QOS-BASED AND SECURE MULTIPATH ROUTING

IN WIRELESS SENSOR NETWORKS

Hind Alwan

A Thesis

In the Department

of

Electrical and Computer Engineering

Presented in Partial Fulfillment of the Requirements

For the Degree of

Doctor of Philosophy at

Concordia University

Montreal, Quebec, Canada

September 2013

© Hind Alwan, 2013

CONCORDIA UNIVERSITY
SCHOOL OF GRADUATE STUDIES

This is to certify that the thesis prepared

By: Hind Alwan

Entitled: QoS-Based and Secure Multipath Routing in Wireless Sensor Networks

and submitted in partial fulfillment of the requirements for the degree of

Doctor of Philosophy (Electrical and Computer Engineering)

complies with the regulations of the University and meets the accepted standards with respect to originality and quality.

Signed by the final examining committee:

Dr. R. Dssouli — Chair

Dr. S. Sampalli — External Examiner

Dr. B. Fung — External to Program

Dr. Y. Shayan — Examiner

Dr. A. Hamou-Lhadj — Examiner

Dr. A. Agarwal — Thesis Supervisor

Approved by

Chair of Department or Graduate Program Director

September 13, 2013

Dean of Faculty

ABSTRACT

QoS-Based and Secure Multipath Routing in Wireless Sensor Networks

Hind Alwan, Ph.D.

Concordia University, 2013

With the growing demand for quality of service (QoS) aware routing protocols in wireless networks, QoS-based routing has emerged as an interesting research topic. A QoS guarantee in wireless sensor networks (WSNs) is difficult and more challenging due to the fact that the available resources of sensors and the various applications running over these networks have different constraints in their nature and requirements. Furthermore, due to the increased use of sensor nodes in a variety of application fields, WSNs need to handle heterogeneous traffic with diverse priorities to achieve the required QoS.

In this thesis, we investigate the problem of providing multi-QoS in routing protocols for WSNs. In particular, we investigate several aspects related to the application requirements and the network states and resources.

We present multi-objective QoS aware routing protocol for WSNs that uses the geographic routing mechanism combined with the QoS requirements to meet diverse application requirements by considering the changing conditions of the network. The protocol formulates the application requirements with the links available resources and conditions to design heuristic neighbor discovery algorithms. Also, with the unlimited resource at the sink node, the process of selecting the routing path/paths is assigned to the

sink. Paths selection algorithms are designed with various goals in order to extend network lifetime, enhance the reliability of data transmission, decrease end-to-end delay, achieve load balancing and provide fault tolerance.

We also develop a cross-layer routing protocol that combines routing at network layer and the time scheduling at the MAC layer with respect to delay and reliability in an energy efficient way. A node-disjoint multipath routing is used and a QoS-aware priority scheduling considering MAC layer is proposed to ensure that real time and non-real time traffic achieve their desired QoS while alleviating congestion in the network.

Additionally, we propose new mechanism for secure and reliable data transmission in multipath routing for WSNs. Different levels of security requirements are defined and depending on these requirements, a selective encryption scheme is introduced to encrypt selected number of coded fragments in order to enhance security and thereby reduce the time required for encryption. Node-disjoint multipath routing combined with source coding is used in order to enhance both security and reliability of data transmission. Also, we develop an allocation strategy that allocates fragments on paths to enhance both the security and probability of successful data delivery.

Analysis and extensive simulation are conducted to study the performance of all the above proposed protocols.

ACKNOWLEDGEMENT

I would like to express my deep and sincere gratitude to my advisor, Prof. Anjali Agarwal for her insightful guidance, constant encouragement and support. This thesis would not be possible without her rich expertise, excellent judgement, enthusiasm and dedication to top-quality research. It is my fortune and honour to have an advisor like her.

Deepest gratitude is also due to the members of the supervisory committee, Dr. Yousef Shayan, Dr. Abdelwahab Hamou-Lhadj, Dr. Benjamin Fung and my external examiner Dr. Srinivas Sampalli. I would also like to thank Dr. Ayda Basyouni for her great support.

Next, I would like to thank my family for always supporting me in my various undertakings, up to and including this one, and I am sure in the next ones as well.

My final words go to my parents, without whom I would never have been able to achieve so much.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

ACK	Acknowledgement
AES	Advanced Encryption Standard
ARQ	Auto Repeat reQuest
DAQ	Dynamic Adjust reQuest
DD	Directed Diffusion
EC	Erasure Coding
EDF	Early Deadline First
FEC	Forward Error Correction
FIFO	First In First Out
ID	Identification
IEEE	Institute of Electrical and Electronic Engineering
ITU	International Telecommunication Union
MATLAB	Matrix Laboratory
NS-2	Network Simulator Version 2
QoS	Quality of Service
RC5	Rivest Cipher Version 5
RREP	Route REPly
RREQ	Route REQuest
RS	Reed-Solomon
SNR	Signal-to-Noise Ratio
SS	Secret Sharing

WSNs	Wireless Sensor Networks

Chapter 1

INTRODUCTION

The open nature of wireless sensor networks (WSNs) [1] recently attracted significant research attention. The wide range of WSNs applications [2] in hostile environments both in civil and military domains where human participation may be too dangerous, sensor networks need to provide a robust service. The fast growth of wireless networks indicate that the network has potential to design many new routing protocols for handling emergency, military and disaster relief operations that require real time information for efficient coordination and planning and to support different quality of service (QoS) requirements [3].

Sensor networks consist of many, normally very small size devices (sensors/nodes) that monitor a certain phenomenon. The main function of these networks is to gather information about the environment and transmit the information to interested users. The use of WSNs in different environments allows the use of many different types of sensors such as seismic, magnetic, thermal, visual, infrared and radar that are capable to monitor different kinds of information that may have different levels of importance given that different applications may have different QoS requirements. Though each individual sensor may have severe resource constraint in terms of energy, memory, communication and computation capabilities; large number of them may collectively monitor the physical world, distribute and process information upon critical environmental events.

Advances in WSNs have enabled a wide range of applications across many fields. Many of these applications have high QoS requirements in terms of end-to-end data delivery delay, reliability and security. For instance, for a real time application like rescue services to detect the location of survivors, delay or failed delivery of data may not be allowed, while it may be acceptable for habitat monitoring of the dynamics and movements of animals. Therefore routing protocols of such networks should have a mechanism to provide reliable and fault-tolerant communication, quick reconfiguration and minimum consumption of energy. Additionally, WSNs' design requirements change with the application, this introduces various design objectives for routing protocols such as energy efficiency, reliability, low delay. Also, security is another important issue to be considered while designing routing protocols in WSNs, as these networks may be deployed in hostile areas.

QoS guarantees in WSNs are difficult and more challenging due to the fact that the available resources of sensors and the various applications running over these networks have different constraints in their nature and requirements. Due to these limitations, the routing techniques developed for other types of networks are not sufficient for WSNs.

1.1 RESEARCH MOTIVATION AND CHALLENGES

The basic function of a QoS-aware routing protocol is to find an optimal route that satisfies a single objective with respect to the links' constraints. Due to the extreme energy constraints of sensor nodes, most of the proposed routing protocols for WSNs have focused on energy efficiency in order to maximize network lifetime [4] [5] [6]. Compared with routing decision using single objective or single link constraint, the

multiple objectives or multi-constraint routing decision is very different. Contradiction may happen due to lack of a standard and a uniform measure for the links' constraints, which can be classified as additive, multiplicative or concave. For additive metrics like delay, the end-to end cost of the path is the sum of the individual link values. For a multiplicative metric like path reliability it is the product of the link qualities along the path. In case of the concave metric like the overall bandwidth of a path it is equal to the minimum, which is the bottleneck value of a link along the path. Therefore, the problem of determining a QoS route that satisfies the multiple constraints has been proven to be NP-complete [7].

Most of the proposed QoS-aware routing protocols for WSNs characterize the network with a single metric such as hop count, delay, reliability, security or energy consumption algorithms to compute paths. However, due to the extreme energy constraints of sensor nodes, most of these protocols focused on energy efficiency in order to maximize network lifetime [8-10]. Yet, many routing protocols that have been designed to provide QoS are more appropriate in some situations having better performance while not suitable in other situations having major limitations. Moreover, supporting different and multiple QoS requirements, and modeling the network as path-based and link-based multiple metrics such as energy, delay and reliability of data transmission, were not considered in the aforementioned works. However, in order to support different and multiple QoS requirements, the protocols need to characterize the network with multiple metrics such as energy, delay and data loss probability. The basic problem is therefore to find a path that satisfies the multiple constraints for QoS routing while respecting the energy constraints of sensor nodes. For these reasons, the proposed

algorithms must be simple so as to respect the limited computation at sensor nodes and should provide an energy efficient solution at every layer of the protocol stack in order to prolong the network lifetime.

Although collective effort of all the communication protocol stack entities is essential for QoS provisioning, MAC layer possesses a particular importance among them since it is responsible for scheduling and allocation of the shared wireless channel and all other upper layer protocols are bound to that. Thus, the MAC layer plays a key role for QoS provisioning and dominates the performance of the QoS support. This calls for a suitable routing protocol tailored to achieve the application-specific QoS and that respects the characteristics of WSNs. Moreover, an efficient allocation of network resources to satisfy the different QoS requirements is the primary aim of a QoS-based routing protocol.

This thesis is motivated by the lack of research in providing QoS guarantees for traffic flows in WSNs for different applications that have different QoS requirements and the lack of research that allows requirements such as timeliness and reliable data delivery to be addressed and traded against each other within the same context while respecting the energy constraints of sensor nodes. Furthermore, cross-layer design has proved to be effective in enhancing the network performance and hence may be integrated in the development of QoS-aware routing protocols for WSNs. Thus, the problem of routing protocol design should not be considered separately from the problem of other layers like MAC layer.

Furthermore, from the viewpoint of security, this thesis is motivated mainly by the observations that most traditional encryption algorithms are complex and may introduce a

severe delay in sensor nodes. For instance, the encryption time of each 128-bit block using the AES algorithm is about 1.8 ms on a MicaZ platform [11]. To make encryption feasible for energy constrained and delay sensitive applications while still maintaining a robust security protection, limited number of packets contingent to different levels of security requirements need to be encrypted in order to enhance data transmission security at lower cost than full packet encryption. Combining the energy efficient techniques used to enhance data security and data QoS is vital to be investigated.

1.2 THESIS CONTRIBUTION

The design of QoS routing protocols in WSNs is a challenging issue. Most of the existing protocols are only suitable for specific types of applications and do not work well in large-scale applications. The goal of this research is to explore efficient multipath routing and QoS provisioning protocols in WSNs. The main contributions of this thesis can be summarized as follows:

1- Solve the conflicts between the requirements and the constraints of WSNs. We formulate the multi-constrained QoS routing problem as a multi-objective constrained optimization problem to determine multipath routes that satisfy different QoS requirements as follows:

   a. We define the required QoS parameters of interest: end-to-end reliability, end-to-end delay and network lifetime.

   b. We propose a novel heuristic mechanism in WSNs to provide multi-objective QoS routing for different applications.

c. The problem of providing QoS routing is formulated as link-based and path-based metrics. In the link metrics, sensor nodes need to consider the distance to sink as well as the application requirements in order to calculate the total cost function of a link that is used to select next hop. Thus, sensor nodes need to have the information of its direct neighboring nodes only. However, in path-based metrics and benefit from the fact that the sink has unlimited resources, the sink is responsible for selecting the routing paths, the number of these paths, and the allocation strategy of data packet on each path in order to achieve the end-to-end requirements in terms of reliability and delay as well as to extend the network lifetime.

2- Adopt cross-layer design by sharing information between MAC and Network layers in order to select the best next node. The process at the network layer comes up with the optimal decision based on the MAC layer parameters. Specifically, we propose to produce a congestion control protocol to work under both single and multipath routing scenarios. The proposed protocol implements per-hop QoS-aware priority scheduling and considers the parameters of MAC layer to achieve the desired QoS.

3- We design a new mechanism for secure multipath data transmission in WSNs, derived from node-disjoint multipath and combined with source coding in order to enhance both security and reliability of data transmission in the network.

a. We define different levels of security requirements and depending on these requirements, a selective encryption scheme is introduced to encrypt selected

number of coded fragments in order to enhance security and reduce the time required for encryption.

b. An allocation strategy that allocates fragments on paths is introduced to enhance both the security and probability of successful data delivery.

1.3 THESIS OUTLINE

Chapter 1 has given an introduction to WSNs as well as overview and scope of this research, the remainder of this thesis is organized into six additional chapters.

Chapter 2 begins by providing background and general consideration on the design of WSNs routing protocols and the motivations behind using multipath routing approach in WSNs to achieve load balancing, increase reliability and to provide fault tolerance. Also, this chapter reviews the main QoS metrics and constraints and the techniques used to provide QoS routing in WSNs and presents a discussion focusing particularly on the security issue of routing protocols in WSNs as well as review some possible network layer attacks in WSNs and the mechanisms used to secure the multipath routing protocols.

Chapter 3 presents detailed review of some state-of-the-art QoS-based routing protocols without and with congestion control mechanisms as well as reviews the secured multipath routing protocols proposed for WSNs.

Chapter 4 describes in detail the proposed multi-objective routing protocol. The network model, structures of the control messages, the strategies used to select next node as well as the algorithms used to select the routing paths and the number of these paths

are discussed in order to achieve the requested QoS in terms of delay, reliability while extending the network lifetime.

In Chapter 5 and based on the node-disjoint multipath proposed in Chapter 4, the link and path cost functions are modified to include the amount of load at sensor nodes in order to provide the requested QoS while avoiding congestion in the network. We study the effect of different parameters on providing the requested requirements and we propose a cross layer QoS-aware scheduling mechanism for WSNs with respect to delay and reliability in an energy efficient way. In the proposed QoS-aware priority scheduling, traffic is classified and prioritized according to their timeline requirement into real time and non-real time traffic. Real time traffic is assigned higher priority than non-real time traffic in order to achieve their desired QoS while alleviating congestion in the network. The developed model is validated through simulation that done using an object oriented programming language, C++, and NS 2.35 simulator.

In Chapter 6, we propose a secure and reliable mechanism for data transmission in WSN. We use node-disjoint multipath combined with source coding in order to enhance the security and the reliability of data transmission. Also, we defined different levels of security requirements and depending on these requirements, a selective encryption scheme is introduced to encrypt selected number of coded fragments in order to enhance security while reducing the time and energy required for encryption. Finally, an allocation strategy that allocates fragments on different paths is proposed. The developed model is validated through analysis results and simulation.

Finally, in Chapter 7 we discuss the conclusions and the future work.

Chapter 2

BACKGROUND AND GENERAL CONSIDERATION

Future WSNs are expected to carry different traffic as well as data to serve both real and non-real time applications. Therefore, the quality of the data delivered to support diverse applications is very important. QoS-aware routing in WSNs is difficult and more challenging due to the fact that the available resources of sensors and the various applications running over these networks have different constraints in their nature and requirements.

2.1 ROUTING PROTOCOLS IN WIRELESS SENSOR NETWORKS - CHARACTERISTIC AND CHALLENGES

Routing protocols for other wireless networks like mobile ad hoc networks or cellular networks cannot be directly applied to WSNs due to the typical characteristics of WSNs, such as severe resource constraints and harsh environmental conditions in addition to the existing design challenges in WSNs like energy consumption, node deployment, QoS requirements, data aggregation and node mobility. For example, for the deployment of a large number of sensor nodes in WSNs it will not be possible to build a global addressing scheme as the overhead of ID maintenance is high. A careful resource management is also required, since each sensor node depends on energy for its activities; thus the failure of one node or link due to its limited battery lifetime, hardware breakdown, communication errors, or malicious attack can affect the entire network.

Generated data traffic has significant redundancy in it since multiple sensors may generate same data within the area of a phenomenon. Such redundancy needs to be exploited by the routing protocols to improve energy and bandwidth utilization. Moreover network scalability, which is the ability of the network to grow without extremely increasing the overhead, and the need to frequent topological changes required by some WSNs application place more challenges on routing protocols. Many routing protocols considering the unique characteristics of WSNs are covered in survey articles presented in [12, 13].

In general, routing protocols proposed for WSNs [1-13] can be classified into three groups depending on the methods used for finding the path, namely, proactive routing in which all paths are computed and maintained in advance and stored in a routing table, reactive routing where all paths are created on demand, and hybrid routing which is a mix of the both the groups. However, in QoS-based routing protocols, the network has to balance its traffic while improving the network performance. WSNs inherit most of the QoS challenges from traditional wireless networks, such as time varying channels and unreliable links. Moreover, in many applications, to extend the network lifetime is considered more important than the quality of data, and this is related to the reduction of the energy dissipation in the sensor nodes. Thus, a network requires an energy-aware routing protocol. For real time applications, data should be delivered in time or otherwise data is considered useless. In this case, the network requires a timeliness-aware routing protocol. However, in other applications, a reliable routing protocol is used since the reliability of data transmission in the network is considered as an important issue.

As a result, the design of routing protocols in WSNs is influenced by many challenging factors. These factors must be overcome before efficient communication can be achieved in WSNs. Therefore, many new algorithms have been proposed for the problem of routing data in sensor networks. These routing mechanisms have considered the characteristics of sensor nodes along with the application and architecture requirements.

2.1.1 Single Path Routing

In single path protocols, the source node selects a single path which can satisfy the application requirements to transmit data towards the sink. Most of the existing routing protocols in WSNs are designed based on the single path routing strategy [14] to deliver data to the destination since it is simple and consumes less energy than multipath routing.

However, in this approach any path is vulnerable to node and link failures, thus acknowledgements and retransmissions are implemented to recover the lost data resulting in large amount of additional traffic and delays in the network. Thus, in critical situations new path needs to be discovered to maintain data transmission from the source to the sink, and such path discovery results in extra energy consumption, overhead, delay and may significantly reduce the network performance. Furthermore, single path routing protocol are incapable of load balancing traffic in the network. Therefore, single path routing technique cannot be considered effective techniques in WSNs due to the resource constraints and the unreliability of wireless links.

2.1.2 Multipath Routing

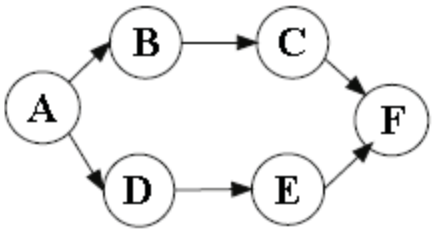
Multipath routing is the most popular approach to improve data transmission reliability, support congestion control and QoS as well as provide fault tolerance in the

network. Fault tolerance ensures that a system is available for use without any interruption in the presence of faults; thus fault tolerance increases the reliability, availability, and consequently dependability of the system. Multipath routing provides additional benefits of load balancing and bandwidth aggregation. The performance gains that can be achieved through using multipath routing approaches in WSNs can be summarized as;

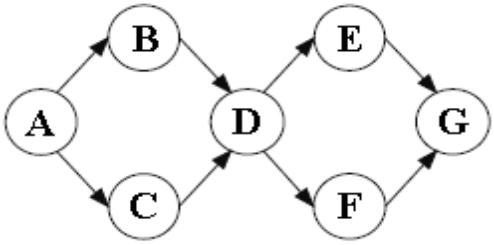
- Enhance reliability
- Provide fault tolerance
- Provide load balancing and bandwidth aggregation
- Improve QoS

Two mechanisms [15] are used to establish multiple paths: disjoint and braided multipath. In disjoint multipath (Figure 2.1(a) and (b)), a number of alternate paths are constructed as node-disjoint or link-disjoint multipath with a use of one as primary path and the others as alternate paths. Thus, alternate paths are not affected by the failure in any or all nodes or links on the primary path of the node-disjoint or the link-disjoint multipath, respectively. Those alternate paths expend significantly more energy than that on the primary path since they could potentially have much longer latency; moreover global topology knowledge is needed to facilitate the creation of the multiple disjoint paths. Using this multipath scheme in a network with n node-disjoint routes from source to destination can tolerate at least $n - 1$ intermediate network component failures. In braided multipath (Figure 2.1(c)), an alternative path is constructed for each node in the primary path that does not include this node, which means alternate paths in a braid partially overlay with the primary path. These alternate paths are not much more

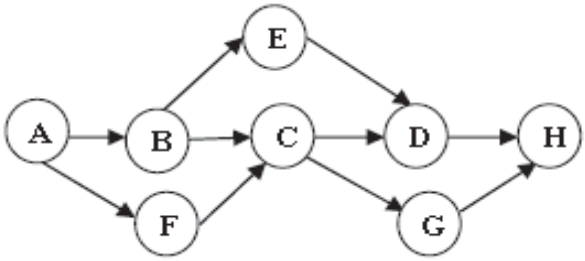
expensive than the primary path in terms of latency and overhead (alternative paths are shorter than in disjoint multipath). However, when all or most of the nodes on the primary path fail, new path discovery is required which introduces an additional overhead.



(a) Node-disjoint path



(b) Link-disjoint path



(c) Braided multipath

Figure 2.1: Multipath Routing, (a) Node-disjoint path, (b) Link-disjoint path and (c) Braided multipath

Based on the discussed issue in this section, we classify some existing researches on fault-tolerant multipath routing protocols in WSNs into two main mechanisms: retransmission and replication.

2.1.2.1 *Retransmission Mechanism*

The most popular mechanism is to retransmit the data packets to the sink on one of the multiple paths using minimum hop count or minimum energy consumption depending on the network requirement, for a predetermined number of times. The process is that the sink node transmits an acknowledgement back to the source when a data packet is received indicating successful transmission. If the acknowledgement is not received by the sender before a timeout, the data packet will be retransmitted. In WSNs the packet loss rate on the wireless link is higher than in other networks, thus the link level retransmissions is the most popular mechanism used. However, this method has some drawbacks in that it increases the network traffic requiring more resource consumption. Additionally, transmitting an acknowledgement message may increase delivery delay and more packet loss due to collisions. Furthermore, more memory space is needed in the sensor nodes to buffer the packet until it receives an acknowledgement from the destination.

In the following, we describe the routing protocols based on retransmission mechanism and highlight their key ideas.

Directed Diffusion protocol (DD) [16] is considered as one of the most popular routing protocols proposed for WSNs; many other routing protocols are either based on DD protocol or follow the similar concept. The basic idea of this protocol is that the sink broadcasts an Interest packet (Figure 2.2(a)) that is periodically refreshed along the

network. This packet is a query which contains the information requested by the sink. By receiving an Interest packet, all the nodes in the network will cache the packet in their memory, then flood it to their neighbors to ensure that all nodes received it. Each node generates a Gradient that includes the data rate and the direction in which the data is send (value and direction) (Figure 2.2(b)). When a node detects data it is compared with the information in its cache, if a match is found the node is considered to be source node and it periodically broadcasts a message at a low rate ensuring sensing a data. When the sink receives several detection events, which means multiple paths exist to the source, it broadcasts a Reinforcement message on one of these paths (usually the one with least delay) by increasing the data rate in the Interest packet, in another words to reinforce a path the sink resends the original Interest message but with a smaller interval (Figure 2.2(c)). When the reinforced path fails as shown in Figure 2.2(d), the sink will not detect any data. It reinitiates the Reinforcement message to use another path for rerouting the lost data. Therefore, in order to provide a fast recovery from path failure the sink must periodically broadcast the Reinforcement messages to quickly find an alternative path that will be constructed on demand in this case. Since this protocol is based on query driven data delivery, it cannot work efficiently for applications such as environment monitoring that require continuous data delivery to the sink. Moreover, this protocol cannot be considered as an energy efficient protocol because of its energy requirement to broadcast the periodic low rate messages. Sensors may also introduce extra overhead when matching data and queries.

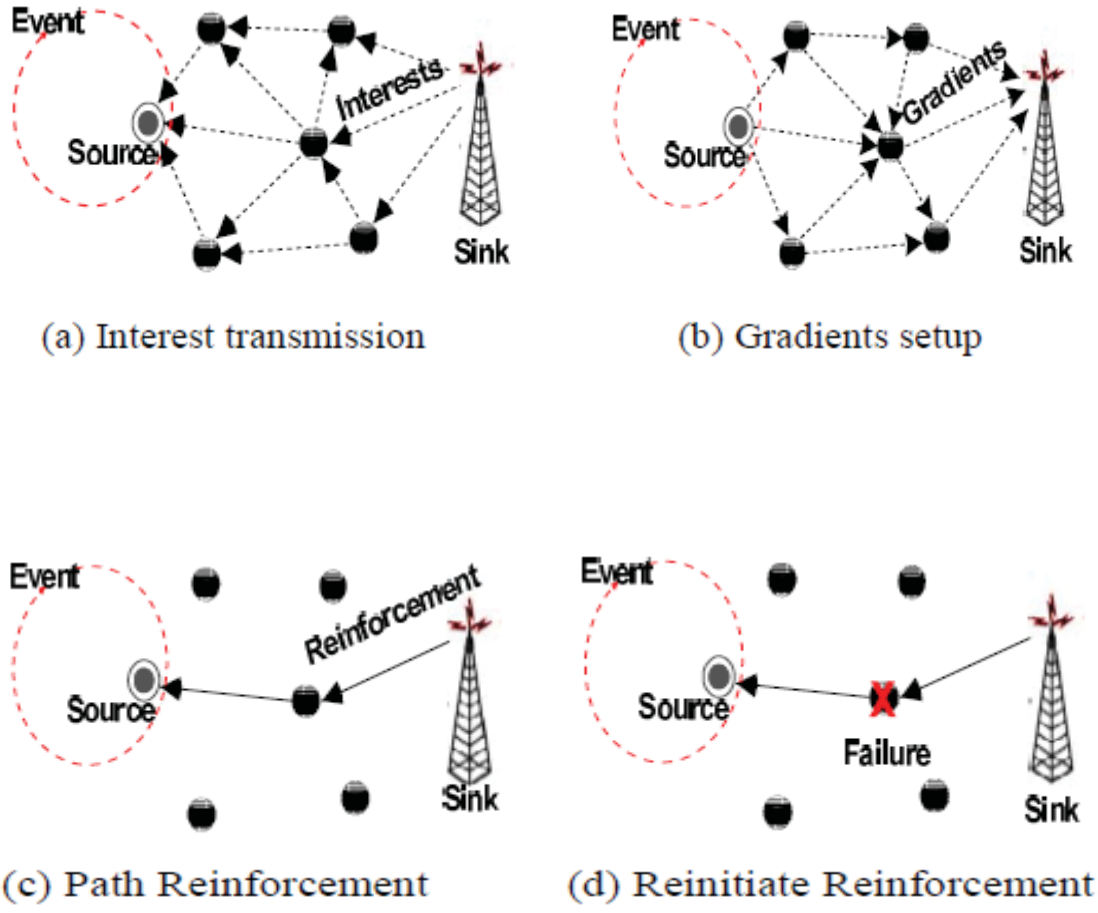


Figure 2.2: Directed Diffusion routing protocol

The Highly Resilient, Energy Efficient Multipath Routing Protocol for Wireless Sensor Networks [17] is based on the DD paradigm. The authors present a multipath routing scheme that finds several partially disjoint paths; these paths are not disjoint paths as in DD protocol, instead they are braided multipath to keep the cost low for maintaining the multipath and to quickly recover from path failure. The protocol also avoids the periodic flooding that is used in DD protocol. The network sets up multipath between the source node and the sink, one of these paths is used as the primary path to route the data packet, while the alternative paths are maintained by continuously sending a ”Keep-

alive" data through them. Thus, in case of primary path failure, nodes can recover rapidly by reinforcing another path to reroute the lost data packets. In this protocol, the energy consumption comes from the fact that all paths from source to sink are set in advance and maintained by periodically sending a low rate data "Keep- alive". It is shown that to be more resilient, the network costs more energy consumption.

Energy consumption is the main metric for Reliable Energy Aware Routing (REAR) in Wireless Sensor Networks [18]. The protocol proposes an energy reservation scheme to route the data to the sink, and to increase the network reliability a backup path for each primary path from the source to the sink is established. The key idea of this protocol is that when the sink receives an interest from a source node that is not in its routing table, the sink establishes two disjoint paths to the source where one of them will be used to deliver the data packet while the second path is used as a backup path when failure occurs. Additionally, part of the energy will be reserved for both paths in all the intermediate nodes along these paths. When a path failure occurs, the intermediate node sends the data packet back to the source node and an error report to the sink. As a result the failed path information is removed from the routing table in both the source and the sink. The reserved energy for that path is released from all the nodes along that path. Finally all the traffic is switched to the backup path. If the service path is set up again, then all the traffic is switched back on it.

This scheme respects the memory constraint of WSNs by using the backup path to eliminate the memory usage. Given that if the service path fails, with REAR the traffic on the primary path will directly be transferred to the backup path thus no caching is needed. Furthermore, REAR eliminates the energy consumption by reserving unequal amount of

energy for both paths, gets rid of the unnecessary packet retransmission, and when the network shows energy shortage, fewer control messages are used.

2.1.2.2 *Replication Mechanism*

Introducing redundancy into packet delivery [19] is another mechanism used to provide fault-tolerant routing protocols for WSNs.

Replication without Coding

One of the replication mechanism that routing protocols adopt to ensure delivery of the original packet to the sink is to transmit multiple copies of the same packet over different paths in order to recover from some path failures. The major drawbacks of this mechanism are the high overhead introduced when the packet is transmitted through each node till it reaches the sink, the maintenance of the path state in each of these nodes, and not being adaptive to channel errors. Erasure Coding (EC) is another replication mechanism used in multipath routing to provide fault tolerance and load balancing in WSNs.

Much research has been recently made to provide routing protocols that transmit multiple copies of the same packet over multipath to achieve higher reliability, including the work presented in the following.

The main idea in the protocol Reliable Information Forwarding (ReInForm) using Multiple Paths in Sensor Networks [20] is for the sink to periodically broadcast a routing update packet in the network such that each node knows its neighbors and the number of hops to the sink. When there is data to be sent, the source node generates a packet with DSP (dynamic packet state) fields in the header that contain the network condition (desired reliability, local channel error and hop distance to the sink). Depending on the

desired reliability identified by the source node, multiple copies of the data packet are created to be sent on multipath to the sink (the number of these multipath is therefore a function of the reliability). Each intermediate node uses the information in the DSP to forward the packet and makes a decision on the number of copies to split the packet to, which means the number of multipath to forward the packet. Moreover, the intermediate nodes decide which neighbors to forward the packet to (usually nodes that are closer to the sink are chosen, otherwise random nodes are chosen). This process continues until the data packet reaches the sink. ReInForm achieves fault tolerance by sending multiple copies of the same packet over randomly chosen paths to the sink. This duplication occurs not only at the source node but at every intermediate node in the network. Thus in this scheme a higher delivery ratio is reached since even if some data packet are lost the original packet can still be recovered from the other duplicated packets. The price to achieve reliability for this scheme is the high energy consumption that arises when the packet is split, transmitted and reconstructed at each node along the network. However, ReInForm needs no packet caching nor state maintenance inside the sensor nodes; thus it meets the memory constraints of WSNs. Also the overhead introduced using ReInForm is shown to be proportional to the desired reliability. The relationship between the reliability of a network and the overhead has been studied in [21]. Erasure code has been used in distributed systems, thus many papers exist in this field covering the load balancing and fault tolerance, but recently it has been used for WSNs to provide fault tolerance, increase network lifetime, and decrease energy consumption. Some of these papers that used EC to provide fault tolerance in addition to reaching other goals are summarized in this section.

Both the articles of Djukic and Valaee [22, 23] assumed a network system similar to TinyDB [24] and a distributed sink that collects its data from a number of receiver "prongs" that are connected with the sink through a reliable and high bandwidth links. When the sink asks for specific information it floods a query along the network. While the query travels in the network it records the path as well as the reliability and energy information on each hop. In [22] optimization procedure made to minimize the total energy consumption across each path in the network with a given bounds on the reliability and efficiency. The idea is that, to minimize the energy consumption in the network, the source node uses the packet loss and energy information carried by the query to distribute the data packet. However, simulation results show that in the sensor node the energy consumption can be decreased by increasing the parity fragments. However, this also decreases the efficiency. This is a cross-layer design since information from Data Link layer is used by the network layer to make a routing decision. While in [23], the goal of the optimization procedure was to maximize the network lifetime while increasing the fault tolerance. The probability of successful fragment transmission and the energy information that is carried by each query can be used to determine the lifetime on each path. Simulation results show that the network lifetime increases as the number of prongs increase. To increase the reliability, the sensors have to transmit more parity fragments, but that decreases the network lifetime.

Replication with Coding

Erasure coding has been used in distributed systems to achieve load balancing and fault tolerance, but recently [25] it has been used for WSNs as a replication mechanism in multipath routing to increase the data transmission reliability while decreasing energy

[22] consumption and increasing network lifetime [23]. The advantage of using data replication is to avoid the costly or impossible data retransmission in WSNs due to the severe resource constraints of sensor nodes. There are many types of EC and the most popular ones are; Reed-Solomon codes (RS) [26], Raptor codes [27] and Tornado codes [28]. RS code is the simplest and the widely used Forward Error Correction (FEC) codes for achieving reliable data transmission in networks.

Using RS codes the source node codes each data packet of size *Mb* bits it receives into *M* fragments each of size *b* bits [29], and generates another *K* parity fragments to have in total a set of *M* + *K* fragments as shown in Figure 2.3. If the sink receives any *M* fragments, it can recover the original data packet allowing at most *K* lost fragments. Denote the fragments allocation as $X = [x_1, x_2, \dots, x_n]$, where x_i is an integer and is the number of fragments allocated to $path_i$ and *n* is the number of node-disjoint paths from source node to sink. The allocation of fragments on each path is determined with a load balancing algorithm where $\sum_{i=1}^{n} x_i = M + K$. The value of *K* determines the loss recovery capability of the code. Given a fixed value of *M* + *K*, smaller *M* means less data information and more redundancy contained in each encoded block, thus the loss recovery capability is better. If z_i is a random variable that indicates the number of fragments received on $path_i$, then we have $\sum_{i=1}^{n} z_i \geq M$. Typically, the code rate is $\lambda = M/(M + K)$, the redundancy ratio is $r = K/(M + K)$, the maximum codeword length for a RS code is $c = 2^b - 1$ and the coding overhead is $h = K/M$.

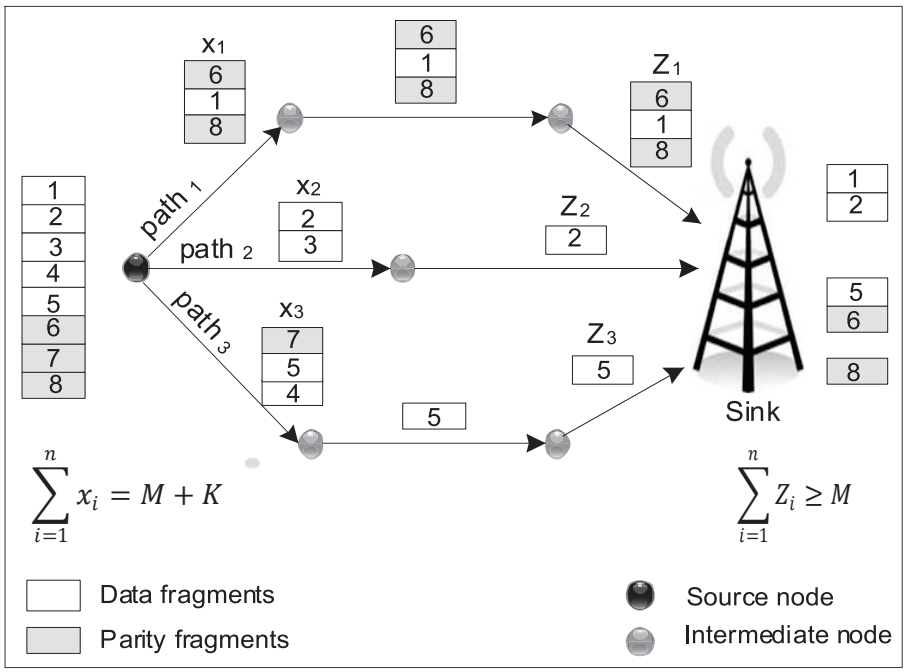


Figure 2.3: Example of data transmission using EC. Note that the data packet, $M = 5$ fragments, the added redundancy, $K = 3$ fragments and $n = 3$ paths.

The work presented in [30], employed EC with routing mechanism to minimize the computation and storage space in WSNs. The main idea is that the coding algorithm can be configured dynamically depending on the feedback information from the routing protocol. The idea is that, the index of the data packet and the fragments in which this packet is divided to at the source node using the EC algorithm, will be send to the sink on Auto Repeat request (ARQ) message. When the sink receives this message it is aware about which fragments are lost, which helps the sink identify the failed paths. Then, the sink will send ARQ-1 containing the failed path and the lost packets information along the healthy paths to the source node. By receiving ARQ-1, the source will update its routing table by marking the failed path and resending the lost packets. If the sink cannot

get the required data after sending ARQ-1 message, then the procedure will repeated by sending ARQ-2. From the number of received packet the sink can estimate the network status. If the packets loss rate is high, the sink sends Dynamic Adjust request (DAQ) message to the source to adjust the coding ratio, in this case to increase the redundant packets. But when the packet loss rate is low, the source reduces the number of redundant packets. This protocol provides a mechanism to avoid the failed path in next packet delivery at the price of increasing retransmission. The simulation results have shown that with path failing knowledge, lower loss rate is achieved. However, this scheme was evaluated using only disjoint multipath routing algorithms.

Authors in [31] used a network system and made assumptions similar to that presented in [22, 23]. However, this work proposed using RS erasure coding as a coding algorithm in the source node. The optimization on the total size of fragments and the number of fragments that can be transmitted on each path was calculated as a design parameter based on minimizing the total cost function. Simulation results showed that there was a slight effect of increasing the number of parity fragment K on network reliability and packet loss. However increasing the number K increases the total cost of packet transmission.

In [32] RS algorithm with Multipath on Demand Routing (MDR) is used to code/decode and route the data packet from the source node to the sink to increase the reliability of packet delivery in WSNs. The key idea of MDR is that, when the source has data to send to the sink, it starts the route request phase by flooding the network with a short message that contains the ID of both the source and the sink as well as the ID of the request. When the sink receives one of these messages it will return a route reply message

with added field representing the number of hops it has traveled so far. This message is returned to the neighbor from whom it received the route request message. Thus, each node that received the route reply message will increment the hop count and forward the message to the neighbor from which it got the route request. After a specific time, the source node will collect all the received route replay messages. It stores the neighbor ID from which it received the reply as well as the path length. Finally, source node will split the data packet according to, number of paths, length of paths and the maximum probability of failure. In this approach coding is done at the source node only as compared with ReInForm where coding the packet happens at each node. Moreover, the routing update packet that is broadcast periodically in ReInForm is eliminated in MDR approach since the source node broadcasts path request when it has data to send. Both these issues can affect the energy consumption in the network.

Table 2.1 introduces the classification and the performance metrics of some existing fault-tolerant research in multipath routing protocols for WSNs according to the retransmission and replication mechanisms.

Table 2.1: Taxonomy of existing fault-tolerant multipath routing protocols in wireless sensor networks.

Protocol	***Performance Metrics***			
Retransmission-Based	***Energy Consumption***	***Memory Usage***	***Recovery Time***	***Overhead***
[16]	High Periodic broadcasts of Interests	Cashing at each node	Time for sink to detect an absence of events and reinitiate Reinforcement	Due to flooding overhead
[17]	Less than [16], set up and alternate braided multipath in advance	Cashing at each node	Time for the sink to detect an absence of events then nodes can quickly reinforce an alternate path.	Due to flooding overhead
[18]	High path reservation process, energy reserved for two paths	Cashing at each node	Time for sending an error message to the sink and source to switch the traffic. Hop-by-hop ACK	Introduced to maintain paths and state in each node
Replication-Based				
[20]	High Packet copied at each node and periodic routing update packet	Low No cashing	None	Proportional to the desired reliability
[22], [23], [30], [31]	Low	Low No cashing	None	Low Erasure codes overhead
[32]	Low Packet coding/ decoding at source node only. Source broadcast path request when it has data	Cashing required	None	High Increased by control messages and erasure coding

2.2 QOS PARAMETERS AND REQUIREMENTS IN WIRELESS SENSOR NETWORKS

The term QoS is widely used in the area of all kinds of networks but still there is no agreement on its exact meaning. International Telecommunication Union (ITU) Recommendation E.800 (09/08) [33] has defined QoS as: ‘‘Totality of characteristics of a telecommunications service that bear on its ability to satisfy stated and implied needs of the user of the service’’. Basically, QoS is the ability of giving different priorities to various applications or packets based on their requirements by controlling the resource sharing.

To support QoS, the link characteristics such as delay, bandwidth, cost and loss rate should be available and manageable. However, obtaining and managing the link characteristics in WSNs is a challenging task because the characteristic of a wireless link change due to resource limitations and harsh environments. Therefore, routing protocols in WSNs must be adaptive to face frequent topology changes. Such frequent changes render the available state information out dated and this required to take into account the current conditions of the links while in the process of route selection. Researchers have proposed many metrics for QoS routing as a set of constraints which can be specified as a wireless link constraints or a path constraints. Link constraints specify the restriction on the use of links such as delay, while a path constraint specifies the end-to-end QoS requirement such as end-to-end transmission delay and reliability. Thus, routing algorithms are required to find specific routes for each application requirements, frequently given in terms of objectives.

2.2.1 Energy Consumption

Energy consumption is a very important QoS parameter in WSNs, due to the limited battery lifetime of sensors. Therefore prolonging the lifetime of the battery prolongs the lifetime of the sensor node. The operation of the battery depends on different factors such as the size of sensor, the unavailability of a power source, and the inaccessibility of the location that makes it difficult to handle sensor nodes once they are deployed.

Energy consumption in a sensor node occurs in three domains: sensing, data processing, and communications. There are many approaches for enhancing energy efficiency in WSNs and extend the network lifetime at different levels. Some approaches try to find out energy efficient routes through the available power in nodes, where load distribution is used to balance energy usage among sensor nodes by selecting a path with high energy nodes rather than the shortest path routing. Routing protocols based on load distribution may result in longer routes which in turn may not provide the lowest-energy route, but prevent overload at selected nodes, ensuring longer network lifetime. On the other hand, some other approaches try to minimize the energy consumption of sensor itself at its operating level [34], one of the most commonly used mechanisms is sleep scheduling [35] in which most sensor nodes are put into a sleep state for most of the time, and are only awakened periodically or on demand. Some mechanisms try to minimize the energy spent in the input/output operations at data transmission levels [36], and some aim to control the RF radio [37]; others target the formulation of sensor networks in terms of their topology and related routing mechanisms [38].

2.2.2 Reliability (Packet Loss)

The reliability of data transmission is an important facet of QoS in WSNs. Reliable data transport is to provide reliable transmission of data and to have the ability to detect and repair packet losses in the network. Existing work to achieve reliability are classified into two major schemes: retransmission and replication-based [25].

Reliability guaranteed data transmission and fault-tolerant routing have been the challenging areas in WSNs research. In WSNs multi-hop routing is used and therefore it is important to have a high reliability on each link in order to enhance the reliability of data transmission. Much work is being done to identify reliable links using metrics such as received signal strength, link quality and packet delivery ratio.

2.2.3 Packet Delay

WSNs have many critical QoS requirements, among which meeting end-to-end delay constraints is an important one for time-sensitive data. However, the end-to-end delay is difficult to be bound for event-driven sensor networks due to their unpredictable traffic pattern. WSNs applications that are capable of providing bounded delay guarantees on packet delivery are referred to as real time applications. Delay is the time elapsed from the departure of a data packet from the source node to the destination node. To achieve the goal of supporting real time applications in WSNs, many problems need to be solved.

In WSNs, a shared (wireless) medium is used for communication. Therefore, a distributed MAC protocol is needed to provide guaranteed bandwidth over multiple hops. The queuing delay is the major delay that influences data transmission in addition to the propagation delay, transmission delay, and the sleep delay. Nevertheless, the transmission delay is usually specific for the actual hardware and the MAC protocol used, thus is fairly

fixed for a specific deployment. In a duty cycle WSN, the sleep delay of each hop is equal to the toggling period. However, the queuing delay plays the major parameter in calculating the delay of data transmission. Queuing delay is constrained by the network capacity in which when the load of traffic in a network beats the network capacity, congestion will happen and this causes a long queuing delay, which contributes to increasing the end-to-end delay of data transmission.

2.3 QOS TRAFFIC SCHEDULING AT MAC LAYER IN WSN

Multiple applications running on WSNs require the network to handle traffic with different priority levels and QoS requirement in an energy efficient way while avoiding collisions and interference. However, in order to provide the required QoS in WSNs while considering the unique properties of sensor networks, energy awareness and robust protocol design at all layers of the networking protocol stack [39] is required. Although collective effort of all the communication protocol stack entities is essential for QoS provisioning, MAC layer possesses a particular importance among them since it is responsible for scheduling and allocation of the shared wireless channel and all other upper layer protocols are bound to that. Thus, MAC layer plays a key role for QoS provisioning and dominates the performance of the QoS support.

Congestion plays an important role in degrading the performance of WSN. Thus an issue of detecting and controlling congestion becomes essential to improve the performance of the network. Congestion in WSNs can happen due to node and/or link congestion [40]. Node congestion occurs when the packet inter arrival rate at a node is greater than the scheduling rate, this results in increasing queuing delay and packet loss

which requires retransmission of packets. On the other hand, link congestion occurs due to channel contention, interference, and packet collision.

To keep traffic levels at an acceptable value and to avoid congestion, a congestion control mechanism that considers the network capacity and the application requirements is required. Recently many researchers try to solve the congestion of WSN through a cross-layer approach using different parameters and different ways. However, they are similar in the basic idea that the information of routing and MAC layers should be combined to solve the local contention and the whole network congestion at the same time. The concept of cross-layer design is to exploit the interactions between layers and promotes adaptability at various layers based on information exchanged. At this point, adaptation refers to the ability of network protocols and applications to observe and respond to changes in network conditions. Some of the congestion control mechanisms used in WSN [40] are summarized as follows.

- Local cross-layer congestion control. This method is based on buffer occupancy. Traffic is classified into two types, the generated traffic and the transit traffic, and placed into two different buffers. The key idea of this method is to control the rate of generated traffic and to regulate the congestion in transit traffic based on the current load on nodes.
- Adaptive duty cycle-based congestion control. In this method a combined mechanism of resource control scheme and traffic control scheme is used. The key idea of this method is to adjust the duty cycle of a node when the congestion degree is below a certain threshold. However, when the congestion degree is above the threshold, the node informs the neighbor nodes to adjust their transmission rates.

- Priority based congestion control. This method introduces the concept of priority index. Each sensor node or traffic type is given a priority index. The key idea is that the node or traffic with higher priority index gets more bandwidth in order to guarantee flexible weighted fairness and efficient congestion control.
- Buffer based congestion avoidance. This method is based on buffer management. The key idea is that when the buffer at node is near to be full, it forces the neighbor nodes to slow down their forwarding rates. This process is adapted by all the sensor nodes in the network in order to achieve the maximum congestion free transmission.

2.4 SECURE ROUTING IN WSN

Secure multipath routing protocols in WSNs can be divided into three categories based on the security related operational objective [41]: the multipath routing protection only, the attack-specific, and the security operations support. The security-based multipath routing protection protocol is the interest of this thesis in which the multipath routing is used to improve the security, increase reliability of data transmission, provide load balancing and decrease the end-to-end delay.

WSNs have general security requirements similar to other traditional networks, such as confidentiality, authenticity, integrity, freshness, resilience and availability of service [41]. Confidentiality is to ensure that sensitive information is protected and not exposed. However, when using multipath routing with one of the approaches presented in Section 2.4.2, the probability of eavesdropping attacks can be reduced since the attacker needs to catch the appropriate fragments for each packet over different paths, in order to

reconstruct the original packet. Authenticity is to verify the identity of the nodes participant in a communication in order to ensure that a trusted node and not a malicious node has sent the packet. In single path routing, if authentication cannot be established due to malicious node, then the path cannot be used to route packets from the source to sink. Therefore, a new path discovery phase must be established. However, in multipath routing, if the authentication fails in a specific path, alternative paths perform authentication between other nodes and the communication is achieved. Integrity and freshness refer to verifying that the packets are accurate and are up-to-date. In single path routing, neighbors' nodes in the path verify the integrity of packets and if a modification has been detected, the node may drop the packet and inform the neighbor node to resend the packet. In this case the delay of the transmission is increased and can affect the network performance. Therefore, to support the integrity and freshness of the packets, multipath routing is used. The use of alternative paths allows the data packet to reach the sink even when some of the paths may be compromised and/or packets may be modified. Resilience and availability of service means that the network has to provide a reliable service and ensure that the information can be obtained when required without interruption when nodes are comprised or failed. In single path routing, packets are sent over one route to the sink and an attacker can break the communication by compromising one or more nodes along the used route. However, in multipath routing, the effect of security attacks that target the availability, reliability and resilience of the network can be reduced. By transmitting data redundantly through multiple paths even if some of the paths are compromised or failed, the communication is uninterrupted, resilient, and the probability that packet can reach the sink is higher compared to single path routing.

2.4.1 Network Layer Attacks in WSNs

The network layer of WSNs is vulnerable to the different types of attacks. The attacks that act on the network layer are called routing attacks. In general routing attacks are classified into two major categories, namely passive attacks and active attacks. In a passive attack, the attacker spies on data exchange in the network without changing it. Therefore, a passive attack does not affect the normal operation of the network; accordingly detection of such an attack is very difficult. One of the possible solutions to this problem is to use a powerful encryption mechanism to secure data transmission in order to reduce the possibility of an attacker receiving useful information from the data overhead. In an active attack, the attacker monitors, listens to and modifies data exchange in the network. Active attacks can be internal from compromised nodes that are part of the network or external from attackers outside the network. However, routing attacks are considered active in nature [42].

Some of the routing attacks [42] in WSNs are summarized briefly in the following:

- **Spoofed, altered and replayed routing information**. In a multi-hop network like WSNs, every node acts as routes. Therefore, an attacker may interrupt routing information through creation of routing loops, producing false error messages, attracting or repelling network traffic from selected nodes, extending or shortening routes and increasing end-to-end latency.
- **Selective forwarding**. If a node is located near the source or the sink, an attacker may compromise this node by including itself on a data flow to launch a selective forward.

- **Sinkhole attack**: An attacker attracts traffic to a specific node by making this node look more attractive to its neighbors using false routing information resulting in selecting this node as the next hop node to route data. Thus, all traffic from this compromised area in the network would flow through this node.
- **Sybil attack**: In the Sybil attack, a node duplicates itself and presents in more than one locations. An attacker can take the identity of multiple nodes to produce multiple paths routing through a single compromised node. Therefore, the Sybil attack targets fault-tolerant schemes such as multipath routing and topology maintenance. However, using authentication and encryption techniques can prevent Sybil attack on the sensor network.
- **Wormhole**: An attacker records packets at one location, tunnels them to another location and then retransmits them into the network.
- **Hello flood attack**: An attacker sends or replays HELLO messages with high-powered transmitter energy to make other nodes believe that it is within their transmission range. However, these nodes are out of the transmission range of the attacker, the attacker falsely appears as shorter route to the sink causing other nodes to transmit to the attacker.

2.4.2 Security Approaches in Multipath WSNs

In single path routing, when a sensor node is compromised all the data on this node including cryptographic keys is compromised which risks the whole path information. Therefore, multipath routing is used to avoid this problem by increasing the confidentiality and robustness of data transmission since when some paths are compromised or failed, data can be recovered from the other reliable paths. Also, in order

to prevent eavesdropping on the transmitted data, multipath routing with coding techniques is used to code data at the source node before transmitting it to the sink. Secret Sharing (SS) scheme and EC technique are the most popular coding techniques used to support secure and reliable data transmission in WSNs.

2.4.2.1 *Secret Sharing Scheme*

In cryptography, secret sharing refers to the method of distributing a secret among parties, each of which allocates a share of the secret [43, 44]. The secret can only be reconstructed when specific numbers of the shares are combined together. As shown in Figure 2.4, a secured message *S* is distributed using SS scheme into *m* pieces called shares and transmitted to the destination over different paths. *S* can be decrypted from any *k* out of *m* shares while no information about *S* can be obtained with *k-1* or less shares. The main drawback of using the SS scheme is the large amount of traffic and the redundancy involved.

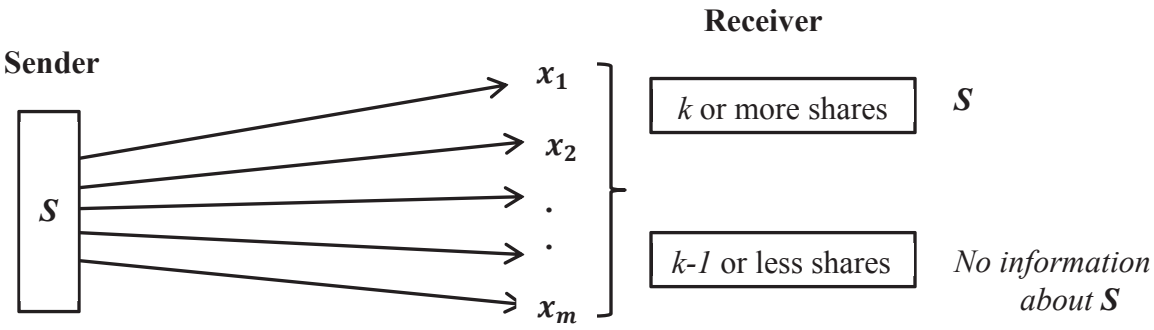


Figure 2.4: Secret sharing scheme

2.4.2.2 *Erasure Coding Technique*

A common approach to enhance data transmission security in WSNs is to use the EC technique (Section 2.1.2) as a replication mechanism in multipath routing to increase data transmission reliability and to support data confidentiality while decreasing the energy consumption. Using EC to enhance the security of data transmission in WSNs is the main emphasis of Chapter 6 and the details of the proposed solution are presented in Section 6.2.4.

2.5 SUMMARY

This chapter is divided into four main sections. The first section presents the primary motivations behind using multipath routing approach in WSNs to achieve load balancing, increase reliability and to provide fault tolerance. We survey and classify some existing researches on fault-tolerant multipath routing protocols in WSNs into two main mechanisms, retransmission and replication. The replication mechanism is further classified to replication without coding and replication with coding mechanism. The second section pertains to quality of service issues of sensor networks with a review of some mechanisms used to provide the required QoS in WSN. The third section highlights the important of using the cross-layer design in order to enhance the network performance. The congestion control and avoidance mechanisms are addressed and reviewed. Finally, the fourth section presents a discussion focusing particularly on the security issue of routing protocols in WSNs. Network layer attacks in WSNs are addressed and the mechanisms used to secure the multipath routing protocols are reviewed.

Chapter 3

LITERATURE REVIEW OF ROUTING PROTOCOLS IN WSN

With the growth in potential use of WSNs, considerable research efforts have been done in QoS routing and covered in comprehensive survey articles presented in [12, 13]. The work in [29] presented routing challenges and design issues in WSNs. They classified all the existing routing strategies based on the network structure and protocol operation. In [25] a brief overview on the existing fault-tolerant routing protocols in WSNs is provided and categorized these protocols into retransmission-based and replication-based protocols. Multipath routing protocols for wireless sensor networks are provided as a survey in [45]. They classify and investigate the operation as well as benefits and drawbacks of the existing multipath routing protocols in sensor networks.

As we will cover more than one topic in this thesis, our literature review is divided into three parts. In the first part, we discuss the state of art QoS routing protocols in WSNs. Second part reviews the traffic scheduling schemes used in QoS routing and in the third part, we review the secure mechanisms in multipath routing protocol.

3.1 QOS-AWARE ROUTING PROTOCOLS

Existing and potential applications of WSNs span a wide range including real time target tracking, homeland security, battlefield surveillance and biological or chemical

attack detection. For example, in many applications, to extend the network lifetime is considered more important than the quality of data and this is related to reducing the energy dissipation in the sensor nodes. Thus, a network requires an energy-aware routing protocol. In real time applications, data should be delivered in time or otherwise data is considered useless. In this case, the network requires a timeliness-aware routing protocol. However, in other applications, a reliable routing protocol is used since the reliability of data transmission in the network is considered as an important issue. Furthermore, the requirement of real time, energy efficient and fault-tolerant communication is extremely important in emerging applications.

One of the first QoS-based routing protocols in WSN is the Sequential Assignment Routing (SAR) [46]. To achieve both energy efficiency and fault tolerance, SAR builds multiple paths from sink to sensors by creating multiple trees where the root of each tree is a one-hop neighbor from sink by taking into account the energy resource on each path and the priority level of the data packet as the QoS metric. SAR provides failure recovery by enforcing routing table consistency between upstream and downstream node on each path. Although SAR provides fault tolerance and recovery, it suffers from the overhead of maintaining routing tables and states at each sensor node specially when the number of sensor nodes deployed is large. The work in [47] avoided the overhead problem presented in SAR by selecting a path from a list of candidate paths that meets the end-to-end delay requirement and enhances the throughput for best effort traffic. However, the protocol is not scalable since global knowledge of the network topology is required by each node.

Position-based routing or geographic routing [48] uses greedy forwarding mechanism for packet delivery in multi-hop wireless networks. Direct neighbor nodes

exchange location information and locally select the neighbor that is closest to the destination. In this case, a sensor node needs to know only the location information of its one-hop neighbors without the knowledge of the entire network; thus discovery floods and state propagation are not required beyond a single hop. However, using greedy forwarding strategy in sensor network, may lead to the dead end problem [49], which occurs when a message is forwarded to a node with no neighbor that is closer to the sink than the node that currently has the packet, and is called local optimum. Avoiding this problem is a challenging issue for any geographic greedy forwarding approach. Although a dense deployment of wireless nodes can reduce the incidence of this problem in the network, but it is still possible for some nodes to experience a local optimum. The greedy forwarding mechanism presented in [48] is modified according to the reliability of links in [50].

The work in SPEED [51] proposed a location-based real time routing protocol for soft end-to-end deadline guarantee to maintain a desired delivery speed in the network. SPEED uses only one delay threshold overall to manage transmission of data packets at the highest transmission velocity. Therefore, it cannot satisfy different requirements for transmission delay. Also, energy metric has not been considered in the design of SPEED protocol, nodes with high transmission velocity are selected without considering the remaining energy of nodes. Therefore, SPEED protocol is not energy efficient.

Another routing protocol which addresses both energy efficiency and QoS is the LQER protocol presented in [52]. LQER protocol makes path selection based on historical states of link quality after minimum hop field is established. The link quality estimation strategy results in reliability as well as energy efficiency.

A multi-objective routing algorithm for resource constrained WSNs was proposed in [53] that calculates the cost of each possible path between the source node and the sink after the application assigns weight for each requirement in order to achieve multi-objective of existing routing protocols.

A multi-constrained QoS multi-path routing (MCMP) protocol is proposed in [54]; the protocol uses braided routes to deliver packets to the sink to enhance network performance with reasonable energy cost and achieves the required QoS in terms of reliability and delay. The end-to-end delay is expressed as an optimization problem and solved by an algorithm based on linear integer programming. However, routing data over the minimum hop count path to satisfy the required QoS leads in some cases to more energy consumption. ECMP protocol [55] is proposed as an extension to MCMP which considers QoS routing problem as a path-based energy minimization problem constrained by reliability, delay, and geo-spatial energy consumption.

Furthermore, applying redundancy to satisfy some QoS requirements in WSNs drains considerable research efforts and some are covered in a survey article presented in [25]. In [56], FEC technique is used to provide fault recovery, balance the energy consumption over sensor nodes and to increase the reliability of data transmission.

Most of the aforementioned routing protocols characterize the network with a single metric such as hop count, delay or minimum energy consumption algorithms to compute paths using single path or multipath routing but not both. Moreover, modeling the network as path-based and link-based multiple metrics such as energy, delay and reliability of data transmission to meet diverse and multiple application requirements by

considering the changing conditions of the network, limiting node resources as well as using the advantages that the sink node has unlimited resources, were not considered.

3.2 QOS TRAFFIC SCHEDULING

Sensors deployed in WSN are energy limited devices and therefore energy efficient communication techniques are the most important requirements in these networks. Cross-layer design with routing and MAC as two important candidate layers has been proposed as a solution for resource constrained WSNs and many researches have been conducted on this perspective [47, 56-61].

Congestion can degrade the network performance and obstruct the application requirements. It can cause packet losses, increased delay, and increased energy consumption. For example, a node may have many packets backlogged due to heavy load, and if it is chosen to forward other packets, it increases the packet latency and may even drop packets due to queue overflow, which in turn reduces the higher layer throughput. Accordingly, the timeliness problem in WSNs is studied from the congestion point of view. Therefore, many solutions have been proposed in the literature to control the congestion in WSN such as rate control, queue management, and traffic prioritization.

A cluster-based QoS-aware routing protocol for WSNs is proposed in [47]. The protocol finds the least cost and energy efficient path that meets the end-to-end delay. A cost function is associated with each link considering the link delay and a class based queuing model is employed to handle both real time and non-real time traffic. The bandwidth is shared for real time and non-real time traffic and is adjusted in order to

satisfy the delay requirements. However, packet collision or loss is not considered in the design of this protocol.

In [56] a node-disjoint multipath routing protocol is proposed to provide reliability and delay requirements of real time applications. The energy, remaining buffer size and signal-to-noise ratio are used as parameters in the link cost function to select the next hop through the paths construction phase. To improve reliability, FEC mechanism is used to introduce data redundancy for data transmission. To achieve the delay requirements of various applications, a queuing model is adopted to manage the real time and non-real time traffic.

A real time communication protocol for large-scale WSNs is presented in [57]. A velocity monotonic scheduling is introduced that inherently accounts for both time and distance constraints in order to reduce the end-to-end deadline miss ratio in sensor network. The velocity of a packet is calculated based on the end-to-end deadlines and the communication distance and assigned priority accordingly. However, the main drawback of this protocol is that in the next hop selection process, only greedy geographic forwarding is considered while the conditions of the local wireless links are not considered. Therefore, load balancing and congestion avoidance in packet transmission are not achieved.

In [58] each node uses its own and its neighbor's state information to adapt its routing and MAC layer behavior by employing a flexible cost function at the routing layer and adaptive duty cycles at the MAC layer that relies on local decisions to equalize the energy consumption of all nodes. In this way the routes can be maintained easily and

little overhead is added. However, decisions are made locally without considering the entire path from the source to the destination.

The QoS-based energy-efficient routing protocol (QuESt) [59] builds a set of non-dominated paths that satisfy the application-specific QoS requirements based on using multi-objective genetic algorithm (MOGA). The protocol optimizes multiple QoS parameters such as end-to-end delay, bandwidth requirements and energy by using the MOGA algorithm as a tool to solve the multiple QoS requirements independently without combining them into a single objective function. A network of a single sink and multiple sources is used, and the available paths between the sources and the sink are created using depth first search (DFS). These available paths are served to the MOGA algorithm to give a status (fitness value) for the QoS parameters on each path. Therefore, the protocol selects the suitable path for each type of traffic based on the QoS status.

A node priority based control mechanism for wireless sensor networks is proposed in [60]. Node priority index is presented to reflect the importance of each node. Packets inter arrival time and service time are used to measure the congestion degree at a node. Moreover, a hop-by-hop congestion control is used in order to control congestion faster and in an energy efficient way. However, the protocol does not consider the sensed data within a node. Moreover, it does not consider any mechanism to handle prioritized heterogeneous traffic in the network.

An extension of SPEED is the MMSPEED [61] protocol, which is proposed to provide QoS differentiation in timeliness and reliability domains based on a cross-layer approach between the network and the MAC layers in WSNs. To support timeliness, multiple network packet delivery speed options are provided for different traffic

according to their end-to-end deadlines, and to support reliability, multipath is used to control the number of delivery paths based on the required end-to-end reaching probability. However, in order to avoid congestion and decrease the packet loss rate, packets are transmitted with respect to the required end-to-end delay parameter. By using the distance to the sink and delay information, each node calculates the required speed and selects the next hop such that the speed requirement is met. And to support reliability, multipath is used and the number of these paths is based on the required end-to-end reaching probability. Although, MMSPEED does some improvements over SPEED and differentiates among different real time levels, it also does not dynamically adjust routing paths according to the available energy at the nodes.

3.3 SECURITY IN MULTIPATH ROUTING PROTOCOLS

In the literature, encryption techniques have been used for secure multipath routing protocols in WSNs. In [62], an extensive survey has been conducted on the current state-of-the-art for secure multipath routing protocols. The security related issues, threats, and attacks in WSNs and some of the solutions can be found in [63].

H-SPREAD [64] protocol is proposed as an extended version of SPREAD protocol [65] which used multipath between a single source-destination pair to deliver multiple secret message shares in order to enhance the data confidentiality in mobile ad hoc networks. H-SPREAD proposed for WSNs a distributed many-to-one multipath discovery protocol by employing two phases of flooding in order to enhance the security and reliability of data transmission. To enhance reliability, H-SPREAD uses an active per-hop packet salvaging strategy, the sender forwards the packet over another path

instead of dropping it when unsuccessful transmission occurs to increase the probability that the data packet is delivered to the sink. Although, H-SPREAD protocol provides security in terms of resilience against node capture, it does not provide any authentication mechanism. Thus, many network layer attacks such as Sinkhole or Wormhole on routing protocols that attract traffic by advertising high quality route to the sink are related with the goal of affecting the construction of paths. Furthermore, the construction of the spanning tree used in this protocol introduces high overhead.

Other possible solutions to support security and reliability of data transmission is the combination of data encryption and FEC technique [66, 67]. The main concept of this combination is to encrypt the original data message, encode the encrypted message using FEC coding, and then route it to the destination. A secure, multi-version, multipath protocol, MVMP, is proposed in [67] to offer a secure and reliable data communication in WSNs. MVMP consists of four steps: divide the original data message into groups, encrypt each group using different cryptographic algorithms, code the encrypted packets using RS codes, and transmit the coded packets on multiple disjoint paths that are assumed to be established before the data transmission. The data packet can be compromised when certain amount of codewords over different paths are intercepted and all the encryption algorithms used for the transmission is known. Moreover, to reconstruct the original message, the attacker needs to make all possible packet combinations, which is a resource challenging task. Although, MVMP protocol uses different cryptographic algorithms in order to enhance data transmission security, this strategy could be expensive in resource constrained environments such as WSNs.

In [68], a secure and reliable node-disjoint multipath routing protocol is proposed in order to minimize the worst case security risk and to maximize the packet delivery ratio under attacks. The multipath routing problem is modeled as an optimization problem and solved by a heuristic algorithm using game theory and a routing solution is derived to achieve a trade-off between route security and delivery ratio in worst scenarios. The protocol focuses on the worst case attack scenarios to achieve the design objective of providing the best security and/or delivery ratio. Although, the protocol assumes using link reliability history in the computations, in WSN the sensors and the communication links change frequently and are time varying. This required a frequent update of the computation of paths to discover the most reliable and secure paths. Also, the protocol assumes that each node has a full knowledge of the whole network topology which is considered an expensive assumption in WSN.

An intrusion-fault tolerant routing scheme proposed in [69] offers a high level of reliability by a secure multipath routing construction topology and uses one way hash chains to secure the construction of a multipath, many-to-one dissemination topology.

A secure and energy-efficient multipath routing protocol for wireless sensor networks is proposed in [70]. Disjoint and braided paths are constructed using a modification of the Breadth First Search algorithm. The sink executes the paths discovery, selection and maintenance in a centralized way. Authors claim that network layer attacks such as Sinkhole and Wormhole are not related since routing paths are selected by the sink node and periodically changed to prolong the lifetime of the network. Also, the protocol addresses the replayed attack by having each packet identified by a unique sequence number to be transmitted only once. However, the protocol does not use

any encryption and authentication mechanism to protect against a number of attacks; this means that an attacker can affect the paths construction process. Moreover, the sink needs to have information of the whole network topology which requires that each node sends its neighbors list to the sink, and this process consumes huge energy and introduces extra overhead.

Enhancing data security in ad hoc networks based on multipath routing is proposed in [71], which is designed on the multipath routing characteristics of ad hoc networks and uses a route selection based on the security costs without modifying the lower layer protocols. The authors claim that the proposed protocol can be combined with solutions which consider security aspects other than confidentiality to improve significantly the efficiency of security systems in ad hoc networks. The protocol in [71] is designed for an ad hoc network where the number of nodes in the network is considerably low and the capability of node is usually better than that of sensor networks. Thus, the protocol cannot directly fit the properties of sensor networks.

3.4 SUMMARY

This chapter has provided an overview of related work in QoS, traffic scheduling and security issues of WSNs and contrasted it with the contributions of this dissertation. It is divided into three sections. Section 3.1 reviews the existing research on QoS-based routing protocols in WSN. Section 3.2 reviews the works that considered the cross-layer approach between the network and the MAC layers in order to provide traffic scheduling mechanism in WSNs. Section 3.3 reviews WSNs multipath routing protocols that provides both security and QoS.

Chapter 4

QOS-AWARE MULTIPATH ROUTING PROTOCOL

In this chapter, the problem of providing QoS routing is formulated as a link and path-based metrics. We present a novel heuristic neighbor selection mechanism in WSNs that uses the geographic routing mechanism combined with the QoS requirements to provide multi-objective QoS routing (MQoSR) for different application requirements. In link-based metrics, the protocol considers the neighbor with the best trade-off between required QoS and proximity; link with the least possible cost is considered based on a cost function defined for each link as a function of distance to sink and the link state in terms of available energy at a node, delay and reliability as well as the application requirements such as end-to-end delay, reliability and energy consumption. In the path-based metrics, the end-to-end delay, reliability of data transmission and network lifetime are considered in selecting the routing paths. Therefore, the next hop selection as well as the routing paths and the number of these paths are dynamically adjusted according to the available parameters at the nodes and the QoS requirements.

4.1 NETWORK MODEL FOR QOS PROVISION

4.1.1 Network Model and Assumptions

We model a WSN with N nodes and one sink as an undirected graph, $G = (S, L, Q)$ in the plane (Figure 4.1), where S denotes the set of vertices that represent the communication sensor nodes, L denotes the set of edges representing links between

nodes, and Q is a nonnegative QoS capacity vector of each edge. The distance of a direct link $l(s_x, s_y) \in L$ between nodes s_x and s_y is d_{s_x, s_y}. A path is defined as a sequence of nodes from the source node to the sink and $P = \{path_1, path_2, ..., path_n\}$ is the set of n available node-disjoint paths between the source node and sink. We assume sensors are homogeneous, each sensor has same transmission radius, a, and they consume equal energy to transmit a bit of data. Furthermore, we assume that the sensor nodes are stationary and at any time, each sensor node is able to compute its available energy level, E_{ava}, as well as record the link performance between itself and its neighbor nodes in terms of delay D_{link}, and reliability R_{link}, where R_{link} is expressed in terms of signal-to-noise ratio (SNR) [60]. Additionally, each node is assumed to know its exact position, the position of nodes within its range of communication, neighbor nodes, and of the sink using localization techniques.

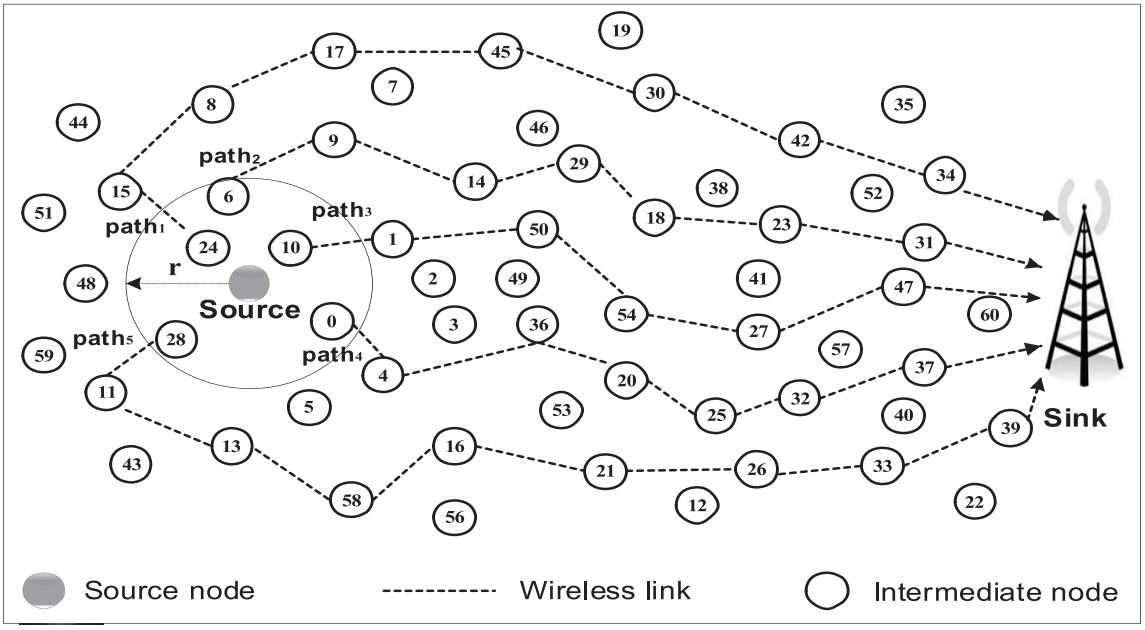


Figure 4.1: Wireless sensor network model

4.1.2 QoS Provisioning

In WSNs, the most important parameters that hinder the goal of guaranteed event perception are time-sensitive and reliable delivery of data transmission, while a minimum energy consumption is desired. In this chapter, a detailed analytical analysis of energy consumption, delay and reliability is presented.

4.1.2.1 *Energy Consumption*

The energy consumed for data transmission, E_{path_j}, on a single path, j, can be written as,

$$E_{path_p} = \sum_{\delta=1}^{hop_j} E_{con_s} \tag{4.1}$$

where hop_j is the hop count of path j and E_{con_s} is the energy consumption at node s to transmit a packet of b bits and is expressed by $E_{con} = e_{tx} + e_{rx}$, where e_{tx} and e_{rx} are the energy consumption to transmit and receive the packet for distance a, respectively.

When sensors have a fixed communication radius, a, nodes located randomly at any distance within the area of πa^2 always have the same power consumption for transmission. Then, $e_{tx} = \left(e_t + \varepsilon_{amp} \times a^2\right) \times b$ and $e_{rx} = (e_r \times b)$, e_t and e_r are the energy consumption to transmit and receive one bit of data, respectively and ε_{amp} is the energy consumption of power amplifier. To transmit b bits of data packet, the available energy at node, E_{ava}, must be larger or equal to the minimum energy threshold required to transmit this packet, E_{thr}.

In multipath routing, the total energy consumption, E_{e2e}, to transmit any data packet is measured as the summation of the energy consumption on all the used paths and is given as,

$$E_{e2e} = \sum_{j=1}^{np} E_{path_j} \tag{4.2}$$

where *np* is the number of multipath used to route the data packet.

We can then present the energy objective function f_E that minimizes the total energy consumption on all used paths as, f_E*: Minimize the energy consumption,* E_{e2e}.

4.1.2.2 *Delay Metric*

Delay is the time elapsed from the departure of a data packet from the source node to the destination node. The delay metric on link l is represented as D_{link} and is the sum of processing, queuing, transmission, and propagation delay. Many of the developing WSNs involve delay sensitive applications with real time delay constraints; meeting such delay constraints require deploying an efficient routing technique that reduces delay and ensures on-time packet delivery.

The delay of a path, D_{path_j}, is the sum of the delays at all the intermediate nodes along the path.

$$D_{path_j} = \sum_{l=1}^{hop_j} D_{link_l} \tag{4.3}$$

Therefore, the end-to-end delay to transmit the data packet to the sink along the selected path or paths is given as:

$$D_{e2e} = max_{1 \le j \le np}\ D_{path_j} \tag{4.4}$$

The delay objective function, f_D, is to ensure that the end-to-end delay on the selected paths is the minimum and/or $D_{e2e} \leq D_{req}$.

where D_{req} is an application-specific parameter which reflects the required end-to-end delay for data delivery.

4.1.2.3 *Reliability Metric*

The transmission reliability is an important index of QoS, calculated to measure the probability of transmission failures and can be expressed in terms of data delivery ratio. If all source nodes send total packets of Pkt_{source}, and the number of packets received by the sink is Pkt_{sink} then the data delivery ratio, denoted as *DDR*, can be written as:

$$DDR = \frac{number\ of\ packets\ received\ at\ the\ sink}{number\ of\ packet\ generated\ by\ source\ nodes} = \frac{Pkt_{sink}}{Pkt_{source}} \tag{4.5}$$

The probability of successful data transmission on path p, R_{path_p}, can be calculated using the following,

$$R_{path_j} = \prod_{l=1}^{hop_j} R_{link_l} \tag{4.6}$$

where R_{link_l} is the link l reliability.

The end-to-end data transmission reliability, R_{e2e}, is given by (4.7) and it is related to the number of used paths.

$$R_{e2e} = 1 - \prod_{j=1}^{np}(1 - R_{path_j}) \tag{4.7}$$

If R_{req} is an application-specific parameter which reflects the required end-to-end reliability for data delivery, then the WSN is considered reliable only if the data reliability satisfies $R_{e2e} \geq R_{req}$. The reliability objective function, f_R, is to maximize the

data transmission reliability, R_{e2e}, such that $R_{e2e} \geq R_{req}$ which is equivalent to minimize ($-R_{e2e}$).

To achieve the required reliability by an application, we use the reliability gained using erasure coding, as described in Section 2.4.1. The source node code each data it receives into *M* data fragments each of size *b*, and generates another *K* coding fragments to have in total a set of *M* + *K* fragments. This set of fragments is then transmitted as sub-packets over *np* selected paths to the sink, such that $\sum_{j=1}^{np} x_j = M + K$ where x_j is an integer representing the number of fragments allocated to $path_j$. To reconstruct the original packet at least *M* fragments should be received by the sink, allowing at most *K* lost fragments and the coding rate is thus defined as *M / (M + K)*. The probability of packets successfully received by the sink, P_{succ}, to achieve the requested reliability, R_{req}, has a binomial distribution that depends on R_{path}, and can be written as:

$$P_{succ}[\, y \geq M] = \sum_{y=M}^{M+K} \binom{M+K}{y} [(R_{e2e})^y][1 - R_{e2e}]^{(M+K-y)} \quad (4.8)$$

4.1.3 Required QoS Model

We model the QoS required by an application into seven different classes as shown in Table 4.1. In all these classes the cost function is used to calculate each link cost after assigning the weighting factors C_E, C_D and C_R that are related to each other by the formula $C_E + C_D + C_R = 1$. The three digit *QoS* field in Table 4.1 represents the requirement in terms of energy, delay and reliability, respectively.

Table 4.1: QoS classes model

Class	*QoS*	*Application Requirements*	*W*			*Goals*
			C_E	C_D	C_R	
1	100	Energy	1	0	0	Increase network lifetime, min E_{e2e}
2	010	Delay	0	1	0	$D_{e2e} \leq D_{req}$
3	001	Reliability	0	0	1	$R_{e2e} \geq R_{req}$
4	101	Energy and Reliability	0.5	0	0.5	min E_{e2e} and $R_{e2e} \geq R_{req}$
5	110	Energy and Delay	0.5	0.5	0	min E_{e2e} and $D_{e2e} \leq D_{req}$
6	011	Delay and Reliability	0	0.5	0.5	$D_{e2e} \leq D_{req}$ and $R_{e2e} \geq R_{req}$
7	111	Energy, Delay and Reliability	0.333	0.333	0.333	min E_{e2e}, $D_{e2e} \leq D_{req}$ and $R_{e2e} \geq R_{req}$

We can then write the required QoS function, C_{req}, as follows:

$$C_{req} = \frac{E_{thr}}{E_{ava}} \times C_E + \frac{D_{link}}{D_{req}} \times C_D + \frac{R_{req}}{R_{link}} \times C_R \tag{4.9}$$

4.2 LINK METRICS AND NEXT NODE SELECTION

The link cost function is used by nodes to select the next hop during the path discovery phase. The link cost function in this protocol is a function that takes into account the requested QoS and the information of neighbors. Modifications are made to the route request (RREQ) and route reply (RREP) messages to enable the discovery of node-disjoint multipath.

4.2.1 Initialization Phase

During this phase, each sensor node is assumed to update the local states of its one-hop neighbors by broadcasting a HELLO message (Figure 4.2) in which the links conditions are reported. Each node then maintains and updates its neighboring table information to record the link performance between itself and its direct neighbor nodes in terms of R_{link}, D_{link}, and E_{ava}. Each sensor node knows the distance to its neighbors and to the sink node.

Sender ID	E_{ava}	D_{link}	R_{link}

Figure 4.2: HELLO message structure

4.2.2 Link Cost Function

Geographic routing protocols are efficient in wireless networks [72], geographic routing accomplished based solely on location information of nodes, nodes need to know only the location of their one-hop neighbors, the discovery floods and state propagation are not required, the used memory at each node is minimal, the bandwidth consumption is

reduced, the energy is conserved and the overhead is minimal. All these gains are an important concern for resource constrained networks like WSNs.

Benefits from geographic routing and in order to minimize the number of sensor nodes used to route data between source and destination [73], we consider the idea of greedy forwarding as one of the metrics to calculate the link cost function. The sender node searches its neighbors' list looking for the neighbor node that is closest to sink while at the same time satisfies the application requirements among all its forwarding candidates. Then the expected progress in distance between a sender node s_x and a receiver node s_y to the sink, $C_{dis_{s_x,s_y}}$, can be defined as follows:

$$C_{dis_{s_x,s_y}} = \frac{1}{(d_{s_x,sink} - d_{s_y,sink})} \tag{4.10}$$

where $d_{s_x,sink}$ and $d_{s_y,sink}$ are the distance of a sender node s_x and a receiver node s_y to the sink, respectively.

The implicit aim of this strategy is to minimize C_{dis} in order to minimize the hop count between source and destination. However, using greedy forwarding strategy in sensor network, may lead to the dead end problem [49], which occurs when a message is forwarded to a node with no neighbor that is closer to the sink than the node that currently has the packet, and is called local optimum. Avoiding this problem is a challenging issue for any geographic greedy forwarding approach. Although a dense deployment of wireless nodes can reduce the incidence of this problem in the network, but it is still possible for some nodes experience a local optimum. In the proposed MQoSR, if a sender node does not have any neighbor closer to the sink than itself, the message is forwarded according to the required QoS function only, equation (4.9).

The link cost function can be written as:

$$C_{link} = C_{dis} \times C_{req} \tag{4.11}$$

where C_{link} is the link cost function and C_{req} is the cost of the required QoS, equation (4.9).

4.2.3 Paths Discovery Phase

The route request phase is started when the source node has data packet to transmit to the sink to which it has no available route by broadcasting a RREQ message (Figure 4.3). Each source node reports the application requirements in terms of R_{req} and D_{req} in the *Required QoS* field of the RREQ message. Each source node also initializes the values in the Path Parameters field, D_{path}, R_{path} and hop to zero, one and zero, respectively. The source node then broadcasts the RREQ to all its neighbors within its transmission range in which the path parameters are updated along the available paths to the sink. The route discovery phase is therefore introduced. Upon receiving the RREQ, each intermediate node selects one node as the next hop from its neighbor list to forward the RREQ depending on the link cost function, equation (4.11).

If N_{s_x} is a set of neighbors of sensor node s_x then the RREQ message will be forwarded to the neighbor whose total cost function is the least ($min_{s_\delta \in N_{s_x}} C_{link}$). This node is chosen and is reserved for that path to avoid having paths with shared nodes. However, if the selected node is already reserved then the next smallest C_{link} node will be selected and so on. The selected node then modifies the Path Parameters field in the RREQ.

The Path Parameters field in RREQ messages are initialized at the source node and updated at each intermediate node as follows,

1. Compare the nodes' available energy with the value reported in E_{min} field, E_{min} is the minimum available energy at a node on any path, and if $E_{ava} < E_{min}$ then $E_{min} = E_{ava}$, otherwise there is no change. Note that the initial value of E_{min} is equal to the E_{ava} of source node. Thus, E_{min} is capturing the minimum reading of energy along the path.
2. $D_{path} = D_{path} + D_{link}$.
3. $hop = hop + 1$.
4. $R_{path} = R_{path} \times R_{link}$.

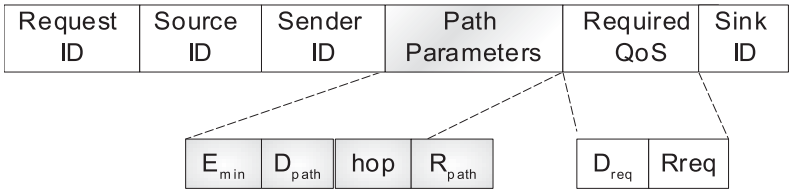


Figure 4.3: RREQ message structure

After receiving all the RREQ messages, the sink estimates the number of all available disjoint paths to the source and obtains the information about each path. The sink podcasts RREP message (Figure 4.4) after evaluating the optimal paths as described in Section 4.3.3.

Request ID	Source ID	Sender ID	No. of paths or Fragments

Figure 4.4: Route replay message structure

4.2.4 Illustrative Example

To understand the role of next node selection process and to underline the importance of selecting different nodes when different QoS is requested, we illustrate MQoSR with the example in Table 4.2, based on Figure 4.5 information. Each sensor node is labeled with four attributes in the form of (distance to sink, available energy, link delay, link reliability). When the source node, *Source* in Figure 4.5, originates a data packet, the RREQ message is broadcasted to all the neighbors of the source node within its transmission range. Consider a node, say *0*, as one of the source neighbor nodes which is located at a distance of, $d_{s_0,sink}$ = 20, from the sink. Node *0* initiates the next node selection process to select next node depending on the requested QoS and the available resources, from its list of neighbor nodes 1, 2, 3, 4 and 5.

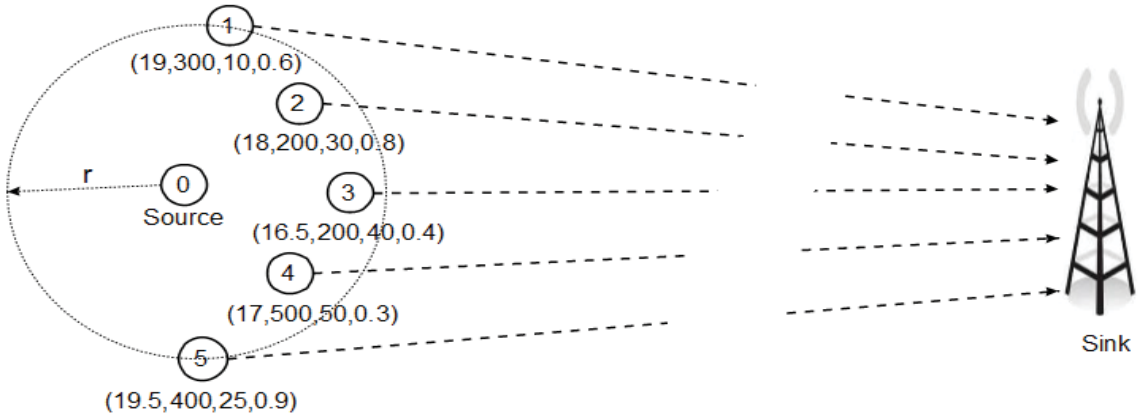


Figure 4.5: Next node selection process in MQoSR, each node is labeled with a $(d_{s,sink}, E_{ava}, D_{link}, R_{link})$ quadruple.

Table 4.2 summarizes the next node selection process introduced in MQoSR protocol; different nodes are selected depending on the requested QoS and the available resource. In Table 4.2, R_{req}, D_{req} and E_{ava} are set to 0.9, 120 and 100, respectively. The

link cost function is fixed for each node while the required QoS cost functions are altered based on the QoS classes.

Table 4.2: Example of next node selection process in MQoSR

Required QoS	***Costs***	***Node 1***	***Node 2***	***Node 3***	***Node 4***	***Node 5***	***Next Node*** $\boldsymbol{min\ C_{total}}$
Class 1-7	C_{link}	1	0.5	0.285	0.333	2	
Class 1	C_{req}	100/300 = 0.333	100/200 = 0.5	100/200 = 0.5	100/500 = 0.2	100/400 = 0.25	4
	C_{total}	0.333	0.25	0. 142	0.066	0.5	
Class 2	C_{req}	10/120 = 0.083	30/120 = 0.25	40/120 = 0.333	50/120 = 0.416	25/120 = 0.208	
	C_{total}	0.083	0.125	0.095	0.138	0.416	1
Class 3	C_{req}	0.9/0.6 = 1.5	0.9/0.8 = 1.12	0.9/0.4 = 2.25	0.9/0.3 = 3	0.9/0.9 = 1	
	C_{total}	1.5	0.562	0.641	0.999	2	2
Class 4	C_{req}	1.833	1.625	2.75	3.2	1.25	
	C_{total}	1.833	0.813	0.784	1.066	2.5	3
Class 5	C_{req}	0.416	0.75	0.833	0.616	0.458	
	C_{total}	0.416	0.375	0.237	0.205	0.916	4
Class 6	C_{req}	1.583	1.375	2.583	3.416	1.208	
	C_{total}	1.583	0.688	0.736	1.138	2.416	2
Class 7	C_{req}	1.916	1.875	3.083	3.616	1.458	
	$\boldsymbol{C_{total}}$	1.916	0.938	0.879	1.204	2.916	3

4.3 PATH METRICS AND END-TO-END QOS

The routing protocol used is the on demand routing protocol that builds multiple node-disjoint paths, whereby a fair fault-tolerant mechanism is provided by the availability of alternate paths and the use of EC at the source node. The process of selecting the routing paths, the number of these paths and the allocation strategy of data packet on each route are related to the end-to-end application requirements and are decided at the sink side.

In general, the multi-QoS optimization routing problem is to find a set of available multi-disjoint paths, $P = \{path_1, path_2, \ldots, path_n\}$ from the source node to the sink node that satisfies the following objective function,

$$f: min(f_E),\ min(f_D),\ min\ (-f_R) \tag{4.12}$$

This can be rewritten as,

$$min \sum_{p=1}^{np} f \times W \tag{4.13}$$

where W is the weight set for the QoS required by an application.

The first term in the objective function, equation (4.12), specifies the energy consumption from data packet transmission; the second term specifies the delay accrued in data transmission, while the third term specifies the reliability of data transmission. Hence, the objective function is to minimize data transmission power in order to extend the network lifetime, minimize the data transmission delay while maximize the data transmission reliability. This function is subject to the following constraints:

$$min \sum f_E \tag{4.14}$$

$$f_D \leq D_{req} \tag{4.15}$$

$$f_R \geq R_{req} \tag{4.16}$$

$$min\ \{np\};\quad n \geq np \geq 1 \tag{4.17}$$

4.3.1 Path Cost Function and Multipath Selection Algorithm

The sink node satisfies an application requirement by selecting path or paths based on the requested QoS and the available paths conditions. After receiving all the RREQ messages and from the information received in each message, the sink node builds the sink decision table (Table 4.3). The sink decision table contains the number of available node-disjoint paths and the quality of each path in terms of the minimum available energy (E_{min}), the path delay (D_{path}), the path reliability (R_{path}) and the number of hops from the source node to the sink (*hop*). For each available path, the path cost (*Cost*) is calculated according to Algorithm 1. These paths are sorted in ascending order according to their cost in order to select the path with the minimum cost first.

Table 4.3: Sink decision table

Path	$\boldsymbol{E_{min}}$	$\boldsymbol{D_{path}}$	$\boldsymbol{R_{path}}$	***hop***	***Cost***
1	E_{min_1}	D_{path_1}	R_{path_1}	hop_1	$Cost_1$
2	E_{min_2}	D_{path_2}	R_{path_2}	hop_2	$Cost_2$

:	:	:	:	:	:
n	E_{min_n}	D_{path_n}	R_{path_n}	hop_n	$Cost_n$

Algorithm 1: Paths Selection Algorithm

Input:

Class type for each packet

Table 4.3 (which contains each path parameters)

R_{req} and D_{req}

Output:

Table 4.3 with *Cost* column sorted in ascending order

Use Table 4.1 to assign C_E, C_D , C_R for each packet of *class* type

for (j=1; j ≤ n; j++)

{

Calculate E_{path_j} using Equation (4.1);

$$Path[j].Cost = \frac{E_{path_j}}{E_{min_j}} \times C_E + \frac{D_{path_j}}{D_{req}} \times C_D + \frac{R_{req}}{R_{path_j}} \times C_R \ ;$$

}

Ascending Sort Path[].Cost; // first Path[].Cost is the smallest one and upwards

Algorithm 1 is executed at the sink side one time only for each class and the performance and complexity of the algorithm depends on the value of the available paths (n). The for loop, *for (j=1; j ≤ n; j++)*, executes n times and takes $O(n)$. The function *Ascending Sort* is executed one time and using Bubble sort function takes $O(n^2)$ or using Merge sort function takes *O(n log n)*. Therefore, the overall algorithm execution time is *O(n log n)*.

4.3.2 Number of Used Paths

The number of paths to route data is based on the requested QoS since the selection criteria can be towards different objectives. We think of path failures as Bernoulli distribution. When a path fails, all the messages sent over the path are lost. On the other hand, when a path succeeds all the messages sent on it are successfully received. In MQoSR protocol, the sink uses multipath routing and EC only when the requested QoS in terms of reliability cannot be achieved. Thus when using multipath routing, the original packet is coded at the source node only to generate $M + K$ coded sub-packets before transmitting on the np selected from the n available node-disjoint paths between the source and the sink. The number of paths used to route data in MQoSR protocol can be defined as the routing strategy, T, and presented as:

$$T = \begin{cases} np \quad if\ n \geq np \geq 2, & Multipath\ routing \\ 1 \quad otherwise, & Single\ path\ routing \end{cases} \tag{4.18}$$

The sink uses the packet reliability, R_{e2e}, to determine the number of multipath, np, using Algorithm 2.

Algorithm 2: Select Number of Used Paths

n= number of available disjoint paths (source to sink)

R_{e2e} =1; // Initialization

np = 0; // Initialization

for (j=1; j ≤ n; j++)

{

$R_{e2e} = R_{e2e} \times \left(1 - R_{path_j}\right)$ // Calculate R_{e2e} using R_{path} received in RREQ

np = np++; // Add another path to the used paths

if ($(1 - R_{e2e}) \geq R_{req}$*)*

{

number of paths to be used = np;

break;

}

}

Drop Packet; // When np = n and R_{req} is not achieved packet is dropped

4.3.3 Route Replay and Data Transmission

Once the sink decides on the number of routing paths, *np*, to be used, it replies to the source node the results through RREP messages that travel on the selected node-disjoint paths. The source node then encodes each data packet using RS codes to have in total a set of $M + K$ fragments and by receiving M fragments out of these $M + K$ the original data packet can be reconstructed, as in Section 3.4. For each data packet M the parity fragments K are added such that the number of fragments on each path follows, $x_j = \lceil (M + K)/np \rceil$, $j = 1, 2, \ldots, np$ and $1 \leq x_p \leq M - 1$. The first paths that have the highest requested QoS level, Algorithm 1, are allocated more fragments than other paths. With such allocation, a high recovery level is achieved since the allocated fragments on any path are less than M. After the selection of *np* disjoint paths for classes 3, 4, 6 and 7, and after adding coding fragments, the source node can begin sending data to the destination along these paths.

4.4 ANALYSIS AND SIMULATION RESULTS

In this section, we present the simulation setup for MQoSR protocol, followed by the performance metrics and comparisons.

4.4.1 Simulation Setup

We have conducted an extensive simulation study using C++ and MATLAB to evaluate the performance of the proposed protocol for various QoS requirements. We compare the proposed MQoSR protocol with the MCMP model [54], which also considers multi-constraints QoS routing and shows its performance to be better than

similar other protocols. Also we compare the proposed protocol with higher achievable performances of an ideal QoS routing protocol, God routing [55, 75] in which each node is aware of the direct links delay, and reliability and use multipath routing based on this knowledge. A fair comparison can only be achieved with careful selection of simulation parameters and by using similar simulation parameters used to evaluate [54] and [75]; we ensure that the obtained results are directly comparable to those published previously.

At higher node densities, the likelihood of finding node-disjoint paths increases [17]. Thus, in order to increase the probability of finding these paths to evaluate the performance of the proposed protocol, we consider a network where 80 to 250 nodes are randomly scattered in a field of 200m × 200m area. We assume that all sensor nodes are static after deployment with transmission range of 40m and initial energy of 2J. The simulation parameters that we use are as follows. Simulation time is set to 1000 sec and the size of a data packet is 128 *bytes* with a fixed generation rate of 1 packet/ sec. Thus, the total number of data packets transmitted through simulation, pkt_{total}, is 1000. e_t = e_r = 90 nJ/ bit, ε_{amp} = 10 pJ/ (bit.m^2). Source node is picked randomly and the position of the sink is fixed in the top left side of the simulation area. To evaluate the worst case where link delay and reliability change suddenly at any transmission instant, link reliability and delay are randomly distributed. Links' reliability are uniformly distributed in the range of [0.8, 0.9] and the delay is in the range of [1, 50] ms. Simulation results are obtained from different configurations, multiple runs, to reduce the effect of the position of sensors and the results shown are averaged over 10 simulation runs.

4.4.2 Performance Metrics

The following performance metrics are used to evaluate MQoSR protocol;

- Probability of successful transmission is the probability of packets achieving the reliability requirements, equation (4.8).
- Probability of packets received on-time is the probability of packets achieving the delay requirements.
- Data delivery ratio, the percentage of the packets sent by the source nodes and received by the sink, equation (4.5).
- Average end-to-end delay per node for each transmission is the period of time packet takes to reach the sink.

Average end-to-end delay = total end-to-end delay / (number of packets received × number of nodes)

- Average energy consumption per transmission, which is the index of the network lifetime; less energy consumption per transmission indicates more network lifetime. Network lifetime is given in terms of when the energy of a first node drops under the energy threshold.

Average energy consumption = total energy consumption / number of packets received.

- Average routing overhead, is the average number of routing packets transmitted to deliver a data packet, each hop transmission is counted as one routing packet. This is an index of the energy efficiency; more messages transmitted to deliver the data packet indicate higher energy consumption.

$$\text{Average routing overhead} = \left(\sum_{j=1}^{j=pkt_{source}} \sum_{p=1}^{np} hop_{jp}\right) / pkt_{total}$$

4.4.3 Simulation Results

We begin by examining the effects of the proposed seven QoS for different requirements in terms of reliability and delay using the same values as in Table 4.2 and the number of network nodes is set to 250. To highlight the ability of MQoSR protocol to distinguish services in the reliability domain, Figure 4.6 compares the probability of achieved reliability and the average energy consumption for the classes (3, 4, 6 and 7) that considered reliability as a metric. An average delay requirement of 90*ms* is used. Figure 4.6 indicates that more than 87% of total packets sent by all sensor nodes in the network achieved the requested reliability for the high reliability requirements (R_{req}= 0.85, 0.9 and 0.95) and 100% achieved for low reliability requirements (R_{req}= 0.7, 0.75 and 0.8). Note that, multipath routing is used and the number of these paths is related to the requested reliability. Thus, the energy consumption is increased when reliability requirement increases.

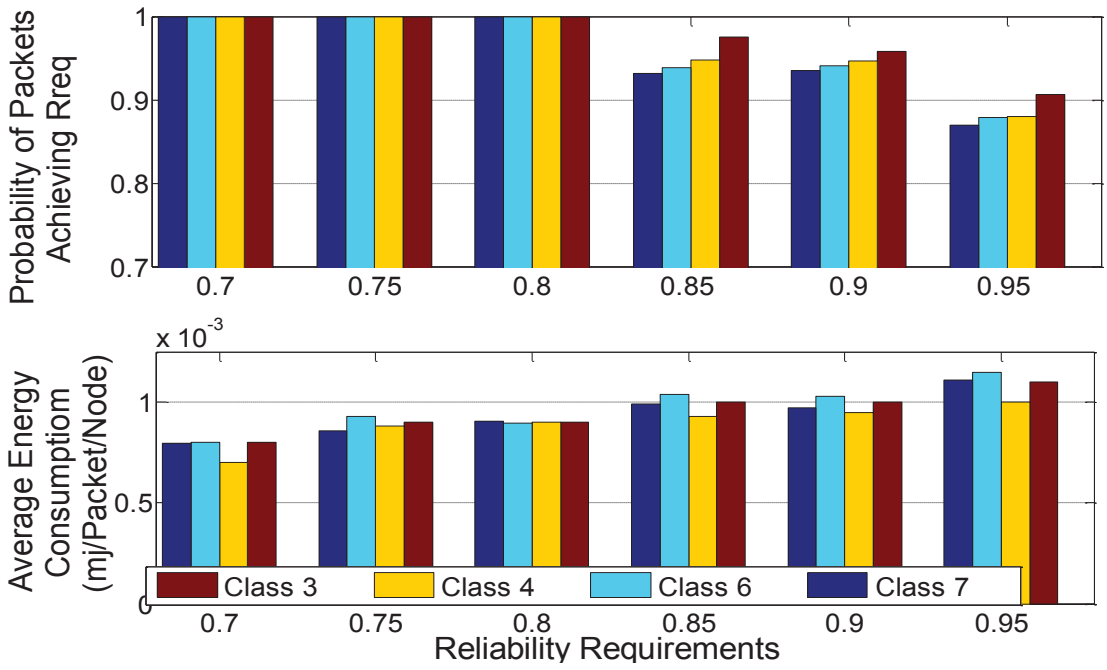


Figure 4.6: Probability of achieved reliability and average energy consumption vs. requested reliability

To distinguish services in the timeline domain, Figure 4.7 illustrates the end-to-end probability of packet received on time and the average energy consumption per transmission for all the classes (2, 5, 6 and 7) that considered delay as a metric, requested reliability of 0.7 is used. The end-to-end delay requirements are achieved up to 84% for the application with strict delay requirements (D_{req} < the average requirements of 90 ms) and up to 99% for the application with a relaxed delay requirement (D_{req} > the average requirements of 90 ms). Note that for classes 6, and 7, the energy consumption per transmission is higher than the other classes since reliability is also considered as metric in these classes and this reflects the energy consumption in the network.

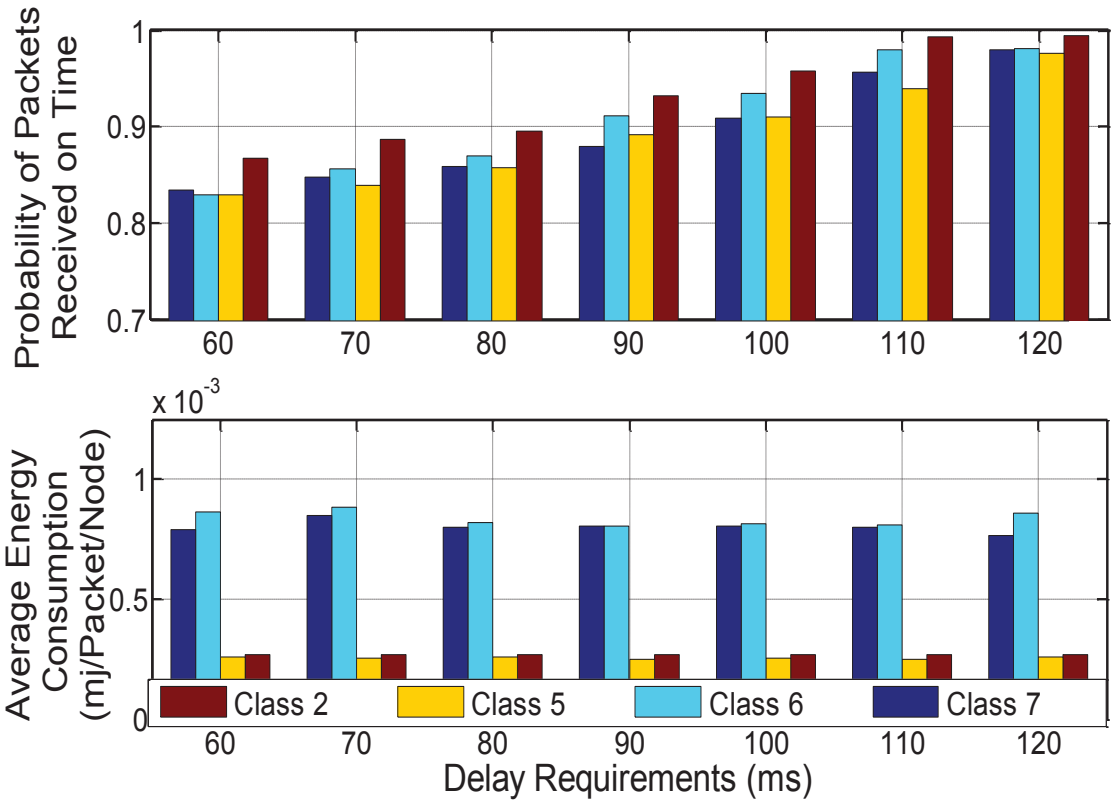


Figure 4.7: Probability of packets received on time and average energy consumption per transmission vs. requested delay

Next, we change the number of network nodes and measure the resulting effects on the end-to-end delay, data delivery ratio, network lifetime and routing overhead. Figure 4.8 shows that the average end-to-end delay per packet for all the classes that consider delay in MQoSR (classes 2, 5, 6 and 7). As expected, routing for the proposed classes with delay metric have much lower average delay than that of the MCMP protocol. Note that God routing achieves the lowest end-to-end delay per transmission. On the other hand all the delay classes in the proposed MQoSR protocol fulfill the delay requirements of 90 msec.

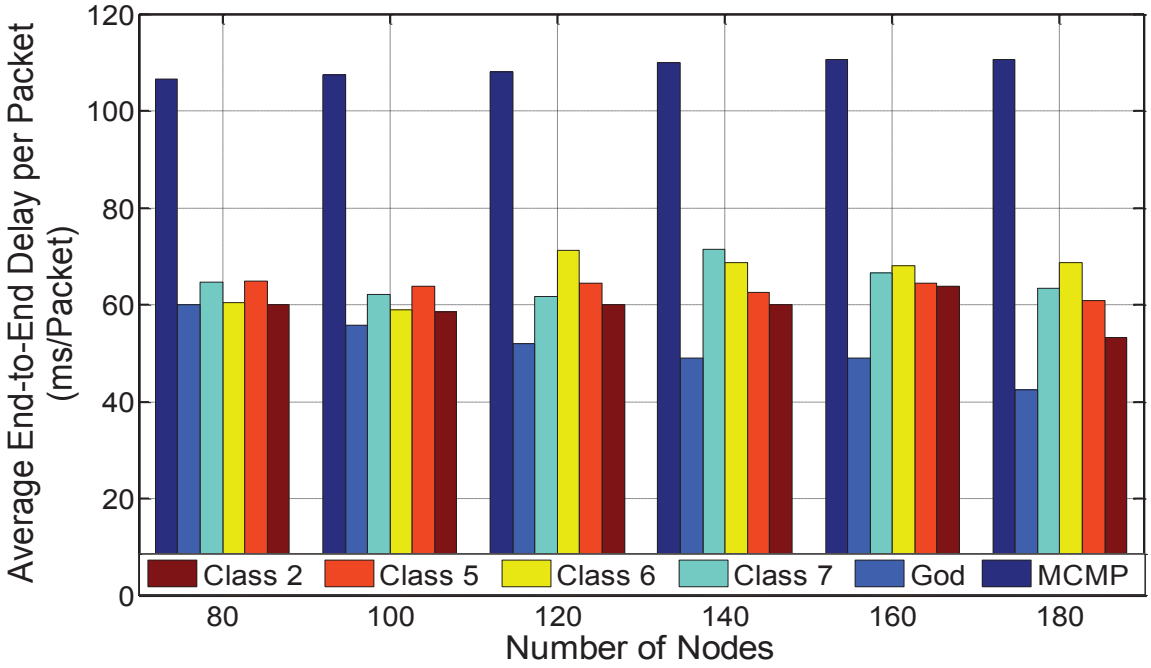


Figure 4.8: Average end-to-end delay per transmission vs. number of nodes

Figure 4.9 illustrates the data delivery ratio achieved for various number of nodes for the classes that consider the reliability metric in MQoSR compared with MCMP and God routing protocols. MQoSR outperforms the data delivery ratio for that of MCMP protocol and reached about 100% delivery ratio similar to God routing. In MQoSR, erasure coding is used to route data on multipath and the selection strategy of links and paths are toward increasing reliability only or reliability combined with other required QoS.

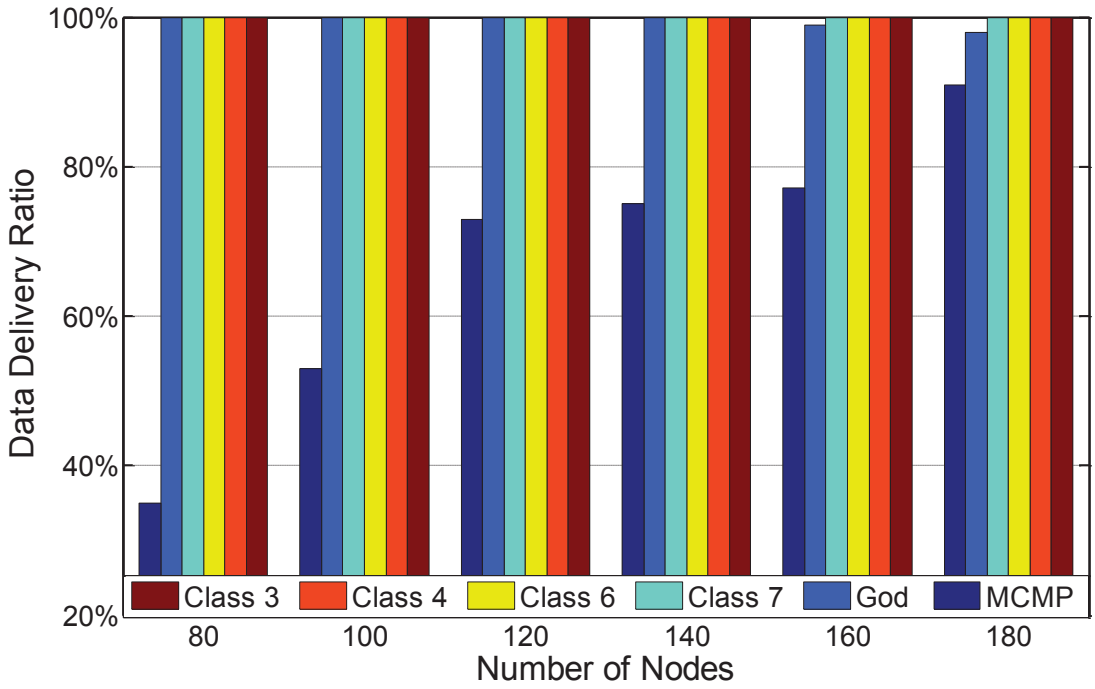


Figure 4.9: Data delivery ratio vs. number of nodes

The average network lifetime obtained in the network for different number of nodes using the classes that consider energy as a metric (classes 1, 4, 5 and 7) are illustrated in Figure 4.10. It is clear that MQoSR protocol highly outperforms the MCMP and God routing protocols in terms of decreasing energy consumption at the network towards extending the network lifetime. This is due to the fact that in MCMP protocol, the data packet is transmitted on more paths than in MQoSR protocol. In MQoSR, erasure coding is used; therefore data packet is split on the used paths. While in God routing protocol the next node selection process decides on links with the least delays or maximum reliability, possibly including nodes with low available energy, thus the same node transmits more packets hereafter the network lifetime depletes earlier. However, the next node selection process used in MQoSR for classes 1, 4, 5 and 7 chooses the next node with the maximum available energy and with the highest progress to destination as well as the

path selection process decides on paths with the maximum available energy of its nodes and the minimum energy consumption among other paths. This energy conservation strategy that is followed by the proposed MQoSR protocol results in extending the lifetime of network nodes.

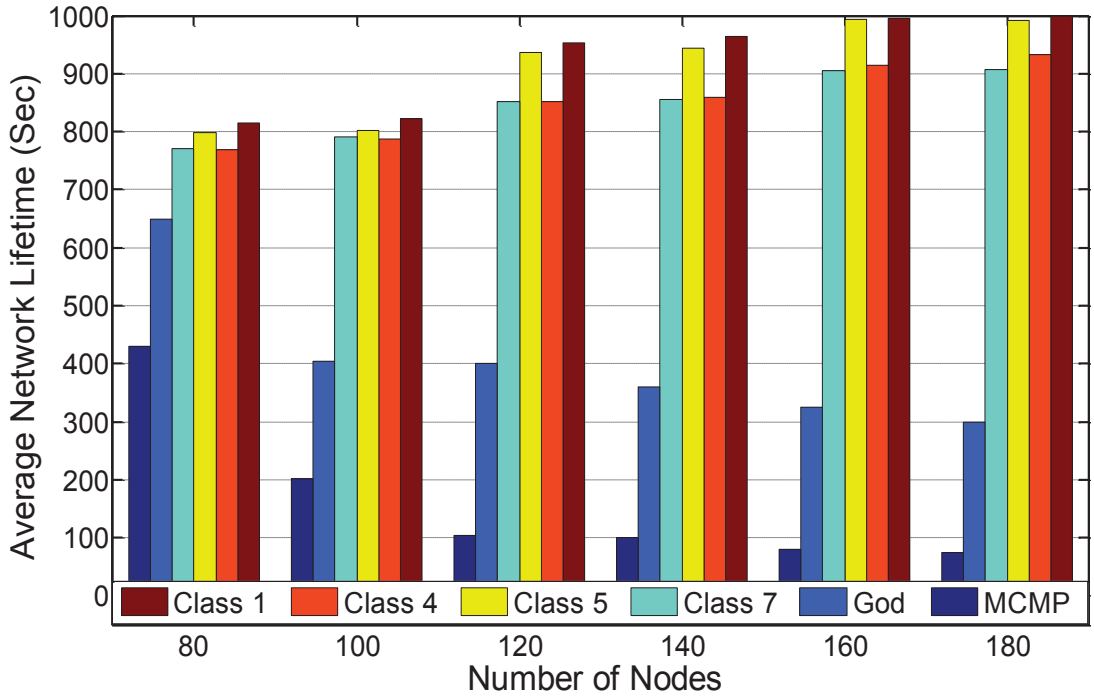


Figure 4.10: Average network lifetime vs. number of nodes

The routing overhead of transmissions per data packet is presented in Figure 4.11. Classes 1, 2, and 5 in MQoSR protocol introduce low routing overhead, similar to God routing, since these classes use a single path routing and the selection process of links and paths are toward decreasing delay and/or energy combined with the minimum distance to destination. Classes 1 and 5 in MQoSR, show an average routing overhead slightly less than that of God routing since links and paths selection strategies for these classes elaborate in decreasing the number of nodes involved in routing to extend the lifetime of

the network. However, the multipath routing classes in MQoSR, classes 3, 4, 6 and 7 introduce a higher routing overhead than God routing, but still gain more advantage than MCMP protocol. This is due to the fact that the number of routing paths used in MQoSR protocol is less than that of MCMP protocol which in turn reflects the communication overhead.

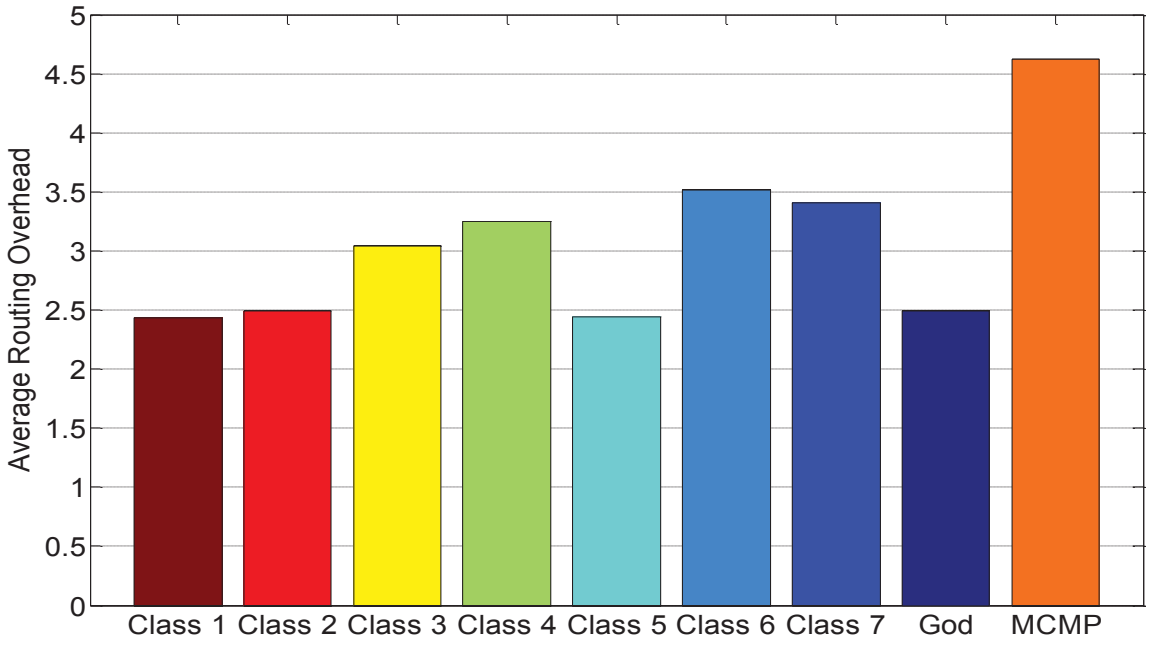


Figure 4.11: Average routing overhead, *N*=250

4.5 SUMMARY

In this chapter, different classes of QoS are modeled to provide multi-objective QoS routing in WSNs to deal with diverse requirements of different applications under various network constraints. The proposed protocol integrates multi-criteria for routing decision by partitioning the requested QoS into two sub-networks cost metrics; the sensor nodes cost metric where the link condition and available resources at each intermediate sensor node is collectively used to direct the data packet along the most appropriate links toward the sink, and the path cost metric where the end-to-end metrics are calculated to achieve the requirements while minimizing the overall network resource consumption. The strength of the MQoSR protocol lies in the fact that the sensor nodes and the sink change their routing policy according to the current QoS requirements by an application.

The proposed protocol is evaluated under different scenarios and the results confirm that MQoSR protocol that takes into account variations of the link weights in selecting a single path or multipath can satisfy the application requirements in terms of reliability and delay in an energy efficient way. Furthermore, MQoSR highly outperforms the MCMP and God routing protocols proposed in the literature in terms of energy consumption, data delivery, average end-to-end delay and routing overhead.

Chapter 5

QOS-AWARE CROSS LAYER ROUTING

Due to the increased use of sensor nodes in a variety of application fields, wireless sensor networks need to handle heterogeneous traffic with diverse priorities to achieve the required QoS while considering the unique properties of sensor networks, energy awareness and robust protocol design at all layers of the networking protocol stack is required.

In the network layer, the main functions are to provide end-to-end data routing and congestion control. Therefore, the end-to-end requirements guarantee cannot be only provided by QoS routing in a network layer; it is needed to investigate the other layers that allocate resources like MAC layer. The MAC layer plays a key role in determining the channel access delay, utilization and also coordinates the sharing of the wireless medium layer and can contribute to energy efficiency by minimizing the number of collisions, overhearing, overhead and ideal listening. Therefore, the MAC layer dominates the performance of the QoS support in the network [76].

In this chapter, we address the cross layer QoS-aware scheduling for wireless sensor network with respect to delay and reliability in an energy efficient way. The concept of cross-layer design in this thesis is about sharing of information among MAC and NET layers in order to select the best next node as shown in Figure 5.1. The process at the network layer comes up with the optimal decision based on the MAC layer parameters. A node-disjoint multipath routing is used and a QoS-aware priority scheduling considering

MAC layer is proposed to ensure that real time and non-real time traffic achieve their desired QoS while alleviating congestion in the network.

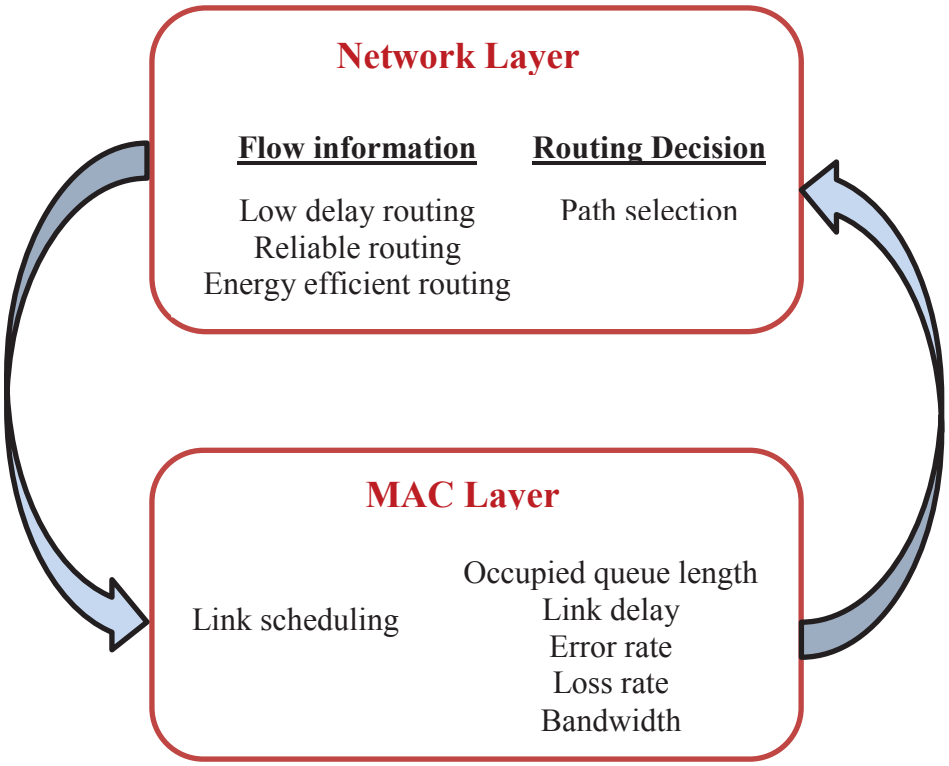


Figure 5.1: Proposed cross-layer design

5.1 PROPOSED PRIORITIZED SCHEDULING

In this section, the joint functionalities among the layers especially the routing and MAC layers are considered. A cross-layer design is proposed between the routing and MAC layers where the end-to-end QoS requirements are enforced through sensors decision of next hops according to the neighbors state and the required QoS. However, the end-to-end requirement is guaranteed jointly by the local decisions of these sensors

and the sink decision on the used paths and the number of these paths, as proposed and presented in Chapter 4.

5.1.1 Network Model and Assumptions

We model a WSN with N nodes and one sink as an undirected graph, $G = (S, L)$ where S is the set of nodes and L is the set of all possible communication links. Let s_x be node x in S and $l_{s_x s_y}$ the link between s_x to s_y where x and $y \in N$. In addition to these defined in Chapter 4, the following definition is used in this chapter:

- Queue length is one of the parameter used to estimate congestion at a node and congestion of a node is represented as the load on that node. Therefore, we use link load, $Load_{s_x}$, as one of the node metric as;

$$Load_{s_x} = B_{Q_{s_x}} / B_{max_{s_x}} \tag{5.1}$$

where $B_{Q_{s_x}}$ and $B_{max_{s_x}}$ are the length of occupied and maximum buffer of node s_x, respectively.

The smaller $Load_{s_x}$ at a node, the more chance to accept new traffic.

5.1.2 QoS Provisioning

The goal of the proposed QoS-aware routing protocol is to achieve the requirements in terms of the end-to-end delay and the reliability of data transmission while extending the network life time. To achieve this goal, the parameters that influence delay, reliability of data transmission and energy consumption at each hop on the routing path/paths should be considered. In the proposed solution, we consider the following parameters in selection of next hop;

- High geographic progress toward the sink. Due to the profits from geographic routing, we considered the idea of greedy forwarding in order to minimize the number of sensor nodes used to route data between source and destination.
- High available energy. To provide load balancing in order to extend the network lifetime.
- High link reliability. Link reliability degradation at a node reflects the interference degree around that node and can lead to packet losses, which affects the reliability of data delivery to the sink.
- Less node congestion. Congestion at a node can lead to packet losses, increase transmission delay and influence the energy efficiency.

5.1.3 Traffic Classification and Prioritization

To support applications with diverse QoS requirements, we classify these requirements into four different classes concerning both delay and reliability. Packets are prioritized, $pkt_{priority}$, by reading the packet header which includes a priority number for each type of packet as follows:

- Class 1: for delay sensitive requirements where packets delivery requires delay constraints only. $pkt_{priority} = 1$.
- Class 2: packets delivery requires delay-bound and reliability. $pkt_{priority} = 1$.
- Class 3: this class belongs to normal applications; packets delivery requires no reliability and no delay constraints. $pkt_{priority} = 2$.
- Class 4: applications with reliability requirements only. $pkt_{priority} = 2$.

The proposed traffic prioritization scheme assigns priorities to traffic at the source node according to the delay requirement in order to guarantee the requested end-to-end delay in multi-hop wireless networks. As shown in Figure 5.2, a classifier is used in the network layer of each node. Therefore, each type of incoming packet is sent to the appropriate queue. The packets that are related to the high priority queue, Q_1, are the delay sensitive packets, the real time traffic with $pkt_{priority}$ = 1. The packets that are related to low priority queue, Q_2, are the non-real time traffic with $pkt_{priority}$ = 2. Then the length of occupied queue at any node, B_{Q_s}, is given as; $B_{Q_s} = B_{Q_1} + B_{Q_2}$, where B_{Q_1} and B_{Q_2} are the length of occupied queues of Q_1 and Q_2, respectively.

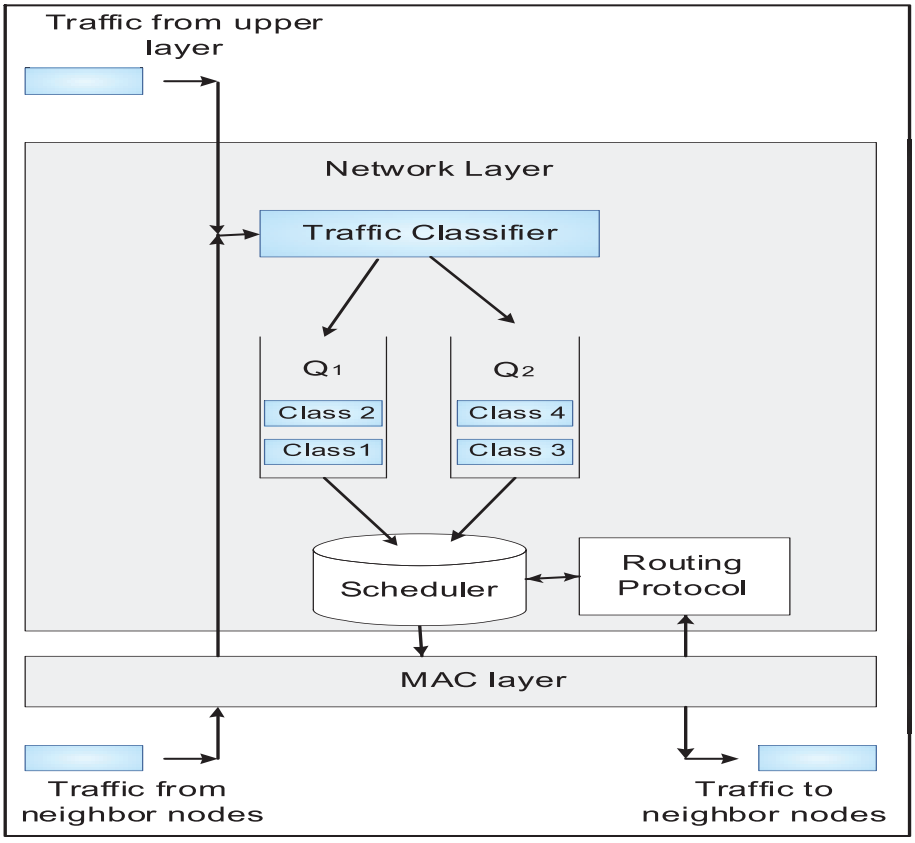


Figure 5.2: Queue model at a node

5.1.4 Queuing Model

In order to support service differentiation and to provide the requested requirements for the high priority traffic, we use a priority queuing protocol which prioritizes the packet transmission process at each node. Queue is used for storing the data temporarily and the length of queue is one of the parameter which is used to get an estimate of congestion at the nodes.

Using priority queue, the scheduler of the sensor node is serving different output queues; the high priority queue is served first. If there is no packet waiting in the high priority queue, then the low priority queue is served. However, if the amount of highest priority traffic is extreme then the lower priority queue may not get any service until the highest priority traffic is served completely, which is commonly known as the starvation problem.

The available bandwidth, *BW,* at a wireless link is shared among these two queues using the weighted round-robin (WRR) fashion. The queues that are used in the WRR are emptied in a round-robin fashion and if the queue has packets to transmit during that time slot it transmits the packets. Otherwise, it passes it to the next queue. The weight for each queue is configured according to the priority of the queue as follows: w_1= 2 is the weight of Q_1 and w_2= 1 is the weight of Q_2. Then, we can calculate the bandwidth assigned for each queue, bw_q , as:

$$bw_q = BW \times w_q \,/\, \sum_{m=1}^{2} w_m \tag{5.2}$$

where q is the queue number and is equal to is 1 or 2.

5.2 END-TO-END QOS SCHEDULING-BASED ROUTING

In this section, we present the parameters used in the proposed scheme, link cost and path cost functions and their influence on providing the required QoS. Also, we review the node-disjoint multipath process used and the criteria used to select these paths.

5.2.1 Initialization Phase

In this phase, sensor nodes introduce themselves to their one-hop neighbor nodes by sending HELLO messages (Figure 5.3). When a node receives a HELLO message, the node records the received information to update its neighbor table entries and the information related to each neighbor in the neighbors set, N_{s_x}. The collected information from the neighbors include, the neighbor identification number *Sender ID*, the available energy E_{ava}, the degree of load *Load* and the link reliability between the two nodes R_{link}. Unlike in Chapter 4, HELLO messages in this chapter include the degree of load *Load* to estimate the expected delay that data packet suffers when routed to next neighbor node.

Sender ID	E_{ava}	*Load*	R_{link}

Figure 5.3: HELLO message structure

5.2.2 Link Cost Function

To decide on the next hop, C_{link} is computed for each neighbor in the N_{s_x} set and the one with the minimum C_{link} value is selected as the next hop. The link cost function in Chapter 4 is updated and used in this chapter such that it includes the load metric *Load* for the candidate node instead of the delay of link in equation (4.9). Since the delay in

this chapter is estimated as the expected queuing delay (*Load*) that data packet suffers when routed to next neighbor node.

$$C_{link} = C_{dis\,s_x, s_y}\,(\alpha\,/E_{ava_{s_y}} + \beta\,Load_{s_y} + \gamma/\,R_{link_{s_x s_y}}) \tag{5.3}$$

where $E_{ava_{s_y}}$ is the available energy at the candidate neighbor s_y, $s_y \in N_{s_x}$, $L_{load\,s_y}$ is the load at node s_y, and $R_{link_{s_x s_y}}$ is the SNR on link $link_{s_x s_y}$. The weight α, β and γ are the weights that indicate the importance of each parameter in selecting the next hop and (α + β + γ) =1.

5.2.3 Path Discovery Phase

In the path discovery phase and in order to construct multi node-disjoint paths to the sink, RREQ message (Figure 5.4) is initiated at the source node as follows:

- *hop =0; hop* is the hop count at the path,
- $Load_{path}$ *=0;* $Load_{path}$ represents the number of loaded nodes along the path,
- D_{path} *= 0;* D_{path} is the end-to-end path delay,
- R_{path} *= 1;* R_{path} is the end-to-end path reliability.

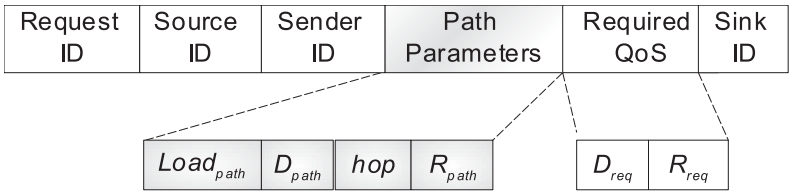


Figure 5.4: RREQ message structure

The source node also reports the application requirements in terms of end-to-end delay, D_{req}, and end-to-end data delivery reliability, R_{req}, in the RREQ message. The RREQ is then send to all the neighboring nodes in the $N_{s_{source}}$ set of the source node.

After receiving the RREQ message, each node in $N_{s_{source}}$ updates the information of the RREQ as follows before sending it to the selected neighbor.

- $hop = hop + 1$,
- If ($Load_s > Load_{thr}$) then $Load_{path} = Load_{path} + 1$. Otherwise, no change to L_{path}. L_{thr} is the value for load threshold,
- $D_{path} = D_{path} + D_{link}$,
- $R_{path} = R_{path} \times R_{link}$.

The RREQ message is then sent to the candidate neighbor with the minimum C_{link} value.

5.2.4 Path Cost Function

By receiving the RREQ messages, the sink estimates the number of all available node-disjoint paths to the source and uses the parameters of each path, the maximum available load of a node at a path $Load_{path}$, end-to-end delay of the path D_{path} and the path reliability R_{path} to calculate the cost function of each path, C_{path}, as follows;

$$C_{path} = \left(\alpha \frac{E_{path}}{E_{thr}} + \beta \frac{D_{path}}{D_{req}} + \gamma \frac{R_{req}}{R_{path}}\right) \times (Load_{path}) \quad (5.4)$$

where E_{thr} is the energy threshold value at a node to participate in a transaction. When a node has $E_{ava} < E_{thr}$ it cannot participate in data transmission, thus it is considered dead.

The path cost function, equation (5.4), is updated from the one presented in Chapter 4 such that the load metric ($Load_{path}$) is used to increase the cost of a path that suffers

from high load in order to avoid the congested node in the network. While in Chapter 4, the minimum available energy of a node on the path is used to avoid path with minimum energy in order to extend the network lifetime.

The sink evaluates the optimal paths for each traffic demand as follows:

- Assign the values of α, β and γ according to the requested requirements.
- Calculate C_{path} for all the available paths, *n*.
- Sort available paths according to their cost such that $C_{path_1} < C_{path_2} < \ldots < C_{path_n}$
- A scheduler is used to determine which path to select for current traffic demands based on the requested services class. In order to reduce network congestion and enhance the network performance, classes with higher priority will be transmitted in routing path/paths with the minimum C_{path}.

5.2.5 Route Reply and Data Transmission

The sink uses the packet requested reliability to determine the number of multipath, *np*, and the priority of each path. RREP message (Figure 5.5) is then sent to the source node through the selected path/paths. The number of these paths is decided according to Algorithm 2, Section 4.3.2. The sink transmits RREP message to the source node through the selected paths and each message carries the priority number of a path for each traffic demands.

Request ID	Source ID	Sender ID	Priority

Figure 5.5: RREP message structure

By receiving the RREP messages, the source node obtains the number of paths to be used and the priority value of each path-based on the path cost function introduced in Section 5.2.4. The path with the least cost function is assigned highest priority and so on. Source node then starts the FEC coding and fragments are assigned to each path such that the first fragment is assigned to the path with the highest priority. The second fragment to the second highest priority path and so on.

5.3 ANALYSIS AND SIMULATION RESULTS

In this section, we present the results from an extensive performance evaluation carried out using C++ code and MATLAB. Before we discuss the results, the simulation setup, network model and the performance metrics used to evaluate the prediction schemes are given.

5.3.1 Simulation Setup and Model

A wireless sensor network which comprises of 300 static sensor nodes is randomly distributed in 200m × 200m area. All sensor nodes have the same transmission radius of 40m. IEEE 802.11 is used for the MAC layer. It has been widely adopted and used in both traditional wireless networks and in multi-hop wireless sensor networks research. Source nodes are located in the left lower corner and sink node is located in the right upper corner of the simulation area like the model shown in Figure 5.6. Two sources targeting to a single sink is considered to generate traffic from 10 to 100 packets/s. First source generates real time traffic, *RT,* at 10% of the generated traffic while the second source generates the non-real time traffic, *NRT*. We change the total packets arrival rate at the sources from 2 to 20 packets/s for *RT* classes. Simulation results are obtained from

different configurations (10 runs) to reduce the effect of the position of sensors. At each point, the results shown are averaged over 10 simulation runs, for the *RT* classes, all the traffic 20 to 200 packets (that is 10 runs with 2 to 20 packets) with a 90% confidence interval, which is not plotted for the sake of legibility.

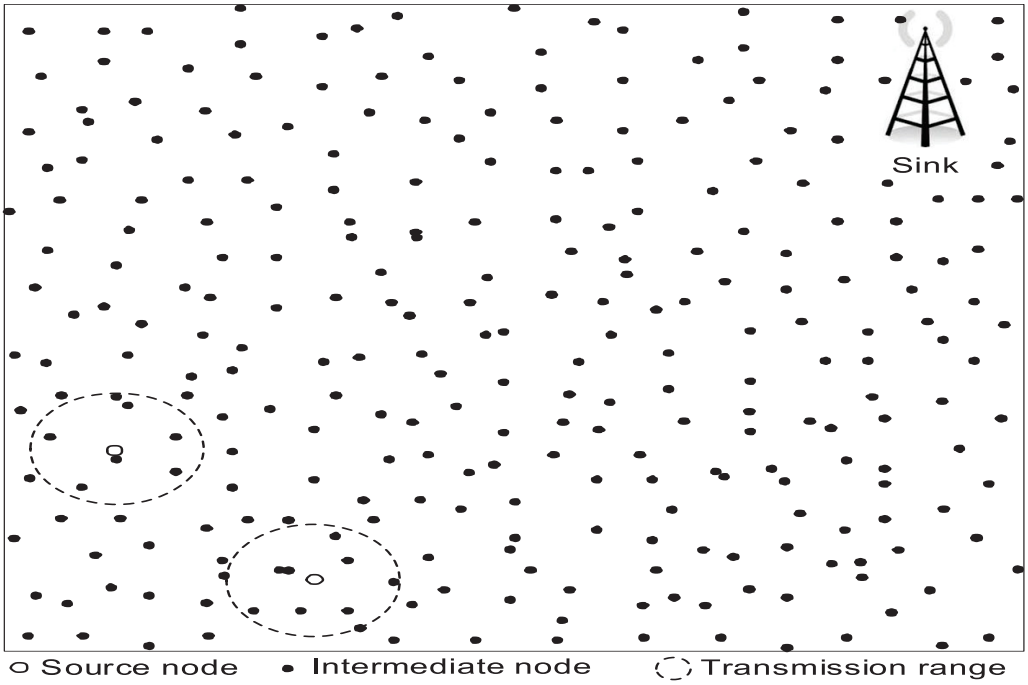


Figure 5.6: Simulation model

The simulation parameters used in this section are presented in Table 5.1.

Table 5.1: Simulation parameters

Network field	*200m × 200m*
Number of sensors	*300*
Simulation time	*100s*
MAC layer	*IEEE 802.11*
Transmission range	*40m*
Packet size (data + overhead)	*128 byte*
No. of source nodes	*2*
No. of sink	*1*
Each queue size	*50 packets*
e_{init}	*2J*
e_t	*50 nJ/bit*
e_r	*50 nJ/bit*
ε_{amp}	*100 pJ/(bit.m^2)*

5.3.2 Performance Metrics

We evaluate mainly the performance according to the following metrics:

- Average end-to-end delay: The average delay per node for each packet transmission to reach the sink.
- On-time reachability: The probability that a packet meets the required deadline.
- Average packet delivery ratio: The ratio of the number of packets received successfully at the sink to the total number of packets transmitted by the sources.
- Average energy consumption: The average energy consumed per-hop to transmit a data packet.

5.3.3 Simulation Results

Figures 5.7 and 5.8 illustrate the average end-to-end delay per packet and the on-time reachability for both real time and non-real time traffic, respectively. In order to focus on the timeliness domain, we use a non-strict reliability requirement of 0.7. Conversely, we use a strict real time requirement of 50ms. From the results, it is clear that the average delay increases as traffic rate increases and this is because traffic arrive faster than it can be process causing the queues at nodes to fill up and as a result increasing the delay of traffic. When more packets are sent, real time traffic are given the highest priority and processed first and this introduces more queuing delay for non-real time traffic at each sensor node.

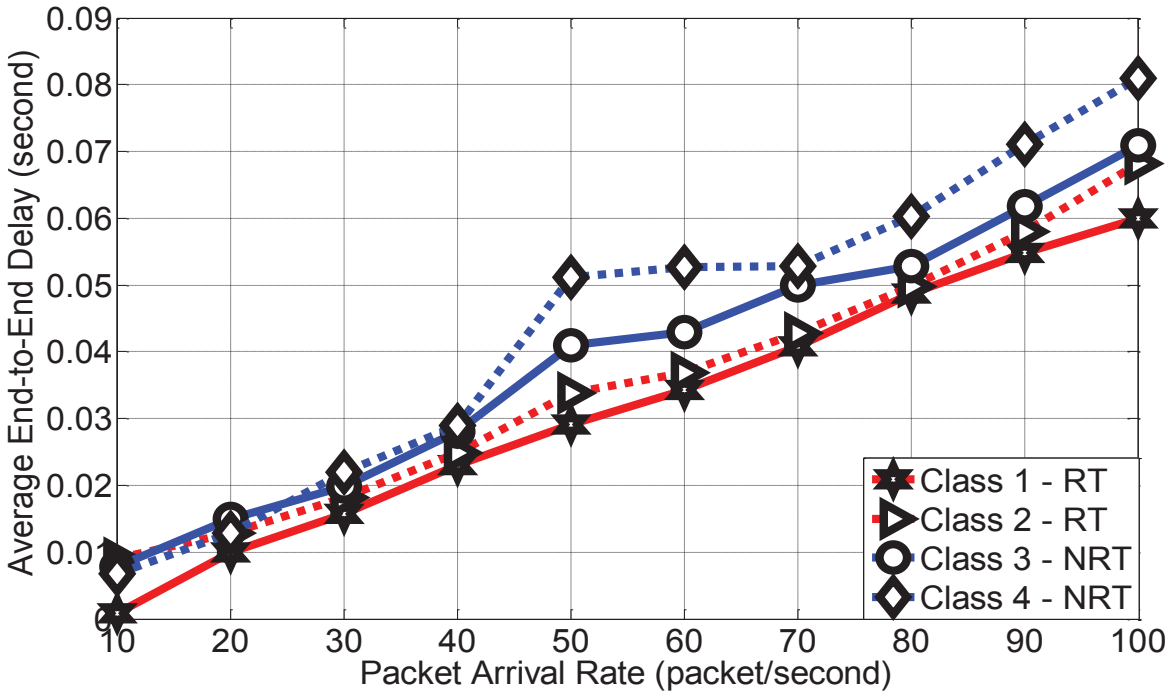


Figure 5.7: Average end-to-end delay

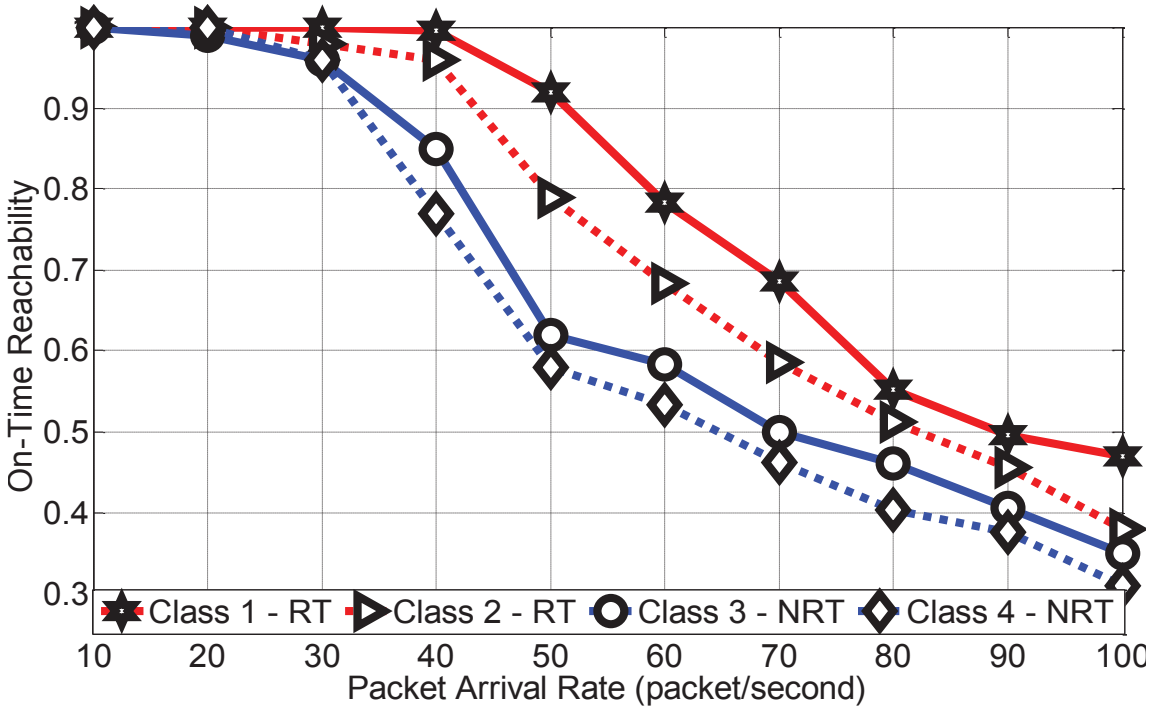


Figure 5.8: On-time reachability

Figure 5.9 shows that the average energy for a sensor node increases with the packet rate. From the results, we can see that Class 3 traffic has the least energy consumption among the other classes even when the arrival rate is increased. In Class 3 the forwarding strategy used consider the energy as the main metric as well as the load avoidance technique adapted to guarantee a fair service to real time and non-real time traffic. However, it is worth emphasizing that the price to meet the required QoS is the overhead introduced in terms of energy consumption.

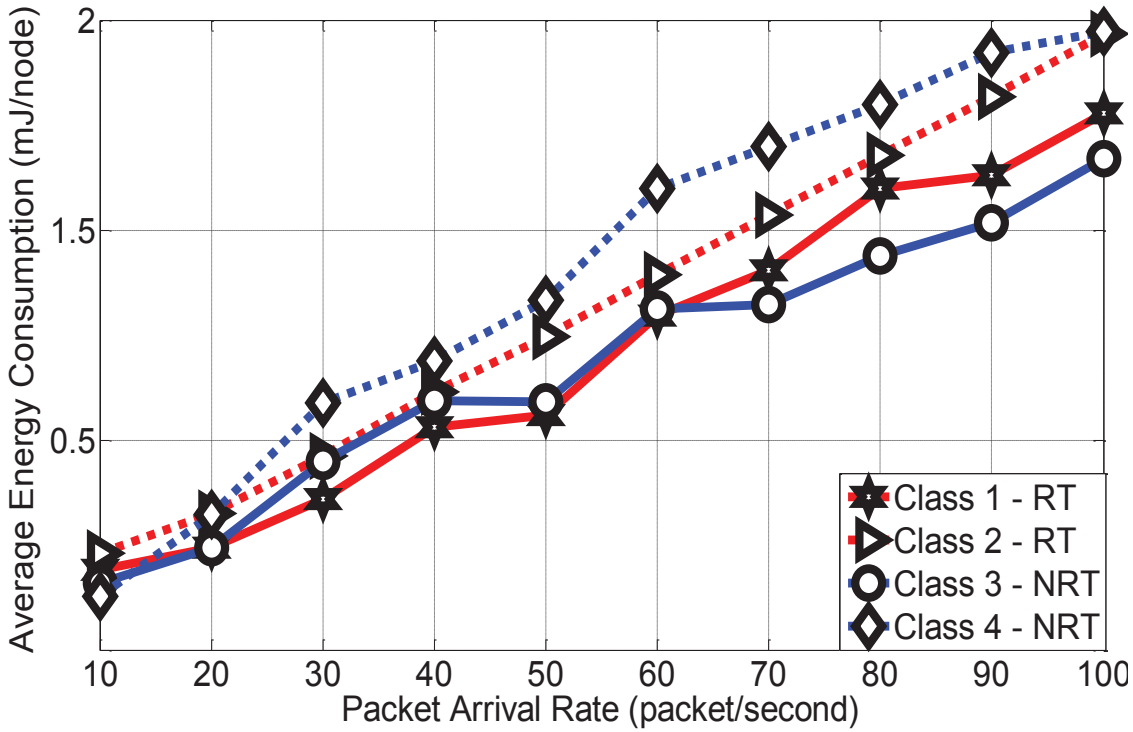


Figure 5.9: Average energy consumption

Figure 5.10 shows the performance of the proposed scheme in terms of the average packet delivery ratio for a strict reliability requirement of 0.9. We observe that even for

higher loads, most of generated packets achieve their reliability requirement. More packets are delivered even under heavy load and this is because the FEC technique is used to enhance the probability that packets are recovered at the sink as well as the forwarding strategy that consider load at sensor nodes is employed to alleviate congestion in the network and ensure that the real time traffic is not only reported fast but also not lost due to queue overflow at sensor nodes.

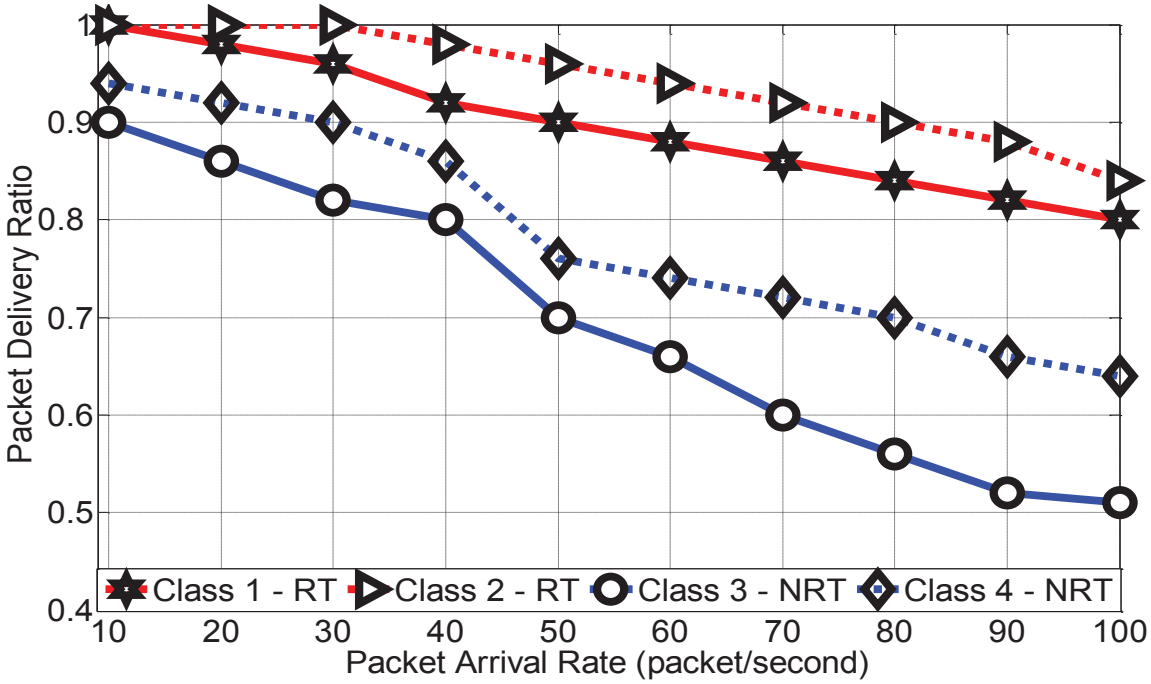


Figure 5.10: Packet delivery ratio

In Figures 5.11 and 5.12, we study the effect of packet drop probability on the performance of the proposed scheme in this chapter compared with MQoSR protocol proposed in Chapter 4. The probability of packet drop is varied from 0.01 to 0.05 and the packets arrival rate is set to 50 packets/second. In both figures, we use Delay (MQoSR) to refer to the applications with delay requirements used in MQoSR protocol and is

equivalent to Class 1 in this chapter and Delay + Reliability (MQoSR) to refer to the applications with delay and reliability requirements used in MQoSR protocol and is equivalent to Class 2 requirements proposed in this chapter. Thus, we are comparing the same routing strategies used for both the classes, nevertheless MQoSR does not consider congestion avoidance and prioritized packet scheduling compared to the proposed scheme in this chapter.

The results in Figure 5.11 show the average end-to-end delay per packet for each class by each protocol. With the increase in packet drop probability, MQoSR reaches high end-to-end delay compared to the proposed scheme in this chapter and this confirms the effectiveness of the congestion avoidance strategy adapted and the priority mechanism used in order to meet the timeline requirement. Note that the end-to-end delay for Class 1 and Class 2 are not affected much compared to Figure 5.7. In Class 1 packet has the highest priority among other classes and thus other low priority packet may be dropped due to congestion. However, In Class 2, multipath routing with FEC technique is used to deliver packet considering link quality as well as delay as metrics, thus packet can be still recovered at the sink when some of its sub-packet are dropped.

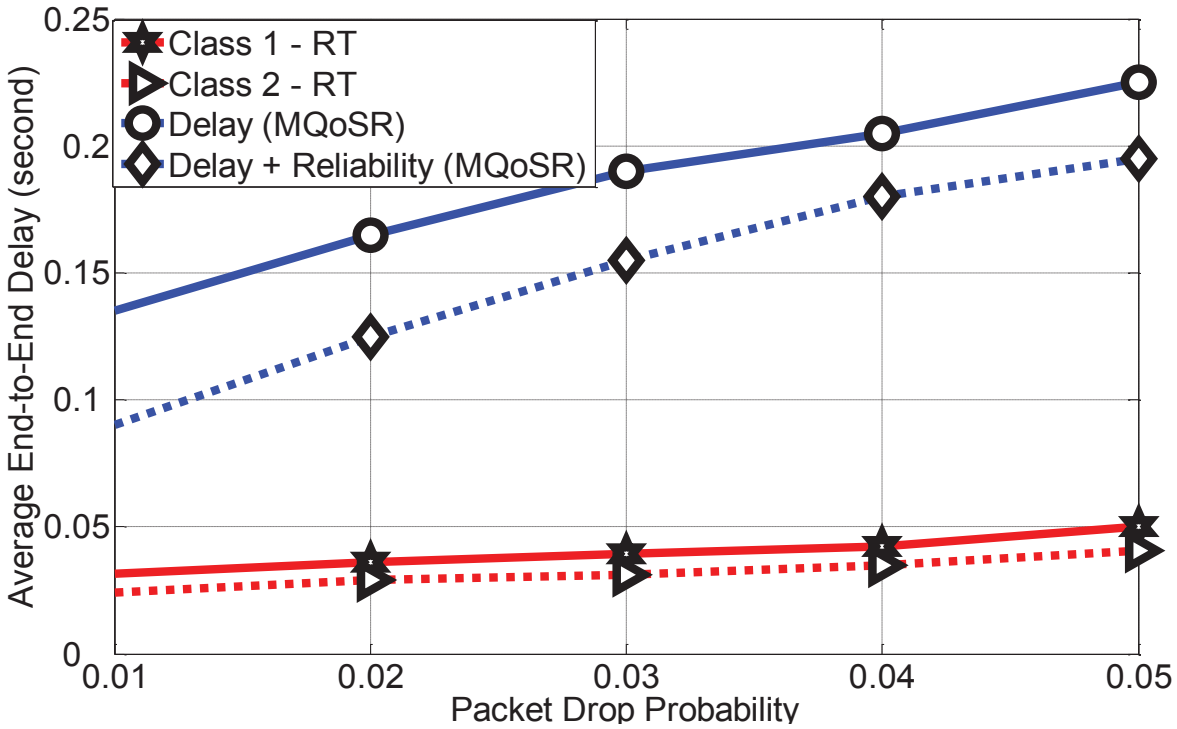


Figure 5.11: Average end-to-end delay

Figure 5.12 shows the results for the average energy consumption. Obviously the proposed scheme in Chapter 5 outperforms the MQoSR protocol in term of energy consumption. This is also confirming the energy efficient scheduling mechanism adapted to achieve the required QoS while avoiding the network congestion in an energy efficient way using the related link and path cost function as discussed in Section 5.2.

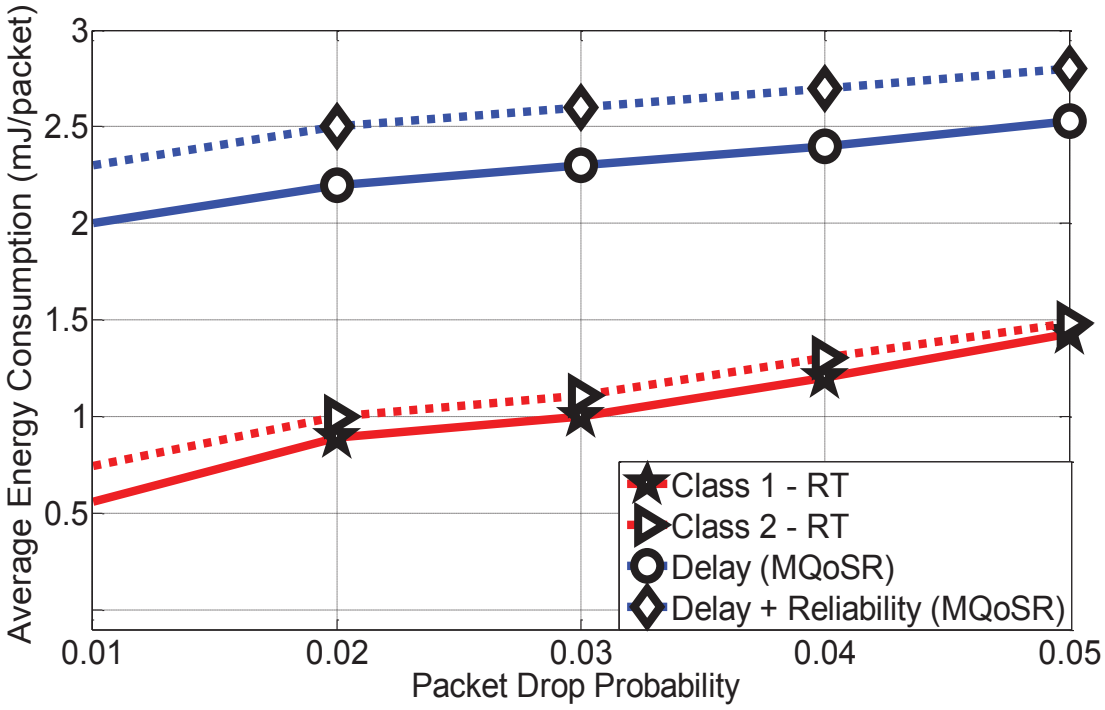


Figure 5.12: Average energy consumption

5.4 ANALYSIS AND SIMULATION RESULTS USING NS-2

We implement the proposed routing protocols for WSNs using an object oriented language C++ which is a common used code in WSNs area. We performed suite of validation tests to verify the fundamental behaviour of these protocols. These validation tests cover the basic functionality of the on demand routing protocol in WSNs such as, node deployment, resources distribution and modification, interferences, node load represented by the buffer size at each node, packets drop due to buffer overflow, different next node selection functions, route setup and withdrawal, and route selection for different route selection criteria using both single and multipath routing mechanism. The simulation scenario is relatively realistic. Nodes are not mobile during transmission, and

multiple independent logical channels are assumed among nodes so that multipath can be deployed independently at the network layer. The code creates nodes randomly in a specified area and resources are initially assigned to each node and are updated during transmission. Nevertheless, to confirm the validity and comparability of our implementation, we implement the protocol using NS-2.35.

Table 5.2 shows the simulation parameters used in our simulation. The default parameters as existed in NS-2.35 for 802.11 MAC has been chosen. We consider four different types of traffic originating from a single node in which two of these traffic are real time traffic and the others are non-real time traffic. The sink node is situated at the upper right corner of the simulation field, and the source node is situated on the left bottom corner.

Table 5.2: Simulation parameters for NS-2

Network field	*200m × 200m*
Number of sensors	*100*
Simulation time	*100 sec*
MAC layer	*IEEE 802.11*
Transmission range	*40 m*
Packet size (data + overhead)	*1024 byte*
No. of source nodes/ No. of sink	*1/1*

Data arrival rate	*10-100 packet/sec*
Size of each queue size	*50 packets*
e_{init}	*100 J*
Transmit power	*15 mw*
Receive power	*13 mw*
Idle power	*12 mw*

We use the same performance metrics as presented in Section 5.3.2 to evaluate the results. The arrival rate of traffic increases from 10 to 100 and in order to focus on the timeliness domain, we use a non-strict reliability requirement of 0.7.

5.4.1 Average End-to-end Delay

Figure 5.13 illustrates the average end-to-end delay per packet for both real time and non-real time traffic. From the results, it is clear that the proposed protocol successfully differentiates service by giving real time traffic (Class 1 and Class2) privileged treatment over low priority traffic (Class 3 and Class 4). Consequently, real time traffic is always combined with low end-to-end delay since it is processed first which causes more queuing delay for non-real time traffic at each sensor node. As mentioned before when traffic rate is high packets are queued waiting to be processed and this waiting time (queuing delay) influences the end-to-end delay as well as increases the interference between adjacent sensor nodes. Therefore, the average delays for all the classes are increased at higher packet rates compared with low packet rates.

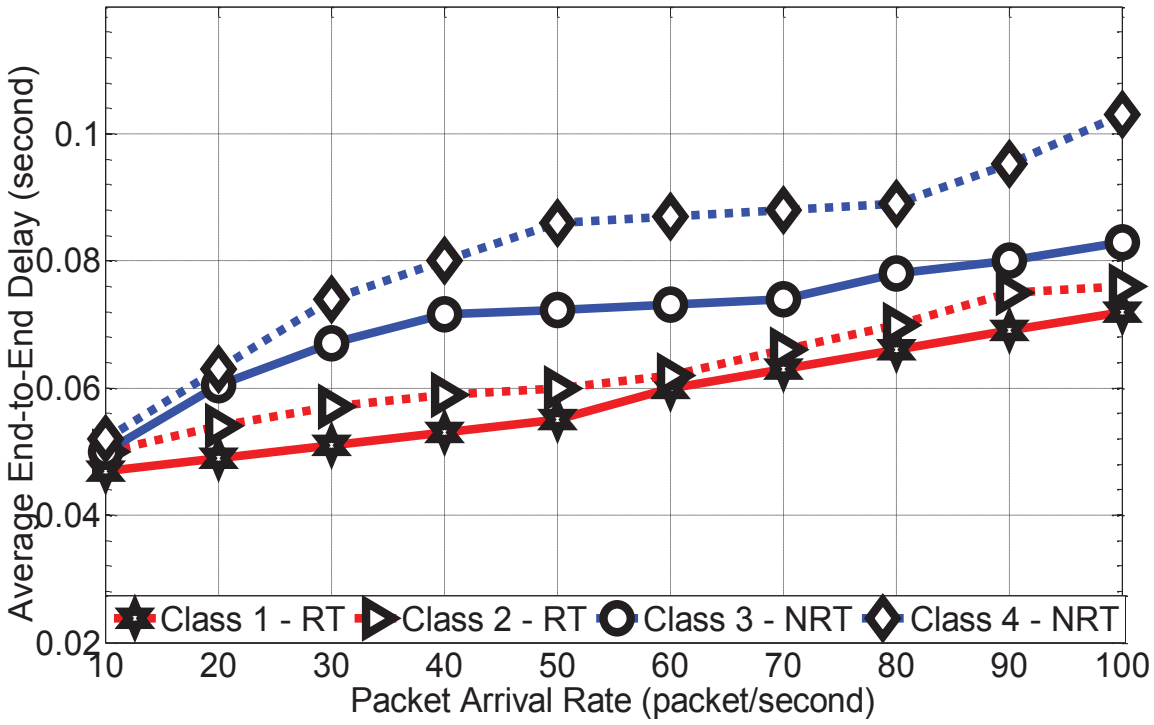


Figure 5.13: Average end-to-end delay

5.4.2 On-Time Reachability

Figure 5.14 illustrates the on-time reachability of packets, which is the probability that a packet achieves the delay requirements. Clearly, the average end-to-end delay for the real time traffic (Class 1 and Class 2) is below the required delay (60ms) up to 60 packet/sec arrival rate (Class 1 = 54ms, Class 2 = 60ms). This means that the number of packets arriving to the sink with end-to-end delay less than or equal 60ms is high. Therefore, we can confirm the results in Figure 5.14 since the probability of reaching the delay requirements for these classes is higher than the other classes (Class1 = 0.95, Class2 = 0.77) and for all the classes these probabilities are proportional to the arrival rate of packets since the higher the rate the more delays packets can suffer.

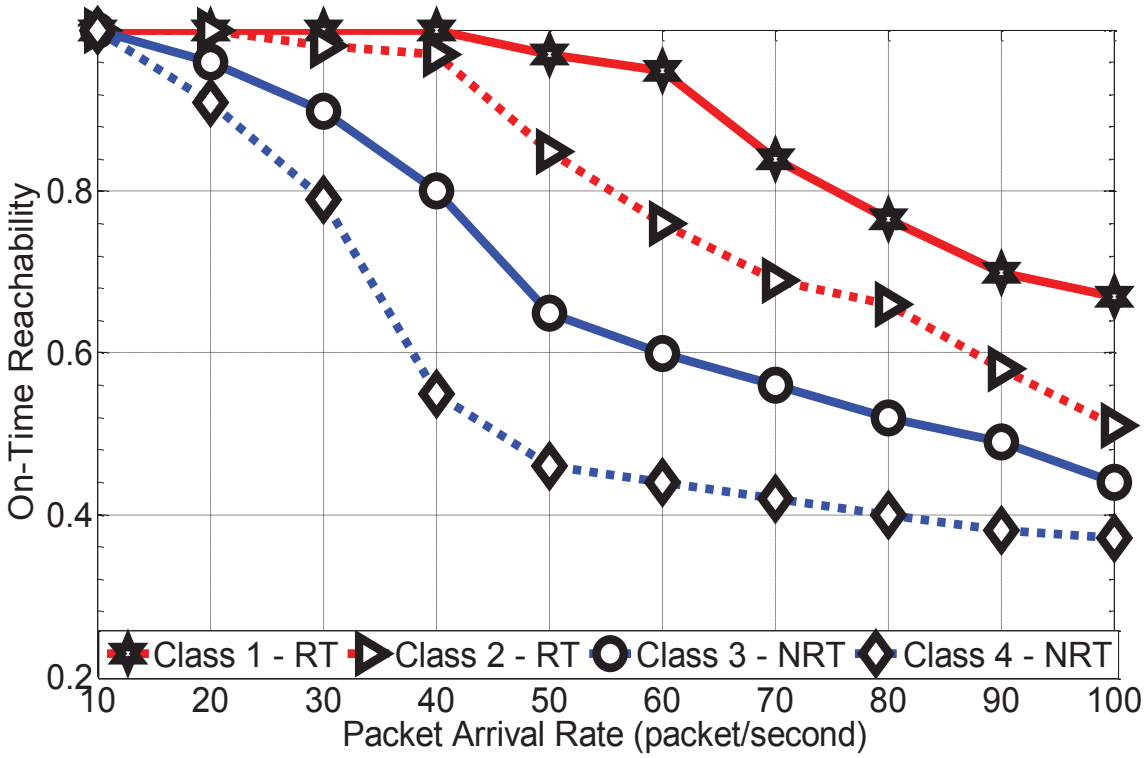


Figure 5.14: On-time reachability

5.4.3 Packet Delivery Ratio

Figure 5.15 shows the average delivery ratio of the proposed protocol. We notice that the mechanism of construction paths considering nodes reliabilities as well as transmitting packets on multiple paths according to the requested reliability while avoiding congestion is very effective mechanism in enhancing the delivery ratio. Class 2 show the highest delivery ratio among the other classes since both reliability and delay are considered in the links and paths selection process. Moreover, since traffic in Class 2 is assigned high priority and processed first, the probability that packets are dropped due to queue overflow is low. On the other hand, the selection process of links and paths in Class 4 primarily depends on the reliability as a metric. But due to the prioritized scheduling mechanism, Class 4 traffic is assigned low priority and packets related to that

class may be dropped due to queue timeout or overflow and this impacts the delivery ratio of Class 4.

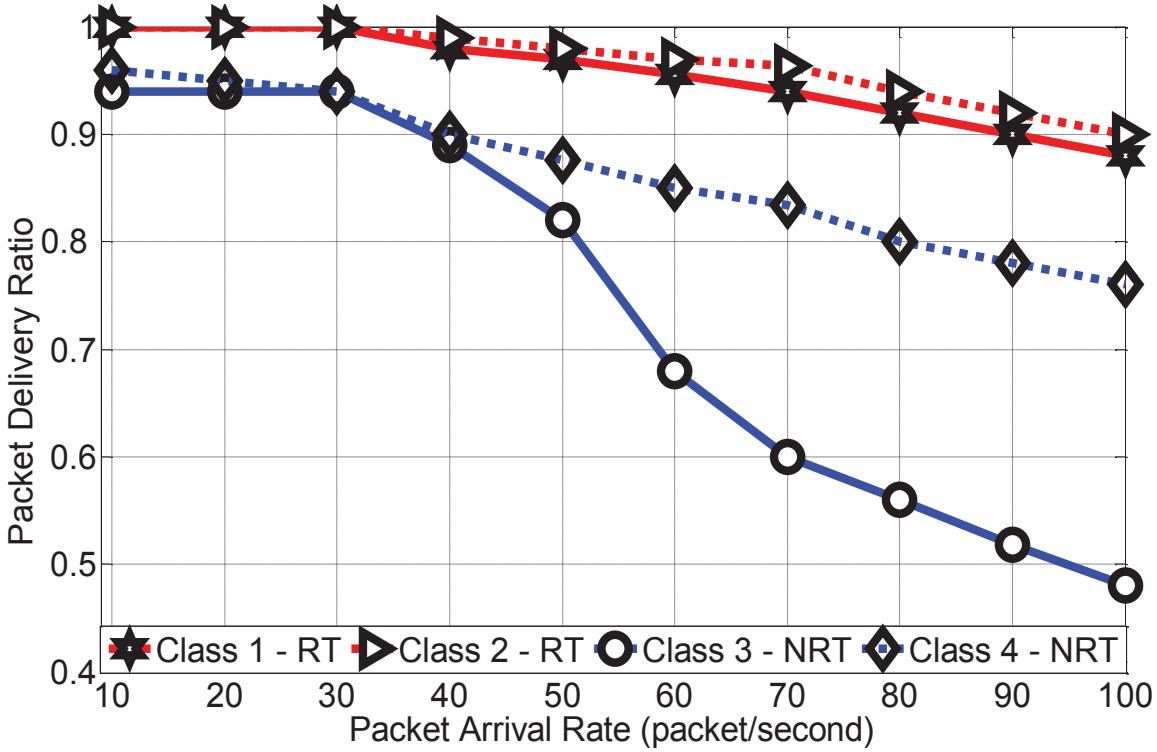


Figure 5.15: Packet delivery ratio

5.4.4 Average Energy Consumption

Figure 5.16 shows how the average energy consumptions of sensor nodes increase with the increase of packet rate. Compared with Figure 5.9 where the results are obtained using C++, we can realize that in Figure 5.16, Class 1 traffic has the least energy consumption among the other classes even when the arrival rate is increased unlike in Figure 5.9 where Class 3 is the least energy consumption among all. Although, the path construction process in Class 1 depends mainly on the energy parameter, Class 1 traffic is assigned low priority and traffic is required to be buffered at high packet rate. This distinction between Figure 5.9 and Figure 5.16 is related to the fact that in implementing

the algorithm in C++, the energy consumption for data waiting at the queue of sensor nodes is not considered which yield in increasing the energy consumption of Class 3. However, it is worth emphasizing that the other classes follow the same behaviours and clarifications that are presented for Figure 5.9. Clearly, the classes that are using single path routing (Class 1 and Class 3) consumed less energy than the classes that are using multiple paths routing (Class 2 and Class 4).

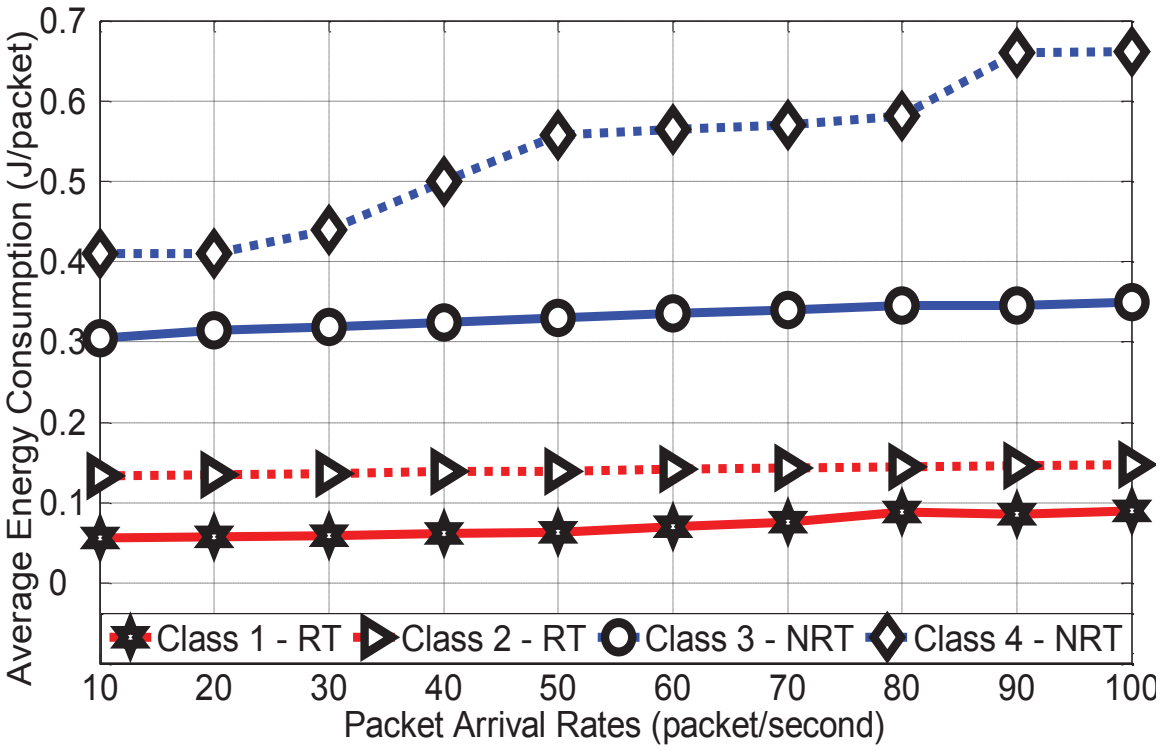


Figure 5.16: Average energy consumption per packet

5.5 SUMMARY

In this chapter, we show that joint optimization across routing and MAC layers which also takes into account the sensor nodes energy constraint is feasible and beneficial. The QoS requirements are enforced through sensors decision of next hops according to the neighbors state. However, the end-to-end requirement is guaranteed jointly by the local decisions of these sensors and the sink decision on the used paths and the number of these paths.

The proposed scheme prioritize traffics according to the requirements into a packet, queue and path: A classifier to check the incoming traffic is used to assign real time and non-real time traffic to different priority queues, a scheduler to handle both queues according to the occupied size and the priority, and at the sink side a priority is assigned to real time traffic on selecting the routes. Moreover, the queue size of each sensor is used as an indicator of node congestion and presented in the link cost function and the path cost function as a metric. In this way the node with the high load has a lower chance to be selected as next hop. Also, by transferring this information to the sink and when the load of traffic on sensors in some area of the network is high due to heavy communication activity, the cost of routing is decreased through this area to protect the traffic from dropping. Extensive simulations using C++ and NS-2.35 are used to evaluate the algorithm and the results have demonstrated the effectiveness of the proposed scheme for different metrics.

Chapter 6

SECURE MULTIPATH QOS ROUTING

WSNs are characterized by severe resource constraints of sensor nodes, unreliable nature of the wireless links, dynamic changing in the size and density of the network, as well as the high risk of physical attacks to sensors.

A secure and reliable multipath routing protocol is presented. The main motivation comes from the observations that most traditional encryption algorithms are complex and may introduce a severe delay in sensor nodes. For instance, the encryption time of each 128-bit block using the AES algorithm is about 1.8 ms on a MicaZ platform [11]. Our approach therefore proposes to encrypt only a certain fraction of the RS codewords while the remaining portion is transmitted unprotected. Our scheme makes encryption feasible for energy constrained and delay sensitive applications while still maintaining a robust security protection.

In this chapter, firstly, a new mechanism for secure and reliable data transmission in WSNs multipath routing derived from node-disjoint multipath is introduced and combined with source coding in order to enhance both security and reliability of data transmission. Using multipath routing, the general security requirements for data transmission in terms of authentication, integrity, freshness, resilience and availability of service are supported as presented in Section 2.4. Secondly, different levels of security requirements are defined and depending on these requirements, a selective encryption scheme is introduced to encrypt selected number of coded fragments in order to enhance security and thereby reduce the time required for encryption. Finally, an allocation

strategy that allocates fragments on paths is introduced to enhance both security and probability of successful data delivery. Security is improved in term of providing confidentiality since the probability of eavesdropping attacks is reduced as the attacker needs to catch the appropriate fragments for each packet over different paths and to decrypt these fragments in order to reconstruct the original packet. Also, we assume that an attacker has no knowledge of the routing protocol strategies and therefore the attacker gets no information about which fragments to compromise over the different paths in order to be able to reconstruct the original message. Therefore, when the attacker tries to attract the traffic of nearby neighbors by making itself look attractive to them, Sinkhole attack, or when two or more attackers establish better communication tunnels between them, Wormhole attack, they cannot get enough fragments to reconstruct the original packet. Moreover, using different paths for different application requirements to route data and permitting the sink to be responsible for the path selection process also eliminate the risk of Sinkhole and Wormhole attacks as each node keeps the information of its one-hop neighors and have no information about the whole routing strategies.

6.1 QOS PROVISIONING

This section presents the QoS parameters used in the proposed scheme and review the analysis models of different strategies to handle secure multipath routing as well as their influence on respecting the WSNs constraints.

6.1.1 Security

A path is compromised when one or more node in the path is compromised. In this paper node-disjoint paths are used, thus we assume that the probability of compromising

of a single path is not correlated with the probability of compromising of other paths. We assume that the source node and the sink are trustworthy. The source node selects *np* paths out of the *n* node-disjoint paths to route the data packet to the sink. The probability that the data packet is compromised, P_{pkt}, is defined as,

$$P_{pkt} = \prod_{j=1}^{np} P_{path_j} \tag{6.1}$$

where P_{path_j} is the probability that $path_j$ is compromised and is given as,

$$p_{path_j} = 1 - \prod_{\delta=1}^{hop_j}(1 - p_{x_\delta}) \tag{6.2}$$

where p_{x_δ} is the probability that sensor node x_δ is compromised, $x_\delta \in hop_j$, hop_j is the number of sensor nodes on *j* and $0 \leq p_{path_j} \leq 1$.

Note that the probability p indicates the security level of a node and could be estimated from the feedback of some security monitoring software or hardware such as firewalls and intrusion detection devices [77]. Additionally, we defined the levels of required security, S_{req}, from the lowest to the highest levels as (1-10^{-1}) to (1-10^{-10}).

The proposed mechanism uses RS coding to send the *M* + *K* fragments on *np* node-disjoint paths. To improve the security of data transmission,

- **Strategy 1:** Allocate fragments on as many paths as possible in order to minimize the probability p_{pkt}. The total number of fragments for each packet is equal to *np*, that is *M* + *K* = *np*. In this case one fragment is transmitted on each path. With such allocation, the probability that the data packet is compromised, P_{pkt}, is equal to the probability that *M* out of *np* paths are compromised, $P_{pkt} = \prod_{i=1}^{M} p_{path_j}$. Thus, the more paths are used, the less P_{pkt} is, and the better the security, Figure 6.1.

However, this strategy could be expensive in resource constraint network like WSNs since it introduces a large storage and communication overhead. Moreover, fragments might be dropped on some paths due to the error prone nature of sensor nodes and wireless links and to reconstruct the original data packet, a minimum of *M* paths are needed to successfully deliver the required number of fragments to the sink.

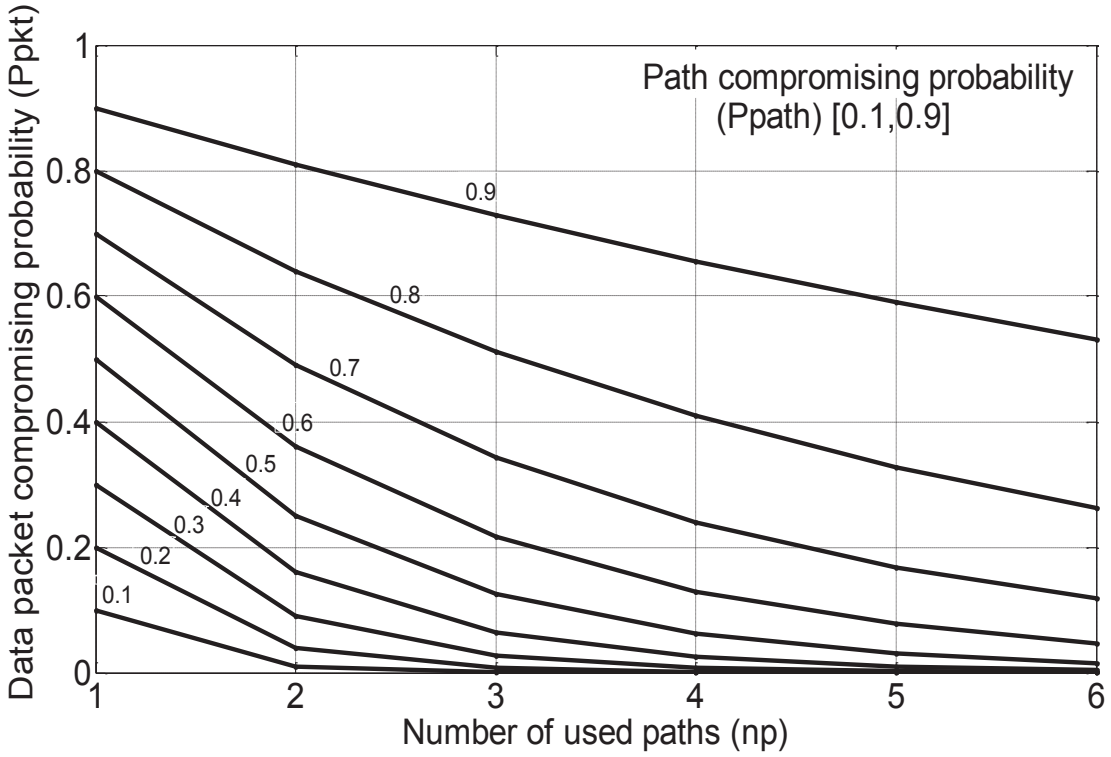


Figure 6.1: Relationship between data packet compromising probability, P_{pkt}, and the number of used paths, *np*, for different path compromising values, p_{path_j} [0.1, 0.9].

- **Strategy 2:** To achieve the highest security level, the allocated fragments on any path, x_j, should be less than *M*. With such allocation an attacker must intercept

more than one path to get the *M* fragments required to reconstruct the data packet. The allocated fragments on each path should be as follows,

$$1 \leq x_j \leq M - 1 \tag{6.3}$$

This strategy is used in the proposed security mechanism.

- **Strategy 3**: Minimize p_{path_j} such that P_{pkt} is minimized, equation (6.1). By using a path that contains as less nodes as possible, the shortest path, and/or path that contains the highest secure nodes among others, minimizes p_{path_j}, equation (6.2).

6.1.2 Reliability

Multipath routing is one way of improving the reliability of data transmission by sending duplicated data via multiple paths. Thus, a packet is delivered to the destination even if some paths fail. The main drawback of the multipath routing is the higher energy consumption and the high probability of network congestion due to the increased number of messages which in turn impact the performance of the network. However, using multipath routing with redundancy and erasure coding, the reliability of data transmission can be improved while respecting the network energy constraint. Similar to the proposed routing mechanism in Chapter 4 and 5, the reliability of data transmission, the successful end-to-end data delivery, is achieved by sending the fragments of RS codeword on *np* selected node-disjoint multipath and to guarantee that the packet is recoverable from any $\lceil np/2 \rceil$ paths, we need to ensure that fragments allocation on any $\lceil np/2 \rceil$ paths follows,

$$\sum_{j=1}^{\lceil np/2 \rceil} x_j \geq M \tag{6.4}$$

6.1.3 Delay

The total path delay, D_{path}, includes the sum of time required for processing, queuing, transmission and propagation for all the nodes along the path. If coding and encryption are used, the path delay equals to (D_{path} + D_{cod} +D_{enc}), where D_{cod} and D_{enc} are the coding time and the encryption time respectively. D_{enc} is related to number of bits to be encrypted, n_{bit}, the unit-block encryption time, T_{blk}, and the encryption block size, L_{blk}, [78]. This is given as follows,

$$D_{enc} = (n_{bit}/L_{blk})\, T_{blk} \tag{6.5}$$

Encryption block size varies between different encryption algorithms and may also vary within the same encryption algorithm while the unit-block encryption time can be measured on specific platforms. Thus, choosing the appropriate block size as well as the total amount of bits to be encrypted can affect the delay performance of the network. Therefore, in the proposed selective encryption approach, a minimum amount of data is selected for encryption contingent to the security requirements. In this way encryption time is reduced due to the need to encrypt fewer packets. Also the energy required to encrypt the extra packets is conserved while still maintaining the required security level.

6.2 PROPOSED SECURITY MECHANISM

The details of the proposed secure routing protocol are discussed in this section.

6.2.1 Initialization Phase

Each sensor node maintains and updates its neighboring table information by broadcasting a HELLO message (Figure 6.2) in which the local states of its one-hop neighbors are reported in terms of the probability that a sensor node is compromised, p.

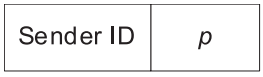


Figure 6.2: HELLO message structure

6.2.2 Path Discovery Phase

As mentioned before, when the source node has data packet to transmit to the sink to which it has no available route, it starts the route discovery phase by transmitting RREQ as shown in Figure 6.3. A RREQ message, is broadcasted to all the neighbors of the source node within its transmission range, in which the required security level (in terms of message compromising probability), S_{req}, as well as, the path information (hop, p_{path}) are transferred to the sink. Each intermediate node updates the information of its one-hop local states, including the path compromising probability and hop count information.

Figure 6.3: RREQ message structure

In order to achieve the shortest hop count from the current node to the sink, we assume that only the neighbors that are closer to the sink than the current node are added to the neighbor list as a candidate node. Since security is the essential metric in choosing

different paths and to maximize the path security, each intermediate node selects one node as the next hop from its neighbor list to forward the RREQ, the neighbor with the highest security among all, smallest p. However, if the selected node is already reserved then the next neighbor with the smallest p will be selected and so on. The selected node then modifies the path information in the RREQ message before forwarding the message to the next selected neighbor. The probability of path compromising, p_{path}, is updated according to equation (6.2) and the value of hop count, *hop*, is increased by one. Note that, the initial values of *hop* and p_{path} at the source node are zero.

6.2.3 Multipath Selection Algorithm

The sink estimates the number of all available node-disjoint paths to the source from the number of the RREQ messages received to decide on choosing the first *np* most secure paths that satisfy the required security level. From these RREQ messages it obtains information about security and number of hops on each path. The sink sends back the RREP (Figure 6.4) through the selected paths. Algorithm 3 is used to determine the number of node-disjoint multipath, *np*, which are used to transmit data message between the source and the sink.

Request ID	Source ID	Sender ID	No. of paths

Figure 6.4: RREP message structure

For each data transmission, given *n* available node-disjoint paths between the source and the sink, the sink sorts these available paths according to the security characteristics of each path (in terms of the probability that path *j* is compromised), such that the first path is the highest secure one and so on. The sink then calculates the probability that a

packet is compromised, P_{pkt}, using equation (6.1). According to equation (6.1) more paths are chosen to lower the P_{pkt} and enhance the security in order to deliver the data packet. The proposed protocol only needs to select the first *np* paths (*np* $\geq$ 2) satisfying $P_{pkt} \leq (1 - S_{req})$.

Algorithm 3: Calculate the number of paths related to the required security level

---

n = number of available node-disjoint paths (source to sink)

Sort for p_{path} *such that* $p_{path_1} < p_{path_2} < \ldots\ldots < p_{path_n}$

np = 1; // Initialization

$P_{pkt_1} = p_{path_1}$ // Calculate the probability of compromising a packet on the first path

for (i = 2; i $\leq$ *n ; i++)*

{

np = np++;

$P_{pkt_i} = P_{pkt_{i-1}} \times p_{path_i}$

if ($P_{pkt_i} \leq (1 - S_{req})$ // If the required security is reached

{

number of paths to be used = np;

break;

}

}

Drop Packet; // When np = n and S_{req} is not achieved packet is dropped

---

6.2.4 Security Mechanism

The following consecutive steps are involved in the routing mechanism to ensure the communication security level and are illustrated in Figure 6.5:

Step 1: Divide the original data message of size *S* into *j* packets each of *M* fragments of size *b* bits. Assume the number of packets is equivalent to the number of paths used to transmit the data, *np*, such that $Mb = \lceil S/np \rceil$. If the last packet is less than *M* fragments, zero padding [67] is applied to meet the length requirements of RS codes.

Step 2: Encode each packet using RS codes to generate *M* data fragments and *K* parity fragments as a codeword of size $M + K$ fragments such that $K \leq M$. For each codeword packet, allocate one fragment on each path starting from the highest secure path and repeat this process till all the $M + K$ fragments are assigned on the selected multipath and ensure that the number of allocated fragments on each path, x_j, follows,

$$x_j = \lceil (M + K) \,/\, np \rceil < M \qquad j=1,2,\ldots,np \tag{6.6}$$

Step 3: Depending on the required security level, the number of fragments to be encrypted, N_{enc}, is calculated as follows,

$$N_{enc} = K + \mu \tag{6.7}$$

where μ is determined according to the required security level and $1 \leq \mu \leq M$.

As shown in Figure 6.5, for a low security requirement, $\mu = 1$, the source node only encrypts any $N_{enc} = K + 1$ of $M + K$ fragments from the codeword. For each codeword, an attacker must receive at least M of the $M + K$ fragments and be able to decrypt the encrypted fragments to restore the codeword. On the other hand, when the required security level is high, then $\mu = M$, which requires to encrypt $N_{enc} = K + M$ fragments for each codeword. In order to compromise the data packet, the attacker must receive and be able to decrypt all M fragments to reconstruct the codeword.

Step 4: Route all the fragments on the *np* node-disjoint paths to the sink with each path carrying x_j fragments according to equation (6.4) and equation (6.6). To enhance security the encrypted fragments from the same codeword are transmitted on different paths.

At the sink side, the encrypted fragments are decrypted first and then all the fragments are decoded to reconstruct the original data packet.

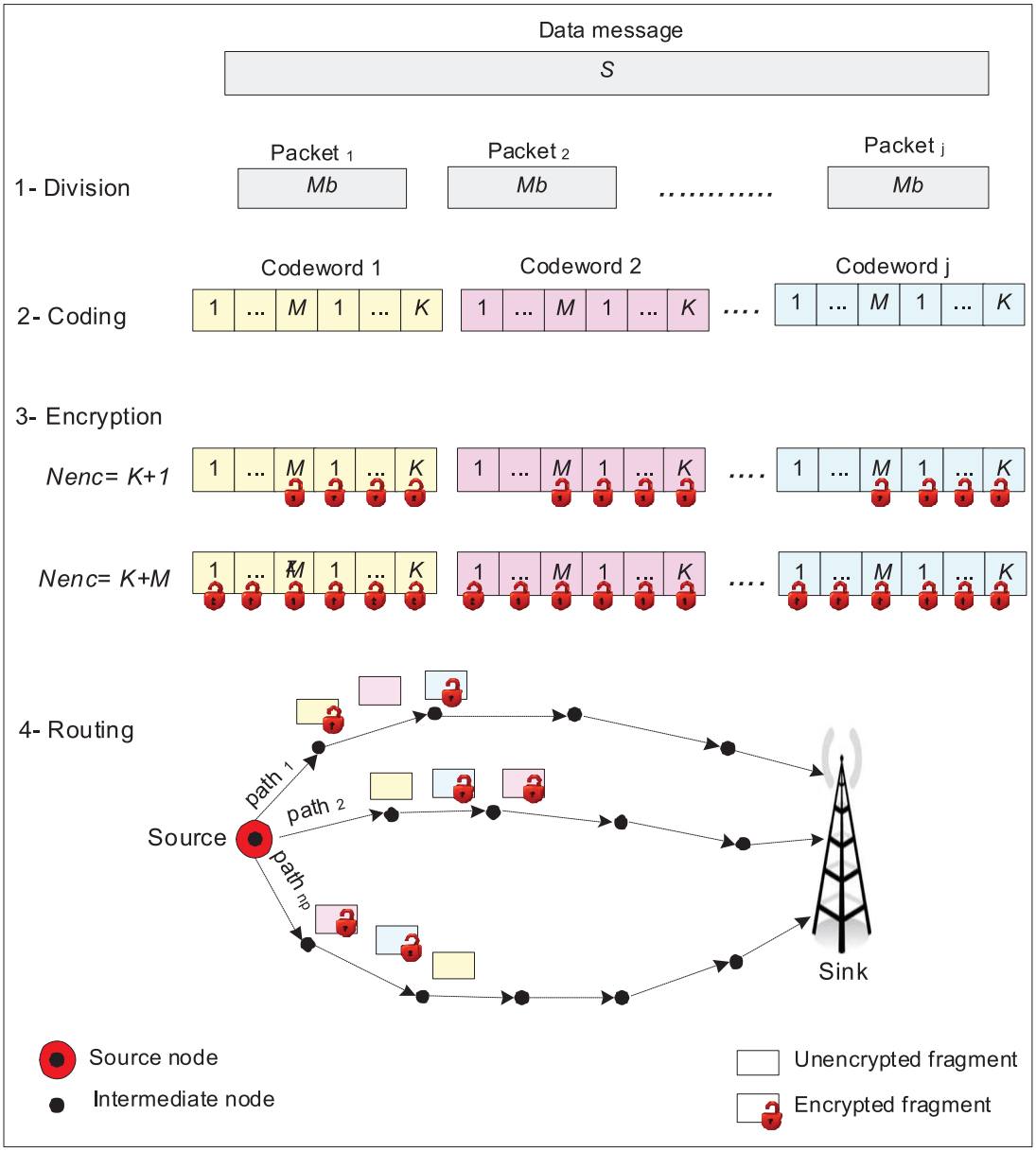


Figure 6.5: Proposed security mechanism.

6.3 ANALYSIS AND SIMULATION RESULTS

In this section, we precisely explain the security and reliability behaviours of the proposed mechanism. For security metric, we describe different scenarios to compromise the data packet and for the reliability metric, we describe the failure models for which we evaluate the resiliency of the proposed mechanism.

6.3.1 Case Study

To help illustrate, we present an example on how the proposed mechanism functions with diverse security levels and attacker scenarios. Suppose we have a 9-byte data message to be transmitted to the sink. Let $np = 3$ and assume using packet-level RS(5, 3) code, where $M = 3$ and $M + K = 5$. Bit-level RS can also be used. The RS codeword packet has the following matrix format,

$$RS\ codeword = \begin{pmatrix} d_{j,1} \\ \vdots \\ d_{j,M} \\ c_{j,1} \\ \vdots \\ c_{j,K} \end{pmatrix}$$

where $d_{j,1} \ldots d_{j,M}$ and $c_{j,1} \ldots c_{j,K}$ are the data and parity fragments for codeword *j,* respectively.

Step 1: Division

For $np = 3$, divide the 9-byte data message to three packets of size 3-byte.

Step 2: Coding

The three packets are coded using RS codes to generate three codewords each of size 5-byte as follows,

$$Codeword\ 1 = \begin{pmatrix} d_{1,1} \\ d_{1,2} \\ d_{1,3} \\ c_{1,1} \\ c_{1,2} \end{pmatrix}, Codeword\ 2 = \begin{pmatrix} d_{2,1} \\ d_{2,2} \\ d_{2,3} \\ c_{2,1} \\ c_{2,2} \end{pmatrix}, Codeword\ 3 = \begin{pmatrix} d_{3,1} \\ d_{3,2} \\ d_{3,3} \\ c_{3,1} \\ c_{3,2} \end{pmatrix}$$

Step 3 & 4: Encryption and Routing

Depending on the required security level, encrypt any N_{enc} fragments, equation (6.7), for each codeword using any encryption algorithm and allocate fragments on *np* paths according to equation (6.4) and equation (6.6).

Scenario 1: For low security requirement, $N_{enc} = K + 1$, $N_{enc} = 3$ fragments.

$$\begin{pmatrix} d_{1,1} \\ d_{1,2} \\ d_{1,3} \\ c_{1,1} \\ c_{1,2} \end{pmatrix} \begin{pmatrix} d_{2,1} \\ d_{2,2} \\ d_{2,3} \\ c_{2,1} \\ c_{2,2} \end{pmatrix} \begin{pmatrix} d_{3,1} \\ d_{3,2} \\ d_{3,3} \\ c_{3,1} \\ c_{3,2} \end{pmatrix}$$

$path_1 = d_{1,1}, c_{1,1}, d_{2,2}, , c_{2,2}, d_{3,3}$

$path_2 = d_{1,2}, c_{1,2},, d_{2,3}, d_{3,1}, c_{3,1}$

$path_3 = d_{1,3}, d_{2,1}, c_{2,1}, d_{3,2}, c_{3,2}$

In this scenario the attacker must intercept at least two paths and decrypt six fragments to get the three codewords.

Scenario 2: For moderate security requirement, $N_{enc} = K + 2$, $N_{enc} = 4$ fragments.

$$\begin{pmatrix} d_{1,1} \\ d_{1,2} \\ d_{1,3} \\ c_{1,1} \\ c_{1,2} \end{pmatrix} \begin{pmatrix} d_{2,1} \\ d_{2,2} \\ d_{23} \\ c_{2,1} \\ c_{2,2} \end{pmatrix} \begin{pmatrix} d_{3,1} \\ d_{3,2} \\ d_{3,3} \\ c_{3,1} \\ c_{3,2} \end{pmatrix}$$

$path_1 = d_{1,1}, c_{1,1}, d_{2,2}, c_{2,2}, d_{3,3}$

$path_2 = d_{1,2}, c_{1,2}, d_{2,3}, d_{3,1}, c_{3,1}$

$path_3 = d_{1,3}, d_{2,1}, c_{2,1}, d_{3,2}, c_{3,2}$

Attacker must intercept at least two paths and decrypt eight fragments to get the three codewords.

Scenario 3: For high security requirement, $N_{enc} = K + M$, $N_{enc} = 5$ fragments.

$$\begin{pmatrix} d_{1,1} \\ d_{1,2} \\ d_{1,3} \\ c_{1,1} \\ c_{1,2} \end{pmatrix} \begin{pmatrix} d_{2,1} \\ d_{2,2} \\ d_{23} \\ c_{2,1} \\ c_{2,2} \end{pmatrix} \begin{pmatrix} d_{3,1} \\ d_{3,2} \\ d_{3,3} \\ c_{3,1} \\ c_{3,2} \end{pmatrix}$$

$path_1 = d_{1,1}, c_{1,1}, d_{2,2}, c_{2,2}, d_{3,3}$

$path_2 = d_{1,2}, c_{1,2}, d_{2,3}, d_{3,1}, c_{3,1}$

$path_3 = d_{1,3}, d_{2,1}, c_{2,1}, d_{3,2}, c_{3,2}$

In this scenario, the attacker needs to intercept at least two paths and be able to encrypt a total of ten fragments to get the three codewords.

For all the above scenarios, an attacker needs to decode each codeword to be able to reconstruct the original data message and the allocation of fragments on the paths, allowing for resilience to a failure of one path, which can be any path, since the three data fragments for each codeword can be obtained from the other two paths.

6.3.2 Multipath Protocols Performance Evaluation and Comparison

In this section we evaluate the proposed mechanism using the same scenario presented in Section 6.3.1 and compare it with the protocols that used the (*k, m)* threshold secret sharing scheme [64, 65] and RS coding technique, MVMP [67]. We present the comparison in Table 6.1 in terms of the total number of transmitted, redundant and encrypted packets as well as the coding redundancy ratio.

Clearly, the number of encrypted packets in MVMP protocol is equal to the encrypted packet of the proposed protocol when the demanded security level is high. However, when the demanded security level is low, the proposed protocol encrypts only three packets while MVMP protocol has a fixed number of fifteen encrypted packets. Note that encrypted packets influence encryption time and energy consumption. We recognize that the encryption delay is related to the total amount of bits to be encrypted for each data packet (Section 3.4). Thus, the proposed security mechanism selects a minimum amount of data for encryption. In WSNs, if sensors run different encryption algorithms, like in MVMP protocol, it may lead to varying computational delays. For instance, the time to execute cipher operations on the Mica2 sensor nodes [79] are: RC5(C) = 0.9 ms, Skipjack(C) = 0.38 ms and RC5(C, assembly) = 0.6 ms. Also in [80], the experiment results show that the encryption process of RC5 algorithm consumes more energy than that of AES on MicaZ platform. Moreover, the proposed security mechanism uses one encryption algorithm while still maintaining a robust security protection unlike MVMP protocol where multiple versions of encryption algorithms are used to maintain the security.

Table 6.1. Multipath routing protocols comparison.

Protocol	*No. of transmitted packets*	*No. of redundant packets*	*No. of encrypted packets*	*Redundancy ratio*
MVMP [67]	$\lceil S/M \rceil \times (M+K) = 15$	$\lceil S/M \rceil \times K = 6$	$\lceil S/M \rceil \times M + K = 15$	$K/(M+K) = 40\%$
Threshold secret sharing scheme	$S \times m = 27$	$(m-1) \times S = 18$	$S \times m = 27$	$(m-1)/m = 66.6\%$
Proposed scheme	$\lceil S/np \rceil \times (M+K) = 15$	$np \times K = 6$	$K + E = [3,15]$	$K/(M+K) = 40\%$

6.3.3 Simulation Setup and Model

We have conducted an extensive simulation study using C++ and MATLAB to evaluate the performance of the proposed mechanism. The validation tests cover the basic functionality of the on demand routing protocol in WSNs. 100 to 500 nodes are randomly scattered in a field of 500m × 500m area. We assume that all sensor nodes are static after deployment with transmission range of 100m. The simulation parameters used are as follows: Source nodes are picked randomly, at least two hops away from the sink, to transmit a data packet at fixed generation rate of 1 packet/sec. The simulation time is 750 sec.

Two types of security scenarios are used in each simulation. In Scenario 1, each node is assumed equally likely to be compromised with probability, $p = 0.14$. In Scenario 2 to evaluate the worst case where the probability that a sensor node is compromised, p, is

changed suddenly at any transmission instant and is randomly distributed as presented in Table 6.2. Simulation results are obtained from different configurations to reduce the effect of the position of sensors. The results shown are averaged over 10 simulation runs.

Table 6.2. Simulation parameters.

Parameters	***Value***
Scenario 1	*100% of nodes,* $p = 0.14$
Scenario 2	*10% of nodes,* $p = 0.50$
	40% of nodes, $p = 0.20$
	50% of nodes, $p = 0.02$
S_{req}	(1-10^{-1}) to (1-10^{-10})
	lowest to highest

6.3.4 Simulation Results

The proposed mechanism depends on the availability of finding multiple node-disjoint paths and to justify the possibility of finding these paths in WSNs, the security requirements are not considered in this step. Figure 6.6 shows the probability of finding the maximal number of node-disjoint paths between the source node and the sink. From the simulation results, the number of paths found in both scenarios is equal. Thus, we only report one result in Figure 6.6, and this indicates that the process of finding the

maximum number of paths depends on the network topology only and not on probability that a sensor node is compromised.

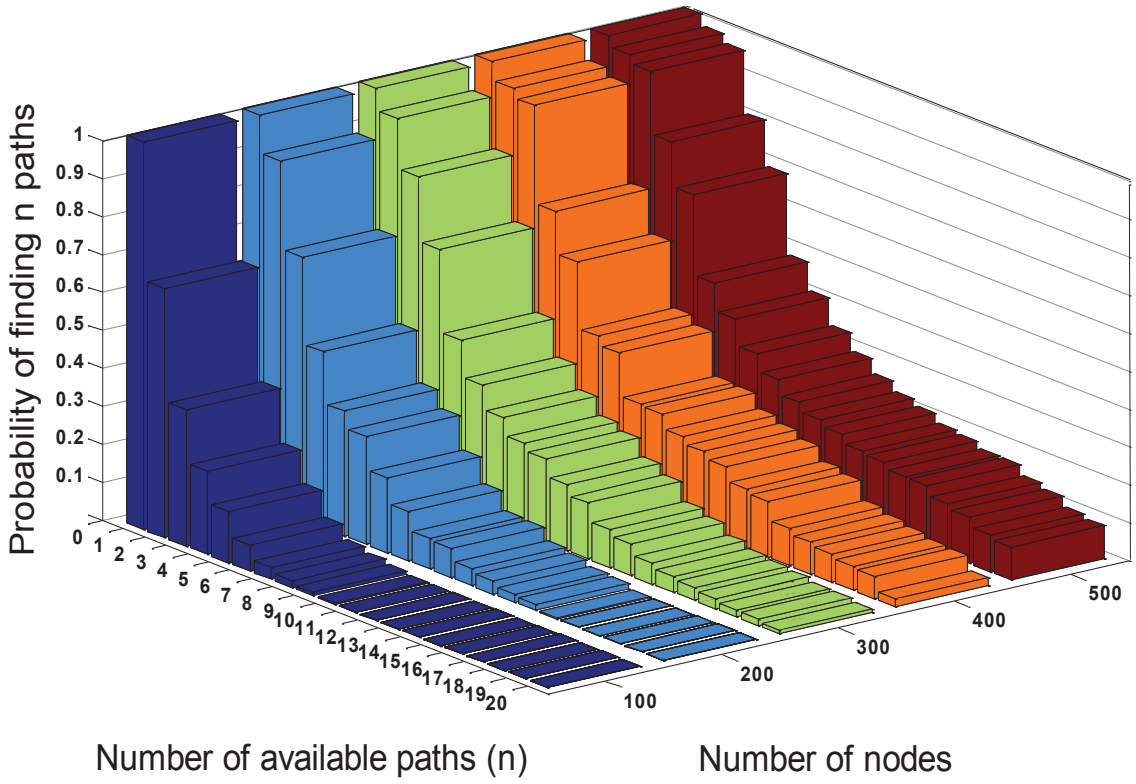


Figure 6.6: Probability of finding *n* node-disjoint paths (Scenario 1/Scenario2)

Figures 6.7 and 6.8 illustrate the security performance and the number of used paths for various network sizes (500 and 300 nodes) as a function of the requested security. A message is compromised when at least *M* fragments are received and N_{enc} fragments are decrypted. It means $\lceil np/2 \rceil$ paths are intercepted out of the *np* used paths. It is clear that the proposed security mechanism is effective in increasing the security performance of a message according to the requested security. The probability that the message is compromised decreases and the number of paths used increases with the increases of the

security requirements. We also observe that when nodes are with different security levels (Scenario 2), the proposed algorithm tends to select more secure paths compared to Scenario 1. However, in both scenarios, the probability that the message is compromised increases as the number of nodes increases. When the number of nodes increases, there are more sensor nodes available for forwarding packets.

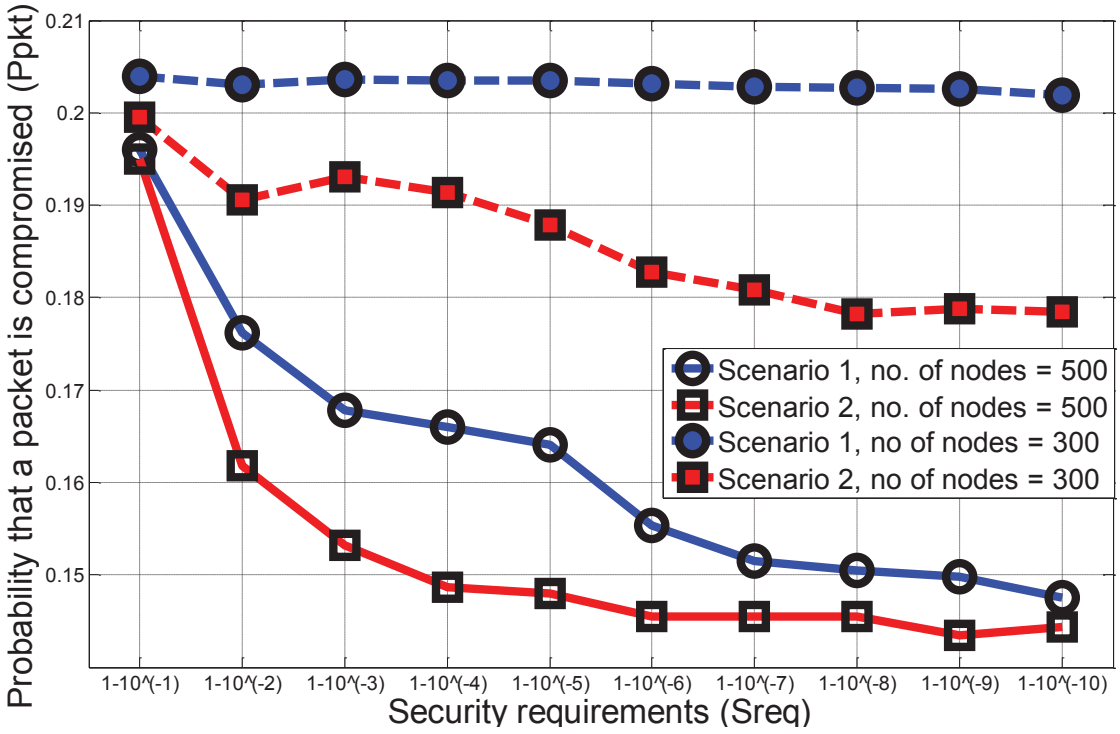


Figure 6.7: Security requirements (S_{req}) vs. packet compromise probability(P_{pkt})

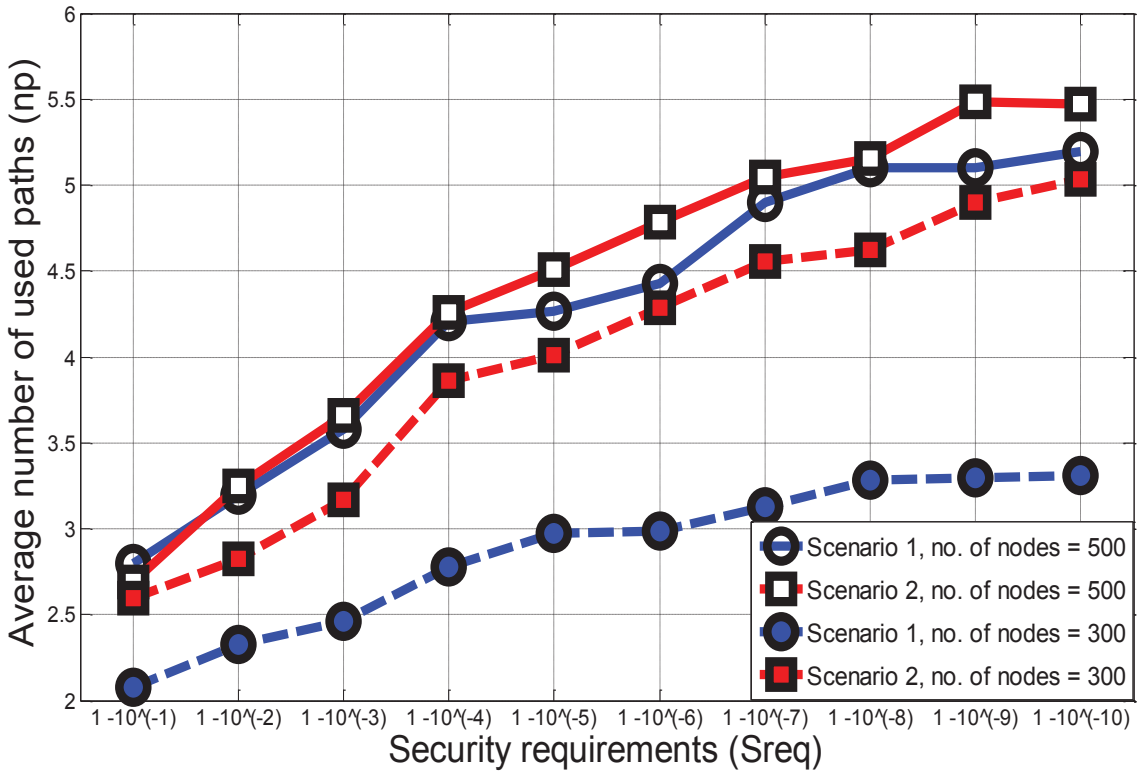


Figure 6.8: Security requirements (S_{req}) vs. average number of used paths (*np*)

In Figure 6.9, the number of encrypted fragments (N_{enc}) for different values of parity fragments ($K = 1, 2, \ldots, K \leq M$) are presented. The data packet is set to $M = 10$ fragments. The number of encrypted fragments used in MVMP mechanism is compared with the lowest and the highest security requirements in the proposed protocol. The other S_{req} values show the same trend (between the two curves), therefore are omitted. In MVMP mechanism all the fragments of the coded packet ($M + K$) are encrypted. Thus, the number of encrypted fragments using MVMP mechanism equals the number of encrypted fragments of the proposed mechanism at the highest security requirements. From the figure, we could observe that when S_{req} is high (S_{req}= 1 - 10^{-10}), the number of encrypted fragments is always 100%. On the other hand, when S_{req} is low (S_{req}= 1 -

10^{-1}), the number of encrypted fragments is related to the size of data packet and the number of added parity fragments. For a data packet of size 10 fragments, when K=1, N_{enc} = 18.18% and when K=10, N_{enc} = 55%. Clearly, the number of encrypted fragments is higher for the highest security requirement to the encrypted fragments of the lowest security; from 81.82% to 45% less fragments are encrypted for the lowest security requirement for K = 1 to 10 respectively. Obviously, when the demanded security level is high, the proposed protocol encrypts $K + M$ fragments similar to MVMP mechanism. However, when the demanded security level is low, $M + 1$ fragments are encrypted. Note that encrypted packets influence encryption time and energy consumption; more encrypted fragments require more time and consumes more energy.

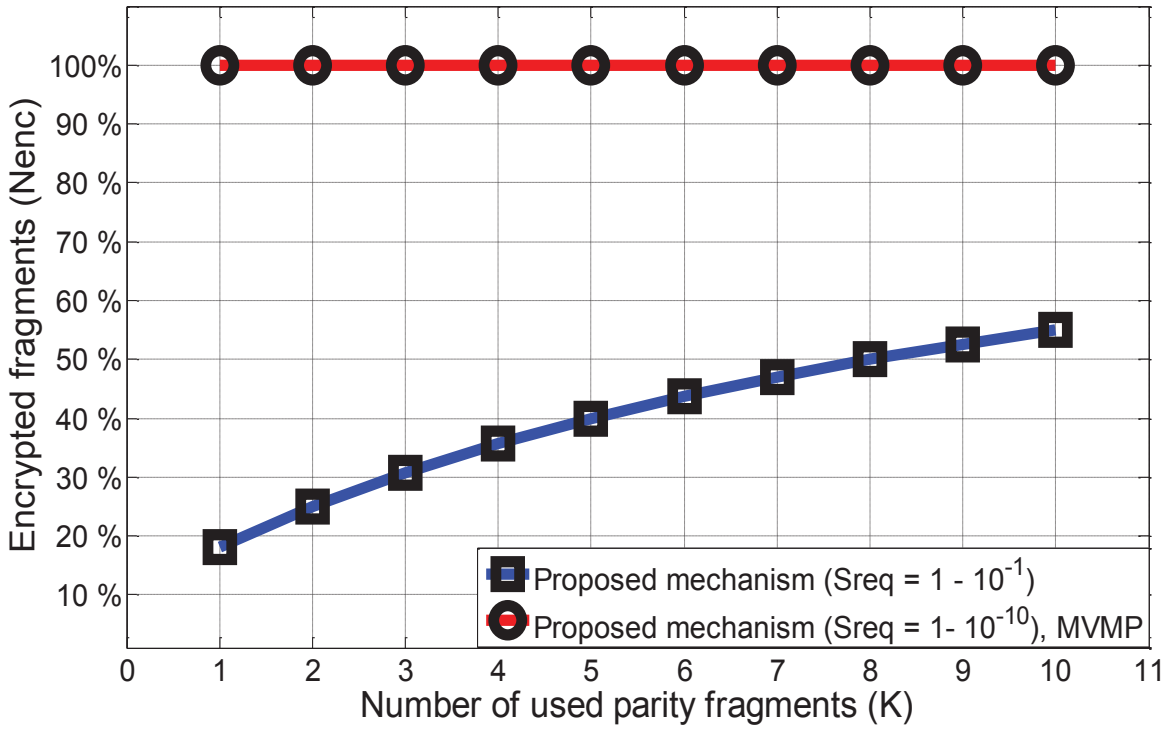


Figure 6.9: Percentage of encrypted fragments (N_{enc}) for a data packet of size M = 10 fragments

6.4 SUMMARY

In this chapter, we propose and evaluate a new secure and reliable routing protocol for WSNs that is designed to handle the application security requirements and reliable data transmission using coding and selective encryption scheme. In the proposed protocol, RS code is used to provide reliability and security. The proposed routing protocol is based on the node-disjoint multipath established depending on the link security parameters. The sink node decides on the paths selection process in order to satisfy the application requirements and the number of these paths is determined to enhance the security. Thus, different number of paths can be used for different security requirements. A novel security mechanism is proposed to support secure data transmission while respecting the network restrictions in terms of energy. The protocol reduces the energy consumption at sensor nodes by moving the path selection process to the sink node. Moreover, reducing the number of encrypted packets based on the required level of security limits energy consumption. Using different paths for different security requirements to route data and permitting the sink to be responsible for the path selection process, attacks such as the Sinkhole and Wormhole are no longer related. Furthermore, using node-disjoint multipath routing, the proposed protocol is protected against selective forwarding attacks [42].

Chapter 7
CONCLUSION AND FUTURE WORK

The overall goal of this research is to solve the conflicts between the requirements and the constraints of WSNs. It will be a key step to take actual WSN applications into reality. We conclude this thesis by summarizing the research discussed in the previous chapters, followed by a section on directions for future research.

7.1 CONCLUSION

In this thesis, we target the major optimization problems that have been proposed to solve the conflicts for years but still exist as major difficulty. We formulated the problem of finding an optimal QoS path as a multi-objective constrained optimization problem to satisfy different QoS requirements. In particular we have

- Proposed a heuristic algorithm for the NP-complete multipath routing constrained problem. A new node-disjoint multi-objective QoS routing protocol is proposed to provide various features like timeliness guarantee, reliability assurance and fault tolerance besides enhanced energy efficiency in WSNs. The required QoS by an application is modeled into seven different classes in terms of the end-to-end delay, reliability and the energy consumption of data transmission. We have shown how to collect the network parameters which can improve the performance of path diversification to provide the required QoS. These parameters are formulated as link-based and path-based cost functions. Each link selects the next hop according to the available resources and the required QoS. However, benefit from the fact that

the sink has unlimited resources, the path selection and the number of paths is assigned to the sink node in which the end-to-end requirements are assured. Single path routing or multipath routing complemented with source coding is used to achieve high level of network reliability and load balancing.

- Proposed a cross-layer design that exploits the characteristics of sensor networks to provide QoS improvement to real time traffic and to provide better service quality in an energy efficient way while avoiding collisions and interference. The MAC layer used in the proposed protocol can distinguish real time traffic and non-real time traffic by deploying IEEE 802.11e which supports service differentiation in the shared channel contention without any extra control overhead in the network layer. Per-hop priority scheduling and QoS consideration of MAC layer is implemented to ensure that real time traffic achieve their desired services. The QoS requirements are enforced through sensors decision of next hops according to the network state. However, the end-to-end requirement is guaranteed jointly by the local decisions of these sensors and the sink decision on the used paths and the number of these paths. Traffic is prioritized according to the requirements into a packet, queue and path scheduling. Real time packets are given higher priority than non-real time packets and placed in the high priority queue where they are scheduled to be served in EDF mechanism. Besides, the real time traffic at the sink side is scheduled first and assigned to path/paths before the non-real time traffic. Moreover, the queue size of each sensor is used as an indicator of node congestion, and presented in the link cost function as a metric. In this way the node with the high load has a lower chance to be selected as next hop. Similarly, by transferring

this information to the sink and when the load of traffics on sensors in some area of the network is high due to heavy communication activity, the cost of routing is decreased through this area to protect the traffic from dropping and to accomplish load balancing in the network.

- Simulation results using C++, NS 2.35 and MATLAB show that the proposed protocols outperform the existing model in the literature remarkably on the basis of factors like average energy consumption, successful data delivery, on-time data delivery, routing overhead, fault tolerance and the probability of packets achieve the end-to-end requested reliability and delay as well as underline the importance of energy efficient solution to enhance network lifetime. It can be concluded that this thesis has a good potential to provide the QoS requirements of applications under the dynamically changing environment of WSNs.
- Finally, we introduced a new mechanism for secure and reliable data transmission in WSNs multipath routing, derived from node-disjoint multipath and combined with source coding in order to enhance both security and reliability of data transmission in the network. Different levels of security requirements are defined and a selective encryption scheme is introduced to encrypt selected number of coded fragments in order to enhance security and thereby reduce the time and energy required for encryption. An allocation strategy that allocates fragments on paths is introduced to enhance both the security and the probability of successful data delivery. Each packet at the source node is divided into fragments using RS codes and these fragments are selectively codded according to the requested security level then transmitted over multiple node-disjoint paths in the network.

Extensive analysis and performance evaluation show that data transmission security and reliability can be enhanced while respecting the resource constraints of WSNs.

7.2 FUTURE WORK

The investigations, performance measurements and analysis work considered so far in this thesis mainly focused on issues at the routing and MAC layers. However, for future research it will be interesting to compute and design an optimal rate allocation and the corresponding channel assignment, with the proposed cross-layer scheduling techniques such that network throughput can be maximized or certain fairness can be achieved.

Additionally, the results of our work in providing secure multipath routing for WSNs may be considered as a solid basis for future research in this field. As future work, we intend to evaluate the proposed mechanism for different routing protocols and under variety of routing attacks as well as to map these protocols to the appropriate applications.

Also, during the course of this thesis the impact of nodes mobility have not been considered. Therefore, it will be interesting to consider and model the impact of nodes mobility on all the proposed routing protocols.

BIBLIOGRAPHY

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, *Wireless Sensor Networks: A survey*, Journal of Computer Network, Elsevier, vol. 38, no. 4, pp. 393-422, 2002.

[2] J. Yick, B. Mukherjee, D. Ghosal, *Wireless Sensor Network Survey*, Journal of Computer Networks, vol. 52, no. 12, pp. 2292-2330, 2008.

[3] D. Chen, P. K. Varshney, *QoS Support in Wireless Sensor Networks: A survey*, International Conference on Wireless Networks, Las Vegas, USA, 2004.

[4] J. H. Chang, L. Tassiulas, *Maximum Lifetime Routing in Wireless Sensor Networks*, IEEE/ACM Transactions on Networking, vol. 12, no. 4, pp. 609-619, 2004.

[5] D. Kandris, M. Tsagkaropoulos, I. Politis, A. Tzes, S. Kotsopoulos, *Energy Efficient and Perceived QoS Aware Video Routing over Wireless Multimedia Sensor Networks*, Journal of Ad Hoc Networks, vol. 9, no. 4, pp. 591-607, 2011.

[6] S. A. Nikolidakis, D. Kandris, D. D. Vergados, C. Douligeris, *Energy Efficient Routing in Wireless Sensor Networks Through Balanced Clustering*, Journal of Algorithms, vol. 6, no. 1, pp. 29-42, 2013.

[7] C. R. Lin, J. S. Liu, *QoS Routing in Ad Hoc Wireless Networks*, IEEE Journals on Selected Areas in Communications, vol. 17, no. 8, pp. 1426-1438, 1999.

[8] P. Zhang, G. Xiao, H. P. Tan, *Clustering Algorithms for Maximizing The Lifetime*

of Wireless Sensor Networks with Energy-Harvesting Sensors, Journal of Computer Networks, vol. 57, no. 14, pp. 2689-2704, 2013.

[9] F. Youssef, D. Merouane, A. Driss, *Maximizing Network Lifetime Through Optimal Power Consumption in Wireless Sensor Networks*, Image and Signal Processing, Springer Berlin Heidelberg, vol. 7340, pp. 200-208, 2012.

[10] K. Donggook, K. Jaesub, P. Kyu Ho, *An Event-Aware MAC Scheduling for Energy Efficient Aggregation in Wireless Sensor Networks*, International Journal of Computer and Telecommunication Networking, vol. 55, no. 1, pp. 225-240, 2011.

[11] A. D. Wood, J. A. Stankovic, *AMSecure: Secure Link-Layer Communication in TinyOS for IEEE 802.15.4-Based Wireless Sensor Networks*, Fourth International Conference of Embedded Networked Sensor systems, pp. 395–396, New York, November 2006.

[12] J. N. Al Karaki, A. E. Kamal, *Routing Techniques in Wireless Sensor Networks: A Survey*, Journal of Wireless Communications, vol. 11, no. 6, pp. 6-28, 2004.

[13] K. Akkaya, M. Younis, *A survey on Routing Protocols for Wireless Sensor Network*, Ad Hoc Network Journal, vol. 3/3, pp. 325-349, 2005.

[14] I. Stojmenovic, X. Lin, *Power Aware Localized Routing in Wireless Networks,* IEEE Transactions on Parallel and Distributed Systems, vol. 12, no. 11, pp. 1122-1133, November 2001.

[15] N. Hoang, V. Son, *Disjoint and braided Multipath Routing for Wireless Sensor*

Networks, International Symposium on Electrical and Electronics Engineering, HCM city, Vietnam, October 11-12, 2005.

[16] C. Intanagonwiwat, R. Govindan, D. Estrin, *Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks*, ACM International Conference on Mobile Computing and Networking, pp. 56-67, Boston, 2000.

[17] D. Ganesan, R. Govindan, S. Shenker, D. Estrin, *Highly- Resilient, Energy-Efficient Multipath Routing in Wireless Sensor Networks*, ACM SIGMOBILE Mobile Computing and Communications, vol. 5, no.4, pp. 11-25, October 2001.

[18] H. Hassanein, J. Luo, *Reliable Energy Aware Routing in Wireless Sensor Networks*, Second IEEE Workshop on Dependability and Security in Sensor Networks and Systems, pp.54-64, Columbia, 2006.

[19] S. Kim, R. Fonseca, D. Culler, *Reliable Transfer on Wireless Sensor Networks*, First IEEE International Conference on Sensor and Ad hoc Communications and Networks, pp. 449-459, California, October 2004.

[20] B. Deb, S. Bhatnagar, B. Nath, *ReInForm: Reliable Information Forwarding Using Multiple Paths in Sensor Networks*, 28th IEEE International Conference on Local Computer Networks, pp. 406-415, Bonn, Germany, October 2003.

[21] S. Dulman, T. Nieberg, J. Wu, P. Havinga, *Trade-off Between Traffic Overhead and Reliability in Multipath Routing for Wireless Sensor Networks*, Journal of Wireless Communication and Networking, vol.3, pp. 1918-1922, New Orleans,

LA, USA, 2003.

[22] P. Djukic, S. Valaee, *Minimum Energy Fault Tolerant Sensor Networks*, Global Telecommunications Conference Workshops, pp. 22-26, 2004.

[23] P. Djukic, S. Valaee, *Maximum Network Lifetime in Fault Tolerant Sensor Networks*, Global Telecommunications Conference, pp. 3101-3106, St. Louis, December 2005.

[24] J. Gehrke, S. Madden, *Query Processing in Sensor Networks*, Journal of Pervasive Computing, vol. 3, no. 1, pp. 46-55, 2004.

[25] H. Alwan, A. Agarwal, *A survey on Fault Tolerant Routing Techniques in Wireless Sensor Networks*, Third International Conference on Sensor Technologies and Applications, Glyfada, Athens, 2009.

[26] I. S. Reed, G. Solomon, *Polynomial Codes over Certain Fnite Felds,* SIAM Journal on Applied Mathematics (SIAP), vol. 8, pp. 300-304, 1960.

[27] A. Shokrollahi, *Raptor codes*, IEEE/ACM Transactions on Networking (TON), Special Issue on Networking and Information Theory, vol. 4, pp. 2551-2567, 2006.

[28] J. W. Byers, M. Lubyt, M. Mitzenmacher, A. Rege, *A Digital Fountain Approach to Reliable Distribution of Bulk Data*, ACM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication, Vancouver, Canada, 1998.

[29] P. Djukic, S. Valaee, *Reliable Packet Transmissions in Multipath Routed Wireless Networks*, Mobile Computing, IEEE Transactions, vol. 5, no. 5, pp. 548-559, 2006.

[30] Z. Xiong, Z. Yang, W. Liu, Z. Feng, *A Lightweight FEC Algorithm for Fault Tolerant Routing in Wireless Sensor Networks*, International Conference on Wireless Communications, Networking and Mobile Computing, pp. 1-4, September 2006.

[31] S. Ali, A. Fakoorian, H. Taheri, *Optimum Reed-Solomon Erasure Coding in Fault Tolerant Sensor Networks*, Fourth International Symposium on Wireless Communication Systems, pp. 6-10, October 2007.

[32] J. Wu, S. Dulman, P. Havinga, T. Nieberg, *Multipath Routing with Erasure Coding for Wireless Sensor Networks*, Fifteenth Annual Workshop on Circuits, Systems and Signal Processing, Veldhoven, Netherlands, November 2004.

[33] ITU Recommendation E.800 (09/08), available at: http://www.itu.int/rec/T-REC-E.800-200809-I/en.

[34] R. Min, M. Bhardwaj, S. Cho, E. Shih, A. Sinha, *Low-Power Wireless Sensor Networks*, Fourteenth International Conference on VLSI Design, pp. 205-210, Bangalore, India, 2001.

[35] Q. Cao, T. Abdelzaher, T. He, J. Stankovic, *Towards Optimal Sleep Scheduling in Sensor Networks for Rare Event Detection*, Fourth International symposium on Information Processing in Sensor Networks, no. 4, Piscataway, NJ, USA, 2005.

[36] K. M. Alzoubi, P. J. Wan, O. Frieder, *Distributed Heuristics for Connected Dominating Sets in Wireless Ad Hoc Networks*, Journal of Communications and Networks, vol. 4, no. 1, pp. 22-29, March 2002.

[37] J. Jeong, T. Hwang, T. He, D. Du, *Mcta: Target Tracking Algorithm Based on Minimal Contour in Wireless Sensor Networks*, 26th Annual IEEE Conference on Computer Communications, pp. 2371-2375, Alaska, USA, 2007.

[38] R. C. Shah, J. M. Rabaey, *Energy Aware Routing for Low Energy Ad hoc Sensor Networks*, IEEE Conference on Wireless Communications and Networking, pp. 350-355, Orlando, Florida USA, March 2002.

[39] B. Yahya, J. Ben-Othman, *Towards a Classification of Energy Aware MAC Protocols for Wireless Sensor Networks*, Journal of Wireless Communications and Mobile Computing, vol. 9, no. 12, pp. 1572-1607, 2009.

[40] D. F. J. Flora, V. Kavitha, M. Muthuselvi, *A Survey on Congestion Control Techniques in Wireless Sensor Networks*, IEEE International Conference on Emerging Trends in Electrical and Computer Technology, pp. 1146-1149, Tamil Nadu, March 2011.

[41] E. Stavroua, A. Pitsillidesa, *A Survey on Secure Multipath Routing Protocols in WSNs*, Journal of Computer Network, vol. 54, no. 13, pp. 2215-2238, September 2010.

[42] C. Karlof, D. Wagner, *Secure Routing in Wireless Sensor Networks: Attacks and*

Countermeasures, First IEEE International Workshop on Sensor Network Protocols and Applications, pp. 113–127, Anchorage, AK, USA, May 2003.

[43] K. Sha, J. Gehlot, R. Greve, *Multipath Routing Techniques in Wireless Sensor Networks: A survey*, Journal of Wireless Personal Communications, vol. 70, no. 2, pp. 807-829, 2013.

[43] A. Shamir, *How to Share a Secret*, Communications of the ACM Magazine, vol. 22, no. 11, pp. 612–613, November 1979.

[44] G. R. Blakley, *Safeguarding Cryptographic Keys*, National Computer Conference, vol. 48, pp. 313-317, 1979.

[45] M. Radi, B. Dezfouli, K. A. Bakar, *Multipath Routing in Wireless Sensor Networks: Survey and Research Challenges*, Journal of Sensors, vol. 12, no. 1, pp. 650-685, January 2012.

[46] K. Sohrab, J. Gao, V. Ailawadh, G. J. Pottie, *Protocols for Self-Organization of a Wireless Sensor Network*, Journal of Personal Communications, vol. 7, no. 5, pp. 16-27, 2000.

[47] K. Akkaya, M. Younis, *An Energy-Aware QoS Routing Protocol for Wireless Sensor Networks*, 23rd International Conference on Distributed Computing Systems, Providence, USA, 2003.

[48] I. Stojmenovic, *Position-Based Routing in Ad Hoc Networks*, IEEE Communications Magazine, vol. 4, no. 7, pp. 128-134, 2002.

[49] D. Chen, P.K. Varshney, *A survey of Void Handling Techniques for Geographic Routing in Wireless Networks*, IEEE Communications Surveys and Tutorials, vol. 9, no. 1, pp. 50-67, 2007.

[50] K. Seada, M. Zuniga, A. Helmy, B. Krishnamachari, *Energy-Efficient Forwarding Strategies for Geographic Routing in Lossy Wireless Sensor Networks*, Second International Conference on Embedded Networked Sensor Systems, Baltimore, USA, pp. 108-121, 2004.

[51] T. He, J. Stankovic, C. Lu, T. Abdelzaher, *SPEED: A stateless Protocol for Real time Communication in Sensor Networks*, 23rd International Conference on Distributed Computing Systems, Providence, USA, 2003.

[52] J. Chen, R. Lin, Y. Li, Y. Sun*, LQER: A Link Quality Estimation Based Routing for Wireless Sensor Networks*, Journal of Sensors, vol. 8, no. 2, pp. 1025-1038, 2008.

[53] B. C. Villaverde, S. Rea, D. Pesch, *Multi-Objective Cross-Layer Algorithm for Routing over Wireless Sensor Networks*, Third International Conference on Sensor Technologies and Applications, Glyfada, Athens, 2009.

[54] X. Huang, Y. Fang, *Multi-Constrained QoS Multipath Routing in Wireless Sensor Networks*, ACM Wireless Networks, vol. 14, no. 4, pp. 465-478, 2008.

[55] A. B. Bagula, K. G. Mazandu, *Energy Constrained Multipath Routing in Wireless Sensor Networks*, Fifth International Conference on Ubiquitous Intelligence and Computing, Berlin, pp. 453-467, 2008.

[56] J. Ben-Othman, B. Yahya, *Energy Efficient and QoS Based Routing Protocol for Wireless Sensor Networks*, Journal of Parallel and Distributed Computing, vol. 70, no. 8, pp. 849-857, 2010.

[57] C. Lu, B. M. Blum, T. F. Abdelzaher, J. A. Stankovic, T. He, *RAP: A Real time Communication Architecture for Large-Scale Wireless Sensor Networks*, IEEE Real time Technology and Application Symposium, San Jose, USA, September 2002.

[58] R. Jurdak, P. Bald, C. V. Lopes, *Adaptive Low Power Listening for Wireless Sensor Networks*, IEEE Transactions on Mobile Computing , vol. 6, no. 8, pp. 988-1004, 2007.

[59] N. Saxena, A. Roy, J. Shin, *QuESt: A QoS-Based Energy Efficient Sensor Routing Protocol*, Journal of Wireless Communications and Mobile Computing, vol. 9, no.3, pp.417-426, 2009.

[60] C. Wang, B. Li, K. Sohraby, M. Daneshmand, Y. Hu, *PCCP: Upstream Congestion Control in Wireless Sensor Networks Through Cross-Layer Optimization*, IEEE Journal on Selected Areas in Communications vol. 25, no. 4, pp. 786-795, May 2007.

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

[62] E. Stavroua, A. Pitsillidesa, *A Survey on Secure Multipath Routing Protocols in WSNs*, Journal of Computer Network, vol. 54, no. 13, pp. 2215-2238, September 2010.

[63] S. K. Singh, M. P. Singh, D. K. Singh, *A Survey on Network Security and Attack Defense Mechanism for Wireless Sensor Networks*, International Journal of Computer Trends and Technology, vol.1, no. 2, pp. 9-17, June 2011.

[64] W. Lou, Y. Kwon, *H-SPREAD: A hybrid Multipath Scheme for Secure and Reliable Data Collection in Wireless Sensor Networks*, IEEE Transactions on Vehicular Technology, vol. 55, no. 4, pp. 1320-1330, July 2006.

[65] W. Lou, W. Liu, Y. Fang, *SPREAD: Enhancing Data Confidentiality in Mobile Ad Hoc Networks*, 23rd Conference of the IEEE Communication Society, pp. 2404-2413, Hong Kong, March 2004.

[66] C. H. Shih, Y. Y. Xu, Y. T. Wang, *Secure and Reliable IPTV Multimedia Transmission Using Forward Error Correction*, International Journal of Digital Multimedia Broadcasting, 8 pages, vol. 2012, 2012.

[67] R. Ma, L. Xing, H. E. Michel, *A New Mechanism for Achieving Secure and Reliable Data Transmission in Wireless Sensor Networks*, IEEE Conference on Technologies for Homeland Security, pp. 274-279, Woburn, May 2007.

[68] L. Chen, J. Leneutre, *On Multipath Routing in Multihop Wireless Networks: Security, Performance and Their Tradeoff*, Journal of Wireless Communication

and Networking, vol. 2009, pp. 1-13, 2009.

[69] Y. Challal, A. Ouadjaout, N. Lasla, M. Bagaa, A. Hadjidj, *Secure and Efficient Disjoint Multipath Construction for Fault Tolerant Routing in Wireless Sensor Networks*, Journal of Network and Computer Applications, vol. 34, no. 4, pp. 1380-139, 2011.

[70] N. Nasser, Y. Chen, *SEEM: Secure and Energy-Efficient Multipath Routing Protocol for Wireless Sensor Networks*, Journal of Computer Communications, Elsevier, vol. 30, no. 11-12, pp. 2401-2412, 2007.

[71] J. Ben-Othman, L. Mokdad , *Enhancing Data Security in Ad Hoc Networks Based on Multipath Routing*, Journal of Parallel and Distributed Computing, vol. 70, no. 3, pp. 309-316, March 2010.

[72] A. M. Popescu, G. I Tudorache, B. Peng, A. H. Kemp, *Surveying position based routing protocols for wireless sensor and Ad hoc networks*, International Journal of Communication Networks and Information Security, vol. 4, no. 1, pp. 41-67, 2012.

[73] H. Takagi, L. Kleinrock, *Optimal Transmission Ranges for Randomly Distributed Packet Radio Terminals*, IEEE Transactions on Communications, vol. 32, no. 3, pp. 246-257, 1984.

[74] D. Chen, P. K. Varshney, *A Survey of Void Handling Techniques for Geographic Routing in Wireless Networks*, IEEE Communications Surveys and Tutorials, vol. 9, no. 1, pp. 50-67, 2007.

[75] T. Houngbadji, S. Pierre, *QoSNET: An Integrated QoS Network for Routing Protocols in Large Scale Wireless Sensor Networks*, Journal of Computer Communications, vol. 33, no. 11, pp. 1334-1342, 2010.

[76] M. Aykut, O. Yigitel, D. Incel, C. Ersoy, *QoS-Aware MAC Protocols for Wireless Sensor Networks: A survey*, Journal of Computer Networks, vol. 55, no. 8, pp. 1982-2004, 2011.

[77] M. Al Ameen, J. Liu, K. Kwak, *Security and Privacy Issues in Wireless Sensor Networks for Healthcare Applications*, Journal of Medical Systems, vol. 36, no. 1, pp. 93-101, 2012.

[78] W. Wang, D. Peng, H. Wang, H. Sharif, *An Adaptive Approach for Image Encryption and Secure Transmission over Multirate Wireless Sensor Networks*, Journal of Wireless Communications and Mobile Computing, vol. 9, no. 3, pp. 383-393, March 2009.

[79] C. Karlof, N. Sastry, D. Wagner, *TinySec: A Link Layer Security Architecture for Wireless Sensor Networks*, Second ACM Conference on Embedded Networked Sensor Systems, pp. 162-175, Baltimore, MD, 2004.

[80] H. Wang, M. Hempel, D. Peng, W. Wang, H. Sharif, H. Chen, *Index-Based Selective Audio Encryption for Wireless Multimedia Sensor Networks*, IEEE Transaction of Multimedia, vol. 12, no. 3, pp. 215-223, April 2010.