Wireless Multimedia Sensor Networks, Security and Key Management

UNIVERSITAT POLITÈCNICA
DE CATALUNYA
BARCELONATECH

Author:
Islam Almalkawi

Advisor:
Prof. Manel Guerrero-Zapata

Submitted to the Departament d'Computer Architecture
in Partial Fulfillment of the Requirements for

Doctor of Philosophy (PhD)

Barcelona - July 2013

بسم الله الرحمن الرحيم

To my great father Dr. Thabet Almalkawi, my sweet mother Husna Mohammad, my lovely wife Maram Malkawi and my charming little daughters Sona Almalkawi and Sereena Almalkawi

Acknowledgements

I am very grateful to the many people who have supported me and encouraged me during my PhD study until this dissertation could be done and ready to be defended.

* First of all, I would like to express my gratitude to my research advisor, Prof. Manel Guerrero-Zapata, for his support, for always keeping his door open to me whenever I needed to see him - willing to help in any way possible for both research and bureaucratic issues, and for guiding me back on track whenever I felt lost. I feel very lucky to have him as an advisor and a friend. I am also very grateful for Manel's time and patience in giving detailed feedback on the many revisions of this dissertation, as well as for all of my research papers and presentations.

* I would like to thank also Prof. Jamal Al-Karaki, Prof. Jose M. Barcelo, and Dr. Julian David Morillo-Pozo for their technical support and feedback during writing my papers.

I am also thankful to all members of the Computer Architecture Department in UPC, especially Prof. Xavier Martorell, and the people in the Administration Office and System Support Office. Very especial thank also to Anna Fabregas from the International Relation office for her great support in residency issues for me and my family in Spain.

* I would also like to thank the members of my thesis committee, Prof. Llorenc Cerda, Prof. Felix Freitag, and Prof. Luis Velasco for their time and insightful suggestions that help greatly improve the quality of this dissertation.

* I am extremely grateful to my parents; they always encouraged me in my studies and I believe that all of my successes in the life are just because of their prayers and supplications for me. My wife Maram also deserves a lot of thanks. In-fact, she exactly demonstrated the saying: Behind Every Successful Man, There Is a Woman. I would also like to express my gratitude to my brothers (Muhammad, Hasan, and Akram) and my sisters (Amal, Areej, Tasneem, and Shahd) for their full support to me in dealing various social and administrative issues in Jordan. I also wish to thank my friends and colleagues in UPC and outside UPC, especially Muhammad Shafiq, Oscar Trullos, Ahmed Helmi, Abhishek Deb, and Amir Darehshoorzadeh for sharing their time and having nice technical and non-technical discussions on the lunch and coffee breaks. There are many, many, many other nice people whom I have left out but were a big part of enriching my life and keeping me sane and in good shape at UPC.

* The acknowledgements will remain incomplete if I do not quote the organizations and the projects that financially supported this thesis. These include the financial support from AGAUR-Generalitat de Catalunya through the grant FI-DGR, the Spanish government projects EuroNF NoE, TIN2010-21378-C02-01 and 2009-SGR-1167, and the Hashemite University in Jordan.

Abstract

Wireless Multimedia Sensor Networks (**WMSNs**) have emerged and shifted the focus from the typical scalar Wireless Sensor Networks (WSNs) to networks with multimedia devices that are capable to retrieve video, audio, images, as well as scalar sensor data. WMSNs are able to deliver multimedia content due to the availability of inexpensive CMOS cameras and microphones coupled with the significant progress in distributed signal processing and multimedia source coding techniques.

In addition to the design restrictions of WSNs - such as resource constraints, scalability, network topology, fault tolerance, production costs, etc- WMSNs have also additional characteristics and challenges because of the nature of the real time multimedia data such as high bandwidth demand, real-time delivery, tolerable end-to-end delay, and proper jitter and frame loss rate. Moreover, there are many different resource constraints in WMSNs involving energy, bandwidth, data rate, memory, buffer size and processing capability because of the physically small size of the sensors and the nature of the multimedia application that is typically producing a huge amount of data. Therefore, to meet the quality of service (QoS) requirements and to use the network scarce resources in a fair and efficient manner, these characteristics of WMSNs should be considered probably at the different layers of the communication protocol stack especially the routing and MAC layers. Moreover, many applications of WMSNs have their additional and special requirements in terms of security and privacy, such as military applications, medical care applications, and other video surveillance systems. In addition to the fact that sensor networks are vulnerable to attacks more easily than the wired networks because of their nature as a broadcast medium.

This dissertation "**Wireless Multimedia Sensor Networks, Security and Key Management**" studies the implementation and development of a secure system of WMSN that is capable of delivering real time multimedia data at an acceptable level of QoS requirements with efficiently using the network scarce resources. For developing this system, *firstly* we outline the design challenges of WMSNs, and we give a comprehensive discussion of the proposed architectures, algorithms and protocols for the different layers of the communication protocol stack for WMSNs along with an evaluation of the existing WMSN hardware and testbeds. Also, we discuss the techniques aimed to improve the energy efficiency of multimedia transmission over wireless sensors networks, such as in-node multimedia signal processing and in-network multimedia communications.

Secondly, we propose a Clustered Multipath Routing protocol for WMSNs, **CMRP**, to satisfy the requirements of delivering different data types and support high data rate multimedia traffic. **CMRP** exploits the hierarchical structure of powerful cluster heads and the optimized multiple paths to support timeliness and reliable high data rate multimedia communication with minimum energy dissipation.

Thirdly, for further improve in the performance of **CMRP** and to reduce collisions at high data rate transmission, we propose a cross-layer based routing protocol that can utilize MAC-layer QoS-based scheduling for more efficient routing mechanism in WMSNs. Our proposed optimization is based on clustered multipath routing protocol and adaptive QoS-aware scheduling for the different traffic classes in WMSNs.

Fourthly, we propose a light-weight distributed key management scheme and intrusion detection system (IDS) that can be used in securing proposed routing protocols and data exchanged in clustered WMSNs, and we apply these security schemes on our proposed routing protocol. The Secure Clustered based Multipath Routing protocol (**SCMR**) -combined with the proposed key management scheme and intrusion detection system- is designed to guarantee message authenticity, integrity, and confidentiality against most types of external and internal attacks, while in the same time they keep a minimal impact on overall network performance in terms of energy efficiency, processing and memory requirement, and communication overhead.

Finally, in order to protect the contextual information in sensor networks, we conduct a complete survey of the state of the art in context privacy preservation in WSNs and the mechanisms used in this field, and we introduce our complete classification for content security and contextual privacy in WSNs. Then we propose our contribution in source/sink location unobservability for WMSNs that aims to hide the location information of the important nodes such as source nodes and base stations.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Wireless Multimedia Sensor Networks

The field of Wireless Sensor Networks (WSNs) is receiving much attention in the networking research community and as an interdisciplinary field of interest. WSNs are becoming more low-cost, low-power, multi-functional, and viable due to the advances in micro-electro-mechanical systems (MEMS), low power and highly integrated digital electronics, and proliferation of wireless communications [1]. Wireless sensor networks (WSNs) typically consist of a large number of intelligent battery-powered sensor nodes with sensing, processing and wireless communicating capabilities [2]. The sensing circuitry measures simple ambient conditions, related to the environment surrounding the sensor such as temperature, humidity or light, and transforms them into an electric signal. Processing such a signal reveals some properties about objects located and/or events happening in the vicinity of the sensor. The sensor sends such collected data (called as scalar data), usually via radio transmitter, to a command center (sink) either directly or through multiple wireless hops [1] [3] [4]. WSNs have wide and varied applications such as real time tracking of objects, monitoring of environmental conditions, monitoring of health structures, and preparing a ubiquitous computing environment, etc [1].

The above mentioned characteristics impose a lot of restrictions on the WSNs design such as fault tolerance, scalability, production costs, network topology, operating environment, hardware constraints, power consumption, etc. These challenges have led to an intensive research in the past few years that addresses the potential collaboration among sensors in data gathering and processing. In most applications, the deployment area has no existing infrastructure for either energy or communication. Therefore, a basic requirement for sensor nodes is to be able to survive with a limited source of energy which is usually a small battery [5]. The network should stay alive and active for a duration of time that depends on the application of the deployed network, and that may last from several weeks to a few years.

Nevertheless, the rapid development and progress of sensors, MEMS, embedded computing, in addition to the availability of inexpensive CMOS (Complementary Metal Oxide Semiconductor) cameras and microphones coupled with the significant progress in distributed signal processing and multimedia source coding techniques, allowed for the emergence of so called wireless multimedia sensor networks. As a result, Wireless Multimedia Sensor Network (WMSN) [6] is a network of wirelessly interconnected sensor nodes equipped with multimedia devices, such as cameras and microphones, and capable to retrieve video and audio streams, still images, as well as scalar sensor data.

WMSNs promise a wide range of potential applications in both civilian and military areas which require visual and audio information such as surveillance sensor networks, law-enforcement reports, traffic control systems, advanced health care delivery, automated assistance to elderly telemedicine, and industrial process control. In these applications multimedia support has the potential of enhancing the level of information collected, enlarging the range of coverage, and enabling multi-resolution views [7] (i.e., in comparison to the measurements of scalar data).

WMSNs have also additional characteristics and challenges, in addition to those of Wireless Sensor Networks (WSNs), because of the nature of the real time multimedia data such as high bandwidth demand, real-time delivery, tolerable end-to-end delay, and proper jitter and frame loss rate. Moreover, there are many different resource constraints in WMSNs involving energy, bandwidth, data rate, memory, buffer size and processing capability because of the physically small size of the sensors and the nature of the multimedia application that is typically producing a huge amount of

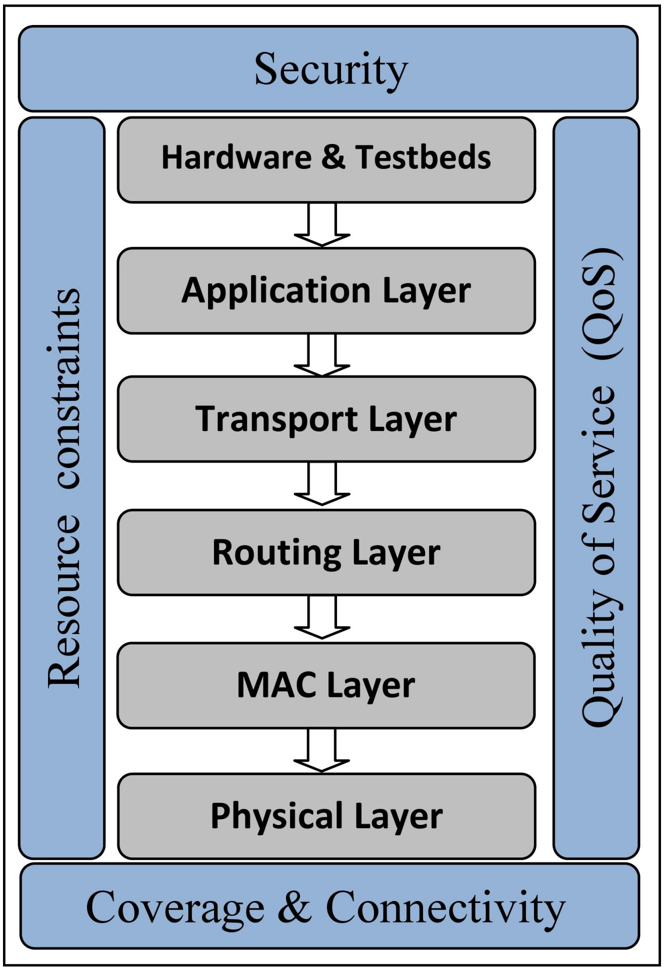


Figure 1.1: Research Challenges in WMSNs

data. Therefore, to meet the quality of service (QoS) requirements and to use the network scarce resources in a fair and efficient manner, these characteristics of WMSNs along with other research issues such as coverage and security - as shown in Figure 1.1 - become a concern, and should be considered probably at the different layers of the communication protocol stack. We outline and discuss these issues in detail in the following sections. Moreover, given the relatively high redundancy in the visual sensor data, WMSNs have additional requirements such as in-node multimedia processing techniques (e.g. multimedia distributed source coding and data compression), application-specific QoS requirements, and multimedia in-network processing techniques (e.g. storage management, data fusion and aggregation).

1.2 Network Architecture

Traditionally, most of the proposed network architectures in scalar wireless sensor networks are based on a flat architecture of distributed homogeneous nodes, where low-power scalar sensors are in charge of performing simple tasks such as detecting scalar physical measurements. But with the emerging of WMSN and its new applications, new types of sensor nodes besides scalar sensors (such as multimedia sensors, processing hubs, storage hubs) with different capabilities and functionalities have been used. This raises the need to reconfigure the network into different architectures in a way the network can be more scalable and more efficient depending on its specific application and QoS requirements. Therefore, based on the designed network topology, the available resources in the network can be efficiently utilized and fairly distributed throughout the network, and the desired operations of the multimedia content can be handled. In general, Network architectures in WMSNs can be divided into three reference models as shown in Figure 1.2.

The first model is the **single-tier flat architecture** where the network is deployed with homogeneous sensor nodes of the same capabilities and functionalities. In this model all the nodes can perform any function from image capturing through multimedia processing to data relaying toward the sink in multi-hop basis. Single-tier flat architecture is easy to manage. Moreover, multimedia processing is distributed among the nodes, which prolongs network life time. The second model is the **single-tier clustered architecture** deployed with heterogeneous sensors where camera, audio and scalar sensors within each cluster relay data to a cluster head. The cluster head has more resources and it is able to perform intensive data processing. The cluster head is wirelessly connected with the sink or the gateway either directly or through other cluster heads in multi-hop fashion. The third model is the **multi-tier architecture** with heterogeneous sensors. In this architecture, the first tier deployed with scalar sensors performs simple tasks, like motion detection, the second tier of camera sensors may perform more complicated tasks as object detection or object recognition, and at the third tier more powerful and high resolution camera sensors are capable to perform

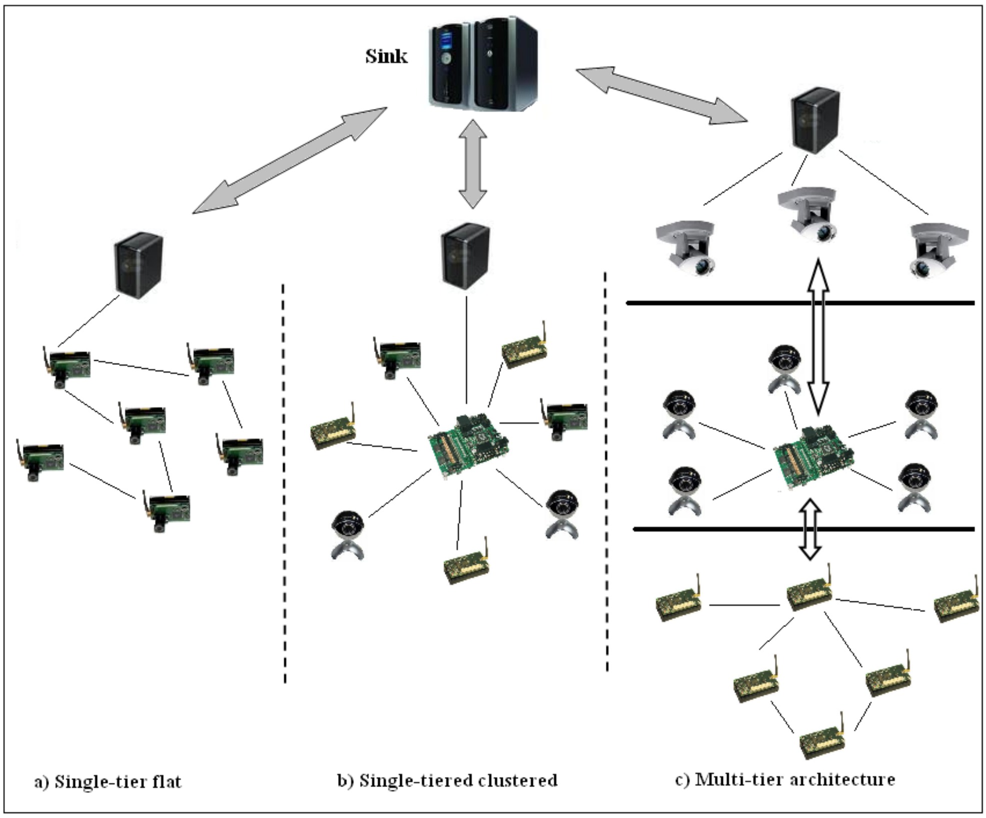


Figure 1.2: Network Architectures Models in WMSN

more complex tasks, like object tracking. Each tier may have a central hub to perform more data processing and communicate with the higher tier. The third tier is connected wirelessly with the sink or the gateway. This architecture can accomplish tasks with different needs with better balance among costs, coverage, functionality, and reliability requirements. On the other side, the use of just one node type in homogeneous flat network is not scalable enough to enclose all complexity and dynamic range of applications offered over WMSNs.

1.3 Thesis Contributions

The main contributions of this thesis can be categorized into four parts. The first part is based on the comprehensive study of the state of the art in all aspects of WMSN's proposed architectures, protocols, existing hardware and testbeds along with the mechanisms used to improve the energy efficiency of multimedia transmission over the sensor networks. The second part of the work studies the designed routing protocol (CMRP) that efficiently handles the communication of different data types. It also shows the network performance improvement from using cross-layer optimization technique between network routing and MAC scheduling. The third part of the contributions explores the design of a complete security system for protecting data content using a key management scheme and Intrusion detection system. The last part of the thesis gives a complete discussion and classification on the existing work done in event unobservability in sensor networks and describes our proposed privacy scheme for protecting contextual information in WMSNs.

1.3.1 Survey the State of the Art of All Aspects of WMSNs

|(1)| First, we survey up-to-date works and cover all research aspects of WMSNs, (e.g., network architecture, communication layer stack, cross layer design, challenge issues like security and coverage, and hardware and testbeds): We add complete descriptions and conduct comparisons among the proposed protocols in WMSNs (e.g., the physical layer technologies used in WMSN, MAC layer protocols proposed for WMSNs, methodologies used in designing the routing protocols in the routing layer, source coding techniques applied in the application layer), and we outline in detail the research challenges at different layers of the communication protocol stack. In addition, we add new classifications for each communication stack's protocols, and a complete taxonomy of the hardware platforms used in WMSNs based on their functionalities and capabilities pertaining to wireless motes, camera motes, and testbeds. Moreover, we review most of the hardware devices and prototypes used in WMSNs and compare them along with their specifications and features. Finally, we tried to stress on open issues for new researchers in this field and to give a view of what we foresee are going to be the future trends. `Almalkawi, I.T.; Guerrero Zapata, M.;`

Al-Karaki, J.N.; Morillo-Pozo, J. "Wireless Multimedia Sensor Networks: Current Trends and Future Directions". Sensors Journal 2010, 10, 6662-6717. [JCR-2010: 1.774 Q1] (14/61 Q1 INSTRUMENTS AND INSTRUMENTATION)

|(2)| Second, in order to reduce the energy consumption in WMSNs and to meet their resource scarcity, we discuss the proposed techniques and point to the research directions that overcome the challenging issues and limitations in order to improve the energy efficiency of multimedia transmission over WMSNs. Islam T. Almalkawi, Mohammad Alaei, Manel Guerrero-Zapata, Jose M. Barcelo-Ordinas, Julian Morillo-Pozo: "Energy Efficiency in Wireless Multimedia Sensor Networks" In IEEE COMSOC MMTC E-Letter. PP.17-20. Vol. 6, No. 12, 2011.

1.3.2 Proposing a hierarchal multipath routing protocol and QoS-aware cross-layer design for WMSNs

|(1)| Our proposed routing protocol, Cluster-based Multipath Routing Protocol (CMRP), aims firstly to cluster the nodes, so that cluster heads can do some aggregation and reduction of data in order to save energy consumption and bandwidth usage, and then to find the maximum number of paths suitable for the different requirements of handling different traffic classes. More specifically, the proposed routing protocol aims to satisfy the following design goals: (1) supporting different traffic classes of different delay and bandwidth requirements by choosing the suitable path for each data type, (2) maintaining minimum end-to-end delay suitable for real-time and non-real-time data packets to meet their playout deadlines, (3) achieving high throughput and packet delivery ratio by selecting the paths with better link quality and avoiding collisions and interferences, (4) saving energy at sensor nodes by moving the multimedia processing complexity as well as the aggregation process to the cluster heads' side, along with preventing path loops and path cycles in establishing the routes, and finally (5) providing load balancing and reliable data delivery by using multi-path routing protocol and two-level QoS-aware scheduling. Almalkawi, I., Guerrero Zapata, M., Al-Karaki, J. "A Secure Cluster-Based Multipath Routing Protocol for WMSNs". Sensors Journal 11, 4, 4401-4424 (2011). [JCR-2011: 1.739 Q1] (14/59 Q1 INSTRUMENTS AND INSTRUMENTATION)

|(2)| We improve the above work, to ensure correct delivery of real-time multimedia data and to utilize efficiently the limited resources, by proposing a solution that provides both Quality of Service (QoS) assurance and energy efficiency. In this contribution, we propose a cross-layer based routing protocol that can utilize MAC-layer QoS-based scheduling for more efficient routing mechanism in WMSNs. Our proposed optimization is based on clustered multipath routing protocol and adaptive QoS-aware scheduling for the different traffic classes in WMSNs. The design exploits the hierarchical structure of powerful cluster heads and the optimized multiple paths along with the adaptive scheduling to support reliable, high throughput, and energy-efficient multimedia transmission in WMSNs. The scheduling mechanism is based on adaptive QoS-aware TDMA approach used at two levels in the network: within clusters and among cluster heads. Our algorithm uses flexible time-slot assignment where a cluster head is responsible to schedule the traffic toward the sink from the sensor nodes based on the type of data and its availability. Almalkawi, I., Guerrero Zapata, M., Al-Karaki, J.: A Cross-Layer-Based Clustered Multipath Routing with QoS-Aware Scheduling for Wireless Multimedia Sensor Networks. International Journal of Distributed Sensor Networks. 2012. [JCR-2011: 0.203 Q4] (71/79 Q4 TELECOMMUNICATIONS)

1.3.3 Proposing security algorithms suitable for multimedia transmission over sensor networks

|(1)| In this contribution, we propose an implementation of a distributed and lightweight security mechanism of key management in order to secure the data communication among the nodes in clustered WMSNs against external attacks. Our proposed scheme of key management is lightweight -in terms of energy efficiency, processing and memory complexity, and communication overhead-, scalable for large scale network, and designed to facilitate the data aggregation at cluster heads. More specifically, the key management scheme is designed to: 1) Satisfy the basic security requirements such as authentication, integrity, and confidentiality without the need of a central key distribution. 2) Protect the network against the majority of outsider attacks, and to resist against insider attacks since the security keys it uses are unique and affect only the local cluster (*i.e.* the stolen material cannot be used in other clusters). 3) Be scalable for large-scale network because every node needs to generate only a small number of shared security keys regardless the total number of deployed nodes. We analyze analytically the effect of clustering the network on the scalability of our security algorithm and the number of needed security keys stored in each node. 4) Allow for message broadcasting within the clusters using unique-cluster security keys and for data aggregation processing since cluster heads can decrypt the sent data if necessary and update the corresponding information before relaying them toward the sink.

Our proposed security algorithm supports both authenticated encryption and authentication only services: with authenticated encryption, the data payload is encrypted using an encryption algorithm (such as MISTY1 or Skipjack algorithm) and the entire packet is authenticated with a message authentication code (MAC) using for example Hash-based Message Authentication Code (HMAC) based on cryptographic hash function or message digest (such as SHA1). Almalkawi, I., Guerrero Zapata, M., Al-Karaki, J. "A Secure Cluster-Based Multipath Routing Protocol for WMSNs". Sensors Journal 11, 4, 4401-4424 (2011). [JCR-2011: 1.739 Q1] (14/59 Q1 INSTRUMENTS AND INSTRUMENT

|(2)| In order to protect the network from advanced insider attacks and eliminate their threats, a second level of security is required. Therefore, we propose a lightweight distributed intrusion detection system (IDS) that can detect malicious attempts of exploiting possible security breaches and warn for suspicious nodes, even if these nodes are using legitimate security keys. To the best of our knowledge, there is no intrusion detection systems specific for WMSNs in the literature. Due to the especial requirements for delivering real time multimedia data, it is required significantly to have an efficient (fast and accurate) and lightweight (minimum overhead) IDS to detect possible intrusions in WMSNs. So, our proposed lightweight IDS is simple, with very little communication overhead, and efficient to identify malicious internal attackers in clustered WMSNs. The intrusion detection scheme prevents malicious attempts in each cluster by discovering compromised nodes both whether they are group member nodes (GMs) or the cluster head (CH) itself. Almalkawi, I.T.; Guerrero Zapata, M.; Al-Karaki, J.N. "Light-weight Security Scheme for Key Management and Intrusion Detection in Clustered Wireless Multimedia Sensor Networks". Submitted to the Journal of Networks and Computer Applications (JNCA), March 2013.

1.3.4 Discussing Event Unobservability in sensor networks and proposing location privacy scheme

|(1)| In this part, we introduce our complete classification for content security and contextual privacy in WSNs that is expected to guide in the design of new improved solutions for WMSNs. We focus in this work in revising the contextual privacy preservation in WSNs: define each form of contextual privacy, explain every possible attack methodologies, and survey the state of the art of existing countermeasures showing their advantages and drawbacks. More specifically, we first investigate the location privacy problem for the source, sink, and query location and review most of the proposed privacy-preserving techniques. Then, we analyze the protection of node identity privacy and explain its presented approaches. We also examine the temporal privacy issues and demonstrate the existing schemes related to this subject. Finally, we discuss the existing solutions comparing to each other and tried to stress on some interesting and challenging open issues for new researchers in this field. Almalkawi, I.T.; Guerrero Zapata, M.; Al-Karaki, J.N. "Event Unobservability in Wireless Sensor Networks: A Survey". Submitted to the Journal of Networks and Computer Applications (JNCA), February 2013.

|(2)| Although most of the security mechanisms used in sensor networks such as encryption, authentication, and intrusion detection allow sensor nodes to protect their transmitted "data content" from being exposed by external and internal attacks, and satisfy most of the needed security requirements (such as confidentiality, authentication, integrity, and availability), still they cannot fully address the location privacy of contextual content in WMSNs. Therefore, a third level of security protection is needed to offer contextual privacy of location information. In this contribution, we propose a source/sink unobservability scheme that hides the location information of important nodes in the network such as sources and sinks. This location privacy scheme avoids generating wide-network dummy messages by exploiting the used source coding technique in the application layer. Almalkawi, I.T.; Guerrero Zapata, M.; Al-Karaki, J.N. "An Efficient Source/Sink Location Unobservability for Wireless Multimedia Sensor Networks". To be submitted to the Journal of..., 2013.

1.4 Thesis Organization

The thesis document consists of seven chapters in total. This first chapter gives a general introduction to the work, lists the contributions of thesis, and summaries the project work. The second chapter reviews the background and related work done in WMSNs. The proposed cluster-based multipath routing protocol and the enhanced QoS-aware cross-layer design are discussed in details in chapter three. These correspond to the contributions mentioned in Section 1.3.2. The details on the implementation design of the security algorithms of key management and intrusion detection related to the contributions listed in Section 1.3.3 can be found in chapters four. The chapters 5 addresses a complete information on the contextual privacy in wireless sensor networks, and then chapter 6 proposes the source/sink location unobservability scheme for WMSNs regarding the contributions listed in the section 1.3.4 of the current chapter. The last chapter contains the conclusions and the future work.

1.5 Summary

Wireless Multimedia Sensor Networks (WMSNs) have emerged and shifted the focus from the typical scalar wireless sensor networks to networks with multimedia devices that are capable to retrieve video, audio, images, as well as scalar sensor data. WMSNs are able to deliver multimedia content due to the availability of inexpensive CMOS cameras and microphones coupled with the significant progress in distributed signal processing and multimedia source coding techniques.

These mentioned characteristics, challenges, and requirements of designing WMSNs open many research issues and future research directions to develop protocols, algorithms, architectures, devices, and testbeds to maximize the network lifetime while satisfying the quality of service requirements of the various applications. In this thesis dissertation, we outline the design challenges of WMSNs and we give a comprehensive discussion of the proposed architectures and protocols for the different layers of the communication protocol stack for WMSNs along with their open research issues. Also, we conduct a comparison among the existing WMSN hardware and testbeds based on their specifications and features along with complete classification based on their functionalities and capabilities. In addition, we introduce our complete classification for content security and contextual privacy in WSNs. Our focus in this field, after conducting a complete survey in WMSNs and event privacy in sensor networks, and earning the necessary knowledge of programming sensor motes such as Micaz and Stargate and running simulation using NS2, is to design suitable protocols meet the challenging requirements of WMSNs targeting especially the routing and MAC layers, secure the wirelessly exchange of data against external attacks using proper security algorithms: key management and secure routing, defend the network from internal attacks by using a light-weight intrusion detection technique, protect the contextual information from being leaked to unauthorized parties by adapting an event unobservability scheme, and evaluate the performance efficiency and energy consumption of employing the security algorithms over WMSNs.

Chapter 2

A State-of-the-Art Survey in WMSNs

2.1 Physical Layer in WMSN

The Physical Layer in WMSNs consists of the basic hardware transmission technologies of a network and defines the means of transmitting raw bits, rather than logical data packets, over the wireless link that is connecting network nodes. It is responsible also for frequency selection, modulation and channel encoding. In WMSNs, the physical layer should be designed in a way that it underlies all the higher-layer communications-related functions and meets the specific requirements and characteristics of WMSN. Therefore:

- The physical layer technology must work in a compatible way with higher layers in the protocol stack to support their application-specific requirements and to meet the design challenges of WMSN. This can be done with higher efficiency if a cross-layer model is used especially between physical layer and MAC layer.

- The physical layer should utilize the available bandwidth and data rate in the best possible way, and to be more power efficient.

- The physical layer should have a good performance (gain) against noise and interference and provide enough flexibility for both different channel and multiple

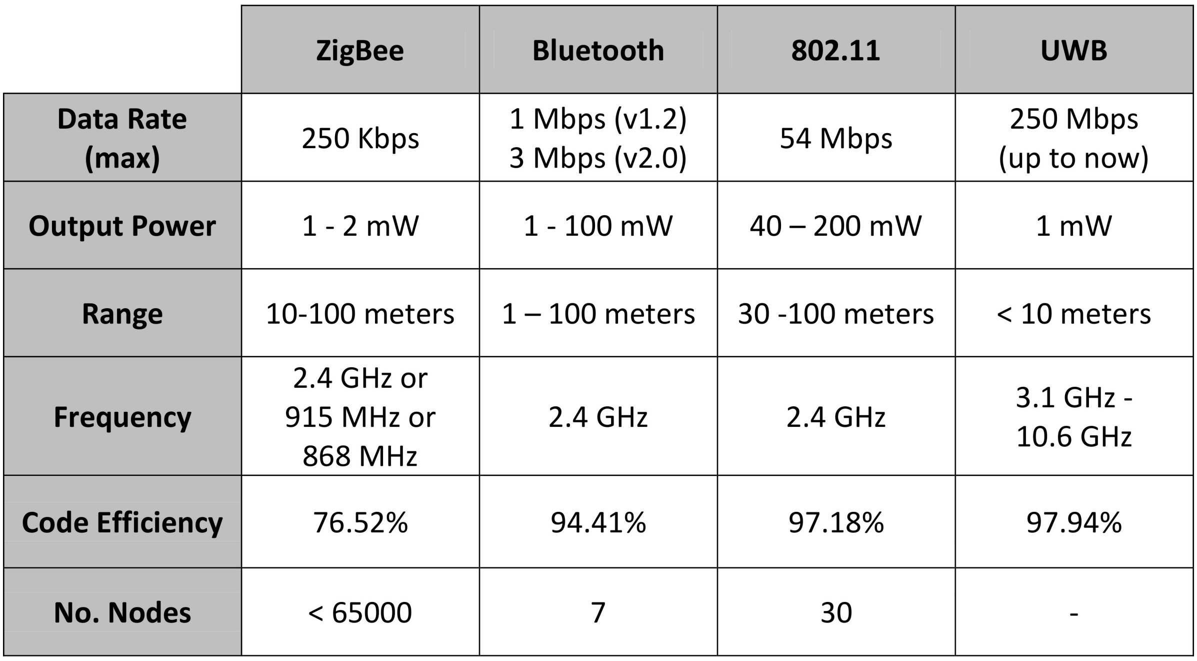
[1] Chapter 2 is based on the publications :
(1) Wireless Multimedia Sensor Networks: Current Trends and Future Directions;
Islam T. Almalkawi, Manel Guerrero Zapata, Gamal N. Al-karaki, Julian Morillo-Pozo. appeared in Sensors Journal 2010, Volume 10, Pages 6662-6717, 2010
(2) Energy Efficiency in Wireless Multimedia Sensor Networks; Islam T. Almalkawi, Mohammad Alaei, Manel Guerrero-Zapata, Jose M. Barcelo-Ordinas, Julian Morillo-Pozo. appeared in IEEE COMSOC MMTC E-Letter, Volume 6, Pages 17-20, No. 12, 2011

path selection.

- The cost of the radio should be taken into account since it will be deployed in large number of nodes.

Physical layer technologies can be classified either into three groups (Narrow band, Spread spectrum, Ultra-Wide band (UWB) technologies) based on the modulation scheme and bandwidth consideration [8], or into different standard protocols (IEEE 802.15.4 ZigBee, IEEE 802.15.1 Bluetooth, IEEE 802.11 WiFi, 802.15.3a UWB). ZigBee [9] is the most common standard radio protocol used in wireless sensor networks because of its lightweight standard and its low-cost and low-power characteristics. ZigBee supports: data rate up to 250kbps at 2.4 GHz, more than 65000 nodes, coding efficiency of 76.52 %, and range of 10-100 meters. ZigBee standard is being used by most of WSN devices such as MICA-family, Tmote sky, and imote2. However, ZigBee standard is not suitable for high data rate applications such as multimedia streaming over WMSN and for guaranteeing application-specific QoS. On the other hand, other standards like Bluetooth and WiFi have higher data rate and code efficiency -as shown in Table 2.1- but they consume more energy. Bluetooth has been used in [10] [11] for wireless communication in WMSN, while WiFi has been used with Stargate device in many projects as shown in the Hardware section later on.

UWB [12] [13] -with coding efficiency of 97.94%, data rate up to 250 Mbps, and nominal range of 10 meters in addition to its immunity to multipath propagation and precise positioning capabilities- has the potential to enable low power consumption, high data-rate of short range wireless communication and seems to be a promising candidate for the physical layer standard of WMSN. UWB spreads the information over a large bandwidth, about 20% of the center frequency or more than 500 MHz. The physical layer of UWB is implemented by using either impulse radio (IR) of extremely short duration pulses, or multiband orthogonal frequency division multiplexing (MB-OFDM) where hybrid frequency hopping and OFDM are applied. IR-UWB has simpler transmitter and rich resolvable multipath components, but it needs a long channel acquisition time and requires high speed analog-to-digital converters, while MB-OFDM-UWB offers robustness to narrowband interference, spectral flexibility, and efficiency but it needs slightly complex transmitter. The multiple access of IR-UWB can be realized by using direct sequence UWB (DS-UWB), or time hopping



	ZigBee	Bluetooth	802.11	UWB
Data Rate (max)	250 Kbps	1 Mbps (v1.2) 3 Mbps (v2.0)	54 Mbps	250 Mbps (up to now)
Output Power	1 - 2 mW	1 - 100 mW	40 – 200 mW	1 mW
Range	10-100 meters	1 – 100 meters	30 -100 meters	< 10 meters
Frequency	2.4 GHz or 915 MHz or 868 MHz	2.4 GHz	2.4 GHz	3.1 GHz - 10.6 GHz
Code Efficiency	76.52%	94.41%	97.18%	97.94%
No. Nodes	< 65000	7	30	-

Table 2.1: Specifications of the Physical Layer Standards in WMSNs

UWB (TH-UWB). The low duty cycle of IR-UWB (<1%), because of the short duration of the pulse, is a key advantage for low power consumption in WMSN, also spreading information over wide bandwidth decreases the power spectral density and in turn reducing the interference with other systems and lowering the probability of interception.

For the network layer, with the new characteristics of UWB such as low power transmission and low accurate ranging capabilities, the addressing and location-aware routing protocols can be optimized for better performance and they can get rid from the overhead caused by traditional flooding technique for routing and IPv6 scheme for addressing. Also UWB properties, especially the positioning capabilities, can be exploited to simplify the hardware used in location-aware routing instead of using of GPS enabled devices. It is pointed in [12] that UWB characteristics should be taken in account in the MAC layer for channel access, scheduling, and error control wherein the low duty cycle and low power transmission reduce the probability of collisions between pulses and interference. TH-IR-UWB system could be used, for example, for simultaneous transmissions based on the adoption of different time hopping code on each active link; and the rich resolvable multipath components of IR-UWB could be exploited, for example, at the receiver side for multipath diversity and channel estimation.

It is worth to point out that, in order to further increase capacity and mitigate the impairment by fading and co-channel interference, multi-antenna systems such as antenna diversity, smart antenna, and MIMO systems, can be combined with UWB. Since UWB has almost impulse-like channel response, the combination with multiple antenna techniques such as MIMO systems may give the possibility of short-range networks with multi-gigabit rates. However, these physical-layer techniques have many challenging problems to be developed for WMSNs. Although UWB appears to be a promising alternative physical layer technology and it has many attractive features, it is still not very mature and there are many challenges and issues that need to be resolved and better understood.

2.2 MAC Layer in WMSN

The design of highly efficient and reliable medium access control (MAC) protocols is critical in wireless sensor networking. Conventionally, the goal is to provide sufficient transmission capacity at the minimum energy cost under a moderate network load condition. A quick look into the existing MAC protocols for WSNs, as surveyed in [14], reveals that lack of standardization and application-specific diverse requirements has deprived WSNs from having a single de-facto standard MAC protocol.

MAC in WMSNs is essential to coordinate the channel access among competing devices. Given the energy constraints of the small, battery powered sensor devices, it is desirable that the MAC layer provides reliable, error-free data transfer with minimum retransmissions while meeting the QoS requirements (*i.e.* bit error rate, transmission rate, delay, fairness, etc.) with efficient resource utilization. MAC layer attempts to address these issues by enforcing channel access, scheduling policies (Figure 2.1), and error control. Therefore, a proposal of MAC layer protocol for WMSNs should satisfy the following features:

- maximize network throughput,
- enhance transmission reliability,

- minimize control overhead,
- be energy-efficient,
- and guarantee a certain level of QoS.

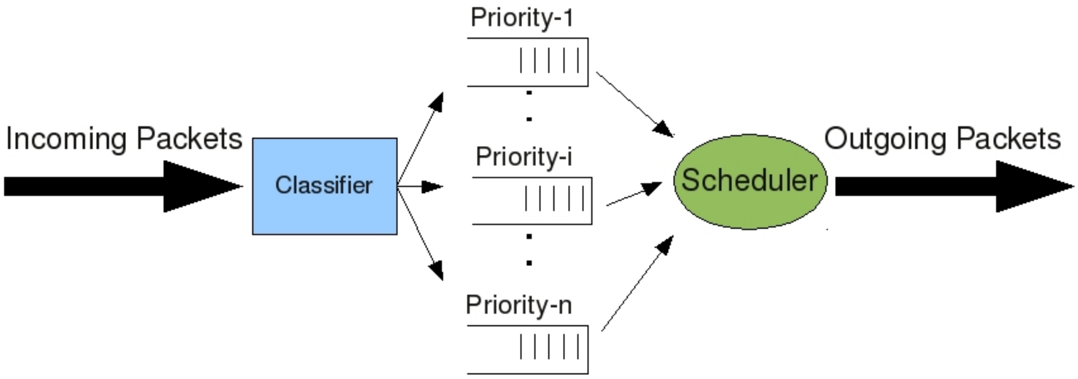


Figure 2.1: Traffic Differentiation and Priority Queueing in WMSNs.

2.2.1 The Affecting Characteristics of WMSN on MAC Protocol Design

In WMSNs, a sensor node may have various kind of sensors, as described in the Hardware section, that gather different types of data with different levels of importance. Therefore, WMSNs generate different traffic classes and these classes require classification, buffering, and different type of services (Figure 2.1). In addition, a WMSN normally demands larger bandwidth and entails higher network throughput to transport large volume of data to remote data sink rapidly and reliably. However, data rates provided by existing commercial sensor products, e.g. 250 Kbps in MICAz, pose some limitations to support multimedia traffic. On the other hand, current sensor nodes, such as MICAz and WINS, already support multiple channels for communication, for example, 40 channels in WINS [15]. Thus, the development of multi-channel MAC protocols, which can effectively utilize the available channel capacity appears as a research direction to achieve a better support for multimedia applications over WSNs.

Moreover, the design of the MAC layer depends on a trade-off between complexity/-cost and the network throughput, which is reflected in the literature of MAC protocol design for wireless sensor networks.

The work in [15] argues that a flat architecture is not suitable for multimedia applications because the transmission of the large volume of data resulting from multimedia applications will quickly drain the battery of the sensor nodes, thus, significantly reducing the network lifetime. Of course, multimedia transmission poses a research challenge and it is true that the multi-tier paradigm is very used in the literature of WMSNs. Although they can be of interest for some scenarios, we do not think, however, that the research focus should be put on the design of MAC protocols that assume some kind of *supernodes* (*i.e.* abundant power supply, out-of-band channel to communicate with the sink, etc.) that allow to get rid of the constraints of WMSNs (such as [15]).

2.2.2 MAC Layer's Main Functions

2.2.2.1 Channel Access

Traditionally, MAC protocols for WSNs can be classified into contention-based and contention-free protocols, according to the methods of cooperation in listen state between neighboring nodes. Contention-free slotted access protocols suffer from synchronization problem and many energy wastes because of synchronization overhead, in addition to channel under-utilization and fixed time-slot assignments. Regarding the structure of these protocols, they use many slots to access the channel and relatively long listening time that wastes more energy. Time Division Multiple Access (TDMA) is the most common example of this class. On the other side, contention-based protocols are based on the random access to the channel. This provides more flexibility to handle different nodes densities, lower delay, and better throughput potential at varying traffic loads. Also, there are no synchronization issues, making these protocols rather simple. The down side of this relaxed, random access approach is the wasted energy due to idle listening and collisions produced with large preamble and hidden nodes problems. Carrier Sense Multiple Access (CSMA) based with, possible, Collision Avoidance (CA) and its variants are examples of this type of protocols.

Simplicity, flexibility and scalability of contention-based random access protocols make them attractive for WSNs. However, transmitting multimedia applications with strict QoS-guarantee offers significant new challenges over these energy-constrained sensor networks and makes conventional MAC protocols not suitable for WMSNs. Design of an efficient sensory MAC-protocol, satisfying QoS requirements, is one major step in end-to-end QoS provisioning over WMSNs.

Most Contention-based MAC protocols in WSNs, such as S-MAC [16] and T-MAC [17], were proposed to support single-channel architecture, as shown in Table 2.2. [15] argues that these protocols are not suitable for multimedia applications (because they are designed to be energy efficient at the cost of increased latency and degraded network throughput) and proposes a multi-channel MAC protocol. Of course, neither S-MAC nor T-MAC was originally thought for WMSNs and probably they would not work properly under the new requirements of such networks. We think that exploiting the multi-channel features in the existing sensor platforms is a promising direction in designing a MAC protocol for WMSN, but it is also a mistake to discard the single-channel paradigm. In fact, we will see a proposed single-channel MAC protocol for WMSNs, [18] (further developed in [19]), that clearly outperforms both T-MAC and S-MAC in terms of MAC latency (both T-MAC and S-MAC attain an average transmission delay of 60 ms, while the delay of [19] is less than 30 ms), MAC throughput (while S-MAC and T-MAC achieves a throughput of 20 Kbps and 10 Kbps, the proposal in [19] achieves an average throughput of 50 Kbps), and energy efficiency (it consumes less energy than S-MAC and 14-18% more than T-MAC).

COM-MAC is an on-demand multi-channel contention-free MAC protocol proposed in [15]. It exploits the fact that current sensor nodes, such as MICAz and WINS, already support multiple channels for communication, e.g. 40 channels in WINS, to develop a multi-channel MAC protocol in order to effectively utilize the available channel capacity through cooperative work from the other sensor nodes. In this way, a better support for high data rates demanded by multimedia applications can be achieved.

As an example of a single-channel MAC protocol, a MAC Protocol for WMSNs is presented in [18] and [19]. It argues that CSMA methods generally offer a lower delay and better throughput, especially at lower traffic loads. Thus, it is based on the CSMA/CA MAC methods to develop a QoS-based MAC protocol for WMSNs. It adaptively adjusts the contention window depending on the QoS requirements and wireless channel characteristics and dynamically adjusts its duty cycle based on the major application traffic to preserve the sensor energy without sacrificing QoS provisioning. It updates the contention window to achieve trade-off between the period wasted on the waiting for the back-off counter to expire and the collisions because of the simultaneous transmissions of more than one sensor nodes. Subsequently depending on the traffic class, it differentiates the packets into different types and updates the contention window in different amount; for higher priority traffic classes (like streaming video) the increment step size is set to be smaller than that of the lower priority while the decrement step size is set to be greater than that of the lower priority. In this way, throughput differentiation between different traffic classes can be easily controlled and adjusted by controlling this step size. As an example, authors in [19] consider a network where the mean capacity of the links is 100 Kbps and show how they can move from a scenario where the streaming video traffic achieves a throughput of 50 Kbps followed by the 40 Kbps throughput of lower priority classes, to a scenario in which streaming video achieves a throughput of 75 Kbps at the cost of 15 Kbps throughput for lower priority classes. They also demonstrate the throughput dynamics of different traffic classes in both the presence and absence of highest priority streaming video traffic: when the streaming video is on, it obtains 45 Kbps while leaving 40 Kbps to lower priority traffic classes. When it is switched off, lower priority classes obtain 80 Kbps and when it is switched on again, it quickly re-attains a higher throughput of 50 Kbps, at the cost of the reduction of lower priority traffic classes throughput to 40 Kbps. Similar conclusions are drawn regarding delay: the lower contention window of streaming video traffic gives it the lowest delay of 10 ms, when compared with 30 ms or 70 ms of lower priority traffic classes. For energy conservation, the proposed MAC protocol is trying to save energy by dynamically changing the duty cycle of the idle listening with the current traffic condition. Of course, in such a strategy, there exists a trade-off between energy consumption and delay. Results in [19] show this phenomena: for a pretty low latency of 10 ms the energy consumed by the protocol is close to 30 mWHr, while if a relatively high latency of 20-30 ms is allowed, the energy consumption reduces to less than 15 mWHr.

A cross-layer communication approach is presented in [20], between the MAC and physical layers, to provide a better QoS for WMSN applications. It is based on the Time-Hopping Impulse Radio Ultra-Wide-Band (TH-IR-UWB) transmission techniques. While [18] improves protocols like S-MAC and T-MAC but still based on CSMA/CA, [20] takes a totally different approach and discards CSMA/CA (as COM-MAC did). This architecture tries to solve the shortcomes of using CSMA/CA for the MAC layer in WSN such as the variable and uncontrollable access delays from using random timer, idle listening from using carrier sense, and increased energy consumption due to for example hidden node problem with the objective of providing QoS for WMSNs. An evaluation through simulation is performed for a network of 49 nodes, taking into account two different groups of traffic flows with different QoS demands (flows in group 1 require 100 Kbps bandwidth, 100 ms. end-to-end delay, and 0% PER while flows in group 2 have 500 kbit/s bandwidth demand, 100 ms end-to-end delay and can admit 10% PER. Results show that sources in group 1 have a throughput of exactly 100 kbit/s, while sources in group 2 show an average throughput of about 480 kbit/s, as some packets are lost. While flows in group 1 do not lose packets, flows in group 2 lose approximately 4% of the packets, which is still below the application requirement. The 0% packet loss in group 1 directly translates into a consistently higher energy consumption which is not studied in this work. Moreover, the aggregate average end-to-end delays of the two groups are well below the threshold end-to-end delay, which suggests that simulations with more restrictive conditions should be performed to better understand the behaviour of this proposal (for example, how the admission control works or how the coordination overhead affects the overall system). On the other hand, the differences in delays between flows in the same groups are very limited between different flows, which demonstrates the basic fairness of the system, and the variance of the delay is also limited. Authors in [20] argue that this shows that under normal circumstances the system leads to much more limited jitter as compared to CSMA/CA based systems. A direct comparison between the system proposed in [20] and a CSMA/CA based system (like, for example, [19]) would be, however, of great interest to verify that.

2.2.2.2 Scheduling and Admission Control

Scheduling, admission control, and buffer management in WMSNs is an open research issue that has attracted the research community in last years but still not really solved.

MAC Protocol	(Single/Multi)- Channel	Contention- (based/free)	Diff. Service	Topology	Cross- Layering
T-MAC [17]	single	free (scheduling-based)	no	clustered	no
S-MAC [16]	single	free (scheduling-based)	no	clustered	no
B-MAC [21]	single	based	yes	flat	yes
MMAC [22]	multi	based (IEEE 802.11)	no	flat	no
MMSN [23]	multi	based	no	flat	no
COM-MAC [15]	multi	free (scheduling-based)	yes	clustered	-
Diff. Service Model [24]	-	-	yes	flat	-
MAC Protocol in [18]	single	based (CSMA)	yes	flat	no
Cross-layer Architecture in [20]	multi (UWB)	free	yes	flat	yes
EQ-MAC [25]	single	based (collision-free)	yes	flat (static)	yes
Node Admission [26]	single	free (TDMA)	no	flat	yes
UWB Technology in [12]	multi	free	yes	flat	yes

Table 2.2: A Comparison between MAC Layer Protocols. Grey rows indicate that the MAC protocol is designed for WSNs but not specifically for WMSNs.

As an starting example, the work in [15] makes some strong assumptions to be considered: on one hand, it uses a coordinated channel scheduling that assumes a *relatively static* network to overcome the contention overhead incurred by multi-channel MAC protocols. On the other hand, it argues that a *flat* structure is not suitable for WMSNs. The protocol operation is divided into three sessions (*Request*, *Scheduling* and *Data Transmission*) and only a heuristic is provided to calculate the scheduling to be used based on priorities. In the cluster-based MAC protocol, COM-MAC [15], a scheduled multi-channel medium access protocol is considered within each cluster where the cluster head coordinates the communication among its members in a contention free manner within both the time and frequency domains. By this way, the nodes, which are assumed to have multiple transceivers being able to operate on a set of available channels simultaneously, avoid collision, idle listening and overhearing problems. The cluster head dynamically allocates time slots and channels for its members according to the current QoS requirements and network traffic status.

One work that attacks the scheduling and buffer management problem is [24] based on the fact that different network applications need different QoS requirements such as packet delay, packet loss, bandwidth and availability. And this can be done not by increasing network capacity (as COM-MAC [15]), but by developing a network architecture which is able to guarantee QoS requirements for high priority traffic. It argues that the sensor networks should be willing to spend more resources in disseminating packets that carry more important information by using a differentiated service model for WMSNs. The proposed model can support two major different types of traffic classes: real time class (Expedited Forwarding or EF) and non real time traffic (Assured Forwarding or AF) which is divided into three classes: high priority AF1, medium priority AF2, and low priority AF3. In this model, real time traffic is buffered in a separate queue with low buffer size while non real time traffics are managed by using random early detection (RED) queue management in separate queues also. The delay performance of different traffic classes is evaluated by using two scheduling mechanisms: Priority Queuing (PQ) and Weighted Round Robin (WRR). It is shown that by using priority queuing (PQ) scheduling mechanism for EF traffic class and weighted round robin (WRR) scheduler for non real time traffic classes, low delay bound and guaranteed network bandwidth for high priority real time traffic can be provided. This work, as it simply considers a scheduling system, does not provide any insight on the physical and MAC layers of WMSNs. The main drawback of this work, however, is that it demonstrates that the proposed system can provide differentiated services but it does not study signaling overheads or energy consumption.

In concordance with [20], the work in [12] shows also that UWB characteristics should be taken into account in the MAC layer for channel access, scheduling, and error control wherein the low duty cycle and low power transmission reduce the probability of collisions between pulses and interference.

A theoretic work done in [26] tries to give insights in where and how to deploy sensor nodes (and how many of them) so that all the nodes can be supported by the limited communication resources in WMSNs. It assumes low-mobility or a static network, flat topology, and a single-channel TDMA-based communication to present a cross-layer design model where node admission in WMSNs and its interaction with resource management and link scheduling is investigated. The interaction is formulated as a two-stage optimization problem: first stage is to maximize the number of admitted sensor nodes, and the second stage is to maximize the network life time. It is shown also that this optimization problem can be presented in equivalent one-stage optimization problem with more compact mathematical form. Note that a common characteristic in all the presented proposals is that some kind of service differentiation (Figure 2.1) is provided: it seems clear that without this feature it is not possible to guarantee the QoS needed by WMSNs.

2.2.2.3 Error Control

Due to the unreliability of the wireless medium in WMSNs, the transmitted data such as multimedia content is exposed to losses or errors mostly caused by multi-path fading, co-channel interference, jamming ... etc. Therefore, in order to improve the perceptual quality of the received multimedia content, techniques for error correction and loss recovery should be employed in WMSNs to combat the unreliability of the wireless channel at the physical and MAC layer. Forward Error Correction (FEC) and Automatic Repeat Request (ARQ) are examples of these techniques. The usefulness of ARQ in sensor network applications is limited by the additional re-transmission cost and overhead. On the other hand, decoding complexity is greater in FEC, as error correction capabilities need to be built-in. Considering this, simple error control codes with low-complexity encoding and decoding might present the best solution for sensor networks. In the design of such a scheme, it is important to have good knowledge of the channel characteristics and implementation techniques.

A comparison between two techniques of error compensation of transmission distortion in multipath WMSN is conducted in [27]. First technique is the Error Concealment (EC) algorithm that reconstructs the distorted multimedia data as closely as the original one by utilizing the use of modified discrete wavelet transform for embedding downsized replicas of original image into itself. EC does not need increasing in bandwidth demand as well as retransmissions and consequent delay, however EC algorithm needs more processing power at the source and sink nodes comparing to the other technique. The second technique is the Forward Error Correction (FEC) based on Reed-Solomon coding that compensates and corrects the wireless link errors by utilizing the use of redundant data. FEC technique can mitigate the wireless link impairments but it cannot solve the errors from instant node failure problem and require considerable increase in the transmission bandwidth. It is shown in [27] that EC technique with multipath routing is more promising than FEC based RS coding to compensate for error and losses in WMSN.

2.2.3 Research Issues

In this section we presented a set of representative research efforts regarding MAC for WMSNs: single-channel/multi-channel and/or scheduled/contention-based proposals appear in the literature. In our opinion, multi-channel MAC protocols are more suitable for WMSNs. We have also identified different considered topologies. Although it can be interesting to exploit to some extend the use of multi-tier WMSNs in the design of MAC protocols, this should not be used to simplify the problems that the transmission of multimedia content represents. In this sense, papers that consider a flat topology are more research challenging.

We discussed the cross-layer design dependencies between MAC layer and other layers of the communication stack especially the physical layer, in the case of using UWB technology. The adoption of the UWB technology as the underlying transmission technique in WMSNs and the potential challenges in this area, appear as an interesting research topic. We believe that this research area will attract the attention of many researchers and boost the applicability of UWB in multimedia networking.

In overall, we think that cross-layering is essential for efficient MAC designs in WMSNs, together with queue-management and traffic classification/prioritization as long as QoS is required for multimedia traffic.

2.3 Routing Layer in WMSN

Routing layer in wireless sensor network aims to deliver the sensed data from the sources to the sink node taking into account several design considerations, such as energy efficiency, link quality, fault tolerance, and scalability. Although there are many routing protocols proposed for the traditional WSN, the design of routing protocols for

WMSN is still an active research area. We believe that the new characteristics and constraints due to the multimedia content handling over the network make the proposed routing protocols for WSNs not directly applicable for WMSNs. The multimedia nature of the collected information (video streaming, still images, audio) adds more constraints on the design of the routing protocols in order to meet the application-specific QoS requirements and network conditions.

There are many traffic classes in WMSNs and can be categorized into three main classes or services depending on their QoS requirements: 1) Event-driven service which is delay intolerant and error intolerant but it requires less bandwidth, so a path with a little traffic and high signal to noise ratio is attractive for this kind of service. 2) Data query service is error intolerant but query-specific delay tolerant applications, so a path with significant congestion and a high signal to noise ratio may be used for this service. 3) Stream query service which is delay intolerant but query-specific error tolerant application (in a sense packet losses can be tolerated to a certain extent), so a path with less traffic and relatively lower signal to noise ratio is better for this type of service.

2.3.1 Routing Methodologies in WMSN

The recent work in routing layer of WMSN, as shown in Table 2.3 and summarized below, tries to handle these new characteristics of WMSNs and its design challenges by either modifying the previous work done in WSNs (e.g. using multiple performance metrics to meet the additional QoS requirements), or proposing new solutions based on new methodologies (e.g. using multi radio or MIMO systems, switching between multiple channels, selecting multi routing paths, or mixing among these methods). Moreover, we should point to the fact that these additional challenges and requirements of WMSNs, mainly by streaming real-time multimedia content, open the call for new research on cross layer design for more optimizing solutions. For example, cross layer design between multimedia source coding techniques at the application layer and the routing protocol in the routing layer can be exploited for better multipath selection or in-network processing. Also, cross layer design between the routing layer and the MAC layer can allow for packet-level service differentiation or priority-based scheduling and for more power efficient routing mechanisms.

Routing Protocol	Methodologies				
	Multi-path	Multi-channel	Geographic	Multi-radio	Hierarchical
ASAR [31]	✓				✓
TPGF [32]	✓		✓		
Swarm-based LANMAR [34]				✓	✓
Radio-Disjoint routing [35]	✓				
Modified Direct Diffusion [36]	✓				
PPDD-based QoS routing [37]	✓	✓			
M-IAR [39]			✓		
Multimedia-aware MMSPEED [41]	✓		✓		
QuESt [44]	✓				

Table 2.3: Methodologies used in proposed routing protocols for WMSN

The work in [28] presents an Ant-based Service-aware routing algorithm (ASAR), a QoS routing model for wireless multimedia sensor network, that chooses appropriate paths for different QoS requirements from different types of services. The proposed algorithm mainly addresses the routing scheme between the cluster head and sink node in which a cluster head transfers the different classes of data. The process starts at the cluster head when it generates the ants for each type of service and then depending on the objective function of each type of data and pheromone value of each path, different paths are found to meet the different QoS requirements. In order to quicken the convergence of the algorithm and optimize network resources, the algorithm quantifies the pheromone value on the sink to decrease the sending frequency of reverse ants or control messages. Presented results show that: ASAR has a significant advantage over Dijkstra and Directional Diffusion (DD) in most metrics; except in packet loss rate for stream query service and delay when compared with DD.

A geographic routing algorithm, Two-Phase Greedy Forwarding (TPGF), is proposed in [29]. It explores near shortest hole-bypassing node-disjoint routing paths for WMSNs. TPGF supports multipath transmission by repeatedly executing the algorithm to find more on-demand node-disjoint routing paths. TPGF also supports near shortest and hole-bypassing path without including the face routing or planarization algorithms in order to maximize the number of available paths. It assumes that all the nodes know their location information and the source nodes know also the location of the base station. The first phase of TPGF is responsible for finding the possible routing path and it consists of two steps: greedy forwarding and step back & mark. The process starts at a source node that chooses the next-hop node which is closest to the sink among all the neighboring nodes and so on. In case that the next-hop does not have any farther neighbor except the previous node, it steps back to its previous-hop and marks itself as a block node. The second phase of TPGF is responsible for optimizing the found routing paths with the least number of hops and by eliminating the paths circles (if any) in the found paths using label based optimization technique. Presented results show that the average path length obtained by TPGF is much shorter than the one obtained by GPSR [30].

The work in [31] suggested to use landmark ad hoc routing protocol (LANMAR) in WMSNs with deploying limited number of mobile swarms, in which the network is divided into groups (LANMAR groups) and each group has a landmark node which is dynamically elected. A swarm is a group of nodes physically close to each other and usually share the same mobility pattern. Comparing to other sensor nodes, the swarm nodes have better capabilities in terms of hardware functionalities and networking capabilities (such as high quality video camera, multiple long radio range, large channel bandwidth, and maybe ability to communicate with satellites) and they can move with relatively high speed. An example of mobile swarm can be a group of tanks or unmanned aerial vehicles (UVH) moving together. The mobile swarms can communicate and exchange information between each other by using satellite communication or mobile backbone network (MBN). With the help of the limited number of mobile swarms, high quality of multimedia streams can be supported in large-scale sensor network. Once there is a hot or interested spot, a swarm can be directed to that area to help forwarding high quality multimedia streams. Results presented in this work show that the delivery rate and average end-to-end delay of their proposed protocol (Swarm-based LANMAR) outperforms LANMAR and AODV.

A non interfering disjoint multipath routing for WMSN is proposed in [32]. It addressed the problem of interference between multiple paths using one channel in WMSN and suggested an incremental on-demand approach in which only one path is built for a given source and additional paths are built when required in case of path congestion or lack of bandwidth. In order to solve the problem of interference between close paths, the proposed solution forces the multipath routing to build paths that are not interfering with each other from the beginning by putting all the interfering nodes of a given path in a passive state. Passive nodes do not further participate in building any other path in future and consequently will not interfere with previously built paths. The process starts at the sink when it floods the network with requests until they reach the source nodes. The source node starts immediately sending data on the selected paths and all the intermediate nodes between the source and the sink will inform their neighbors to switch to the passive state. The proposed work argues that putting some nodes in a passive or sleep mode increases the overall throughput and reduces the consumed amount of energy in the network. Results obtained through simulation show that the proposed protocol achieves better throughput with less energy consumption by using fewer non-interfering paths when compared to multipath schemes without interference awareness.

Modifications on, Direct Diffusion, the routing protocol for WSN are done in [33] to support multipath routing for WMSN based on link quality and latency metrics. One of the modifications includes using Costp, which is a product of expected transmission count (ETX) and delay, as a performance metric instead of pure delay that was used in Direct Diffusion. Since close paths interfere with each other and consequently have poor SNRs which are indirectly used to estimate ETX values, they have less probability to be selected by using Costp metric and this also will lead to increase throughput. The other modification is to reinforcing multiple links at the sink to obtain disjoint path from the source, and in order to match multipath routing. However, this routing protocol does not consider the bandwidth as QoS metric for routing decision or prioritizes the incoming packets to schedule them but it does consider the playout deadline in a sense that the data arrives after the deadline will be discarded. Results, show that the presented protocol for multipath video streaming over WSNs obtains higher throughput (even the double in some conditions) than its single path counterpart (EDGE) through the use of multiple disjoint paths.

A design of QoS aware routing protocol is presented in [34] to support high data rate for WMSNs by ensuring bandwidth and end-to-end delay requirements of real time data and maximized throughput of non-real time data. The routing protocol uses multiple paths, multiple channels, and QoS packet scheduling technique based on the dynamic bandwidth adjustment and path-length-based proportional delay differentiation (PPDD) techniques to meet the bandwidth and delay requirements respectively. These requirements (bandwidth and delay) are adjusted locally at each node based on the path-length and incoming traffic in static flat wireless network where all the nodes are homogeneous multimedia sensor nodes capable of performing all possible application tasks (video, audio, scalar data) and equipped with single radio interface and multi-channels. QoS-aware packet scheduling policy considers different priorities for real time packets and non-real time packets by using at each node a classifier that checks the type of the incoming packets and sends to appropriate queues, and a scheduler that schedules the packets according to the delay and bandwidth requirements. The proposed protocol -as shown through simulation- clearly improves average delay per real-time packet, average lifetime of a node, and throughput of non-real-time data when compared with single-r and multi-r mechanisms [35].

An extension for a routing protocol, multimedia enabled Improved Adaptive Routing (M-IAR), is done in [36] to enable handling multimedia content by taking into account two extra QoS parameters, end-to-end delay and jitter. M-IAR is a flat multi-hop routing protocol that exploits the geographical location of the sensor nodes, by assuming that all the nodes know their positions and the positions of their neighbors and the sink node, in order to find the shortest route containing the least number of nodes between the source and the sink. M-IAR uses two types of ants: forward ant and backward ant. Forward ant is used by the source node to explore the path toward the sink and selects the next-hop neighbor with the highest probability according to the mentioned metrics. The backward ant is used by the sink node and uses the global information from the forward ant to update the probability values and reinforce the visited nodes. Unfortunately, the authors of the paper do not compare the results obtained from the proposed protocol with any other protocol.

The work in [37] uses the game theory and ant colony algorithm to solve the problem of QoS routing in WMSN. The idea of using game theory together with the ant colony algorithm is to overcome the shortcoming of the current ant-based routing protocols for WMSNs such as: the long time needed by forward ants to find the destination and the overhead occurred from using plenty of backward ants to update the routing probability distribution. Therefore, game theory is used depending on the assumption that the sensor nodes are rational and have selfish action (*i.e.* they try maximize their payoffs with minimum cost). Unfortunately, the paper does not obtain any kind of performance nor does compare the proposed protocol with any other protocol.

Multimedia-aware Multipath Multi-Speed (Multimedia-aware MMSPEED) routing protocol is proposed in [38].Multimedia-aware MMSPEED is an extension over MMSPEED routing protocol to take into account the embedded information in the received packets in which near optimum path is reserved for I-packets and marginal paths are used for P-frames. MMSPEED protocol [39], which is also an extension for SPEED protocol [40] that was designed for WSN, can differentiate between flows with different delay and reliability requirements and has significant potential in video transmission applications. However, experimental results in [38] show that MMSPEED is not compatible with some special features of Multimedia traffic such as high video frame rate and packet's information dependency.

QoS-based energy-efficient routing protocol is proposed in [41] called QuESt that is capable of carrying multimedia content over WMSNs. The proposed routing protocol aims to build set of non-dominated paths that satisfy application-specific QoS parameters based on using multi-objective genetic algorithm (MOGA). It is shown that determining optimal routes satisfying multiple QoS parameters (end-to-end delay and bandwidth) simultaneously in energy-constraint sensor network is a NP-complete problem, hence the authors prefer to use MOGA algorithm as a tool to solve this NP-problem by treating the multiple QoS parameters independently without combining them into a single objective function. The end-to-end delay is modeled by using Weibullian distribution to capture the inherent long-range dependency of data traffic, and the bandwidth is modeled by taking the product distribution of the individual links. It is assumed to have one single sink and multiple sources in the network, then the possible routing paths between the sources and the sink are found using depth first search (DFS). These initial paths then are fed to the MOGA algorithm to give a status (fitness value) for the QoS parameters for each path. Finally, the proposed routing protocol will select the path that is suitable of each type of data traffic based on the QoS parameters status.

2.3.2 Future Research Directions

In general -as we believe- protocols inspired in ant colonies and game theory will keep being published. They will look good on paper, specially with all those formulas. Nevertheless, nobody will consider seriously to put them into real use because deep down everybody will suspect that less glamorous but more down to earth approaches are the way to go. In addition, many of those protocols have a too long adaptation time to routing topology changes to be feasible to many scenarios.

Multipath routing is the way to go in WMSNs because these wireless networks need to exploit the network bandwidth to its limit and sometimes in short bursts. Of course, that makes everything more complicated, affects transport layer, and introduces many new problems. Nevertheless, we foresee this to be the approach that will win in the long run. But, the future proposals will have to be designed as integral solutions that cover routing, MAC layer, security and sometimes even transport layer.

Geographical routing has been touted by many as the routing that suits best WMSNs [42] [43] [44]. Nevertheless, we foresee exactly the opposite. This is due to the fact that these mechanisms require that nodes know their geographical positions (most of the times assuming they have a positioning system like GPS) being arguably an unfeasible requirement for many scenarios. In addition, these routing protocols have no security provisions at all. Thus, very simple attacks can be devised against them. Moreover, in WMSNs bandwidth is such a big issue, that more complex routing protocols that discover better routes are going to be required.

Quality of Service is clearly a needed feature in many WMSNs, specially if there is some sort of real time streaming. Nevertheless, there will be many scenarios in which the use of different message priorities are going to give more or less the same performance with less complexity. Expect that sooner or later the general recommendation in this issue will be: If you don't really need QoS, use message priorities.

2.4 Transport Layer in WMSN

Transport layer is a group of protocols that run over the network layer to enable end-to-end message transmission. Transport layer aims to provide several services such as: same order delivery, data reliability and loss recovery, flow and congestion control, and possibly QoS requirements (e.g. fairness and timing). TCP and UDP are examples of standard transport protocol that are currently used for the Internet. However, these traditional transport protocols cannot be directly implemented over wireless sensor network [45] [6] because WSN in general and WMSN in particular have their distinctive features, which make them different than typical Internet network, and they have very wide range of applications that need special requirements. Some of the features of WMSN are the following:

- *Network topology*: The network topology of WMSN is dynamic due to wireless link condition and node status, and generally it takes the shape of multihop many-to-one (like a star-tree) topology that is either flat or hierarchal. These variations in network topology should be taken into account in designing a transport protocol for WMSN.

- *Traffic characteristics*: Most of the traffic in WMSN is generated from the source nodes toward the sink and, depending on the application, this traffic can be continues, event-driven, query-driven, or hybrid. Also in many cases, the source node can send its multimedia traffic using multipath route to the sink and this feature can be exploited to design a suitable transport protocol for keeping the quality of multimedia streaming.

- *Resource constrains*: The sensor nodes have limited resources in terms of battery power, communication bandwidth, and memory that require less expensive and more energy efficient solutions for congestion control and reliability.

- *Application-specific QoS*: As we mentioned before, WMSN has diverse applications from surveillance and target tracking to environmental and industrial applications. These applications may focus on different sensory data (scalar, snapshot, or streaming) and therefore they need different QoS requirements in terms of reliability level, real-time delivery, certain data rate, fairness, etc.

- *Data redundancy*: Collected sensory data - in general- in WMSN has relatively high redundancy and hence many WMSN applications use multimedia processing, such as feature extraction, data compression, data fusion, and aggregation to decrease the amount of data while keeping the important information. Therefore, reliability against packet loss becomes an issue in WMSN especially if these packets contain important original data such as Region of Interest (ROI).

UDP (User Datagram Protocol) uses a simple transmission model without implicit hand-shaking mechanism to provide timeliness for real-time applications like streaming media, but it does not guarantee data reliability, nor does it provide flow and congestion control. On the other hand, TCP (Transmission Control Protocol) is connection-oriented transport protocol based on 3-way hand-shaking mechanism to provide reliable and ordered delivery of data. However, TCP has several disadvantages with respect to WMSN, which are:

- Overhead of the connection establishment mechanism in TCP might not be suitable for event-driven applications.

- TCP uses end-to-end congestion control that requires longer response time (comparing with hop-by-hop control) and may cause more packet loss in case of congestion.

- The reliability mechanism in TCP is also based on end-to-end retransmission which consumes more energy and bandwidth than hop-by-hop retransmission.

- TCP assumes that packet loss is due to congestion only and hence triggers the rate adjustment process to reduce the traffic rate whenever it detects packet loss. This behavior in TCP leads to decrease the throughput in WMSN because congestion is not only the reason for packet loss, also wireless link condition and bit-error level cause packet loss that cannot be solved by rate reduction.

- Fairness is an issue in TCP, because congestion control mechanism in TCP can discriminate against sensor nodes that are far away from the sink node.

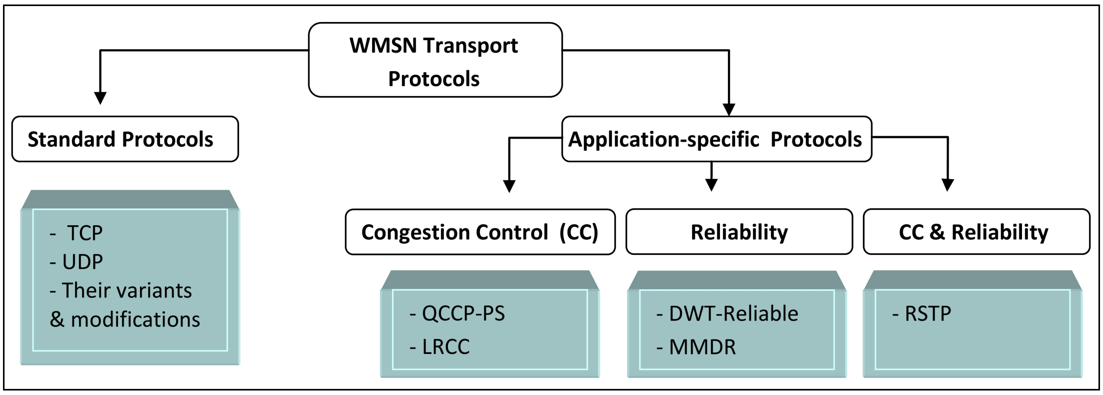


Figure 2.2: WMSN Transport Protocols Classification.

2.4.1 Congestion Control

Congestion control is one of the services done by transport layer protocols to mitigate congestion in the network. Congestion, in wireless sensor networks, not only wastes the scarce energy due to a large number of retransmissions and packet drops, but also hinders the event detection reliability and link utilization. As we pointed before, there are two main reasons for causing congestion in WMSN: The first one is called node-level congestion that is due to the packet arrival rate exceeds the packet-service rate causing buffer overflow in the node and can result in packet loss, and increasing queuing delay. This is more likely happen at sensor nodes close to the sink, as they usually carry more combined upstream traffic. The second one is link-level congestion that is related to the wireless channel condition due to contention, interference, and bit-error rate. Congestion control mechanism consists of three steps: congestion detection, congestion notification, and rate adjustment. There are two ways to detect congestion either using active method such as timer or acknowledgment, or proactive method using queue length as in QCCP-PS [46], LRCC [47], CODA [48], packet service time as in CCF [49], or ratio of packet service time over packet inter-arrival time as in PCCP [50]. Congestion notification can be done either explicitly by using special control messages as in LRCC [47], or implicitly by piggybacking congestion information in normal data packets (e.g. using congestion notification, CN, bit as in QCCP-PS [46], Fusion [51], CCF [49]). Rate adjustment is done by step-by-step decreasing of the traffic rate into the congested area in case of using CN bit as in QCCP-PS [46], or using accurate rate adjustment if there is enough available information as in PCCP

[50]. Some techniques as in LRCC [47] try to reduce the traffic rate into the congested area, but in the same time, avoid reducing the source stream traffic rate -in order to maintain the received quality of multimedia streaming- by balancing the stream traffic over multiple paths and reducing the traffic toward the current congested path.

A Queue based Congestion Control Protocol with Priority Support (QCCP-PS) is presented in [46] to deal with congestion control in the transport layer. QCCP-PS focuses only on congestion control as it is very important for WMSN to support different applications. QCCP-PS congestion control mechanism is based on hop-by-hop approach and consists of three parts: congestion detection unit, congestion notification unit, and rate adjustment unit. To detect congestion, QCCP-PS uses the queue length as an indication of congestion degree where there is separate queue to store input packets from each child node in addition to a queue for the source traffic from the receiving node itself. The output of this detection process is called congestion index. QCCP-PS assumes that each sensor node has different priority, and depending on that the rate assignment to each traffic source, as well as its local traffic source, is based on its priority index and its congestion index. Therefore, the sending rate of each traffic source is increased or decreased depending on its congestion degree and its priority. Congestion notification and the new adjusted rates are sent implicitly by piggybacking the new rate value of each child node with the sending data of each sensor node. It has been shown that QCCP-PS has better performance than other protocols such as Priority based Congestion Control Protocol (PCCP) that uses priority index for decreasing rate only in case of congestion.

Another congestion control framework for WMSN is proposed in [47] to provide the necessary bandwidth and to alleviate the congestion problem to multimedia streaming. The proposed congestion control mechanism is implemented on the top of multipath routing facility. By exploiting this feature, the congestion control mechanism is based on load repartitioning over the multiple paths, instead of decreasing the transmission rate in case of congestion in order to maintain the quality of video streaming. The load repartition based congestion control (LRCC) uses queue length as congestion detection indicator along with collision rate. LRCC also uses explicit congestion notification by using especial control messages (called congestion notification messages). In reception of these notification messages, the source node will try to balance its traffic on the available paths while reducing the amount of data sent on the current congested path in order to reduce the congestion as well as maintain its sending rate unchanged.

2.4.2 Reliability and Loss Recovery

Loss recovery is another service that can be done by the transport protocols and aims to insure reliable packet delivery by retransmitting the lost packets. Loss recovery mechanism consists of three steps: loss detection, loss notification, and retransmission recovery, and these steps can be done either using end-to-end as in STCP [52] or hop-by-hop as in RMST [53] approach. In many applications of WMSN, hop-by-hop approach is preferred because it is more energy efficient (no need to send control messages or data packets over multiple hops), takes less response time, and requires less memory to cache packets for recovery. However, hop-by-hop loss recovery requires intermediate nodes to cache packets for future retransmission, and cannot guarantee packet delivery in case of node failure. Loss detection is either based on the sender by using timer or overhearing, or based on the receiver by using packet sequence number as in RSTP [54]. Loss notification is done explicitly by using acknowledgment messages or implicitly by overhearing success transmission from the next hop. Retransmission in WMSN is preferred to be based on hop-by-hop approach (it is also referred as link-by-link) where the packets can be cached at the intermediate nodes for less duration and for faster retransmission in case of packet loss. However, there is an overhead of using the limited buffer space at the intermediate node for caching packets for other nodes, as well as performing timely storage and flushing operations on the buffer. Therefore, this rises the need for in-network storage mechanism [55] or collaboration-based distributed cache points over the network [56]. Some mechanisms as in DWT-Reliable [57] and MMDR [58] provide reliability for multimedia streaming transmission in WMSN by exploiting the source video coding techniques by which the source traffic can be splitted into multiple streams. Each stream can be given different priority, depending on its resolution-level of the original content, and transferred over the network using priority-based routing or multipath routes.

The same concept of using multimedia processing to prioritize data packets for reliable transmission has been used in [59] where WMSN transparent protocol, called

Reliable Synchronous Transport Protocol (RSTP) is demonstrated. RSTP aims to reduce the large transmission delay of transferring multiple images between the source nodes and the sink due to transmission errors and limited bandwidth, and to provide ordered delivery. RSTP exploits the semantic of Progressive JPEG image stream to prioritize different parts of the stream and schedule the transmission based on the importance of the information. Progressive JPEG is used here rather than baseline JPEG because the first uses the coarse-to-clear data presentation mode which separates the DCT coefficients using multiple scans. By this way, the image can be divided into a number of scans that has smaller size than the original image and the low-quality (coarse) version of the image can be transmitted faster over WMSN. In RSTP, the less important parts within a data stream, which are the high frequency parts or the high quality version, would be discarded if there is not enough bandwidth. As a result, the limited bandwidth will be shared equally among the sensor nodes and the same transmission rate is maintained. Also the packets of images taken at earlier time would be scheduled first so that the temporal properties of the reference images are reserved. Once all the reference images reach the same quality level at the sink side, the image reconstruction process can start. For connection establishment and termination, RSTP uses similar technique of three-way handshake of the classic TCP protocol with the except that the receiver, not the sender, initiates the connection. For loss recovery, the receiver tracks the sequence number of the received packets and sends retransmission request if there is a gap in the sequence numbers or when the timeout alarms without receiving the last packet. RSTP incorporates TCP-ELN for the congestion control that uses transmission requests for packet losses caused by bit-errors and the classical TCP's congestion control mechanism for packet losses due to congestion.

A transmission scheme is proposed in [57] for reliable transmission of images in wireless multimedia sensor network, which is based on two-dimensional discrete wavelet image transform (2D DWT) and semi-reliable transmission to achieve energy conservation. DWT divides the image into separable sub-bands of multi-resolution representations. For example, the image can be decomposed into 4 sub-bands (LL, LH, HL, and HH) where the LL sub-band has the half size version of the input image and contains the low-pass information, and the others contain high-pass information. The LL sub-band can also be transformed again to have more levels of resolution. Then each sub-band is compressed to reduce its size by using entropy coding for lossless compression (Lempel-Ziv-Welch, LZW is used here) because lossless compression is less complex and require less computation. As a result, image data can be divided into multiple packets of different priorities where lowest resolution image data packet has the highest priority and should be reliably transmitted. Other packets can be handled with a semi-reliable transmission policy in order to save energy. Semi-reliable transmission enables priority-based packet discarding by intermediate nodes according to their battery's energy state. Packets of certain priorities are only forwarded by intermediate nodes if their battery energy level is above a given threshold.

Another framework that is based on multipath routing scheme is presented in [58], but this work considers offering reliability against bit-error from the wireless links only. The multipath multi-stream distributed reliability (MMDR) framework is proposed to reliably transport video content over WMSN by exploiting the features of multi-stream coding of video data and multipath routing. MMDR partitions the source encoded video data into multiple streams using one of the source coding techniques such as Layered Coding (LC), Multiple Description Coding (MDC), or Distributed Video Coding (DVC). Then Low Density Parity Check (LDPC) codes are used for channel coding the multiple video streams to compensate for the error prone wireless links. After that, MMDT will balance the video streams traffic over the available multipath route. Then at intermediate nodes, MMDR uses progressive error recovery algorithm (D-PERA) that tries to recover the bit errors by partially decoding the LDPC encoded packets.

2.4.3 Open Research Issues

From the above discussion, we can conclude that the existing transport protocols in sensor networks can be classified into two groups, as shown in Figure 2.2: Standard protocols (which include TCP, UDP, and their variants and modifications), and Non-standard or Application-specific WMSN transport protocols (which can focus on Congestion control only, Reliability only, or both Congestion control and Reliability).

In many real-time applications, if a standard transport protocol is to be used, UDP is preferred over TCP especially when timeliness is concerned than reliability. However, because of the unique characteristics of WMSNs that we mentioned in the beginning of this section, we believe that TCP with some modifications can better handle multimedia content over WMSNs. Some work is done in this direction such as Sensor Transmission Control Protocol (STCP) [52] that proposes some modifications in the TCP header fields in order to support video transmission and differentiated service model.

Most of the proposed application-specific transport protocols do not take into consideration the multimedia requirements in WMSN and none of them addresses its diverse concerns. This can be seen clearly in the performance evaluation conducted in [60], where it was shown that many of the proposed transport protocols cannot provide acceptable video transmission and do not support real-time communication in WMSN. Therefore, we believe that designing a transport protocol with appropriate performance metrics for both reliability and congestion control and based on the application layer source coding techniques will be a promising direction in this research area.

2.5 Application Layer in WMSN

The application layer in WMSN provides heterogeneous functionalities and supports many services, which include: 1) Multimedia processing and source coding techniques that depend on the application-specific requirements and capability of the hardware. 2) Effective communication with other application programs over the network to support collaboration in-network multimedia processing mechanisms. 3) Traffic Management and Admission Control.

2.5.1 Multimedia Source Coding Techniques

In order to have the ability to handle multimedia content over wireless sensor networks and to support real time multimedia applications, multimedia processing and source coding techniques have been widely used in the application layer of WMSN. Multimedia processing techniques aim to reduce the amount of multimedia traffic transferred over the network by extracting the useful information from the captured images and videos while in the same time maintaining the application-specific QoS requirements. However, in WMSN, these techniques should be designed in such a way that they meet current hardware capabilities, more power efficient to match the battery constrains in

WMSN, and have high compression efficiency to reduce the size of the multimedia content and to meet the available supported data rate and bandwidth in the network.

The existing video coding techniques that have been used in the literature for WMSN vary in their high compression efficiency, error resiliency, and low encoding-decoding complexity and can be classified into four groups:

- Layered Coding (LC) [61] is type of video source coding technique by which the original data is encoded to one important base layer (coarse version) and one or more less important successive enhancement layers (to get the fine version). At the destination side, the base layer can be combined again with all or subset of the higher-quality layers to achieve the desired level of video resolution. However, the loss of the base layer makes the information received from the the enhancement layers useless. The same principle of Layered Coding technique was used in [54] and [57].

A similar concept of LC was used in the work presented in [62] that proposes an image-pixel-position information based resource allocation scheme to transmit wavelet-based compressed images with best effort quality in WMSNs. Using wavelet for image compression will transform the image to coefficients **that** describe its information with different significant-level values that can be further compressed more easily to have at the end some large magnitude coefficients and many small magnitude coefficients with **“0”** bits. These small magnitude coefficients stand for the image-pixel-position information that is more important than the large magnitude coefficients that contains the image-pixel-value information (*i.e.* brightness). It is shown that the communication loss or bit errors in position information will have significantly higher effect on the overall quality of the received image than the loss or errors in value information. This is because the correct decoding of position data segment depends on the correct decoding of previous position data segments only, but the correct decoding of value data segment depends on previous bit-planes of both position data and value data segments. So, by allocating the scarce resources in WMSN on the position information more than the magnitude value information, position data segments are effectively protected to enhance image quality while value data segments are less protected to improve energy efficiency.

- Predictive Video Coding (PVC) [63], used in MPEG-x and H.26x standards, is based on the idea of reducing the bit rate generated by the source encoder by exploiting data statistics. PVC coding employs two modes for encoding the video: 1) Intra-frame coding mode (I-frame) that is used to reduce the redundancy within one frame by exploiting the spatial correlation in the frame, and 2) Inter-frame coding mode (P-frame) or motion compensated predictive that is used to reduce data redundancy in subsequent frames by exploiting both spatial and Temporal correlation. Performance evaluation of PVC over Stargate and TelosB is conducted in [64] showing the energy consumption in both video compression and transmission.

- Multiple Description Coding (MDC) [65] is used to enhance the error resiliency of video delivery by splitting the multimedia content to two or more independent and equal important streams (multiple descriptions). Each description alone provides acceptable low quality version of the original and combining all descriptions together gives higher resolution. This technique can be used in conjunction with multi path transport approach to achieve load balancing and meet the available bandwidth as shown in [33] and [32].

- Distributed Video Coding (DVC) [66], used for low complexity encoding by shifting the complexity to the sink side, incorporates concepts from source coding with decoder side information for creating an Intra-coded frame along with a side information frame. Therefore, in this technique, multimedia content can be partitioned into multiple streams consisting both intra-coded and decoder side information frames by using simple and low power encoder while the decoder at the destination side can be complex exploiting the availability of the resource such as energy and processing power capability. Two practical DVC encoders are proposed in the literature, Wyner-Ziv (WZ) [67] and PRIZM [68]. DVC coding has been used for WMSN in [69] and [64]. It is shown in [70] through practical implementation of DVC in WMSN that there is a tradeoff between computation and transmission power consumption depending on the encoding schemes used in implementing DVC codec. While a computational intensive scheme, such as discrete cosine transform (DCT), consumes more computational power, it achieves significant compression hence less needed transmission power. On the other hand a less computational intensive scheme, such as a pixel based codec needs less computational power but more transmission power. Therefore, the choice of either scheme (DCT or pixel based) to implement the DVC codec for multimedia content in WMSN depends on the tolerable distortion (quality) and power consumption.

As another methodology for real time video compression and communication over WMSN, Address-Event Representation (AER) is demonstrated in [71]. AER overcomes the limitations of video transmitting over the network in terms of data rate and bandwidth. It is shown in [71] that for streaming video of size 320x240 differencing, Crossbow MICAz will be hardly able to deliver the video content at about 2fps at 250 Kbps data rate which it will be hard to understand for the end-user and consume all the communication capabilities of the sensor nodes. Therefore, the proposed algorithm in [71] encodes the video using AE representation, by using custom image sensors capable of detecting intensity-differencing information, and perform a zero-computation compression of the frame-difference video. This compression technique enables sensor nodes to stream temporal frame-difference video over the network at high rates where frame-differencing video can be obtained by subtracting each pixel intensity value in the previous frame from the corresponding pixel intensity value of the current frame. By this way, each pixel value now in the processed video can be presented by only 2 bits (comparing to grayscale video that has a size of 8 bits for each pixel) requiring less bandwidth and also preserving the privacy of the users due to the reduction of intensity information, but it still reveals the motion information. Then frame-difference video can be further compressed by using Address-Event Representation (AER), where events are signaled when changes in pixel intensity reach a certain threshold. In a frame-difference AER image, the pixels that experience large intensity changes will generate events first and more frequently than other events and thus it will be available at receiver side immediately. In order to further compress AER video, it is only required to read less events outputted by the AER algorithm without any computation. It is shown that compression of a 320x240-pixel frame-difference images using AER is resulting in 50 times less bandwidth requirement than non-AER scanning video. It is also shown that this technique is only useful with sparse images that have limited number of events (less than 9000 events) otherwise it will become less efficient than other scanning image techniques.

In [72], eight popular image compression algorithms are reviewed and compared with each other to find the most suitable image compression algorithm for implementation in WMSNs and can meet its requirements including a fast and efficient image processing capability, low memory requirement, high compression quality, less complex system, and low computational load. Image compression process, removing the highly correlated redundant information within the image, contains basically two stages: image transformation stage and entropy coding stage. According to [72], image coding process are categorized into: first generation, which focuses more on how the information contained in the transformed image can be efficiently encoded (such as Discrete cosine transform (DCT), Embedded zerotree wavelet (EZW), Set-partitioning in hierarchical trees (SPIHT), Embedded block coding with optimized truncation EBCOT), and second generation, which put more importance on how the useful information can be extracted and exploited (such as pyramidal coding, Directional decomposition, segmentation based coding, vector quantization). The second generation image compression requires more complex and extensive image processing compared to the first generation image coding as shown in [72]. Among the first generation algorithms, it is shown that SPIHT wavelet-based image compression is the most suitable hardware implemented image compression algorithm for WMSN because it is less complex and needs less computational load and memory allocation.

2.5.2 Collaboration In-network Multimedia Processing

The use of densely deployed sensor nodes in the wireless sensor network provides an inherent protection against normal and provoked system faults. The redundancy of information gathered by neighboring nodes can be exploited for more accurate and robust observation results through effective data fusion and aggregation. In WMSN, the redundancy of information can also be found as overlapping of FoVs of camera sensors located in the same cluster. In addition, the redundancy can be exploited for networking to avoid single-points of communication failure. Therefore, it is necessary to develop efficient and distributed filtering and in-network cooperative processing mechanisms to enable real time retrieval of useful information. For example, in data aggregation, two cameras may collectively fuse their information to obtain a lower bandwidth aggregated result, which is then routed (and possibly fused with other sensor node readings along the way or within the cluster) to the cluster head or the sink.

A distributed filtering architecture is presented in [73] for authoring wide-area sensor-enriched services that supports scalable data collection from high bit-rate multimedia sensors by greatly reducing the bandwidth demands. This architecture (Internet-scale Resource-Intensive Sensor Network, IrisNet) enables the use of application-specific filtering of sensor data near their sources and provides interfaces that simplify collecting and manipulating these data. Also it reduces the processing and bandwidth requirements by detecting repeated computations among the services and eliminating as much of the redundancy as possible. In order to reduce the bandwidth consumed, IrisNet uses distributed filtering in which each service processes its desired sensor feeds on the CPU of the sensor nodes where the data are gathered instead of transferring the raw data across the network. In addition, in order to reduce the computation demands of this approach, IrisNet includes a mechanism for sharing results between sensing services running on the same node.

A collaborative hybrid classifier learning algorithm is proposed in [74] to achieve online vector machine learning for target classification in WMSN. The collaborative hybrid learning algorithm uses progressive distributed computing paradigm for in-network multimedia processing in each cluster, and peer-to-peer paradigm between the cluster heads. The authors show that this algorithm overcomes the disadvantages of using centralized learning paradigm or distributed mobile agent learning paradigm, as consuming too much energy and bandwidth as the case of centralized learning paradigm, and imbalance energy consumption and the need for centralized processing center as the case of distributed mobile agent learning paradigm.

An Artificial Immune System (AIS) based recognition scheme is presented in [75]. The proposed scheme can be used to construct high resolution images from multiple low resolution images captured by multiple sensor nodes in order to improve pattern recognition success rate and image communication energy efficiency. Principal Component Analysis (PCA) is used here for image dimension reduction that extracts only its major or principal components in order to be transmitted to the base station for pattern recognition. In order to reduce the information redundancy and extra number of captured image frames for energy efficient image processing and communication, a sleep/awake schedule algorithm is performed. A node's sleep/awake status depends on the minimum needed number of image frames to be acquired by the sensor node for successful pattern detection and on the residual energy of the sensor node within a cluster group.

In [76], A compacted probabilistic binary visual classification for human targets in WMSN is presented, where a Gaussian process classifier (GPC) is used for classifier learning instead of support vector machine (SVM). Although SVM has outstanding performance in target classification, SVM causes overhead cost from using parametric optimization algorithm and restricts its application area by getting only the transformed distance between samples and hyper-plane rather than classification probability as in the case of GPC. Also in order to decrease the transmission energy and communication overhead, the GPC classifiers are trained in processing center -single node data processing- and transferred to all sensor nodes rather than using distributed processing. In order to improve the performance of GPC classifier and decrease computing complexity, the raw data -after background subtraction- is refined and the dimension of feature data is compacted by using integer lifting wavelet transform (ILWT) and rough set (RS). And in order to increase robustness and accuracy, committee decision fusion is used to combine individual decisions from different nodes with dynamic weight selection depends on the number of correct classifications of a sensor node and its current energy status.

2.5.3 Traffic Management and Admission Control

The application layer of WMSN also supports, in addition to multimedia processing source coding techniques and in-network multimedia processing, network traffic management and admission control functionalities which are directly related to application-specific QoS requirements. Therefore, based on the traffic class of the application, WMSN needs to provide a differentiated service.

2.6 Cross Layer Optimization

In the previous sections, we discussed the existing proposed solutions designed for WMSNs that follow the classical layered structure of the communication protocol stack. Some of these proposals may achieve a good performance in terms of some metrics related to each of their intended individual layers, but these performance metrics are not jointly optimized to maximize the overall network performance with minimum energy consumption. Moreover, most of these existing solutions do not provide enough support for multimedia applications since the work is done at the lower layers of the communication stack for optimizing functionalities such as optimal routing, reliable delivery, efficient resource management, and other tasks without taking into consideration the especial requirements of handling multimedia content over WMSNs. For example, multimedia compression and source coding algorithms at the application layer should be considered when designing the functionalities at the lower layers and vice versa:

- Cross layer design between multimedia source coding techniques at the application layer and the routing protocol in the routing layer can be exploited for better multipath selection or in-network processing.

- Cross layer design between the routing layer and the MAC layer can allow for packet-level service differentiation or priority-based scheduling and for more power efficient routing mechanisms.

- Cross layer design between MAC layer and the physical layer, especially in the case of using UWB technology. The adoption of the UWB technology as the underlying transmission technique in WMSNs and the potential challenges in this area, appear as an interesting research topic.

- Cross layer design between the routing layer and transport layer especially in the case of multiple paths routing for optimizing the selection of better or most adequate paths that guarantee the required QoS and reliable delivery for each type of multimedia content.

Therefore, in WSNs in general and in WMSNs in particular as they are considered resource-constraint environments, we believe that the correlation characteristics and functionality interdependencies among the layers of the communication stack cannot be neglected and should be exploited for better performance and efficient communication, consequently, cross layer design stands as the most promising alternative to

inefficient traditional layered protocol architectures. Recent works in WMSNs show that cross-layer integration and design techniques result in significant improvement in terms of quality of service, energy conservation, and exchanging information between different layers of the communication stack:
A cross layer communication architecture is presented in [20] to provide QoS for WMSN applications based on the Time-Hopping Impulse Radio Ultra-Wide-Band (TH-IR-UWB) transmission technique. This architecture tries to solve the short-comes of using CSMA/CA for the MAC layer in WSN such as the variable and uncontrollable access delays from using random timer, idle listening from using carrier sense, and increased energy consumption occurs due to for example hidden node problem and to provide QoS for WMSN. The proposed system guarantees the end-to-end QoS requirements to handle multimedia content and packet level service differentiation at the network layer in terms of throughput, end-to-end packet error rate and delay by the joint hop-by-hop local decisions of the participating nodes. This is done by an admission control protocol where the sender node that wants to establish connection sends request packet describes its requirements to its neighbors, and among the replies the sender node selects the one who has the most positive advance toward the sink and able to satisfy the needed requirements and this continue iteratively until the end-to-end path is established to the sink. The cross-layer system also provides receiver-centric scheduling based on time hopping sequence of impulse radio of Ultra-Wide-Band MAC and physical layer that allows multiple parallel transmissions, prevents collisions at the receiver node (by using unique TH sequence for each receiver) and saves energy by avoiding idle listening and wasteful transmissions (by turning on exactly on the incoming transmission). This is done by scheduling scheme where each sensor node is responsible to schedule the data packets from its children nodes (closer to sink has higher priority to schedule first) by sending scheduling packets contain different time hopping sequence for each one of them, and also dynamic channel coding to adapt to interference.

Another cross layer design for WMSNs is implemented in [77] where an extension of a routing protocol is presented along with path priority scheduling algorithm for efficient communication of real-time video over WMSNs. The proposed routing protocol is an extension of DGR, Directed Geographical Routing, for constructing multiple disjoint paths in order to enlarge the aggregate bandwidth, facilitate load balancing, and guarantee packet delivery. Using hop-by-hop deviation angle adjustment method, a path can be established using any initial deviation angle specified at the source node, and then other disjoint paths are constructed by changing the value of the deviation angle. To meet the delay constraint of video frames, a path priority scheduling algorithm is used that specifies the number of paths used and assign video sub-streams according to the status of the paths. Using this scheduling algorithm, the weight of each path will be calculated based on the estimated available bandwidth, path delay, and path energy level. Then, by using path weight along with packet priority, shorter delay paths will be used for time-constrained packets while other paths are used for balancing energy and bandwidth usage for other traffic. In case the required bandwidth is larger than the aggregate bandwidth, least priority packets will be dropped. But if the available bandwidth is still not enough, intermediate nodes should decide whether to forward these packets or drop them if they cannot meet their deadline and inform the source node to use either multiple frame selecting or intra-frame refreshing.

In [78], a context-aware cross-layer optimized multi-path multi-priority (MPMP) transmission scheme is proposed in which a multipath routing is used in the routing layer in conjunction with a context-aware multipath selection algorithm in the transport layer. TPGF, Two-Phase geographic Greedy Forwarding, routing that is described in the Routing Section is used to explore maximum number of node-disjoint routing paths while CAMS, Context-Aware Multipath Selection algorithm, is used to choose the maximum number of paths from all found node-disjoint paths for maximizing the delivery of the important data to the sink and guaranteeing end-to-end transmission delay. CAMS algorithm selects the proper routing paths that are suitable for each type of multimedia content based on two types of priority: end-to-end transmission delay based priority for constraint real-time video communication, and context-aware multimedia based priority (image vs. audio) that depends on the importance of multimedia stream which can reflect precisely the event in WMSN.

In [79], a cross layer design is considered for data gathering in WMSN where an adaptive scheme called (RRA) is used to dynamically adjust the "transmission Radius and data generation Rate Adjustment". Based on the result that the transmission radius of sensor nodes and the data generation rate of sensor nodes are two critical factors in affecting the data gathering performance, the proposed scheme aims to address this research challenge by taking into account the interaction among physical layer, routing layer, and transport layer. RRA scheme first minimizes the end-to-end transmission delay in the network while using minimum data generation rate, and then an optimal transmission radius is calculated in this phase. Then by using this derived transmission radius, the data generation rate can be adjusted to increase the amount of gathered data. Therefore, the cross layer framework of RRA scheme can be summarized into four steps: 1) choosing the optimal transmission radius for sensor nodes at the physical layer, 2) constructing multiple routing paths by using multipath routing protocol like TPGF in the routing layer, 3) selecting the suitable paths from the found paths by the routing protocol at the transport layer, and 4) adjusting the data generation rate of source nodes in the physical layer. Another cross layer design combining network and MAC layers for QoS enhancement in WMSNs is presented in [80]. At network layer, the proposed framework is aiming to find near-optimal paths satisfying the application-specific QoS requirements based on using Multi-Objective Genetic Algorithm (MOGA), while at MAC layer the routing information is used in the MAC algorithm, which is based on CSMA/CA, for QoS-based packet classification and automatic adaptation of the contention window.

2.7 Coverage and Connectivity

Coverage problem becomes critical in WMSN because the deployed multimedia sensors do not have omni-directional (antenna) coverage as the case of the scalar sensors, but they have the feature of capturing direction-sensitive multimedia content with larger sensing radii when there is a line of sight (LOS). Therefore, many proposed algorithms try to exchange local information between neighboring multimedia nodes to determine the most beneficial orientations of their coverage taking into account minimizing the effects of occlusions and overlapping sensing regions and improving the cumulative quality of sensed information from the region of interest. Knowing the overlapping areas between cameras allows exploiting the redundancy in camera sensor coverage in the network and also can be used to track moving objects in the environment.

In [81], a distributed algorithm is proposed to detect the coverage of multimedia nodes and determines their orientation. The algorithm assumes that each multimedia node knows its location information and the location of all the obstacles around it. It starts with broadcasting HELLO messages between the neighbors to exchange the location information of each other and current FoV of each multimedia sensor. By using these information, each multimedia sensor detects the best pose for its field of view using one of the three tests. In Perimeter Test, each multimedia node, with the ability of panning 360^o to scan its FoV disk, determines whether a visible FoV (full coverage without occluded or overlapping regions) exists in its FoV disk otherwise the node runs Neighbor-Distance Test. The node can know the overlapping regions by using the received location information from the neighbors. Neighbor-Distance Test examines whether a node has a visible FoV that might be overlapped with other neighbor, but the node needs to find the smallest overlapping FoV by scanning the FoV disk. Finally, if a node could not find a visible FoV even overlapped with other neighbor, the node runs the Obstacle-Distance test to avoid the occlusions from closer obstacles and maximize the visible FoV.

An automated calibration protocol, called Snapshot, is presented in [82]. Snapshot determines and calibrates the location, orientation, and range of camera sensor in WMSNs using the inherent imaging abilities of the cameras themselves and four reference points along with using the principles from optics and geometry in the calibration process. Snapshot uses low-fidelity camera sensors (such as CMUcam or Cyclops) connected with limited computational recourse processors (Crossbow motes or Intel Stargates) and some of the sensors are equipped with a Cricket ultrasound receiver as a wireless calibration device for the reference point. This technique assumes that the intrinsic parameters (focal length, lens distortion, principal point) are known from the manufacture or can be estimated offline prior to deployment, and it only tries to determine the extrinsic parameters (coordinates of the camera, camera orientation, and the FoV of each camera) in order to use as less reference points as possible and reduce the computational cost. In order to determine the location of a camera sensor, the four reference points should be in the visual range of the camera and should not three of them lie along a straight line. By using two reference points and the principles of geometry, the location of the camera lies in closed surface. And by using combination of two reference points each time with the rest of the four points and taking the intersection points of the resulting surface from each calculation with the other, the location of the camera can be estimated after eliminating the false solutions using the optics principles. The orientation of the camera in the three axes (pan, tilt, and rotate) is also determined by using three reference points and the camera location from the previous process. For estimation the visual range of the camera and overlap between other cameras, Snapshot assumes that the size of the internal camera CMOS sensor and the focal length are known offline, and from projection the coordinates of the corners of the CMOS sensor through the camera lens (its location determined from the previous process), the coordinates of a polyhedron or pyramid can be determined using the geometry principles. An object in the volume of the polyhedron is in the visual range of the camera. In order to find the overlapping area between the other cameras, an intersection between any edges of the calculated polyhedron with any other polyhedrons from other cameras is considered as overlapping area. It is shown that this calibration technique gives an error of 1-2.5 degrees in estimating the camera orientation and 5-10 cm error in determining the camera location.

In [83], a scheduling method based on cooperation among nodes for object detection in WMSNs is proposed where the network is clustered based on the overlapping coverage areas of the multimedia sensor nodes. To calculate the overlapping area, the proposed algorithm computes the vertex points of the FoV of each multimedia node by assuming the FoV of a node as an isosceles triangle. A multimedia node's location represents one of the three vertexes of the FoV and the other two vertexes will be found by the algorithm by knowing the coverage direction angel, orientation angel, and the sensing range of each node. Then the overlapping area is calculated using a decomposition method for intersection polygons. A cluster consists of a subset of multimedia nodes with high overlapping FoV areas where the size of the overlapping area between FoVs of two nodes determines whether they can be in the same cluster. In each cluster, nodes are scheduled to sequentially wake up, capture an image and monitor presence of object, applying object detection procedures such as background subtraction or motion detection and finally go to sleep.

2.8 Security in WMSN

Many applications of WMSN have their additional and special requirements in terms of privacy and security, such as military applications, medical care applications, and other video surveillance systems. In addition to the fact that sensor networks are vulnerable to attacks more easily than the wired networks because of their use of a broadcast medium. Therefore, in order to guarantee message authenticity, integrity, and confidentiality, security in WMSN should be taken into account at the design phase of the network, while at the same time maintaining the efficiency and scalability of the network.

In order to assure authentication and data integrity for multimedia data delivered over WMSN, an energy-aware adaptive wavelet-based watermarking technique for real time image delivery is presented in [84]. This technique embeds additional data called a watermark into some location in an image object so it can be detected later to make an affirmation about the object. The watermarking locations or positions are adaptively chosen by using two thresholds to insert the watermark according to network conditions so that the energy efficiency and security can be achieved. In order to degrade the effect of the distortion of watermarked image, the proposed scheme embeds the watermark into few positions as possible to make it invisible and allocates extra network resources to protect this embedded watermark from high distortion so it can be detectable. In addition, it also embeds watermark coding redundancies into the original image so the watermark becomes more robust to packet loss. The frequency-based (middle band) discrete wavelet transform (DWT) has been selected because it is more robust, easy to recognizable and authenticated at the receiver side, and it reduces computation complexity and process delay by exploiting the correlation of the inter-frames.

In [85], a privacy paradigm called HoLiSTiC is proposed that secures routing and topology information for WMSNs against outsider attacks. The paper assumes a clustered network with some nodes equipped with free-space optical (FSO) capabilities. Also it assumes that the BS and CHs have bidirectional communication links and the camera and transport nodes have unidirectional links. The proposed protocol requires that each node to have an individual key shared with the BS and pairwise keys shared between adjacent visual nodes in a cluster. In addition, every network entity has two pre-deployed network-wide keys. All keys are employed for symmetric cryptography to provide a variety of security services.

A secure data converter architecture is proposed in [86] for WMSN that employs fingerprinting and encryption capabilities for simultaneously digitize and authenticate sensor readings. The proposed architecture suggested hardware modifications to the data converter and aims to reduce the computational complexity of the security algorithms at the aggregation points in the systems that need in-network processing. This can be done by embedding an authenticator payload into the data converter or the modulator output in a way that it is not easily extractable without access to the secret key, and can be used to verify the integrity of the sensor reading.

New trends in security schemes for WMSNs seem to point to energy-aware and lighter-weight security schemes than for traditional networks. Nevertheless, their nodes can use more processor-consuming algorithms that the ones that would be suitable for Wireless Scalar Sensor Networks (WSSNs). This is so, because at least the nodes that have to process multimedia will have more processing capabilities than the typical scalar node.

We foresee that for WMSNs symmetric cryptography will be the chosen approach - over asymmetric cryptography- since its lighter processing requirements since it makes a lot easier to solve several security problems related to eavesdropping and compromised nodes.

Watermarking has also been proposed as a way to provide data integrity. Nevertheless, watermarking solutions might be vulnerable to attacks from entities that know how the watermarks are done. In addition, watermarking alone does not solve data authentication. Therefore, we do not foresee watermarking solutions becoming a mainstream approach, but a marginal solution for very specific problems.

Moreover, in order to preserve battery and to save bandwidth, many WMSNs will use some sort of data aggregation. We consider that security and aggregation schemes cannot be devised separately. Therefore, new security schemes will have to be both energy-aware and designed in together with the aggregation scheme.

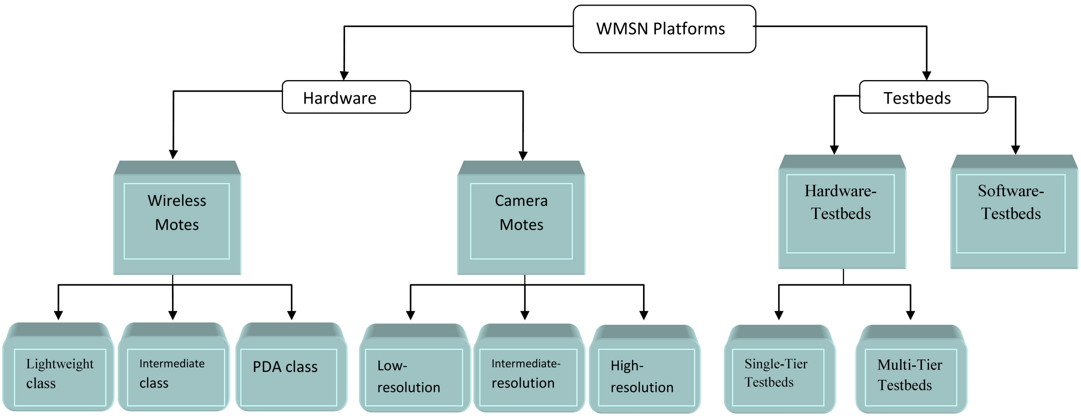


Figure 2.3: WMSN Platforms Classification

2.9 Hardware and Testbeds

In order to have the capability of handling multimedia applications in WMSN, the ability to support their requirements and challenges, and to examine and test the proposed protocols and algorithms developed for WMSN, the underlying enabling technology and platforms are required to be more efficient and cover the drawbacks of the existing hardware designed for WSN for detecting scalar events. Therefore, many works have been presented in the literature to modify the existing platform (hardware) or present new hardware implementation and testbeds. These proposed platform and testbeds are more powerful and have more potential to process and handle multimedia traffic efficiently in terms of processing power, memory, data rate, power consumption, and communication capabilities. The work in [87] described the different applications of WMSN and some of the devices and testbeds used in WMSN, but in this section, we introduce most currently off-the-shelf hardware as well as available research prototypes and show their specifications and performance comparing to each other. In addition, we categorize the existing platforms and research prototypes according to their capabilities and functionalities in WMSN as shown in Figure 2.3 below.

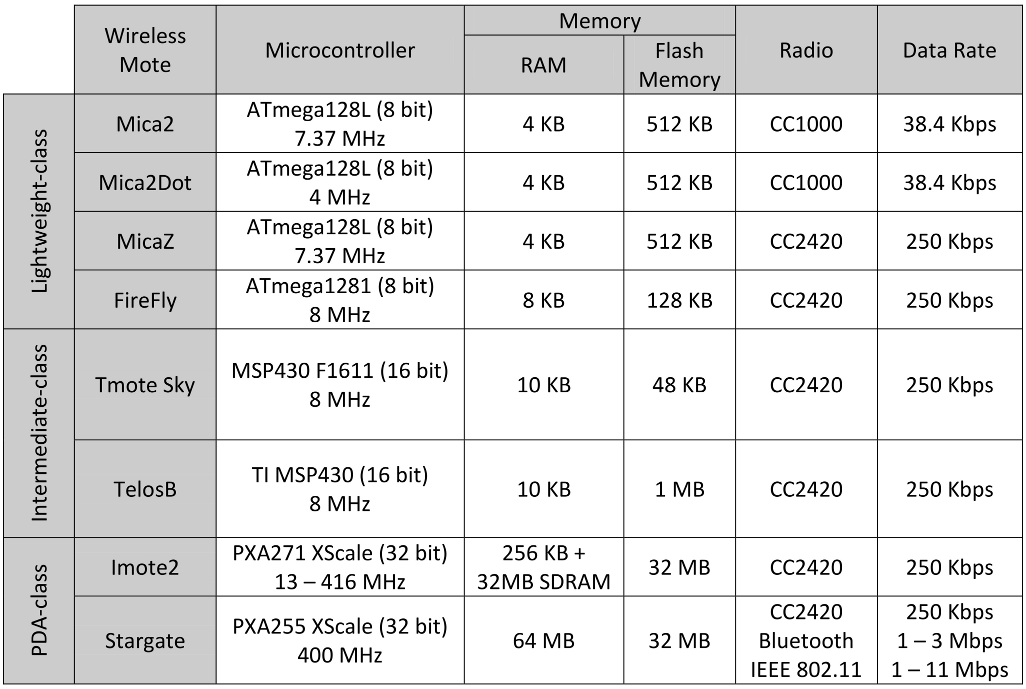
2.9.1 Wireless Motes

There are several devices of wireless motes that can be used as WMSN motes and most of them are available in commercial products as shown in Figure 2.4. Depending on their processing power and storage capacity, these wireless motes can be classified into three groups:

1) Lightweight-class Platforms: The devices in this category are designed initially for detecting scalar data, such as temperature, light, humidity ... etc, and their main concern is to consume less amount of energy as possible. Therefore, these devices have low processing power capability and small storage and most of them are equipped with a basic communication chipset (e.g. IEEE 802.15.4 on CC2420 radio). The CC2420 chipset only consumes 17.4 and 19.7 mA for sending and receiving respectively and has maximum transmit power of 0 dBm with data rate of 250 Kbps. Table 2.4 shows examples of lightweight-class wireless motes, Mica-family motes [88] and FireFly [89], and compares their specifications.

2) Intermediate-class Platforms: The devices in this group have better computational and processing capabilities and larger storage memory than lightweight-class devices. However, they are also equipped with low bandwidth and data rate communication module (e.g. CC2420 chipset which is IEEE 802.15.4 compatible). Tmote Sky [90] is an example of Intermediate-class mote designed by Moteiv (Sentilla) that uses low power 8 MHz 16-bit MSP430 F1611 RISC processor from Texas Instruments featuring 10kB of RAM, and 48kB of flash. Tmote Sky uses Chipcon CC2420 radio for IEEE 802.15.4/ Zigbee for maximum data rate of 250 Kbps. Tmote Sky has been used to implement camera mote with CITRIC [91] and CMUCam3 [92].

3) PDA-class Platforms: The devices in this category are more powerful in terms of computational and processing power and they are designed to process multimedia content in a fast and efficient manner. These devices can run different operating systems (e.g. Linux, TinyOs, and run Java applications and .NET micro frameworks) and support multiple radios with different data rates (e.g. IEEE 802.15.4, IEEE 802.11, and Bluetooth). However, these devices consume relatively more energy. Stargate and



	Wireless Mote	Microcontroller	Memory		Radio	Data Rate
			RAM	Flash Memory		
Lightweight-class	Mica2	ATmega128L (8 bit) 7.37 MHz	4 KB	512 KB	CC1000	38.4 Kbps
	Mica2Dot	ATmega128L (8 bit) 4 MHz	4 KB	512 KB	CC1000	38.4 Kbps
	MicaZ	ATmega128L (8 bit) 7.37 MHz	4 KB	512 KB	CC2420	250 Kbps
	FireFly	ATmega1281 (8 bit) 8 MHz	8 KB	128 KB	CC2420	250 Kbps
Intermediate-class	Tmote Sky	MSP430 F1611 (16 bit) 8 MHz	10 KB	48 KB	CC2420	250 Kbps
	TelosB	TI MSP430 (16 bit) 8 MHz	10 KB	1 MB	CC2420	250 Kbps
PDA-class	Imote2	PXA271 XScale (32 bit) 13 – 416 MHz	256 KB + 32MB SDRAM	32 MB	CC2420	250 Kbps
	Stargate	PXA255 XScale (32 bit) 400 MHz	64 MB	32 MB	CC2420 Bluetooth IEEE 802.11	250 Kbps 1 – 3 Mbps 1 – 11 Mbps

Table 2.4: Comparison of the Features of Wireless Motes

Imote2 are examples of PDA-class platforms. Stargate board [93], designed by Intel and manufactured by Crossbow, uses 400 MHz 32-bit Marvell's PXA255 XScale RISC processor with 32 MB of Flash memory and 64 MB of SDRAM and runs Linux operating system. It can be interfaced with Crossbow's MICA2 or MICAz motes for IEEE 802.15.4 wireless communication as well as PCMCIA IEEE 802.11 wireless cards or compact Flash Bluetooth. Thus, Stargate board can be used as a sensor network gateway, robotics controller card, or distributed computing platform. It forms a camera mote when it is connected with camera device (e.g. webcam) as shown in [94] [95] [96]. Imote2 [97], also designed by Intel and manufactured by Crossbow, is a wireless sensor node platform built around the low-power 32-bit PXA271 XScale processor and integrates an 802.15.4 radio (CC2420) with a built-in 2.4GHz antenna. It can operate in the range 13-416 MHz with dynamic voltage scaling and includes 256 KB SRAM, 32 MB Flash memory, 32 MB SDRAM, and several I/O options. It can run different operating systems such as TinyOs and Linux with Java applications and it is also available with .NET micro framework. It integrates many I/O options making it extremely flexible in supporting different sensors including cameras, A/Ds, radios,

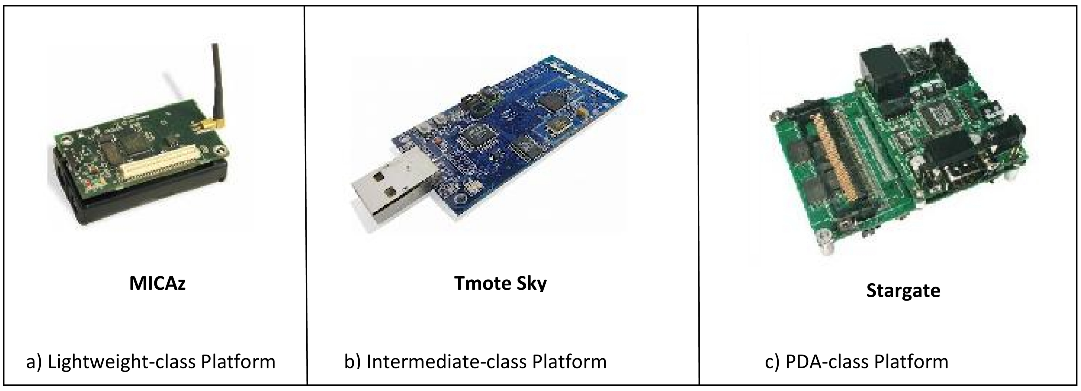


Figure 2.4: Examples of Wireless Mote Platforms

etc. The PXA271 processor includes a wireless MMX coprocessor to accelerate multimedia operations and add media processor instructions to support alignment and video operations. Imote2 has been used as a camera mote in [98] [99].

2.9.2 Camera Motes

In order to reduce the amount of resources required by transmitting multimedia traffic (images, videos) over WMSN, the multimedia content should be intelligently manipulated and processed using appropriate compression and coding algorithms along with other application-specific multimedia processing such as background subtraction, feature extraction, etc. However, most of these algorithms are complex and require high computational and processing power as well as larger memory for buffering frames. Sometimes, these requirements cannot be satisfied with the only resources offered by the wireless motes, which we mentioned before, especially if they require floating-point operations for efficient multimedia processing. Therefore, camera sensor may coupled with additional processor (microcontrollers, DSPs, FPGAs, etc) and memory resources before relaying the processed data to the wireless mote for wireless communication. Nevertheless, the additional processor and memory resources require more energy consumption and cost and this makes a tradeoff between energy consumption and cost on one side with computational power and traffic amount on the other side. It has been shown in [100] that the time needed to perform relatively complex operations on a 4 MHz 8-bit processor such as the ATmega128 is 16 times higher than the time needed with a 48 MHz 32-bit ARM7 device, while the power consumption of the 32-bit processor is only six times higher. Hence, this indicates that the powerful processor (such as 32-bit ARM7 architecture) is more power-efficient in multimedia applications. Table 2.5 shows the existing multimedia platforms and research prototypes for WMSN and compares between their specifications.

From Table 2.5, we can conclude that camera motes have different capabilities (resolution, processing power, storage, and others) and accordingly, depending on their capabilities and features, they have different functionalities and play different rules in the network. For example, low resolution cameras can be used at the lower-tier of multi-tier network for simple object detection task to exploit their low-power consumption feature that allows them to be turned on most of the time (or in duty cycle manner). Cyclops, CMUCam3, and eCam [101] are examples of *low-resolution* cameras. Intermediate and high resolution cameras can be used at higher-tiers of the network for more complex and power-consuming tasks, such as object recognition and tracking. These types of cameras consume more power and hence there are only woken up on-demand by lower-tier devices, e.g. in case detecting an object of interest. Webcams, attached for example with Stargate board or Imote2, can be considered as *intermediate-resolution* cameras, while PTZ cameras used in [94] is an example of *high-resolution* camera. Figure 2.5 shows commercial product examples of camera mote platforms used in WMSNs.

Cyclops [102] is a small camera device developed for WMSN. Cyclops is compatible with the computationally constrained wireless sensor nodes (motes) and exploits the characteristics of CMOS camera sensors as they are low power, low cost, and small size sensors. Cyclops platform isolates the requirement of camera module for high speed data transfer from the low-speed capability of the embedded controller and provides still images at low rates. It is designed to be interfaced with the common motes used in wireless sensor networks such as MICA2 and MICAz. Cyclops hardware architecture consists of an imager (Agilent compact CIF CMOS ADCM-1700), an 8-bit RISC ATMEL ATmega128L micro-controller (MCU), a Xilinx XC2C256 CoolRunner complex programmable logic device (CPLD), an external 64KB SPRAM, and an external 512KB Flash. The MCU controls Cyclops to capture images and communicate with host to provide image interface, while CPLD provides high speed clock, synchronization, and memory control required by the image capturing that cannot be satisfied by using a lightweight processor. So, CPLD acts as a lightweight frame grabber to provide on-demand access to high speed clocking at capture time and perform a limited amount of image processing such as background subtraction or frame differentiation. Cyclops firmware is written in nesC language and runs under TinyOS operating system. In addition to the libraries provided by TinyOS, Cyclops also provides primitive structural libraries (such as matrix operation libraries or histogram libraries) and advanced or high-level algorithms libraries (such as coordinate conversion and background subtraction). The authors show in the performance analysis that Cyclops is a low power device and its energy consumption depends on the power consumption of different states (such as image capturing, memory access, micro-controller processing, sleep ... etc) and their time duration as well as on the input image size and the ambient light intensity.

FireFly Mosaic, a vision-enabled wireless sensor platform and image processing framework presented in [103], uses camera motes consisting of FireFly wireless node coupled with a CMUcam3 camera sensor. The FireFly nodes run the Nano-RK real-time operating system and communicate wirelessly using the RT-link collision-free TDMA-based protocol. FireFly Mosaic is designed to be low-cost, energy efficient, and scalable compared to the centralized wireless webcam-based solution. The used RT-link TDMA-based link wireless communication provides tight global time synchronization to prevent collisions and save energy while Nano-RK operating system provides hooks for globally synchronized task processing and camera frame capturing. While the network communication relies on TDMA-based link layer, the internal communication between the camera and the wireless node is based on the Serial Line IP (SLIP). The CMUcam3 camera of FireFly Mosaic consists of CMOS (OmniVision OV6620) camera ship capable of capturing fifty 352x288 color images per second, frame buffer (Averlogic AL440b FIFO), and 23-bit (LPC2106 ARM7TDMI) micro-controller running at 60MHz with built-in 64KB RAM and 128KB Flash memory. Also CMUcam3 has four on-chip servo controller outputs which can be used to actuate a pan-tilt device. In the other hand, FireFly sensor node has a low-power ATMEL

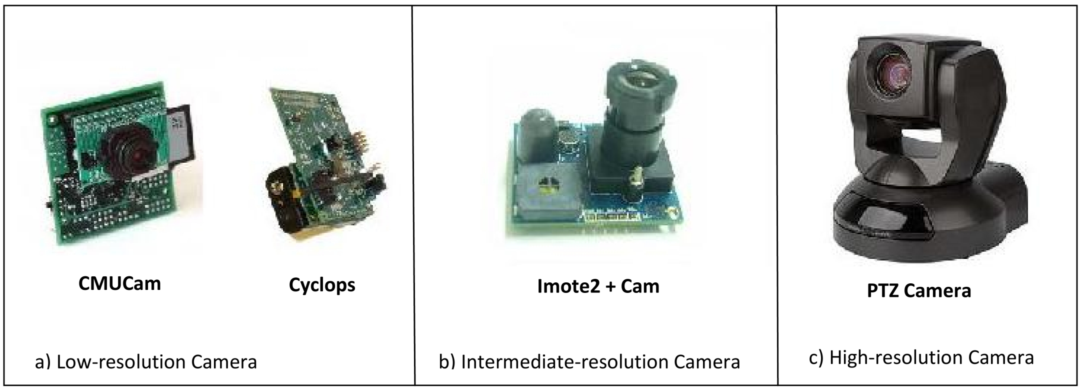
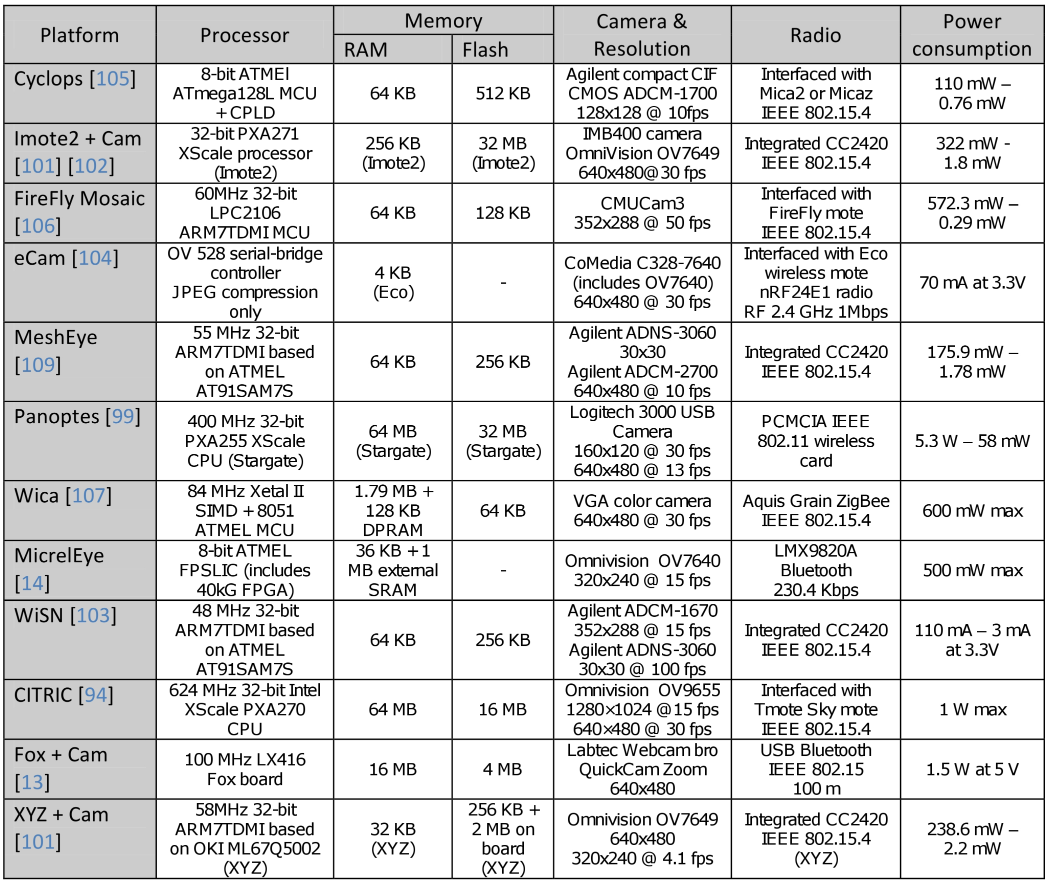


Figure 2.5: Examples of Camera Mote Platforms

ATmega 1281 8-bit processor with 8KB RAM and 128KB Flash memory, connected with Chipcon CC2420 802.15.4 radio capable of transmitting a 250Kbps for up to 100 meters. CMUcam3 is an open-source camera comes with several libraries (named CC3) and example applications such as JPEG compression, frame differencing, color tracking, convolutions, edge detection, connected components analysis, and a face detector. This several image processing algorithms can be run at the source and only the results may be sent over the multi-hop wireless channel to the FireFly gateway. CMUcam3 can be also interfaced with other type of sensor nodes such as TolesB and Tmote Sky motes running different operating systems.

Wica [104] is another camera mote designed for wireless multimedia sensor network. The wireless camera mote is based on an SIMD (Single Instruction Multiple Data) video-analysis processor and an 8051 micro-controller as a local host, and it is using the IEEE802.15.4 standard (ZigBee) for its wireless communication. The camera consists basically of four components: one or two VGA color image sensors, an SIMD processor for low-level image processing, a general purpose processor for intermediate and high-level processing, and control and communication module. The SIMD processor is of type IC3D from Philips'Xetal and it consists of Linear Processor Array (LPA) with 320 RISC processors. 8051 controller from ATMEL is used as a general purpose processor and it includes 1.79MB RAM, 64KB Flash, and 2KB EEPROM to store the parameters and instruction code for IC3D processor. Both processors are coupled using a 128KB dual port RAM that enables them to work in a shared workspace asyn-



Platform	Processor	Memory		Camera & Resolution	Radio	Power consumption
		RAM	Flash			
Cyclops [105]	8-bit ATMEl ATmega128L MCU + CPLD	64 KB	512 KB	Agilent compact CIF CMOS ADCM-1700 128x128 @ 10fps	Interfaced with Mica2 or Micaz IEEE 802.15.4	110 mW – 0.76 mW
Imote2 + Cam [101] [102]	32-bit PXA271 XScale processor (Imote2)	256 KB (Imote2)	32 MB (Imote2)	IMB400 camera OmniVision OV7649 640x480@30 fps	Integrated CC2420 IEEE 802.15.4	322 mW - 1.8 mW
FireFly Mosaic [106]	60MHz 32-bit LPC2106 ARM7TDMI MCU	64 KB	128 KB	CMUCam3 352x288 @ 50 fps	Interfaced with FireFly mote IEEE 802.15.4	572.3 mW – 0.29 mW
eCam [104]	OV 528 serial-bridge controller JPEG compression only	4 KB (Eco)	-	CoMedia C328-7640 (includes OV7640) 640x480 @ 30 fps	Interfaced with Eco wireless mote nRF24E1 radio RF 2.4 GHz 1Mbps	70 mA at 3.3V
MeshEye [109]	55 MHz 32-bit ARM7TDMI based on ATMEL AT91SAM7S	64 KB	256 KB	Agilent ADNS-3060 30x30 Agilent ADCM-2700 640x480 @ 10 fps	Integrated CC2420 IEEE 802.15.4	175.9 mW – 1.78 mW
Panoptes [99]	400 MHz 32-bit PXA255 XScale CPU (Stargate)	64 MB (Stargate)	32 MB (Stargate)	Logitech 3000 USB Camera 160x120 @ 30 fps 640x480 @ 13 fps	PCMCIA IEEE 802.11 wireless card	5.3 W – 58 mW
Wica [107]	84 MHz Xetal II SIMD + 8051 ATMEL MCU	1.79 MB + 128 KB DPRAM	64 KB	VGA color camera 640x480 @ 30 fps	Aquis Grain ZigBee IEEE 802.15.4	600 mW max
MicrelEye [14]	8-bit ATMEL FPSLIC (includes 40kG FPGA)	36 KB + 1 MB external SRAM	-	Omnivision OV7640 320x240 @ 15 fps	LMX9820A Bluetooth 230.4 Kbps	500 mW max
WiSN [103]	48 MHz 32-bit ARM7TDMI based on ATMEL AT91SAM7S	64 KB	256 KB	Agilent ADCM-1670 352x288 @ 15 fps Agilent ADNS-3060 30x30 @ 100 fps	Integrated CC2420 IEEE 802.15.4	110 mA – 3 mA at 3.3V
CITRIC [94]	624 MHz 32-bit Intel XScale PXA270 CPU	64 MB	16 MB	Omnivision OV9655 1280×1024 @15 fps 640×480 @ 30 fps	Interfaced with Tmote Sky mote IEEE 802.15.4	1 W max
Fox + Cam [13]	100 MHz LX416 Fox board	16 MB	4 MB	Labtec Webcam bro QuickCam Zoom 640x480	USB Bluetooth IEEE 802.15 100 m	1.5 W at 5 V
XYZ + Cam [101]	58MHz 32-bit ARM7TDMI based on OKI ML67Q5002 (XYZ)	32 KB (XYZ)	256 KB + 2 MB on board (XYZ)	Omnivision OV7649 640x480 320x240 @ 4.1 fps	Integrated CC2420 IEEE 802.15.4 (XYZ)	238.6 mW – 2.2 mW

Table 2.5: WMSN Camera Motes Features and Specifications

chronously. The Aquis Grain Zigbee from ChipCon'CC2420 transceiver implements the wireless communication module. The multimedia processing in this camera sensor mote is divided into three levels: low, intermediate, and high-level image processing. Low-level image processing (pixel level) is manipulated by the SIMD processor and it is associated with typical kernel operations such as convolutions, data dependent operations using neighboring pixels, and initial pixel classification. The intermediate and high-level image processing (object level) are done by the general purpose processor because it has the flexibility to implement complex software tasks, run an operating system, and do networking application.

In [98] the authors present a camera mote for behavior recognition in wireless multimedia sensor networks based on biologically inspired address-event imagers and sensory grammars. In Address Event Representation (AER), the camera networks operate on symbolic information rather than images by filtering out all redundant information at the sensor level and outputting only selected handful of features in address-event representation. This leads to minimize power consumption and bandwidth (they only consume a few μW of power in active state and use different computation model that is faster and more lightweight than conventional image processing techniques), and helps to offer privacy concerns as certain features are being transmitted. Then the output of the AER imagers can be connected into the sensing grammar that converts low-level sensor measurements to higher-level behavior interpretation based on probabilistic context free grammars (PCFGs). PCFGs are very similar to the Hidden Markov Models and they are used because of their expressiveness, generative power, and modularity. The authors developed three different platforms to experiment the above techniques where each platform is built on top of the XYZ sensor node [105]. XYZ uses an OKI ML67Q5002 processor based on ARM7TDMI core running at 58MHz. The processor has 32KB of internal RAM and 256KB of Flash, and there is additional 2Mbit memory available on-board. The first platform is (XYZ-ALOHA) an XYZ sensor node with ALOHA image sensor that is composed of four quadrants of 32x32 pixels and it is able to generate 10,000 events in 1.3sec with a power consumption of 6 μ W per quadrant. The ALOHA image sensor uses the simple ALOHA medium access technique to transmit individual events to a receiver. The second platform is (XYZ-OV) an XYZ sensor node with a camera sensor from Omnivision that can capture images at resolution of VGA (640x480) and QVGA (320x240). Currently, imote2 has been used with Omnivision OV7649 camera as a third platform. The paper shows an example for assisted living application where the prototype network (imote2-OV) was able to distinguish between "cooking" from "cleaning" actions done by a person in a kitchen.

An energy-efficient smart camera mote, called MeshEye [106], is proposed for distributed intelligent surveillance application in WMSN. MeshEye mote architecture is designed to support in-node image processing, with sufficient processing power capabilities, for distributed intelligent algorithms in wireless sensor network of two tiers while minimizing component count and power consumption. In the first tier, a low-resolution stereo vision system is used to determine position, range, and size of moving objects in its field of view. The second tier contains high resolution cameras that are triggered in case of detecting objects by the first tier. The MeshEye mote has an Atmel AT91SAM7S microcontroller board with 64KB SRAM and 256KB Flash memory, and the mote can host up to eight kilopixel imagers (Agilent ADNS-3060) and one VGA camera module (Agilent ADCM-2700). The wireless communication module uses CC2420 2.4GHz IEEE 802.15.4 RF transceiver that can support up to 250 Kbit/s. Although the supported data rate is not high enough for multimedia streaming, the authors show that it is still possible by conducting in-node intermediate-level visual processing for efficient image compression and/or descriptive representations (such as axis projection, color histogram, or object shape). Also the authors present a basic power model that estimates the energy consumed in different operation modes by the battery-powered MeshEye mote.

In [96] the design, implementation, and performance of video-based sensor networking architecture using visual sensor platform (called Panoptes) are introduced for delivering high quality video over 802.11 wireless networks. The initial developed hardware platform of Panoptes was the Applied Data Bitsy board utilizing the Intel StrongARM 206-MHz embedded processor connected with Logitech 3000 video camera via USB. Because of the limitations found by using this design such as slow video capturing, low processing power, high power consumption, and small available memory, the authors prefer to use Crossbow Stargate platform that has twice processing power more than the Bitsy board, consumes less power, and has smaller size. The second design of Panoptes node based on Stargate platform offers video capturing at reasonable frame rate (more than 15 fps) using Logitech 3000 pro webcam. After video capturing, the software module in Panoptes provides video frame compression, both spatially and temporally, using JPEG, differential JPEG, and conditional replenishment. Also the software module in Panoptes provides other functionalities such filtering for dropping similar video frames, buffering management, and adaptation for network status. These functionalities can be accessed simply by function calls based on Python language by which the user can chose the preferred algorithm or method in each subcomponent (e.g. selecting compression algorithm or filtering method) at the run time without the need of manually reprogramming the nodes. At the end, the authors show the implementation and performance of a video aggregation application and some algorithms (such as prioritizing buffer management algorithm and bit-mapping algorithm for video querying) using Panoptes nodes.

As an example of medium-resolution camera mote, an embedded camera mote platform [10] based on Fox board has been introduced for wireless multimedia sensor network applications. The designed platform has several sensors including GPS positioning receiver, current consumption sensor, and image sensor beside the wireless transceiver. The Fox board LX416 has 100 MHz CPU, 4MB Flash, and 16MB of RAM running GNU/Linux as operating system and because of these capabilities it is attended to be use for high-level device of multi-tier model. The platform can be connected via USB ports with webcam (QuickCam Zoom or Labtec Webcam) and Bluetooth dongle. The designed platform is using Bluetooth (IEEE 802.15) for data transmission, rather than 802.11 comparing to other high-level platforms like Panoptes or the one used in SensEye, because of the availability of USB-Bluetooth dongles, open-source software support and moderate power consumption. The current consumption sensor is used as an energy analyzer to study energy consumption of nodes during image transmission and it is shown in the experimental results that image grabbing and transmission needs more power than image routing.

The work in [11] proposes the design of a wireless video sensor node, called MicrelEye, for video processing and image classification in wireless multimedia sensor networks. The device is equipped with a VGA CMOS (OV7640 from Omnivision) image sensor, a reconfigurable processing engine, and a Blue tooth 100m transceiver. The design is intended to be low-cost low-power multimedia sensor node that can support dynamic reconfiguration capabilities and local processing for multimedia content, such as back ground subtraction, image recognition and classification, before wireless transmission. An optimized hardware-oriented support vector machine-like (SVM-like) algorithm called ERSVM is used for image classification process. The devise uses a System on Chip (SoC) for the processing engine, ATMEL FPSLIC, which includes AVR 8-bit RISC MCU, 40K gates FPGA, and 36KB SRAM. An external 1MB SRAM is also added to provide the required memory resources for multimedia processing and enable parallelized computation between hardware and software. With these specifications, the device targets a power budget of 500 mW and supports people detection at 15 fps at QVGA (320x240) image resolution. For wireless communication, a 100m LMX9820A Bluetooth transceiver has been used because of its low power of consumption, the ease to interface MicrelEye with other devices, and its high data rates (up to 704 kbps). In Video processing algorithm, the FPGA starts the process by acquiring frames from the image sensor, and then a back ground subtraction is done on each acquired frame. After that the region of interest (ROI), 128x64 subimage, is extracted and stored into on-chip memory to be processed by MCU. The MCU will conduct on the ROI a feature extraction to form the feature vector and image classification using ERSVM algorithm.

A camera mote called CITRIC is developed in [91] for wireless multimedia sensor networks to enable in-network processing of images in order to reduce communication overheads. The hardware design of the camera platform consists of camera sensor, PDA class processor, 64MB RAM, 16MB Flash, and microphone. The camera sensor is a 1.3 megapixel OmniVision OV9655 camera that can support different image resolution -from SXGA (1280x1024) through VGA, CIF to 40x30- outputting 8bit/10bit images at a rate of 30 fps in VGA and lower resolution and typically consumes 90mW in active state. The processor is PXA270 frequency-scalable (up to 624MHz) fixed-point, and it has 256KB internal SRAM and a wireless MMX coprocessor to accelerate multimedia operations. Then this camera device is connected to standard sensor network mote (Tmote Sky) to form the wireless camera mote, CITRIC, which communicates over the IEEE 802.15.4 protocol at a rate of 250 Kbps. The camera mote first performs pre-processing functions on the captured images from the camera sensor and then sends the results over the network to a central server. Also the paper proposes a back-end client/server architecture to provide user interface to the system and support further centralized image processing. The authors implement three applications over the proposed platform which are image compression, target tracking, and camera localization. In image compression application, it is shown that Compressed Sensing (CS) using random matrices provides unique advantages in lossy compression than JPEG standard when both are implemented using the integer DCT implementation that is supported by the fixed point arithmetic processor. The single target tracking application is implemented via background subtraction using frame differencing. Then the foreground pixels are processed for identification and tracking.

An implementation of Dual-Camera sensor is presented in [99]. Each of them comprises of a low-power and high-power tier and they are physically connected together to have similar FoV. The low-power camera sensor node (Tier-1) consists of a MICAz mote equipped with a low fidelity Cyclops camera sensor, and a 1GB NAND flash for storing images. The high-power camera sensor node (Tier-2) consists of a more-powerful platform, imote2 equipped with a high fidelity Enalab camera (OV7649 CMOS camera supports color VGA (640x480) resolution), and a 1GB SD card for image storage. The system uses the low-power Tier-1 for object detection and the high-power Tier-2 for energy efficient object recognition and classification. The wireless communication is based on IEEE 802.15.4 Zigbee standard at 2.4GHz.

2.9.3 Testbeds

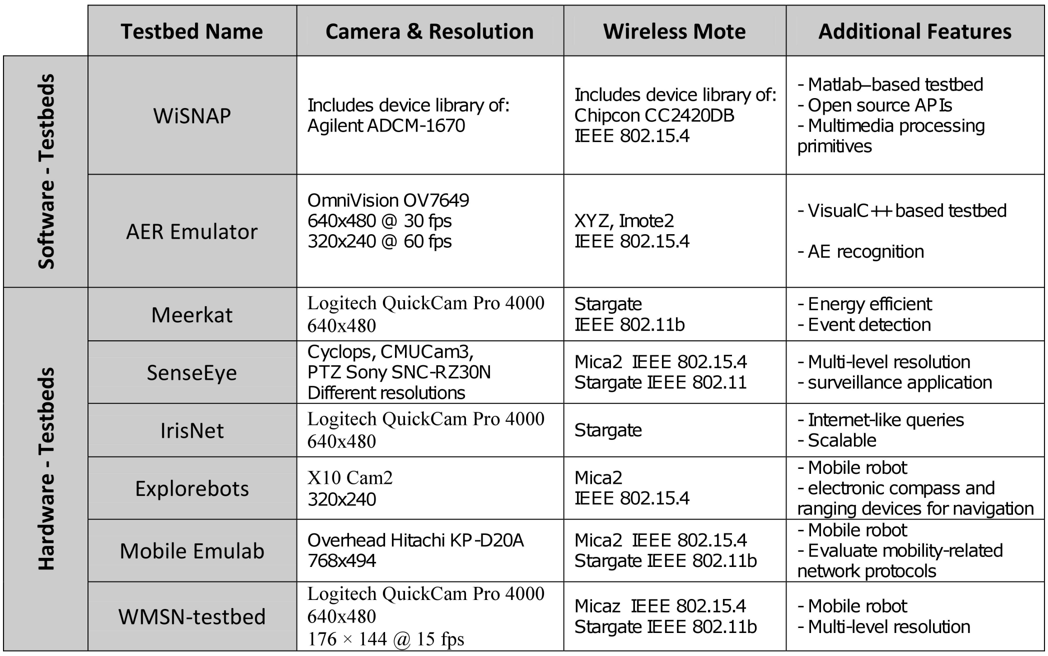
To evaluate different protocols and algorithms pertaining for different networking layers (transport, network, MAC, or physical layer) of wireless multimedia sensor network or test various applications over the WMSN, researchers may perform analytical analysis, conduct experiments, or use simulations. Sometimes, analytical analysis neither gives an accurate model for such complicated wireless system nor truly depicts the behavior of real-time wireless networks. Also, in many cases, tests and experiments in wireless sensor network in general and in wireless multimedia sensor network in particular are somehow complex and time-consuming, and hard to be re-conducted by other researchers. For these reasons, simulation has been the preferred methodology for many researchers in the wireless multimedia sensor network domain. However, the existing simulators have many defects and are unable to model many critical characteristics of real-time wireless systems. Also, because of not following the scientific research standards in conducting of such simulation studies, simulation results are sometimes doubtful and have less credibility [107]. For these reasons and in order to minimize the differences in results between theoretical and practical approaches, which will significantly affect the behavior of real-time systems, testbeds have been increasingly used by the researchers and developers to evaluate their proposed algorithms and applications.

WMSN testbeds are used for better understanding and satisfying the practical and technical challenges of networks deployed in real-time systems. While testbeds have become the preferred method for testing and evaluating with wireless multimedia sensor network applications, they also provide means for integrating several individual sensors on a common wireless platform in a controlled and instrumented environment. Thus, research on experimental testbeds with current hardware and software platforms, allows users not only to demonstrate applicability and evaluate application-level and network-level performance metrics (e.g. detection probability, end-to-end delay, jitter, quality of received multimedia streams, etc) in real environments, but also to validate research prototypes. Compared with conducting real-time experiments and field deployments, testbeds give considerable efficiency in testing potentially long-time experiments, which is important in debugging, validation, and integration phases of reliable wireless multimedia sensor networks. WMSN testbeds can be classified into two categories, *Software Testbeds* and *Hardware Testbeds*. Table 2.6 illustrates the existing software and hardware testbeds found in the literature and summarizes their specifications and important features.

2.9.3.1 Software Testbeds

To facilitate advanced research in wireless multimedia sensor network technology, software driver interfaces and libraries are designed to help researchers in testing and evaluating various algorithms and applications through using easy-to-use Application Program Interfaces (APIs) and functions. These APIs and functions provide testing environment through abstraction layers that hide the low-level details of the underlying hardware in order to enable easy and fast development of multimedia sensor network applications.

WiSNAP [108] is a Matlab-based software testbed designed for wireless multimedia sensor networks, where the developers can test and evaluate algorithms and applications using its standardized and easy-to-use Application Program Interfaces (APIs). WiSNAP provides a Matlab framework as a high-level and powerful programming environment for implementing interfaces to the existing wireless motes and image



	Testbed Name	Camera & Resolution	Wireless Mote	Additional Features
Software - Testbeds	WiSNAP	Includes device library of: Agilent ADCM-1670	Includes device library of: Chipcon CC2420DB IEEE 802.15.4	- Matlab–based testbed - Open source APIs - Multimedia processing primitives
	AER Emulator	OmniVision OV7649 640x480 @ 30 fps 320x240 @ 60 fps	XYZ, Imote2 IEEE 802.15.4	- VisualC++ based testbed - AE recognition
Hardware - Testbeds	Meerkat	Logitech QuickCam Pro 4000 640x480	Stargate IEEE 802.11b	- Energy efficient - Event detection
	SenseEye	Cyclops, CMUCam3, PTZ Sony SNC-RZ30N Different resolutions	Mica2 IEEE 802.15.4 Stargate IEEE 802.11	- Multi-level resolution - surveillance application
	IrisNet	Logitech QuickCam Pro 4000 640x480	Stargate	- Internet-like queries - Scalable
	Explorebots	X10 Cam2 320x240	Mica2 IEEE 802.15.4	- Mobile robot - electronic compass and ranging devices for navigation
	Mobile Emulab	Overhead Hitachi KP-D20A 768x494	Mica2 IEEE 802.15.4 Stargate IEEE 802.11b	- Mobile robot - Evaluate mobility-related network protocols
	WMSN-testbed	Logitech QuickCam Pro 4000 640x480 176 × 144 @ 15 fps	Micaz IEEE 802.15.4 Stargate IEEE 802.11b	- Mobile robot - Multi-level resolution

Table 2.6: WMSN Testbeds Features

sensors though simple easy-to-use functions and libraries. These functions and libraries hide the internal details of dealing with mote or sensor specific interfaces from the end-users and provide the users with many powerful and rich image processing tools. Currently, WiSNAP includes device libraries for Agilent's ADCM-1670 camera module, Agilent's ADNS-3060 optical mouse sensor, and Chipcon's CC2420DB IEEE 802.15.4, but it can be extended to include and support any kind of sensor or wireless mote as it is an open source architecture. WiSNAP consists of two application program interfaces: 1) an image sensor API that enables frame capturing from image sensors after identifying the type of image sensor and number of frames, and 2) a wireless mote API that provides access to wireless motes through functions for initialization and MAC packet transmission and reception. Then these set of functions provided by the mentioned APIs are matched with the corresponding device libraries that lie below the API layer in WiSNAP program stack and provide a set of hardware-dependent functions (such as Agilent ADCM-1670 image sensor). Two application examples of using WiSNAP development platform are presented for event detection and node localization. Event detection is based on tracking of the number of changed pixels that exceed a certain threshold between successive image frames. For node localization, the distance is estimated using the received signal strength indicator RSSI from the node and the direction is calculated by extracting the relative angle of a continuously blinking LED on that node from the captured images using frame differencing of adjacent frames.

Also in [98], a software testbed based on address event representation of image sensing is developed. The software testbed consists of an emulator of AER imagers - written in VisualC++ and runs under windows. the AER Emulator takes an 8-bit grayscale input stream from a COTS USB camera and outputs a queue of events to a text file. The AER classification is done in a way similar to the Hidden Markov Models (HMMs). At the receiver side, the image array can be obtained by converting the event frequency data into the original feature (e.g. light intensity) using two ways, Histogram reconstruction or Inter-event reconstruction.

2.9.3.2 Hardware Testbeds

Hardware testbeds involve deploying of hardware devices, such as multiple types of cameras with different resolutions and image-processing abilities, and wireless communication hardware that may support multiple standards and different data rates. Besides that, hardware testbeds provide supporting software for data monitoring and user interface. Depending on the hierarchal organization supported by the network, Hardware testbeds can be further divided into single-tier or multi-tier testbeds.

A) Single-Tier Hardware Testbeds:

Meerkats [95] is a testbed of wireless network of battery-operated camera nodes used for monitoring and surveillance of wide areas. The Meerkats node, which is based on Stargate board and using 802.11b wireless card, is equipped with sufficient processing and storage capabilities (when compared to a Cyclops node) for running relatively complex image processing algorithms. The goal of the work is to measure the tradeoff between application-specific performance and power efficiency (or network life time) for a given resource management strategy. Meerkats currently composed of eight visual sensor nodes, each of which consists of -as we mentioned- a battery powered Crossbow Stargate board, which has an XScale PXA255 CPU (400 MHz) with 32MB flash memory and 64MB SDRAM, connected with a Logitech QuickCam Pro 4000 webcam via USB and IEEE 802.11b PCMCIA wireless card. The Stargate platform is selected for the Meerkats node because it is running an open source operating system (Linux kernel 2.4.19), it can be easily connected to a webcam, the image sensor in Meerkats node, and it provides sufficient processing and storage capabilities. A laptop acts as base station or information sink running a multithreaded server program. For energy conservation, the Meerkats node operates according to a specific duty cycle in which it switches periodically its components (processor, camera, radio) into different operation states (sleep, idle, active) and performs a specified tasks. Meerkats's energy performance evaluation can be seen at [109]. The Meerkats node performs all the image-related tasks such as image acquisition, processing, and compression when it is in the active state. For example, for event detection, the moving blobs in the image are detected using a fast motion analysis algorithm and the relevant information is compressed using JPEG standards. The communication, based on multi-hop routing, between Meerkats nodes are established using the Dynamic Source Routing (DSR) routing protocol through IEEE 802.11b links.

The Mobile Emulab network testbed [110] provides a remotely accessible mobile wireless and sensor testbed. The mobile testbed can provide accurate positioning and monitoring using video camera equipments, and enable automated experiments by both on-site and off-site users by using open-source software and COTS equipments. The testbed consists of Acroname Gracia robots attached with Mica2 motes, Stargate boards with IEEE 802.11b cards, and low-cost Hitachi KP-D20A cameras. The testbed is used in an indoor field of sensor-equipped motes and webcams, and can provide simple path planning as well as vision-based tracking system accurate to 1 cm. Mobile Emulab testbed allows remote user to position the robots, control all the computers and network interfaces, run arbitrary programs, and log data in a database. Emulab testbed allow, through precise positioning and automation, quick evaluation of localization and mobility protocols in sensor-driven applications.

Low-cost, vision-enabled, and flexible autonomous mobile robots were designed in the Explorebots testbed in [111] for indoor experimentation on multi-hop ad hoc and sensor networking. The wireless robots are equipped with MICA2 sensor motes for sensing and wireless communication, in addition to built-in electronic compass, velocity and distance sensors, motor movement control, and sonic-based ranging sensors that can be used for navigation. The hardware components of the robot consist of mobile platform, 8-bit Rabbit semiconductor R3000 programmable microprocessor with Flash memory, a 320x240 pixel X10 Cam2 color camera, sensing elements, communication devices, and batteries. Explorebots testbed has been used for target localization experiments by processing the sound and light sensors outputs to guide the robots towards the target source, in addition to validate hybrid routing protocols.

B) Multi-Tier Hardware Testbeds:

IrisNet [112], internet- scale resource-intensive sensor network services, is an example of wireless multimedia sensor network Multi-Tier Hardware testbed that provides shared internet-scale long-lived software platform for many sensor applications. IrisNet has been designed to overcome the difficulties of building large-scale distributed networks comprise of many scalar and visual sensors, and the challenges of dealing with large volumes of collected data. Therefore, the proposed platform enables the creation of a planetary infrastructure of multimedia sensors and enables application-specific processing of the collected data by these sensors using their processing capabilities. IrisNet allows user to query the collected information, stored in distributed XML database infrastructure close to its sources, by using internet-like queries. IrisNet also provides a number of multimedia processing primitives that new applications can use as building blocks such as camera calibration, key-points or reference points implementation, and image stitching. The architecture of IrisNet is two-tiered: Sensing Agents tier (SAs) for data collection and filtering, and Organizing Agents tier (OAs) for data storage and querying. There are three steps in order to develop an application using IrisNet. First, the application developer creates the sensor database XML schema that defines the attributes, tags, and hierarchies used to describe and organize distilled sensor data. Second, the application developer writes the software running in the SAs (called senselet) to filter the collected sensory data and update the database defined by the schema. Third, the application developer provides an application-specific front end interface for end users to access the application.

In [94], the design and implementation of SenseEye is presented, a multi-tier network of heterogeneous wireless sensor nodes and cameras. SenseEye is designed for surveillance application in WMSN where resource-constrained low-power elements are employed to perform simpler application tasks while more capable high-power elements are used for more complex tasks. The work aims to exploits the advantages of multi-tier sensor network comparing to a single-tier network such as low cost, wide coverage, high functionality, and high reliability by proposing numerous mechanisms and optimizations for object detection, object localization, inter-tier wakeup, object recognition, and tracking. SenseEye is implemented in three-tier network consisting of four types of camera sensors where nodes within each tier are assumed to be homogeneous while different tiers are assumed to be heterogeneous with respect to their capabilities. The processing power, networking capabilities, and imaging resolution improve from the lower tier to the higher tier at the expense of increased power consumption. The lowest tier consists of low-power sensor motes such as MICA2 equipped with low fidelity and resolution camera sensors such as Cyclops or CMUcam3. The second tier consists of Stargate nodes equipped with higher fidelity and medium resolution webcams. The third tier contains a sparse deployment of high resolution pan-tilt-zoom (Sony SNC-RZ30N) cameras connected to embedded PCs. In this system, no base station is assumed and the communication between Tier 1 and Tier 2 is low rate through 900MHz radio while the communication between Tier 2 and Tier 3 is done through 802.11 radio. The main design principles of the proposed system are mapping each task to the lowest powerful tier that has the sufficient resources to accomplish the needed tasks reliably within the required latency, exploiting the wakeup on-demand and triggering of the higher tier nodes only when necessary in order to save energy, and exploiting the information redundancy from overlaps in cameras coverage to improve energy-efficiency and performance (e.g. overlaps camera coverage information can be used in object localization for intelligently wakeup the correct nodes in the higher tier). The paper shows a practical example of the proposed system where the nodes in the first tier are always turned on or duty-cycled (woke up periodically) and used for object detection (through simple frame differencing) and then, in case of detecting an object, the nodes try to localize the detected object exploiting the information redundancy from overlapping camera coverage and using triangulation techniques for localization. After that, the nodes in Tier 1 woke up the nodes in Tier 2, which are close the detected object (within their FoV), that in turn perform object recognition by capturing photos of the object, identifying object features, and searching the database for a match. Finally, the corresponding nodes in the Tier 3 are woken up to perform object tracking with the help with the other tiers as the detected object is moving.

The WMSN-testbed [13] at the Broadband Wireless Networking (BWN) Laboratory at Georgia Tech is based on commercial off-the-shelf (COTS) advanced devices and has been built to demonstrate the efficiency of algorithms and protocols for multimedia communications through wireless sensor networks. The testbed is integrated with the scalar sensor network testbed, which is composed of a heterogeneous collection of Imote2 and Micaz motes from Crossbow. The testbed allows the integration of heterogeneous devices in experimental testbeds and includes three different types of multimedia sensors: low-end imaging sensors, medium-quality webcam-based multimedia sensors attached with Stargate boards, and pan-tilt cameras mounted on Acroname GARCIA mobile robots. The testbed uses both IEEE 802.15.4 and IEEE 802.11b for wireless communication and it is capable to deliver JPEG video streaming in QCIF format (176 144) at 15 fps.

2.10 Conclusions

In this chapter, we outline the design challenges of WMSNs, give a comprehensive discussion of the proposed architectures, algorithms and protocols for the different layers of the communication protocol stack for WMSNs, and evaluate the existing WMSN hardware and testbeds. In next chapters, we present our proposed solutions for routing, security and privacy in WMSNs.

Chapter 3

Cluster-based Multipath Routing Protocol For WMSNs

3.1 Introduction

Routing in WMSNs is very challenging and critical because of their characteristics and constraints that make them different from the existing communication and scalar wireless sensor networks [113; 114], such as: 1) Large number of heterogeneous sensor nodes with different capabilities and functionalities is deployed. 2) Careful resource management for multimedia transmissions is required as sensor nodes are tightly constrained in terms of battery energy, processing power, storage capacity, and available bandwidth. 3) Also delivering the collected multimedia data in WMSNs (video streaming, still images, audio) adds more constraints on the design of the routing protocols in order to meet their QoS requirements such as end-to-end delay, SNR (signal-to-noise ratio) level, packet (frame) loss rate, etc. 4) In addition, the use of densely deployed nodes provides significant redundancy in the collected sensor data, e.g. overlapping of FoVs (Field of Views) of camera sensors. Such redundancy needs to be exploited to improve energy and bandwidth utilization, and for more accurate and robust observation results through effective data fusion and aggregation.

[1] Chapter 3 is based on the following publications:
(1) *A Secure Cluster-Based Multipath Routing Protocol for WMSNs; Islam T. Almalkawi, Manel Guerrero Zapata, Gamal N. Al-karaki* appeared in *Sensors Journal 11, Volume 4, Pages 4401-4424, 2011*
(2) *A Cross-layer based Clustered Multipath Routing with QoS-aware Scheduling for Wireless Multimedia Sensor Networks; Islam T. Almalkawi, Manel Guerrero Zapata, Gamal N. Al-karaki* appeared in *International Journal of Distributed Sensor Networks, 2012*

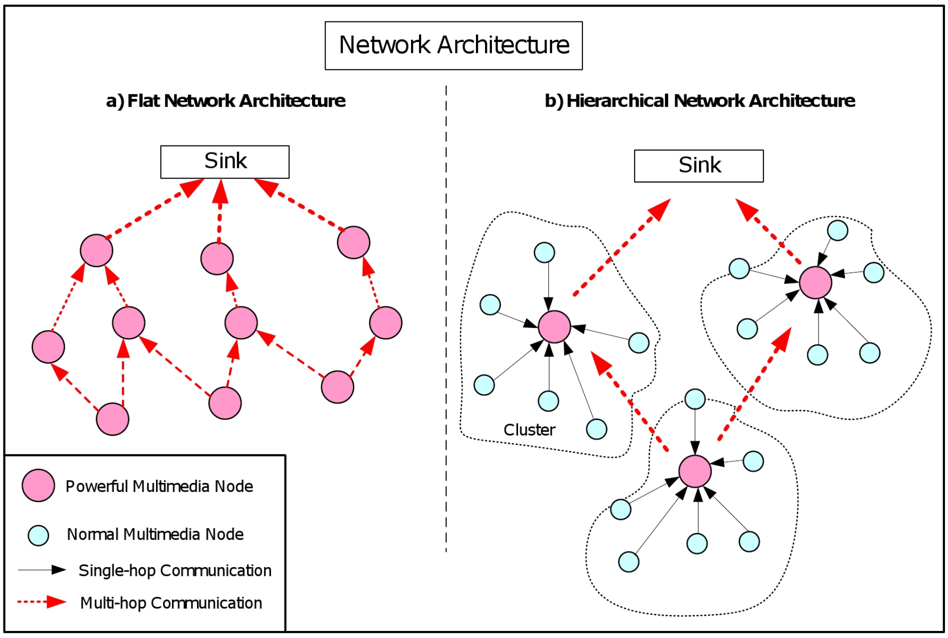


Figure 3.1: Flat vs. Hierarchical Network Architecture.

In general, two types of WMSNs architecture are widely used [113]: flat and hierarchical (cluster-based) network architecture as shown in Figure 3.1. In flat architecture, the network is deployed with homogeneous sensor nodes of the same capabilities and functionalities, which can perform any task from image capturing through multimedia processing to packet relaying toward the sink in multi-hop basis. On the other hand, at cluster-based architecture, the network is divided into clusters. Heterogeneous sensor nodes are deployed in each cluster, where camera, audio and scalar sensors relay data to a cluster head that has more resources and able to perform intensive data processing. The cluster head is wirelessly connected with the sink or the gateway either directly or through other cluster heads in multi-hop fashion. For WMSNs, cluster-based network architecture has more advantages than a flat network especially for image processing and transmissions. In the homogeneous flat network, all the nodes should have the same hardware capabilities and functionalities for multimedia processing and transmission, and this leads to increase the energy consumption and the cost of the deployed network. Also, a single-tier flat architecture can cause the sink to overload with the increase in sensors density, which can affect the performance of the network and cause

latency in communication and tracking events. Moreover, in cluster-based network, cluster heads can perform data aggregation and filtering to reduce the amount of transmitted data, and do better scheduling among the nodes within clusters.
Therefore in this chapter, we pursue a cluster-based with multipath routing protocol to allow the network to cover a large area of interest and cope with additional load without degrading the quality of service. Our proposed routing protocol aims to cluster the nodes, so that cluster heads can do some aggregation and reduction of data in order to save energy consumption and bandwidth usage, and to find the maximum number of paths suitable for the different requirements of handling different traffic classes.

Design of an efficient QoS-aware MAC protocol [115; 116] is another important step for correct delivery of real-time multimedia data and for end-to-end QoS provisioning over WMSNs. It is desirable that the MAC layer provides reliable and error-free data transfer with minimum retransmissions while meeting the QoS requirements with efficient resource utilization. The existing sensory MAC protocols are mostly based on variants of the Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) [117] and Time Division Multiple Access (TDMA) [116]. Contention-based approach, such as CSMA/CA, is preferred when network traffic load is not very intensive and the channel condition is relatively unreliable because the probability of potential collision and congestion is low. However, for heavier traffic loads as the case of WMSNs, contention-based approach leads to increase wasted energy and delays due to idle listening and collisions produced with large preamble and hidden node problems. On the other side, contention-free approach, like TDMA, is more appropriate for multimedia applications with reliable channel conditions and heavier traffic load. However, it suffers from clock synchronization problem, in addition to channel under-utilization and fixed time-slot assignments in case of static slotted scheduling.

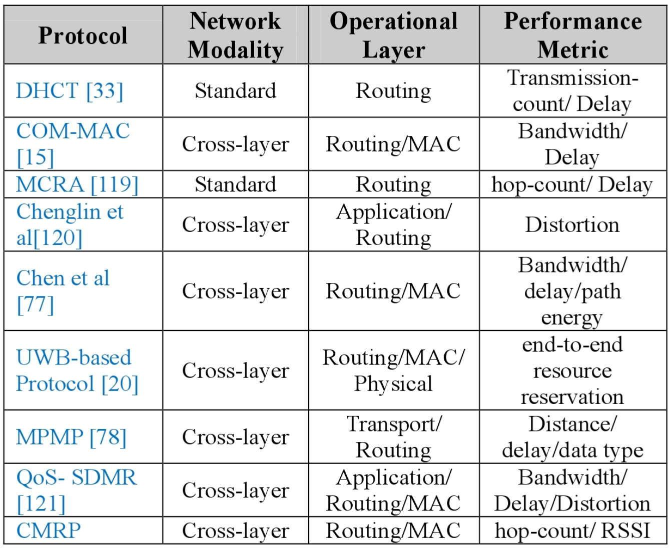
A quick look into the existing proposals in routing protocols in sensor networks reveals that they are following the standard structure of the communication protocol stack and do not pay intention for the interdependencies and joint functionalities among the layers especially the routing and MAC layers. Therefore in this chapter, we also adopt a cross-layer design between the routing and MAC layers where our clustered multipath routing protocol is combined with an adaptive QoS-aware scheduling to optimize the performance of the routing and reliable delivery with minimum resource consumption. Our proposed scheduling mechanism is based on adaptive QoS-aware TDMA approach used at two levels in the network: within clusters and among cluster heads. Our algorithm uses flexible time-slot assignment where a cluster head is responsible to schedule the traffic toward the sink from the sensor nodes based on the type of data and its availability.

3.2 Related Work

Several cross-layer and communication protocols were developed to address the aforementioned issues for WMSNs and surveyed in [118]. Here in this section we focus on cross-layer optimization proposals that include routing and scheduling functionalities and we summarize them in Table 3.1:

A routing protocol, DHCT [33], is proposed for WMSNs to support multipath routing and to reduce interferences between close paths by using a performance metric (called Costp), which is product of expected transmission count and the delay. Using Costp metric allows selecting paths with minimum interferences to each other, hence it will increase throughput. Also DHCT reinforces multiple links at the sink to obtain disjoint path from the source for multipath routing. However, this routing protocol does not consider the bandwidth as QoS metric for routing decision or prioritizes the incoming packets to schedule them as the case in CMRP, but it does consider the play-out deadline in a sense that the packet arrives after the deadline will be discarded. Also having only disjoint paths means that not considering interconnecting paths that can be effectively used along with packet scheduling to utilize the available capacity and increase throughput.

COM-MAC [15] is a multi-channel multipath MAC protocol with packet scheduling to meet the bandwidth and delay requirements in WMSNs. The routing protocol uses multiple paths, multiple channels, and QoS packet scheduling technique based on the dynamic bandwidth adjustment and path-length-based proportional delay differentiation (PPDD) techniques. These requirements (bandwidth and delay) are adjusted locally at each node based on the path-length and incoming traffic in static flat wireless network where all the nodes are homogeneous multimedia sensor nodes for the same capabilities (video, audio, scalar data) and equipped with single radio interface



Protocol	Network Modality	Operational Layer	Performance Metric
DHCT [33]	Standard	Routing	Transmission-count/ Delay
COM-MAC [15]	Cross-layer	Routing/MAC	Bandwidth/ Delay
MCRA [119]	Standard	Routing	hop-count/ Delay
Chenglin et al[120]	Cross-layer	Application/ Routing	Distortion
Chen et al [77]	Cross-layer	Routing/MAC	Bandwidth/ delay/path energy
UWB-based Protocol [20]	Cross-layer	Routing/MAC/ Physical	end-to-end resource reservation
MPMP [78]	Cross-layer	Transport/ Routing	Distance/ delay/data type
QoS- SDMR [121]	Cross-layer	Application/ Routing/MAC	Bandwidth/ Delay/Distortion
CMRP	Cross-layer	Routing/MAC	hop-count/ RSSI

Table 3.1: Cross-layer Design and Communication Protocols for WMSNs.

and multi-channels. However, unlike CMRP, it uses static time slots at control channel and does not propose any mechanism for passive nodes for better channel utilization. CMRP uses adaptive time slots assignment that can be changed dynamically depends on data availability, data type, and number of active nodes.

Another multi-constrained routing algorithm, MCRA [119], is proposed to provide end-to-end delay and packet loss ratio suitable for multimedia content, and balance the energy consumption. In MCRA, routing discovery starts at the sink node that floods the network with interest-messages. The source node that receives these interest messages and matches the needed query, selects the path of minimum hop-count to send its data. When receiving data packet, the sink node calculates the coordinates of the source node using the logical coordinates (hop-count). MCRA tries to reduce the amount of flooding by either not forwarding the interest-messages already received them before, or by merging multiple interests in one message. However, the routing depends on flooding the network with interest-messages from sink to sources using all nodes to find the paths, not only using some powerful nodes to discover the routes as in CMRP. Also, the selection of the paths by the sources in MCRA is based only on minimum hop-count without considering the quality of the link that can be estimated from the received signal strength as in the case of CMRP.

The cross-layer design described in [120] jointly optimizes the source coding techniques for multimedia processing and compression in the application layer with the network coding along with the routing functions to minimize the distortion with maximum network lifetime following multiple paths. However, this approach does not provide any proposal for controlling node channel access or scheduling process among the nodes based on the data type and thus interferences and collisions cannot be avoided which leads to degrade network throughput and the quality of received data.

Another cross-layer design for WMSNs is implemented in [77] where an extension of a geographical routing protocol is presented for multipath routing along with path priority scheduling algorithm for efficient communication of real-time video over WMSNs. Using hop-by-hop deviation angle adjustment method, a path can be established using any initial deviation angle specified at the source node, and then other disjoint paths are constructed by changing the value of the deviation angle. If there is no path satisfying the required data rate, video coding parameters are adapted along with using frame skipping, reference frame selecting and intra-frame refreshing techniques in order to lower the bandwidth consumption. To meet the delay constraint of video frames, a path priority scheduling algorithm is used that gives a weight for each path calculated based on the estimated available bandwidth, path delay, and path energy level. Then, by using path weight along with packet priority, shorter delay paths will be used for time-constrained packets while other paths are used for balancing energy and bandwidth usage for other traffic. However, like most geographical routing protocols, the proposed cross-layer design assumes that nodes are location aware and that the density of nodes is high.

A cross-layer system is presented in [20] to provide QoS for WMSN applications based on the Time-Hopping Impulse Radio Ultra-Wide-Band (TH-IR-UWB) transmission technique. This architecture tries to solve the shortcomings of using CSMA/CA for the MAC layer and to provide QoS for WMSNs. Routing process starts at the source nodes by sending request packets describing their requirements to their neighbors, and among the replies a source node selects the one who has the most positive advance toward the sink and able to satisfy the needed requirements and this continue iteratively until the end-to-end path is established to the sink. The cross-layer system also provides dynamic channel coding and receiver-centric scheduling based on time hopping sequence of impulse radio of Ultra-Wide-Band MAC and physical layer.

This allows for multiple parallel transmissions, prevent collisions at the receiver node (by using unique TH sequence for each receiver) and save energy by avoiding idle listening and wasteful transmissions (by turning on exactly on the incoming transmission). However, this cross-layer system is based on end-to-end resource reservation that causes higher overhead and it does not consider the source rate adaptation with the network conditions.

In [78], a context-aware cross-layer multi-path multi-priority (MPMP) transmission scheme is proposed where algorithms in routing and transport layers are used. In this scheme, a multipath geographic routing protocol is used to explore maximum number of node-disjoint routing paths. Also, a context-aware multipath selection algorithm in the transport layer is used to choose the maximum number of paths from all found node-disjoint paths for maximizing the delivery of the important data to the sink. The selection algorithm selects the proper routing paths that are suitable for each type of multimedia content based on two types of priority: end-to-end transmission delay based priority for constraint real-time video communication, and context-aware multimedia based priority (image vs. audio) for the most valuable information to the sink. However, the underlying routing protocol considers only the distance between the nodes and the sink to discover the routes and does not take into account other important parameters such as link quality and bandwidth. Also, this protocol does not support different type of traffic (video and scalar data at the same time).

A cross-layer framework is presented in [121] for QoS support in WMSNs, which optimizes the functionalities of communication protocols to maximize the number of video stream requests to be delivered without affecting their quality. The proposed design uses a source-directed multipath routing (SDMR) protocol that interacts with enhanced IEEE 802.11e MAC standard for QoS scheduling, and data link layer for multirate transmission modes. Also the cross-layer design is capable of interacting with the application layer to choose an appropriate Group of Picture (GoP) size according to network conditions and the feedbacks received from the sink. However, the SDMR routing protocol assumes a flat network with dense deployment for sensor nodes for estimating the upper bound number of hops in order to calculate the required end-to-end delay. Also SDMR establishes only disjoint paths (maximum three non-interfering paths) from a source to the sink.

3.3 Network Architecture and System Model

In this section, we briefly describe the network architecture model adopted by our proposed routing protocol, the communication pattern among the nodes, and the assumptions made in implementing the routing algorithm: Our proposed network architecture's model is following the *single-tier clustered architecture* [113], as shown in Figure 3.2, deployed with heterogeneous sensor nodes (such as camera, audio, and scalar sensors) that communicate directly in a certain schedule with a cluster head and relay their sensed data to it. However these heterogeneous sensor nodes have the same radio interface and propagation range. A cluster head has more resources, more powerful, and it is able to perform intensive data processing. These powerful nodes, cluster heads, are deployed uniformly in the network, and they are wirelessly connected with the sink either directly (in case of 1^{st}-level cluster heads) or through other cluster heads in multi-hop fashion. The communication among the sensor nodes both within a cluster and the communication between cluster-heads are managed by our proposed cluster-based multi-path routing protocol to efficiently handle multimedia content over the network and maintain the energy consumption of sensor nodes. Nodes within a particular cluster communicate directly with their cluster head in a certain schedule and each cluster head is responsible for performing data aggregation and fusion in order to decrease the amount of transferred data and the number of transmitted messages to the sink. A cluster head is also responsible for selecting a suitable path for each type of data, e.g. paths with good link quality or minimum delay are appropriate for multimedia content, disjoint paths are suitable for multimedia streaming, and paths with less strict QoS conditions can be used for scalar data.

Our proposed routing protocol is based on hop count and received signal strength index (RSSI) of the sensory message as an indication on the link quality and distance between the nodes. RSSI can be calculated in a large-scale wireless sensor network using the following propagation model:

$$P_r = P_t G_t G_r \left(h_r h_t\right)^2 d^{-4} L^{-1} \tag{3.1}$$

Where P_t and P_r are the power level of transmitted and received message respectively, G_t and G_r are the transmitter (Tx) and receiver (Rx) antenna gain respectively, h_t and

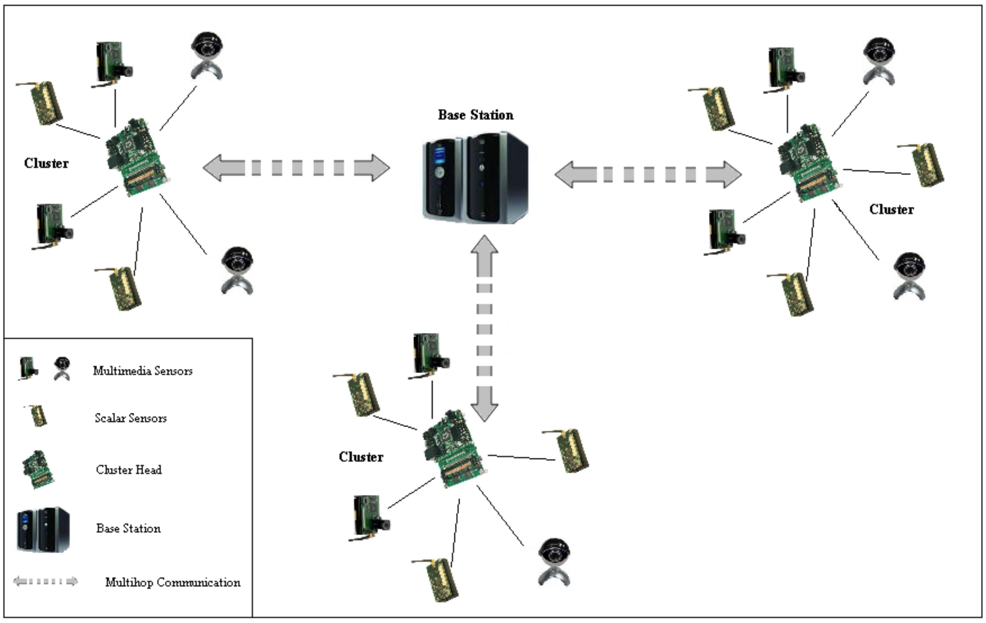


Figure 3.2: Single-tier Clustered Architecture.

h_r are antenna height for Tx and Rx respectively, d is the distance between Tx and Rx, and L is system loss factor. Then, received signal strength (RSS) of a message can be equal to:

$$RSS = P_r/P_t \rightarrow RSS = G_t G_r \left(h_r h_t\right)^2 d^{-4} L^{-1} \tag{3.2}$$

If we assume that antenna gain of Tx and Rx are equal to 1, antenna height of Tx and Rx are equal to 1, and system loss factor (L) also equals 1, then RSS can be approximated as a function of distance between the transmitter and receiver as a dominating factor affecting its value: RSS = $1/d^4$

For more accurate propagation model, signal-to-noise ratio (SNR) and bit error rate (BER) should be taken into account [122] along with the received signal strength in order to consider noises (from receiver and environment) and interferences from other packets arrived simultaneously:

$$SNR = 10\log\left(\frac{P_r}{N_p + \sum_{i=1}^{n-1} P_r}\right) \tag{3.3}$$

$$BER = 0.5 \times erfc\left(\sqrt{\frac{P_r \times BW}{N_p \times R}}\right) \tag{3.4}$$

Where BW is the channel bandwidth, N_p is noise power, n is number of interfering packets, and R is data rate.

3.4 Cluster-based Multipath Routing in WMSN

This section describes the routing operation of our proposed Cluster-based Multipath Routing Protocol for WMSNs, *CMRP*, which is based on the hierarchical structure of multiple paths established depending on hop count and received signal strength (along with measured SNR & BER) as an indication on the link quality and distance between the nodes. CMRP depends on the local information exchanged among the nodes to establish the routes to the sink and does not require any coordination measurement equipments or position message exchange.

3.4.1 Route Discovery

Here we explain our proposed Clustered Multipath Routing Protocol, CMRP, and then we demonstrate the scheduling algorithm in the next section. We are using two performance metrics: hop-count (as indication for distance from the sink and delay), and received signal strength (combined with SNR & BER) as indication for link quality (interference and noise level) and distance from the sender. Two thresholds (upper, and lower) are used in CMRP to compare the signal strength value of the received packets.

The selection of the values of the two thresholds is very critical in clustering the network and connecting them together. The upper threshold is used to determine the 1^{st}-level cluster heads and group member nodes (as described below). The upper threshold should be adjusted in a way that: it should not be very large value (close to the max value) so that you will not find any node receives your message in this power level or only a few nodes. In this case the cluster size will be very small with many chances of having only singleton clusters, and the load will be high on few 1^{st}-level cluster heads for serving many paths passing through them. Also if the upper threshold is low (below the mid value close to the lower threshold), the cluster size will be very high and cause cluster heads to overload with many group members and suffer high interferences in both inside clusters and at the sink side. The lower threshold is used to establish the links between cluster heads. Having a relatively high value of the lower threshold (close to the mid value) may prevent connecting the cluster heads in different levels and this leads to have a weak network connectivity. Also if the lower threshold is very low (close to the min value), then the network can have low link quality links between cluster heads with high possibility of packet drops.

In the initializing phase of CMRP, the base station starts sending periodic broadcast messages, called *BS-Msg*, to the surrounding powerful nodes. *BS-Msg* contains the identification number (ID) of the base station and the relevant security information to authenticate the communication with other nodes (if any). The nodes that receive *BS-Msg* messages compare the received signal strength index (RSSI) with the upper threshold (*Thr-High*). If RSSI is greater than *Thr-High*, these nodes respond to the base station by sending back acknowledgment messages informing their joining the base station as their parent. Then, they start acting as 1^{st}-level cluster heads (1^{st}-CH) -as shown in an example in Figure 3.3- and broadcast periodically control messages called *CH-Msg* to their neighboring nodes. *CH-Msg* contains the ID of the cluster head, number of hops between the cluster head and the base station in the current found path, IDs of the nodes joining this path up to the current cluster head, and the relevant security information (if any). For each *CH-Msg* received by the surrounding nodes of the 1^{st}-level CHs, RSSI is measured and compared with two thresholds, *Thr-High* and *Thr-Low*.

If the signal strength of the received *CH-Msg* is greater than Thr-High, the receiving node will start behaving as a group member and sending back an acknowledgment message informing its joining to the corresponding cluster head. Receiving a *CH-Msg* with RSSI greater than *Thr-High* indicates that the sender (CH) is in near region, as seen from eq 3.2, and the quality of the link is good and thus this CH can better serve the communication toward the base station. In case a node receives more than one message from different CHs with RSSI larger than *Thr-High*, the node selects the cluster head of the highest signal strength value, as shown in pseudo-code of CMRP in Figure 3.4.

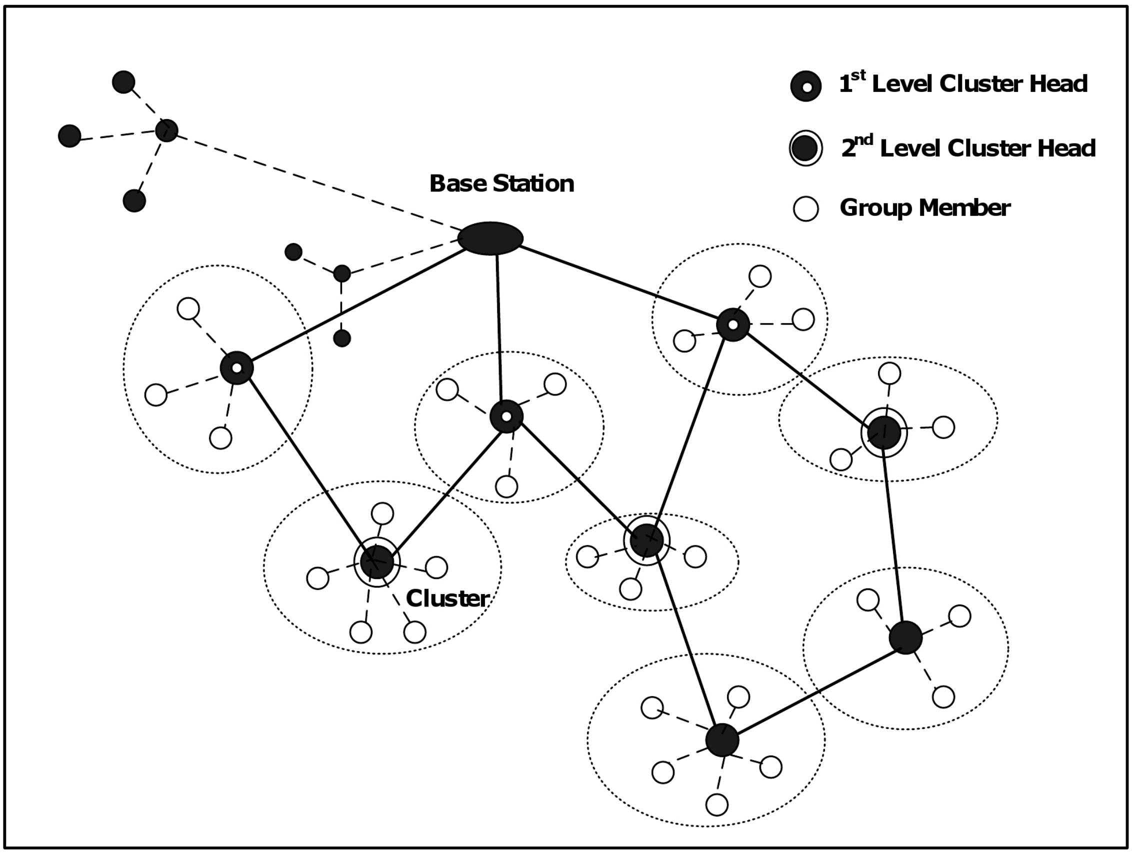


Figure 3.3: A Simple Example of Cluster-based Multipath Network.

The powerful nodes that only receive messages with received signal strength between *Thr-High* and *Thr-Low* will start acting as new cluster heads, in this case 2^{nd}-level cluster heads, and respond back to the sender informing their selection of him as one of their possible parents toward the base station. New cluster heads may receive different *CH-Msgs* from previous-level cluster heads. In this case new cluster heads consider these messages in order to construct multiple paths toward base station and sort these paths based on certain criteria (such as link quality, end-to-end delay, bandwidth, or number of hops in the path). Paths with good conditions, like high link quality, short end-to-end delay, enough bandwidth, or less number of hops, are reserved for multimedia communication that requires certain level of quality of service requirements. Other paths will be used for other types of data that does not require strict QoS requirements such as scalar data. If the RSSI is less than *Thr-Low*, the mes-

```
1.  Broadcasting authenticated BS-Msg periodically
2.  For each node i receives BS-Msg → Calculate RSSI
     if RSSI > Thr-High
       act as (1st-level) CH
       send authenticated CH-Msg periodically
     else
       ignore BS-Msg
     End if
   End for
3.  For each node i receives CH-Msg → Calculate RSSI
     if RSSI > Thr-High
       act as GM
       add sender to CH-table
       start selectMyCH_timer
     else
     if RSSI > Thr-Low
       act as CH
       add sender to parent_table & add path to available_paths_table
       start selectMyParent_timer
     else
        ignore CH-Msg
     End if
    End for
  4.  After selectMyCH_timer or selectMyParent_timer expired:
    For each CH → select parent from parent_table &
                  select corresponding path from available_paths_table
                  suitable for each traffic class
    For each GM → select CH from CH-table with max RSSI
```

Figure 3.4: Pseudo-code of the Main Part of the Routing Protocol.

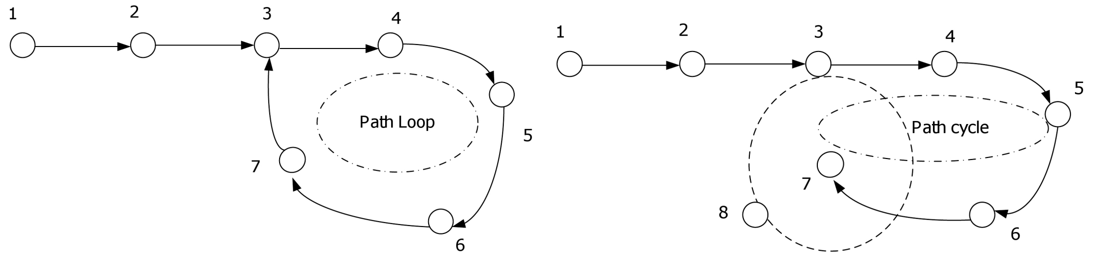
sage is considered as lost or ignored. This process continues in the same manner to build the network until all nodes join the network and determine their rules, *i.e.* cluster head or group member, and all possible paths are found.

In case that cluster heads do not receive any message from other nodes informing joining them as group members (*i.e.* singleton clusters), then these cluster heads will behave either as: 1) Forwarder nodes to relay data toward the sink, if they receive messages from other powerful nodes informing their joining as next-level cluster heads, or 2) temporarily normal nodes, group members, and join any other closer clusters based on RSSI. These nodes can create later on their own clusters when new group members nodes are deployed in their vicinity.

After the network is established and all possible routes are found, base station and cluster heads will reduce the rate of sending broadcast control messages (*BS-Msg*, *CH-Msg*) in order to save channel capacity and energy. We keep sending these broadcast messages even with lower rate, which has a negligible effect on the network performance, for the sake of adding new nodes to the network.

3.4.2 Route Optimization and Local Repair

In order to optimize the found routes in route discovery phase, path loops and path cycles should be prevented. For *path loops*, each CH that receives *CH-Msg* from other nodes checks first the IDs of the nodes joining the path to know whether it already joined this path before or not as shown in Figure 3.5(a). If a CH receives *CH-Msg* belongs to one of the paths already found before, it checks the conditions and the status of the given path in order to update its routing information about this path and reflects these changes (if any) on its decision of selecting the proper path for each type of data. Moreover, for path optimization with minimum number of hops, a CH checks for every given path whether it is a child for any participating node in this path (except of course its direct parent in this path). In this case, it is better for the CH to communicate directly to that parent instead of making a path cycle as shown in an example in Figure 3.5(b). Thus, if a *path cycle* is found, the cluster head deletes this path from its routing information (as it is just a longer version of already found path) and keeps the



(a) A Path Loop **(b)** A Path Cycle

Figure 3.5: Examples of Path loop and Path Cycle.

shorter path.

The acknowledgment system is critical for WMSNs to achieve a low frame loss rate that affects the quality of video perception, and to detect any node or link-failure. After receiving a certain number of data packets, a CH sends an acknowledgment message (*Ack-Msg*) to the sender (lower-level CH or GM) and in the same manner waits an *Ack-Msg* from its parent (higher-level CH or sink) confirming receiving the data packets. So, if a node did not receive an *Ack-Msg* from its parent, it will assume that there is a node-failure or link-failure and it will select another parent (*i.e.* another path) -depending on its routing information tables - suitable with the current type of data. There is no need to initialize the entire network for establishing the routes again in order to overcome the existing failure; it just affects the nodes along the failed path and because of that it is called *local repair*. If the parent is the sink and there is no response from its side, then the node should communicate with other reachable 1^{st}-level CHs, based on RSSI, to deliver data packets through them. If this node cannot communicate with any 1^{st}-level CH, then it should send negative *Ack-Msg* to its children nodes (lower-level CHs and GMs) informing about this link-failure. Then children nodes will have to select another parent CH according to their routing information table.

The same procedure is used with GMs to check their CH: After sending a certain number of data packets to, a GM waits an *Ack-Msg* from its CH confirming receiving those packets. If the GM did not receive *Ack-Msg*, then it assumes that there is a link-failure or node-failure and joins another CH based on its routing information table.

3.4.3 CMRP Life Time Analysis

In this subsection, we are interested in analyzing the effect of multipath routing in the expected life time of a link between a CH and BS. According to CMRP, a link (P) consists of multiple paths where each path (p_i) contains certain number of intermediate CHs (n). A path will be broken once the battery energy (E_i) of any intermediate node residing on it depleted. E_i is independent random variable distributed uniformly between *0* and *Emax* (full battery energy), and for simplification we can express: $C_i = E_i/Emax$ where C_i now is an independent random variable uniformly distributed between *0* and *1*. Then, we can define the following parameters:

$$Path\,life\,time: \quad p_i = min(C_1, C_2, ..., C_{n_i}) \tag{3.5}$$

$$Link\,life\,time: \quad P = \sum_{i=1}^{N}(p_i) \tag{3.6}$$

Where n is the number of nodes in the path i and N is the number of paths in the link P.
Then the expected (average) life time of the link (P) is:

$$\mathbb{E}\{P\} = \sum_{i=1}^{N}\mathbb{E}(p_i) \tag{3.7}$$

Since all node energy indexes (C_i) are random variables uniformly distributed between *0* and *1*, then the minimum random variable along one path p_i follows a Beta distribution with parameters *1* and n. The probability density function of Beta distribution is:

$$f(x;\alpha,\beta) = \frac{x^{\alpha-1}\cdot(1-x)^{\beta-1}}{B(\alpha,\beta)}$$

, where $B(\alpha,\beta)$) is the Beta function: $B(\alpha,\beta) = \int_0^1 t^{\alpha-1}\cdot(1-t)^{\beta-1}dt$
Then substituting $\alpha = 1$ and $\beta = n$ gives us the probability density function of the path life time:

$$\mathbb{P}\{p_i = x\} = n.(1-x)^{n-1} \tag{3.8}$$

As the mean value of the Beta distribution is: $\{X\} = \frac{\alpha}{\alpha+\beta}$, then the expected path life time of this probability function can be expressed as:

$$\mathbb{E}\{p_i\} = \frac{1}{n+1} \tag{3.9}$$

Finally, the average link life time can be calculated as:

$$\mathbb{E}\{P\} = \frac{N}{n+1} \tag{3.10}$$

3.5 Two-level QoS-aware Scheduling

After establishing the network, all group members (sensor nodes) in each cluster are assigned to a cluster head and each cluster head in the network knows now its parents toward the BS (for multiple paths). Before data transmission, we introduce two-level QoS-aware scheduling: *low-level scheduling* within each cluster among the group members, and *High-level scheduling* among the cluster heads at higher level in order to increase the packet delivery ratio and throughput for multimedia data. The two-level scheduling is shown in a simple example in Figure 3.6.

Besides their low energy efficiency, most contention-based protocols are generally not designed for sending real-time multimedia data and are not suitable for delay-sensitive WMSNs because each node has to contend for medium access to send every packet, thus the delay for data delivery could be potentially unbounded. The needed time required to resolve collision is based on the load condition of the network and number of nodes in clusters, which makes it very difficult to guarantee a bounded delay. Therefore, we prefer to adopt TDMA protocol to access the channel as it has a natural advantage of collision-free medium access, and it is more appropriate for transmitting multimedia applications with QoS at reliable channel conditions and heavier traffic load. In order to avoid channel under-utilization and to decrease the delay, dynamic time-slot is assigned to the nodes depending on the amount of data to be transmitted and the time for sending *Ack-Msgs* if needed.

At low level, before GMs start sending their different types of data to their CHs, each CH should schedule the data transmissions among its GMs within the cluster in

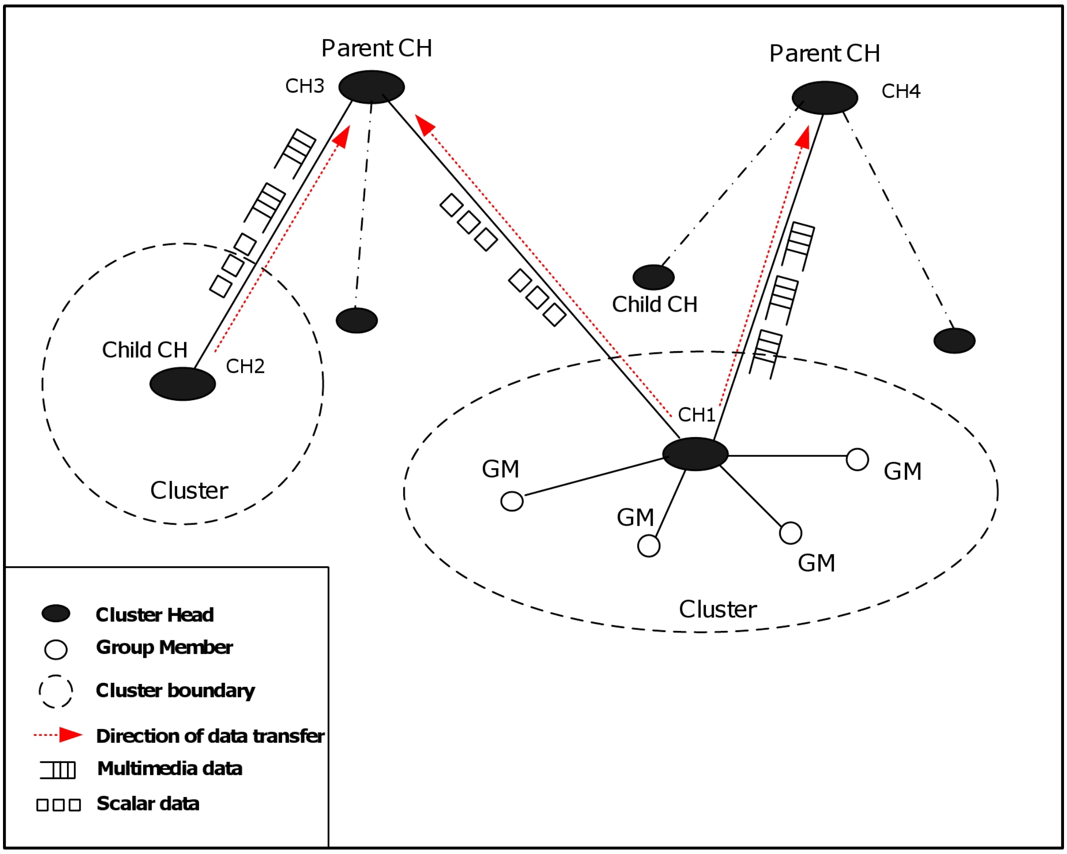


Figure 3.6: A Simple Example of Two-level Scheduling.

order to give higher priority to the nodes that demand higher or strict QoS requirements for their data, and to avoid collisions and interferences at receiver side. The low-level scheduling process is initiated by the CH by sending a broadcast message asking each GM in the cluster to send a request message (*Req-Msg*) informing about the type of data to be transmitted, its amount, and its requirements (such as playout deadline, BW... etc). This broadcast message, *Assign-Msg*, contains the control slot assignment, based on TDMA (*i.e.*, time slot to each GM) in the cluster. The duration of the time slot is enough to any node in the cluster to send its *Req-Msg* and the time slot is unique for each node to avoid collisions. During the request phase, each GM sends a *Req-Msg* to its CH at the allocated time slot informing about the data to be sent (if available) and its QoS requirements.

Then based on the collected information from the request phase, each CH generates a transmission schedule for the active GMs and distributes it in the cluster. The

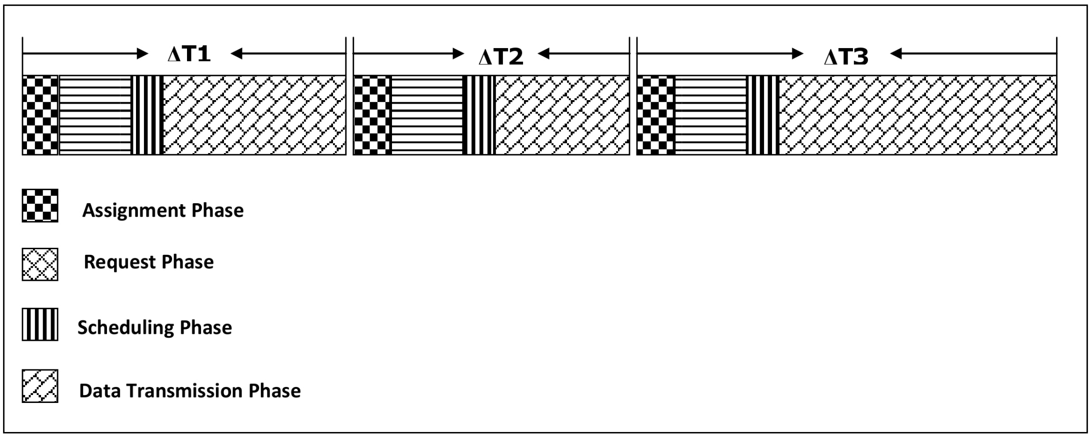
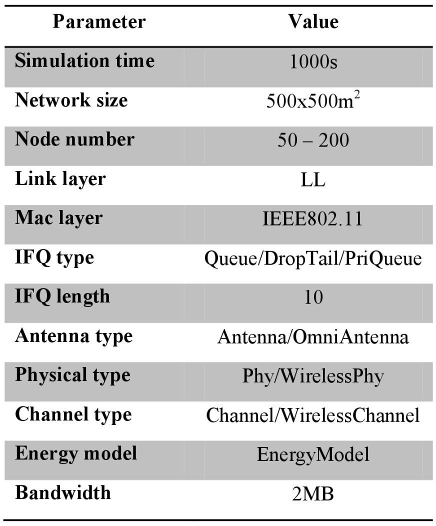


Figure 3.7: A Cluster Time Intervals for Scheduling Process.

resulting schedule is sent to all GMs by broadcasting a scheduling message, called *Sched-Msg*, to inform each GM with the specified time schedule for sending each type of data. The duration of the time-slot depends on the amount and type of data to be transmitted as requested by each GM.

By this way, multimedia streaming and time-critical data can be transmitted first, then less priority data such as still images and then scalar data can be sent later. Moreover, for better energy efficiency, GMs can turn off their radio transceiver when the schedule has been received until the time slot for transmission a certain data type approaches or to the end of the data transmission phase if they are passive nodes. After receiving the schedule, each GM will transmit its data during the assigned time slots for each data type and the CH sends, after receiving a certain number of data packets, an *Ack-Msg* to the sender as described before. When the data transmission phase complete, a CH sends again the *Assign-Msg* to its GMs to send their requests. The time-intervals of the scheduling operation in a cluster are shown in Figure 3.7.

At higher level, each intermediate cluster head -in the same manner done at low level- schedules the traffic toward the sink from other cluster heads (its children) based on the type of the data and its QoS requirements. For example, as shown in Figure 3.6, CH1 selects the path through CH4 to send its streaming multimedia data and the path through CH3 to send the other types of data based on the proposed routing algorithm,



Parameter	Value
Simulation time	1000s
Network size	500x500m^2
Node number	50 – 200
Link layer	LL
Mac layer	IEEE802.11
IFQ type	Queue/DropTail/PriQueue
IFQ length	10
Antenna type	Antenna/OmniAntenna
Physical type	Phy/WirelessPhy
Channel type	Channel/WirelessChannel
Energy model	EnergyModel
Bandwidth	2MB

Table 3.2: Simulation Parameters.

while CH2 sends all its data to CH3. CH3 and CH4 then need to schedule the transmission from children CHs following the same steps done at low level inside the cluster.

3.6 Performance Evaluation

In order to evaluate the performance of *CMRP*, several simulation experiments (over 100) with various random topologies were run. We implemented *CMRP* using NS-2 version 2.34. NS-2 [123] is an open source, discrete event simulator which is widely used for research purposes. It has an excellent implementation of the IEEE802.11 standards at physical, Data link and higher layers. We simulate the proposed routing protocol assuming a multi-hop network of size 500m x 500m deployed with different number of sensor nodes ranging from 50 to 200 in randomized grid. The sink is located in the center of the network. The traffic is CBR of 600 packets/sec and the packet size is 316 bytes. Table 3.2 shows our simulation environment and other parameters used in our simulation, and Table 3.3 lists the features of "Salvat Cluster" that we used to run our simulations.

In the simulations, we focus in measuring the performance metrics after the network has set up to exclude the communication overhead of the most exchanged control

Features	Salvat Cluster
* USP 73 nodes Xeon Dual-Core 5148	
* Motherboard Intel S5000VCL	
* Intel SR1530 chassis	
* Intel 5000V	
* 2 Dual-Core Intel Xeon 2.333 GHz, 1333MHz FSB, 4MB Cache	
* 12 GB RAM in 6 modules of 2 GB	
* Hard Drive Seagate Barracuda 320 GB S-ATA-2	
* 2 Intel PRO/1000 Gigabit Ethernet network cards	

Table 3.3: Features of Salvat Cluster and its Picture.

messages. Control messages include broadcast messages (*BS-Msg*, *CH-Msg*) sent at very low rate and acknowledgment messages (*Ack-Msg*) used for data receiving notification and local repair. However, these control messages are considered in measuring the energy consumption during simulation time.

3.6.1 CMRP Performance Evaluation

In this subsection, we compare performance evaluation results (average) of *CMRP* with DHCT [124], MCRA [119], and EDGE [125] protocols. We consider in this comparison four important performance metrics: throughput, end-to-end delay, packet delivery ratio, and power consumption:

End-to end delay is one of the important QoS parameter that we consider in designing our proposed routing protocol to handle the real-time traffic and deliver the packets within their playout deadlines. End-to-end delay is the time difference from the time

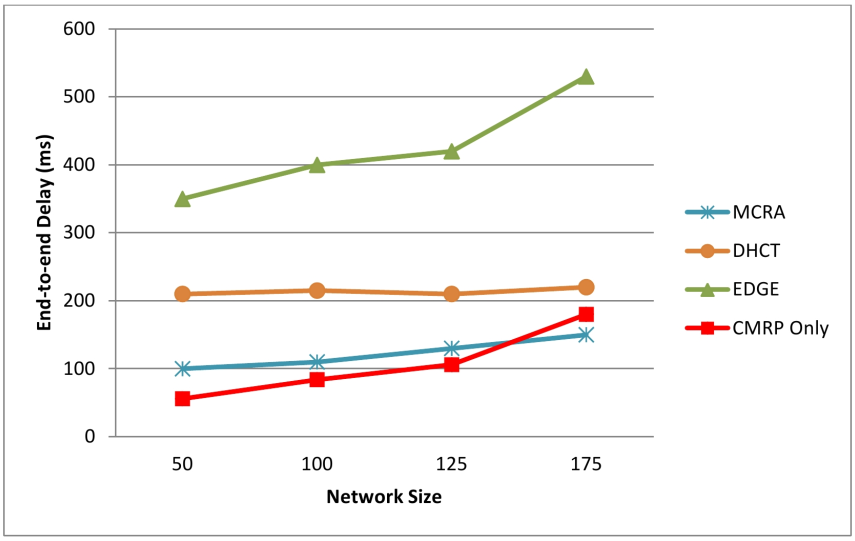


Figure 3.8: End-to-End Delay of Our Protocol Compared with Other Protocols.

a source node sends its data packet to the time the sink receives it, and it can be measured as sum of (transmission delay, propagation delay, queuing delay, and processing delay at each hop). We obtain an average end-to-end delay of 75 ms which satisfies the end-to-end delay requirements of real-time multimedia packets. Figure 3.8 shows a comparison between our proposed protocol, *CMRP*, with the other protocols (DHCT, MCRA, and EDGE) in terms of average end-to-end delay with different node number. It can be seen clearly that our protocol outperform the other protocols at low node density due to the hierarchical architecture of powerful cluster heads that always select the route with lowest number of hops of better link quality and hence minimum delay. However, at higher network size, we notice that *CMRP* encounters high interferences and collisions inside clusters (then the need of retransmission) and because of that we see it has less performance than MCRA.

In Figure 3.9, the mean throughput of our protocol is shown compared with the other protocols. The throughput is measured as the total number of packet received at the sink over the simulation period. Selecting of multiple paths of better link quality and minimum delay leads to load balancing and efficient utilization of the wireless spectrum, and hence achieves higher throughput and much better performance than other protocols at low node density as seen in Figure 3.9. And at higher node density,

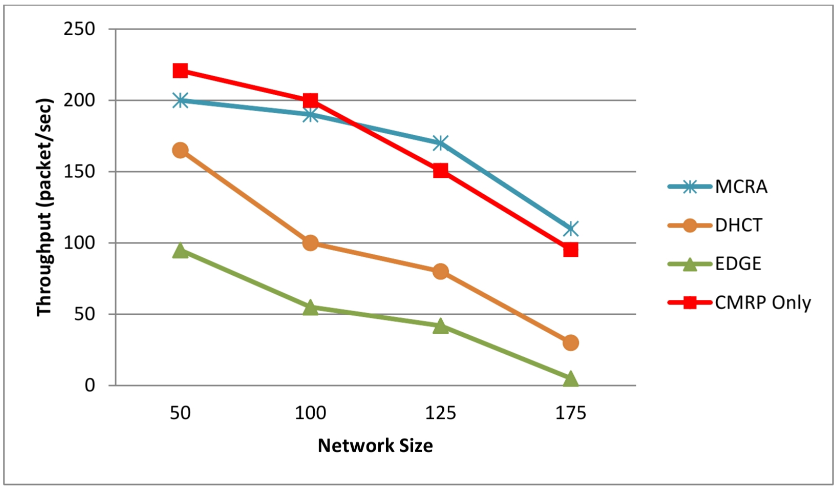


Figure 3.9: Throughput of Our Protocol Compared with Other Protocols.

we can see that *CMRP* has a bit worse performance than MCRA, but still better than other protocols.

Figure 3.10 shows the average packet delivery ratio (PDR) of our protocol with different node number. PDR is measured as the total number of data packets received at the sink over the total number of data packets sent by all sources in the network. It is shown that *CMRP* outperforms the other protocols, which confirms the previous result for low network size, due to the use of multiple paths that are constructed with better link quality based on the received signal strength (along with SNR and BER). Also, the use of the fast mechanism of local repair through the acknowledgment system minimizes the effects of any node failure or link break and hence decreases the number of lost packets.

Average energy consumption is shown in Figure 3.11 where we can realize that our proposed protocol *CMRP* has less energy dissipation comparing with other protocols (DHCT, MCRA, and EDGE) at low network node numbers. This result due to the fact that most of the nodes in the clustered network are GMs and need only to communicate with their CHs regardless of the number of nodes in the cluster. Also, the paths found by *CMRP* are optimized in terms of number of hops since the routing algorithm depends on the hop-count as one of its metrics along with preventing path loops and

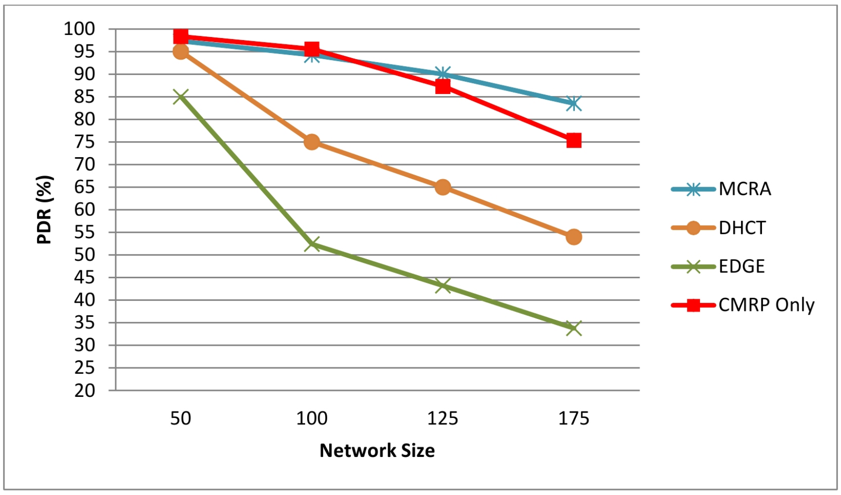


Figure 3.10: PDR of Our Protocol Compared with Other Protocols.

path cycles, which lead to minimum number of packet forwarding from a source to the sink. Moreover, the possibility of having aggregation process and data fusion at CHs reduce the size of correlated data within a cluster and thus decrease the needed amount of energy to deliver them.

In order to improve the performance of our proposed routing protocol, especially at high network size, by reducing the interferences and collisions inside clusters and eliminating their effect on the average network performance, we introduced our proposed cross-layer design solution for better network scheduling. The results from using this optimized solution are shown in next subsection.

3.6.2 Cross-layer Routing Performance Evaluation

In this subsection, we compare the average end-to-end delay of our proposed routing protocol combined with the two-level scheduling technique (*CMRP+2_level scheduling*) with the proposed routing protocol only (CMRP-only) and the other protocols (DHCT and MCRA). We select these protocols to compare with to show how our proposed cross-layer design methodology will outperform the other recent protocols that are based on the classical layered structure of the communication stack. For simulation environment, we use the same parameters mentioned in Section 3.6, except that we use our proposed scheduling based on TDMA at the MAC layer for *CMRP+2_level*

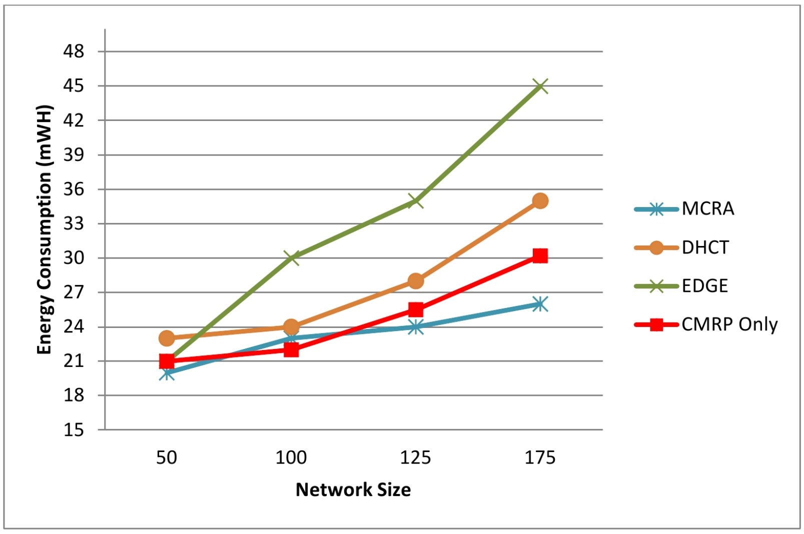


Figure 3.11: Energy Consumption of Our Protocol Compared with Other Protocols.

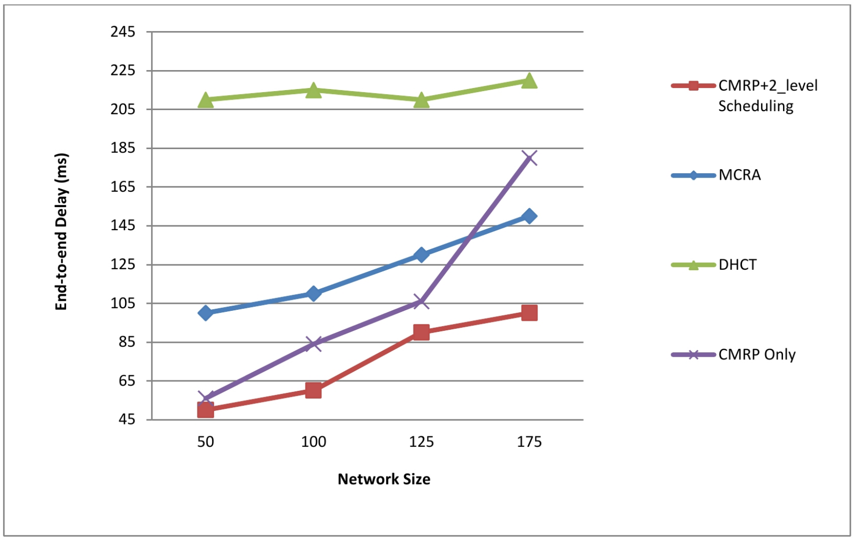


Figure 3.12: End-to-End Delay Performance of Our Cross-layer Routing Protocol.

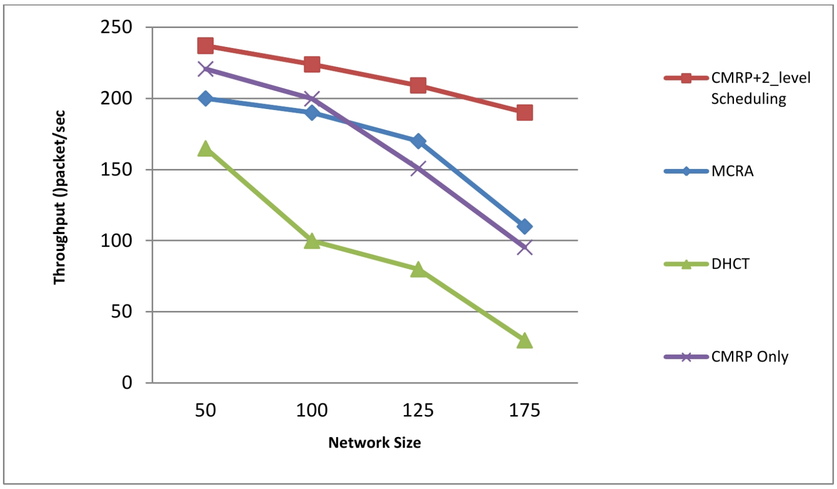


Figure 3.13: Throughput Performance of Our Cross-layer Routing Protocol.

scheduling.

Figure 3.12 shows the end-to-end delay, which is one of the important QoS parameters as the real-time multimedia packets have strict playout deadlines. It is shown clearly that our cross-layer design has the minimum end-to-end delay and outperforms the other protocols because it depends on selecting the path of better link quality and minimum hop-count through powerful cluster heads. Notice that CMRP-only performs well at low node density, but with dense deployment end-to-end delay increases significantly due to the interferences and collisions within the clusters and among cluster heads which cause to retransmission lost packets again.

Our proposed cross-layer protocol achieves higher throughput, as shown in Figure 3.13, than the other protocols by efficiently utilizing the wireless spectrum and distributing the load via adopting adaptive TDMA-based scheduling and selecting multiple paths of better link quality and minimum delay respectively. Without implementing the scheduling scheme, we notice that CMRP-only's performance is degrading with increasing the number of nodes due to the time wasted for retransmitting lost packets and changing paths to overcome the interferences and collisions and hence lead to decrease number of received packets at the sink during the simulation time.

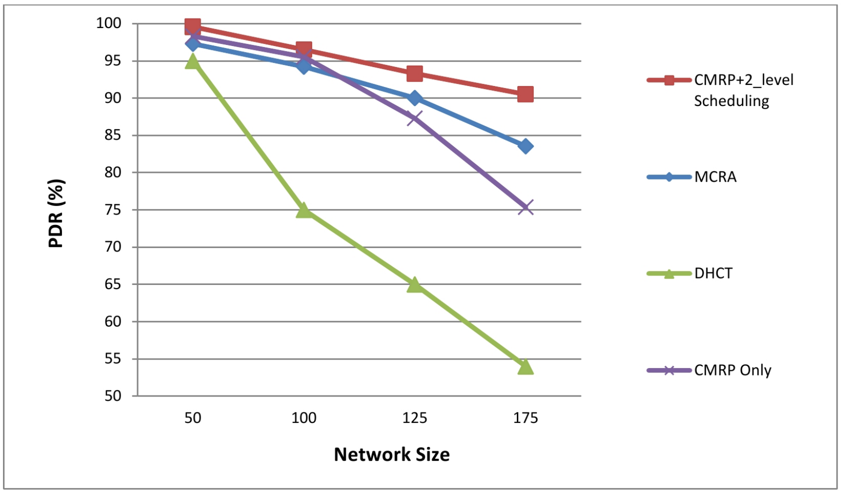


Figure 3.14: PDR of Our Cross-layer Routing Protocol Compared with Other Protocols.

Average packet delivery ratio (PDR) is shown in Figure 3.14 where our proposed cross-layer design outperforms the other protocols. We obtain this result due to the use of the two-level scheduling that prevents collisions and minimizes interference, besides the selection of paths with better link quality based on the received signal strength (along with SNR and BER). Also, as mentioned before, the use of the fast mechanism of local repair through the acknowledgment system minimizes the effects of any node failure or link break and hence decreases the number of lost packets.

With respect to average energy consumption, our proposed design has less energy consumption than the other protocols as shown in Figure 3.15 with different node numbers. Both CMRP-only and CMRP+2_level scheduling protocols have good energy efficiency at low node density because of the many benefits -which we mentioned before- from the clustered network architecture. However at higher node densities, we notice that CMRP-only suffers from packet collisions and interferences and consumes more energy for retransmitting lost packets, while our cross-layer design exploits the benefits from the adaptive two-level scheduling to prevent such problem and hence has less energy consumption.

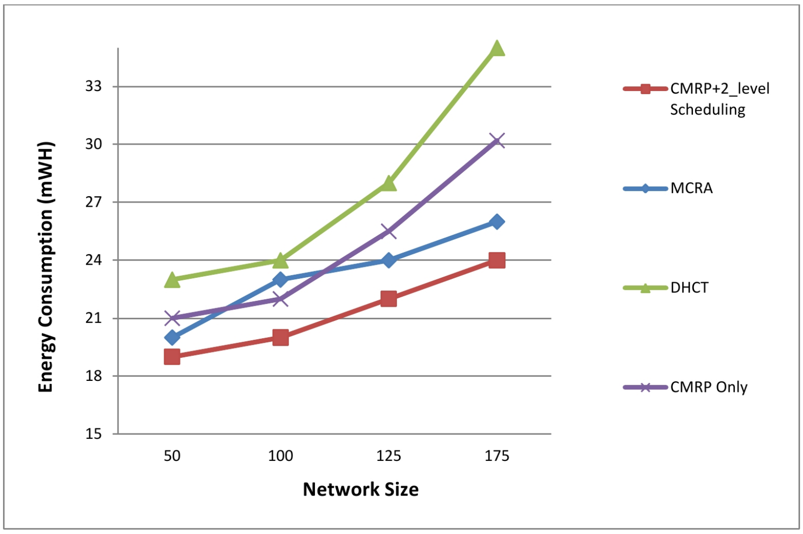


Figure 3.15: Energy Consumption of Our Cross-Layer Routing Protocol and Other Protocols.

3.7 Conclusions

In this chapter, we proposed a Cluster-based Multipath Routing protocol (CMRP) for WMSNs designed to handle the additional requirements of reliable data delivering of different traffic classes and provide load balancing by using multipath routing. The proposed routing protocol, CMRP, is based on the hierarchical structure of multiple paths established depending on the hop count and received signal strength as an indication on the link quality, delay, and distance between the nodes. CMRP maintains minimum end-to end delay suitable for real-time and non-real-time data packets to meet their playout deadline, and achieves high throughput and packet delivery ratio by selecting the paths with better link quality and avoiding collisions and interferences. CMRP reduces energy consumption at sensor nodes by moving the multimedia processing complexity as well as the aggregation process to the cluster heads side along with preventing path loops and path cycles in establishing the routes.

Then, we presented a cross-layer communication architecture for WMSNs between the routing and MAC layers, where CMRP routing protocol has been pursued in conjunction with an adaptive QoS-aware scheduling to maximize the overall network performance with minimum energy consumption, reliable delivery, and efficient resource management. Our design aims to exploit correlation characteristics and functionalities between the two layers to maximize the overall network performance with minimum energy consumption in order to handle the additional requirements of delivering reliable multimedia data. Our proposed scheduling protocol is based on TDMA approach with flexible time-slot assignment that adaptively assigns slots to various traffics from active nodes.

Performance evaluation results show that CMRP-routing-only clearly outperforms the preexisting ones (DHCT, MCRA, EDGE) in all average end-to-end delay, throughput, packet delivery ratio and battery power consumption. Also simulation results demonstrate that our cross-layer design can improve the performance of CMRP-routing-only and achieve better than other protocols in terms of average end-to-end delay, throughput, packet delivery ratio and battery power consumption.

In order to secure the wireless communication among the nodes in WMSN, we develop in next chapter a light-weight key management scheme for secure routing and intrusion detection system to eliminate the threats from outsider and insider attacks.

Chapter 4

Security Schemes of Key Management and Intrusion Detection for Clustered WMSNs

4.1 Introduction

Many applications of WMSNs have special requirements in terms of privacy and security [113] such as military applications, medical care applications, and video surveillance systems. In addition to the fact that sensor networks are more vulnerable to attacks than wired networks due to the fact that in a broadcast medium, adversaries can easily eavesdrop on, intercept, inject, and alter transmitted data [126]. Therefore to meet these requirements and challenges, security mechanisms -such as encryption and authentication- are essential to secure communications over WMSNs with minimal impact on overall performance through balancing their security features against the communication and computational overhead required to implement them [127]. Also, these security mechanisms should be scalable with the large number of sensor nodes and cope with the hostile environment of the deployed network.

In general, there are many types of security attacks on sensor networks such as: jamming, tampering, altered routing information, sinkhole, Sybil, wormholes, ac-

[1] Chapter 4 is based on the following works:
(1) *A Secure Cluster-Based Multipath Routing Protocol for WMSNs; Islam T. Almalkawi, Manel Guerrero Zapata, Gamal N. Al-karaki* appeared in *Sensors Journal 11, Volume 4, Pages 4401-4424, 2011*
(2) *Light-weight Security Scheme for Key Management and Intrusion Detection in Clustered Wireless Multimedia Sensor Networks; Islam T. Almalkawi, Manel Guerrero Zapata, Gamal N. Al-karaki* submitted to *Journal of Networks and Computer Applications (JNCA), March 2013*

knowledgment spoofing, and hello-flood attacks [128]. Attacks can be classified into outsider attacks and insider attacks [128]. Outsider attacks are where the attacker has no special access to the sensor network, and the insider attacks are the ones in which the attacker is an authorized participant in the sensor network. Insider attacks can be either compromised sensor nodes running malicious code or external devices that use stolen key material and data from legitimate nodes to attack the network.

Key management is the core of the security mechanisms in sensor networks since most existing cryptographic security algorithms depend on the security of the cryptographic keys and hence the distribution and management of keys has a vital importance [129; 130]. Key management protocols need to be efficient in terms of processing, storage, and communication requirements along with satisfying the basic security requirements. Key management can be based either on public key cryptography or symmetric encryption/decryption algorithms. Public key algorithms (such as RSA, Diffie-Hellman, and ECC [131; 132]) require smaller number of necessary public/private keys in large networks and these public keys do not need to be changed frequently to keep them secure. However, Public key algorithms are computationally expensive, have slower throughput rate, and use larger key sizes, which make them not suitable for the hardware capabilities of typical sensor nodes. On the other side, symmetric key ciphers (such as DES and AES [133; 134]) are between two to four orders of magnitude faster [135] and have high rates of data throughput and relatively short size keys, which make them suitable for WMSNs and more desirable. However, symmetric key algorithms need efficient key management protocols to find a way to reduce the required number of private keys in order to support large scale networks.

Symmetric encryption schemes used in sensor networks can be based on block ciphers (such as AES, skipjack, and MISTY1 [133]) or stream ciphers (such as RC4 and Salsa20 [136]). Stream ciphers typically execute at a higher speed than block ciphers and have lower hardware complexity. For instance, the RC4 algorithm takes 86 μs for encryption on the ATmega128 (16 MHz) processor and only takes 5 μs on the XScale (400 MHz) processor [137], while AES algorithm takes about 1.8 ms to encrypt a 128-bit block of data on Atmega128 processor [138]. However, stream ciphers have not gained widespread confidence in their security strengths. On the other hand, block ciphers can be used for both secure modes required for sensor networks: pairwise secure links and secure group communications, and they can offer code size optimization in case of using them for both encryption and authentication operations. Also it has been shown that a light-weight energy efficient security scheme can be based on block cipher scheme providing a sufficient level of security for sensor networks [133; 139]. In addition, block cipher schemes operating in a streaming mode (such as CTR mode) can encrypt streaming data with different input sizes which leads to simplify the hardware complexity while keeping the transmission efficiency.

Key management and secure routing allow the nodes to use security mechanisms that protect their transmitted information from being exposed by an unauthorized party and guarantee the integrity of their data. Although this level of security protects the network from the outsider attacks, still it cannot satisfy the security goals against most of internal attacks. Therefore, a second level of security is required: An Intrusion Detection System (IDS) [140; 141] that can detect malicious attempts of exploiting possible security breaches and warn for suspicious nodes, even if these nodes are using legitimate security keys. In general, there are three main techniques used by intrusion detection systems [142]: signature based detection, anomaly based detection, and specification based detection. Signature based detection systems compare the observed behavior with known attack patterns (signatures) to detect intrusions. Predefined threat action patterns need to be stored in the system which, for some simple nodes might lead to have storage problem [143]. Signature based detection techniques (also known as misuse detection techniques) may not be effective to detect new attacks in case of the lack of corresponding signatures. On the opposite, anomaly based detection systems [144] focus on normal behaviors, rather than suspicious behaviors, by matching the observed behavior with pre-established normal profiles (usually established by machine learning or training) to identify the abnormal behaviors. Anomaly detection can detect unknown attacks. However, normal profiles are usually very difficult to build, and it is more prone to have false positives. Specification based detection techniques [145] combine the advantages of signature based detection and anomaly detection by using manually developed specifications that describe the correct operation and compare the observed behavior with these specifications. However, the downside is that the development of detailed specifications can be complex and time-consuming.

IDS systems can be also classified into three categories based on network structure: A standalone IDS, cooperative/distributed IDS, and hierarchical IDS. In the standalone

IDS [146; 147], all nodes have equipped with IDS agent and the nodes detect intrusions by themselves using these agents without exchanging any information with each other. In the cooperative/distributed IDS [148; 149], each node runs its IDS agent and cooperates with its neighbors to detect intrusions. In the hierarchical IDS [150; 151], the network is divided into clusters and each node communicate with its parent(s) in a hierarchical way to detect intrusions.

In this chapter, we illustrate the implementation of a distributed and light-weight security protocol in terms of energy efficiency, processing and memory complexity, and communication overhead in order to secure the communication among the nodes in clustered WMSNs. We explain our proposed security key management scheme in details. Also, we analyze analytically the effect of clustering the network on the scalability of our algorithm and the number of needed security keys stored in each node. Moreover, we describe our proposed lightweight intrusion detection technique to protect clustered WMSNs from internal attacks. To the best of our knowledge, there is no intrusion detection systems specific for WMSNs in the literature. Due to the especial requirements for delivering real time multimedia data, it is required significantly to have an efficient (fast and accurate) and light-weight (minimum overhead) IDS to detect possible intrusions in WMSNs. However, the previous ID schemes designed for scalar WSNs are not very suitable for such cases and have several problems to be applied to WMSNs, as we show in Related Work section. Moreover, we measure the overhead introduced by the security scheme in scenarios where nodes send the sensed information to the sink in cluster-based architecture.

4.2 Related Work

Existing security mechanisms, proposed protocols in authentication and encryption, secure routing, key management and distribution, and intrusion detection systems for WSN in general were surveyed and discussed in [152; 153; 154; 155; 156]. Also, several proposals in the literature targeted the security implementations that are specific of WMSNs and some of them are surveyed in [157; 158].

A security scheme for collaborative image transmission from image sensors to cluster heads is proposed in [159]. The proposed protocol in that paper does not use a key-based security scheme, but it depends on a secret image sharing approach on multiple node-disjoint paths for image delivery. In this approach, the image is separated into overlapped and non-overlapped regions where the later is directly transmitted without encryption after compression, while the former is transmitted with distribution ratio via multiple paths. The proposed scheme also exploits the inter-image correlation among sensors sharing overlapped regions in order to not easily disclose the original subimage. However, in this scheme, misjudgments may frequent happen if many unauthorized nodes (attackers) join the network since there is no applied security schemes for sharing security keys or intrusion detection system. Also this scheme is not distributed in building the decisions and depends only on the sink node.

Another non key-based security scheme is presented in [160] where a wavelet-based watermarking technique is used to ensure authentication and data integrity for real time image delivery. This technique embeds additional data called a watermark into some location in an image object so it can be detected later to make an affirmation about the object. The watermarking locations or positions are adaptively chosen by using two thresholds to insert the watermark according to network conditions so that the energy efficiency and security can be achieved. The proposed scheme embeds the watermark into as few positions as possible to make it invisible and allocates extra network resources to protect this embedded watermark from high distortion. In addition, it also embeds watermark coding redundancies into the original image so that the watermark becomes more robust to packet loss. However, watermarking based techniques do not satisfy all needed security requirements; they assure image data integrity and authentication but data is not safe from unauthorized access. Also if some of watermarking components are degraded due to packet loss during wireless transmissions, then the watermark information may not be extracted or verified. Extra network resources are needed to be allocated to protect their transmissions over WSNs.

In [161] a selective encryption approach is proposed for transmitting audio data. The proposed security scheme first differentiates the important audio data from less significant audio information by using modified discrete cosine transform (MDCT) transformation coefficient index. Then the MDCT-based audio samples are distributed into different packets according to their importance level. The important audio packets are encrypted using AES algorithm while additional network resources are allocated to protect them from packet losses. However, key generation and distribution for encryption in this approach are not explained and it is not addressed how the network can be protected against internal attacks.

Providing network privacy is discussed in [162] by introducing the concept of distributed visual secret sharing. In this scheme, the images collected by the sensor nodes are used to generate a large number of image copies with a large amount of random noise and distributed to different nodes of the network. The image copies are generated in such a way that if an attacker captures some of them, it will not be able to create any meaningful image. On the other hand, if the sink collects most of them, it will be able to almost perfectly recreate the original image. While this might open an interesting research field, the huge transmission overhead required by this scheme, makes very difficult to defend its feasibility in its current state.

A secure data converter architecture for WMSN that employs fingerprinting and encryption capabilities for simultaneously digitize and authenticate sensor readings is proposed in [163]. The proposed architecture suggested hardware modifications to the data converter and aims to reduce the computational complexity of the security algorithms at the aggregation points in the systems that need in-network processing. This can be done by embedding an authenticator payload into the data converter or the modulator output in a way that it is not easily extractable without access to the secret key, and can be used to verify the integrity of the sensor reading. However, this technique has several practical difficulties including modulator matching between the sender and receiver and the needed hardware modification.

Many papers address the problem of intrusion detection systems for WSNs. Most of these papers focus on general local detection, *i.e.* detecting attacks is done locally through collaboration among nodes at their neighborhood as in [144], while other proposals focus on detecting attacks against specific applications such as secure aggregation [164; 165; 166] and secure localization [167; 168].

A rule-based intrusion detection system for WSNs using monitor nodes is presented in [147]. The monitoring nodes are responsible for detecting attacks in their neighborhood. They listen to transmitted packets in their radio range that are not addressed to them, and detect packet collisions (if any) when they try to send packets. If the number of detected failures exceeds the expected number of occasional failures, an intrusion alarm is issued. However the promiscuous listening at packets intended for other nodes breaks the privacy of the other nodes especially if the packets contain sensitive information. Also, the monitoring nodes need to share the security keys with their neighbors to collect the required information if cryptography is used, which is not explained in the paper.

A similar ID system is proposed in [148] where two anomaly detection rules are based on the average receive power and average packet arrival rate. An intruder is identified whenever a significant deviation from the normal profile is detected. However, the IDS performance is dependent on the appropriate choice of the parameter values that defines what a significant deviation is.

A cluster-based IDS technique is used in [146] to detect routing attacks in sensor networks. It proposes 12 general features for detecting sinkholes and periodic route error attacks based on the AODV routing protocol. In this approach each sensor node monitors the routing messages it receives. The used clustering algorithm is called fixed-width clustering that creates circular clusters of fixed radius for a data set. The clusters are labeled normal if they contain more than a given fraction of the total data points, otherwise they are labeled anomalous. However, the paper assumes that routing protocols for ad hoc networks (AODV) can also be applied to WSNs, and it does not describe the collaboration among the nodes for making decisions in case of anomalies. Also, the nodes have to operate in promiscuous mode to listen the neighboring node which usually consumes more energy. A similar approach is proposed in [169] where the routing tables are used to detect anomaly behaviors. The paper uses DSDV and DSR ad hoc routing protocols for WSNs. However, since DSDV is a proactive routing protocol, this is arguably rendering the proposal unfeasible for large sensor networks.

In [150] a distributed clustering-based anomaly detection technique is proposed for sensor networks where fixed-width clustering is used. In this approach, each node sends only summary statistics of its own local data to their parents, which requires less communication overhead. The parent nodes receive the cluster information from their children, combine any overlapping clusters, and send the combined set of merged clusters to their parents as well. This process continues up to the sink where the anomaly detection algorithm is applied to the collected clusters, using the average inter-cluster distance of the K nearest neighbor (KNN) clusters, to identify each one as either normal or anomalous. Based on this technique, the detection rate using KNN approach depends on the cluster size, so it can give a high detection rate if an appropriate cluster size is chosen. However, determining the proper size of the cluster is not explained in that paper and might not be easy. Also, this approach depends only on the sink to apply the intrusion detection algorithm which may lead to have a long time for detecting the anomalies as the sink can be far away from the source of the anomalies, and may lead to lose important information during this process making it difficult to detect the intruders.

Most of the existing security protocols in WMSNs -as surveyed in related works section- have practical difficulties and strong requirements: like having many different shared keys between each of the nodes and the base station, requiring extra communication overhead, or the need some hardware modification, which are arguably too strong requirements for those proposals to be feasible.

4.3 System Model and Assumptions

We choose a sensor network architecture based on the single-tier clustered architecture model [113], where the network is partitioned into clusters using a clustering routing algorithm. Every cluster contains a number of heterogeneous sensor nodes (called group members) such as camera, audio and scalar sensors that transmit their sensed data directly to a cluster head. Cluster heads have more resources, are more powerful, and are able to perform intensive data manipulation and in-network processing such as aggregation and data fusion. Cluster heads are wirelessly connected with the sink or the gateway either directly or through other cluster heads in hierarchical multi-hop approach as shown in an example in Figure 4.1.

We use a clustering algorithm detailed in [170]. In this algorithm, the clustering algorithm starts from the base station (BS) that sends periodically broadcast messages called (*BS-Msgs*). The nodes that receive those messages and satisfy certain conditions -such as certain number of hops from BS, SNR, BER, RSS- will act as 1^{st}-level cluster heads and inform BS using acknowledgment messages (*Ack-Msgs*). 1^{st}-level cluster heads also will start broadcasting messages called (*CH-Msgs*) to the surrounding nodes. Depending on certain criterions (based on the clustering algorithm), the nodes that receive *CH-Msgs* -except BS- can be either cluster heads of lower levels or group

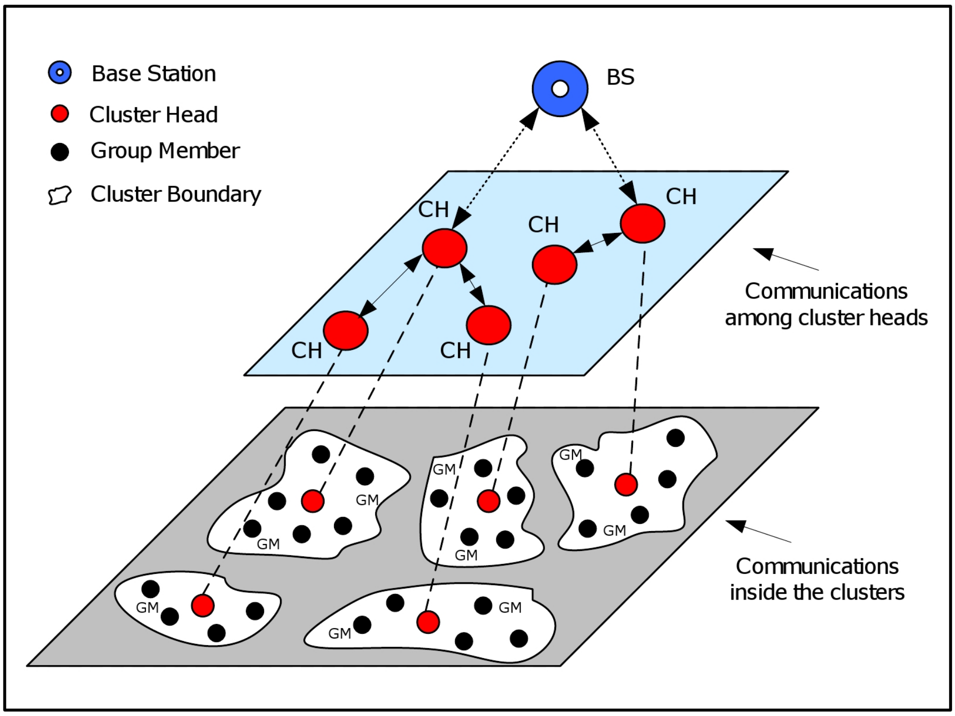

Base Station
Cluster Head
Group Member
Cluster Boundary
BS
CH
CH
CH
CH
CH
Communications among cluster heads
Communications inside the clusters
GM

Figure 4.1: A Simple Example of Clustered WMSNs

members (GMs) of the 1^{st}-level CHs. In the same way, these nodes will acknowledge their CHs or parents about their joining using *Ack-Msgs*. This process continues in the same manner to build the network until all nodes join the network and determine their rules, *i.e.*, cluster head or group member, and all possible paths are found.

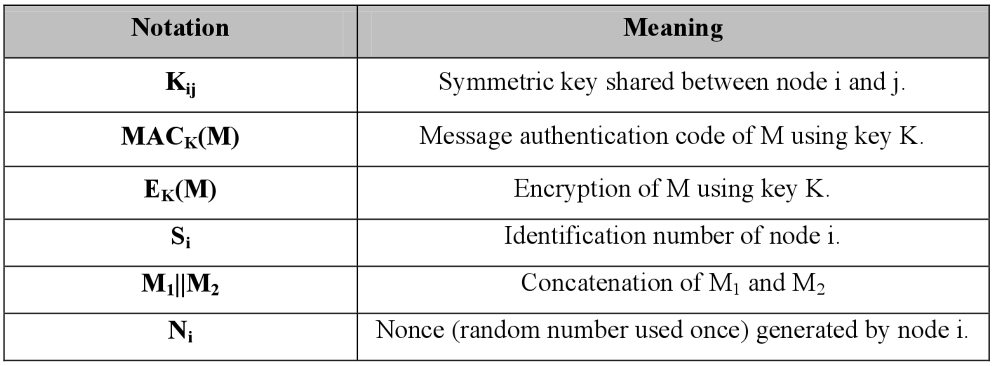
With respect to our security scheme, we assume that all nodes have unique identification numbers (IDs) since we use them to generate different set of unique keys for each node as described in the next section. The base station is assumed to be trusted during all network operation time and thus it can keep all the security keys in its memory. Source nodes (group members) in each cluster always encrypt and authenticate their data using the appropriate generated security keys before sending them to the base station through their cluster head. Only cluster heads (and of course the base station) have the ability, through using the appropriate security keys, to decrypt the data sent by their group members during the aggregation process. Also we assume that the time required for establishing secure links, at the route discovery phase of the clustering routing algorithm, is smaller than the time needed by an adversary to compromise a node during node deployment.

4.4 Light-Weight Distributed Key Management Scheme

In this section, we explain in detail our proposed lightweight distributed key management scheme. Our security protocol provides privacy and integrity against most of external attacks and limits the effect of insider attacks (*i.e.* nodes that have been compromised and controlled by an attacker who now possesses all valid keys) since the generated keys are unique and cannot be used in different areas of the network. In addition, our proposed security scheme allows for aggregation processing since cluster heads can decrypt the transmitted data if necessary and update the corresponding information. Moreover, our proposed security algorithm supports both authenticated encryption and authentication only services: with authenticated encryption, the data payload is encrypted using an encryption algorithm such as MISTY1 or Skipjack algorithm [133] and the entire packet is authenticated with a message authentication code (MAC) using for example Hash-based Message Authentication Code (HMAC) based on cryptographic hash function or message digest (SHA1 [171]). The MAC is computed over the encrypted data and the packet header. The decryption operation is done only at the sink and at cluster heads during aggregation process (if needed). In authentication only service where the data is not sensitive, our security scheme authenticates the entire packet with a MAC tag, but the data payload is not encrypted. The notations shown in Table 4.1 are used for establishing formulas of the needed security keys, as described in the following subsections.

4.4.1 Key Management

Our security scheme uses symmetric keys to encrypt and authenticate messages. Therefore, sensor nodes will only receive messages from other nodes that share the same security keys. The key management is composed of two phases: a key generation phase and a key distribution phase. In the key generation phase, a Master key (K_m) is installed in the sensor motes when they are programmed with the intended software. The key distribution phase provides a means for distributing or synchronization of the security keys (unique-node key, pair-wise key, and unique-cluster key) among the nodes. The



Notation	Meaning
K_{ij}	Symmetric key shared between node i and j.
$MAC_K(M)$	Message authentication code of M using key K.
$E_K(M)$	Encryption of M using key K.
S_i	Identification number of node i.
$M_1\|\|M_2$	Concatenation of M_1 and M_2
N_i	Nonce (random number used once) generated by node i.

Table 4.1: Security Notations and their Definitions.

key management protocol satisfies the following security properties of a key establishment:

- The shared security keys between a node and the neighboring nodes and/or BS cannot be used to recover the master key. This is guaranteed because of the one-wayness of a secure hash function.

- The master key (K_m) is kept by the nodes as long as it is necessary to establish security keys with their neighbors, and then it is deleted from their memory. The base station holds K_m along with the unique-node keys of each sensor node in the network.

1. **The unique-node key** (K_i) is the shared key between every node (i) in the network and the trusted base station and used to verify node identities and exchange data securely. The derivation of this key for each node is as follows:

$$K_i = F(K_m, S_i) \tag{4.1}$$

   where $F()$ is a secure hash function, K_m is the master key and S_i is the unique identifier (number) for the i-th sensor node.

2. **The pair-wise key** (K_{ij}) is the shared key between two neighboring nodes i and j, such as each GM and its CH in a cluster and every CH and its parent in the

network, and it is computed as follows:

$$K_{ij} = F(K_m, S_i, S_j) \qquad where S_i < S_j \tag{4.2}$$

Notice that since both nodes have the master key, they both can compute the pair-wise key. This key now can be used to send information from S_i to S_j (and vice versa) in a secure and authenticated manner. Also this key is needed especially for aggregation process and data fusion at the CH. In order to support data aggregation, CHs need to be able to look at data sent by their GMs and perform some function on the data if necessary.

3. **The unique-cluster key** (K_{ch}) is the shared key within a cluster in the network and it is different from other cluster-keys of other clusters. This key is used for command dissemination and broadcasting messages and it is generated as follows:

$$K_{ch} = F(K_m, S_i, S_i) \tag{4.3}$$

where S_i is the unique identifier number for the cluster head. Once the unique-cluster key is settled, node i can broadcast commands/messages to all the nodes in the cluster encrypted and authenticated using K_{ch}.

4.4.2 Implementation of the Key Management Scheme

In this subsection, we are focusing on implementing our security scheme and authenticating the participating sensor nodes during the process of establishing the routes in the clustered WMSN network. As we mentioned before, all the nodes in the network have the stored master key (K_m) before deployment, which is used by each node to compute the *unique-node key* shared between the node and BS as shown in equation 4.1. Before BS starts broadcasting its *BS-Msg*, it first authenticates itself by properly calculating the MAC tag of this message using the master key (K_m) and including this MAC in one of the fields of *BS-Msg*, as follows:

$$BS \longrightarrow S_j : BS - Msg, \tag{4.4}$$

$$= MAC_{Km}(BS - Msg||S_{BS}), S_{BS}, N_{BS}, BS - Msg$$

Upon receiving *BS-Msg* message by any surrounding node S_j, the node S_j checks the MAC code using the same master key to confirm the ID of the sender before processing the other data and extracting the routing information. If the MAC is verified, S_j will connect with BS and add it in his routing table along with all relevant security information such as nonce identifier (random number used once composed of random number and time stamp) and sequence number used to avoid replay attacks, and then the *pair-wise key* will be generated as shown in equation 4.2. On the other hand, when the verification process is valid, these surrounding nodes reply to BS by sending back *Ack-Msgs* authenticated with MAC using the unique-node key to avoid impersonation attempts and to authenticate themselves to the BS as 1^{st}-level cluster heads.

$$1^{st} - levelCH(S_j) \longrightarrow BS : Ack - Msg,$$

$$= MAC_{Ki}(Ack - Msg||S_j), S_j, N_j, Ack - Msg \tag{4.5}$$

In the same manner, cluster heads include in their *CH-Msgs* the MAC tag using the master key and other relevant security information in order to authenticate the communication and exchange data securely between them and other possible cluster heads and/or their group members:

$$CH \longrightarrow S_j : CH - Msg, \tag{4.6}$$

$$= MAC_{Km}(CH - Msg||S_{CH}), S_{CH}, N_{CH}, CH - Msg$$

After receiving *CH-Msg* and verifying the MAC, each node S_j behaves as a group member in the constructed cluster or as cluster head of lower level sends back an *Ack-Msg* to its CH/parent, which is also properly authenticated by including the MAC tag using the generated pair-wise key.

$$S_j \longrightarrow CH : Ack - Msg,$$

$$= MAC_{Kij}(Ack - Msg||S_j), S_j, N_j, Ack - Msg \tag{4.7}$$

After establishing the network and each node knows its parent (if it is a cluster head) and its cluster head (if it is a group member), all the cluster heads and the group members compute the *unique-cluster key* of their clusters as shown in equation 4.3. This key will be used later on for authenticating any cluster head messages for broadcasting messages or disseminating commands (*Cmd-Msg*) within the cluster.

$$CH \longrightarrow GMs : Cmd - Msg, \tag{4.8}$$
$$= MAC_{Kch}(E_{Kch}(Command)||S_{CH}), S_{CH}, Cmd - Msg$$

Finally, the master key should be deleted from the memory of all nodes since the security of our protocol depends on the deletion of the master key at the end of the process. We should take care that the deletion is unrecoverable, for example by overwriting the master key (in practice several times like using the 'shred' linux command or a similar one with the file where the key is stored. Shred overwrites a file many times with random bits; with enough passes it becomes impossible to recover the old contents of the file). Now, the communication between the nodes and the base station has been secured and the sources can send their data securely:

$$S_j \longrightarrow CH : Data - Msg, \tag{4.9}$$
$$= MAC_{Kij}(Header||E_{Kij}(Data)||S_j), S_j, Data - Msg$$

4.4.3 Node Addition

Notice that after deletion of the master key from the network, we will encounter a problem for node addition, node movement, or when a path condition changes during a local repair or rediscovery process. In all these cases, we cannot establish a new connection between two nodes that did not communicate before without having the appropriate shared security keys between them. And in order to generate these keys, the nodes should know the master key that is already deleted from their memory.

One possible solution to this problem is to assume that the base station is secure and trusted at any time during the network operation, so that there is no need to delete the master key from its memory. In the case of node addition, a new node has the master key but the existing nodes do not have. So, the new node first generates the unique-node key from the master key and waits until it hears broadcast messages (*CH-Msg*) from other CHs, then it selects one of them, or more depending on the proposed routing protocol, as its CH or parent toward BS. The new node then sends a request message authenticated with the MAC tag using the unique-node key to the selected CH asking the base station for the appropriate keys (pair-wise and unique-cluster keys) for the new connections. The selected CH forwards the request message as it is (without verifying the MAC) toward BS. Then, BS verifies the MAC of this message, generates the requested security keys using the master key, and sends them back to the requesting node and its CH or parent. The new node then establishes the new connections and deletes the master key from its memory.

For node movement or when the path conditions have changed, the same process of adding new nodes will be followed except that these nodes do not have the master key as well but they have already generated the unique-node keys.

4.4.4 Security Performance of the Key Management Scheme

We examine in this subsection the security protection achieved by our proposed key management scheme against some well-known security attacks in sensor networks. The scalability analysis and the memory requirement of the proposed key management scheme are discussed in the next subsection. Having our efficient key management scheme will enable the network to use secure routing and protect the exchanged data from being exposed by an unauthorized party and guarantee the integrity of their data. With this level of security, most of the outsider attacks against WMSN routing protocols can be prevented (through encryption and authentication) using the shared security keys. Initially, using the shared master key can prevent unauthorized nodes from joining the network and hence attacks like selective forwarding, acknowledgment spoofing, wormhole, and sinkhole attacks by external nodes are disallowed. Our key management scheme also verifies node identities and bidirectional link, as well as authenticating broadcast messages.

Identities can be verified by sharing a symmetric unique-node key for every node with a trusted base station. Two neighboring nodes can authenticate the bidirectional link between them by using the master key to verify other identity and establish a shared pair-wise key for securing the communication between them. Broadcasting can be authenticated by using a unique-cluster key derived from the master key and shared, for example, within a cluster or group of nodes. Therefore, Hello-flood attacks are avoided in most situations since broadcasting messages in the network are always encrypted with master key and unique-cluster key. Also by using the unique-node and pair-wise keys, attacks such as sybil attack are prevented because a single node cannot present multiple identities without having the corresponding security keys. The network is also protected against replay attacks as all messages exchanged in the network are tracked by a time stamp and sequence number.

4.4.5 Scalability Analysis of the Proposed Scheme

In this subsection, we are interested in analyzing the effect of clustering the network on the number of symmetric keys that should be stored dynamically in each cluster head, which mainly depends on the number of nodes inside the cluster. The distribution of wireless sensor nodes can be modeled by the Poisson distribution [172], where (N) sensor nodes are uniformly deployed in a network of (R) area. The whole WMSN's area R is clustered into a set of clusters, each of which has an (r) area and a cluster head in the center of the cluster. If we assume that the number of nodes inside each cluster equals to (n), then we can use the Poisson distribution with cluster node density ($\lambda = \frac{N}{R} \cdot r$) to represent:

$$P_r(n|r) = \frac{\lambda^n \times e^{-\lambda}}{n!} = \frac{((N/R) \cdot r)^n \times e^{-(N/R) \cdot r}}{n!} \tag{4.10}$$

Where $P_r(n|r)$ is the probability of having n nodes deployed in a cluster of an area r.

If we assume that the cluster has a shape of circle as shown in Figure 4.2, which is a suitable approximation of the coverage of an omnidirectional antenna, then the expected (average) number of nodes in a cluster of radius (f) - factor of the transmission range (t) of the cluster head determined by the clustering algorithm- and area of $r = \pi f^2$ can be represented as:

$$\mathbb{E}\{n\} = \sum_{n=0}^{N} n \cdot P_r(n|r) = \sum_{n=0}^{N} n \cdot \frac{\lambda^n \times e^{-\lambda}}{n!} \tag{4.11}$$

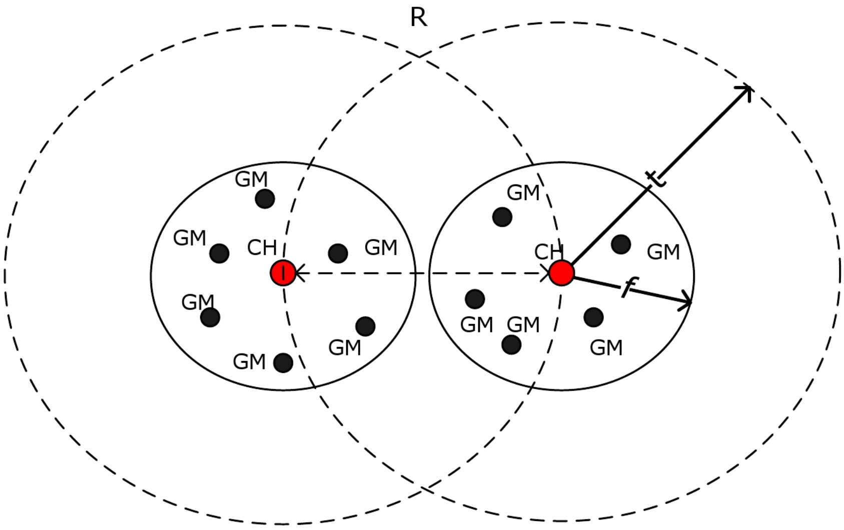


Figure 4.2: An Example of Clustered WMSNs of Area R

For $N >> 1$, then $\sum_{n=0}^{N} n \cdot \lambda^n / n! = \lambda \cdot e^{\lambda}$ and the expected number of nodes in area r, $\mathbb{E}\{n\}$, can be simplified to:

$$\mathbb{E}\{n\} = \lambda = \frac{N}{R} \cdot \pi f^2 \tag{4.12}$$

Then, from equations 4.10 and 4.12, we can determine the probability of having an average number of neighboring nodes for a sensor node in a cluster as follows:

$$P_r(n = \mathbb{E}\{n\} | r) = \frac{\lambda^{\lambda} \cdot e^{-\lambda}}{\lambda!} \tag{4.13}$$

Equation 4.13 can be simplified by using Stirling's formula and substituting $\lambda! \approx \sqrt{2\pi\lambda} \cdot (\lambda / e)^{\lambda}$, then:

$$P_r(n = \mathbb{E}\{n\} | r) = \frac{\lambda^{\lambda} \cdot e^{-\lambda}}{\lambda!} \approx$$

$$\frac{\lambda^{\lambda} \cdot e^{-\lambda}}{\sqrt{2\pi\lambda} \cdot (\frac{\lambda}{e})^{\lambda}} = \frac{1}{\sqrt{2\pi\lambda}} \tag{4.14}$$

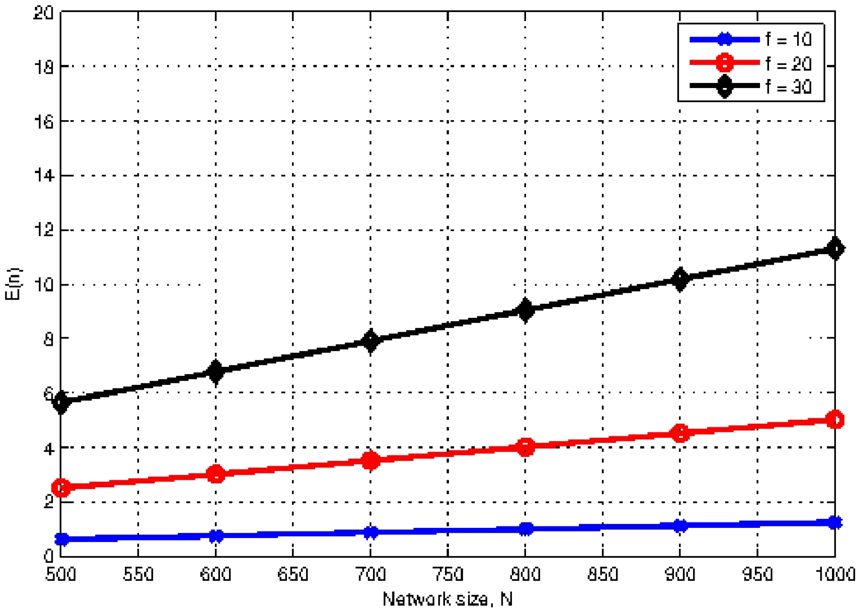


Figure 4.3: Average Number of Nodes in a Cluster with Different Cluster Radius Values (f, meter).

So,

$$P_r(n = \mathbb{E}\{n\}\,|r) = \frac{1}{\pi f \cdot \sqrt{2N/R}} \tag{4.15}$$

We can notice that the density of the number of nodes after the clustering process remains the same as the deployment of the sensor nodes is randomly uniform. Figure 4.3 shows the average number of nodes in a cluster with different cluster radiuses and different node numbers (network size) deployed in an area of 500×500 m^2. For an example: if the number of randomly deployed nodes N = 900 inside a network area R = 500×500 m^2 and cluster radius f = 30 m, then the average number of nodes inside a cluster (from equation 4.12) n is about 10 with the probability of having this average number of neighboring nodes is around 12.5 % (from equation 4.15). Therefore, we can notice that the number of nodes inside the cluster after the clustering process remains small regardless the total number of nodes. This means that our key management scheme scales nicely as it requires only local information for key management without needing of central distribution, and the number of security symmetric keys needed to be stored at each node does not depend on the network size but only on node density (which does not increase too fast with network size).

4.5 Light-weight Intrusion Detection Scheme

In this section, we introduce a light-weight intrusion detection technique for cluster-based WMSNs that can detect internal attacks or third party's attempts of exploiting possible insecurities and warn for malicious attacks. Our intrusion detection scheme prevents malicious attempts in each cluster by discovering compromised nodes both whether they are group member nodes (GMs) or the cluster head (CH) itself. Although in internal attacks an adversary can capture sensor nodes and retrieve their security keys, we assume that such adversary cannot deploy malicious nodes that outnumber the legitimate nodes, by replicating compromised nodes or introducing new intruders in several parts of the cluster.

4.5.1 Checking GMs

In WMSNs, the use of densely deployed sensor nodes leads to having redundant information for the same events occurred in the network. Because in such dense networks, any event (or object) occurred in any given time is detected by a group of sensor nodes that are mostly in the same neighborhood (vicinity). This redundancy can be exploited to verify the behavior of a group member (source node) by comparing the transmitting activity of that node with its neighbors in the same time interval. In our intrusion detection scheme, we assume that each CH keeps track of each GM in the cluster after the network is set up and clustered (e.g. neighboring nodes, shared security keys, number of received packets, etc). Therefore, each CH is responsible to check its GMs in the cluster by comparing the received data packets from a certain suspicious node with the data received from its surrounding nodes for the same event.

So when a CH suspects on a certain GM, it compares the data packets (or their description) transmitted by the GM with the data packets from the neighbors of that node. If these data packets have similarity and reveal the same occurred events, then it means the node is working probably and not intruder, otherwise, the node is probably misbehaving.

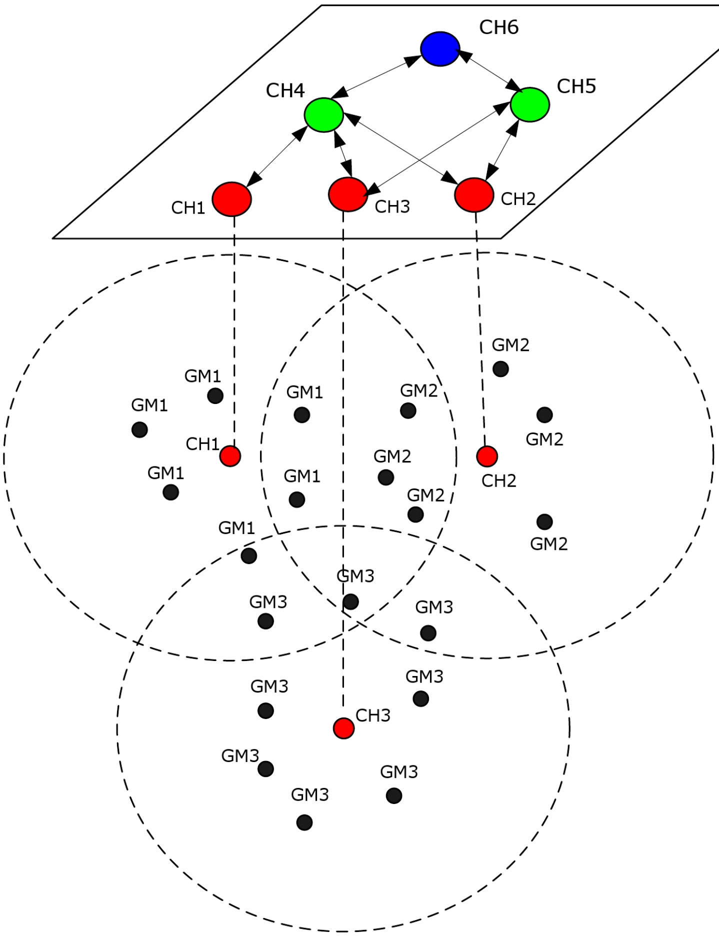


Figure 4.4: An example for IDS for Clustered WMSN.

The same process also is used to check whether a node is not reporting data of detected events (or just keeps sending normal data indicating no-event detection). In this situations, The CH is also responsible for monitoring and detecting these cases and checks if a GM is not reporting an occurred event even though its neighbors are sending packets for that event. Notice that in checking GMs, our IDS does not need to exchange messages among the nodes or require any extra communication overhead. CH is only monitoring the data packets transmitted by GMs in the cluster and comparing them to detect any possible intrusion.

4.5.2 Checking a CH

Having the cluster heads as monitor nodes on their group members at their clusters satisfies most of the security requirements as they have direct communication with each node and they can compare easily the behavior of any node with its neighbors. However, a single monitoring node in a cluster fails to meet the "trust no node" requirement and could be a single point of failure, since the cluster head itself can be compromised by the adversary. Therefore, in order to check a CH, other nodes should monitor the behavior of it to see whether it forwards correctly the packets it receives.

For the data packets originated from nodes inside a cluster, the group member nodes that reside on the intersection areas with other clusters, as shown in Figure 4.4, are responsible to verify the behavior of their CHs. In this case, those nodes are communicating with the CHs of other clusters to trust their own CHs. For example in Figure 4.4, in cluster 1, GM nodes in the intersection area between cluster1 and cluster2 suspect on their cluster head (CH1). Therefore, they send the same description of their data to both CH1 and CH2 and inform CH2 to verify the behavior of their CH (CH1). In its turn, CH2 communicates with the parent CH (CH4) to compare whether the same event description is sent by CH1.

For the data packets forwarded from other CHs, the sending (children) CHs are responsible to monitor the behavior of the receiving (parent) CHs by exploiting the multiple-path route to the sink. For example in Figure 4.4, if CH3 suspects on the behavior of CH4 whether it forwards correctly the data received, CH3 sends the same data description to CH5 as well. CH5 then verify these data descriptions with its parent (CH6), which is also the parent of CH4. If CH6 receives the same data from CH4 as described by CH3, then CH4 is working probably. If CH6 does not receive the same described data or receives different data for the same intended event, then it means that CH4 is probably a compromised node (intruder).

4.6 Simulation Performance Evaluation

Due to the limited energy lifetime of battery-powered sensor nodes, any proposed security scheme has to be energy efficient. Since communication operations in sensor networks is much more expensive than computation process, we use communication

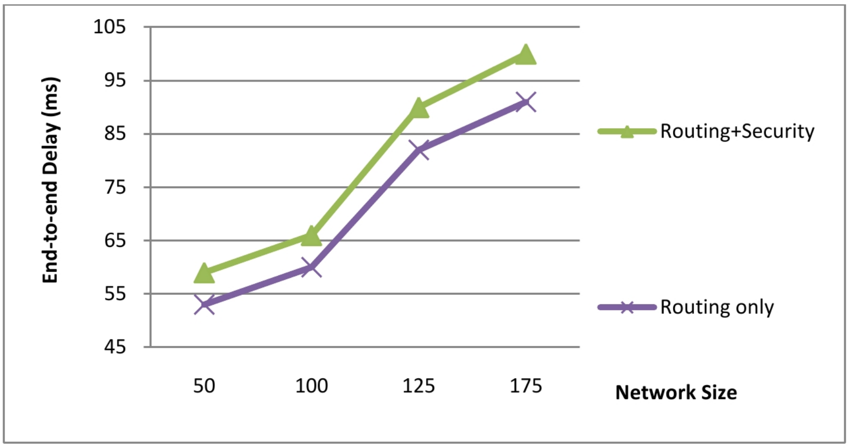


Figure 4.5: Average End-to-end Delay of a Secure Clustered WMSN.

cost to measure the performance of our protocol. We conduct several simulation experiments (over 100) with various random topologies using NS-2 simulator (version 2.34) to evaluate the network performance of the security scheme. As explained in the previous chapter, we use similar settings to simulate the proposed security scheme considering a multi-hop network of size 500 m × 500 m deployed with different number of sensor nodes ranging from 50 to 200 in randomized grid. The sink is located in the center of the network. The traffic is CBR of 600 packets/sec and the packet size is 316 bytes. The rest of the simulation environment and other parameters can be found in Table 3.2.

4.6.1 Network performance

Our proposed security scheme of key management is intended to be a light weight security protocol that provides sufficient level of security to WMSNs with insignificant effect on overall network performance in terms of end-to-end delay, network throughput, and energy consumption, as shown in Figure 4.5, Figure 4.6, and Figure 4.7 respectively.

The security protocol needs only to add few extra bytes (160 bits in the case of SHA-1 that can be further truncated) on the packet for the MAC tag. This small increase on the packet size causes small amount of additional delay for transmitting

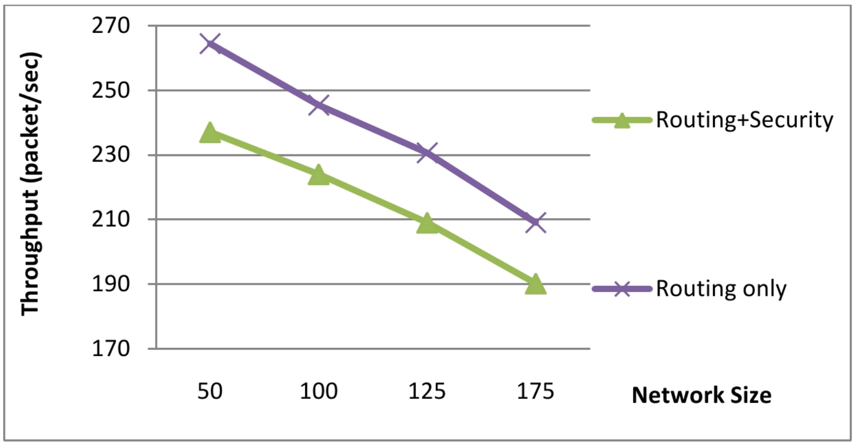


Figure 4.6: Average Throughput of a Secure Clustered WMSN.

those bits as well as processing them at each hop for authentication purpose. Figure 4.5 shows the average end-to-end delay of delivering data packets from sources to base station in both cases: having the proposed security scheme implemented (Routing+Security) and without security (Routing only). Note that data decryption process, which needs higher processing energy consumption and delay than authentication only service, is only done at the sink side and cluster heads in case of aggregation operations. In Figure 4.6, it is shown the effect of applying our security scheme on the network throughput, which can be understood from above result.

Our security scheme does not require any transmission of extra messages among the nodes in order to authenticate each other and secure the communication between them. It only uses the same messages exchanged during routes discovery phase to add the needed security information for authentication and encryption process. In addition, our security protocol uses different type of keys that allow not only secure pairwise communication, but also broadcast messages within a cluster without the need of making multiple transmission of the same message to all neighbors. Also, using different type of keys gives cluster heads in clustered networks the ability to perform the necessary aggregation and data fusion operations. Figure 4.7 shows the energy consumption performance of our proposed security scheme.

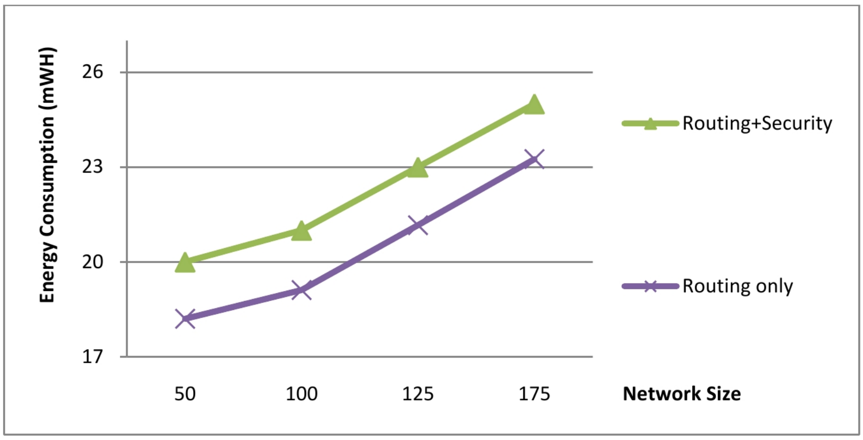


Figure 4.7: Average Energy Consumption of a Secure Clustered WMSN.

4.6.2 Scalability and Memory Requirements

Scalability: Our security scheme scales well as it requires only local information for key management without needing of central distribution. Furthermore, the number of security keys needed to be stored at each node does not depend on the network size but only on node density (*i.e.*, average number of group members within a cluster and number of neighboring cluster heads). For example, in case of using our proposed clustering protocol, average number of GMs or the size of cluster can be determined by adjusting the value of *Thr-High*. Also it is shown in Figure 4.8 that the density of cluster heads in the network is decreasing with the network size and tends to be fixed at large network size.

Memory requirements: We can notice that the majority of nodes in the network (*i.e.*, GMs) needs only to store three security keys: unique-node, pair-wise, and unique-cluster keys. On the other hand, each CH needs to store the pair-wise keys it shares with its group members and parents, in addition to the unique-node and unique-cluster keys. That means storing M keys (for $M = K_i + K_{ch} + (m+n)K_{ij}$ where m is number of parents and n is number of nodes in the cluster). So, if the average number of parents is 6 (*i.e.*, six different paths to BS) and average number of nodes in the cluster is about 10 does not need considerable memory space.

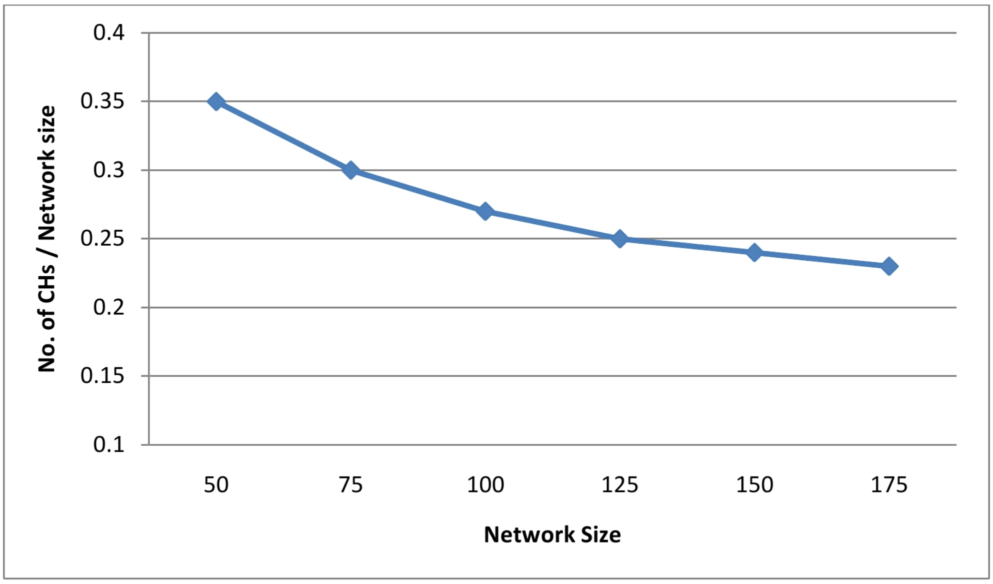


Figure 4.8: Density of Cluster Heads vs Network Size.

4.7 Conclusions

In this chapter, we presented a light weight distributed security scheme of key management and intrusion detection system suitable for securing the communication over clustered WMSNs with minimal impact on overall network performance through balancing its security features against the communication and computational overhead required to implement it. Our proposed security protocol is based on symmetric key ciphers used to authenticate and encrypt the transmitted data and it only requires the sensor nodes to share keys with their cluster heads or one-hop parents. It protects against the majority of outsider attacks, and it resists against insider attacks since the stolen keys are unique and affect only the local area. The key management scheme is energy efficient with no extra communication overhead, scalable for large scale network, and designed to facilitate the data aggregation at cluster heads and message broadcasting within the clusters using unique-cluster security keys. The proposed light-weight distributed IDS is simple, with very little communication overhead, and efficient to identify malicious internal attackers in clustered WMSNs. Performance evaluation results show that our proposed security scheme is appropriate for securing multimedia delivery while being resilient against general security threats, it has an insignificant effect on network performance metrics (such as average end-to-end delay, throughput, and energy consumption), and it is scalable while having minimum memory requirements. In next chapter, we give the reader a complete view of the state of the art at all aspects of event unobservability issue before introducing our proposed location privacy scheme.

Chapter 5

Event Unobservability in Sensor Networks

5.1 Introduction

WSNs are often used in applications where it is difficult or infeasible to set up a wired network, especially in harsh and hostile environments, such as habitat monitoring, military surveillance, health care, and target tracking. Security and privacy in many of these applications are of paramount importance, especially when we know that WSNs are vulnerable to attacks more easily than the wired networks because of their nature as a broadcast medium [126]. In addition, the lack of resources because of the reduced cost and size of sensor nodes make WSNs extremely vulnerable to different types of attacks, from the hardware to the application layer.

Most of the security mechanisms used in sensor networks such as encryption, authentication, and secure routing allow sensor nodes to protect their transmitted information (data content) from being exposed by an external unauthorized party and to guarantee data integrity. Although this level of security protects the network from the outsider attacks and satisfies most of the needed security requirements (such as confidentiality, authentication, integrity, and availability), still it cannot fully address the privacy of contextual content in wireless sensor networks. Therefore, another level of

[1] Chapter 5 is based on the publications:
Event Unobservability in Wireless Sensor Networks: A Survey; Islam T. Almalkawi, Manel Guerrero Zapata, Gamal N. Al-karaki submitted to *Journal of Networks and Computer Applications (JNCA), February 2013*

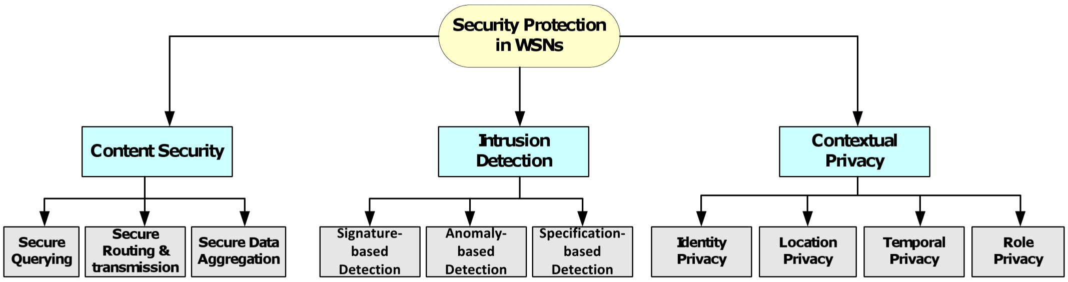


Figure 5.1: Taxonomy of Security Protection in WSNs.

security protection is needed to offer contextual privacy, which is called event unobservability, anonymity, or privacy [173] (we will use these terms synonymously for the rest of the chapter).

Figure 5.1 presents a complete classification for security protection in WSNs, which illustrates the major security fields in WSNs that are used to have a totally secure system. These fields include **Content Security** which concerns about securing data content from unauthorized users and external adversaries, **Intrusion Detection** which concerns about protecting the data from compromised nodes and internal attacks, and **Contextual Privacy** which concerns about concealing contextual information from attackers. Content security consists of three main operations for securing collected data in different phases during data communication in sensor networks: Secure Routing & Transmission [128], Secure Querying [174], and Secure Data Aggregation [175; 176]. Intrusion detection also consists of three main mechanisms for detecting internal attacks in WSNs, which are: Signature-based detection [177], Anomaly-based detection [178], and Specification based detection [145]. On the other hand, contextual privacy (which is our focus in this chapter) includes four main types of event privacy in WSNs. In the following sections, we define each type of the event privacy and then we zoom in (extend) this part to explain in details all the forms of privacy and their most proposed mechanisms in the literature.

However, providing event unobservability in sensor networks is challenging task for many reasons. *First*, wireless sensor networks use broadcast medium to exchange messages among nodes, which makes it easy for the adversary to eavesdrop the network traffic. The adversary can use available information like message transmission time and frequency to perform traffic analysis -as explained later- and disclose critical information about entities or events in the field. *Second*, it is very expensive to apply traditional anonymous communication techniques for hiding the communication between sensor nodes and sinks. Event unobservability schemes should take into account the limited resources of sensor networks in terms of energy, storage, computation, and communication and the use of low-cost devices in order to be affordable. *Third*, the proposed event privacy for sensor networks should consider the different capabilities of possible attackers (either local or global). While local attackers can listen and detect message transmission within a certain range, global attackers can monitor the communication in the entire network either by scattering their nodes throughout the network or by using powerful site surveillance device with hearing range covers the network area.

The resource scarcity, ad-hoc broadcast deployment, immense scale of WSNs, and the enhanced adversary capabilities make securing WSNs a particularly challenging problem and has no correspondence in the wired setting, hence opening research directions for novel solutions to address context privacy. In this work, we introduce our complete classification for content security and contextual privacy in WSNs that is expected to guide in the design of new improved solutions. We focus in this chapter in revising the contextual privacy preservation in WSNs: define each form of contextual privacy, explain every possible attack methodologies, and survey the state of the art of existing countermeasures showing their advantages and drawbacks. More specifically, we first investigate the location privacy problem for the source, sink, and query location and review most of the proposed privacy-preserving techniques. Then, we analyze the protection of node identity privacy and explain its presented approaches. We also examine the temporal privacy issues and demonstrate the existing schemes related to this subject. Finally, we discuss the existing solutions comparing to each other and tried to stress on some interesting and challenging open issues for new researchers in this field.

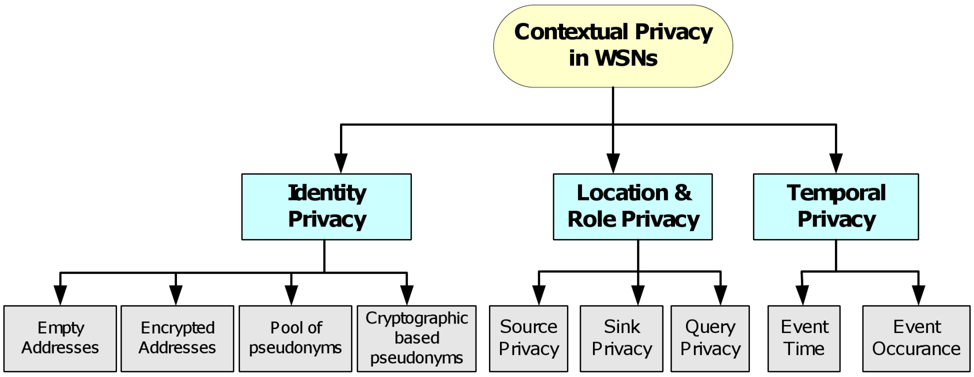


Figure 5.2: Taxonomy of Contextual Privacy in WSNs.

5.2 Contextual Privacy in WSNs

Contextual information in sensor network is the information gathered through generating, transmitting, and routing data messages within the network. Contextual content has many aspects or properties that can be used to expose entities in a communication system. For example, delivering sensor data to the destination node may leak the locations of some critical events or identities of important sensor nodes in the field to the attacker who maybe passively monitoring the network. In addition to the location or identity of the interested entity, the attacker can figure out more sensitive information from the contextual content like whether and when a particular event occurred. For instance, in the endangered animal monitoring application [179], the adversary (hunter) may be interested in any information about the animal not only its location, but also whether and when an animal is detected by the system. Moreover, it is argued in [180] that information of the carrier frequency used in wireless communication can be also sensitive, because an attacker can use this information to find out the hardware platform used by sensor nodes and exploit the vulnerabilities of the particular version of the software running on that platform.

Specifically, the contextual property can be identity, location, role, existence, or time occurrence and thus leads to the following forms of privacy, as shown in Figure 5.2:

- **Identity privacy:** In general in any network of set of nodes or entities, each node has its unique identity (i.e ID or IP address) that makes the node distinguishable from others and reachable by addressing this identification or address. Therefore, it is necessary that the identity or address of the node must not be revealed to an external attacker or even to internal entities in some cases. For example, concealing the identity of the sink node, for its importance role in the network, can be done through encrypting the destination field in the message or forwarding the message randomly until it reaches the sink node.

- **Location privacy:** Which is the most popular and studied in the literature, and it is concerned with protecting the location information of nodes or entities (especially the event sources and sinks) in the network. Physical location information of nodes can be compromised by observing the wireless signals from nodes and traffic flow in the network. Failure to protect such information can completely subvert the intended purposes of sensor network applications. For example, in animal hunter scenario, revealing the location information of the source node will end with the hunter to locate and capture the monitored animals. Also determining the sink location by an attacker and compromising it can make the entire network rendered useless.
  Notice that identity privacy is different than location privacy. For example, the attacker may overhear the exchanged messages in the network and know the IP address/ID of the sink node without being aware of its location. On the other hand, the adversary can make use of signal detection techniques and traffic analysis tools and know the approximate location of a source node without determining its identity.

- **Role privacy:** In general, any node or entity in a network has its own role and functionality that can be a range of being source, relay, cluster head, sink node, etc. Sometimes, depending on the application and network topology, some nodes have critical roles especially the source and sink nodes. For example, an adversary from monitoring the network can determine the source nodes where the traffic is generated and started, and also he/she can determine the sink node where the traffic density increases and the message flow ends. Therefore, role privacy aims to hide the activity of the nodes and their functions at the network in a way that all nodes appear undistinguishable from each other. However, most of the work done in this subject combines it with location privacy and deal with them as a one problem. For example, hiding the role of the sink problem is done through concealing its location.

- **Event existence & time occurrence privacy:** One of the goals of an eavesdropper is to figure out whether an interesting event occurred in the network and when it happened. The adversary can know these information, no matter how strong the encryption scheme used, by simply monitoring passively the traffic pattern in the network. For example, in animal-hunter scenario, if an animal passed through a certain area in the network, the surrounding sensor nodes will detect this event and report that to the sink node. This will change the traffic pattern at the place where the animal is detected and let the hunter know about this occurrence.

Therefore, *event privacy* in sensor networks is concerned with protecting the context associated with the measured readings and transmission of sensed data, and it can be defined as the state of being not identifiable by an adversary at a given time. Event privacy strength depends on the adversary's capability, his current knowledge, and his ability to learn. For example, contextual information of a target might be completely anonymous to a certain adversary while it can be revealed easily to another stronger adversary. Therefore, type and complexity of countermeasures for event privacy should take into account the capabilities of the attackers.

5.3 Location Privacy in WSNs

Among the different types of privacy, as mentioned before, location privacy is of special interest and it is an important security issue in sensor networks. This is because location privacy plays an important role in WSNs in preserving location information

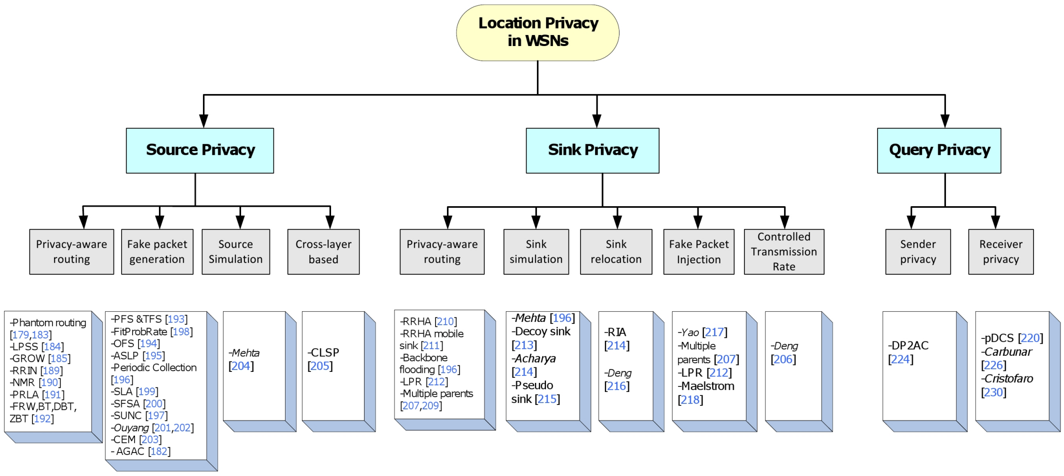


Figure 5.3: Location Privacy Techniques in WSNs. (Names in italic indicate the last name of the first author of the proposed technique.)

of critical sensor nodes, such as event sources, storage hops, and sink nodes. Lack of location privacy can expose significant information about the entities in the field and the traffic carried on the network. Consequently, location privacy can be classified into: Source Location privacy, Sink Location privacy, and Query Location Privacy. In the following sections, we review the existing techniques for protecting location privacy of event source, sink node, and queried node.

An attacker might try to gain contextual information about the location of the reported events either from the content of the packets or from the traffic pattern generated due to the operation of the network. Packets contain information both in the payload and in the header and in many cases header is kept unencrypted because it is used at every hop for routing purposes. Therefore, an attacker may exploit header information to retrieve sensitive information about the sender and recipient of the packet. Also, an attacker might use traffic analysis and packet tracing tools to infer the locations of data sources and sinks. For example, a region in the network with high activity should be close to a sink, while a region where the packets are originated should be close to an event source.

Most of the proposed location privacy techniques in the literature are depending on introducing randomness to packet routing process in order to increase the anonymity of traffic patterns observed by the adversary and to defeat the adversarial traffic analysis attacks. However, the performance cost associated with privacy protection in terms of packet delay, energy consumption, and packet delivery is significant and cannot be neglected. Thus, successful techniques should take into account the limited resources in sensor networks and try to offer a trade off between the strength of the privacy protection and network performance.

5.3.1 Source Location Privacy

Source location privacy, or source unobservability, aims to hide the source location from being leaked to attackers and mislead them in order to increase the *safety period*, which is the number of packets sent by the source node before the source is localized. Indeed, the interest of the attacker is not the node itself but the location of the event. However, the attacker might use that information to get an approximation of the location of the event. On the other hand, attackers can determine source location by using signal detection techniques such as trilateration/triangulation, or angle of arrival method using multiple antennas [181]. Even with a single transceiver, attackers can detect the source of wireless signal of a source node by calculating the message reception rate or using signal strength with considerable accuracy [180].

This problem -source location privacy- was first analyzed in [179] by the well-known "Panda Hunter Game". It proposes a scenario where a large sensor network is deployed with panda-detecting sensors to monitor the behavior of pandas in their environment. Whenever a panda comes into the sensing range of a sensor node, it will generate event messages and transmit them towards the base station. Meanwhile, a panda hunter (attacker) attempts to find the location of pandas and hunt them by taking advantage of the already existing infrastructure. In this game, the proposed source location privacy technique should protect the location information of source nodes from being leaked to attackers, while in the same time it should not disturb the main objective of the network and allow data to move in efficient manner towards the sink. On the other hand, attackers try by eavesdropping the wireless communication among nodes and using traffic analysis methods to reveal data sources location, and thereby the pandas.

However, as we mentioned before, countermeasures for location privacy should take into consideration the attackers capabilities, local or global, active or passive, external or internal. While, most of the time, the common assumption made in location privacy is that attackers are external entities who do not have the security keys to decrypt data packets, and they are passive entities who do not confuse the normal operation of the network (by using for example traffic injection, channel jamming, and denial of service attacks (DoS)), we found many proposed schemes assume the attacker be either local or global eavesdropper [182]. Local adversary usually starts eavesdropping packet transmission at somewhere in the network close to the base station. Upon receiving the first packet, the attacker can determine its sender node by using radio frequency localization techniques, and subsequently he/she moves towards that node. This hop-by-hop trace process is repeated until the attacker reaches the real source of the data. The attacker is able to find the event source because the packets tend to follow relatively static paths to reach the base station. On the other hand, global attacks who is able to eavesdrop and analyze all the communication in the network can easily infer the locations of source nodes by observing the first node initiates the communication with sink node.

In this section, we are discussing the different privacy techniques that have been developed in the literature to counter these different adversary capabilities, as shown in Figure 5.3, and we classify them into four categories: Privacy-aware routing, Fake packet generation, Source simulation, and Cross-layer based techniques.

5.3.1.1 Privacy-aware routing

Routing-based privacy protocols can provide a degree of source location privacy especially against local attacks. The main idea is to prevent the adversaries from tracing back to the source location through traffic monitoring and analysis. For example, at a certain node while back tracking, an attacker selects the next node towards the source depending on which node sent out the most recent message. And because the possible path for each message can be different, the adversary may be drawn in different

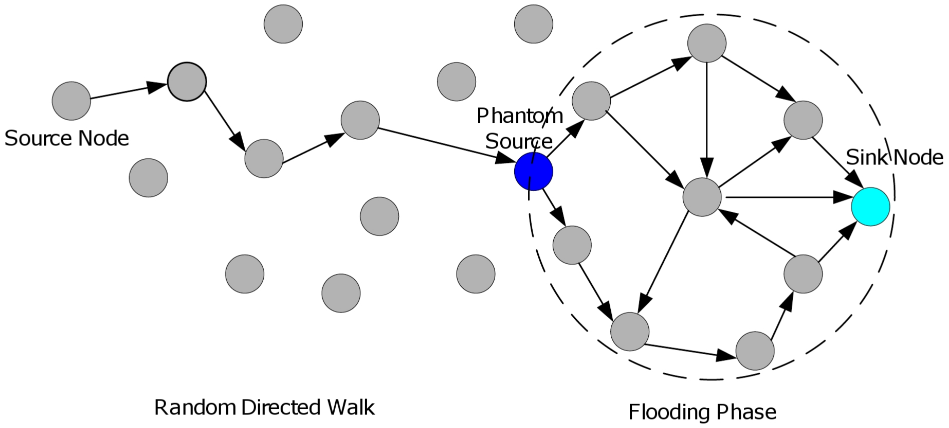


Figure 5.4: Phantom Routing.

directions by different messages. Therefore, the aim from the privacy-aware routing scheme is to make this happen even when the messages have the same source node. As a result, the adversary will spend extra time on some nodes which are not on the correct shortest path to the source node, and hence the time required to find the source node will be prolonged.

The popular Panda-hunter game model is presented in [179; 183] to formalize the source location privacy problem and it also introduced the phantom routing protocol. *Phantom routing* is used in message delivery from the location of the panda to the sink for preserving its location privacy, and it involves two phases as shown in Figure 5.4: A random walk phase and then a flooding or single-path routing phase. In the random walking phase, the message from the real source will be routed randomly for a certain number of hops to reach what called a phantom source (a selected node that will initiate the second phase). The path that will be taken by the message in the first phase is either pure random walk or a directed walk. In order to avoid random walks canceling each other and to make sure that the phantom source will be far away from the real source (which will make the real source's location hard to be traced back by the adversaries), the directed walk is preferred and it can be implemented depending on the relative geographical locations of the nodes or on the hope-count from the sink node. In the second phase, the selected phantom source sends the message to the sink node by

flooding (Phantom Flooding) or through single-path routing (Phantom Single-path). Surprisingly, both provide the same privacy protection level -especially if the path between a source and the sink is not long- because the shortest path is always contained in a flooding (as first packet arrived uses shortest path).
Also this basic flooding (called also baseline flooding), where each node broadcasts the packet it receives from one neighbor to all of its other neighbors, consumes significant amount of energy in the whole network and reduces the network life time. Therefore, probabilistic flooding is employed where each node forwards the packets following a probability distribution, which means that not all the nodes will involve in the flooding phase. This will reduce the energy consumption and also reduce the probability of an attacker reaching the source.

However, phantom routing is still consumes considerable amount of energy due to random walking and flooding phases. In addition, the end-to-end delay will be high for the same reasons and even the packet delivery to the sink will not be guaranteed in case of using probabilistic flooding phase due to the randomness level in this approach. Also, this scheme assumes local attacker with limited coverage, comparable to that of sensor nodes, and the attacker tries to trace back to the source location in a hop by hop fashion. But when the attacker becomes more powerful, *i.e.* global attacker with a hearing range much more than that of the sensors, the privacy level becomes low and the capture likelihood will be very high.

The idea of using directed walk in routing to provide source location privacy was also adopted in [184]. The proposed scheme (called *LPSS*, Location Privacy Support Scheme) sorts the neighboring nodes of each sensor node in the network into two groups based on the hop count from the sink node; same level or closer. The possibility of selecting one node to be next hop in transmitting event packets from one of these two groups depends on the trade off between delivery delay and location privacy. At the first step, the node randomly picks one group, and later, the destinations will be chosen from the neighboring nodes of another group. In this way, event packets can be sent in a random way towards the sink and make sure that the random walk will not loop back towards the source node. Also, LPSS uses fake packet injection where each node transmitting real event packet will also send fake packets to a randomly chosen node that is farther away from the sink node. This fake packet has a predefined TTL

(Time To Live) parameter that will be decremented each time the packet is forwarded away from the sink. When TTL value reaches zero, the fake packet will be discarded. Obviously, this scheme assumes only local attacks of hearing radius similar to a node transmission range and will fail to support location privacy when attacks have a global vision of network operation and much longer hearing distance. Also, LPSS suffers from high packet delivery latency and energy consumption as each packet is forced to be routed through different random paths.

Inspired by the work presented in [179] and to improve the performance of phantom routing, a two-way random walk (called *GROW*, Greedy Random Walk) is presented in [185] based on rumor routing [186]. GROW implements random walk from both the source nodes and the sink node(s), as shown in Figure 5.5, where the sink node first creates a static random walk (path of receptors) by sending queries. Then, source node generates agents and randomly routes them in the network. When a path of an agent intersects with a path of a query, the agent sets up a path from the source to the sink node. Different from the directed walk, Bloom Filter [187] is proposed to store all the visited nodes in the network for each message to prevent the adversaries from hopping back. Grow attempts to expand the created paths in the random walk as far as possible by choosing the next hop node that did not participate in other paths. This will reduce the probability of random walks staying close to the sources, and will create random paths with non intermediate nodes in common. Also, using two intersecting random walks will guarantee delivering the packets to the sink node with the probability decreases exponentially with time compared with single directed path [188].

Despite these advantages, this design allows the adversaries to recover significant routing information from the received messages from the data stored in the bloom filters, and it only assumes local attacks (specifically the back tracking attack model). In fact, this design is "not realistic" in implementing the routes for large-scale sensor networks. Also, it is worth mentioning that, due to the random walk of every agent and query, the latency of message is more instable.

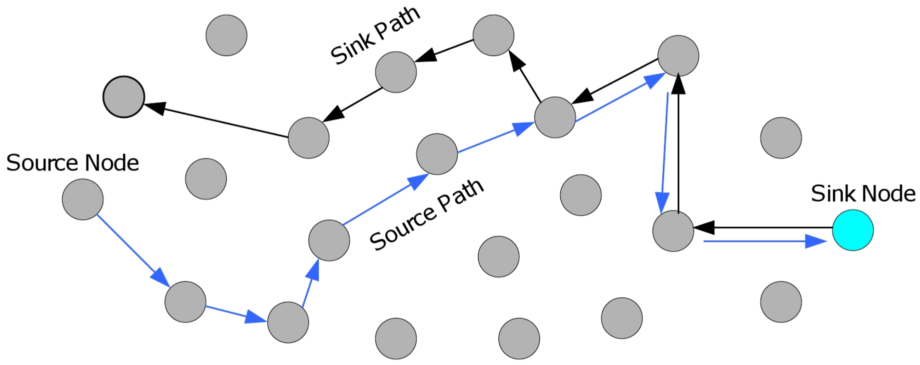


Figure 5.5: GROW Routing.

Instead of using the number of hops as metric in reaching a phantom source, the scheme proposed in [189] uses the distance from a source to randomly select an intermediate node. In this scheme (called *RRIN*, Routing through a Randomly selected Intermediate Node), sending data messages to the sink or the destination node is done in two phases. First, the source node sends its messages to an intermediate node, and second the intermediate node forwards these messages to the sink. In the first phase, the source node selects randomly an intermediate node in its neighborhood based on its relative location and then transmits its messages to that node. The selected intermediate node will be located within a random distance determined by the source node. The intermediate node is expected to be far away from the real source node so that the adversaries cannot track back the real source node. In the second phase, the intermediate node forwards the messages upon receiving them to the sink node normally according to the routing protocol (shortest path for example).

The main advantage of RRIN is that intermediate nodes are randomly chosen from nodes whose locations are at a specific distance from the source, which is one of the limitations of random walks. However, this scheme assumes that the attackers are unable to monitor the entire network. So under global attacks, an adversary can easily track the data sending paths and find the source location. Also, the proposed scheme assumes that the sensor nodes know their relative locations in the network which is a strong requirement that adds more complexity in implementing this scheme.

A modification on RRIN scheme is proposed in [190] where the source location privacy-aware routing is provided through three phases, as shown in Figure 5.6: routing to a randomly selected intermediate node, then routing in a Network Mixing Ring (NMR), and finally message forwarding to the sink node. The three-phase source privacy scheme aims to offer network-level privacy by adding the NMR phase where a large ring is generated in the network around the sink consisting of sensor nodes (called ring nodes). As in RRIN, a source node first sends the packets to an intermediate node and then the intermediate node sends the packets to the closest ring node in the network mixing ring. The packets from all sources will be routed in the mixing ring in a clockwise direction and changed their appearance at every hop along the ring for a random number of times before sending them to the sink. The message forwarding in the ring aims to act as a network-level mix to thwart traffic analysis so that it would be infeasible for the adversaries to distinguish the sources from the message forwarder nodes.
However, this modified scheme still considers only local attacks and it will fail easily under global attacks assumption who can find the source node that firstly initiates the packet transmission outside the ring area. Also, the randomize way of sending a packet from a source to the sink (random way to the intermediate node and packet forwarding along the ring for random times) will definitely increase the end-to-end delay and energy consumption as well as decreasing the network throughput. In addition, ring nodes are more likely to deplete their batteries than other nodes and this will end with isolating the sink node from nodes outside the ring.

A modification on Phantom Single-Path routing is proposed in [191] where inclination angles are used to direct random walks in order to enhance source location privacy. The scheme (called *PRLA*, Phantom Routing with Locational Angel) prioritizes the selection of phantom sources with large angle of arrival to ensure that every random walk gets away from the region close to the source node and hence increasing the safety period. The process of creating the paths starts at the sink node that floods the network with hello messages, so each node in the network can set up the shortest path to the sink. Then the source node floods the surrounding area of certain number of hops in order to let every node to calculate the inclination angles, as illustrated in Figure 5.7, and the transmitting probability based on the distance from the sink. So, based on a

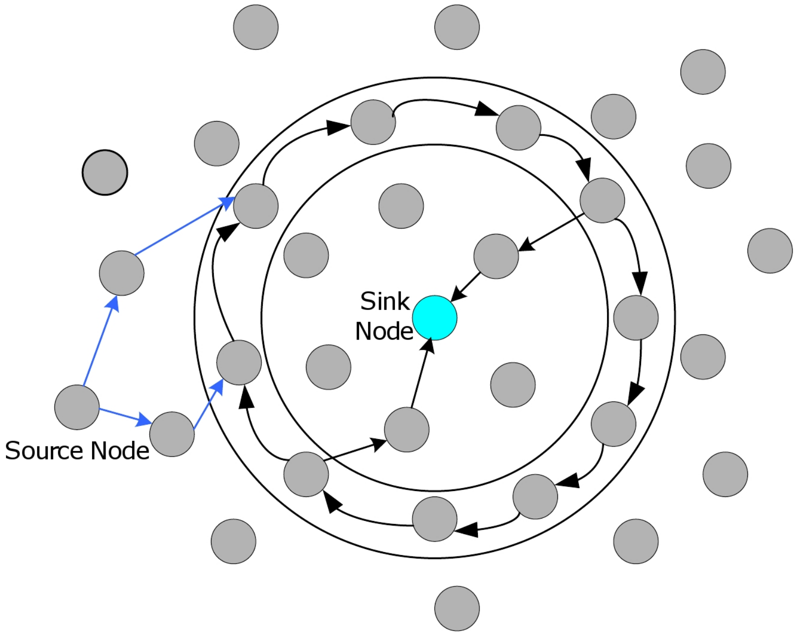


Figure 5.6: Routing through Network Mixing Ring.

certain selected inclination angle, a data packet will be transmitted randomly from the source node for a certain number of hops until it reaches the phantom source. Then the packet will be forwarded to the sink node using the shortest path.
PRLA improves the safety period compared with phantom routing by simply directing random walks based on the angle of arrival. And the path of the directed random walk phase in PRLA could be optimized in terms of number of hops (directly routed to the selected phantom source instead of random walking) while keeping an adequate privacy level. However, it is clear that PRLA assumes only local attackers and cannot handle global attacks that can easily break this scheme and detect the source location. Also we can notice, based on the flooding and random techniques used in this scheme, that the communication overhead will be significant and the message latency will be instable.

In [192], four end-to-end location privacy protection techniques are proposed to protect against a local eavesdropper. The four schemes are forward random walk (FRW), bidirectional tree (BT), dynamic bidirectional tree (DBT), and zigzag bidi-

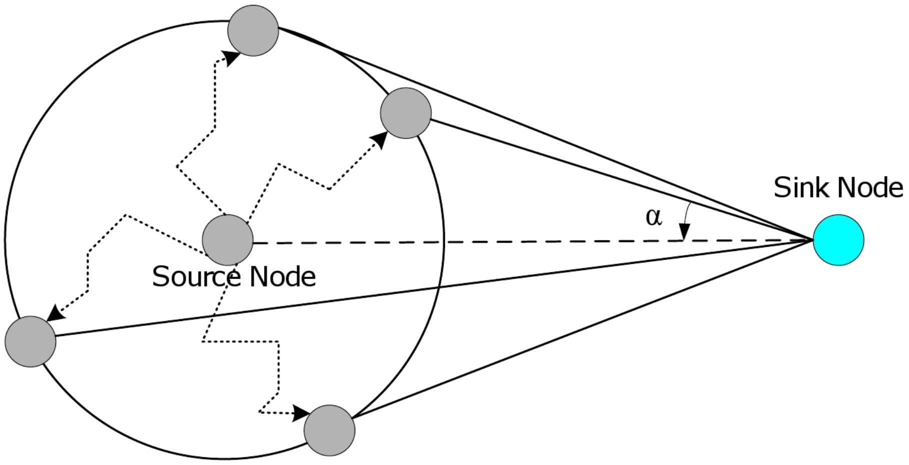


Figure 5.7: Phantom Routing with Locational Angel.

rectional tree (ZBT). The forward random walk approach is similar to one proposed in [184], where every node forwards a data packet to a random neighboring node from closer group. This procedure is repeated at each node until the packet reaches the sink. To increase the location anonymity, the scheme employs bidirectional tree topology where data packets are routed from source to sink using the shortest path and dummy packets are sent from intermediate nodes near both the source and the sink creating branches from the direct path. The dynamic bidirectional tree scheme just combines the above two methods to generate branches of the trees dynamically which can improve the performance. Finally, in the zigzag bidirectional tree, a source first sends its packets to a proxy source (selected random node near the source area) and this node will forward the packets to a proxy sink (another selected random node near the sink area) that in turn forwards the packets to the real sink, making it more difficult for the adversary to obtain the location of the source and sink.

Clearly, the schemes proposed in [192] assume only local attacks with limited view of the network and they will fail to protect the location privacy for both the source and the sink against global attacks with powerful resources. Also these schemes cause extra communication overhead from using dummy packets and increase the end-to-end delay from using random ways (or zigzag) to route the real packets.

5.3.1.2 Fake packet generation/ Periodic collection

In this scheme, source location privacy is provided through periodically sending packets by the source nodes or by the intermediate nodes. These packets can be either real data packets or dummy packets (in case no real packets to send). The mean idea behind this scheme is to perturb the traffic patterns observed by the adversary and hide the real traffic generated by the source nodes. Therefore, both real and fake packets must appear indistinguishable from the attacker's point of view. In order to decrease the communication overhead due to these dummy packets, many solutions were proposed to discard them in some level at the network while keeping an acceptable level of privacy such as using proxies [193; 194], data aggregation[195], authentication process [196], or network coding [197].

An event source unobservability solution against global adversary is proposed in [193], which is based on introducing dummy traffic to hide the real event sources, in combination with mechanism to drop these dummy packets before reaching the base station. In this scheme, each source node continuously sends encrypted packet every period of time (following an exponential distribution) whatever these packets contain real data or dummy messages. If a node detects an event, it postpones sending the encrypted real data to the next probabilistic interval, so that this real event cannot be distinguished from the dummy traffic based on time analysis or rate monitoring attacks. In order to reduce network traffic and drop the dummy messages, two filtering schemes are proposed in [193]: proxy-based and tree-based filtering schemes. In Proxy-based Filtering Scheme (*PFS*), some sensor nodes are selected as proxies to collect and filter dummy packets from the surrounding nodes. The locations of these proxies are determined during network planning and based on local search heuristics. Then, every node selects the proxy which is nearest to it as its default proxy. In Tree-based Filtering Scheme (*TFS*), proxies are organized into tree hierarchy where proxies closer to base station filter traffic from proxies farther away. In both filtering scheme, the proxy will send buffered real data and new generated dummy data at the same rate of transmission.

However, this approach [193] may introduce significant delay to the data transmission process as real data are sent only at specific time based on a certain distribution.

To solve this problem, another scheme is proposed in [198] called *FitProbRate* (Fitted Probabilistic Rate) that tries to send the real data as soon as possible. Similar to the scheme in [193], FitProbRate employs network-wide dummy messages to achieve source location privacy against global attacks. Every node in the network sends out periodically dummy messages with intervals following a certain probabilistic distribution (exponential distribution). In order to reduce the transmission latency, FitProbRate scheme introduces statistically strong source anonymity in which real event messages are transmitted as soon as possible with less disturbed time intervals that could not be detected by using statistical methods. So, real events will be transmitted earlier than the next prescheduled fake transmissions satisfying the following condition: the distribution of the entire message transmissions (fake and real) of each node is "statistically" very similar to the transmission of only fake messages. Statistical similarity is achieved via adopting a statistic test called Anderson-Darling Test to keep the message intervals of each node/cell following an exponential distribution. Hence, the real event message transmission latency is reduced and meanwhile statistically strong source anonymity for sensor networks could be achieved.
The drawback of this scheme is that the consistently transmission of dummy messages, every time there is no real event messages, consumes significant amount of sensor energy, and also it does not include any mechanism to delete these generated dummy messages from the network at any level which leads to increase the network collisions and decrease the packet delivery ratio (unless it is combined with the filtering scheme in [193]).

A different proxy assignment methodology, than the ones presented in [193], for maximizing lifetime of event unobservability in WSNs is proposed in [194]. The proposed scheme protects the network from revealing the sources of the real event by periodic packet transmission combined with dummy traffic filtering at proxy nodes called Optimal Filtering Scheme (OFS). In OFS, proxies can be organized as a general directed graph rather than a tree (as in the case of TFS [193]), so that each proxy can filter packets from every other proxy as well as from normal sensor nodes. As shown in Figure 5.8a, the hierarchy imposed by TFS is restrictive since each proxy in this scheme can perform only a single type of aggregation (or filtering). For example, level-1 proxies (P_1) can just filter f-flows (packets generated by normal nodes) and pass

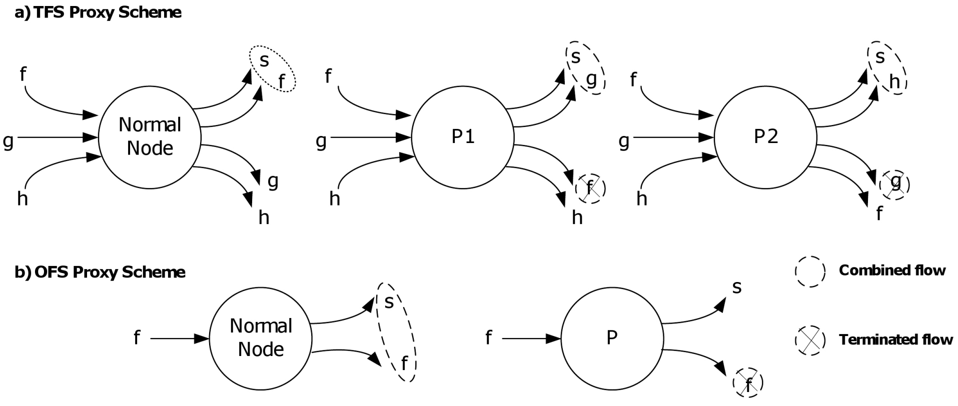


Figure 5.8: In- and Out-flows from Normal Nodes and Proxies under TFS and OFS Schemes.

the other flows (g-flows and h-flows generated by level-1 and level-2 proxies respectively), and level-2 proxies (P_2) can just filter g-flows and pass the other flows. But, in OFS scheme, a proxy (P) can filter out any packets from other nodes/proxies and generates only its flow (S_i) as shown in Figure 5.8b. Therefore, the lifetime increases with OFS scheme because it carries the least amount of data within the network while preserving the event-unobservability, comparing with PFS and TFS schemes.

Another technique for filtering the network wide dummy packets is proposed in [195] where the source location privacy scheme does not require sensor nodes to be placed at the exact predetermined positions to act as proxies, as in [193; 194], but it relies on the aggregation process at cluster heads and filtering at intermediate nodes. This technique based on flexible node filtering and data aggregation, called Aggregation based Source Location Privacy (*ASLP*), aims to further reduction in the amount of traffic in the network that is due to the transmission of fake packets. The overall communication overhead is reduced by making intermediate nodes act as proxies that filter out fake messages or by aggregating multiple messages in a single transmission.
However, ASLP still depends on sending fake packets (in case of no event detection) to hide the location of real sources. Also, source nodes in this scheme need to wait to the beginning of the time interval (following an exponential distribution) to send their data packets, in addition to the delay occurred at cluster heads for waiting a predefined

time period to send packets to their parents.

In [196], another way of filtering the dummy packets used for the source location privacy is proposed based on authentication process. The technique aims to hide the locations of monitored objects from being detected by global attackers while keeping an acceptable level of added transmission overhead. In this technique, called *Periodic Collection*, every sensor nodes independently and periodically send packets whether there are real data or not. At the end of a predefined time period, a sensor node checks if it has any real data packet in its buffer to send. If so, the node encrypts the packet with pairwise key shared with the next hope and forwards it to that node. Otherwise, the node sends a dummy packet without properly authenticating it. When the next node receives a packet, it first verifies its message authentication code (MAC). If correct, the node accepts the packet and stores it in its buffer, otherwise the packet will be dropped. In this way, the traffic pattern in the network will be independent of the presence of detecting events and the sources will be hidden. However, periodic collection, with large sending time period, can only be applied to applications that collect data at low rate and do not have strict requirements on the data delivery latency, while using small time period will increase the traffic and energy consumption in the network.

A distributed algorithm, similar to one proposed in [198], is presented in [199] for maximizing the uniformity of the traffic over the network by mixing the real event data with limited dummy traffic. This scheme, called Source Location Anonymity (*SLA*), assumes that the network has fixed amount of resources to send dummy traffic and try to share it among sensors in order to hide source locations. Two techniques are proposed, constant rate or probabilistic rate packet sending schemes. In these two approaches, limited amount of fake packets are used in order to hide the real event traffic patterns. The probabilistic messaging ensures that an attacker will not be able to identify the source of the messages and match it to a specific source, since even when there is no event, messages are sent to the base station in order to create confusion to the attacker. It is shown analytically that a higher level of anonymity degree can be obtained by increasing the total amount of dummy traffic, and maximum anonymity can be reached after allocating a certain amount of resources for the dummy traffic. However, it has the same drawback found in [198] where there is no specified way to filter the generated dummy packets that are allowed to reach up to the base station.

It was shown in [200] that a global adversary having more precise observation and using more complex tools can perform in a more efficient way to breach strong privacy mechanisms as the one proposed in [198]. The key observation is that, although an adversary might not be able to distinguish between real and fake transmissions, there still exists a source of information leakage that can affect the security of such designs. Recall the design implementation of [198]: In the absence of real events, nodes are transmitting independent identically distributed (iid) fake messages according to a certain distribution with a certain rate. However, the nodes are transmitting real events as soon as possible to reduce delivery latency (earlier than the next prescheduled fake transmission) following the condition that the distribution of the entire message transmissions (fake and real) of each node is statistically similar to the transmission of only fake messages. Based on that, a global attacker monitoring the transmission of a node with no event detection in the vicinity for a certain time interval knows by such design the node has been transmitting fake messages for the duration of that time interval and he/she records the time intervals of sending these packets. And when an event is detected, that node will report location information about this event in its transmission with time intervals follow the design in [198] as explained above. Hence, the attacker will have the ability to distinguish between the time interval (long) when no real activities are reported and the time interval (short) when real event is in the vicinity of the node. By counting the number of short-long inter-delays, as shown in Figure 5.9, the attacker might be able to distinguish intervals containing real events and this will be sufficient to infer private location information, even though the adversary was unable to distinguish between individual transmissions.

In other words, the source of information leakage happens in previous designs because the inter-transmission times during fake intervals are iid's while inter-transmission times during real intervals are neither independent nor identically distributed. In theory, the only way to guarantee that a sequence of random variables is statistically indistinguishable from a given iid sequence is to generate it as an iid sequence with the same distribution. One solution to this problem is to return to the trivial mechanism by having the inter-transmission times during real event intervals same like of the inter-transmission times during fake intervals, but then we will drop the benefit of having

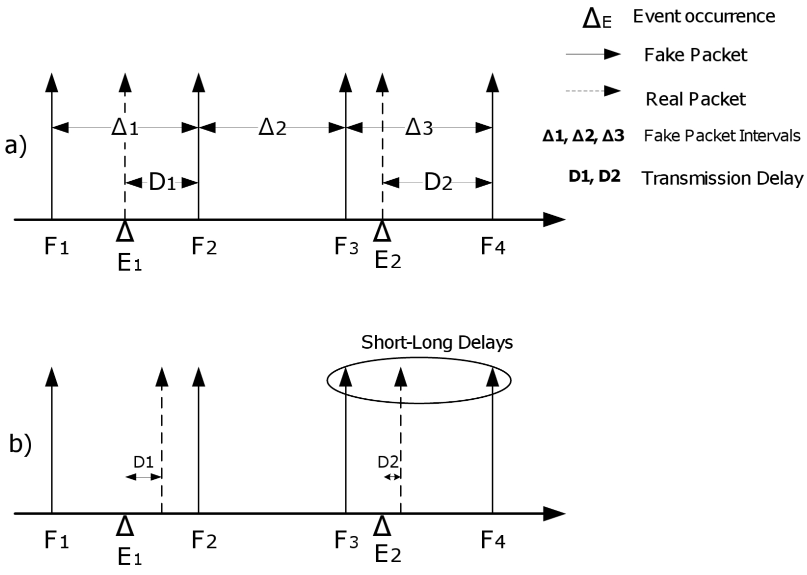


Figure 5.9: An illustration of transmitting real events: a)Real event must be delayed until the next scheduled fake transmission, b) Real events are transmitted as soon as possible with keeping the distribution of fake and real transmissions statistically similar to the transmission of only fake messages.

minimum latency. Therefore, it was concluded in [200] that using statistical framework for source anonymity (SFSA) for designing fake intervals with a distribution that is easiest to emulate during real intervals is the most logical solution.

Filtering dummy packets based on network coding is proposed in [197], where a security scheme for source unobservability is presented (called *SUNC*, Source Unobservability by Network Coding). Similar to the above approaches, SUNC depends on transmitting dummy packets to offer source unobservability and thwart traffic monitoring attacks. However, in this scheme, these dummy packets are specially designed to be absorbed at intermediate nodes based on network coding without the need of having trusted proxies to filter them. The basic idea of this scheme is that the sources send at a certain rate their data packets and if there is no available data they send dummy packets instead. The dummy packets have random contents to prevent content correlation attacks and they are encrypted to avoid analyzing them. The most important property of the dummy packets is the dummy nullity where these packets can be combined with real data packets at intermediate nodes in accordance with the network coding principles without destroying the coded data packets. By this way, dummy packets will be absorbed or disappeared at the intermediate nodes to prevent traffic explosion when combined with data packets without the need of having proxies at some level in the network. Dummy nullity can be implemented by setting the packet plain-text to be null when generating the dummy packet. Notice in this scheme, intermediate nodes do not require distinguishing between the real data and dummy packets as the dummy packets are removed by the use of the network coding. However, the proposed scheme requires high computational overhead (from network coding and homomorphic encryption operations) and high implementation overhead (from the required tight synchronization mechanism).

Another solution is proposed in [201] for designing fake packet sending intervals and it is about sending fake packets (purely) randomly without following a specific distribution. Cryptographic methods are also employed to mask the real identities of sensor nodes. This privacy scheme includes two components: probabilistic message hiding and one-way hash based anonymity. The first component aims to hide the real source event messages in the network by transmitting fake (dummy) messages at a pseudo random time interval (generated by pseudo random number generator). In order to reduce the energy cost of producing these fake messages, every node probabilistically sends fake messages at the start of each random time interval. And in order to minimize the delivery time of real event messages, it assumes that every node can predict the output of pseudo random number generator of the neighboring nodes and thus forwards the real event message to the neighbor that has minimum waiting time. The other component aims to hide the real IDs of the sensor nodes in the network by using one-way keyed hash chain. The first value of this hash chain will be the original ID of a node, and at any time, this node can apply the hash function to its current ID to generate a new ID.

However, this technique does not include any mechanism to get rid of the fake message from the network which will cause high communication overhead. Also, this technique is not suitable for delay-sensitive applications because of the variable delay caused from the waiting time at each node and from the different length paths based on the nodes of minimum waiting time (not the shortest paths). Furthermore, there is a security drawback in this scheme: if we assume that the output of pseudo random number generator can be predictable by a normal node, then an attacker can also predict this number by monitoring a node's packet transmissions and hence the attacker can guess that the node with minimum waiting time will receive a real event packet.

The work of [201] was extended in [202] to propose four schemes for source location privacy: naive, global, greedy, and probabilistic. In the naive approach, every node periodically sends messages whatever they are real event messages or dummy messages. The time interval of these messages is kept quite long to lower the communication cost, but this leads to have long delay for event messages to reach the base station. Therefore, the global and greedy approaches come to improve the naive approach and reduce the delivery latency without increasing the communication cost by taking advantage of using the shortest delivery path. The global approach assumes the knowledge of global network topology and transmission schedules of all sensor nodes and can always discover the routing path that leads to the shortest delivery latency. In global approach, the time interval of sending messages is no longer fixed but random generated by pseudo random number generator, and the source node can compute the path with the shortest overall waiting time by predicting the forthcoming pseudo-random numbers for itself and all sensor nodes. However, the global approach requires, as explained, global information from all the sensor nodes and stored in all source nodes, which will make it very hard to implement. Therefore, the greedy approach is proposed to overcome this issue and try to accomplish the same objective as the global approach but it only requires the knowledge of local network topology and transmission schedules. In greedy approach, the node will select one of its neighbors to send the real event message that has the shortest waiting time and less hop-count to the base station. The probabilistic approach aims to further improve performance by reducing communication overhead without sacrificing location privacy. In probabilistic approach, a node will send real event data (if any) at the end of the waiting period and select the next hope node according to the greedy approach. Otherwise, in case no real event data, a node will send dummy message at the end of the waiting period with a certain probability. This approach assumes that the combined radio ranges of the nodes that send dummy packets can cover the whole network, and the adversary

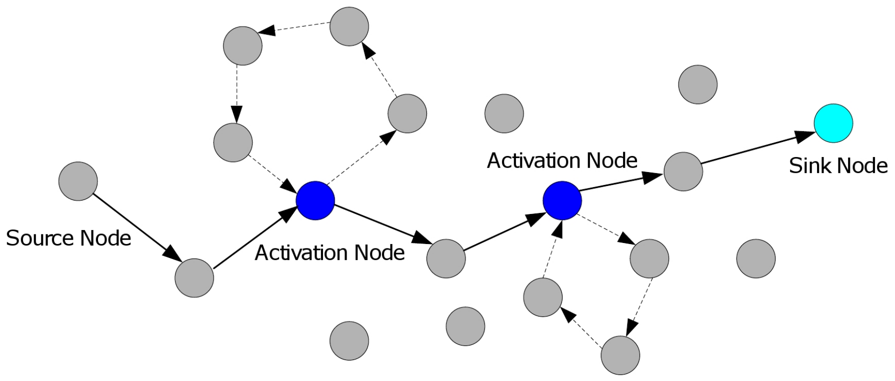


Figure 5.10: Cyclic Entrapment Method.

cannot determine the location of the message's sender by overhearing.

Another way of using fake packets through loops is proposed in [203], called Cyclic Entrapment Method (*CEM*). CEM aims to preserve the performance advantage of shortest path routing while also proving source location privacy. In CEM, several link loops are generated in the network to mislead attackers where each node decides with a probability whether to create loop or not. Each loop consists of several sensor nodes and a loop will be activated when a message encounters it while routed from a source to the sink node. So, when a node participating in a loop received a real message from a source node it will activate the loop by sending fake messages along the loop, as shown in Figure 5.10. A message path is most likely crossing different loops, and hence attackers may be misled by these loops and keep back-tracing on the loops. Although CEM can increase the expected time required for an adversary to locate a source node (*i.e.* it has a good safety period), CEM considers only local (mote-class) attacks and the source location privacy will be enormously degraded when the attacker has the ability of observing traffic in large area. Also, the generated loops in this scheme are fixed and attackers (local or global) can identify them if they record the nodes that they visited. Moreover, producing fake messages along the several activated loops will affect the performance of the network and bring high energy costs.

In [182], a proposed source location privacy introduces a new attack model called "active global attack". This proposal, called Active Global Attack Countermeasure (*AGAC*), assumes that the "passive global attack" model adopted in [193; 198; 202; 204] is not realistic because it assumes that an attacker merely monitors the traffic without taking any action. Under such an attack model, the corresponding countermeasures focus on making all sensors [193; 198; 202] or k sensors [204] transmit dummy messages to disguise the real source location. In general, such approaches are more robust to traffic analysis, but at the cost of higher message overhead. On the other hand, in active global attack model, the attacker is not only a global eavesdropper but also a realistic tracker that devises an optimal route to traverse suspicious spots one by one to find real events, under certain constraints such as time, resource, and event duration. Therefore, based on the observation that [204] was able to guarantee only k-anonymity while [193; 198; 202] were highly demanding in terms of overhead, the solution proposed in [182] is a dynamic source anonymity scheme that seamlessly switches on demand from a statistically-strong source anonymity scheme (*i.e.* [193; 198; 202]) to a k-anonymity scheme (*i.e.* [204]). How to solve the hand off problem in a secure and distributed manner was left as future work.

5.3.1.3 Source simulation

Source simulation is one of the schemes used for source location privacy where multiple candidate traces are created in the network to hide the traffic generated by real objects. These traces can be created by either having false events or selecting one or more nodes to simulate the behavior of real data sources in order to confuse the adversaries. In [204], source simulation scheme is adopted for source location privacy, where different tokens will be preloaded before deployment to randomly selected set of sensor nodes. These tokens will be passed around between sensor nodes to simulate the behavior of real events. In this way, the node that has the token (called token node) will emit a signal as if it is a real object. This will trigger event detection in the local area and generate traffic but for false objects. Although this technique can simulate the detection of real objects with traffic generation from false events, still there is a probability that a global attacker can observe real event sources as the generated traffic is not the same throughout the network and depends on detecting events.

5.3.1.4 Cross-layer based privacy

Source location privacy based on cross-layer solution between the MAC and the routing layers is proposed in [205]. This scheme, called Cross-Layer based Source Privacy (*CLSP*), avoids using techniques that introduce high communication overhead such as flooding or network-wide dummy messages by utilizing beacon broadcast at the MAC layer. The cross layer solution has two phases, as shown in Figure 5.11: MAC-layer broadcasts and direct routing. In the MAC-layer broadcast phase, the source node after detecting an event appends the data description in a beacon payload frame, which is a control message widely used in WSNs for network configuration purposes. The modified beacon frame then will be sent, after encrypting it, to all neighboring nodes. Notice that beacons are sent out regularly - regardless the occurrence of events- at a predefined beacon interval depending on the MAC protocol which essentially forms constant-rate dummy messages without using the network packets. The nodes that receive the modified beacon will decrypt the frame and append its data description on their beacon frames in the same way. The beacon broadcasts process continues for a certain number of hops where the modified beacon reaches a node called pivot node selected by the source node. Then the pivot node starts the direct routing phase where the data description will be routed directly to the sink node using the routing layer protocol. Also [205] proposes another scheme called double cross-layer solution to enhance the source privacy where the pivot node will send the data description packet to a random node instead of forwarding it directly to the sink. Then that random node will initiate another session of beacon broadcasts for a certain number of hops until modified beacon reaches another pivot node that will send it directly to the sink using the routing layer.

The main drawback of this proposed cross-layer solution is that this scheme can only handle data event description (small-size data) in order to exploit the beacon frame broadcasts, and it is not suitable for high data rate applications like data streaming because of the slow beacons rate. If the source data rate increases and becomes higher than the beacons data rate, then the messages need to be buffered in the source node and may causing overflow. Also, using beacons to replace dummy message may increase

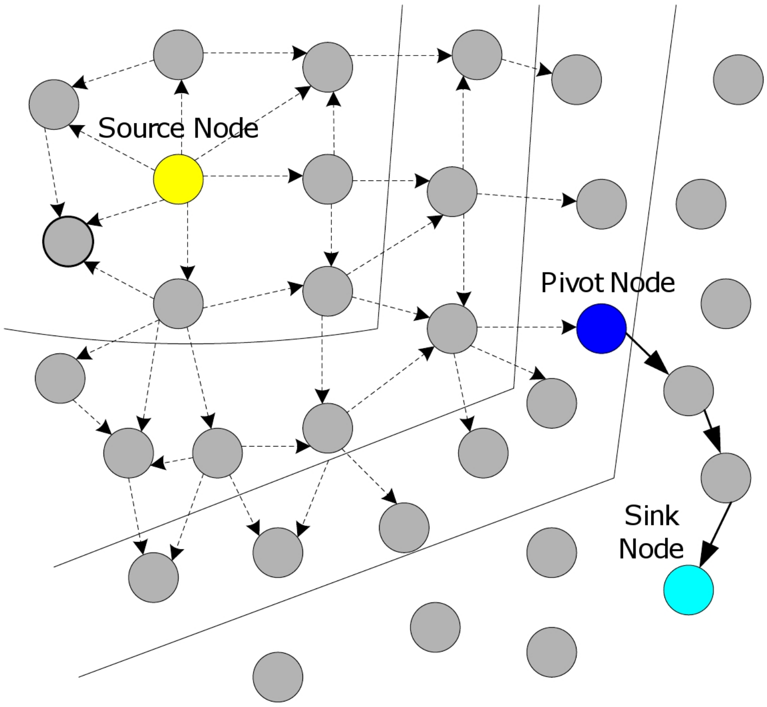


Figure 5.11: Cross-layer based Source Location Privacy.

the delivery delay because beacons are sent out at fix time interval which is usually long period to allow sensor sleep and generate less traffic.

5.3.2 Sink Location Privacy

Sink location privacy in sensor networks has also attracted much attention recently and it deals with concealing the sink location. In WSNs, the sink is not only responsible for collecting and analyzing data, but -typically- it is also used as the gateway connecting the WSN with the wired network. Therefore, unlike the failure of a subset of normal nodes, compromising the sink node can create permanent damage to sensor network and its intended application. In general, adversaries can reveal sink location by performing three traffic analysis attacks [206; 207]: 1) content analysis attacks, 2) rate monitoring attacks, and 3) time correlation attacks.

In *content analysis attacks*, an adversary looks for valuable data in either packet headers or payloads - such as sink ID, hop count from the sink, distance from the sink, etc- which might lead him/her to the sink. One kind of these attacks is proposed in [208] for tracking anonymous sinks by embedding a secure signal (Pseudo-Noise code) in transmitted data packets in order to mark the data traffic in an invisible manner. This signal is carried along with the traffic from the source node to the sink, so the attacker can recognize the location of the sink by tracking this signal. However, in order to track the embedded signal, this attack assumes that the intermediate nodes are only forwarding the packets towards the sink without making any changes on the data packets. In case the intermediate nodes make some changes on the received packets (e.g. re-encrypting the packets with different security keys), the generated packets will be different than the original ones and it will be difficult to track them.

In a *rate monitoring analysis*, the adversary monitors the packet transmission rate of nodes close to him/her self and moves closer to the nodes that have a higher packet sending rate [209]. In a *time correlation analysis*, the adversary observes the correlation in sending time between a node and its neighboring node that is assumed to be forwarding the same packet, and deduces the path by following each forwarding operation as the packet propagates towards a sink node [209].

Protecting sink location privacy can be achieved using different mechanisms, which can be classified into five categories: 1) Privacy-aware routing, 2) Sink simulation, 3) Sink relocation, 4) Fake packet injection, and 5) Controlled transmission rate. In this section, we review the existing privacy-preserving techniques for sink location against both local and global attacks.

5.3.2.1 Privacy-aware Routing

One way of providing sink location privacy is to adopt a privacy-aware routing protocol that can deliver the packets to their intended destination in a way that makes it difficult for an adversary to track them to reach the sink. In literature, privacy-aware routing was built through four approaches: 1) Random routing with hidden address [210; 211], 2) Backbone flooding [196], 3) Directed random walk routing [212], and 4) multiple parents routing [207].

Random routing paths -with hidden address- are established by omitting the destination address (*i.e.* sink ID) from packet headers and force the nodes to forward these packets randomly in the network until they reach the intended destination. In this way, revealing sink location information will be difficult as the packets are not sent directly towards the sink. The implementation of this approach can be seen in [210], where sink location anonymity is provided through hiding the sink address in data packets and using random routing. By removing the destination field from the packet format, attackers cannot find the identity of the sink(s) in captured transmitted packets, and by using random routing, they cannot predict the location of the sink(s) by observing the traffic flow in the network. In this scheme, called Randomized Routing with sink Hidden Address (*RRHA*), the packets are routed from the sources in random paths without having a specific destination until they reach the sink. Before sending a packet, a source node encrypts the packet with a unique symmetric key shared with the sink. The source node then selects randomly any one of its neighbors and sends its packet to it. The receiving node will do the same process and forward the packet to any randomly selected neighboring node. This process continues hop-by-hop until the packet reaches the sink, or it will be discarded if the packet's hop-count reaches a pre-defined value of the random path. In order to increase the chance to reach the sink, the packet can be forwarded to multiple paths and the intermediate nodes can avoid sending the packet to already visited nodes.

Using RRHA scheme can provide good level of sink unobservability especially against local attacks. However, under the assumption of having a global attack that knows the predefined maximum hop-count value, then if this attacker noticed a packet was terminated at a node before reaching this value, then it means that this node most likely is the sink. In terms of network performance, RRHA increases the delay of received packets by using random paths especially in large-scale networks. Also, this approach decreases the packet delivery ratio of the total received packets when packets can be discarded if they reach the pre-defined maximum hop-count value. It is obvious also that the routes found by RRHA will not be optimized in length and many nodes will be visited during packet delivery and this will increase the energy consumption and decrease the network life time. Even though a copy of a packet might reach the sink in one path, in case of using multiple paths, the other copies will still be forwarded in the other paths until they reach the sink or discarded and this will increase the network traffic.

Another work on providing sink location privacy based on random routing with hidden address is presented in [211]. The proposed approach assumes mobile sinks that can move in random movements and collect intended data from neighboring nodes. Similar to RRHA, sensor nodes do not know the location of the sink or its ID number. The sensed data is sent through random paths and stored in random nodes. In this way, an attacker cannot predict the location of the sink from either capturing sent packets and reading the destination field, or observing the traffic flow towards the sink. Before sending the data packets, a source node first encrypts the data via symmetric security key shared with the sink and then selects randomly a neighboring node in order to forward the packet to it. When an intermediate node receives a data packet, it stores locally a copy of the data and then selects randomly another node from its neighbor (not previous hop) and forwards the packet. This process repeats hop-by-hop until the hop-count of the packet reaches a predefined value, then the nodes stop forwarding this packet. In case a node's buffer is full, the node will remove the oldest data to free space for the newly arrived data. In order to collect these stored data, the mobile sink moves randomly around the network and requests the data from the local neighbors. In this way it will be difficult for the attacker to trace the location of the sink or predict its movement.

However, this scheme is not considering global attacks where an adversary can collect and analyze all the communications in the network. In this case, the attacker can easily track the sink because it is the only moving node that communicates with all its surrounding nodes. Also the effect of this scheme on network performance is significant: Storing a copy of data of each transmitted or forwarded packet by the participating nodes requires storage capability that does not fit the hardware limitation of sensor nodes. And in case of reaching maximum limit of buffer size, the oldest data will be deleted which will affect the successful delivery of these old packets or increase the delay for reaching the sink since the sink needs to move towards other nodes who do not delete yet these packets. Also there is no guarantee a packet can reach the sink especially in large-scale deployment if a packet took a random path at one side of the network and sink is moving randomly at another side. In fact, the design of having moving sink(s) for collecting stored data is not realistic for many applications especially in large-scale sensor networks.

A sink location privacy scheme against global eavesdropper based on flooding is proposed in [196]. It is called *Backbone Flooding* where the packets are sent to connected group of nodes (backbone) instead of sending them to a few fake sinks only. The backbone is created after the network is deployed. Upon receiving data packets, the backbone nodes will broadcast these packets to the surrounding nodes. In this scheme, real sink(s) should be located in the communication range of at least one backbone member in order to receive packets from any source in the network.
Although the backbone flooding can support sink location privacy if the backbone is constructed to cover a large area of the network so that it will be difficult for attackers to locate the real sinks, this approach makes the network topology inefficient and often hard to manage, especially for dynamic applications. In addition, this technique requires additional energy consumption in order to broadcast the packets along the backbone nodes to cover a large area of the field.

A sink location privacy based on directed random walk routing is presented in [212]. The Location Privacy Routing, *LPR*, scheme supports path diversity in order to minimize the traffic direction towards the sink and has the ability to tune the trade off between the privacy strength and communication overhead. To implement LPR, each sensor node divides its neighboring nodes into two groups based on the geographical location of the nodes (if available) or on the hop count from the sink node. The first group contains the nodes that are closer to the sink from the sender node, and the second group consists of the other neighbors that are further away (or at the same distance) from the sink node. When a node sends a real data packet, it selects a neighbor randomly from one of the two groups as the next hop. The selection of the group depends on a predefined probability value (P_f from the further group and 1-P_f from the closer group) where P_f has to be smaller than 50% to guarantee every real packet reaches the sink. Following this way, the forwarding direction of the real packets is not always towards the sink and becomes random (instead of directed) one using different paths. However, due to the use of directed random walk, the end-to-end delay and energy consumption are increased because the constructed paths are quite longer than the shortest one.

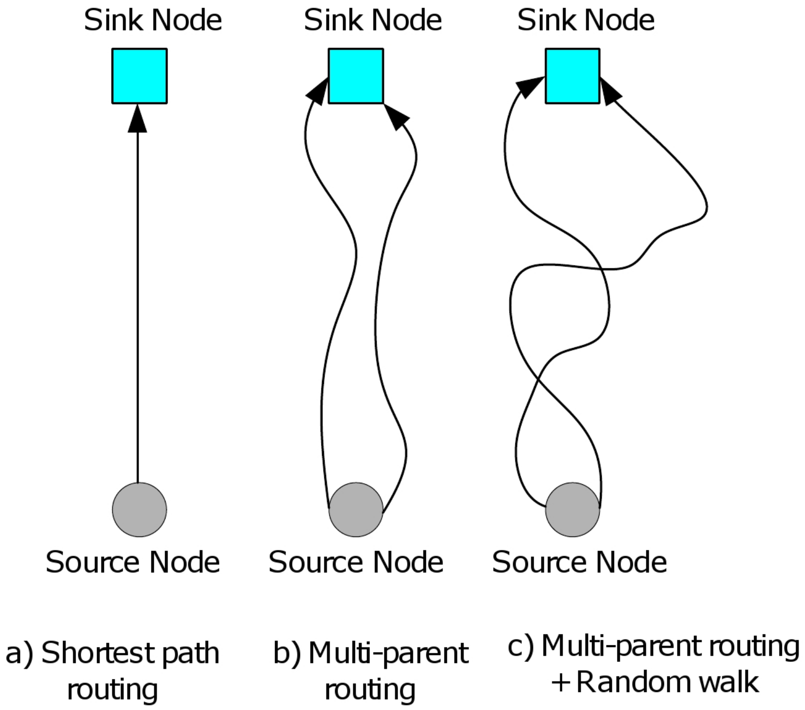


Figure 5.12: Traffic Patterns with Different Privacy Routing Techniques.

A multiple-parent scheme was introduced in [207; 209] to balance the traffic load between parents and children nodes, such that an adversary cannot easily identify which parent node is closer to the sink. In this scheme, a hierarchical routing scheme is adopted in which each sensor node has multiple parent nodes. Then a sensor node selects each time one of its parents randomly to forward the real data packet towards the base station. This multi-parent routing aims to make the traffic pattern more disperse but still looks close to shortest path routing as shown in Figure 5.12. Therefore, a controlled random walk is added to the path of a packet in order to misdirect adversaries. In this random walk, a node will forward its real data packet to one of its parent nodes with a certain probability (P_r); otherwise, it will forward the packet to one of its other neighbor nodes. This way makes it more difficult for the adversary to identify the location of the sink by tracing the data transmission.

5.3.2.2 Sink Simulation

In sink simulation privacy scheme, some nodes simulate (virtual) sinks at specified locations in the network and create multiple candidate traces towards these fake sinks in order to hide the traffic between real sources and real sinks. An example is given in [196] where the proposed scheme selects the location of fake sinks during deployment to be within communication range of real sinks, and the sources are requested to send their data packets only to the fake sinks. Whenever the fake sinks receive packets, they broadcast them locally so that the real sinks can receive them as well. However, by using this scheme, an attacker can easily track the path of sending packets from the real sources up to one-hop from the real sinks. The attacker then just needs to find out which neighbor node is the real sink. The attacker can for example monitor the transmission activity of these nodes; the node that never sends packets in the surrounding area of the fake sink will be the real sink, as the proposed scheme assumes passive sinks (only receive packets). In addition, this scheme requires manual set up before the network starts its operation.

A similar approach is proposed in [213], which is called the *decoy sink* protocol. This scheme aims to hide the sink location against traffic analysis attacks by moving the high communication activity area around the sink node (called hot spot) away from it. This is done by creating a fake base station node in a location away from the real base station. All sensor nodes will first send their data to that fake base station. The fake base station or the decoy sink will aggregate these data readings into summary messages that will then be re-routed to the real base station. This scheme can be further extended to use multiple decoy sinks to increase the randomness of traffic patterns and robustness.
Although this scheme can successfully move the high communication traffic from the area around the real sink to an area away from it around the decoy sink, it does not provide any privacy communication between the decoy sink and the real sink. An attacker based on traffic analysis techniques can reach the decoy sink, and then he can achieve his goal of rendering the network useless by either compromise (destroy) this decoy sink node or track the path between the decoy sink and the real sink using packet tracing techniques. Also, this scheme implicitly requires that the decoy sink node has

processing and storage power similar to the real sink which may not be practical in some applications.

An approach for increasing sink anonymity in WSNs is proposed in [214], but this time the sink simulates the behavior of a normal node rather than a data sink. This is done by allowing the sink to transmit some of the data packets it receives with varying intensity so that an adversary believes these packets belong to ordinary sensor nodes. In this approach, the sink selectively forwards data packets (called BAR packets) with varying TTL parameter to random sensor nodes in its vicinity and in the same manner these packets are forwarded away by the neighboring nodes. So when a node receives a BAR packet, it will forward the packet in the opposite direction of the sink to a random node after decrementing the TTL value. However, this scheme imposes significant overhead that includes the additional control traffic needed for topology management, the extra usage of energy and bandwidth in forwarding packets with useless or redundant payload.

A protection scheme for sink location privacy is proposed in [215] consisting of anonymous topology discovery along with fake packet injection. In this scheme, the anonymous topology discovery aims to eliminate the potential threats against base station during the (periodic) topology discovery phase, while the fake packet injection is used to protect the base station location privacy throughout the data transmission phase. In the anonymous topology discovery phase, base station (BS) randomly chooses a sensor node (e.g. 5 hops away) to act as a *pseudo sink* that initiate the topology discovery process. The selected pseudo BS then initiates the process of network topology discovery by broadcasting periodic discovery packets that will be propagating through the network to all nodes. The nodes between the real BS and the pseudo BS keep the real value of the hop count from the real BS and hence create a tunnel that forwards the data packets to the real BS. Also, the transmitted data packets during the data transmission phase that reach the pseudo BS will be routed to the real BS. Each node sends a real data packet to next hop will send also a fake packet with a specific TTL value to another neighbor to conceal the real packets.

Although this scheme is one of the few works that consider the sink location privacy during the topology discovery as well as the data transmission phase, it has some security breaches that could break its privacy strength. One of these weakness points is that the paper only focuses on the countermeasures against packet tracing attacks and ignores traffic analysis attacks like rate monitoring. Adversaries that depend on monitoring traffic pattern and traffic densities on different network locations can notice that the area around the sink node still has high traffic load and most of the paths are pointed towards that area. Also, the pseudo BS is selected randomly to be within a predefined number of hops from the BS, but this scheme does not ensure that these hops are away from the real BS. Theoretical analysis shows that if the message is routed *h* hops randomly, then it is highly possible that the distance between the start node and the end node is within *h/5*. Moreover, this scheme is not considering global attacks that have the capability of monitoring the entire network and can easily defeat the privacy offered by the proposed scheme.

5.3.2.3 Sink relocation

An approach for concealing the location of a base station is to simply change its location from time to time. If a base station is frequently relocated to a more concealed position within the network, it will become difficult for an adversary to figure out the current location of that base station. It will scrap the adversary's effort to track the base station and force him/her to start again from scratch. An example of this approach is the Relocation for Increased Anonymity (*RIA*) algorithm [214] that chooses the new position of the sink based on two factors: The anonymity increment of the new location, and the impact on network performance. This scheme assumes that the network is divided into cells of group of nodes and the sink node knows the threat level of each cell. The threat level defined as the probability of a cell containing the sink. The sink decides to move to a new location if the threat level of its current cell or its immediate neighboring cell becomes relatively high (compared with a certain threshold). RIA algorithm selects the new location in a cell that has a moderate threat level with enough number of nodes to keep a good level of network connectivity. However, relocation has to deal with at least three issues: determine the best moment in time to move, selecting the next location, and finally moving the sink node in an energy-efficient way. Also, this scheme encounters extra overhead as the forwarding tables in some or all sensor nodes will have to be updated if a sink changes its location. We think that this scheme is probably not feasible in most applications with the exception of military ones.

In [216], another sink relocation scheme is proposed for better sink location privacy with minimal overhead. It assumes that the sink knows the complete topology of the sensor network after the route discovery phase, so that it only needs to determine its new nearest neighbor nodes when it relocates to a new location. After communicating with the new neighbors, the sink reconstructs the new topology, computes new routes, and downloads new forwarding tables that will be distributed to affected nodes. It was shown using this proposal that the number of exchanged packets that are required to build the routing paths -after a sink is relocated- is less than the number of packets exchanged if the route discovery protocol is used again.

5.3.2.4 Fake Packet Injection

Fake packet injection scheme is introduced to prevent an adversary from identifying the real data transmission pattern directed towards the sink and hence hide the sink location. A protecting scheme for sink location privacy is proposed in [217] based on injecting fake packets. The proposed scheme focuses on protecting the sink location privacy against packet tracing attacks by routing the real packets through the shortest path while the fake packets will be routed to some random destination and some fake sinks. As a result, the path diversity is provided so that it will be difficult for an attacker to distinguish the real packets from the fake packets, and hence the chance of finding the real sink by packet tracing attacks is reduced. Also end-to-end delay of real packets is not affected as they are using the shortest paths. In this scheme, the intermediate node that receives more than one packet from different sources will be considered as an intersection node. After receiving a certain number of packets, these nodes will inject a dummy packet towards a fake sink or random node per each received real packet. The location of the fake sinks are determined before deployment and selected to be far away from the real sink.

Although this scheme has a better safety period and performance comparing to other schemes, it considers only packet tracing attacks only and ignores other kind of attacks such as rate monitoring attacks. Also, this scheme assumes only local attacks whose detection radius is equal to the transmission range of sensor nodes. Under global traffic analysis attacks, the sink location privacy provided by this scheme will be degraded as the traffic around the sink location will be clearly noticed.

To keep a balance between energy consumption and sink location privacy against traffic analysis attacks (particularly the packet rate monitoring and time correlation attacks), two algorithms were added to support multiple parent routing [207]. For better energy efficiency, P_r value (the probability of a node forwards its real data packet to one of its parent nodes) is set typically over 0.5, and the result will be having the possibility a node forwards a packet to one of its parent nodes higher than the possibility it forwards the packet to any one of its other neighbors. Therefore, to solve this problem and to prevent time correlation attacks, two fractal propagation methods were proposed to generate fake packets and propagate them in the network in order to introduce more randomness in the communication pattern. When a node hears that its neighbor node is forwarding a real packet to the base station, the node generates a fake packet with a certain probability (P_c), and forwards it to one of its neighbor nodes. These fake packets having a predefined TTL value are then routed along random fake paths. P_c is controlled according to the data forwarding rate; the higher the rate is, the lower is P_c. The other method aims to hide the high communication area around the sink node by introducing several random areas of high communication activity (hot spots) to mislead the adversary to believe it as the sink node. These hot spots are created by transmitting fake packets into random paths with a higher probability towards areas that have forwarded fake packets in the past. Notice that these algorithms assume that the adversaries have a traffic view over a limited surrounding area with a single attacker gradually moving based on local decisions towards areas of higher traffic until reaching the sink node. Therefore these countermeasures cannot stand against attackers that have global information about the whole network. Also, these algorithms rely on creating fake packets to obscure the communication patterns towards the sink. This leads to have more communication overhead and energy consumption.

A similar approach, LPR, is proposed in [212] using a location privacy routing protocol along with fake packet injection. The proposed scheme aims to make the directions of both incoming and outgoing traffic at a sensor node uniformly distributed so that it will be hard for an adversary to locate the location of the sink node using locally gathered information. After dividing the neighboring nodes into two groups, closer and further groups, and constructing the random walk -as explained before in section 5.3.1.1- fake packets are injected towards the opposite direction of the sink in order to minimize local information exposed to adversaries. Each time a node forwards a real data packet to a next hop; it also transmits a fake packet to a neighbor node that is randomly chosen from its further group. Each fake packet has a predefined TTL value specifying the maximum number of hops it will be forwarded away from the sink. However, LPR assumes the hearing radius of the adversary is equal to the sensor transmission range which means that it considers providing sink location privacy against local attacks only. Attacks that can monitor the all traffic in the network can notice the traffic load around the sink location and hence find it. Also, LPR only takes into account the packet-tracing attacks and ignores the rate monitoring attacks that can be used to deduce the location or the direction of the sink by monitoring the traffic densities at various locations in the network. In addition, the proposed scheme has a degrading performance (high end-to-end delay and energy consumption) as the paths used to deliver the real packets are random and can take long routes without having a clear mechanism to avoid creating path loops.

In order to provide countermeasures against both traffic analysis and packet tracing attacks for sink location privacy in WSNs, a scheme called *Maelstrom* is proposed in [218]. The basic idea of Maelstrom is to create pre-assigned several maelstrom areas (hot spots) in the network, which serve as sinks for only fake packets generated by sensor nodes. Fake packets are used to protect the sink from packet tracing attacks while the created maelstrom areas that simulate the high communication area generated around the real sink will protect the sink from traffic analysis attacks. After constructing the routing information in the network and each node knows its hop-count from the sink, the sink starts creating the maelstrom areas by selecting the nodes that will behave as the center of these maelstrom areas (fake sinks). This is done through sending certain number of configuration packets by the sink node evenly to its neighbors. These configuration packets will be forwarded away from the sink node using the constructed routing information for a pre-defined number of hops. The node that receives any of these configuration packets in their last hop will be the center of a maelstrom area. After selecting the centers of maelstroms, each of them will initiate the process of building routing information and constructing paths to reach them by other sensor nodes just the same what the sink did before. In this case, each node in the network now has its routing information of the real sink to deliver the real data packets as well as the routing information to the nearest maelstrom center node to forward the fake packets. Similar to the routing approach used in [212], a node sends or forwards a real data packet to one of its neighbors in the closer set with a certain probability (*p*), otherwise it will send the packet to one of the neighbor nodes in the equal list with a probability of (1-*p*). Any node forwards a real data packet to the sink will also send a fake packet to the nearest maelstrom center node using the same way of routing a real packet. Notice here that fake packets are sent to certain destinations rather than to random nodes as in other fake packet injection methods. This way avoids the traffic volumes from different directions canceling each other, and creates high communication traffic areas from gathering fake packets.

Notice also that, this scheme does not hide the hot spot generated around the real sink but it creates other hot spots in different areas in the network to trap the attackers that are tracking sending the packets from the sources and lead them to fake destinations. However, it is possible that in some cases traffic analysis attackers located near the sink area can detect the high traffic area around the sink and find it. Also, this scheme is assuming local attacks with limited hearing range comparing to normal nodes and not considering powerful adversaries with global view. A global attacker can still observe the high communication area around the sink node as well as around the other fake centers and may find the sink node with a high possibility. Furthermore, a global attacker using traffic analysis method, after monitoring the network, can notice that the area around the real sink has higher traffic volume than other areas generated by gathering fake packets. The real sink node collects real packets from whole the network while other maelstrom centers collect fake packets only from near surrounding nodes.

5.3.2.5 Controlled transmission rate

The inherent traffic pattern in WSNs facilitates an adversary to find the base station because the primary flow of traffic is from all sensor nodes towards the BS over a relatively fixed multi-hop path. As data packets are transferred from source nodes towards the BS, their paths merge when they get near the BS. Hence, the nodes close to the base station have to send comparatively more packets, not only their own data but also relay data from nodes further away from BS. Therefore, the surrounding area around BS features a high transmission rate and increased traffic flow.

In [206], the problem of providing base station location privacy against external adversaries in sensor networks is discussed through multi-path routing and fake message injection. This scheme proposes a secure multi-path routing to multiple base stations to resist against isolation attacks (e.g. DOS, spoofing, and jamming attacks), and anti-traffic analysis strategies to help disguise the location of the base station from global eavesdroppers. Multiple paths construction process starts at the base station by broadcasting request messages to the surrounding nodes. These messages are authenticated using a one-way hash chain to prevent attacker from sending forged request messages. Nodes that receive the request messages from a base station will record the senders of these packets as their parent nodes for that base station and broadcast them again to other nodes. Then each node verifies its neighboring nodes using shared pairwise keys and broadcast cluster key. The proposed scheme then uses anti-traffic analysis mechanisms to hide the traffic pattern and prevent attackers from analyzing the traffic to discover the location of the base station. The anti-traffic analysis mechanisms include hiding the packet destination address, de-correlating packet sending time, and controlling packet sending rates. Hiding the packet destination address is done through encrypting the data packet content along with the destination address field using the appropriate shared security keys. De-correlating packet sending time is done through randomly delaying the sending time in each cluster where every node is assigned a slot and randomly chooses a time within its slot to send its packet to its parent. Controlling packet sending rates is done through sending packets at a certain rate where every node in the network has to transmit messages at a constant rate to its parent and if there is no available data packets to send, dummy packets will be injected.

The drawback of this scheme is that it does not consider attacks with global view that can monitor the whole traffic in the network and assumes only attackers with hearing range close to the normal nodes. Therefore, the fake packet injection method used in this scheme can help in controlling the traffic rate of the sensor nodes but not hiding the traffic direction towards the base station. Notice that in this scheme, fake packets are injected to the next hop towards the sink node using the real data paths and are not routed towards other fake paths. With respect to network performance, this scheme introduces extra delay for delivering packets from using this traffic rate model, in addition to extra energy consumption from using fake packet injection method.

5.3.3 Query Privacy

In this section, we address the problem of ensuring query privacy in situations where queries are used (usually issued by a third party) to retrieve the collected data in the network. Query privacy includes hiding who initiates the query and which node matches the query, but does not cover securing the requested data itself because this is covered by data query security, as shown in our taxonomy of security protection in WSNs (Figure 5.1).

So far, it has been often assumed in traditional WSNs that network operator and sensor owner are the same entity, and the collected data by sensor nodes is ultimately destined for the owner of the network represented by a base station or a sink. In this approach, users interested in sensed data can issue data queries to sensor nodes through the base station which in turn forwards query results from sensor nodes to the users. However, in large-scale network or in hazardous environments this might not be the case. If queries are issued by many users, then sensor nodes around the base station will lose their energy quickly since they always participate in relaying data to and from the base station. In addition, the base station may become a bottleneck node representing a single point of failure.

Therefore, another approach might be used where users are allowed to freely roam in the sensor network and directly access (query) sensed data without involving the base station. In this approach, multiple users and entities would often collaborate although they might not trust each other, and collected data would usually be queried on demand by third parties (users). Therefore, as individual sensors can be subject to queries, we face two privacy issues: (1) queries might not be willing to disclose their initiators, and (2) queries do not want to reveal which sensor is being interrogated.

Revealing such information will lead to disclose node's query interests and/or expose sensitive information to other users or eavesdroppers. For example, in medical WSN, if an adversary knows that queries have been frequently issued to specific sensor nodes that cover a patient's house, then the adversary can infer that the health of the patient is getting more attention probably due to some health problems. Investigating the trade off between protecting the user location privacy and queried location privacy has been studied in [219], where a simple metric was proposed based on K-anonymity (queries) method allowing a user to know how well his/her selection of K nodes protects the privacy of the area of interest (location of queried sensor nodes). In addition, a quantitative measure of how much information the K queries leak about the user's location was defined.

The intuitive approach to address the query privacy problem would be for the user to always query the entire set of sensors and select from the collected information the interested data. This would achieve perfect privacy but would result in a great waste of energy. Another solution would be to constantly collect all sensor readings in real time, store them at some data storage nodes or servers and allow clients to query the data from them.

This can be seen in data-centric wireless sensor networks (DCS), as in [220], where query privacy support is addressed by storing the data of the same attributes (event type, geographic location, or sensed time period) in certain storage cells (group of nodes) as shown in Figure 5.13. The sensing data is forwarded from the detecting cells to the storage cells based on a mapping function such as Geographic Hash Table (GHT) [221]. The proposed privacy-enhanced DCS network, called *pDCS*, offers different levels of data privacy based on different cryptographic keys. So, even if an attacker can compromise a sensor node and obtain all its security keys, he cannot decrypt the data stored in the compromised node (as the data is encrypted by the detecting nodes, not by the storing nodes). In addition, several query optimization techniques are proposed based on Euclidean Steiner Tree [222] and Keyed Bloom Filter [187] to minimize the query overhead without losing any query privacy. In pDCS scheme, the detecting cells first determine the location of the related storage cells through one of the three proposed keyed mapping functions (Group-key-based, Time-based, or Cell-based mapping). In this way, the network is defended against mapping attacks and an attacker cannot determine the mapping from the detecting cell to the storage cell. Storing the data in other cells than the originating cells protects the network from the tampering (readout) attack since the storage cells do not possess the decryption keys.

pDCS offers different levels of location privacy based on different cryptographic keys and allows a trade off between privacy and query efficiency. Although pDCS addresses access control by authenticating network users before granting them data access rights, the privacy of users is not considered. In addition, using Euclidean Steiner Tree scheme for query optimization will increase the size of the packet with many header fields which leads to consume more energy in transmitting these packets. Moreover, there is another issue regarding data aggregation in DCS network after using the proposed privacy scheme: The node that takes responsibility of doing data aggregation for correlated data needs to have a full access on the transmitted data and this is done only by having the same security keys used for encrypting the data at the detecting nodes, which is not the case in this scheme (unless the scheme uses, for example, homomorphic encryption for secure data aggregation [223]).

Disconnecting the mapping between a user identity and the query issued by that user was introduced in [224]. In this scheme, called DP^{2}AC, each user interested in sensed data purchases some tokens from the network owner in order to send a query with an unspent token to any sensor node. Once validating the token, the sensor node provides the user with an appropriate amount of requested data matching with the denomination of the token. Such a token not only controls access to the sensed data, but also hides the user identity through using blind signatures. The use of tokens in conjunction with blind signature leads to a desirable property: the validity of each token can be verified by any sensor node, and no one including the network owner can tell the identity of the token holder. In this way, the network owner can prevent unauthorized access to sensed data, while users can protect their data access privacy.

However, as reported in [225], due to the use of blind signatures, each query cannot be signed or authenticated by the user. As a result, an adversary can easily intercept

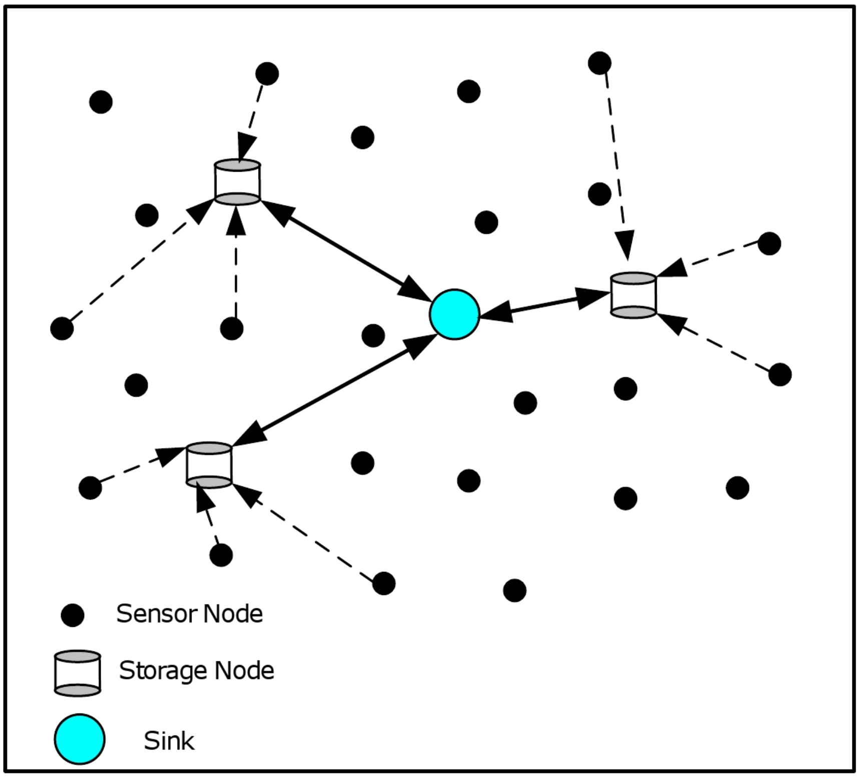


Figure 5.13: Data-Centric Wireless Sensor Networks (DCS).

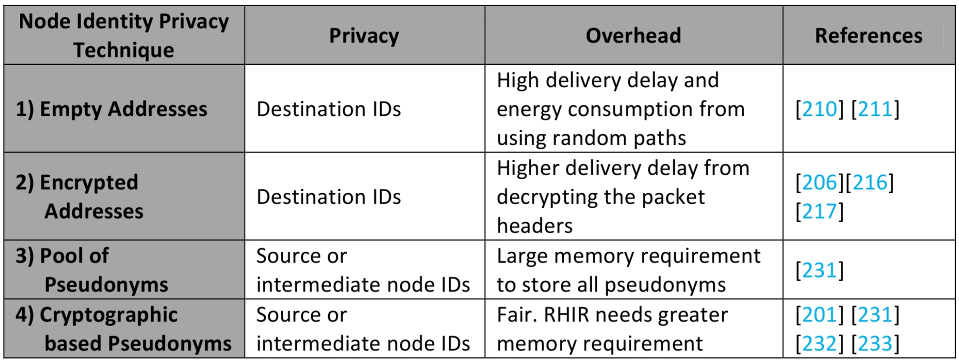
the token and impersonate any authorized network users to modify the query command and then obtain the responses from sensor nodes. Also, network-wide flooding is required once token-reuse detection runs, and the scheme needs to store tokens in every node local memory which will not be suitable with resource constrained network. Moreover, using a token for one time only for sending queries will restrict the number of user queries allowed in DP^{2}AC.

The scheme presented in [226] attempts preserving the privacy of clients querying sensor networks, through untrusted servers by hiding the identities of the queried sensors and the relationships between individual queries from the servers. Hiding the identities of queried sensors is done by target-region transformation technique. The main idea of the transformation function is to map one region into multiple regions, such that the target region cannot be distinguished from the other uninteresting regions. Multiple transformation functions were used in [226] such as uniform, randomized and hybrid. These transformation functions are similar to the k-anonymity algorithms (e.g., [227]) which hide the target's real identity using other (K-1) similar objects so that it is impossible for the adversary to distinguish the target from the (K-1) other objects. But the cost of such an anonymity-based technique is high in a WSN, because query dissemination and data collection in the uninteresting regions consume a large amount of energy. Also, this scheme uses two-server approach with a routing scheme for data transmission similar to onion routing [228; 229] and this leads to privacy compromised if the two servers collude. Furthermore, each sensor needs to store a key for each user which is not efficient in terms of storage requirement.

Another query privacy scheme in WSNs relates to the privacy of the ID of queried sensors is addressed in [230]. This scheme allows an external user to collect readings from sensors of his interests without leaking their identities to adversaries including the network operator. However, the protocol provided relies on onion-routing like solution, hence introducing a non-negligible burden on both computational and communication. Furthermore, the level of privacy achieved still needs to be fully validated.

5.4 Node Identity Privacy

Cryptographic mechanisms used in protecting the data content of sensor packets keep the confidentiality and the integrity of the payload (readings) of these packets, but not the headers. Cryptographic mechanisms usually are not applied on the header of the packets in order to allow the intermediate nodes to process these packets (if needed) and route them towards their intended destination. However, there is much relevant information contained in the packet headers and can be exploited by attackers to reveal important data from the network. Source, sender, and destination IDs are examples of these important information that are sent clearly (in plain text) in packet headers. An eavesdropper, after capturing a sufficient number of packets and reading the node identities, might be able to produce a logical network map that could be related to the physical node locations. For example, if the eavesdropper gets to know the ID of an important node in the network such as the sink node, he/she can generate packets destined to that node and then sends them in the network. Then from tracking these packets, which will be routed towards the specified node, the attacker may find the



Node Identity Privacy Technique	Privacy	Overhead	References
1) Empty Addresses	Destination IDs	High delivery delay and energy consumption from using random paths	[210] [211]
2) Encrypted Addresses	Destination IDs	Higher delivery delay from decrypting the packet headers	[206][216] [217]
3) Pool of Pseudonyms	Source or intermediate node IDs	Large memory requirement to store all pseudonyms	[231]
4) Cryptographic based Pseudonyms	Source or intermediate node IDs	Fair. RHIR needs greater memory requirement	[201] [231] [232] [233]

Table 5.1: Comparison among Different Node Identity Privacy Techniques

location of that node and capture it.
Therefore, protecting the node identity is one of the important issues in providing network privacy in WSNs, as shown in Figure 5.2. Node identity privacy techniques are explained below along with their advantages and drawbacks, and a comparison among these techniques is shown in Table 5.1.

One common way that could provide the node identity concealment is omitting the ID fields in the packet header format and keeping them blank. In this approach, called *Empty Address*, sensor nodes do not know the location of the sink(s) or its ID address, and the packets are sent through random paths until they reach the intended destination, and/or stored in random nodes and base station moves to collect the data from these nodes [210; 211]. Obviously, using random paths to reach the base station will increase the latency of delivering packets and the energy consumption from extra transmissions. Some schemes use *Encrypted Address* in order to hide the node ID by encrypting the packet header as well along with the packet payload [206; 216; 217]. For example, in [206], every node encrypts the destination ID, packet type, and the contents of the packet with its cluster key. However, the current sender's ID remains in plain-text so that the receiver node can choose the correct cluster key to decrypt the packet and know the destination ID. The main overhead of this technique is the need for a receiver to decrypt the header of a packet to check the source and next hop addresses, and consequently the time needed to deliver a packet from a source to sink node will be increased.

Another common way of providing node identity privacy is using *Pseudonyms*. A pseudonym is a name or identifier that can be used instead of the real node ID. These pseudonyms need to be variable and random in a way an attacker cannot relate an ID to a specific node, otherwise using fixed pseudonyms will have the same effect (no protection) of using real node IDs. For that reason, several mechanisms have been proposed to periodically renew pseudonyms:

Two anonymous schemes for clustered WSNs are proposed in [231] to provide node identity privacy. In this work, secret keys are shared between sensors and base station, and the nodes in each cluster are considered indistinguishable. Anonymous routing is performed by using pseudonyms that are generated by one of the two schemes. The first scheme is a simple identity anonymity scheme, called Simple Anonymity Scheme (*SAS*), which provides every node with a randomly distributed set of pseudonyms from a network-wide pool of pseudonyms. Each node will select one a pseudonym for its ID in each transmission to a specific neighbor. The second scheme, called Cryptographic Anonymity Scheme (*CAS*), aims to overcome the drawback of the first scheme which is the large amount of memory needed to store the complete pseudonyms space. CAS solves this problem by using cryptographic anonymity method (keyed hash function) in order to generate the new pseudonyms for each transmission.

However, the scheme in [231] assumes that the shared security keys are not compromised and sensor nodes internally store all the material to generate fresh pseudonyms. In case an attacker is able to compromise the nodes, he/she might be able to easily obtain past and future identifiers. To counter this problem, an identity privacy scheme proposed in [201] aims to hide the real IDs of the sensor nodes in the network by using one-way keyed hash chain. The first value of this hash chain will be the original ID of a node, and at any time, this node can apply the hash function to its current ID to generate a new ID.

Similar approach [232] proposes a node identity privacy scheme based on keyed hash chains. In Hashing-based ID Randomization (*HIR*), every node shares a pairwise secret key with its neighbors and the new pseudonym are generated for the new message by hashing the previous ID. This created keyed hash chain makes it more difficult for an adversary to obtain the old pseudonyms. To further reduce the risk of node compromise attacks, Reverse Hashing-based ID Randomization (*RHIR*) is used that firstly creates the one-way hash chain during deployment and then uses it in reverse order [232; 233]. In RHIR, the attacker cannot compute the next ID that will be used by the node even if the attacker compromised the node and knows its current ID and the hash key because the hash chain is used in reverse. The downside of RHIR is the greater memory requirements since the hash values are stored and not discarded until they are used.

5.5 Temporal Privacy

Temporal privacy provides the protection of the information concerned about the existence of detected events and their time occurrence. In other words, one form of contextual attacks is to figure out whether an interesting event occurred in the network and when it happened. An adversary can know these information, no matter how strong the encryption scheme used, by simply monitoring passively the traffic pattern in the network.

For example, in animal-hunter scenario, if an animal passed through the network and it was detected by the surrounding sensor nodes. These nodes will report that event to the sink node. Consequently, this will change the traffic pattern at the place where the animal is detected. Now if an adversary is able to associate the origin time of the packet, then the adversary will be able to track the animal's movement and use that information for hunting. This type of attacks depends on the assumption that the needed time delay of transmitting an event packet through the network is fixed (*i.e.* the time needed for a data packet passing through intermediate sensor nodes along a certain routing path is the same every time and depends mainly on the number of hops). Therefore, when an adversary notices a change in the traffic pattern indicating an event detection occurrence, he/she can get the arrival time of an eavesdropped packet and then deduct from it the average delay encountered in that route.

In order to protect the network temporal privacy of event detection, two parameters should be taken into account in designing temporal privacy scheme: traffic pattern and packet delay, as shown in Table 5.2 that compares between the temporal privacy

Temporal Privacy Technique	Privacy	Overhead	References
1) Event Occurrence	Independent traffic pattern	High energy consumption from sending periodic dummy packets	[206]
2) Event Time	Variable packet time delay	Higher delivery delay and required higher buffer space.	[234][235]

Table 5.2: Comparison between Different Temporal Privacy Techniques

techniques. As the change in traffic pattern is an indication of an event detection incidence, then one way to hide this leakage of information is to keep the traffic pattern unchanged during network operation or independent of the presence of real objects. This can be done through periodically sending packets by sensor nodes at a reasonable frequency regardless of whether there is real data to send or not [206]. Obviously, the traffic pattern will be independent from the behavior of real objects in the field.

Another way to hide the time of detecting real events is to make the packet time delay variable and not predictable by adding some random delay during packet transmission so that an adversary cannot accurately estimate the original generation time of the message. This temporal ambiguity of the time of detecting event would introduce spatial ambiguity also as the object moves and make it harder for the adversary to track it. This technique is useful for delay tolerant application where timely delivery of data packet is not important. The Rate-Controlled Adaptive Delaying scheme (*RCAD*) [234] follows this direction and provides temporal privacy by buffering the data locally for a random period of time, according to an exponential distribution, at the intermediate sensors located along the routing path. However, buffering packets at intermediate nodes may lead to the requirement of large amount of buffer space at each node, especially for large-scale sensor networks. The trade off between the protection of temporal privacy and the efficiency of buffer space is an important issue and discussed in [234] where buffer preemption technique is included to handle the problem of overloaded buffers. The delay distribution will be adjusted as a function of the incoming traffic rate and the available buffer space, and when the node buffer is full, this technique chooses a message to be transmitted immediately without further delay.

Other similar techniques are presented in [235] where a countermeasure is proposed to prevent traffic analysis attacks based on the temporal patterns (like packet time arrival) by maximizing the entropy of the packet inter-arrivals using additional exponential random delays.

5.6 Discussion and Open Issues

In this section, we discuss some interesting issues that appeared during reviewing the existing solutions proposed for privacy preservation in WSNs:

A trade off between privacy degree and network performance: From the review of the existing works in event unobservability in WSNs, one can see that the benefit of providing privacy protection usually comes at the cost of other network performances such as time delay, successful delivery, and power consumption. Hence, there is a trade off between the level of provided privacy and other performance metrics, mainly the end-to-end delay and energy consumption: A higher privacy degree (*i.e.* larger safety period) and better protection against both local and global attacks lead to (using existing proposals) increase considerably time delay and energy consumption.

For example, in proposed privacy schemes [193; 194; 196], dummy packets were proposed to be sent by every sensor node all the time, so there would be no way for an adversary (whatever local or global) to determine which ones are reporting real events. Although in this case the source location privacy is perfectly protected, the energy cost is very high and the lifetime of this sensor network would be very short. In contrast, privacy solutions such as [203; 217] that propose data packets always follow a single path to the sink with minimum use of dummy packets have the minimum possible time delay and energy cost. However, event privacy offered by these techniques is relatively weak with respect with the capability of the attacker and it can be easily broken. Therefore, successful practical solutions should be designed to achieve the objective of event unobservability while in the same time maintain a minimum extra communication overhead in order to guarantee the delivery quality (*i.e.* delivery latency and delivery ratio) of transmitted data and prolong network lifetime.

Phantom single-path vs. phantom flooding routing: Both of phantom single-path and phantom flooding routing have the same first phase of random walk until the packets reach the phantom source. Then, in phantom single-path, the packets will be forwarded from the phantom source to the sink through a single path (usually the shortest path). But, in phantom flooding, the packets will be forwarded to the sink by flooding them into multiple paths. As one would assume, both the single (shortest) path and flooding methods can achieve a 100% delivery ratio and -although it might not be very intuitive- they have the same end-to-end packet delay; as the first packet arrived through flooding will be routed via the shortest path (the shortest path is always contained in a flooding). However, the phantom single-path still has better network performance in terms of energy efficiency because it uses less number of transmissions. In order to make phantom flooding more energy efficient, probabilistic flooding was introduced where only some nodes forward the packets based on a certain probability distribution. In this way, probabilistic forwarding improves the energy efficiency, but does not necessarily deliver every packet to the sink node (sacrifices the 100% delivery ratio). Regarding the privacy preservation performance, both techniques assume only local attacks with limited coverage and have similar performance against backtracking attackers starting from the sink node (tracking the shortest path). However, a single-path phantom routing, besides significantly reducing the energy consumption, provides a higher privacy protection level (against attackers located in different positions away from the sink location) since the resulting single paths originated from the phantom sources will avoid the hearing range of the adversary more easily than a flooding-based method.

Privacy-aware routing vs. network-wide fake traffic: Regarding providing location privacy, we notice that most of privacy-aware routing-based solutions, such as [183; 184; 185; 212], assume only local attacks whose coverage area are relatively similar to normal sensor nodes. But, in the presence of a global adversary who is able to monitor the traffic of the entire network, routing-based solutions have been shown to leak private information, as explained before, besides their effect on network performance. Therefore, works in [193; 200] argued that the intuitive approach to report a real event without revealing to a global adversary its location information is to force nodes to transmit fake packets even if there are no real events to be reported. When real events are detected, they can be reported within the transmissions of these fake packets.

Fake packet transmission strategies: However, this fake packet generation approach should take into account when to send the real event within the dummy packets in order to make them undistinguishable with each other. Sending real data as soon as detecting them (or when they arrive) does not completely solve the location privacy problem because they can be still observed by global attackers using statistical analysis. If sending the real data are postponed to the next scheduled fake message, then we will have either high latency or high energy consumption depending on the transmission scheduling rate: If the transmission scheduling rate (pre-specified probabilistic distribution or deterministic) is slow, then the latency will be high and it will not be acceptable to a delay-sensitive application. If transmission scheduling rate is fast, then nodes will be sending packets more frequently and this leads to shorter battery lives and higher collision. So, one solution to this problem is to transmit independent identically distributed (iid) fake packets according to a certain distribution with a certain rate. But, nodes will transmit real event messages as soon as they can (earlier than the next prescheduled fake transmissions), with keeping the distribution of the entire message transmissions (fake and real) of each node statistically similar (using statistical goodness of fit tests) to the transmission of only fake messages [198]. Hence, the real event message transmission latency is reduced and meanwhile statistically strong location anonymity for sensor networks could be achieved.
However, as we explained before in section 5.3.1.2, a strong global attacker having more precise observation and using more complex tools can breach this statistically privacy mechanism [200]. For a certain time interval, this attacker has the ability to distinguish between the time interval when no real activities are reported and the time interval when real event is transmitted by a node, as illustrated in Figure 5.9.

5.7 Conclusions

In this chapter, we analyzed in details all the aspects of having privacy preservation of the contextual information in wireless sensor networks along with the research challenges carried in this filed. We have surveyed the state of the art of wide range of solutions proposed to countermeasure different kind of privacy attacks. Based on this study, we have introduced a complete taxonomy of security protection and event unobservability techniques in WSNs. We explained most of the proposals used in each technique showing their advantages and drawbacks in terms of their network performance efficiency and protection capability against different level of attacks. More specifically, we first surveyed the location privacy problem for the source, sink, and query location and reviewed most of the proposed privacy-preserving techniques. Then, we analyzed the protection of node identity privacy and explained its state of the art approaches. We also examined the temporal privacy issues and addressed the existing schemes related to this subject. After drawing a complete picture for event unobservability in WSNs, we propose, in next chapter, a source/sink location unobservability scheme for WMSN that hides the location information of important nodes in the network such as sources and sinks.

Chapter 6

Source/Sink Location Unobservability in WMSNs

6.1 Introduction

Although most of the security mechanisms used in sensor networks such as encryption, authentication, and intrusion detection allow sensor nodes to protect their transmitted (data content) from being exposed by external and internal attacks, and satisfy most of the needed security requirements (such as confidentiality, authentication, integrity, and availability), still they cannot fully address the location privacy of contextual content in WMSNs. Therefore, a third level of security protection is needed to offer contextual privacy of location information for important nodes such as source and sink nodes.

Among the different types of privacy, as mentioned in the previous chapter, location privacy is of special interest and it is an important security issue in WMSNs. This is because location privacy plays an important role in WMSNs in preserving location information of critical sensor nodes, such as event sources, storage hops, and sink nodes. Lack of location privacy can expose significant information about the entities in the field and the traffic carried on the network.

[1] Chapter 6 is based on the publications:
An Efficient Source/Sink Location Unobservability for Wireless Multimedia Sensor Networks; Islam T. Almalkawi, Manel Guerrero Zapata, Gamal N. Al-karaki will be submitted to *Journal of ..., 2013*

However, providing event unobservability in sensor networks is challenging task and should take into account the trade off between privacy efficiency degree and network performance: A higher privacy degree (*i.e.* larger safety period) and better protection against both local and global attacks lead to (using existing proposals) increase considerably time delay and energy consumption. Therefore, our proposed location privacy scheme aims to exploit the joint design between the source coding techniques in the application layer and the multipath packet transmission in order to provide a strong privacy level against both local and global attacks, while in the same time maintain network performance efficiency without having to use a network-wide dummy packets.

6.2 Network and Attack Models

1) Network Model

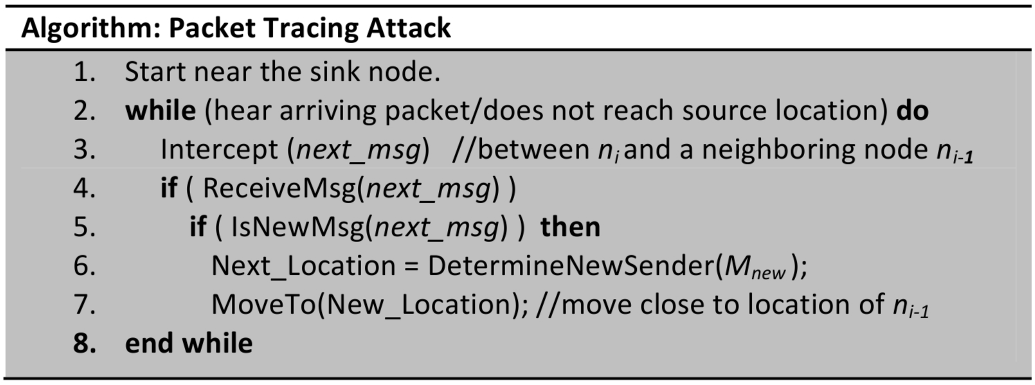
In our proposal, we consider a WMSN consisting of a number of sensors deployed in a specific area, and following an architecture based on the single-tier clustered architecture model [113]. In this model, the network is divided into clusters using a clustering routing algorithm, such as the one detailed in [170]. Each one of these clusters contains different types of sensor nodes (heterogeneous nodes, called group members GMs) such as camera, audio and scalar sensors in addition to a powerful node acts as a cluster head. These nodes have a certain limited transmission range for wireless communication that allows the group member nodes to exchange messages directly with their cluster heads. Cluster heads have more resources, are more powerful than GMs, and are able to perform intensive data manipulation and in-network processing such as aggregation and data fusion. Cluster heads have the same limited transmission range and transmit packets on hierarchical multi-hop fashion through other cluster heads to reach the destinations (sink nodes).

The clustering algorithm starts from the base station (BS) that sends periodically broadcast messages called (*BS-Msgs*). The nodes that receive those messages and satisfy certain conditions -such as certain number of hops from BS, SNR, BER, RSS- will act as 1^{st}-level cluster heads and inform BS using acknowledgment messages (*Ack-Msgs*). 1^{st}-level cluster heads also will start broadcasting messages called (*CH-Msgs*) to the surrounding nodes. Depending on certain criterions (based on the clustering algorithm), the nodes that receive *CH-Msgs* -except BS- can be either cluster heads of lower levels or group members (GMs) of the 1^{st}-level CHs. In the same way, these nodes will acknowledge their CHs or parents about their joining using *Ack-Msgs*. This process continues in the same manner to build the network until all nodes join the network and determine their rules, i.e., cluster head or group member, and all possible paths are found.

2) Attack Model

We consider providing location privacy against *global* eavesdroppers who aim to identify the source and sink nodes and attack them. By a global eavesdropper, we mean an attacker who can monitor and access the whole traffic in the network. Moreover, the attacker is assumed to be *external* who can only monitor communication channels among the nodes and will not compromise or control any node. Also, the attacker is assumed to be *passive* who cannot conduct active attacks such as event triggering, packet injection, channel jamming, or denial of service attacks.
More precisely, adversaries try to determine the location of a source/sink node by: 1) simply examining the content of an event packet to find any useful information leads to the source/sink ID or location, 2) tracing back the multi-hop packet transmitting to reach their sources in case the packets are encrypted as shown in Figure 6.1, 3) conducting rate monitoring attack to observe the high transmission rate area which is usually around the sink node, or 4) performing time correlation attacks to notice the correlation in packet sending time between a node and its neighbor to deduce a forwarding path toward the sink node.

All exchanged messages are assumed to be encrypted with secret keys shared among authorized nodes, and thus attackers cannot readout the content of messages even if they intercept on communications. A key generation and management are adopted as the one described in [236]. The sink node is assumed to be trusted during all network operation time and thus it can keep all the security keys in its memory.



Algorithm: Packet Tracing Attack

```
Start near the sink node.
while (hear arriving packet/does not reach source location) do
   Intercept (next_msg)   //between $n_i$ and a neighboring node $n_{i-1}$
   if ( ReceiveMsg(next_msg) )
      if ( IsNewMsg(next_msg) )  then
         Next_Location = DetermineNewSender($M_{new}$);
         MoveTo(New_Location); //move close to location of $n_{i-1}$
end while
```

Figure 6.1: Packet Tracing Attack.

Cluster heads have the ability, through using the appropriate security keys, to decrypt the data packets sent by their group members or lower level cluster heads and then encrypt them with different keys in order to change the appearance of the packet content and prevent packet tracking attacks.

6.3 Source/Sink Location Privacy Scheme

In this section, we propose an event unobservability scheme for WMSNs that provides location privacy for both source and sink nodes. The proposed event unobservability scheme is based on a joint design optimization between the application layer functionality and packet transmission operation. The privacy scheme exploits the source coding mechanism used in the application layer (to compress the size of multimedia data) in a way that allows a node to send periodically real packets only at predefined times. A source coding technique, such as Layer Coding (LC) or Multiple Description Coding (MDC), splits the original multimedia content into multiple independent and different important streams. Each one of these streams (called layer or description) alone provides an acceptable size of low quality version of the original multimedia data, and thus it can be transmitted (inside a packet) through the network. This technique can be used in conjunction with multipath routing approach to achieve load balancing and an acceptable level of QoS requirements (considering the available resources such as bandwidth and data rate). At the destination side, the base layer or one description stream packet can give a low quality (coarse version) of the desired data, and combining again all or subset of the higher-quality layers (or more descriptions) together achieves better quality and higher resolution (fine version).

Using these principles of source coding techniques in controlling packet transmission, a source node (active node) can continually send real data packets at a certain scheduled period. Because the source node will keep sending the encoding multimedia data (layers or descriptions) of a captured image, then it will start transmitting the processed data of the subsequent captured images in a certain rate. So having this behavior of streaming multimedia processing in mind, the event occurrence information can be hided from attackers since most of the nodes in the network are periodically sending data packets. The rest of the nodes (passive nodes) that are not participating in sending or forwarding packets in any path can inject dummy packets at a certain time period similar to the sending interval of real packets. These dummy (fake) packets are similar to real data packets in size and they are encrypted, so that the content cannot be revealed and distinguished from real packets. For energy efficiency and for reducing the traffic load in the network, the dummy packets will be terminated at nodes that have real data in their buffer to send.

Then for providing global source location unobservability in WMSNs, we propose our privacy scheme that works as follows. Every active source node in the network sends out its consecutive real data packet -after encrypting it- with intervals following a certain kind of distribution, e.g. constant or probabilistic. The source node does not only send the real data packets to the intended next hop destination(s) toward the sink -in case of using multiple paths- but also to all neighboring nodes. The next hop nodes forward these real data packets -after re-encrypting them- at time intervals of the same distribution toward the sink following the routing paths. Other nodes that receive real data packets and they are not intended to receive them (not on their path to sink) will simply drop these packets and continue sending/forwarding their own data (if any). Passive nodes will emulate the behavior of sending real data by transmitting encrypted dummy packets to all neighboring nodes at the same time intervals, with the difference that these fake packets are not forwarded by any active node.

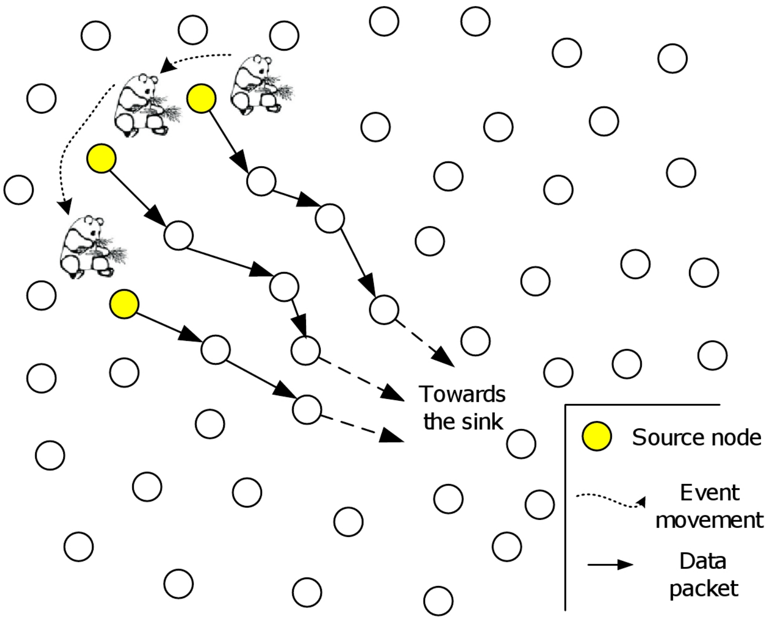


Figure 6.2: Movement Pattern Leaks Location Information.

One of the advantages from exploiting the streaming output from the source coding technique is having a uniform traffic pattern during network operation that is independent from event occurrence without using a network-wide fake packet injection. Only limited amount of fake packets are used to conceal the traffic in the entire network, and most of exchanged packets are real data that can be used to increase the perception quality of received images and videos at the sink. As soon as the sink is satisfied with the quality of a certain multimedia data from a certain source or when a source is capturing new event observations, then that source node can stop sending successive layers or descriptions of the old data and start sending packets from the new data. Notice also that the generated dummy packets by passive nodes are not propagating throughout the network and will be dropped as they reach active nodes, and thus there is no need to employ a proxy mechanism to filter them before reaching the sink node.

Figure 6.2 shows an example of a sensor network that aims to detect the movements of a panda animal. As the panda moves from one place to another, the sensor node that is located in nearby area and detects this movement will trigger an event detection in its local area and start sending its event observation towards the sink node. The generated traffic of event detection by these source nodes can be easily tracked through packet tracing attacks or observed by a global adversary. Recall the packet tracing attack: Upon receiving a packet, the attacker moves toward the sender node whose location is determined by frequency localization techniques. This process continues hop-by-hop until the attacker reaches the source node or stops in case the attacker does not hear any new incoming packet. However, by using our source location privacy technique, this kind of attacks will not be valid any more since every node in the network will be receiving packets from all other neighboring nodes repeatedly at predefined times. Thus there is no clue which path of them is the direction towards the real source node. Even that if we assume a global attack who is able to eavesdrop and analyze all the communication in the network, cannot infer the location of a source node by observing the first node initiates the communication with sink node, because the sources are also receiving packets from other nodes and the traffic pattern is independent of the presence of detecting events and hence the sources will be hidden.

Our proposed location privacy scheme can be extended to provide sink location unobservability also. One way the global attacker can determine the location of the sink is to identify the region exhibiting a high number of transmissions at high data sending rate, which is called *hot spot* area. As we know in multi-hop network transmission, the nodes close to the sink node have to relay more packets to it than the nodes far from the sink as shown in Figure 6.3. Our privacy scheme leads to have a uniform traffic pattern throughout the network, which is independent from event detection where all the nodes are transmitting their packets periodically at certain times, and this will eliminate creating hot spot area around the sink node(s).

Also, the sink node in our privacy scheme will imitate the behavior of a normal sensor node so it will be difficult for an attacker to find it. This is done by having the sink node transmitting packets also to some of its surrounding nodes. These packets are similar to the normal real data packets and sent following the same periodic sending distribution.

Notice that, an adversary cannot reveal the sink location (or finds any useful contextual information) by using content analysis attack because all the exchanged packets

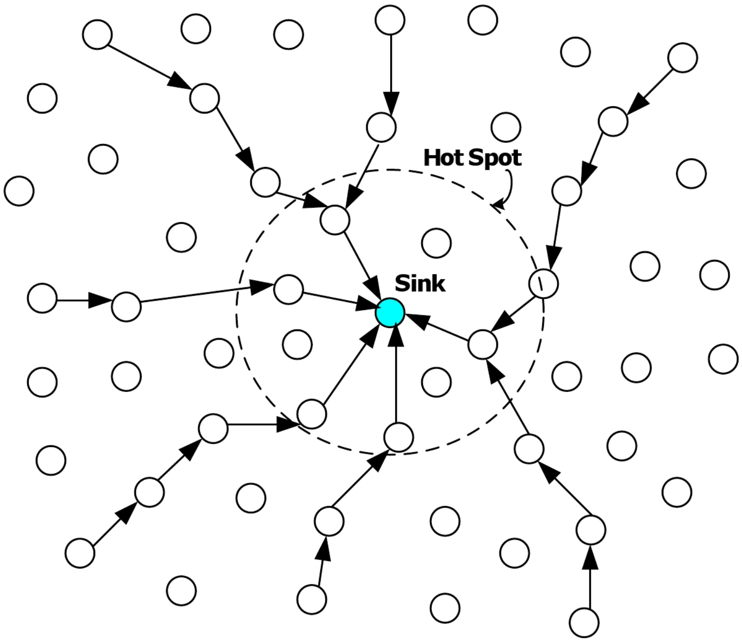


Figure 6.3: Hot Spot Area around the Sink Node.

in the network are completely encrypted (both headers and payloads). Therefore, the adversary cannot look for valuable data in these packets - such as sink ID, hop count from the sink, distance from the sink, etc- which might lead him/her to the sink. In addition, the attacker cannot disclose the location of the sink using rate monitoring attacks, since all the nodes in the network are sending their packets (either real or dummy) at a certain rate following a constant or probabilistic distribution. Moreover, our privacy scheme prevents threats that try to determine sink location by time correlation attacks. Attacker cannot observe the correlation in sending time between a node and its neighboring node, which is assumed to be forwarding the same packet, as the nodes are transmitting their packets at predefined time regardless the time of receiving them. Also a node is not only sending the packets to its next hop node(s), but also to all of its neighboring nodes. These nodes (neighboring nodes) are also transmitting packets periodically, so that there is no way by which an attacker can deduce the packet propagation path towards the sink node by following each forwarding operation.

6.4 A Mathematical Model for Hierarchal WMSN

In this section, we will briefly present a mathematical model for packet transmission rate (data rate) and its effect on the appearance of hot spot area in WMSN. The model consider sensor network with continuous sending packets (simulating the behavior of streaming data). In this network, source nodes transmit their data at certain intervals toward a base station in multi-hop fashion. Intermediate nodes, besides transmitting their own data (if any), will also forward data packets from other nodes at certain intervals.

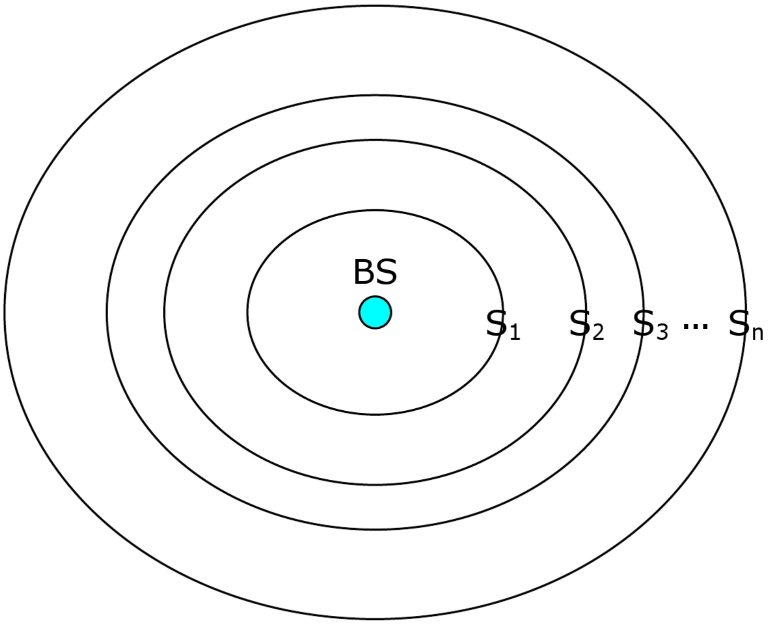


Figure 6.4: WMSN Partitioned into Spheres.

If we assume that the total number of nodes in the network is N and the base station is deployed in the center of the network. Then, we can divide the total number of nodes into circular regions of non-empty subsets around BS as shown in Figure 6.4: S_0, S_1, ..., S_n. Where N = S_1 U S_2 U ... U S_n, $S_i \bigcap S_j = \Phi$ for all $i \neq j$. So, the first round area surrounding the BS is deployed with S_0 nodes and the next spherical strip around the BS will be deployed with S_1 number of nodes, and so on. The circular strip area Sn contains only leaf nodes and each one of them transmits exactly one data packet in each interval. The nodes in the sphere area S_{n-1} forward the packet received from the leaf nodes in S_n as well as transmit their own data packets (one per interval). Corresponding to the partial union of the circular areas S, we use the notion of b_i, where $b_i = S_1 U S_2 U ... U S_i$.

Now, we can represent the data transmission rate at each specific circular area S_i as follows:

$$\alpha_i = \frac{N - b_i + S_i}{S_i} \tag{6.1}$$

Where $N - b_i$ denotes the total number of nodes outside the area b_i, hence, the total number of packets received by the nodes in the spherical area S_i in each time interval. Therefore, the nodes in S_i must forward/transmit $(N - b_i + S_i)$ packets at each interval; the packets received from the outer circular area bi plus their own data packets S_i. We assume here that the used routing protocol is equally distributing the packets in multipath through the nodes in each spherical strip area, thus having the denominator S_i.

As a result, the leaf nodes in S_n are generating data packets at a certain data rate equal to α_i, where $\alpha_n = \frac{N-b_n+S_n}{S_n}$. Notice that in this case $N - b_n = \Phi$, hence $\alpha_n = 1$ which confirms the fact that the leaf nodes are sending only their own data packets at a rate of one packet per interval.

However, the nodes in the spherical area S_1, which surrounds the BS, will have to forward many packets transmitted by all other nodes in the network in the same time interval. This high transmission rate area (α_1) appears as a hot spot area and can be easily identified by adversaries looking for the sink location by using rate monitoring attacks.

$$\alpha_1 = \frac{N - b_1 + S_1}{S_1}, \quad \alpha_1 \gg \alpha_n \tag{6.2}$$

Therefore, in order to eliminate the appearance of the hot spot area around the sink node, we need to decrease the number of packets transmitted by each node at every interval in this area, hence reducing the data rate value and making it closer to the outside area's data rate so that the attackers cannot notice the differences. This can be done by enlarging the denominator S_1 in eq 6.2 either by increasing the number of nodes deployed in the hot spot area, or by increasing the surrounding area S_1 around the sink to include more nodes inside.

The first solution requires the need of manual deployment of nodes in the area around the BS, and this requirement is not feasible in many applications. On the other hand, the second solution sticks with the initial node deployment distribution and it can be adopted during routing establishment phase. So, if we consider using our proposed routing protocol *CMRP* [170], then the nodes in the hot spot area are the 1^{st}-level cluster heads. And in order to increase the number of 1^{st}-level CHs, we need to adjust the value of *Thr-High* to allow more nodes accept received *BS-Msgs* and act as 1^{st}-level CHs around the sink node, thus decreasing packet transmission rate by these nodes:

$$\alpha_1 = \frac{N - b_1 + S_1}{S_1 + K} \tag{6.3}$$

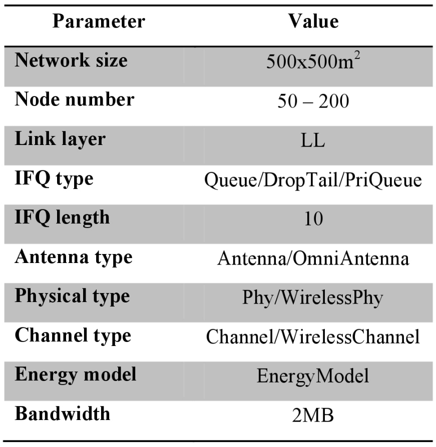
Where K is the number of additional nodes inside the spherical area S_1 from adjusting the value of *Thr-High*.

6.5 Experimental Evaluation

IN this section, we evaluate the performance of our proposed Sink/Source location unobservability scheme using NS-2 simulations. The considered network has an area of 500m × 500m and it is deployed with number of sensor nodes ranging from 50 to 500. The sensor nodes are randomly distributed in randomized grid and the sink is located in the center of the network. We are using a constant traffic rate (CBR) of 600 packets/sec and packet size of 316 bytes. Table 6.1 lists the other parameters used in our simulation environment.

6.5.1 Safety Period

The privacy conservation level is measured by the number of packets the source node has sent before this node or the destination node is found by an attacker. This privacy level can be measured by what so called safety period. The commonly used strategy by previous work, such as Phantom and probabilistic flooding, to enhance the safety period was increasing the number of steps required to reach the source node from the



Parameter	Value
Network size	500x500m^2
Node number	50 – 200
Link layer	LL
IFQ type	Queue/DropTail/PriQueue
IFQ length	10
Antenna type	Antenna/OmniAntenna
Physical type	Phy/WirelessPhy
Channel type	Channel/WirelessChannel
Energy model	EnergyModel
Bandwidth	2MB

Table 6.1: Simulation Parameters.

base station or vice versa. In this case, the safety period probability of the source's location with a distance *d* from the sink node and after *h* random walk steps will be given by [237]:

$$P = 1 - e^{\frac{-d^2}{hwalk}} \tag{6.4}$$

We can see from above equation that the safety period for a certain location depends on the distance from the sink, which is not variable if assuming fixed locations, and the number of steps of the random walk. Therefore, larger random walk length may improve the safety period of these schemes and hence their privacy level, but the packet latency will be higher. On the other hand, by using our proposed scheme, the safety period is increased by keeping a uniform traffic patter throughout the network while in the same time maintain the optimized routes between the source nodes and the base station. Having a uniform traffic pattern that is independent from event detection in the network leaves no way for attackers to deduce the correct path to the source or sink node, thus having the maximum possible safety period as shown in Figure 6.5.

6.5.2 End-to-end Delay and Paket Delivery Ratio

One of the important quality of service requirements for multimedia delivery in sensor networks is the end-to end delay along with packet delivery ratio for having a good

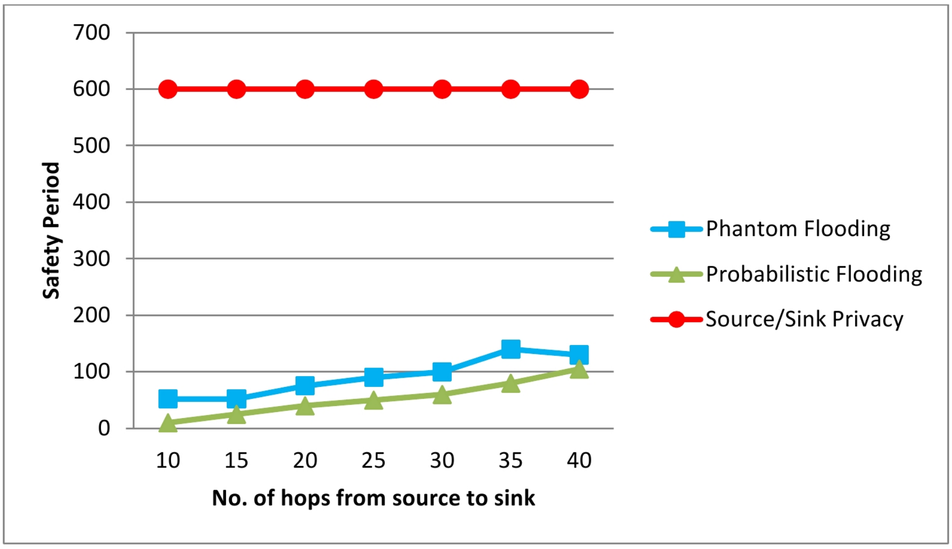


Figure 6.5: Safety Period of Our Privacy scheme Compared with Others.

quality perception of received data. Therefore, any proposed privacy scheme for WMSNs should take into consideration not to sacrifice severely the network performance in terms of end-to-end delay and packet delivery ratio for having a good privacy level. Otherwise, the proposed privacy scheme will spoil the intended main purpose of the network which is successfully delivering of multimedia content.

Our proposed location privacy scheme does not affect the operation of the routing protocol or change the constructed optimized paths between the sources and the base station. Even though the source nodes (and then the intermediate nodes) transmit the data packets to all neighboring nodes, the optimized route (found by the routing protocol) is still contained and the packets will reach the intended destination directly. This will result in obtaining 100% of packet delivery ratio that the routing protocol can achieve without using a privacy scheme.

With respect to the end-to-end delay, our proposed privacy scheme will not introduce extra latency from changing the routes, using random walk, or adding loops. Our privacy scheme will keep using the optimized paths, as we described before, which guarantees obtaining minimum hop-count paths with better link quality offered by CMRP comparing with existing routing protocols. However, our location privacy scheme forwards the data packets towards the sink hop-by-hop at predefined time following a certain distribution (We use constant rate in this simulation). This packet transmission mechanism may add buffer delay at each hop in order to have a uniform traffic pattern in the network, but in case using the same constant traffic rate of the routing protocol then the end-to end delay will not be affected as shown in Figure 6.6.

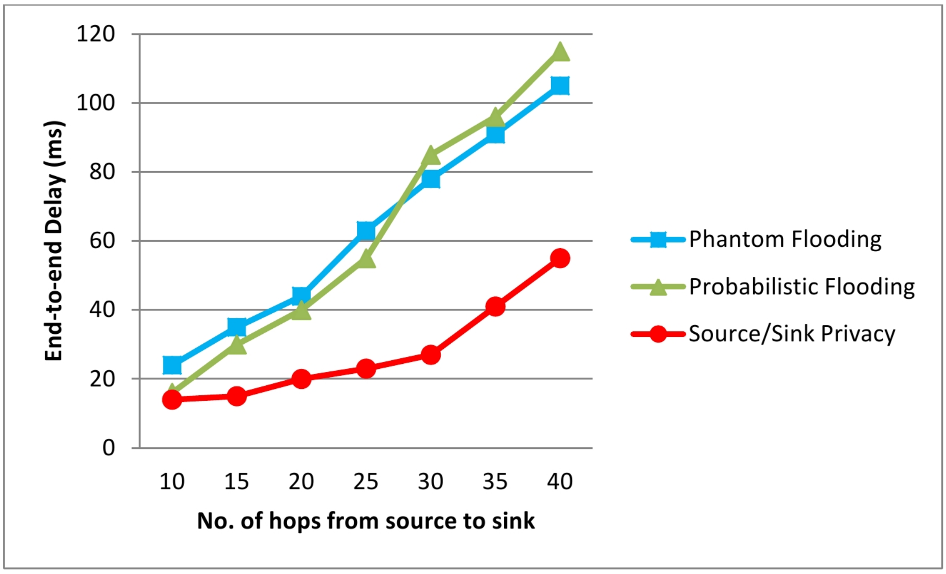


Figure 6.6: End-to-end Delay of Our Privacy Scheme Compared with Others.

6.6 Conclusions

In this chapter, we introduced our proposed scheme of location privacy for both sources and sinks. Our location unobservability scheme provides a third level of security protection, privacy preservation, and it aims to hide the location information of these important nodes from being leaked to attackers. This is done based on a cross-design optimization between the source coding technique in the application layer and the packet transmission operation in order to produce a uniform traffic pattern throughout the network. This generated traffic pattern has a uniform shape and it is independent from event detection process in the network, which makes it very difficult for the attackers to figure out the location information of intended nodes. By using this way in providing the location privacy, our privacy scheme can offer a very good privacy level against both local and global attacks (measured by the safety period) while in the same time maintain the network performance efficiency without the need of generating wide-network dummy packets. Simulation evaluation results prove that our proposed location privacy scheme has a strong privacy level compared to existing techniques and has a very little effect on network performance in terms of end-to-end delay and packet delivery ratio.

Chapter 7

Conclusions and Future Work

This chapter presents detailed conclusions of the research pursued during this thesis work. Moreover, it also through some light on future research and potential future targets.

7.1 Conclusions

|*| In this thesis, we discussed and surveyed in detail the research carried on Wireless Multimedia Sensor Networks (WMSNs). We analyzed the major technical challenges and research issues in designing algorithms, protocols, architectures, and hardware for WMSN. We discussed most of the existing solutions for WMSN at the different layers of the communication stack: physical, MAC, routing, transport, and application along with the possible cross layer implementation. Furthermore, we discussed other complementary research issues in WMSN such as coverage and security issues. Finally, we surveyed and classified the existing off-the-shelf devices, prototypes, and testbeds implemented for WMSNs.

|*| Following the required background, we proposed a Cluster-based Multipath Routing protocol (CMRP) for WMSNs designed to handle the additional requirements of reliable data delivering of different traffic classes and provide load balancing by using multipath routing. The proposed routing protocol, CMRP, is based on the hierarchical structure of multiple paths established depending on the hop count and received signal strength as an indication on the link quality, delay, and distance between the nodes.

CMRP maintains minimum end-to end delay suitable for real-time and non-real-time data packets to meet their playout deadline, and achieves high throughput and packet delivery ratio by selecting the paths with better link quality and avoiding collisions and interferences. CMRP reduces energy consumption at sensor nodes by moving the multimedia processing complexity as well as the aggregation process to the cluster heads side along with preventing path loops and path cycles in establishing the routes. Performance evaluation results show that CMRP clearly outperforms the preexisting ones (DHCT, MCRA, EDGE) in all average end-to-end delay, throughput, packet delivery ratio and battery power consumption.

|*| Then in chapter 3, we presented a cross-layer communication architecture for WMSNs between the routing and MAC layers, where CMRP routing protocol has been pursued in conjunction with an adaptive QoS-aware scheduling to maximize the overall network performance with minimum energy consumption, reliable delivery, and efficient resource management. Our design aims to exploit correlation characteristics and functionalities between the two layers to maximize the overall network performance with minimum energy consumption in order to handle the additional requirements of delivering reliable multimedia data. Our proposed routing protocol provides load balancing by establishing multiple paths based on the hop count and received signal strength as an indication of the link quality, delay, and distance between the nodes. Our proposed scheduling protocol is based on TDMA approach with flexible time-slot assignment that adaptively assigns slots to various traffics from active nodes. The simulation results demonstrate that our cross-layer design can improve the performance of CMRP-routing-only and achieve better than other protocols in terms of average end-to-end delay, throughput, packet delivery ratio and battery power consumption.

|*| In chapter 4, we presented a light weight distributed security scheme of key management and intrusion detection system suitable for securing the communication over clustered WMSNs with minimal impact on overall network performance through balancing its security features against the communication and computational overhead required to implement it. Our proposed security protocol is based on symmetric key ciphers used to authenticate and encrypt the transmitted data and it only requires the sensor nodes to share keys with their cluster heads or one-hop parents. It protects against

the majority of outsider attacks, and it resists against insider attacks since the stolen keys are unique and affect only the local area. The key management scheme is energy efficient with no extra communication overhead, scalable for large scale network, and designed to facilitate the data aggregation at cluster heads and message broadcasting within the clusters using unique-cluster security keys. The proposed light-weight distributed IDS is simple, with very little communication overhead, and efficient to identify malicious internal attackers in clustered WMSNs.Performance evaluation results show that our proposed security scheme is appropriate for securing multimedia delivery while being resilient against general security threats, it has an insignificant effect on network performance metrics (such as average end-to-end delay, throughput, and energy consumption), and it is scalable while having minimum memory requirements.

|*| Then in chapter 5, we analyzed in details all the aspects of having privacy preservation of the contextual information in wireless sensor networks along with the research challenges carried in this filed. We have surveyed the state of the art of wide range of solutions proposed to countermeasure different kind of privacy attacks. Based on this study, we have introduced a complete taxonomy of security protection and event unobservability techniques in WSNs. We explained most of the proposals used in each technique showing their advantages and drawbacks in terms of their network performance efficiency and protection capability against different level of attacks. More specifically, we first surveyed the location privacy problem for the source, sink, and query location and reviewed most of the proposed privacy-preserving techniques. Then, we analyzed the protection of node identity privacy and explained its state of the art approaches. We also examined the temporal privacy issues and addressed the existing schemes related to this subject.

|*| After drawing a complete picture for event unobservability in WSNs, we proposed in chapter 6 our source/sink location unobservability scheme for WMSNs. Our privacy scheme aims to protect the location information of the important nodes in the network such as the source and sink nodes against global attacks in an energy efficient manner while in the same time maintain the performance efficiency of the sensor network. The proposed event unobservability scheme is based on a joint design optimization between the application layer functionality and packet transmission operation in order to avoid

using network-wide dummy packet injection and to increase the quality of received multimedia content.

7.2 Discussion and Future Work

7.2.1 Challenges in WMSNs

|*| **In physical layer:** in order to further increase capacity and mitigate the impairment by fading and co-channel interference, multi-antenna systems such as antenna diversity, smart antenna, and MIMO systems, can be combined with UWB for short-range networks with multi-gigabit rates. However, these physical-layer techniques have many challenging problems to be developed for WMSNs. Although UWB appears to be a promising alternative physical layer technology and it has many attractive features, it is still not very mature and there are many challenges and issues that need to be resolved and better understood.

|*| **In MAC layer:** we think that cross-layering is essential for efficient MAC designs in WMSNs, together with queue-management and traffic classification/prioritization as long as QoS is required for multimedia traffic. We believe that in future work we need to consider multi-channel MAC protocols that could be more suitable for WMSNs in the sense of increasing the capacity and reducing the interferences.

|*| **In transport layer:** Most of the proposed application-specific transport protocols do not take into consideration the multimedia requirements in WMSN and none of them addresses its diverse concerns. This can be seen clearly in the performance evaluation conducted in [60], where it was shown that many of the proposed transport protocols cannot provide acceptable video transmission and do not support real-time communication in WMSN. Therefore, we believe that designing a transport protocol with appropriate performance metrics for both reliability and congestion control and based on the application layer source coding techniques will be a promising direction in this research area.

|*| **In application layer:** having multimedia processing schemes (such as source coding techniques) in the application layer of WMSNs is essential in reducing the amount of multimedia traffic transferred over the network by extracting the useful information from the captured images and videos while in the same time maintaining the application-specific QoS requirements. However, in WMSNs, these techniques should be designed in such a way that they meet current hardware capabilities, more power efficient to match the battery constrains in WMSN, and have high compression efficiency to reduce the size of the multimedia content and to meet the available supported data rate and bandwidth in the network. we foresee that exploiting the behavior of these multimedia processing techniques for designing cross-layer optimization with the other layers, especially routing and MAC layers, will be a promising direction for future work.

7.2.2 Routing in WMSNs

Proposed routing protocols for WMSNs needs to be more efficient to handle the multimedia data and transfer it to the intended destinations in a way that sustains the energy level of the network as long as possible while maintaining the quality of the received content at the same time. We notice that most of the existing proposed protocols for WMSNs follow the classical layered structure of the communication protocol stack without taking into consideration the especial requirements of handling real-time multimedia content over WMSNs. We believe that the correlation characteristics and interdependencies among the layers of the communication stack in WMSNs cannot be neglected and should be exploited for better performance and efficient communication. So, cross-layer optimization can be the solution to meet the especial requirements of WMSN and its design challenges in order to provide enough support for multimedia applications and maximize network performance. Also, we believe that the future direction in routing protocol for WMSNs should adopt hierarchal (cluster-based) scheme with multipath routing because these wireless networks need to exploit the network bandwidth to its limit and sometimes in short bursts. the future proposals will have to be designed as integral solutions that cover routing, MAC layer, and sometimes even transport layer in order to increase the efficiency of supporting the Quality of Service

requirements for transporting the multimedia content.
Regarding improving the design of CMRP, we will focus in future work on optimizing the threshold values based on network configuration both in mathematical representation and in simulation. This will result in better cluster distribution and network connectivity. Also, studying in more details the effect of the communication overhead and synchronization problem in our scheduling protocol is planned.

7.2.3 Security in WMSNs

Proposed security mechanisms for WMSNs should have minimal impact on overall performance through balancing their security features against the communication and computational overhead required to implement them. We foresee that for WMSNs symmetric cryptography will be the chosen approach -over asymmetric cryptography- since its lighter processing requirements since it makes a lot easier to solve several security problems related to eavesdropping and compromised nodes. Moreover, in order to preserve battery and to save bandwidth, many WMSNs will use some sort of data aggregation. We consider that security and aggregation schemes cannot be devised separately. Therefore, new security schemes will have to be both energy-aware and designed in together with the aggregation scheme.

7.2.4 Privacy in WMSNs

From the review of the existing works in event unobservability in sensor networks, one can see that the benefit of providing privacy protection usually comes at the cost of other network performances such as time delay, successful delivery, and power consumption. Hence, there is a trade off between the level of provided privacy and other performance metrics, mainly the end-to-end delay and energy consumption. We think that successful practical privacy solutions should be designed to achieve the objective of event unobservability while in the same time maintain a minimum extra communication overhead in order to guarantee the delivery quality (*i.e.* delivery latency and delivery ratio) of transmitted data and prolong network lifetime. In our proposed location privacy scheme, we exploit the joint design between the source coding techniques in the application layer and the multipath packet transmission in order to avoid using network-wide fake packet injection while maintain a strong level of offered privacy against global attacks. In future work, the proposed event unobservability scheme can be analyzed more, for example, how to optimize its performance in terms of buffer time and delivery delay. Also more other attacks models can be considered such as insider and/or active attacks.

Appendices

Appendix A

Publications

A.1 Accepted Journals

1. Wireless Multimedia Sensor Networks: Current Trends and Future Directions; Islam T. Almalkawi, Manel Guerrero-Zapata, Jamal N. Al-Karaki, and Julian Morillo-Pozo;

   In Sensors Journal 2010, 10(7), 6662-6717. [JCR-2010: 1.774 Q1] (14/61 Q1 INSTRUMENTS AND INSTRUMENTATION).

2. A Secure Cluster-Based Multipath Routing Protocol for WMSNs; Islam T. Almalkawi, Manel Guerrero-Zapata, and Jamal N. Al-Karaki;

   In Sensors Journal 2011, 11(4), 4401-4424. [JCR-2011: 1.739 Q1] (14/59 Q1 INSTRUMENTS AND INSTRUMENTATION).

3. A Cross-layer based Clustered Multipath Routing with QoS-aware Scheduling for Wireless Multimedia Sensor Networks; Islam T. Almalkawi, Manel Guerrero-Zapata, and Jamal N. Al-Karaki;

   In International Journal of Distributed Sensor Networks Volume 2012 (2012), Article ID 392515, 11 pages. [JCR-2011: 0.203 Q4] (71/79 Q4 TELECOMMUNICATIONS).

A.2 Other Publications and Submitted Papers

1. Energy Efficiency in Wireless Multimedia Sensor Networks; Islam T. Almalkawi, Mohammad Alaei, Manel Guerrero-Zapata, Jose M. Barcelo-Ordinas, and Julian Morillo-Pozo;

   In IEEE COMSOC MMTC E-Letter. PP.17-20. Vol. 6, No. 12, 2011.

2. Light-weight Security Scheme for Key Management and Intrusion Detection in Clustered Wireless Multimedia Sensor Networks; Islam T. Almalkawi, Manel Guerrero-Zapata, and Jamal N. Al-Karaki;

   Submitted to the Journal of Networks and Computer Applications (JNCA), March 2013.

3. Event Unobservability in Wireless Sensor Networks: A Survey; Islam T. Almalkawi, Manel Guerrero-Zapata, and Jamal N. Al-Karaki;

   Submitted to the Journal of Networks and Computer Applications (JNCA), February 2013.

4. An Efficient Source/Sink Location Unobservability for Wireless Multimedia Sensor Networks; Islam T. Almalkawi, Manel Guerrero-Zapata, and Jamal N. Al-Karaki;

   Will be submitted to the Journal of ..., 2013.

References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, “Wireless sensor networks: a survey,” *Computer Networks*, vol. 38, no. 4, pp. 393–422, 2002.

[2] R. Min, M. Bhardwaj, S.-H. Cho, E. shih, A. Sinha, A. Wang, A. Chandrakasan, and E. S. A. Sinha, “Low-power wireless sensor networks,” in *In VLSI Design*, 2001, pp. 205–210.

[3] J. N. Al-karaki and A. E. Kamal, “Routing techniques in wireless sensor networks: A survey,” *IEEE Wireless Communications*, vol. 11, pp. 6–28, 2004.

[4] K. Akkaya and M. Younis, “A survey on routing protocols for wireless sensor networks,” *Ad Hoc Networks*, vol. 3, pp. 325–349, 2005.

[5] A. Salhieh and L. Schwiebert, “Power aware metrics for wireless sensor networks,” *In in the 14th IASTED Conference on Parallel and Distributed Computing and Systems (PDCS 2002) Symposium*, vol. 26, pp. 326–331, 2002.

[6] I. F. Akyildiz, T. Melodia, and K. R. Chowdhury, “A survey on wireless multimedia sensor networks,” *Comput. Netw.*, vol. 51, no. 4, pp. 921–960, 2007.

[7] R. Cucchiara, “Multimedia surveillance systems,” in *VSSN ’05: Proceedings of the third ACM international workshop on Video surveillance & sensor networks*. New York, NY, USA: ACM, 2005, pp. 3–10.

[8] K. Wong, “Physical layer considerations for wireless sensor networks,” in *Networking, Sensing and Control, 2004 IEEE International Conference on*, vol. 2, 2004, pp. 1201–1206.

[9] “IEEE 802.15 WPAN task group 4 (tg4),” *http://grouper.ieee.org/groups/802/15/pub/TG4.html*.

[10] E. Capo-Chichi and J.-M. Friedt, “Design of embedded sensor platform for multimedia application,” in *Distributed Framework and Applications, 2008. DFmA 2008. First International Conference on*, Oct. 2008, pp. 146–150.

[11] A. Kerhet, M. Mango, F. Leonardi, A. Boni, and L. Benini, “A low-power wireless video sensor node for distributed object detection,” in *Journal of Real-Time Image Processing*, vol. 2, 0-0 2007, pp. 331–342.

[12] E. Karapistoli, I. Gragopoulos, I. Tsetsinas, and F.-N. Pavlidou, “UWB technology to enhance the performance of wireless multimedia sensor networks,” in *Computers and Communications, 2007. ISCC 2007. 12th IEEE Symposium on*, July 2007, pp. 57–62.

[13] I. Akyildiz, T. Melodia, and K. Chowdury, “Wireless multimedia sensor networks: A survey,” *Wireless Communications, IEEE*, vol. 14, no. 6, pp. 32–39, December 2007.

[14] K. Kredo, II and P. Mohapatra, “Medium access control in wireless sensor networks,” *Comput. Netw.*, vol. 51, no. 4, pp. 961–994, 2007.

[15] C. Li, P. Wang, H.-H. Chen, and M. Guizani, “A cluster based on-demand multi-channel MAC protocol for wireless multimedia sensor networks,” in *Communications, 2008. ICC ’08. IEEE International Conference on*, May 2008, pp. 2371–2376.

[16] W. Ye, J. Heidemann, and D. Estrin, “Medium access control with coordinated, adaptive sleeping for wireless sensor networks,” *IEEE/ACM Transaction on Networking*, vol. 12, no. 3, pp. 493–506, 2004.

[17] T. V. Dam and K. Langendoen, “An adaptive energy-efficient MAC protocol for wireless sensor networks,” in *Proc. of the ACM Conf. on Embedded Networked Sensor Systems (SenSys)*, 2003.

[18] N. Saxena, A. Roy, and J. Shin, “A QoS-based energy-aware MAC protocol for wireless multimedia sensor networks,” in *Vehicular Technology Conference, 2008. VTC Spring 2008. IEEE*, May 2008, pp. 183–187.

[19] R. N. Sexena, A. Roy, and J. Shin, “Dynamic duty cycle and adaptive contention window based QoS-MAC protocol for wireless multimedia sensor networks,” *Comput. Netw.*, vol. 52, no. 13, pp. 2532–2542, 2008.

[20] T. Melodia and I. Akyildiz, “Cross-layer quality of service support for UWB wireless multimedia sensor networks,” in *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*, April 2008, pp. 2038–2046.

[21] J. Polastre, J. Hill, and D. Culler, “Versatile low power media access for wireless sensor networks,” in *SenSys ’04: Proceedings of the 2nd international conference on Embedded networked sensor systems*. New York, NY, USA: ACM, 2004, pp. 95–107.

[22] J. So and N. H. Vaidya, “Multi-channel MAC for ad hoc networks: handling multi-channel hidden terminals using a single transceiver,” in *MobiHoc ’04: Proceedings of the 5th ACM international symposium on Mobile ad hoc networking and computing*. New York, NY, USA: ACM, 2004, pp. 222–233.

[23] G. Zhou, C. Huang, T. Yan, T. He, J. A. Stankovic, and T. F. Abdelzaher, “MMSN: Multi-frequency media access control for wireless sensor networks,” in *INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings*, April 2006, pp. 1–13.

[24] M. Yaghmaee and D. Adjeroh, “A model for differentiated service support in wireless multimedia sensor networks,” in *Computer Communications and Networks, 2008. ICCN ’08. Proceedings of 17th International Conference on*, Aug. 2008, pp. 1–6.

[25] H. Aghdasi, M. Abbaspour, and M. Moghadam, “An energy-efficient and high-quality mac protocol for image transmission in wireless sensor networks,” in *Circuits and Systems for Communications, 2008. ICCSC 2008. 4th IEEE International Conference on*, May 2008, pp. 838–842.

[26] R. J. H. V. S. T. C. Phan, K.T.; Fan, “Network lifetime maximization with node admission in wireless multimedia sensor networks,” in *IEEE Transactions on Vehicular Technology: Accepted for future publication, 2009. IEEE*, May 2009.

[27] P. Sarisaray, G. Gur, S. Baydere, and E. Harmanc, “Performance comparison of error compensation techniques with multipath transmission in wireless multimedia sensor networks,” in *Modeling, Analysis, and Simulation of Computer and Telecommunication Systems, 2007. MASCOTS ’07. 15th International Symposium on*, Oct. 2007, pp. 73–86.

[28] Y. Sun, H. Ma, L. Liu, and Y. Zheng, “ASAR: An ant-based service-aware routing algorithm for multimedia sensor networks,” in *Frontiers of Electrical and Electronic Engineering in China*, vol. 3, 2008, pp. 25–33.

[29] L. Shu, Y. Zhang, L. Yang, Y. Wang, and M. Hauswirth, “Geographic routing in wireless multimedia sensor networks,” in *Future Generation Communication and Networking, 2008. FGCN ’08. Second International Conference on*, vol. 1, Dec. 2008, pp. 68–73.

[30] B. Karp and H. T. Kung, “GPSR: greedy perimeter stateless routing for wireless networks,” in *MobiCom ’00: Proceedings of the 6th annual international conference on Mobile computing and networking*. New York, NY, USA: ACM, 2000, pp. 243–254.

[31] M. Gerla and K. Xu, “Multimedia streaming in large-scale sensor networks with mobile swarms,” *SIGMOD Rec.*, vol. 32, no. 4, pp. 72–76, 2003.

[32] M. Maimour, “Maximally radio-disjoint multipath routing for wireless multimedia sensor networks,” in *WMuNep ’08: Proceedings of the 4th ACM workshop on Wireless multimedia networking and performance modeling*. New York, NY, USA: ACM, 2008, pp. 26–31.

[33] S. Li, R. Neelisetti, C. Liu, and A. Lim, “Delay-constrained high throughput protocol for multi-path transmission over wireless multimedia sensor networks,” in *World of Wireless, Mobile and Multimedia Networks, 2008. WoWMoM 2008. 2008 International Symposium on a*, June 2008, pp. 1–8.

[34] M. Hamid, M. Alam, and C. S. Hong, “Design of a QoS-aware routing mechanism for wireless multimedia sensor networks,” in *Global Telecommunications Conference, 2008. IEEE GLOBECOM 2008. IEEE*, 30 2008-Dec. 4 2008, pp. 1–6.

[35] K. Akkaya and M. Younis, “Energy and QoS aware routing in wireless sensor networks,” *Cluster Computing*, vol. 8, no. 2-3, pp. 179–188, 2005.

[36] M. Rahman, R. GhasemAghaei, A. El Saddik, and W. Gueaieb, “M-IAR: Biologically inspired routing protocol for wireless multimedia sensor networks,” in *Instrumentation and Measurement Technology Conference Proceedings, 2008. IMTC 2008. IEEE*, May 2008, pp. 1823–1827.

[37] K. Zongwu, L. Layuan, S. Qiang, and C. Nianshen, “Ant-like game routing algorithm for wireless multimedia sensor networks,” in *Wireless Communications, Networking and Mobile Computing, 2008. WiCOM ’08. 4th International Conference on*, Oct. 2008, pp. 1–4.

[38] S. Darabi, N. Yazdani, and O. Fatemi, “Multimedia-aware MMSPEED: A routing solution for video transmission in WMSN,” dec. 2008, pp. 1–3.

[39] E. Felemban, C.-G. Lee, and E. Ekici, “MMSPEED: multipath multi-SPEED protocol for QoS guarantee of reliability and timeliness in wireless sensor networks,” *Mobile Computing, IEEE Transactions on*, vol. 5, no. 6, pp. 738–754, june 2006.

[40] T. He, J. Stankovic, T. Abdelzaher, and C. Lu, “A spatiotemporal communication protocol for wireless sensor networks,” *Parallel and Distributed Systems, IEEE Transactions on*, vol. 16, no. 10, pp. 995–1006, oct. 2005.

[41] N. Saxena, A. Roy, and J. Shin, “QuESt: a QoS-based energy efficient sensor routing protocol,” *Wireless Communications and Mobile Computing*, vol. 9, no. 3, pp. 417–426, 2009.

[42] C. Ma and Y. Yang, “Battery aware routing for streaming data transmissions in wireless sensor networks,” in *Mobile Networks and Applications Vol 11*. Springer, 2006, pp. 757–767.

[43] L. Shu, Z. Zhou, M. Hauswirth, D. L. Phuoc, Y. Peng, and L. Zhang, “Transmitting streaming data in wireless multimedia sensor networks with holes,” in *MUM’07: Proceedings of the Sixth International Conference on Mobile and Ubiquitous Multimedia*. ACM, 2007.

[44] L. Campelli, I. Akyildiz, L. Fratta, and M. Cesana, “A cross-layer solution for ultrawideband based wireless video sensor networks,” in *IEEE Globecom 2008*. IEEE, 30 Nov.- 4 Dec, 2008.

[45] C. Wang, K. Sohraby, B. Li, M. Daneshmand, and Y. Hu, “A survey of transport protocols for wireless sensor networks,” *Network, IEEE*, vol. 20, no. 3, pp. 34–40, May-June 2006.

[46] M. Yaghmaee and D. Adjeroh, “A new priority based congestion control protocol for wireless multimedia sensor networks,” in *World of Wireless, Mobile and Multimedia Networks, 2008. WoWMoM 2008. 2008 International Symposium on a*, June 2008, pp. 1–8.

[47] M. Maimour, C. Pham, and J. Amelot, “Load repartition for congestion control in multimedia wireless sensor networks with multipath routing,” in *Wireless Pervasive Computing, 2008. ISWPC 2008. 3rd International Symposium on*, May 2008, pp. 11–15.

[48] C.-Y. Wan, S. B. Eisenman, and A. T. Campbell, “CODA: congestion detection and avoidance in sensor networks,” in *SenSys ’03: Proceedings of the 1st international conference on Embedded networked sensor systems*. New York, NY, USA: ACM, 2003, pp. 266–279.

[49] C. T. Ee and R. Bajcsy, "Congestion control and fairness for many-to-one routing in sensor networks," in *SenSys '04: Proceedings of the 2nd international conference on Embedded networked sensor systems*. New York, NY, USA: ACM, 2004, pp. 148–161.

[50] C. Wang, K. Sohraby, V. Lawrence, B. Li, and Y. Hu, "Priority-based congestion control in wireless sensor networks," in *Sensor Networks, Ubiquitous, and Trustworthy Computing, 2006. IEEE International Conference on*, vol. 1, June 2006, pp. 1–8.

[51] B. Hull, K. Jamieson, and H. Balakrishnan, "Mitigating congestion in wireless sensor networks," in *SenSys '04: Proceedings of the 2nd international conference on Embedded networked sensor systems*. New York, NY, USA: ACM, 2004, pp. 134–147.

[52] Y. Iyer, S. Gandham, and S. Venkatesan, "STCP: a generic transport layer protocol for wireless sensor networks," in *Computer Communications and Networks, 2005. ICCCN 2005. Proceedings. 14th International Conference on*, Oct. 2005, pp. 449–454.

[53] F. Stann and J. Heidemann, "RMST: reliable data transport in sensor networks," in *Sensor Network Protocols and Applications, 2003. Proceedings of the First IEEE. 2003 IEEE International Workshop on*, May 2003, pp. 102–112.

[54] A. Boukerche, J. Feng, R. Werner, Y. Du, and Y. Huang, "Reconstructing the plenoptic function from wireless multimedia sensor networks," in *Local Computer Networks, 2008. LCN 2008. 33rd IEEE Conference on*, Oct. 2008, pp. 74–81.

[55] Y. Yan, G. Chen, and S. Das, "A collaboration-based storage management scheme in multimedia sensor networks," in *Embedded and Ubiquitous Computing, 2008. EUC '08. IEEE/IFIP International Conference on*, vol. 1, Dec. 2008, pp. 288–294.

[56] N. Dimokas, D. Katsaros, and Y. Manolopoulos, "Cooperative caching in wireless multimedia sensor networks," *Mob. Netw. Appl.*, vol. 13, no. 3-4, pp. 337–356, 2008.

[57] V. Lecuire, C. Duran Faundez, and N. Krommenacker, "Energy-efficient image transmission in sensor networks," *Int. J. Sen. Netw.*, vol. 4, no. 1/2, pp. 37–47, 2008.

[58] S. Qaisar and H. Radha, "Multipath multi-stream distributed reliable video delivery in wireless sensor networks," in *Information Sciences and Systems, 2009. CISS 2009. 43rd Annual Conference on*, March 2009, pp. 207–212.

[59] A. Boukerche, Y. Du, J. Feng, and R. Pazzi, “A reliable synchronous transport protocol for wireless image sensor networks,” in *Computers and Communications, 2008. ISCC 2008. IEEE Symposium on*, July 2008, pp. 1083–1089.

[60] O. Akan, “Performance of transport protocols for multimedia communications in wireless sensor networks,” *Communications Letters, IEEE*, vol. 11, no. 10, pp. 826–828, october 2007.

[61] H. Radha, M. van der Schaar, and Y. Chen, “The MPEG-4 fine-grained scalable video coding method for multimedia streaming over IP,” *Multimedia, IEEE Transactions on*, vol. 3, no. 1, pp. 53–68, Mar 2001.

[62] W. Wang, D. Peng, H. Wang, H. Sharif, and H.-H. Chen, “Energy-constrained distortion reduction optimization for wavelet-based coded image transmission in wireless sensor networks,” *Multimedia, IEEE Transactions on*, vol. 10, no. 6, pp. 1169–1180, Oct. 2008.

[63] R. Puri, A. Majumdar, P. Ishwar, and K. Ramchandran, “Distributed video coding in wireless sensor networks,” *Signal Processing Magazine, IEEE*, vol. 23, no. 4, pp. 94–106, July 2006.

[64] J. Ahmad, H. Khan, and S. Khayam, “Energy efficient video compression for wireless sensor networks,” in *Information Sciences and Systems, 2009. CISS 2009. 43rd Annual Conference on*, March 2009, pp. 629–634.

[65] Y. Wang, A. Reibman, and S. Lin, “Multiple description coding for video delivery,” *Proceedings of the IEEE*, vol. 93, no. 1, pp. 57–70, Jan. 2005.

[66] B. Girod, A. Aaron, S. Rane, and D. Rebollo-Monedero, “Distributed video coding,” *Proceedings of the IEEE*, vol. 93, no. 1, pp. 71–83, Jan. 2005.

[67] A. Aaron, S. Rane, E. Setton, and B. Girod, “Transform-domain Wyner-Ziv codec for video,” *Proc. SPIE Visual Communications and Image Processing*, Jan. 2004.

[68] R. Puri and K. Ramchandran, “PRISM: A new robust video coding architecture based on distributed compression principles,” *Allerton Conference on Communication, Control, and Computing*, Oct. 2002.

[69] C. Yaacoub, J. Farah, and B. Pesquet-Popescu, “Joint source-channel Wyner-Ziv coding in wireless video sensor networks,” in *Signal Processing and Information Technology, 2007 IEEE International Symposium on*, Dec. 2007, pp. 225–228.

[70] R. Halloush, K. Misra, and H. Radha, “Practical distributed video coding over visual sensors,” in *PCS’09: Proceedings of the 27th conference on Picture Coding Symposium*. Piscataway, NJ, USA: IEEE Press, 2009, pp. 121–124.

[71] E. Culurciello, J. H. Park, and A. Savvides, “Address-event video streaming over wireless sensor networks,” in *Circuits and Systems, 2007. ISCAS 2007. IEEE International Symposium on*, May 2007, pp. 849–852.

[72] L. W. Chew, L.-M. Ang, and K. P. Seng, “Survey of image compression algorithms in wireless sensor networks,” in *Information Technology, 2008. ITSim 2008. International Symposium on*, vol. 4, Aug. 2008, pp. 1–9.

[73] S. Nath, Y. Ke, P. B. Gibbons, B. Karp, and S. Seshan, “A distributed filtering architecture for multimedia sensors,” In First Workshop on Broadband Advanced Sensor Networks (BaseNets, Tech. Rep., 2004.

[74] S. Wang, X. Wang, L. Ding, D. Bi, and Z. You, “Collaborative hybrid classifier learning with ant colony optimization in wireless multimedia sensor networks,” in *Intelligent Control and Automation, 2008. WCICA 2008. 7th World Congress on*, June 2008, pp. 3341–3346.

[75] H. Wang, D. Peng, W. Wang, H. Sharif, J. Wegiel, D. Nguyen, R. Bowne, and C. Backhaus, “Artificial immune system based image pattern recognition in energy efficient wireless multimedia sensor networks,” in *Military Communications Conference, 2008. MILCOM 2008. IEEE*, Nov. 2008, pp. 1–7.

[76] X. Wang, S. Wang, and D. Bi, “Compacted probabilistic visual target classification with committee decision in wireless multimedia sensor networks,” *Sensors Journal, IEEE*, vol. 9, no. 4, pp. 346–353, April 2009.

[77] M. Chen, V. Leung, S. Mao, and M. Li, “Cross-layer and path priority scheduling based real-time video communications over wireless sensor networks,” in *Vehicular Technology Conference, 2008. VTC Spring 2008. IEEE*, 11-14 2008, pp. 2873–2877.

[78] L. Shu, Y. Zhang, Z. Yu, L. T. Yang, M. Hauswirth, and N. Xiong, “Context-aware cross-layer optimized video streaming in wireless multimedia sensor networks,” *In Springer The Journal of Supercomputing (JoS)*, 2009.

[79] L. Shu, M. Hauswirth, Y. Zhang, J. Ma, and G. Min, “Cross layer optimization on data gathering in wireless multimedia sensor networks within expected network lifetime,” *Accepted in Springer Journal of Universal Computer Science (JUCS)*, 2009.

[80] R. N. Sexena, A. Roy, and J. Shin, "Cross-layer algorithms for qos enhancement in wireless multimedia sensor networks," *IEICE Trans. Commun.*, vol. E91-B, no. 8, Aug. 2008.

[81] N. Tezcan and W. Wang, "Self-orienting wireless multimedia sensor networks for maximizing multimedia coverage," in *Communications, 2008. ICC '08. IEEE International Conference on*, May 2008, pp. 2206–2210.

[82] X. Liu, P. Kulkarni, P. Shenoy, and D. Ganesan, "Snapshot: A self-calibration protocol for camera sensor networks," in *Broadband Communications, Networks and Systems, 2006. BROADNETS 2006. 3rd International Conference on*, Oct. 2006, pp. 1–10.

[83] M. Alaei and J. M. Barcelo-Ordinas, "A cluster-based scheduling for object detection in wireless multimedia sensor networks," in *Q2SWinet '09: Proceedings of the 5th ACM symposium on QoS and security for wireless and mobile networks.* New York, NY, USA: ACM, 2009, pp. 50–56.

[84] H. Wang, D. Peng, W. Wang, H. Sharif, and H.-H. Chen, "Energy-aware adaptive watermarking for real-time image delivery in wireless sensor networks," May 2008, pp. 1479–1483.

[85] D. Kundur, U. N. Okorafor, and W. Luh, "HoLiSTiC: Heterogeneous lightweight sensornets for trusted visual computing," in *Intelligent Information Hiding and Multimedia Signal Processing, 2006. IIH-MSP '06. International Conference on*, Dec. 2006, pp. 267–270.

[86] D. Kundur, T. Zourntos, and N. Mathai, "Lightweight security principles for distributed multimedia based sensor networks," in *Signals, Systems and Computers, 2004. Conference Record of the Thirty-Eighth Asilomar Conference on*, vol. 1, Nov. 2004, pp. 368–372.

[87] I. Akyildiz, T. Melodia, and K. Chowdhury, "Wireless multimedia sensor networks: Applications and testbeds," *Proceedings of the IEEE*, vol. 96, no. 10, pp. 1588–1605, Oct. 2008.

[88] W. S. Platform, "MICA-family wireless mote platform specifications," *http://www.xbow.com/Products/productdetails.aspx?sid=156*.

[89] R. Mangharam, A. Rowe, and R. Rajkumar, "FireFly: a cross-layer platform for real-time embedded wireless networks," *Real-Time Syst.*, vol. 37, no. 3, pp. 183–231, 2007.

[90] W. S. Platform, “TmoteSky platform specifications,” *http://www.sentilla.com/moteiv-transition.html*.

[91] P. Chen, P. Ahammad, C. Boyer, S.-I. Huang, L. Lin, E. Lobaton, M. Meingast, S. Oh, S. Wang, P. Yan, A. Yang, C. Yeo, L.-C. Chang, J. Tygar, and S. Sastry, “CITRIC: A low-bandwidth wireless camera network platform,” in *Distributed Smart Cameras, 2008. ICDSC 2008. Second ACM/IEEE International Conference on*, Sept. 2008, pp. 1–10.

[92] C. M. University, “CMUcam3 integration with Tmote Sky sensor node,” *http://www.cmucam.org*, Sep. 2007.

[93] W. S. Platform, “Crossbow stargate platform,” *http://www.xbow.com/Products/productdetails.aspx?sid=85*.

[94] P. Kulkarni, D. Ganesan, P. Shenoy, and Q. Lu, “Senseye: a multi-tier camera sensor network,” in *MULTIMEDIA ’05: Proceedings of the 13th annual ACM international conference on Multimedia*. New York, NY, USA: ACM, 2005, pp. 229–238.

[95] J. Boice, X. Lu, C. Margi, G. Stanek, G. Zhang, and K. Obraczka, “Meerkats: A power-aware, self-managing wireless camera network for wide area monitoring,” in *in Distributed Smart Cameras Workshop - SenSys06*, 2006.

[96] W.-C. Feng, E. Kaiser, W. C. Feng, and M. L. Baillif, “Panoptes: scalable low-power video sensor networking technologies,” *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 1, no. 2, pp. 151–167, 2005.

[97] C. wireless sensor platform, “Intel mote2 platform,” *http://www.xbow.com/products*.

[98] T. Teixeira, E. Culurciello, J. Park, D. Lymberopoulos, A. Barton-Sweeney, and A. Savvides, “Address-event imagers for sensor networks: evaluation and modeling,” in *Information Processing in Sensor Networks, 2006. IPSN 2006. The Fifth International Conference on*, 0-0 2006, pp. 458–466.

[99] D. Xie, T. Yan, D. Ganesan, and A. Hanson, “Design and implementation of a dual-camera wireless sensor network for object retrieval,” in *IPSN ’08: Proceedings of the 7th international conference on Information processing in sensor networks*. Washington, DC, USA: IEEE Computer Society, 2008, pp. 469–480.

[100] L. B. R. I. Downes and H. Aghajan, “Development of a mote for wireless image sensor network,” *In Proc. of Cognitive Systems and Interactive Sensors, COGIS*, March 2006.

[101] C. Park and P. H. Chou, "eCAM: ultra compact, high data-rate wireless sensor node with a miniature camera," in *SenSys '06: Proceedings of the 4th international conference on Embedded networked sensor systems*. New York, NY, USA: ACM, 2006, pp. 359–360.

[102] M. Rahimi, R. Baer, O. I. Iroezi, J. C. Garcia, J. Warrior, D. Estrin, and M. Srivastava, "Cyclops: in situ image sensing and interpretation in wireless sensor networks," in *SenSys '05: Proceedings of the 3rd international conference on Embedded networked sensor systems*. ACM, 2005, pp. 192–204.

[103] A. Rowe, D. Goel, and R. Rajkumar, "FireFly Mosaic: A vision-enabled wireless sensor networking system," in *Real-Time Systems Symposium, 2007. RTSS 2007. 28th IEEE International*, Dec. 2007, pp. 459–468.

[104] R. Kleihorst, A. Abbo, B. Schueler, and A. Danilin, "Camera mote with a high-performance parallel processor for real-time frame-based video processing," in *Advanced Video and Signal Based Surveillance, 2007. AVSS 2007. IEEE Conference on*, Sept. 2007, pp. 69–74.

[105] D. Lymberopoulos and A. Savvides, "XYZ: a motion-enabled, power aware sensor node platform for distributed sensor network applications," in *Information Processing in Sensor Networks, 2005. IPSN 2005. Fourth International Symposium on*, April 2005, pp. 449–454.

[106] S. Hengstler, D. Prashanth, S. Fong, and H. Aghajan, "Mesheye: A hybrid-resolution smart camera mote for applications in distributed intelligent surveillance," in *Information Processing in Sensor Networks, 2007. IPSN 2007. 6th International Symposium on*, April 2007, pp. 360–369.

[107] S. Kurkowski, T. Camp, and M. Colagrosso, "Manet simulation studies: The incredibles," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 9, pp. 50–61, 2005.

[108] S. Hengstler and H. Aghajan, "WiSNAP: a wireless image sensor network application platform," in *Testbeds and Research Infrastructures for the Development of Networks and Communities, 2006. TRIDENTCOM 2006. 2nd International Conference on*, 0-0 2006, pp. 6–12.

[109] C. Margi, V. Petkov, K. Obraczka, and R. Manduchi, "Characterizing energy consumption in a visual sensor network testbed," in *Testbeds and Research Infrastructures for the*

Development of Networks and Communities, 2006. TRIDENTCOM 2006. 2nd International Conference on, 0-0 2006, pp. 1–8.

[110] D. Johnson, T. Stack, R. Fish, D. M. Flickinger, L. Stoller, R. Ricci, and J. Lepreau, "Mobile emulab: A robotic wireless and sensor network testbed," in *INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings*, April 2006, pp. 1–12.

[111] T. A. Dahlberg, A. Nasipuri, and C. Taylor, "Explorebots: a mobile network experimentation testbed," in *E-WIND '05: Proceedings of the 2005 ACM SIGCOMM workshop on Experimental approaches to wireless network design and analysis*. New York, NY, USA: ACM, 2005, pp. 76–81.

[112] J. Campbell, P. B. Gibbons, S. Nath, P. Pillai, S. Seshan, and R. Sukthankar, "IrisNet: an internet-scale architecture for multimedia sensors," in *MULTIMEDIA '05: Proceedings of the 13th annual ACM international conference on Multimedia*. New York, NY, USA: ACM, 2005, pp. 81–88.

[113] M. A.-K. J. M.-P. J. Almalkawi, I.T.; Guerrero Zapata, "Wireless multimedia sensor networks: Current trends and future directions," *Sensors 10*, vol. 7, pp. 6662–6717, June 2010.

[114] I. F. Akyildiz, T. Melodia, and K. R. Chowdhury, "A survey on wireless multimedia sensor networks," *Comput. Netw.*, vol. 51, pp. 921–960, March 2007.

[115] K. Kredo, II and P. Mohapatra, "Medium access control in wireless sensor networks," *Comput. Netw.*, vol. 51, no. 4, pp. 961–994, Mar. 2007.

[116] M. A. Yigitel, O. D. Incel, and C. Ersoy, "QoS-aware MAC protocols for wireless sensor networks: A survey," *Computer Networks*, vol. 55, no. 8, pp. 1982–2004, 2011.

[117] N. Saxena, A. Roy, and J. Shin, "Dynamic duty cycle and adaptive contention window based QoS-MAC protocol for wireless multimedia sensor networks," *Comput. Netw.*, vol. 52, no. 13, pp. 2532–2542, Sep. 2008.

[118] D. G. Costa and L. A. Guedes, "A survey on multimedia-based cross-layer optimization in visual sensor networks," *Sensors*, vol. 11, no. 5, pp. 5439–5468, 2011.

[119] X. Yan, L. Li, and F. An, "Multi-constrained routing in wireless multimedia sensor networks," in *Wireless Communications Signal Processing, 2009. WCSP 2009. International Conference on*, 2009, pp. 1–5.

[120] C. Li, J. Zou, H. Xiong, and Y. Zhang, “Joint coding/routing optimization for correlated sources in wireless visual sensor networks,” in *Global Telecommunications Conference, 2009. GLOBECOM 2009. IEEE*, 2009, pp. 1–8.

[121] G. Shah, W. Liang, and X. Shen, “Cross-layer design for QoS support in wireless multimedia sensor networks,” in *Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE*, 2010, pp. 1–5.

[122] W. Xiuchao, “Simulate 802.11b channel within NS2,” *http://cir.nus.edu.sg/reactivetcp/report/*, 2004.

[123] NS2:, “The Network Simulator version 2,” *http://www.isi.edu/nsnam/ns/index.html.*

[124] S. Li, R. Neelisetti, C. Liu, and A. Lim, “Delay-constrained high throughput protocol for multi-path transmission over wireless multimedia sensor networks,” *A World of Wireless, Mobile and Multimedia Networks, International Symposium on*, vol. 0, pp. 1–8, 2008.

[125] S. Li, A. Lim, S. Kulkarni, and C. Liu, “EDGE: A routing algorithm for maximizing throughput and minimizing delay in wireless sensor networks,” in *Military Communications Conference, 2007. MILCOM 2007. IEEE*, 2007, pp. 1–7.

[126] A. Perrig, R. Szewczyk, J. D. Tygar, V. Wen, and D. E. Culler, “SPINS: security protocols for sensor networks,” *Wirel. Netw.*, vol. 8, pp. 521–534, September 2002.

[127] L. Tobarra, D. Cazorla, F. Cuartero, G. Diaz, and E. Cambronero, “Model checking wireless sensor network security protocols: TinySec + LEAP + TinyPK,” *Telecommunication Systems*, vol. 40, pp. 91–99, 2009.

[128] C. Karlof and D. Wagner, “Secure routing in wireless sensor networks: attacks and countermeasures,” in *Sensor Network Protocols and Applications, 2003. Proceedings of the First IEEE. 2003 IEEE International Workshop on*, May 2003, pp. 113–127.

[129] N. Kettaf, H. Abouaissa, and P. Lorenz, “An efficient heterogeneous key management approach for secure multicast communications in ad hoc networks,” *Telecommunication Systems*, pp. 29–36, 2008.

[130] J. Kim and K. Kim, “A scalable and robust hierarchical key establishment for mission-critical applications over sensor networks,” *Telecommunication Systems*, pp. 1–12, 2011.

[131] A. Wander, N. Gura, H. Eberle, V. Gupta, and S. Shantz, "Energy analysis of public-key cryptography for wireless sensor networks," in *Pervasive Computing and Communications, 2005. PerCom 2005. Third IEEE International Conference on*, march 2005, pp. 324–328.

[132] A. Liu and P. Ning, "TinyECC: A configurable library for elliptic curve cryptography in wireless sensor networks," in *Proceedings of the 7th international conference on Information processing in sensor networks*, ser. IPSN '08. Washington, DC, USA: IEEE Computer Society, 2008, pp. 245–256.

[133] Y. W. Law, J. Doumen, and P. Hartel, "Survey and benchmark of block ciphers for wireless sensor networks," *ACM Trans. Sen. Netw.*, vol. 2, pp. 65–93, February 2006.

[134] T. Robertazzi and T. Robertazzi, "Advanced encryption standard (AES)," in *Basics of Computer Networking*, ser. SpringerBriefs in Electrical and Computer Engineering. Springer New York, 2012, pp. 73–77.

[135] D. W. Carman, P. S. Kruus, and B. J. Matt, "Constraints and approaches for distributed sensor network security," *NAI Labs Technical Report No. 00-10*, 2002.

[136] N. Fournel, M. Minier, and S. Ubeda, "Survey and benchmark of stream ciphers for wireless sensor networks," in *Information Security Theory and Practices. Smart Cards, Mobile and Ubiquitous Computing Systems*, ser. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, 2007, vol. 4462, pp. 202–214.

[137] P. Ganesan, R. Venugopalan, P. Peddabachagari, A. Dean, F. Mueller, and M. Sichitiu, "Analyzing and modeling encryption overhead for sensor network nodes," in *Proceedings of the 2nd ACM international conference on Wireless sensor networks and applications*, ser. WSNA '03. New York, NY, USA: ACM, 2003, pp. 151–159.

[138] A. D. Wood and J. A. Stankovic, "AMSecure: secure link-layer communication in tinyos for IEEE 802.15.4-based wireless sensor networks," in *Proceedings of the 4th international conference on Embedded networked sensor systems*, ser. SenSys '06. New York, NY, USA: ACM, 2006, pp. 395–396.

[139] X. Zhang, H. Heys, and C. Li, "Energy efficiency of symmetric key cryptographic algorithms in wireless sensor networks," in *Communications (QBSC), 2010 25th Biennial Symposium on*, may 2010, pp. 168–172.

[140] B. Sun, L. Osborne, Y. Xiao, and S. Guizani, "Intrusion detection techniques in mobile ad hoc and wireless sensor networks," *Wireless Communications, IEEE*, vol. 14, no. 5, pp. 56–63, october 2007.

[141] A. Nadeem and M. Howarth, "Protection of MANETs from a range of attacks using an intrusion detection and prevention system," *Telecommunication Systems*, pp. 1–12, 2011.

[142] S. Axelsson, "Intrusion detection systems: A survey and taxonomy," *Department of Computer Engineering, Chalmers University, Technical Report 99-15*, March 2000.

[143] A. Mishra, K. Nadkarni, and A. Patcha, "Intrusion detection in wireless ad hoc networks," *Wireless Communications, IEEE*, vol. 11, no. 1, pp. 48–60, feb 2004.

[144] S. Rajasegarar, C. Leckie, and M. Palaniswami, "Anomaly detection in wireless sensor networks," *Wireless Communications, IEEE*, vol. 15, no. 4, pp. 34–40, aug. 2008.

[145] P. Brutch and C. Ko, "Challenges in intrusion detection for wireless ad-hoc networks," in *Applications and the Internet Workshops, 2003. Proceedings. 2003 Symposium on*, jan. 2003, pp. 368–373.

[146] C. Eik Loo, M. Yong Ng, C. Leckie, and M. Palaniswami, "Intrusion detection for routing attacks in sensor networks," *International Journal of Distributed Sensor Networks*, vol. 2, no. 4, pp. 313–332, 2006.

[147] A. P. da Silva, M. H. Martins, B. P. Rocha, A. A. Loureiro, L. B. Ruiz, and H. C. Wong, "Decentralized intrusion detection in wireless sensor networks," in *Proceedings of the 1st ACM international workshop on Quality of service & security in wireless and mobile networks*, ser. Q2SWinet '05. New York, NY, USA: ACM, 2005, pp. 16–23.

[148] I. Onat and A. Miri, "An intrusion detection system for wireless sensor networks," in *Wireless And Mobile Computing, Networking And Communications, 2005. (WiMob'2005), IEEE International Conference on*, vol. 3, 2005, pp. 253–259.

[149] R. Roman, J. Zhou, and J. Lopez, "Applying intrusion detection systems to wireless sensor networks," in *Consumer Communications and Networking Conference, 2006. CCNC 2006. 3rd IEEE*, vol. 1, jan. 2006, pp. 640–644.

[150] S. Rajasegarar, C. Leckie, M. Palaniswami, and J. C. Bezdek, "Distributed anomaly detection in wireless sensor networks," in *Communication systems, 2006. ICCS 2006. 10th IEEE Singapore International Conference on*, oct. 2006, pp. 1–5.

[151] L. Yao, N. An, F. Gao, and G. Yu, “A clustered routing protocol with distributed intrusion detection for wireless sensor networks,” in *Proceedings of the joint 9th Asia-Pacific web and 8th international conference on web-age information management conference on Advances in data and web management*, ser. APWeb/WAIM’07. Berlin, Heidelberg: Springer-Verlag, 2007, pp. 395–406.

[152] X. Chen, K. Makki, K. Yen, and N. Pissinou, “Sensor network security: a survey,” *Communications Surveys Tutorials, IEEE*, vol. 11, no. 2, pp. 52–73, 2009.

[153] A. Perrig, J. Stankovic, and D. Wagner, “Security in wireless sensor networks,” *Commun. ACM*, vol. 47, pp. 53–57, June 2004.

[154] A. Vaseashta and S. Vaseashta, “A survey of sensor network security,” *Sensors & Transducers Journal*, vol. 94, no. 7, pp. 91–102, July 2008.

[155] S. A. Camtepe and B. Yener, “Key distribution mechanisms for wireless sensor networks: a survey,” Rensselaer Polytechnic Institute, NY, Tech. Rep., 2005.

[156] W. Znaidi and M. Minier, “Key establishment and management for WSNs,” *Telecommunication Systems*, pp. 1–13, 2010.

[157] L. A. Grieco, G. Boggia, S. Sicari, and P. Colombo, “Secure wireless multimedia sensor networks: A survey,” *Mobile Ubiquitous Computing, Systems, Services and Technologies, International Conference on*, vol. 0, pp. 194–201, 2009.

[158] M. Guerrero-Zapata, R. Zilan, J. Barcelo-Ordinas, K. Bicakci, and B. Tavli, “The future of security in wireless multimedia sensor networks,” *Telecommunication Systems*, vol. 45, pp. 77–91, 2010.

[159] H. Wang, D. Peng, W. Wang, H. Sharif, and H.-H. Chen, “Image transmissions with security enhancement based on region and path diversity in wireless sensor networks,” *Trans. Wireless. Comm.*, vol. 8, pp. 757–765, February 2009.

[160] ——, “Energy-aware adaptive watermarking for real-time image delivery in wireless sensor networks,” in *Communications, 2008. ICC ’08. IEEE International Conference on*, May 2008, pp. 1479–1483.

[161] H. Wang, M. Hempel, D. Peng, W. Wang, H. Sharif, and H.-H. Chen, “Index-based selective audio encryption for wireless multimedia sensor networks,” *Multimedia, IEEE Transactions on*, vol. 12, no. 3, pp. 215–223, april 2010.

[162] D. Kundur, W. Luh, U. Okorafor, and T. Zourntos, “Security and privacy for distributed multimedia sensor networks,” *Proceedings of the IEEE*, vol. 96, no. 1, pp. 112–130, jan. 2008.

[163] D. Kundur, T. Zourntos, and N. Mathai, “Lightweight security principles for distributed multimedia based sensor networks,” in *Signals, Systems and Computers, 2004. Conference Record of the Thirty-Eighth Asilomar Conference on*, vol. 1, 2004, pp. 368–372.

[164] Y. Yang, X. Wang, S. Zhu, and G. Cao, “SDAP: A secure hop-by-hop data aggregation protocol for sensor networks,” *ACM Trans. Inf. Syst. Secur.*, vol. 11, no. 4, pp. 18:1–18:43, Jul. 2008.

[165] L. Buttyán, P. Schaffer, and I. Vajda, “RANBAR: RANSAC-based resilient aggregation in sensor networks,” in *Proceedings of the fourth ACM workshop on Security of ad hoc and sensor networks*, ser. SASN ’06. New York, NY, USA: ACM, 2006, pp. 83–90.

[166] L. Zhang, H. Zhang, M. Conti, R. Di Pietro, S. Jajodia, and L. Mancini, “Preserving privacy against external and internal threats in WSN data aggregation,” *Telecommunication Systems*, pp. 1–14, 2011.

[167] W. Du, L. Fang, and N. Peng, “LAD: Localization anomaly detection for wireless sensor networks,” *Journal of Parallel and Distributed Computing*, vol. 66, no. 7, pp. 874–886, 2006, special Issue 19th International Parallel and Distributed Processing Symposium - IPDPS 2005.

[168] D. Liu, P. Ning, A. Liu, C. Wang, and W. K. Du, “Attack-resistant location estimation in wireless sensor networks,” *ACM Trans. Inf. Syst. Secur.*, vol. 11, no. 4, pp. 22:1–22:39, Jul. 2008.

[169] V. Bhuse and A. Gupta, “Anomaly intrusion detection in wireless sensor networks,” *Journal of High Speed Networks*, vol. 15, pp. 33Ű–51, 2006.

[170] I. Almalkawi, M. Guerrero Zapata, and J. Al-Karaki, “A Cross-Layer-Based Clustered Multipath Routing with QoS-Aware Scheduling for Wireless Multimedia Sensor Networks,” *International Journal of Distributed Sensor Networks*, vol. 2012, Abril 2012.

[171] D. Eastlake and P. Jones, “US Secure Hash Algorithm 1 (SHA1),” Tech. Rep. 3174, 9 2001.

[172] M. Peng, H. Chen, Y. Xiao, S. Ozdemir, A. V. Vasilakos, and J. Wu, "Impacts of sensor node distributions on coverage in sensor networks," *J. Parallel Distrib. Comput.*, vol. 71, no. 12, pp. 1578–1591, Dec. 2011.

[173] N. Li, N. Zhang, S. K. Das, and B. Thuraisingham, "Privacy preservation in wireless sensor networks: A state-of-the-art survey," *Ad Hoc Netw.*, vol. 7, no. 8, pp. 1501–1514, Nov. 2009.

[174] J. Shi, R. Zhang, and Y. Zhang, "Secure range queries in tiered sensor networks," in *INFOCOM 2009, IEEE*, april 2009, pp. 945–953.

[175] W. He, X. Liu, H. Nguyen, K. Nahrstedt, and T. Abdelzaher, "PDA: Privacy-preserving data aggregation in wireless sensor networks," in *INFOCOM 2007. 26th IEEE International Conference on Computer Communications. IEEE*, may 2007, pp. 2045–2053.

[176] A. Coen Porisini and S. Sicari, "SeDAP: Secure data aggregation protocol in privacy aware wireless sensor networks," in *Sensor Systems and Software*, ser. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, G. Par and P. Morrow, Eds. Springer Berlin Heidelberg, 2011, vol. 57, pp. 135–150.

[177] A. Mishra, K. Nadkarni, and A. Patcha, "Intrusion detection in wireless ad hoc networks," *Wireless Communications, IEEE*, vol. 11, no. 1, pp. 48–60, feb 2004.

[178] S. Rajasegarar, C. Leckie, M. Palaniswami, and J. C. Bezdek, "Distributed anomaly detection in wireless sensor networks," in *Communication systems, 2006. ICCS 2006. 10th IEEE Singapore International Conference on*, oct. 2006, pp. 1–5.

[179] C. Ozturk, Y. Zhang, and W. Trappe, "Source-location privacy in energy-constrained sensor network routing," in *Proceedings of the 2nd ACM workshop on Security of ad hoc and sensor networks*, ser. SASN '04. New York, NY, USA: ACM, 2004, pp. 88–93.

[180] S. Pai, S. Bermudez, S. Wicker, M. Meingast, T. Roosta, S. Sastry, and D. Mulligan, "Transactional confidentiality in sensor networks," *Security Privacy, IEEE*, vol. 6, no. 4, pp. 28–35, july-aug. 2008.

[181] J. Hightower and G. Borriello, "Location systems for ubiquitous computing," *Computer*, vol. 34, no. 8, pp. 57–66, aug 2001.

[182] Y. Yang, S. Zhu, G. Cao, and T. LaPorta, “An active global attack model for sensor source location privacy: Analysis and countermeasures,” in *Security and Privacy in Communication Networks*, ser. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, Y. Chen, T. Dimitriou, and J. Zhou, Eds. Springer Berlin Heidelberg, 2009, vol. 19, pp. 373–393.

[183] P. Kamat, Y. Zhang, W. Trappe, and C. Ozturk, “Enhancing source-location privacy in sensor network routing,” in *Proceedings of the 25th IEEE International Conference on Distributed Computing Systems*, ser. ICDCS ’05. Washington, DC, USA: IEEE Computer Society, 2005, pp. 599–608.

[184] L. Kang, “Protecting location privacy in large-scale wireless sensor networks,” in *Communications, 2009. ICC ’09. IEEE International Conference on*, june 2009, pp. 1–6.

[185] Y. Xi, L. Schwiebert, and W. Shi, “Preserving source location privacy in monitoring-based wireless sensor networks,” in *Parallel and Distributed Processing Symposium, 2006. IPDPS 2006. 20th International*, april 2006, pp. 1–8.

[186] D. Braginsky and D. Estrin, “Rumor routing algorthim for sensor networks,” in *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*, ser. WSNA ’02. New York, NY, USA: ACM, 2002, pp. 22–31.

[187] A. Broder and M. Mitzenmacher, “Network applications of bloom filters: A survey,” in *Internet Mathematics*, vol. 1, 2002, pp. 636–646.

[188] S. Shakkottai, “Asymptotics of query strategies over a sensor network,” in *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, vol. 1, march 2004.

[189] Y. Li, L. Lightfoot, and J. Ren, “Routing-based source-location privacy protection in wireless sensor networks,” in *Electro/Information Technology, 2009. eit ’09. IEEE International Conference on*, june 2009, pp. 29–34.

[190] Y. Li and J. Ren, “Mixing ring-based source-location privacy in wireless sensor networks,” in *Computer Communications and Networks, 2009. ICCCN 2009. Proceedings of 18th Internatonal Conference on*, aug. 2009, pp. 1–6.

[191] W. Wei-ping, C. Liang, and W. Jian-xin, “A source-location privacy protocol in WSN based on locational angle,” in *Communications, 2008. ICC ’08. IEEE International Conference on*, may 2008, pp. 1630–1634.

[192] H. Chen and W. Lou, “From nowhere to somewhere: Protecting end-to-end location privacy in wireless sensor networks,” in *Performance Computing and Communications Conference (IPCCC), 2010 IEEE 29th International*, dec. 2010, pp. 1–8.

[193] Y. Yang, M. Shao, S. Zhu, B. Urgaonkar, and G. Cao, “Towards event source unobservability with minimum network traffic in sensor networks,” in *Proceedings of the first ACM conference on Wireless network security*, ser. WiSec ’08. New York, NY, USA: ACM, 2008, pp. 77–88.

[194] K. Bicakci, H. Gultekin, B. Tavli, and I. E. Bagci, “Maximizing lifetime of event-unobservable wireless sensor networks,” *Comput. Stand. Interfaces*, vol. 33, no. 4, pp. 401–410, Jun. 2011.

[195] W. Yang and W. T. Zhu, “Protecting source location privacy in wireless sensor networks with data aggregation,” in *Proceedings of the 7th international conference on Ubiquitous intelligence and computing*, ser. UIC’10. Berlin, Heidelberg: Springer-Verlag, 2010, pp. 252–266.

[196] K. Mehta, D. Liu, and M. Wright, “Protecting location privacy in sensor networks against a global eavesdropper,” *Mobile Computing, IEEE Transactions on*, vol. 11, no. 2, pp. 320–336, feb. 2012.

[197] Y. Fan, J. Chen, X. Lin, and X. Shen, “Preventing traffic explosion and achieving source unobservability in multi-hop wireless networks using network coding,” in *Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE*, dec. 2010, pp. 1–5.

[198] M. Shao, Y. Yang, S. Zhu, and G. Cao, “Towards statistically strong source anonymity for sensor networks,” in *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*, april 2008, pp. 51–55.

[199] A. Abbasi, A. Khonsari, and M. S. Talebi, “Source location anonymity for sensor networks,” in *Proceedings of the 6th IEEE Conference on Consumer Communications and Networking Conference*, ser. CCNC’09. Piscataway, NJ, USA: IEEE Press, 2009, pp. 588–592.

[200] B. Alomair, A. Clark, J. Cuellar, and R. Poovendran, “Toward a statistical framework for source anonymity in sensor networks,” *IEEE Transactions on Mobile Computing*, vol. 12, no. 2, pp. 248–260, Feb. 2013.

[201] Y. Ouyang, Z. Le, J. Ford, and F. Makedon, "PrivaSense: providing privacy protection for sensor networks," in *Proceedings of the 5th international conference on Embedded networked sensor systems*, ser. SenSys '07. New York, NY, USA: ACM, 2007, pp. 415–416.

[202] Y. Ouyang, Z. Le, D. Liu, J. Ford, and F. Makedon, "Source location privacy against laptop-class attacks in sensor networks," in *Proceedings of the 4th international conference on Security and privacy in communication netowrks*, ser. SecureComm '08. New York, NY, USA: ACM, 2008, pp. 1–10.

[203] Y. Ouyang, X. Le, G. Chen, J. Ford, and F. Makedon, "Entrapping adversaries for source protection in sensor networks," in *World of Wireless, Mobile and Multimedia Networks, 2006. WoWMoM 2006. International Symposium on a*, 0-0 2006, pp. 10–34.

[204] K. Mehta, D. Liu, and M. Wright, "Location privacy in sensor networks against a global eavesdropper," in *Network Protocols, 2007. ICNP 2007. IEEE International Conference on*, oct. 2007, pp. 314–323.

[205] M. Shao, W. Hu, S. Zhu, G. Cao, S. Krishnamurthy, and T. L. Porta, "Cross-layer enhanced source location privacy in sensor networks," in *Proceedings of the 6th Annual IEEE communications society conference on Sensor, Mesh and Ad Hoc Communications and Networks*, ser. SECON'09. Piscataway, NJ, USA: IEEE Press, 2009, pp. 324–332.

[206] J. Deng, R. Han, and S. Mishra, "Intrusion tolerance and anti-traffic analysis strategies for wireless sensor networks," in *Dependable Systems and Networks, 2004 International Conference on*, june-1 july 2004, pp. 637–646.

[207] D. Jing, H. Richard, and M. Shivakant, "Countermeasures against traffic analysis attacks in wireless sensor networks," in *Proceedings of the First International Conference on Security and Privacy for Emerging Areas in Communications Networks*, ser. SECURECOMM '05. Washington, DC, USA: IEEE Computer Society, 2005, pp. 113–126.

[208] E. M. Shakshuki, T. R. Sheltami, N. Kang, and X. Xing, "Tracking anonymous sinks in wireless sensor networks," in *Proceedings of the 2009 International Conference on Advanced Information Networking and Applications*, ser. AINA '09. Washington, DC, USA: IEEE Computer Society, 2009, pp. 510–516.

[209] J. Deng, R. Han, and S. Mishra, “Decorrelating wireless sensor network traffic to inhibit traffic analysis attacks,” in *In Elsevier Pervasive and Mobile Computing Journal, Special Issue on Security in Wireless Mobile Computing Systems*, vol. 2, 2006, pp. 159–186.

[210] E. C.-H. Ngai, “On providing sink anonymity for sensor networks,” in *Proceedings of the 2009 International Conference on Wireless Communications and Mobile Computing: Connecting the World Wirelessly*, ser. IWCMC ’09. New York, NY, USA: ACM, 2009, pp. 269–273.

[211] E. C.-H. Ngai and I. Rodhe, “On providing location privacy for mobile sinks in wireless sensor networks,” in *Proceedings of the 12th ACM international conference on Modeling, analysis and simulation of wireless and mobile systems*, ser. MSWiM ’09. New York, NY, USA: ACM, 2009, pp. 116–123.

[212] Y. Jian, S. Chen, Z. Zhang, and L. Zhang, “Protecting receiver-location privacy in wireless sensor networks,” in *INFOCOM 2007. 26th IEEE International Conference on Computer Communications. IEEE*, may 2007, pp. 1955–1963.

[213] W. Conner, T. Abdelzaher, and K. Nahrstedt, “Using data aggregation to prevent traffic analysis in wireless sensor networks,” in *Proceedings of the Second IEEE international conference on Distributed Computing in Sensor Systems*, ser. DCOSS’06. Berlin, Heidelberg: Springer-Verlag, 2006, pp. 202–217.

[214] U. Acharya and M. Younis, “Increasing base-station anonymity in wireless sensor networks,” *Ad Hoc Netw.*, vol. 8, no. 8, pp. 791–809, Nov. 2010.

[215] X. Li, X. Wang, N. Zheng, Z. Wan, and M. Gu, “Enhanced location privacy protection of base station in wireless sensor networks,” in *Mobile Ad-hoc and Sensor Networks, 2009. MSN ’09. 5th International Conference on*, dec. 2009, pp. 457–464.

[216] J. Deng, R. Han, and S. Mishra, “Enhancing base station security in wireless sensor networks,” Department of Computer science, University of Colorado, Tech. Rep., April 2003.

[217] L. Yao, L. Kang, P. Shang, and G. Wu, “Protecting the sink location privacy in wireless sensor networks,” *Personal and Ubiquitous Computing*, pp. 1–11, 2012.

[218] S. Chang, Y. Qi, H. Zhu, M. Dong, and K. Ota, “Maelstrom: receiver-location preserving in wireless sensor networks,” in *Proceedings of the 6th international conference on Wireless algorithms, systems, and applications*, ser. WASA’11. Berlin, Heidelberg: Springer-Verlag, 2011, pp. 190–201.

[219] R. Vogt, M. A. Nascimento, and J. Harms, "On the trade-off between user-location privacy and queried-location privacy in wireless sensor networks," in *Proceedings of the 8th International Conference on Ad-Hoc, Mobile and Wireless Networks*, ser. ADHOC-NOW '09. Berlin, Heidelberg: Springer-Verlag, 2009, pp. 241–254.

[220] M. Shao, S. Zhu, W. Zhang, G. Cao, and Y. Yang, "pDCS: Security and privacy support for data-centric sensor networks," *Mobile Computing, IEEE Transactions on*, vol. 8, no. 8, pp. 1023–1038, aug. 2009.

[221] S. Ratnasamy, B. Karp, L. Yin, F. Yu, D. Estrin, R. Govindan, and S. Shenker, "GHT: A geographic hash table for data-centric storage." ACM Press, 2002, pp. 78–87.

[222] P. Winter and M. Zachariasen, "Euclidean steiner minimum trees: An improved exact algorithm," *Networks*, vol. 30, no. 3, pp. 149–166, 1997.

[223] J. Albath and S. Madria, "Secure hierarchical data aggregation in wireless sensor networks," in *Wireless Communications and Networking Conference, 2009. WCNC 2009. IEEE*, april 2009, pp. 1–6.

[224] R. Zhang, Y. Zhang, and K. Ren, "Distributed privacy-preserving access control in sensor networks," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 23, no. 8, pp. 1427–1438, aug. 2012.

[225] M. Li, W. Lou, and K. Ren, "Data security and privacy in wireless body area networks," *Wireless Commun.*, vol. 17, no. 1, pp. 51–58, Feb. 2010.

[226] B. Carbunar, Y. Yu, W. Shi, M. Pearce, and V. Vasudevan, "Query privacy in wireless sensor networks," *ACM Trans. Sen. Netw.*, vol. 6, no. 2, pp. 14:1–14:34, Mar. 2010.

[227] L. Sweeney, "k-anonymity: a model for protecting privacy," *Int. J. Uncertain. Fuzziness Knowl.-Based Syst.*, vol. 10, no. 5, pp. 557–570, Oct. 2002.

[228] M. Reed, P. Syverson, and D. Goldschlag, "Anonymous connections and onion routing," *Selected Areas in Communications, IEEE Journal on*, vol. 16, no. 4, pp. 482–494, may 1998.

[229] P. Syverson, M. Reed, and D. Goldschlag, "Onion routing access configurations," in *DARPA Information Survivability Conference and Exposition, 2000. DISCEX '00. Proceedings*, vol. 1, 2000, pp. 34–40.

[230] E. De Cristofaro, X. Ding, and G. Tsudik, "Privacy-preserving querying in sensor networks," in *Computer Communications and Networks, 2009. ICCCN 2009. Proceedings of 18th Internatonal Conference on*, aug. 2009, pp. 1–6.

[231] S. Misra and G. Xue, "Efficient anonymity schemes for clustered wireless sensor networks," *Int. J. Sen. Netw.*, vol. 1, no. 1/2, pp. 50–63, Sep. 2006.

[232] Y. Ouyang, Z. Le, Y. Xu, N. Triandopoulos, S. Zhang, J. Ford, and F. Makedon, "Providing anonymity in wireless sensor networks," in *Pervasive Services, IEEE International Conference on*. IEEE, 2007, pp. 145–148.

[233] Y. Li and J. Ren, "Preserving source-location privacy in wireless sensor networks," in *Sensor, Mesh and Ad Hoc Communications and Networks, 2009. SECON '09. 6th Annual IEEE Communications Society Conference on*, june 2009, pp. 1–9.

[234] P. Kamat, X. Wenyuan, W. Trappe, and Z. Yanyong, "Temporal privacy in wireless sensor networks," in *Distributed Computing Systems, 2007. ICDCS '07. 27th International Conference on*, june 2007, p. 23.

[235] P. Kamat, W. Xu, W. Trappe, and Y. Zhang, "Temporal privacy in wireless sensor networks: Theory and practice," *ACM Trans. Sen. Netw.*, vol. 5, no. 4, pp. 28:1–28:24, Nov. 2009.

[236] I. Almalkawi, M. Guerrero Zapata, and J. Al-Karaki, "A Secure Cluster-Based Multipath Routing Protocol for WMSNs," *Sensors 11*, vol. 4, pp. 4401–4424, March 2011.

[237] J. Yao, "Source-location privacy based on directed greedy walk in wireless sensor networks," in *Wireless Communications Networking and Mobile Computing (WiCOM), 2010 6th International Conference on*, 2010, pp. 1–4.

Declaration

I herewith declare that I have produced this work without the prohibited assistance of third parties and without making use of aids other than those specified; notions taken over directly or indirectly from other sources have been identified as such. This work has not previously been presented in identical or similar form to any other Spanish or foreign examination board.

The thesis work was conducted from `January 2009` to `July 2013` under the supervision of Prof. Manel Guerrero Zapata.

Islam Almalkawi,
Barcelona, July 2013.