving Average formula [179], [180], [81], [36] and we calculate the moving average of DEF as below:

  $$MDEFi = \beta\, MDEFi^* + (1 - \beta)DEFi \quad \text{such as:}$$

  - $MDEFi$: is the moving average of the cluster node i for the new TMP period.
  - $MDEFi^*$: is the moving average of the cluster node i for the last TMP period.
  - $DEFi$: is the detection event frequency of the cluster node i for the new TMP period.
  - β: is weight constant, in our simulation $\beta = 0.5$, with this value we give the priority to the new of DEF and at the same time we consider the history of DEF of each cluster nodes.

  $MDEFi$ is calculated at the end of each MTP and thus the mobile sink moves toward the cluster that has the maximum value of $MDEF$

7.2.6 Simulation metrics

To evaluate the above parameters the below metrics:

- **Energy consumption**

  The energy consumption is calculated using the same formula defined in Section4.5.2 of the fourth chapter.

The energy consumption of a single operation of data transmission from sensor node to the sink can be quantified as below:

Let $Etx_{node_i(d)}$ is the amount of energy required for node i to transmit the sensed data d to its cluster head, $Etr_{cluster_{head(d)}}$, $Etx_{cluster_{head(d)}}$are respectively the energy required for cluster head to receive and transmit the same data d, and n is the number of cluster head involved in the data dissemination processes of the data d , thus the total energy consumption(E) of this operation is:

$$E = Etx_{node_i(d)} + Etr_{cluster_{head(d)}} + Etx_{cluster_{head(d)}} + n \times (Etr_{cluster_{head(d)}} + Etx_{cluster_{head(d)}})$$

- **Response time**

  As defined in Section6.5.2, response time is an average duration of all the successful disseminated data from the source to the sink node. The average response time ($TpsAcc$) is calculated using the same formula:

$$TpsAcc = \frac{\sum_{K=1}^{N} Resp\_time}{K}$$

$$Resp\_time = Sent\_time - Received\_time$$

Where:

$-$ $Sent\_time$: is the sending data time.

$-$ $Received\_time$: is the data receiving time by the sink node.

$-$ $Resp\_time$: is the needed time for successful delivered data.

$-$ N : is the number of successful delivered data.

7.2.7 Simulation results

a) Energy consumption

We calculate energy consumption as defined above. We use the same threshold_max value (threshold_max= 10KW) calculated in the first experience of the last chapter (Section6.5.1).

Figure7.1 below represents the evolution of energy consumption according to the density of sensor nodes. It permits also to evaluate the impact of scalability on energy consumption.

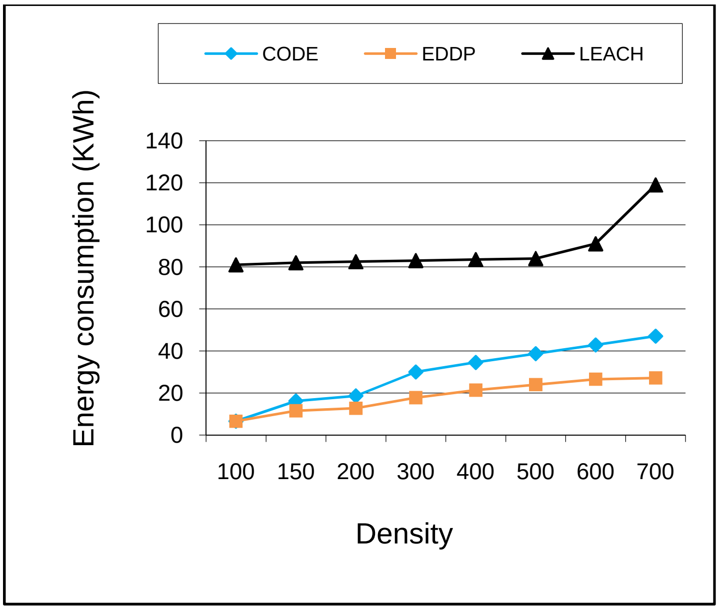


Figure 7.1: Energy consumption and density.

For LEACH the energy consumption is very high comparing with two others protocols. It remains stable for density of 100 and 500 nodes/Km2, and then increases when the density reaches 600 and 700 nodes/Km2. This is explained by the dynamic creation group's procedure which needs to send high number of control messages. Concerning protocol CODE and our proposal, the energy consumption increases in parallel with the density. This is caused by the flooding technique used by some nodes in network to communicate their co-ordinates.

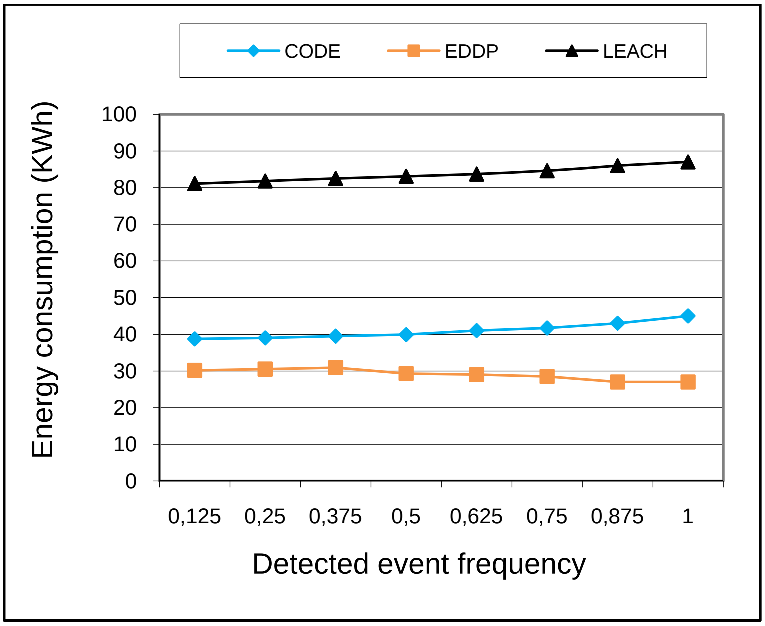


Figure 7.2: Energy consumption and detected event frequency.

Comparing these protocols, CODE is more effective than LEACH, because the last one consumes more energy for the creation and the re-creation of the dynamic groups, whereas CODE uses a static virtual grid. However, more our protocol appeared more effective than CODE, because the initialization and transferred request phases used in CODE have been eliminated.

Figure 7.2 shows the evolution of energy consumption according to the detection event frequency.

For the three protocols, the energy consumption is not very affected by the number of detected event, which means that the most energy consumption is provided from the protocol design. Energy consumption increases slowly with the sensing data increasing. In our protocol, sensor sink can move towards the cluster node that provides more data, thus optimize energy and latency also. Moreover, using dynamic energy threshold to select cluster head improves load balancing and prolong sensor node's life time.

b) Response time

Figure 7.3 represents the evolution of the response time according to the density of sensor nodes. The response time for LEACH protocol is unstable. This permits to say that the density does not have a direct influence on the response time and this instability is related to the protocol design.

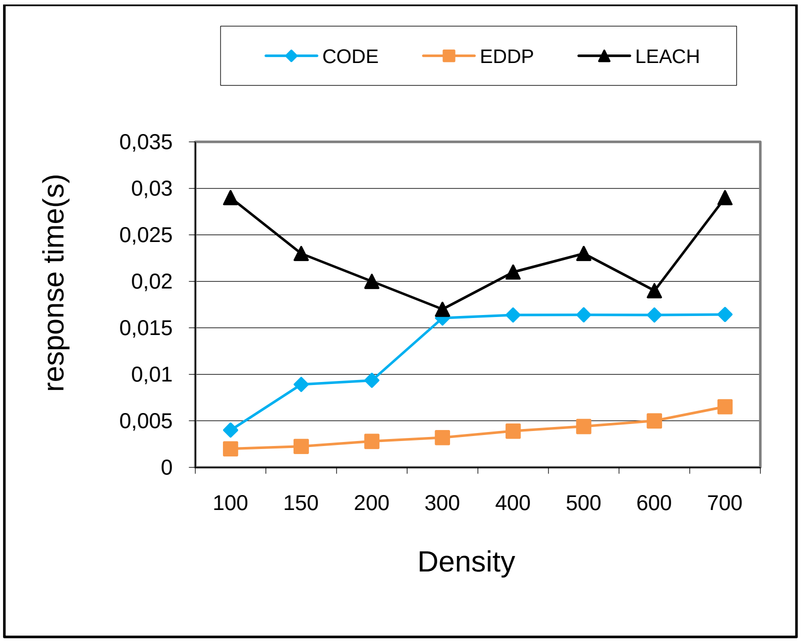


Figure 7.3: Response time and density.

Concerning the two other protocols, the response time increases slowly in the beginning and becomes stable. The highest response time is usually corresponds to LEACH protocol, this last uses the concept of period transmission which impacts data dissemination process. Our protocol appears more effective than CODE and gives the best response time.

In the experience presented in Figure 7.4, we evaluate the behavior of the response time according to the detected event frequency. When the detection frequency increases the network overhead decreases also, therefore the response time or the latency will be increased. Consequently, the real time application will be influenced.

During this experience we notice that the protocol Leach gives the highest response time which increases in parallel with the detected event frequency. In the other hand, the response time in Code is not more influenced and remains stable until the value 0,75 request/second where the response time starts to increase.

Moreover, the best response time is that given by our protocol, where it remains low and not influenced by network density. In EDDP, sink moves periodically towards the sensor nodes that provide more data dissemination and thus, improves energy consumption both for source sensor node and the other nodes participated in data dissemination process.

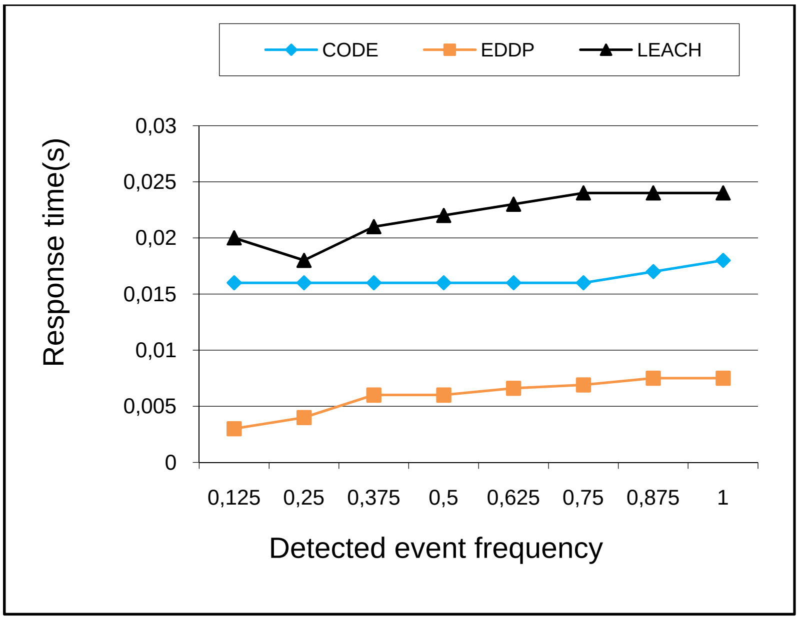


Figure 7.4: Response time and detected event frequency.

7.5 Conclusion

In this chapter we have presented an extended version of our previous data dissemination protocol presented in the last chapter by considering the sink mobility. This solution is cluster based, the cluster head is selected periodically according to its available energy and dynamic power threshold. In this solution we have considered the mobility of the sink. In our mobility model, the sink moves periodically and randomly towards a specified destination that has chosen based on data dissemination frequency. The simulation results show that the sink mobility model ameliorates the performances of the network.

Chapter 8
Conclusion and future research directions

8.1 Conclusion

Data management is the process by which data are disseminated in the sensor network. One of the major tasks of wireless sensor networks is to disseminate useful information from data sources to users at the minimum power consumption where sensor nodes must operate on limited power sources for extended lifetime. Especially, how to achieve power-efficiency and multi-hop communication in these applications are the major concerns since wireless communication often dominates the energy dissipation in wireless sensor networks.

The main research focus of this thesis was to investigate how node to sink communications could be achieved efficiently in a static and mobile sink wireless sensor network. The overall objective of the protocols designed in this dissertation was to maximize the functionality of MAC and network layers through the design of efficient, distributed and scalable algorithms. The term efficient can be interpreted in many ways depending on the emphasis placed by the application being considered. Although the majority of wireless sensor networks related research focuses primarily on techniques to extend the lifetime of the network, it is more important to know that network's lifetime is not the only issue that is of big concern to the end-user. Service quality requirements, which could refer to parameters such as latency depending on the application requirements, are definitely very important for end-users in case of real time applications. Hence, the research approach of this thesis incorporates the term efficiency with two aspects: energy-efficiency and latency and thus how to achieve data dissemination with minimum energy consumption and low latency.

To achieve the target of this thesis, we have divided our researches into two main parts. We focused in the first part on the problem of energy dissipation. We have studied in depth on the sources of energy waste and we have concluded that the major sources of energy consumption occurred in the MAC layer. These sources are collision, overhearing, idle listening and packet overhead.

Medium Access Control (MAC) protocols must perform the functionality required by the application while utilizing the limited resources available on sensor nodes. Limited energy resources place strict limits on the operations that a sensor node may accomplish and differentiate sensor networks from other networks. Application and protocol designers must utilize the hardware resources of the sensor nodes judiciously to conserve energy and prolong the network's lifetime. In this part and in order to ameliorate energy consumption and minimize latency, we have proposed two MAC protocols:

- The first protocol is synchronous based, it organizes sensor nodes under pair an odd clusters activating in load balancing mode. Each sensor node has its own active/sleep and send/receive periods. We proposed a mechanism for load balancing between clusters in the network and between sensors in each cluster. Although the energy consumption in the network is more improved, this mechanism has great negative influence on the latency. Thus, protocol is more recommended for no real time applications.

- The second protocol proposed to overcome the weakness of the first proposal. This protocol is asynchronous based. We proposed a mechanism to wake up the destination node and minimize the overhead. This mechanism is based on short preamble containing the Id of the destination node. We proposed another mechanism to avoid the problem of hidden host, this mechanism is used only if the hidden host exists. Simulation results show that this protocol outperforms the other protocols in terms of network energy consumption and latency.

In the second part of this thesis, we focused on the data dissemination issue. Exactly, we studied the issue of data sending from source node to the sink. In wireless sensor network, the sensed data need to be transmitted to the sink for further analysis, management, and control. Therefore, a data dissemination protocol is required to provide effective data transmission from sensor nodes to the sink. Moreover, it should guarantee successful transmission from nodes to the sink. Accordingly, we reviewed the state of the art data dissemination methods in wireless sensor networks. We classified the studied protocols based on the number and the type of sink (mobile or stationary) and we provided the below solutions:

- To achieve reliable data dissemination of events from source to the sink, we proposed a data dissemination protocol, namely Clustering Data Dissemination Protocol (CDDP). In this protocol, sensor field is considered as virtual grid and each grid cell represents a cluster node. Data dissemination performed locally between sensor node and its cluster head and externally between the cluster heads of different cluster node. In this protocol we consider that the sensor nodes and sink are stationary. We proposed mechanism to select the next best cluster head regarding the sink position. In CDDP, the cluster head role is rotated periodically among the sensor nodes in the same cluster. This mechanism requires sending more control packets and increases the network overhead. To overcome these weaknesses, we extended CDDP in another proposal namely Data Dissemination and Power Management (DDPM). In DDPM we proposed a mechanism to elect a cluster head. This mechanism based on dynamic energy threshold and the residual energy available at each node. We evaluated the performance of this protocol, the simulation results show that DDPM improves the energy consumption and reduces the latency.

- In high delivery data rate, static sink may pose several problems. The sensor nodes, which closer to the sink, have a heavier workload than their peers, as they are called to relay traffic that they do not generate. This additional burden curtails their lifetime, disconnecting the base station from the rest of the network and therefore making the network useless. This is known as the hot spot problem. The one-hop neighbors of the sink are typically more affected by this problem. To overcome this problem, we proposed data dissemination protocol considering the sink mobility. In this protocol namely Energy-based Data Dissemination Protocol (EDDP), we proposed new sink mobility scheme. This mobile sink moves periodically based on the data dissemination frequency of each cluster nodes. The mobile sink determines its destination based on the maximum data dissemination frequency recorded during the last period. The simulation results show that both energy and latency are improved.

Our hypothesis was that addressing an efficient data dissemination with energy saving requires special attention calling for both MAC and network layers. Looking back at our research objective mentioned in Chapter1 Section1.4 and Section1.5 and here at our contributions, it can be concluded that we have addressed both effectiveness and efficiency for data dissemination in wireless sensor networks by putting special attention on energy saving and latency. Therefore, we have demonstrated, with the help of several protocols presented in this thesis that our original hypothesis was valid.

Having summarized the contributions, in the next section we provide a list of potential future research directions.

8.2 Future research directions

As part of the future work, it is more desirable to test and integrate the presented methods on real, large scale wireless sensor network deployments. Moreover, further improvements can be incorporated to enhance and extend the presented results. In the following we present a list of possible future directions:

- The protocol proposed in chapter 4 gives good performances in term of energy consumption, nerveless it suffers from latency due to the large size of cluster nodes and the active/standby mode of its pair and old clusters. Using sub-cluster may overcome this problem and improve the latency. Therefore, several sensor nodes in the same cluster may have the sending mode in the same period and send their data in the same time. Thus, data transmission's delay could be ameliorated. However, the impact of the sub-cluster creation process on the network performances has to be well studied and investigated.

- Hybrid MAC protocol combines between the synchronous and asynchronous schemes while compensating for their disadvantages under different traffic load conditions. A hybrid MAC protocol adapts its behavior to the level of traffic loads which can be presented in different forms of the sensed traffic intensity in real time. Combining between the protocols proposed in chapter4 and chapter5 and design an hybrid MAC protocol seem an ideal approach to achieve optimal performance on Medium Access Control Protocols for Wireless Sensor Networks

- A typical way of extracting information from a sensor network is to disseminate queries from sink node to sensor nodes, asking them to send data which have the properties specified in the queries. The main consideration in designing query dissemination algorithms is to efficiently forward queries from sink node to sensor nodes. The extension of the protocols presented in chapter 6 and chapter 7 by considering the users' queries is more important issue for several applications. It is vital to determine the manner by-which the sinks need to disseminate the query to reach all sensor nodes in the area of interest.

- Sensor nodes near to the single sink deplete their battery power faster than those far apart due to their heavy overhead of relaying messages. Non-uniform energy consumption causes degraded network performance and shortens network's life time. Extending the protocols presented in chapter 6 and chapter 7 by having multiple sinks in the network gives advantages such as energy efficiency, reliability and alleviation of the hotspot problem. On the other side, it adds additional requirements. The main concern in a multi-sink deployment is balancing the network load by partitioning the network between sinks. It is also important to balance the load between sensor nodes in each partition.

This thesis has proposed a class of protocols and algorithms which are designed for handling data management in wireless sensor networks and are able to disseminate data efficiently in the network with the presence of static and mobile sink. We have illustrated how energy saving maybe achieved at the MAC layer and how data may be disseminated efficiently at the network layer. It is hoped that the protocols designed in this dissertation can be helpful to get a few steps closer to real sensor network applications.

Bibliography

[1] A. Bachir, M. Dohler, T.Watteyne, and K. Leung, “MAC essentials for wireless sensor networks,” IEEE Commun. Surv. Tutorials, vol.12, no.2, pp. 222–248, 2010.

[2] A. Chakrabarti, A, Sabharwal, and B. Aazhang, “Using predictable observer mobility for power efficient design of sensor networks,” In Proceedings of the 2nd international conference on Information processing in sensor networks (IPSN'03), 2003, pp. 129-145.

[3] A. D. Domenico, E. C. Strinati, and M.G.D. Benedetto, “A Survey on MAC Strategies for Cognitive Radio Networks,” IEEE Communications Surveys & Tutorials, Vol. 14, No. 1, pp. 21-44, First Quarter 2012.

[4] A. E. Tumer and M. Gunduz, “Energy-efficient and fast data gathering protocols for indoor wireless sensor networks,” Sensors, vol. 10, 2010, pp. 8054–8069.

[5] A. El-Hoiydi, “Aloha with Preamble Sampling for Sporadic Traffic in Ad-hoc Wireless Sensor Networks,” in Proceedings of IEEE International Conference on Communications, April 2002.

[6] A. El-Hoiydi, “Spatial TDMA and CSMA with preamble sampling for low power ad hoc wireless sensor networks,” Proceedings of ISCC 2002, Seventh International Symposium on Computers and Communications, Pages:685 - 692, 1-4 July 2002.

[7] A. El-Hoiydi, J. D. Decotignie, “WiseMAC: an ultra low power MAC protocol for multi-hop wireless sensor networks,” In Proceedings of the International Workshop on Algorithmic Aspects of Wireless Sensor Networks (Algosensors), Lecture Notes in Computer Science, pp. 18–31, 2004. ISBN 3-540-22476-9.

[8] A. Gonga, O. Landsiedel, and M. Johansson, “MobiSense: Power-Efficient Micro-Mobility in Wireless Sensor Networks,” In Proceedings of the 7th IEEE International Conference on Distributed Computing in Sensor Systems, IEEE DCOSS’11, Barcelona, Spain, 2011, pp. 1-8. URL: http://www.cse.chalmers.se/~olafl/papers/2011-06-dcoss-gonga-mobisense.pdf.

[9] A. Howard, M. Mataric, and G. Sukhatme, “Mobile Sensor Network Deployment using Potential Fields: A Distributed,” In Proceedings of the 6th International Symposium on Distributed Autonomous Robotics Systems (DARS02), June 2002.

[10] A. Karmouch and S. Hashish, “Deployment-based Solution for Prolonging Network Lifetime in Sensor Networks,” In Proceedings of IFIP International Federation for Information Processing, vol. 264, pp. 85-96, 2010.

[11] A. Kinalis, S. Nikoletseas, D. Patroumpa, J. Rolim, “Biased Sink Mobility with Adaptive Stop Times for Low Latency Data Collection in Sensor Networks,” GLOBECOM '09, pp.1-6, 2009.

[12] A. Kinalis, S. Nikoletseas, D. Patroumpa, J. Rolim, “Biased Sink Mobility with Adaptive Stop Times for Low Latency Data Collection in Sensor Networks,” Information Fusion, Elsevier, pp. 1-6, April 2012.

[13] A. Manjeshwar and D. P. Agarwal, "APTEEN: A hybrid protocol for efficient routing and comprehensive information retrieval in wireless sensor networks," Parallel and Distributed Processing Symposium., Proceedings International, pp. 195-202, 2002.

[14] A. Manjeshwar and D. P. Agarwal, "TEEN: a routing protocol for enhanced efficiency in wireless sensor networks," In 1st International Workshop on Parallel and Distributed Computing Issues in Wireless Networks and Mobile Computing, April 2001.

[15] A. Milenkovic, C. Otto, and E. Jovanov, "Wireless sensor networks for personal health monitoring: Issues and an implementation," Computer Communications, vol. 29, pp.2521-2533, August 2006.

[16] A. S. K. Pathan and C. S. Hong, "An Efficient Scheme for Secure Data Transmission in Wireless Sensor Networks," Proceedings of the 17th Joint Conference on Communication & Information (JCCI 2007), 2-4 May 2007, PyeongChang, Korea, p. 48 (CD).

[17] A. Sankar and Z. Liu, "Maximum lifetime routing in wireless ad-hoc networks," in Proc. IEEE INFOCOM, vol.2, 2004, pp.1089–1097.

[18] A. Woo and D. Culler, "A transmission control scheme for media access in sensor networks," In ACM MobiCom, pp. 221-235, 2001.

[19] B. Chen, K. Jamieson, H. Balakrishnan, and R. Morris, "Span: An energy-efficient coordination algorithm for topology maintenance in Ad Hoc wireless networks," in Proceedings of the Seventh Annual International Conference on Mobile Computing and Networking, Rome, Italy, July 2001.

[20] B. Krishnamachari, "Medium access and sleep scheduling," Networking wireless sensors, Book chapter, pp. 82–102, 2005.

[21] B. Krishnamachari, D. Estrin, and S. Wicker, "The impact of data aggregation in wireless sensor networks," In Proceedings of the 22nd International Conference on Distributed Computing Systems Workshops, 2002.

[22] C. C. Enz, A. El-Hoiydi, J. D. Decotignie, V. Peiris, "WiseNET: An Ultralow-Power Wireless Sensor Network Solution," IEEE Computer,vol. 37, Issue. 8, pp. 62-70, August 2004.

[23] C. E. Perkins and E. M. Royer, "Ad-Hoc On Demand Distance Vector Routing", In Proceedings Workshop Mobile Computing Systems and Applications, Feb. 1999.

[24] C. Guo, Lizhi C. Zhong, and J. M. Rabaey, "Low Power Distributed MAC for Ad Hoc Sensor Radio Networks," in Proceedings of IEEE GlobeCom, vol. 5, pp. 2944-2948, 2001.

[25] C. Intanagonwiwat, R. Govindan and D. Estrin, "Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks," In Proceedings of the Sixth Annual International Conference on Mobile Computing and Networks (MobiCOM 2000), pp. 56-67, August 2000.

[26] C. Intanagonwiwat, R. Govindan, D. Estrin, J. Heidemann, F. Silva, "Directed Diffusion for Wireless Sensor Networking," IEEE/ACM Transaction Networking 2003, vol.11, no.1, pp. 2-16, February 2003.

[27] C. J. Merlin and W. B. Heinzelman, “Schedule Adaptation of LowPower-Listening Protocols for Wireless Sensor Networks,” IEEE Transactions on Mobile Computing, May 2010, Vol. 9, Issue: 5, pp. 672-685.

[28] C. J. Ritsema, H. Kuipers, L. Kleiboer, E. Elsen van den, K. Ostindie, J. G. Wesseling, J.-W. Wolthuis, and P. Havinga, “A new wireless underground network system for continuous monitoring of soil water contents,” Water Resources Research, vol. 45, pp.1-9, 2009.

[29] C. K. Nguyen, and A. Kumar, “An energy-aware medium-access-control protocol with frequent sleeps for wireless sensor networks,” In Proceedings of the 10th IEEE Symposium on Computers and Communications (ISCC ’05), IEEE, Washington, D.C., 2005, pp. 386–391.

[30] C. K. Nguyen, and A. Kumar, “Medium access control with adjustable sleeps for wireless sensor networks,” In Proceedings of the 11th IEEE Symposium on Computers and Communication (ISCC ’06), IEEE, Washington, D.C., 2006, pp. 270-276.

[31] C. Moser, L. Thiele, D. Brunelli, and L. Benini, “Adaptive power management for environmentally powered systems,” IEEE Transactions on Computers, vol. 59, 2010, pp. 478–491.

[32] C. P. Low, C. Fang, J.M. Ng, and Y.H. Ang, “Efficient load-balanced clustering algorithms for wireless sensor networks,” Computer Communications Journal, Elsevier, vol. 31, March 2008. ISSN 0140-3664. doi: 10.1016/j.comcom.2007.10.020, pp. 750–759.

[33] C. S. Park, K. W. Lee, Y. S. Kim, and S. J. Ko, “A Route Maintaining Algorithm using Neighbor Table for Mobile Sinks,” Wireless Networks Journal, Volume 15, Issue 4, May 2009, pp. 541-551.

[34] C. Schurgers, V. Tsiatsis, S. Ganeriwal, M. Srivastava, “Optimizing sensor networks in the energy-latency-density design space,” IEEE Transactions on Mobile Computing, vol. 1, no.1, pp. 70-80, 2002.

[35] D. B. Johnson and D. A. Maltz, “Dynamic Source Routing in Ad Hoc Wireless Networks,” Mobile Computing, chapter 5, pp. 153-181, Kluwer Academic Publishers, 1996.

[36] D. C. Montgomery, “Introduction to Statistical Quality Control,” Third Edition, New York: John Wiley & Sons, Inc. 1996.

[37] D. Lee and K. Chung, “RA-MAC: An Energy Efficient and Low Latency MAC Protocol Using RTS Aggregation for Wireless Sensor Networks,” International Conference on Advanced Technologies for Communications (ATC), Hanoi, Oct. 6-9, 2008, pp 150-153.

[38] D. Luo, D. Zuo, and X. Yang, “An optimal sink selection scheme for multi-sink wireless sensor networks,” In Proceedings of the International Conference on Computer Science and Information Technology, ICCSIT’08, September 2008. doi: 10.1109/ICCSIT.2008.161, pp. 544 –548.

[39] D. Turgut and L. Boloni, “Heuristic Approaches for Transmission Scheduling in Sensor Networks with Multiple Mobile Sinks,” The Computer Journal, vol. 54, issue. 3, December2009. doi: 10.1093/comjnl/bxp110, pp. 332–344.

[40] E. Ekici, Y. Gu, and D. Bozdag, "Mobility-Based Communication in Wireless Sensor Networks," IEEE Communications Magazine, vol. 44, issue. 7, September 2006, doi: 10.1109/MCOM.2006.1668382, pp. 56–62.

[41] E. Felemban, C. Lee and E. Ekici, "MMSPEED: Multipath Multi-SPEED Protocol for QoS Guarantee of Reliability and Timeliness in Wireless Sensor Networks," IEEE Transactions on Mobile Computing, vol. 5, no. 6, pp. 738-754, June 2006.

[42] E. Felemban, C. Lee, E. Ekici, R. Boder and S. Vural,"Probabilistic QoS Guarantee in Reliability and Timeliness Domains in WirelessSensor Networks" In Proceedings of IEEE INFOCOM, vol. 4, pp. 2646- 2657,March 13-17, 2005.

[43] E. I. Oyman and C. Ersoy, "Multiple sink network design problem in large scale wireless sensor networks," In Proceeding of IEEE International Conference on Communications (ICC 2004), volume 6, IEEE, 2004. ISBN 0-78038533-0. doi: 10.1109/ICC.2004.1313226, pp. 3663–3667.

[44] E. Shih, S. Cho, N. Ickes, R. Min, A. Sinha, A. Wang, and A. Chandrakasan, "Physical layer driven protocol and algorithm design for energy-efficient wireless sensor networks," in Proceedings of the Seventh Annual International Conference on Mobile Computing and Networking, Rome, Italy, pp. 272-287, July 2001.

[45] E. Y. Lin, J. Rabaey, and A. Wolisz, "Power-Efficient Rendez-vous Schemes for Dense Wireless Sensor Networks," in IEEE International Conference on Communications, Vol. 7, June 20-24, 2004, Paris, France, pp. 3769-3776.

[46] F. Wang and J. Liu, "Networked Wireless Sensor Data Collection: Issues, Challenges, and Approaches," IEEE Communications Surveys & Tutorials, Volume: 13, Issue: 4, Fourth Quarter 2011, pp. 673-687.

[47] G. Gupta and M. Younis, "Load-balanced clustering of wireless sensor networks," In Proceedings of the 2003 IEEE International Conference on Communications, ICC'03, vol. 3, may 2003. doi: 10.1109/ICC.2003.1203919, pp.1848–1852.

[48] G. P. Halkes, T. Van Dam, and K. G. Langendoen, "Comparing energy-saving MAC protocols for wireless sensor networks," Mobile Networks and Applications Journal, Volume 10 Issue 5, pp. 783–791, October 2005.

[49] H. Karl and A. Willig. Protocols and Architectures for Wireless Sensor Networks. John Wiley & Sons, Ltd, Chichester, UK, ISBN 9780470095126, April 2005.

[50] H. Dai and R. Han, "A node-centric load balancing algorithm for wireless sensor networks," In Proceedings of the 2003 IEEE Global Telecommunications Conference, GLOBECOM'03, vol. 1, December 2003. doi: 10.1109/GLOCOM.2003.1258297, pp. 548–552.

[51] H. Luo, F. Ye, J. Cheng , S. Lu and L. Zhang, "TTDD: two-tier data dissemination in large-scale wireless sensor networks," Kluwer Academic Publishers' Wireless Networks, vol. 11, issue 1-2, pp. 161-175, January 2005.

[52] H. S. Kim, T. F. Abdelzaher, and W. H. Kwon, "Minimum-energy asynchronous dissemination to mobile sinks in wireless sensor networks," In Proceeding of Embedded Networked Sensor Systems (SenSys'03), 2003, pp. 193-204.

[53] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, “RTP: A transport protocol for real-time applications,” Audio-Video Transport Working Group, RFC 1889, Jan. 1996. Available at: http://www.ietf.org/rfc/rfc1889.txt [last accessed: 15 August, 2013]

[54] H. Wang, X. Zhang, F. Nat-Abdesselam, and A. A. Khokhar, “DPSMAC: An Asynchronous MAC Protocol for Wireless Sensor Networks,” In Proc. 14th International Conference of High Performance Computing (HiPC’07), Springer-Verlag Berlin, Heidelberg, 2007, pp. 393-404.

[55] H. Xuan and S. Lee, “A Coordination-Based Data Dissemination Protocol for Wireless Sensor Networks,” Proceedings of the Sensor Networks and Information Processing Conference, pp. 13-18, December 2004.

[56] I. Chatzigiannakis, A. Kinalis, S. Nikoletseas, “ Sink Mobility Protocols For Data collection in Wireless Sensor Networks,” In Proceedings of The International Workshop On Mobility Management and Wireless Access, MobiWac '06, pp. 52-59 2006.

[57] I. Chatzigiannakis, A. Kinalis, S. Nikoletseas, “Efficient data propagation strategies in wireless sensor networks using a single mobile sink ,” Computer Communications, vol. 31, no. 5, pp. 896-914, March 2008.

[58] I. Chatzigiannakis, A. Kinalis, S. Nikoletseas, “Wireless sensor networks protocols for efficient collision avoidance in multi-path data propagation,” In Proceedings of the ACM International Workshop on Performance Evaluation of Wireless Ad Hoc, Sensor, and Ubiquitous Networks, 2004, pp. 8–16.

[59] I. Demirkol and C. Ersoy, “Energy and delay optimized contention for wireless sensor networks,” Computer Networks: The International Journal of Computer and Telecommunications Networking, August 2009, vol. 53, issue. 12, pp. 2106-2119.

[60] I. Demirkol, C. Ersoy, and F. Alagoz, "MAC protocols for wireless sensor networks: A survey," IEEE Communications Magazine, vol. 44, issue. 4, pp. 115-121. September 2006.

[61] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, “A survey on sensor networks,” IEEE Communications Magazine, vol. 40, issue. 8, 2002, pp. 102–114.

[62] I. Gragopoulos, I. Tsetsinas, E. Karapistoli, and F. N. Paulidou, “Performance Study of a New MAC Algorithm to Reduce Energy and Delay in Wireless Sensor Networks,” New Technologies, Mobility and Security, Springer, 2007, pp. 39-51.

[63] I. Joe and H. Ryu, “A Patterned Preamble MAC Protocol for Wireless Sensor Networks,” IEEE International Conference on Computer Communications and Networks (ICCCN07), Honolulu, HI, 2007, pp. 1285-1290.

[64] I. Krontiris, Z. Benenson, T. Giannetsos, F. Freiling, and T. Dimitriou, “Cooperative intrusion detection in wireless sensor networks,” Wireless Sensor Networks, Lecture Notes in Computer Science, pp. 263-278, 2009.

[65] I. Raicu, L. Schwiebert, S. Fowler, and E.K.S. Gupta, “Local load balancing for globally efficient routing in wireless sensor networks,” International Journal of Distributed Sensor Networks, vol. 1, issue. 2, 2005, doi:10.1080/15501320590966431, pp.163–185.

[66] I. Rhee, A. Warrier, J. Min, and L. Xu, “DRAND: Distributed randomized TDMA scheduling for wireless ad hoc networks,” in Proc. ACM MobiHoc, New York, pp. 190–201, 2006.

[67] I. Rhee, A. Warrier, M. Aia, J. Min, and M. Sichitiu, “Z-MAC: A Hybrid MAC for Wireless Sensor Networks,” IEEE/ACM TRANSACTIONS ON NETWORKING, vol. 16, no. 3, pp. 551-524, JUNE 2008.

[68] IEEE Standard 802.15, “Wireless Medium Access Control and Physical Layer Specification for Low Rate Wireless Personal Area Networks,” IEEE Std. 802.15, Part 15.4, 2006.

[69] J. Ai, J. Kong, D. Turgut, “An adaptive coordinated medium access control for wireless sensor networks,” in Proceedings of the International Symposium on Computers and Communications, vol. 1, pp. 214–219, 2004.

[70] J. Al-Karaki, and Ahmed E. Kamal, “Routing Techniques in Wireless Sensor Networks: A Survey,” IEEE Communications Magazine, vol. 11, no. 6, pp. 6-28, December 2004.

[71] J. F. Kurose, K. W. Ross, “Computer Networking: A Top-Down Approach Featuring the Internet,” third edition, Addison Wesley, 2005.

[72] J. Garcia-Luna-Aceves, J. Raju, “Distributed assignment of codes for multihop packet-radio networks,” In MILCOM Proceedings, pp. 450-461, Monterey, CA, USA, 2-5 Nov. 1997.

[73] J. H. Chang and L. Tassiulas, “Energy conserving routing in wireless ad-hoc networks,” in Proc. IEEE INFOCOM, vol. 1, 2000, pp. 22–31.

[74] J. H. Huang, Y. Y. Chen, Y. T. Huang, P. Y. Lin, Y. C. Chen, Y. F. Lin, S. C. Yen, P. Huang, and L. J. Chen, “Rapid Prototyping for Wildlife and Ecological Monitoring,” IEEE Systems Journal, vol. 4, no. 2, pp.198–209, June 2010.

[75] J. Kim, J. On, S. Kim, and J. Lee, “Performance Evaluation of Synchronous and Asynchronous MAC Protocols for Wireless Sensor Networks,” Second International Conference on Sensor Technologies and Applications (SENSORCOMM’08), Cap Esterel, France, Aug. 2531, 2008, pp. 500-506.

[76] J. Kim, X. Lin, and N.B. Shroff, “Optimal Anycast Technique for Delay-Sensitive Energy-Constrained Asynchronous Sensor Networks,” IEEE INFOCOM, 2009, pp. 612-620.

[77] J. Kim, X. Lin, N. B. Shroff, and P. Sinha, “On Maximizing the Lifetime of Delay-Sensitive Wireless Sensor Networks with Anycast,” in Proc. IEEE INFOCOM, Pheonix, AZ, April 2008, pp. 807-815.

[78] J. Luo and J.P. Hubaux, “Joint mobility and routing for lifetime elongation in wireless sensor networks,” In Proceedings of the 24th Annual IEEE International Conference on Computer Communications (INFOCOM2005), vol. 3, 2005. ISBN 0-7803-8968-9. doi: 10.1109/INFCOM.2005.1498454, pp. 1735–1746.

[79] J. Polastre, J. Hill, D. Culler, “Versatile low power media access for wireless sensor networks, in: Proceedings of the International Conference on Embedded Networked Sensor Systems (SenSys), pp. 95–107, 2004.

[80] J. Polastre. A Unifying Link Abstraction for Wireless Sensor Networks. Ph.D. dissertation, University of California, Berkeley, October 2005.

[81] J. S. Hunter, “The Exponentially Weighted Moving Average,” Journal of Quality Technology, vol.18, pp. 203-210, 1986.

[82] K. Akkaya and M. Younis, “An Energy-Aware QoS Routing Protocol for Wireless Sensor Networks,” In Proceedings of the IEEE Workshop on Mobile and Wireless Networks (MWN2003), May 2003.

[83] K. Akkaya, and M. Younis, “A survey on routing protocols for wireless sensor networks,” Elsevier, Ad Hoc Networks, vol. 3, no. 3, pp. 325–349, May 2005.

[84] K. Akkaya, M. Younis, and W. Youssef, “Positioning of base stations in wireless sensor networks,” IEEE Communications Magazine, vol. 45, issue. 4, 2007, pp. 96–102.

[85] K. Dantu, M. Rahimi, H. Shah, S. Babel, A. Dhariwal, and G.S. Sukhatme, “Robomote: enabling mobility in sensor networks,” In Proceedings of 4th International Symposium on Information Processing in Sensor Networks (IPSN 2005), pp. 404-409, April 2005.

[86] K. Hwang, D. Eom, “Adaptive Sink Mobility Management Scheme for Wireless Sensor Networks,” Lecture Notes in Computer Science Volume 4159, Springer Verlag, 2006, pp. 478-487.

[87] K. Hwang, J. In, and D. Eom, “ tree for minimum energy data aggregation of multiple mobile sinks in wireless sensor networks,” Lecture Notes in Computer Science Volume 3868, Springer Verlag, 2006, pp. 132-147.

[88] K. J. Wong and D. Arvind, “Speckmac: Low-power decentralized MAC protocol low data rate transmissions in Specknets,” Proc. 2nd International Workshop on Multi-hop Ad Hoc Networks: from Theory to Reality (REALMAN’06), ACM SIGMOBILE, May 2006, Italy, pp. 71-78.

[89] K. Jamieson, H. Balakrishnan, and Y. C. Tay, “Sift: A MAC Protocol for Event-Driven Wireless Sensor Networks,” Third European Conference on Wireless Sensor Networks (EWSN06), Zurich, Switzerland, February 2006, Vol. 3868, pp. 260-275.

[90] K. Kar, M. Kodialam,T. Lakshman, and L.Tassiulas,“Routing for network capacity maximization in energy-constrained ad-hoc networks,” in Proc. IEEE INFOCOM, vol. 1, 2003, pp. 673–681.

[91] K. Kredo II, P. Mohapatra, “Medium access control in wireless sensor networks,” Computer Networks, vol. 51, no. 4, pp. 961–994, March 2007.

[92] K. Langendoen, “Energy-Efficient Medium Access Control,” Book chapter, Medium access control in wireless networks, pp. 535–560, 2008.

[93] K. Sohrabi, J. Gao, V. Ailawadhi, and G. J. Pottie, “Protocols for Self-Organization of a Wireless Sensor Network,” IEEE Personal Communications, vol. 7, no. 5, pp. 16-27, Oct. 2000.

[94] K. Sohrabi, J. Gao, V. Ailawadhi, and G. Pottie, “Protocols for self organization of a wireless sensor network,” IEEE Personal Communications, vol. 7, pp. 16-27, 2000.

[95] L. Bao, J. Garcia-Luna-Aceves, “A new approach to channel access scheduling for ad hoc networks,” in Proceedings of the International Conference on Mobile Computing and Networking (MobiCom), pp. 210–221, 2001.

[96] L. Bernardo, R. Oliveira, M. Pereira, M. Macedo, and P. Pinto, “A Wireless Sensor MAC Protocol for Bursty Data Traffic,” In 18th Annual IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC), Athens, Greece, Sept. 3-7, 2007, pp. 1-5.

[97] L. Bernardo, R. Oliveira, M. Pereira, M. Macedo, and P. Pinto, “A Wireless Sensor MAC Protocol for Bursty Data Traffic,” In 18th Annual IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC), Athens, Greece, Sept. 3-7, 2007, pp. 1-5.

[97] L. F. W. van Hoesel and P. J. M. Havinga, “A lightweight medium access protocol (LMAC) for wireless sensor networks: reducing preamble transmissions and transceiver state switches,” In: 1st International Workshop on Networked Sensing Systems (INSS’04), Tokyo, Japan, pp. 205-208.

[98] L. Kleinrock, F.A. Tobagi, “Packet switching in radio channels: Part I- Carrier sense multiple-access modes and their throughput-delay characteristics,” IEEE Transactions on Communications, vol. 23, no.12, pp.1400–1416, 1975.

[99] L. Li, J. Halpern, P. Bahl, Y. M. Wang, and R. Wattenhofer, “Analysis of a cone-based distributed topology control algorithm for wireless multi-hop networks,” in Proc. ACM PODC, 2001, pp. 264-273.

[100] L. Zhao, B. Kan, Y. Xu, and X. Li, " FT-SPEED: A Fault-Tolerant, Real-Time Routing Protocol for Wireless Sensor Networks," In proceedings of international conference on wireless communications, Networking and Mobile Computing, WiCom'07, pp. 2531-2534, Sept 2007.

[101] M. Buettner, G. V. Yee, E. Anderson, and R. Han, “X- MAC: A short preamble MAC protocol for duty-cycled wireless sensor networks,” In Proceedings of 2nd ACM Conference on Embedded Networked Sensor Systems (SenSys’06), pp. 307–320, 2006.

[102] M. C. Vuran, I. F. Akyildiz, “Spatial correlation-based collaborative medium access control in wireless sensor networks,” IEEE/ACM Transactions on Networking vol. 14, no. 2, pp. 316– 329, 2006.

[103] M. Chen, T. Kwon, and Y. Choi, “Energy-efficient differentiated directed diffusion (EDDD) in wireless sensor networks,” Computer Communications, vol. 29, no. 2, 2006, pp. 231–245.

[104] M. Chu, H. Haussecker, and F. Zhao, "Scalable Information-Driven Sensor Querying and Routing for ad hoc Heterogeneous Sensor Networks," The International Journal of High Performance Computing Applications, vol. 16, no. 3, August 2002.

[105] M. Di Francesco, S. K. Das, and G. Anastasi, “Data collection in wireless sensor networks with mobile elements: A survey,” ACM Transactions on Sensor Networks, Article 7, Vol. 8, no. 1, 31pages, August 2011. doi:10.1145/1993042.1993049.

[106] M. Doudou, D. Djenouri , N. Badache, A. Bouabdallah, “Synchronous contention-based MAC protocols for delay-sensitive wireless sensor networks: A review and taxonomy,” Elsevier, Journal of Network and Computer Applications (2013). http://dx.doi.org/10.1016/j.jnca.2013.03.012i

[107] M. Doudou, D. Djenouri, and N. Badache, "Survey on Latency Issues of Asynchronous MAC Protocols in Delay-Sensitive Wireless Sensor Networks," IEEE Communications Surveys & Tutorials, Volume: 15, Issue: 2, Second Quarter 2013, pp. 528-550.

[108] M. E. Keskin, I. K. Altinel, N. Aras, and C. Ersoy, "Lifetime Maximization in Wireless Sensor Networks Using a Mobile Sink with Nonzero Traveling Time," The Computer Journal, vol. 54, no. 12, May 2011, doi: 10.1093/comjnl/bxr048, pp. 1987-1999.

[109] M. Guerroumi, N. Badache and S. Moussaoui, "Clustering Pair and Odd to Optimize the Energy Consumption in Wireless Sensor Networks" Communications of SIWN, Vol. 6, pp. 127-131, April 2009.
http://fatech.org.uk/press/tai/cosiwn0006.htm

[110] M. Guerroumi, N. Badache, S. Moussaoui, "MAC Protocol with Low Energy Consumption in Wireless Sensors Network,"Second International Conference on Advances in Future Internet, Venice, Italy, 18-25 July 2010.
http://doi.ieeecomputersociety.org/10.1109/AFIN.2010.14

[111] M. Guerroumi, N. Badache, S. Moussaoui, "Mobile Sink and Power Management for Efficient Data Dissemination in Wireless Sensor Networks," Springer, Telecommunication Systems, accepted, Jun 2013.

[112] M. Guerroumi, N. Badache, S. Moussaoui, "Sink Mobile for Efficient Data Dissemination in Wireless Sensor Networks," Communications in Computer and Information Science, vol. 253, Part.1, pp 635-645, 2012.
http://www.springerlink.com/content/p2v21718l2403735/

[113] M. Guerroumi, N. Badache, S. Moussaoui," Clustering data dissemination in wireless sensors network," International Conference on Communications, Computing and Control Applications (CCCA), pp. 1-6, 2011.
http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=6031226.

[114] M. Guerroumi, N. Badache, S. Moussaoui," Data Dissemination and Power Management in Wireless Sensor Networks," Advances in Computing and Communications, Communications in Computer and Information Science, vol.193, part. 6, pp. 593-607, 2011. http://www.springerlink.com/content/w4k202l372877730/.

[115] M. Guerroumi, A.K. Pathan, N. Badache, S. Moussaoui," Strengths and Weaknesses of Prominent Data Dissemination Techniques in Wireless Sensor Networks," IJCNIS, vol. 5, No. 3, Dec 2013, pp. 158-175.

[116] M. H. Anisi, A. H, Abdullah, and S. A. Razak, "Energy-efficient and reliable data delivery in wireless sensor networks," Wireless Networks, vol. 19, 2013, pp. 495–505.

[117] M. H. S. Gilani, I. Sarrafi, M. Abbaspour, "An adaptive CSMA/TDMA hybrid MAC for energy and throughput improvement of wireless sensor networks," Ad Hoc Networks, vol. 11, issue. 4, pp. 1297-1304, 2011.

[118] M. Hamalainen, P. Pirinen, and Z. Shelby, "Advanced wireless ict healthcare research," In Proceedings of 16th IST Mobile and Wireless Communications Summit, pp.1-5, July 2007.

[119] M. Joa-Ng, I. Lu, “Spread spectrum medium access protocol with collision avoidance in mobile ad-hoc wireless network”, In IEEE INFOCOM '99, pp. 776-783, New York, NY, USA, 21-25 March 1999.

[120] M. Khan, W. Gansterer and G. Haring, “Congestion Avoidance and Energy Efficient Routing Protocol for Wireless Sensor Networks with a Mobile Sink,” Journal of networks, vol. 2, no. 6, pp. 42-49, Dec 2007.

[121] M. Khan, W. Gansterer and G. Haring, “In-network storage model for data persistence under congestion in wireless sensor network,” In Proceedings of the First International Conference on Complex, Intelligent and Software Intensive Systems (CISIS'07), pp. 221-228, 2007.

[122] M. Lee and S. Lee, “Data Dissemination for Wireless Sensor Networks,” In Proceedings of the 10th IEEE International Symposium on Object and Component-Oriented Real-Time Distributed Computing (ISORC'07), 2007.

[123] M. Liu, J. Cao, “A Distributed Energy-Efficient data Gathering and aggregation Protocol for Wireless sensor networks,” Journal Software, vol.16, pp. 2106-2116, 2005.

[124] M. Liu, J. Cao, G. Chen, “An Energy-Aware Routing Protocol in Wireless Sensor Networks,” international access journal on the science and technology of sensors and biosensors, pp 445-4462, Sep 2009.

[125] M. Ma and Y. Yang, “SenCar: An Energy-Efficient Data Gathering Mechanism for Large-Scale Multihop Sensor Networks” IEEE Transactions on Parallel and Distributed Systems, vol. 18, no. 10, pp. 1476-1488, Oct 2007.

[126] M. Miller and N. Vaidya, “A MAC Protocol to Reduce Sensor Network Energy Consumption Using a Wakeup Radio,” IEEE Transactions on Mobile Computing,, Vol. 4, No. 3, May-June 2005, pp. 228-242.

[127] M. Radenkovic and B. Wietrzyk, “Wireless Mobile Ad-hoc Sensor Networks for Very Large Scale Cattle Monitoring,” In Proceedings of in 6th International Workshop on Applications and Services in Wireless Networks*, (ASWN’06)*, Germany, 2006.

[128] M. Vecchio, A. Viana, A. Ziviani, R. Friedman, “DEEP: Density-based proactive data dissemination protocol for wireless sensor networks with uncontrolled sink mobility,” Computer Communications, vol. 33, no. 8, pp. 929-939, May 2010.

[129] M. Younis, M. Youssef, and K. Arisha, "Energy-aware routing in cluster-based sensor networks," In Proceedings of the 10th IEEE/ACM International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS), 2002, pp. 129-136.

[130] M. Zorzi, and R. Rao, “Geographic random forwarding (GeRaF) for ad hoc and sensor networks: Energy and latency performance,” IEEE Trans, Mobile Comput, vol. 2, no.4, pp 349-365, 2003.

[131] M. Zorzi, and R. Rao, “Geographic random forwarding (GeRaF) for ad hoc and sensor networks: Multihop performance,” IEEE Trans, Mobile Comput, Vol.2, no.4, pp.337-348, 2003.

[132] Modeling the P-Median problem with OPL. IBM ILOG OPL documentation for ODM Enterprise, IBM, from: http://pic.dhe.ibm.com/infocenter/odmeinfo/v3r3/index.jsp?topic=/ilog.odms.ide.odme.help/Content/Optimization/Documentation/ODME/_pubskel/ODME_pubskels/startALL_ODME_Eclipse_and_Xplatform_doc258.html [last accessed: 7 July, 2013]

[133] N. Bulusu, J. Heidemann, and D. Estrin, "Gps-less low cost outdoor localization for very small devices," IEEE Personal Communications Magazine, vol.7, no.5, pp.28–34, October 2000.

[134] N. Bulusu, J. Heidemann, D. Estrin and T. Tran "Self-Configuration Localization Systems: Design and experimental Evaluation," ACM Transactions, vol. 3, no. 1, pp. 24-60, February 2004.

[135] N. Dimokas, D. Katsaros, and Y. Manolopoulos, "Energy efficient distributed clustering in wireless sensor networks," Journal of Parallel and Distributed Computing, vol. 70, 2010, pp. 371–383.

[136] O. Younis and S. Fahmy, "HEED: A Hybrid, Energy-Efficient, Distributed Clustering Approach for Ad Hoc Sensor Networks," IEEE Transactions on Mobile Computing, vol. 3, 2004, pp. 366-379.

[137] P. B. Mirchandani and R. L. Francis, "Discrete Location Theory," Ser. Wiley-Interscience Series in Discrete Mathematics and Optimization. Wiley & Sons, NewYork, Total Pages 576, July 1990.

[138] P. Casari, A. Marcucci, M. Nati, C. Petrioli, and M. Zorzi, "A detailed simulation study of geographic random forwarding (GeRaF) in wireless sensor networks," In Proceedings of the Military Communications Conference (MILCOM), pp. 17–20, 2005.

[139] P. Chatterjee and N. Das, "A distributed algorithm for load-balanced routing in multihop wireless sensor networks," In Proceedings of the 9th international conference on Distributed computing and networking, ICDCN'08, Springer-Verlag, 2008. ISBN 3-540-77443-2, 978-3-540-77443-3, pp. 332–338.

[140] P. Dutta, S. Dawson-Haggerty, Y. Chen, C. Liang, and A. Terzis, "Design and Evaluation of a Versatile and Efficient Receiver-Initiated Link Layer for Low-Power Wireless," In Proc. 8th ACM Conference on Embedded Networked Sensor Systems (Sensys'10), Zurich, Switzerland, Nov. 3-5, 2010, pp. 1-14.

[141] P. G. Park, C. Fischione, A. Bonivento, K. H. Johansson, and A. Sangiovanni-Vincentelli, "Breath: a self-Adapting protocol for wireless sensor networks in control and automation," IEEE SECON, San Francisco, CA, USA, 2008, pp. 323-331.

[142] P. Huang, L. Xiao, S. Soltani, M.W. Mutka, and N. Xi, "The Evolution of MAC Protocols in Wireless Sensor Networks: A Survey," IEEE Communications Surveys & Tutorials, Vol. 15, Issue: 1, pp. 101-120, First Quarter 2013.

[143] P. Hurni, and T. Braun, "MaxMAC: A Maximally Traffic-Adaptive MAC Protocol for Wireless Sensor Networks," European Conference on Wireless Sensor Networks (EWSN), Coimbra, Portugal, February 17-19, 2010, pp 289-305.

[144] P. Hurni, T. Braun, and M. Anwander, "Evaluation of WiseMAC and Extensions on Wireless Sensor Nodes," Springer Telecommunication Systems Journal, vol.43, no.1-2, pp. 49-58, February 2010.

[145] P. Hurni, T. Braun, and M. Anwander, “Evaluation of WiseMAC and Extensions on Wireless Sensor Nodes,” Telecommunication Systems Journal, Springer, February 2010, Vol.43, No.1-2, pp. 49-58.

[146] P. Kumar, M. Gunes, Q.Mushtaq, and B. 2Blywis, “A real-time and energy-efficient MAC protocol for wireless sensor networks,” International Conference on Wireless and Optical Communications Networks (WOCN’09), Cairo, Egypt, April 28-30, 2009, pp. 1-5.

[147] P. Levis, N. Lee, M. Welsh, and D. Culler, “TOSSIM: Accurate and Scalable Simulation of Entire TinyOS Applications,” in Proceedings of the First ACM Conference on Embedded Networked Sensor Systems (SenSys), 2003.

[148] P. Lin, C. Qiao, and X. Wang, “Medium access control with a dynamic duty cycle for sensor networks,” In Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC), vol. 3, 2004, pp. 1534-1539.

[149] P. Suriyachai, U. Roedig, and A. Scott, “A Survey of MAC Protocols for Mission-Critical Applications in Wireless Sensor Networks,” IEEE Communications Surveys & Tutorials, Volume: 14, Issue: 2, pp. 240-264, Second Quarter 2012.

[150] Q. Dong, and W. Dargie, “A Survey on Mobility and Mobility-Aware MAC Protocols in Wireless Sensor Networks,” IEEE Communications Surveys & Tutorials, Volume: 15, Issue: 1, pp. 88-100, First Quarter 2013.

[151] Q. Fang, J. Gao, L. Guibas, “Locating and bypassing routing holes in sensor networks,” In 23rd Conference of the IEEE Communications Society (InfoCom), 2004, Mobile Network Application 2006.

[152] Q. Yu, C. Tan, and H. Zhou, “A Low-Latency MAC Protocol for Wireless Sensor Networks,” International Conference on Wireless Communications, Networking and Mobile Computing (WiCom07), Shanghai, Sept. 21-25, 2007, pp. 2816-2820.

[153] Q. Zhang, Y. Liua, H. Guoa, Q. Zhang, “The Design of Hybrid MAC Protocol for Industry Monitoring System Based on WSN,” Elsivier, Procedia Engineering, vol. 23, pp. 290 – 295, 2011.

[154] R. Bagrodia, X. Zeng, and M. Gerla, “GloMoSim - A Library for Parallel Simulation of Large-scale Wireless Networks,” Workshop on Parallel and Distributed Simulation, California at Los Angles, pp. 154-161, 1999.

[155] R. Braynard, A. Silberstein, and C. Ellis, “Extending Network Lifetime Using an Automatically Tuned Energy-Aware MAC Protocol,” Proc. European Workshop on Wireless Sensor Networks (EWSN), Zurich, Switzerland, 2006, pp. 244-259.

[156] R. Hongliang. and M. Q. H. Meng, “Understanding the mobility model of wireless body sensor networks,” In Proceedings of the IEEE International Conference on Information Acquisition, pp. 306-310, Aug 2006.

[157] R. Rajagopalan and P.K. Varshney, “Data-aggregation techniques in sensor networks: a survey,” IEEE Communications Surveys & Tutorials, Volume: 8, Issue: 4, Fourth Quarter 2006, pp. 48-63.

[158] R. Yadav, S. Varma, N. Malaviya, “A SURVEY OF MAC PROTOCOLS FOR WIRELESS SENSOR NETWORKS,” UbiCC Journal, vol.4, no.3, pp.827-833, August 2009.

[159] S. Basagni, A. Carosi, E. Melachrinoudis, C. Petrioli, and Z. Wang, “Controlled sink mobility for prolonging wireless sensor networks lifetime,” Wireless Networks, vol.14, no.6, pp.831-858, February 2007.

[160] S. Chatterjea, L. van Hoesel, and P. Havinga, “AI-LMAC: an adaptive, information-centric and lightweight MAC protocol for wireless sensor networks,” In Proceedings of the Intelligent Sensors, Sensor Networks, and Information Processing Conference, 2004, pp. 381-388.

[161] S. De Vito, E. Massera, G. Burrasca, A. Di Girolamo, M. Miglietta, G. Di Francia, and D. Della Sala, “Tinynose: Developing a wireless e-nose platform for distributed air quality monitoring applications,” In Proceedings of IEEE Sensors Conference, pp.701-704, Oct 2008.

[162] S. Du, A. K. Saha, and D. B. Johnson, “RMAC: A routing-enhanced duty-cycle MAC protocol for wireless sensor networks,” In Proceedings of the 26th Annual IEEE Conference on Computer Communications, 2007, pp. 1478-1486.

[163] S. Ghosh, P. Veeraraghavan, S. Singh, and L. Zhang, “Performance of a Wireless Sensor Network MAC Protocol with a Global Sleep Schedule,” International Journal of Multimedia and Ubiquitous Engineering, vol. 4, no. 2, April, 2009.

[164] S. H. Lee, S. Lee, H. Song, and H. S. Lee, “Wireless sensor network design for tactical military applications: Remote large-scale environments,” In Proceedings of IEEE Military Communications Conference, MILCOM’09, pp. 1-7, Oct 2009.

[165] S. Hashish and A. Karmouch, “An adaptive rendezvous data dissemination for irregular sensor networks with multiple sinks,” Computer Communications , vol. 33, issue 7, pp. 176-189, 2010.

[166] S. K. Yadav, “A Novel Survey on Various Grid Based Data Dissemination,” International Journal of Advanced Research in Computer Science and Software Engineering, vol. 2, issue 7, pp. 153-158, July 2012.

[167] S. Kim, S. Pakzad, D. Culler, J. Demmel, G. Fenves, S. Glaser, and M. Turon, “Wireless sensor networks for structural health monitoring,” In Proceedings of the 4th international conference on Embedded networked sensor systems, SenSys ’06, pp. 427-428, USA, 2006.

[168] S. Kim, S. Son, J. Stankovic, S. Li, Y. Choi, “SAFE: A Data Dissemination Protocol for Periodic Updates in Sensor Network,” ICDCS Workshops, pp. 228-234, 2003.

[169] S. Kim, S. Son, J. Stankovic, Y. Choi, “Data dissemination over wireless sensor networks,” IEEE Communications Letters, vol. 8, no. 9, pp. 561-563, Sep. 2004.

[170] S. Lindsey, C. Raghavendra, “PEGASIS: Power-Efficient Gathering in Sensor Information Systems," IEEE Aerospace Conference Proceedings, vol. 3, pp. 1125-1130, 2002.

[171] S. Liu, K. Fan, and P. Sinhf, “CMAC: An Energy-Efficient MAC Layer Protocol Using Convergent Packet Forwarding for Wireless Sensor Networks,” ACM Transactions on Sensor Networks, vol. 5, no. 4, Article 29, November 2009.

[172] S. Mahlknecht and M. Boeck, “CSMA-MPS: A Minimum Preamble Sampling MAC Protocol for Low Power Wireless Sensor Networks,” in IEEE Workshop on Factory Communication Systems, Vienna, Austria, Sept. 22-24, 2004, pp. 73-80.

[173] S. Rashwand, J. V. Misic, V. B. Misic, S. Biswas, and M. M. Haque, “A Novel Asynchronous, Energy Efficient, Low Transmission Delay MAC Protocol for Wireless Sensor Networks,” ICDCS Workshops Proceedings, 2009, pp. 186-193.

[174] S. Sanati, M. H. Yaghmaee, and A. Beheshti, “Energy Aware Multi-path and Multi-SPEED Routing Protocol in Wireless Sensor Networks,” In Proceedings of the 14th International CSI Computer Conference (CSICC'09), pp. 640-645, 2009.

[175] S. Sharma, D. Kumar, and R. Kumar, “QOS-Based Routing Protocol in WSN,” Advances in Wireless and Mobile Communications. Vol.1, pp. 51–57, 2008.

[176] S. Singh, C. Raghavendra, “PAMAS – power aware multi-access protocol with signaling for ad hoc networks,” SIGCOMM Computer Communications Review, vol. 28, issue. 3, 1998, pp. 5-26.

[177] S. Singh, M. Singh, and D. Singh, “Routing Protocols in Wireless Sensor Networks –A Survey,” International Journal of Computer Science & Engineering Survey (IJCSES), vol.1, no.2, November 2010.

[178] S. Tilak, N. B. Abu-Ghazaleh, and W. Heinzelman, “A taxonomy of wireless micro-sensor network models,” ACMSIGMOBILE Mobile Computing and Communications Review, vol. 6, no.2, pp. 28-36, April 2002.

[179] S. V. Crowder, “A Simple Method for Studying Run-length Distributions of Exponentially Weighted Moving Average Charts,” Technometrics, vol.29, pp. 401-408, 1987.

[180] S. V. Crowder, “Average Run Lengths of Exponentially Weighted Moving Average Charts”, Journal of Quality Technology, vol.19, pp. 16 -164, 1987.

[181] S. Yessad, F. Nait-Abdesselam, T. Taleb, and B. Bensaou, “R-MAC: Reservation medium access control protocol for wireless sensor networks,” In Proceedings of the 32nd IEEE Conference on Local Computer Networks, IEEE, Washington, D.C., 2007, pp. 719-724.

[182] T. He, J. A. Stankovic, C. Lu, and T. F. Abdelzaher, “SPEED: A Stateless Protocol for Real-Time Communication in Ad Hoc Sensor Networks,” In International Conference on Distributed Computing Systems (ICDCS '03), USA, pp. 46- 55, May 2003.

[183] T. van Dam, K. Langendoen, “An adaptive energy-efficient MAC protocol for wireless sensor networks,” in Proceedings of the International Conference on Embedded Networked Sensor Systems (SenSys), pp. 171–180, 2003.

[184] T. Wu, and S. Biswas, “A self-reorganizing slot allocation protocol for multi-cluster sensor networks,” In Proceedings of the 4th International Symposium on Information Processing in Sensor Networks, IEEE, pp. 309-316, April 2005.

[185] T. Yang, L. Barolli, G. Mino, F. Xhafa, and A. Durresi, “ Impact of Mobile Event Movement on the Performance of Wireless Sensor Networks,” In Proceedings of the 5th International Conference on Systems and Networks Communications, pp. 88–93. IEEE, Aug 2010.

[186] T. Zheng, S. Radhakrishnan, V. Sarangan, “PMAC: An adaptive energy-efficient MAC protocol for wireless sensor networks,” in Proceedings of the IEEE International Parallel and Distributed Processing Symposium, pp. 65–72, 2005.

[187] US Naval Observatory (USNO) GPS Operations, http://tycho.usno.navy.mil/gps.html

[188] V. Drabkin, R. Friedman, G. Kliot, M. Segal, “RAPID: reliable probabilistic dissemination in wireless ad hoc networks,” In Proceedings of the 26th IEEE International Symposium on Reliable Distributed Systems (SRDS), pp. 13-22, 2007.

[189] V. Rajendran, J. J. Garcia-Luna-Aceves, and K. Obraczka, “Energy-efficient, application-aware medium access for sensor networks,” In Proceedings of the 2005 International Conference on Mobile Ad Hoc and Sensor Systems Conference (MASS 2005), 2005, pp. 623-630.

[190] V. Rajendran, K. Obraczka, and J.J. Garcia-Luna-Aceves, “Energy-efficient, collision-free medium access control for wireless sensor networks,” Wireless Networks, Vol. 12, issue. 1, Feb 2006, pp. 63–78.

[191] W. B. Heinzelman, A. P. Chandrakasan and H. Balakrishnan, “An application-specific protocol architecture for wireless micro sensor networks,” IEEE Transactions on Wireless Communications, vol. 1, no. 4, pp. 660 – 670, October 2002.

[192] W. Ben Jaballah and N. Tabbane, “Multi path Multi SPEED Contention Window Adapter,” IJCSNS International Journal of Computer Science and Network Security, vol.9, no.2, February 2009.

[193] W. H. Liao and H. H. Wang, “An asynchronous MAC protocol for wireless sensor networks,” Journal of Network and Computer Applications, 2008, Vol. 31, No. 4, pp. 807-820.

[194] W. Heinzelman, A. Chandrakasan and H. Balakrishnan, "Energy-Efficient Communication Protocol for Wireless Mi-crosensor Networks," In Proceedings of the 33rd Hawaii International Conference on System Sciences (HICSS '00), January 2000. doi:10.1109/HICSS.2000.926982.

[195] W. R. Heinzelman, J. Kulik and H. Balakrishnan, “Adaptive protocols for information dissemination in wireless sensor networks,” Proceedings of the ACM MobiCom’99, Seattle, Washington, pp.174-185, 1999.

[196] W. Wang, V. Srinivasan, and K.C. Chua, “ Extending the lifetime of wireless sensor networks through mobile relays,” IEEE/ACM Transactions on Networking, vol. 16, no. 5, pp. 1108-1120, October 2008.

[197] W. Y. Poe and J. B. Schmitt, “Self-organized sink placement in large-scale wireless sensor networks,” In Proceedings of the 17th IEEE International Symposium on Modeling, Analysis Simulation of Computer and Telecommunication Systems, MASCOTS’09, September 2009. doi: 10.1109/MASCOT.2009.5366741, pp. 1–3.

[198] W. Ye, J. Heidemann, D. Estrin, “An energy-efficient MAC protocol for wireless sensor networks,” in: Proceedings of the Joint Conference of the IEEE Computer and Communications Societies (InfoCom), vol. 3, pp. 1567-1576, 2002.

[199] W. Ye, J. Heidemann, D. Estrin, "Medium access control with coordinated adaptive sleeping for wireless sensor networks," IEEE/ACM Transactions on Networking vol.12, no. 3, pp 493–506, 2004.

[200] X. Shi and G. Stromberg, "SyncWUF: An Ultra Low-Power MAC Protocol for Wireless Sensor Networks," IEEE Transactions on Mobile Computing, Vol. 6, Issue 1, Jan. 2007, pp. 115-125.

[201] X. Y. Li, W. Z. Song and W. Wang, "A unified energy-efficient topology for unicast and broadcast," In Proceedings of the 11th annual ACM MobiCom, 2005, pp. 1–15.

[202] Y. C. Chang, J. R. Jiang, J. P. Sheu, and H. Y. Shih, "ADCA: An Asynchronous Duty Cycle Adjustment MAC Protocol for Wireless Sensor Networks," Global Telecommunications Conference IEEE GLOBECOM 08, New Orleans, LO, Nov.30-Dec. 4, 2008, pp. 383-3870.

[203] Y. E. Sagduyu and A. Ephremides, "The problem of medium access control in wireless sensor networks," IEEE Wireless Communications, vol. 11, issue. 6, 2004, pp. 44-53.

[204] Y. Sun, O. Gurewitz, and D.B. Johnson, "RI-MAC: a receiver-initiated asynchronous duty cycle mac protocol for dynamic traffic loads in wireless sensor networks," Proc. 6th ACM conference on Embedded network sensor systems (SenSys'08), 2008, pp. 1-14.

[205] Y. Sun, S. Du, O. Gurewitz, and D. B. Johnson, "DW-MAC: A low latency, energy efficient demand wakeup MAC protocol for wireless sensor networks," In Proceedings of the 9th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc '08), ACM, New York, 2008, pp. 53-62.

[206] Y. Wang, C. Tsai and H. Mao, "HMRP: Hierarchy-Based Multipath Routing Protocol for Wireless Sensor Networks," Tamkang Journal of Science and Engineering, vol. 9, no 3, pp. 255-264, 2006.

[207] Y. Xu, J. Heidemann, and D. Estrin, "Geography-informed Energy Conservation for Ad Hoc Routing," In Proceedings of the ACM/IEEE International Conference on Mobile Computing and Networking, pp. 70-84, 2001.

[208] Y. Xu, J. Heidemannn, and D. Estrin, "Geography informed energy conservation for ad hoc routing," In Proceedings of the Seventh Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom 2001), Rome, Italy, July 2001.

[209] Y. Yao and J. Gehrke, "The cougar approach to in-network query processing in sensor networks," In SIGMOD Record, September 2002.

[210] Y. Z. Zhao, M. Ma, C. Y. Miao, and T. N. Nguyen, "An energy-efficient and low-latency MAC protocol with adaptive scheduling for multi-hop wireless sensor networks," Computer Communications, Vol. 33, pp. 1452-1461, 2010.

[211] Y. Z. Zhao, M. Ma, C.Y. Miao, and T. N. Nguyen, "An energy-efficient and low-latency MAC protocol with adaptive scheduling for multi-hop wireless sensor networks," Computer Communications, Vol. 33, pp. 1452-1461, 2010.

[212] Y. Z. Zhao, C. Miao, M. Ma, J.B. Zhang, and C. Leung, "A survey and projection on medium access control protocols for wireless sensor networks," ACM Computing Surveys, vol.45, no.1, pp. 1-37, November 2012.

[213] Y. Zhang and Q. Huang, “A Learning-based Adaptive Routing Tree for Wireless Sensor Networks,” Journal of Communications, vol. 1, issue. 2, 2006, pp.12–21.

[214] Z. Li, B. Zhao, Y. Qu, K. Chen , “An adaptive and distributed STDMA Scheme for Ad Hoc and Sensor Networks, ” in Networking, International Conference on Systems and International Conference on Mobile Communications and Learning Technologies, 2006.