2009

Enabling reliable and power efficient real-time multimedia delivery over wireless sensor networks

Tarik El-Amsy

University of Windsor

Enabling Reliable and Power Efficient Real-Time Multimedia Delivery over Wireless Sensor Networks

By

TARIK EL-AMSY

A Thesis
Submitted to the Faculty of Graduate Studies
through Computer Science
in Partial Fulfillment of the Requirements for
The Degree of Master of Science at the
University of Windsor

Windsor, Ontario, Canada

Library and Archives
Canada

Published Heritage
Branch

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque et
Archives Canada

Direction du
Patrimoine de l'édition

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence
ISBN: 978-0-494-57629-8
Our file Notre référence
ISBN: 978-0-494-57629-8

DECLARATION OF PREVIOUS PUBLICATION

This thesis includes one original paper that has been previously published in peer reviewed conference proceedings, as follows:

Thesis Chapter	Publication title/full citation	Publication status
Part of Chapter 4 and Chapter 5	*T. Elamsy, and R. El-Marakby. "Flooding Zone Initialization Protocol (FZIP): Enabling efficient multimedia diffusion for multi-hop wireless networks". IEEE Symposium on Computers and Communications (ISCC08). pp: 1056–1061, 2008.*	published

I certify that I have obtained a written permission from the copyright owner(s) to include the above published material(s) in my thesis. I certify that the above material describes work completed during my registration as graduate student at the University of Windsor.

I declare that, to the best of my knowledge, my thesis does not infringe upon anyone's copyright nor violate any proprietary rights and that any ideas, techniques, quotations, or any other material from the work of other people included in my thesis, published or otherwise, are fully acknowledged in accordance with the standard referencing practices. Furthermore, to the extent that I have included copyrighted material that surpasses the bounds of fair dealing within the meaning of the Canada Copyright Act, I certify that I have obtained a written permission from the copyright owner(s) to include such material(s) in my thesis.

I declare that this is a true copy of my thesis, including any final revisions, as approved by my thesis committee and the Graduate Studies office, and that this thesis has not been submitted for a higher degree to any other University or Institution.

ABSTRACT

There is an increasing need to run real-time multimedia applications, e.g. battle field and border surveillance, over Wireless Sensor Networks (WSNs). In WSNs, packet delivery exhibits high packet loss rate due to congestion, wireless channel high bit error rate, route failure, signal attenuation, etc... Flooding conventional packets over all sensors redundantly provides reliable delivery. However, flooding real-time multimedia packets is energy inefficient for power limited sensors and causes severe contentions affecting reliable delivery.

We propose the Flooding Zone Initialization Protocol (FZIP) to enhance reliability and reduce power consumption of real-time multimedia flooding in WSNs. FZIP is a setup protocol which constrains flooding within a small subset of intermediate nodes called Flooding Zone (FZ). Also, we propose the Flooding Zone Control Protocol (FZCP) which monitors the session quality and dynamically changes the FZ size to adapt to current network state, thus providing a tradeoff of good quality and less power consumption.

DEDICATION

To my beloved parents, wife, and kids

ACKNOWLEDGMENTS

After ALLAH's guidance, mercy, and blessing, there are a number of people without whom this thesis might not have been completed and to whom I am greatly thankful.

First and foremost, I would like to thank my supervisor, Dr. Randa El-Marakby, for her endless support and encouragement. I appreciate her continuous advices and constructive criticism.

Next, I would like to thank my parents and my beloved wife Rana. Without their patience and encouragement, this work would not come to light.

I am also grateful to my committee readers, Dr. A. Asfour and Dr. B. Boufama, for their effort reading and commenting on this thesis. Their comments and recommendations helped improving my thesis quality.

Finally, I would like to thank Nizar, Fadi, and Anas, my friends and colleagues at the University of Windsor. Their support, discussions, and advice have always helped me to complete this thesis. Thank you all.

To each of the above, I extend my genuine appreciation.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1. INTRODUCTION

Wireless Sensor Network (WSNs), are emerging technologies and have been receiving a large attention in the last decade. A typical wireless sensor network consists of large number of small size, low cost, power scarce, and self-organizing sensor nodes which are often densely deployed in the area to be monitored [ASSC02][RSZ04] [SACL03]. Habitat monitoring, fire and earth quack detection are only some WSNs applications. However, the extremely constrained resources of sensors, non-rechargeable batteries, and the large scale deployment have influenced the researchers to seek several scalable and energy efficient data communication protocols.

More recently, advances in miniaturizing multimedia hardware (e.g. tiny CMOS cameras and microphones), have encouraged the integration of sensors nodes with multimedia capabilities [MRX08] [AMC07]. These advances have enabled new multimedia WSN's applications such as traffic control, health monitoring, and security surveillance for military battlefield. However, in order for these envisioned applications to be successful, multimedia WSNs necessitate efficient and scalable real-time multimedia protocols in terms of timeliness, reliability, as well as power efficiency.

This chapter gives a brief overview of real-time multimedia applications general requirements. In more details, it highlights the challenges and requirements in providing efficient real-time multimedia delivery over multi-hop WSNs. Then, we present our research statement and the proposed solution. Finally, thesis organization is presented.

1.1 Real-time Multimedia Applications Requirements

Real-time multimedia applications involve transporting audio/video digital data which are massive in size and require high transmission bandwidth. A typical multimedia session includes the transmission of continuous, enormous size, and great number of audio/video packets to a receiving application. In addition, real-time multimedia applications are delay-sensitive and packet loss-tolerant. This strict requirement differs from elastic application requirements such as Web, e-mail, or FTP, where long delay is annoying but not principally harmful while packet-loss free delivery is of critical importance.

Timeliness consideration is critical for delay sensitive real-time applications as packets incurring a delay more than few hundred milliseconds are useless for some applications. For example, an interactive voice over IP conversation can tolerate up to 400 ms delay [KR03]. Any additional latency will result in users complaining about the quality. On the other hand, real-time multimedia applications are loss-tolerant. That is to say, infrequent loss produces only insignificant glitches while playing the incoming audio/video flow. In addition, these losses can be corrected, partially or fully, by some loss anticipation mechanisms such as Forward Error Correction (FEC) or interleaving [KR03]. However, there is a limit of packet loss which can be tolerated by the application. In summary, for preserving good audio and video quality, a real-time multimedia protocol should provide a service such that delay is minimal and packet loss is not very large.

1.2 Challenges of real-time multimedia delivery in WSNs

These special characteristics and constrains of real-time multimedia bring challenges and new Quality of Service (QoS) requirements to the wired network (e.g. Internet). Several network software, architectures, algorithms, and protocols have been proposed to overcome this problem. However, when it comes to WSNs, providing a similar service for multimedia applications is more challenging. In a wired network, wired links have high and steady bandwidth rate. Packet losses due to transmission errors are rare and often negligible. The main source of packet loss in wired networks, which degrades the quality of a multimedia session, is router congestion. A congested router imposes significant queuing delay on outgoing packets which increases delay and jitter[1] (delay variation). In addition, if the congested router's buffer gets full, arriving packets are dropped and not delivered to the receiving application. This fact has drawn the

[1] The maximum difference between the delay of the transmission of two sequential packets for a period of time [Sye08].

attention of researchers to solve the problem of real-time multimedia, especially in wired network, by solving the congestion problem. Two new main approaches are proposed: Differentiated Service (DiffServ) and Integrated Services (IntServ) [KR03].

However, the problem of transporting real-time multimedia streaming in WSNs is more challenging. The harsh and error prone nature of a wireless network links incur severe transmission errors and fluctuating bandwidth rates. The quality of the wireless link in sensor networks is highly variable and unpredictable. The degradation of audio/ video quality can be caused not only by late delivery due to congestion, but also by actual packet loss over unreliable wireless links before reaching the final destination. Furthermore, sensor nodes include limited and often irreplaceable power sources. Thus, an efficient real-time multimedia session mandates optimized energy spending to prolong overall network life time. In the following subsections, real-time multimedia communication challenges from the prospective of WSNs are discussed in more details.

1.2.1 Timeliness Requirement (End-To-End Delay)

Different than elastic applications, multimedia applications are delay-sensitive and require stringent end-to-end latency. Thus, each packet has to reach the destination within the specified deadline, after which it becomes useless (i.e. considered lost). As many applications in WSNs need to be deployed in rural areas (e.g. fire detection, border surveillance), WSNs are envisioned to support remote user monitoring by linking the sink node to the Internet [ASSC02]. In such situation, the real-time multimedia application requires further strict latency which spans the WSN network topology as well as the Internet. The main sources of delay for real-time multimedia transportation in WSNs are summarized in the following points:

Congestion

Multimedia traffic is massive in size and requires high bandwidth while sensor nodes are scarce in memory and bandwidth resources. Sensor nodes which are located close to the base station (Sink) receive more traffic, coming from relatively farther nodes, and thus can easily get congested. Congestion causes late packet delivery due to queuing delay or entire packet loss when routing buffer is full [KR03].

A solution to this problem is the use of Quality of Service (QoS) aware routing protocol [SKK08] [WL06]. QoS aware routing can distinguish between different traffic priorities and thus can provide better service for high priority traffic. QoS is a complex task and requires efficient scheduling policies with resource limited sensor nodes. Furthermore, the shared wireless link and the density of sensor nodes within a particular area make the quality and the bandwidth rate of the wireless link unpredictable. Thus, QoS mechanisms can't honor a guaranteed service.

Routing

Multi-hop wireless networks, in general, have self-organizing capabilities where each node works as a sensor and a router. Multi-hop wireless networks topology is dynamic and mandate frequent routing links update and new path exploration processes. A link between two neighboring nodes could be down due to node failure (i.e. jamming or running out of power), interference, or node movement. Moreover, new nodes can be deployed frequently to replace nodes which run out of power[ASSC02]. Therefore, wireless nodes execute distributed routing algorithms to discover routing paths continuously. Transferring data between nodes over multi-hops wireless networks has a store and forward nature, which adds extra processing and queuing delay as the number of hops of the path is increased. A typical flat WSN deployment consists of thousands of sensor nodes. Transmission over the shortest path (i.e. least number of hops between source and destination) is desirable not only because it reduces latency but also it reduces total energy depletion. Nevertheless, in the sense of timeliness that is not true always. For example, when a forwarding node becomes congested due to serving many sessions, end-to-end delay may vary and there would be no guarantee of timely delivery. In such case, a relatively longer but less busy path can deliver packets faster than the shortest path. Detecting such case in a self-organizing and dynamic nature of WSNs is complex and the associated overhead might be too expensive in terms of time and power.

Another important source of delay in multi-hop wireless networks is the miss-interpretation of temporary fault transmission as a route failure. Wireless transmission can fail due to several factors such as interference, collision, or hidden terminal problem [EE06]. A temporary transmission failure due to interference can be miss-interpreted by the routing protocol as a path failure [ZH06]. In such case, the routing protocol blocks transmission, starts a path exploration process to find a new route, then enables transmission again. Such long process increases delay which can lead to packet loss due to late arrival. In addition, the energy consumption overhead of such process is high.

1.2.2 Reliability Requirement (Packet Loss ratio)

Multimedia applications can tolerate packet losses up to a certain level [KR03]. Hence, they come under the category of soft real-time multimedia applications. In that sense, reliability for a real-time multimedia protocol in WSNs doesn't mandate or guarantee a 100% packet delivery ratio. Reliability, in this thesis, is argued as the protocol ability to provide adequate level of packet delivery ratio required by the application at the destination node. That is to say, a reliable multimedia protocol doesn't necessitate receiving all packets, rather it has to receive certain amount of multimedia data required by the application.

Packet loss in wired networks is caused mainly by congested routers which discard packets when the routing buffer gets full, thus the packet never arrives at the receiving node. Packet loss due to transmission error is rare and uncommon in wired network links. In contrast to wired links, a wireless link usually has high transmission error rate caused by shadowing, fading, path loss, interference from other transmitting nodes, or hidden terminal problem[2]. An end-to-end path in WSNs has an even higher error rate since it is the concatenation of multiple wireless links. The error rate grows exponentially as the number of hops between source and destination increases [WCK02]. To better explain this problem, assume that e is the error probability a packet gets during transmission, then the chances of the packet to be delivered to next hop without errors

[2] The hidden terminal problem occurs when two nodes, out of the communication vicinity of each other, try to communicate simultaneously with a third node in their communication ranges.

is $(1 - e)$. Clearly, the chance of successful delivery for n hops distance will decrease to $(1 - e)^n$. This problem necessitated the use of different reliability mechanisms at the MAC and the Network Layers (e.g. ARQ and CTS/RTS/ACK) [SH03]. These methods depend on negotiating before transmission and acknowledgment of reception and the process can be repeated several times before giving up. Although these reliability methods reduce the chance of packet loss caused by transmission errors, it can be too expensive in terms of latency for real-time multimedia especially in large WSNs deployments (i.e. packet loss due to late arrival).

Some researchers have proposed redundant transmission to provide reliability. By transmitting the same packet more than one time repeatedly, a packet has more chance to arrive error free at the destination [DBN03]. In such situations, ARQ and other reliability mechanisms can be avoided. However, the disadvantage in such approach is the increased delay and bandwidth requirements and thus might not be practical for real-time multimedia delivery.

1.2.3 Power efficiency Requirement

As sensor nodes have limited power supply which often can't be recharged, power efficiency is a major requirement when designing a communication protocol for WSNs. Since multimedia applications include transmission of huge number of packets, sensor nodes can get drained out of power quickly, thus interrupting the communication. In addition, as the reduction in processor size and cost has outpaced that of a battery, energy constraint is a dominant factor for system design choices in WSNs.

The goal of a power efficient communication protocol is to extend the lifetime of the deployed WSN. This can be achieved by optimizing wireless communication (i.e. reducing the number of wireless transmission to save power) and load balancing traffic among all nodes as possible.

It is important to realize that wireless communication subsystem of a sensor node consumes relatively more energy than other subsystems (e.g. sensing, data processing). For this, efficient communication protocols in WSNs tend to avoid wireless transmission as much as possible. The example illustrated in [PK00] effectively exemplifies this disparity where the energy required to transmit 1 KB over a distance of 100 meters is comparable to the energy required to execute 3 million instructions by a 100 (MIPS) processor. Hence, local data processing before relaying data packets to next hop is an effective way to minimizing power consumption in WSNs [ASSC02]. For example, in a fire detection WSN application, when an intermediate node receives multiple temperature reading data packets, it will be more power efficient to relay a single data packet containing the average temperature instead of relaying each data packet individually. In addition, load balancing communication among the different nodes extends the overall lifetime of the WSNs. That is to say, unbalanced utilization in WSNs segments the network into isolated islands of sensor nodes.

1.3 Problem Description and Proposed Solution

The success of real-time multimedia applications in WSNs necessitates low packet loss and lower power consumption. In a wired network, a real-time multimedia session can experience bad quality (i.e. high packet loss rate) because of congestion. However, packet loss, as discussed in the previous section, is more often in WSNs and occurs due to diverse reasons other than congestion. That is to say, packet loss can be due to:

- Wireless channel characteristics including shadowing, collision, fading, interference, channel contention, and hidden terminal problem.
- Dynamic nature of WSN topology, which leads to routing path failure caused by node mobility or node power failure. This results in interruption imposed by new path exploration and recovery processes.
- Congestion due to sensor nodes' limited routing queue and massive multimedia data size.
- Wireless sensors' limited bandwidth and resources.

An intuitive way of solving the packet loss problem in WSNs is the use of flooding protocol [ASSC02, AK04]. In flooding, several copies of the same packet are transferred by all nodes over all paths to the destination node redundantly. In this way, if a packet is lost for any reason (e.g. congestion, fading, interference, etc...), another copy or copies will have the chance to reach the destination in time over different path(s). However, flooding has two major drawbacks when used for real-time multimedia data in WSNs. First, flooding of multimedia packets over all nodes consumes huge amount of power which causes sensor nodes (which are limited in power) to drain out of power quickly. Moreover, while flooding works fine for classical data sensors (e.g. temperature reading) which are small in size and intermitted (i.e. send data periodically and non continuous), the participation of all nodes in flooding continuous multimedia data can lead to severe broadcast collision and channel contention which degrades overall reliability and scalability.

To solve the problem of huge power consumption and reduce packet collision of flooding real-time multimedia packets to the whole WSN, we introduce the flooding zone concept. Instead of flooding over all network nodes, the flooding-storm can be constrained by initializing a small set of interconnected nodes between source and destination called "flooding Zone". The initialized flooding zone acts as a subset which constrains the flooding-storm propagation inside it. As a result, flooding can achieve not only reduced power consumption but also reduced packet loss rate in comparison with overall flooding. Consequently, we propose the Flooding Zone Initialization Protocol (FZIP) [EE08]. FZIP is developed to enable reliable and power efficient real-time multimedia flooding in a WSN environment. FZIP can be used by multimedia flooding protocols (e.g. Video diffusion) to initialize a suitable Flooding Zone (FZ) before starting flooding real-time multimedia data. Only initialized nodes (i.e. FZ members) participate in the multimedia data flooding. The selection of the flooding zone members has a direct impact on the flooding performance (i.e. packet loss rate, power consumption, and delay). Thus, the flooding zone size is estimated by a customizable mechanism that provides a tradeoff of reliability, timeliness, and energy efficiency.

Because WSNs is highly dynamic and the network state may change over time, having a fixed FZ initialized by FZIP will not provide the same level of service during the whole real-time multimedia session lifetime. To optimize power spending and ensure good level of quality in such situations, the FZ size may need to be changed several times during the session. Thus, the Flooding Zone Control Protocol (FZCP) is proposed. FZCP monitors the multimedia delivery quality at the sink node and dynamically changes the FZ size when deteriorated performance is encountered. Monitoring the incoming multimedia data helps to detect undesirable performance issues as they happen before taking a corrective action including computing and reinitializing a new FZ. For example, if packet loss is high and quality is bad, FZCP increases the FZ size to reduce packet loss and improve quality. However, if packet loss is minimal and quality is very good, FZCP reduces the FZ size to reduce power consumption while delivering good quality. In this way, the flooding zone size, during the lifetime of the multimedia session, is no longer fixed and can flexibly change over time according to current network conditions.

Using FZIP and FZCP with flooding, delivery of real-time multimedia becomes responsive to dynamically changing network conditions allowing reliable delivery even over WSNs with harsh network conditions and high transmission error rates.

1.4 Objectives, Contributions, and Significance

The main objective of this thesis is to enable reliable and energy efficient real-time multimedia delivery for wireless sensor networks. To the best of our knowledge, our work is different than other by studying the problem of reliable real-time multimedia delivery in WSNs at the different network layers (i.e. physical, data link, and network). That is to say, most of the previous work consider the congestion problem, as the main source of quality deterioration, and neglect many other important aspects encountered in the real world implementation. Wireless sensors are implanted in the real world and get affected by the dynamically changing physical environment in which they reside. We believe that without making real-time multimedia communication protocols resilient to rapidly changing and harsh wireless networks operating conditions, the implementation of wireless multimedia sensor network technology may be jeopardized.

To achieve this objective, this thesis makes the following contributions:

- We introduce and propose the novel concept of Flooding Zone which makes flooding of multimedia data efficient in WSNs. Furthermore, an analysis and a discussion of the impact of flooding zone size on the flooding performance are presented.

- FZIP protocol, a distributed algorithm protocol which can help multimedia flooding protocols to initialize a suitable flooding zone. FZIP is simple, light, hardware independent, easy to customize, and MAC and Network layer independent. These features conform to sensors limitations and facilitate straightforward integration into diverse WSNs architecture and applications.
- FZCP protocol, a complimentary protocol which monitors the multimedia delivery and changes the FZ size to dynamically adapt to current network conditions, thus maintaining good level of quality and reduce power consumption.
- FZIP and FZCP have been implemented and validated in the Network Simulator (NS-2) [NS2].

Several flooding-based protocols [FM06] [ROG04] [M04] have been proposed as a mean of reliable delivery for simple applications in WSNs (e.g. fire detection, temperature reading). However, there has been little or no work on using flooding for multimedia delivery in WSNs because of the associated high power consumption and severe broadcast collisions caused by flooding. However, the introduced FZ concept along with the proposed FZIP [EE08] and FZCP protocols enable resilient, reliable, and power efficient real-time multimedia data delivery in WSNs. Constraining the broadcast storm within a small and carefully selected set of nodes helps reduce power consumption and collision significantly in comparison with normal open flooding. Moreover, monitoring the flooding performance and adapting to the current network state provides soft quality of service assurance for transporting mission critical real-time multimedia data under wide range and highly changing wireless sensor networks conditions.

1.5 Thesis Organization

The thesis is organized as follows. Chapter 2 provides literature review and background about wireless networks architecture and requirements. We review ad hoc networks in general and wireless sensor networks more specifically. We also review the Real-time Transport Protocol (RTP) with its associated control protocol (RTCP). In chapter 3, several related works are reviewed and compared to our proposed protocols. Chapter 4 starts with an explanation and an analysis of the flooding problem and our motivation of the Flooding Zone concept. Then, it describes our proposed FZIP and FZCP solution protocols in details. Chapter 5 describes our simulation topology, experimental scenarios, and the gathered results. Chapter 6 concludes the thesis and gives some future works. Appendix A is for the abbreviations.

1.6 Summary

As the need to optimize power expenditure has driven most of the research in wireless sensor networks so far, very little work has been done in the context of real time multimedia applications. Moreover, while congestion avoidance protocols are effective solutions in wired networks, most of these work need to be rethought in the prospective of WSNs limitations. Mechanisms to efficiently deliver application-level QoS under dynamic and high error prone WSNs, and to map these requirements to metrics (most notably reliability, power consumption, and latency) have not been the primary concern in the majority of research on WSNs. The success of efficient real-time multimedia protocols in WSNs mandates mechanisms to balance between reliability (tolerable packet-loss rate), and power efficiency to preserve acceptable performance. Moreover, the communication protocol should be responsive to dynamically changing wireless network conditions. A real-time multimedia protocol which is not adaptive to network conditions and does not have a notion of performance awareness would either be spending unnecessary extra overhead or fail to provide the expected level of service.

2. LITRETURE REVIEW

This chapter starts with a brief, but important, overview of wireless networks architecture highlighting the difference between infrastructure mode and ad hoc mode. A closer look at ad hoc networks is followed with a brief discussion of their types, applications, and challenges. Next, an overview of Wireless Sensor Networks (WSNs) is presented, discussing their hardware architecture, applications, and limitations.

2.1 Wireless Networks

The recent advances in producing low cost and robust wireless communication technologies accelerated the widespread deployment of wireless communication enabled devices such as laptops, gaming devices, and PDAs. Owing to their "anytime anywhere" capabilities, wireless communication is one of the most promising developments in computer networking. It is expected, in the near future, that wireless devices will be embedded in automobiles, pets, kitchen appliances, cameras, and buildings enabling new type of applications and playing an essential part of human's lives [KR03] [ASC01]. In what follows, different wireless network architectures are discussed.

The architecture of wireless networking can be classified into two categories: infrastructure mode and infrastructure-less mode. The main difference which distinguishes the former from the later is the presence of special nodes, called Access Points (APs), which need to be installed and configured.

2.1.1 Infrastructure mode

In an infrastructure mode wireless network such as a Wireless Local Area Network (WLAN), the wireless network consists of one or more mobile clients directly connected to a Centralized Base Station (BS)/Access Point (AP) over wireless links [KR03]. Mobile clients of such networks have no direct connection with each other, but rather, all wireless communication must always pass through the central AP. This network can be stand alone, in which only clients of this network can share information, or the base station can be wired to an ISP to provide network clients with access to the internet. In addition, it is possible to have multiple APs connected together, with or without wires, to form a distributed system (DS). A wireless distributed system is used to overcome the limited wireless communication ranges. It also enables the mobile clients to move to various places (e.g. meeting rooms, classrooms, cafeterias, etc...) without losing access to the network services.

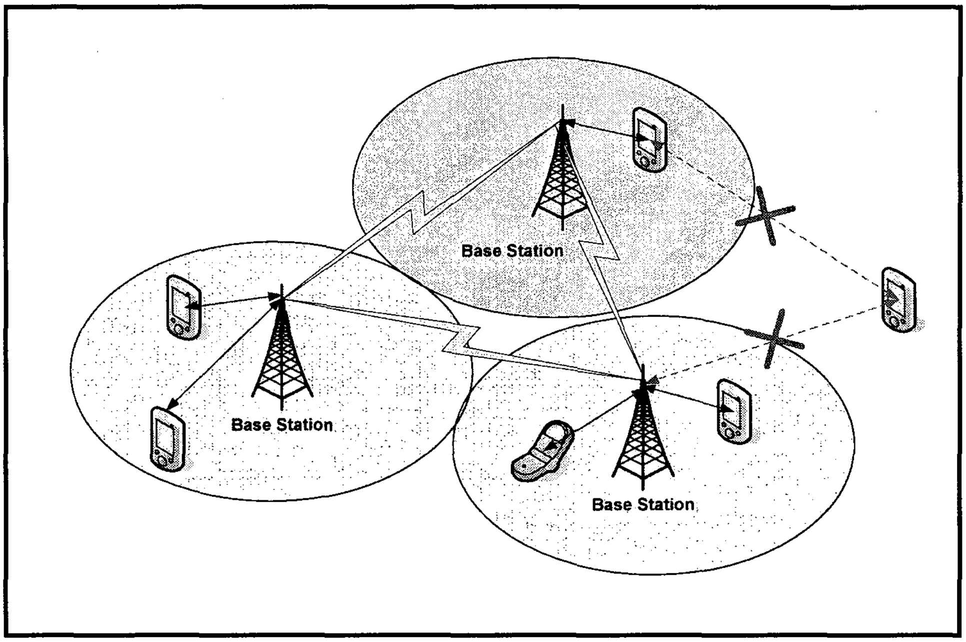


Figure 1: A distributed system of an infrastructure mode wireless network.

In an infrastructure mode wireless network, the APs and all the clients must be configured to use the same network name known as Service Set Identifier (SSID) [KR03]. Due to its centralized topology; infrastructure mode wireless network has the advantage of scalability, centralized security and robustness in comparison to infrastructure-less mode wireless networks, as will be explained in the following subsection.

2.1.2 Infrastructure-less mode (Ad hoc mode)

Recently, and in contrast to this traditional infrastructure mode communication paradigm, infrastructure-less multi-hop wireless networks are receiving a lot of attention. Due to their infrastructure-less nature, such networks demonstrate the advantage of a quick and cost-effective deployment. In an infrastructure-less wireless network, e.g. MANET [FJL00], wireless mesh networks [AX05], and wireless sensor networks [ASSC02], wireless communication involves multiple wireless nodes which relay data in a hop-by-hop fashion from a source node to a destination node [FJL00]. Multi-hop wireless networks are infrastructure-less, self-organizing, and self-healing networks. It facilitates new type of applications where network can be set up on the fly and doesn't require costly and timely consuming configuration and installation.

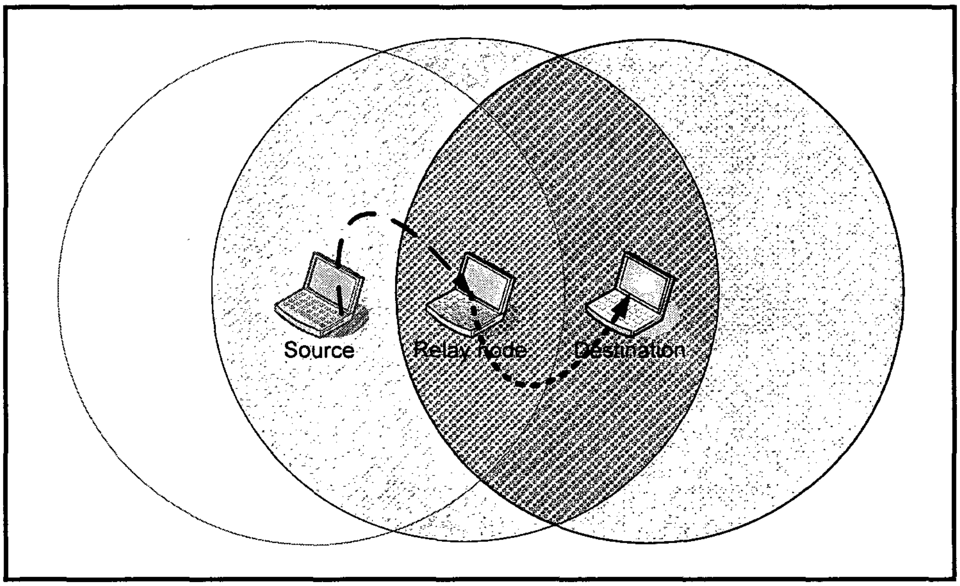


Figure 2: Example of an infrastructure-less wireless network

Some research studies and projects in this field are dealing with issues concerning wireless multi-hop communications such as: routing [RFC3561][MZP01], mobility support, resource management, media access control, Quality of Service (QoS) [WL06] [MBNP03], self-organization and configuration, and security.

2.2 Ad hoc Networks

An ad hoc[3] network is an independent collection of communication devices (nodes), possibly mobile, communicating with each other over a shared, bandwidth constrained, and unfixed wireless links. The decentralized nature of ad hoc networks and node mobility may change the network topology rapidly and unpredictably any time [FLJ00]. In ad hoc networks, nodes may not have direct connection to each other, thus nodes should be able to route/relay packets on behalf of other nodes across the network. Therefore, ad hoc nodes incorporate routing functionalities and perform network activity such as discovering the topology and delivering other nodes messages. This makes ad hoc networks different than structured wired networks, in which dedicated and pre-configured routers perform this task. This multi-hop communication is also in contrast to infrastructure mode wireless networks where communication of all clients is managed through a dedicated and previously setup AP.

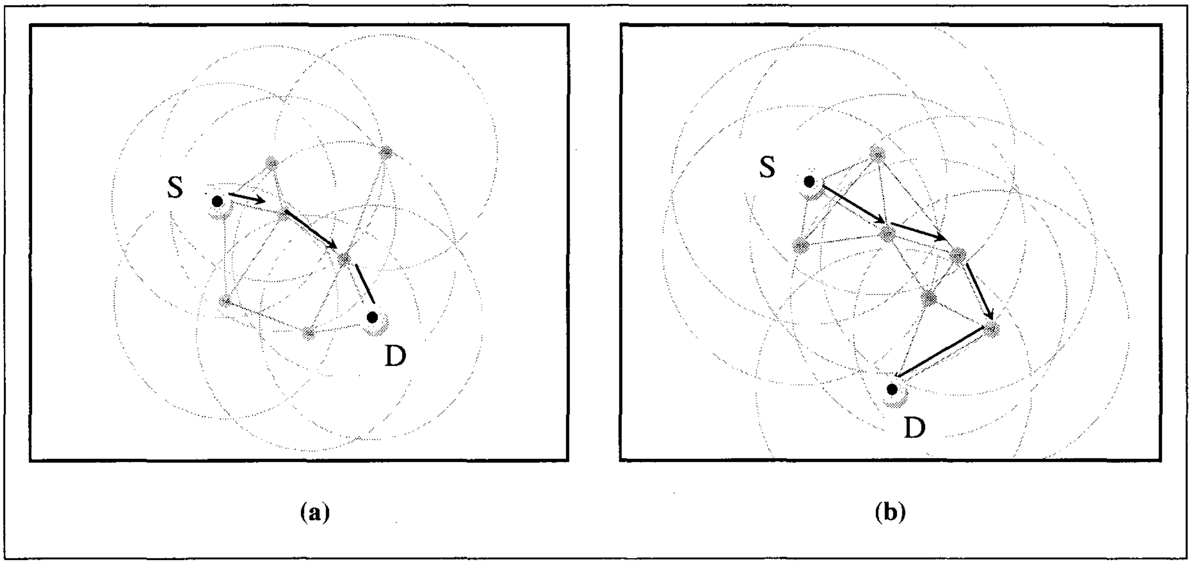


Figure 3: Ad hoc network diagram showing topology changes as nodes are moving.

In Figure 3-a, the source node (S) is sending data to destination node (D). As the nodes are moving, a different path is established for communication between (S) and (D) dynamically (See Figure 3-b).

[3] Ad hoc is a Latin phrase which means "for this [purpose]"

The earliest wireless ad hoc network "packet radio" (PRNET) based on ALOHA net was first implemented in the 1970s [FJL00]. Although ad hoc networks are not new, it became a hot research area only in the last decade. The availability of cheap devices with wireless communication capabilities (e.g. PDAs, Laptops, MP3 players) motivated new type of applications benefiting from ad hoc networks "anywhere and anytime communication" availability.

The unstructured wireless links of ad hoc networks is useful for many application areas such as military, disaster rescuing operations, and public-safety. For example, rescue teams require fast and effective communications when they rush to a rural disaster area to rescue victims. In such cases, it is time consuming to run cabling and setup networking hardware. The rescue team members can utilize their laptops, smart protection helmets equipped with wireless cameras, and PDAs to enable wireless data communications on the fly as soon as they reach the disaster scene.

Based on their mobility support, wireless ad hoc networks are divided into two main categories: quasi-static [IPK05] and MANET [RR02].

2.2.1 Quasi-static ad hoc networks

A static/quasi-static ad hoc network consists of static (non-mobile) nodes. This static nature makes routing protocols simpler. However, due to power failure, temporary device jamming, and link instability, the network topology may frequently change.

2.2.2 Mobile Ad hoc Networks (MANETs)

In Mobile Ad hoc Networks (MANETs) [Lon01], all nodes may be mobile and thus the network topology is expected to be frequently changing. Coping with rapid topology changes, efficiency is one of the main challenges in MANET.

The challenges associated with ad hoc networking cover the entire layer model of the standard protocol stack. Efficient Media Access Control (MAC) layer protocols need to minimize collisions while providing fair access. It is also important for the MAC layer to provide some sort of transmission error recovery and reliability mechanism to the shared wireless link (e.g. CTS/RTS/ACK). In the network layer, the routing protocol needs to discover a routing path based on application requirements such as expected packet loss rate, time, and energy consumption. It also mandates self-organizing and auto-configuration capabilities to smoothly operate in dynamic and changing network topology. Finally, users' applications need to handle link disconnection possibilities with varying delay and packet loss characteristics [RR02].

2.3 Wireless Sensor Networks (WSNs)

The technological advances in manufacturing Micro Electro-Mechanical Systems (MEMS) and wireless communications have enabled the production of small size and low cost wireless sensor devices [AMC07]. A typical sensor device combines a limited processing unit, few kilobytes of memory, and a wireless communication component operated by a limited and often irreplaceable battery. These devices have the potential of sensing specific phenomena within its vicinity (e.g. temperature, motion, light, etc...) and send the gathered information wirelessly to a base station (sink) over a multi-hop with store-and-forward fashion topology. The formation and collaboration of many autonomous sensor nodes is known as Wireless Sensor Network (WSNs). A WSN typically consists of a large number of sensor nodes often deployed densely and randomly in the area to be monitored and thus requires self-organizing capabilities in order to effectively collaborate in an ad hoc manner. In general, the majority of the wireless sensor networks architectures consider stationary (non-mobile) sensor nodes. While WSNs with mobile sink nodes and cluster heads seems to be important for power efficiency [AMC07] [CMY06], little work consider architectures with mobile sensor nodes.

2.3.1 Applications of WSNs

The early motivations behind the development of WSNs were military applications such as battlefield monitoring systems. However, the previously described functionalities and capabilities of WSNs extended their usage to a wide range of other new applications. WSNs applications are envisioned to be used in many civilian and environmental application areas, including natural disaster detection, habitat monitoring, industrial purposes, healthcare applications, home automation, traffic monitoring control on the highways and railways, etc... [AMC07, ASSC02]

For example, in a fire detection WSN application, thousands of low cost sensor nodes are deployed densely and randomly, possibly by a helicopter, in the forest. These sensors are configured to periodically report temperature readings to one or more base stations. The base stations, also known as sinks, are more powerful computers which can be connected to the internet through a communication gateway allowing remote application monitoring and management. In healthcare applications, physiological data of patients can be reported to a remote health center allowing doctors to better understand their patient's conditions in a more convenient way [ASSC02]. Early disaster detection application such as "Tsunami"[4] can save thousands of human lives. In summary, WSNs can enable better understanding of the environment and is envisioned to be an essential part of people's lives in the near future.

2.3.2 WSNs Architecture

Looking inside typical sensor node hardware, we find it consists of four components as shown in Figure 4: a sensing unit, a processing unit, a communication unit and a power source unit [ASSC02]. The sensing unit captures the analog signal observed from the phenomenon then changes it, using an analog to digital converter (ADC), to a digital signal before feeding it to the processing unit. The processing unit, which consists of a small microprocessor and memory, apply required computation before transporting the produced data via the communication unit. Consecutively, the communication unit transports the data coming from its processor by transmitting it on the wireless link. Finally, the power unit, which is often small and 'un-rechargeable' battery, supplies the above mentioned units with required power to operate. A sensor may also have additional application specific components such as a Global Positioning System (GPS) or a mobilization component [ASSC02]. Nevertheless, it is argued in [SHS01] that supplying

[4] Tsunami: a destructive ocean wave caused by an underwater earthquake.

all sensor nodes with GPS hardware is not practicable for sensor networks due to production cost and energy consumption issues.

A typical WSN consists of many sensor nodes and one or more base station nodes (sink). The sink nodes are more powerful nodes with less power concerns. Sensor nodes are scattered in a sensor field as illustrated in Figure 4. Each wireless sensor node senses the area around it and route collected data in a hop-by-hop infrastructure-less fashion to reach the sink. The sink can be linked to the Internet to route the data to the end user.

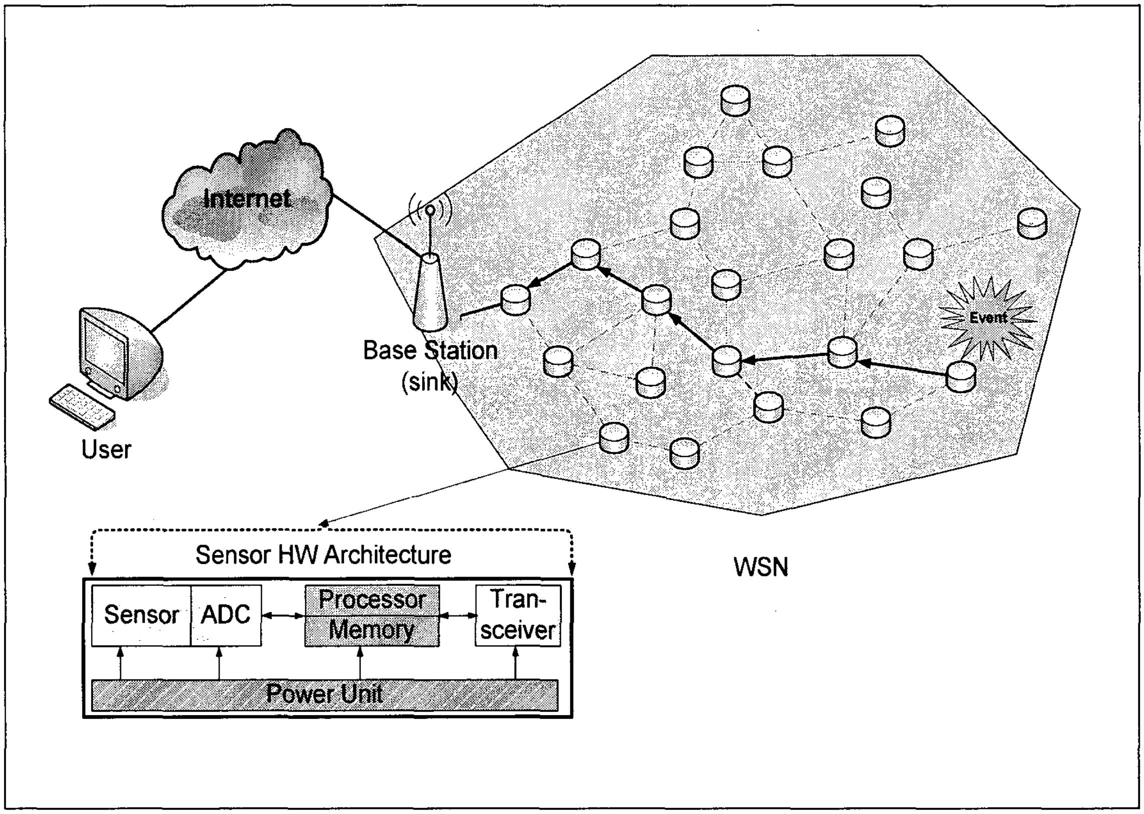


Figure 4: Wireless sensor hardware and network architecture

2.3.3 Characteristics and limitations of WSNs

A wireless sensor network shares many similarities with an ad hoc network and is commonly considered a member of the mobile ad hoc network family. Nevertheless, WSNs have numerous characteristics and limitations which differentiate them from ad hoc networking. The large and densely deployment of sensor network along with the extreme limitation in power and processing capabilities prevent direct applying of ad hoc network protocols and algorithms to sensor networks. To emphasize this fact, the differences between WSNs and ad hoc networks are briefly outlined and discussed below as follows.

2.3.3.1 Network size and density

A WSN application generally requires large number of sensor nodes deployed densely in the area to be monitored. The number of sensor nodes deployed in a WSN (e.g. fire detection application) can be a number of orders of magnitude higher than the number of nodes in an ad hoc network [HHS08]. The number of sensor nodes deployed in fire detection application, for example, may include hundreds or thousands of sensors. Moreover, because sensor nodes are prone to failure, sensors are densely deployed in the region to be monitored. The required communication protocol, data dissemination framework, or routing algorithm in such scenarios must be scalable in a way to support applications with large number of nodes. They must also efficiently utilize the high density nature of the sensor networks.

2.3.3.2 Frequently changing multi-hop wireless topology

A sensor node communicates with a base station (sink) over several unstructured multi-hops pattern. A sensor node, in such ad hoc topology, acts as a sensor and a router sending its own data as well as relaying data received from neighbors in a hop-by-hop store and forward paradigm. A typical WSN network topology is frequently changing even with static sensor nodes (i.e. non-mobile nodes). The large numbers of error prone, inaccessible, unattended, and possibly mobile sensor nodes, make the multi-hop network topology frequently changing. In addition, sensor device failure is a common incident due to energy depletion or jamming. Moreover, additional sensors might be redeployed in some areas over time to replace sensors which ran out of power. Therefore, having a fixed topology in WSNs is an impracticable task for large scale deployment. Handling frequent topology changes in WSNs that have large number of nodes and limited power supply requires special networking protocols compared to ad hoc networks with lower number of nodes and less power constraints.

In addition, sensor nodes may not have unique addressing identifiers because of the large amount of overhead for the large number of deployed sensors which makes id-based routing approaches unsuitable. Rather, sensor nodes use broadcast communications mainly instead of point-to-point communications used in ad hoc networks.

Data centric routing which depends on routing packets based on data they have rather than node id are unique and promising data dissemination paradigms in WSNs (e.g. Directed Diffusion [IGE00]).

2.3.3.3 Extreme limitation in power, processing capabilities, and memory

One of the most important differences and limitations between ad hoc and WSNs is the low power consumption requirement. While wireless nodes in ad hoc network (e.g. PDA, laptop) carry limited power source which can be recharged, sensor nodes carry extremely limited and generally irreplaceable power batteries. For this important reason, sensor network protocols can’t neglect power conservation for achieving high quality of service (QoS). They must be capable of providing flexible trade-off mechanisms that enable power saving options at the cost of lower quality (e.g. throughput, reliability or transmission delay). In addition, the limited processing capabilities and small size memory can't execute complex algorithms or cache huge size of associated data. Therefore, the development of communication protocols for sensor networks are influenced by the limited power supply and resource capabilities of sensors.

2.4 Multimedia Real Time Protocol (RTP)

RTP is an application layer protocol [SCFJ03], which is widely used for transporting real-time multimedia data over the Internet. RTP provides several functions for end-to-end network transportation of real-time multimedia data (e.g. audio and video) and supports multicast and unicast network communications. The most important functionalities supported by RTP are: payload type identification, sequence numbering, time-stamping, and data transmission QoS monitoring.

RTP is independent of the underlying transport and network layers and does not provide resource reservation. For this, RTP doesn’t guarantee quality of service for real-time multimedia transportation over the Internet. In addition, RTP is integrated into the application processing and can be tailored as needed through headers modifications and additions [SCFJ03].

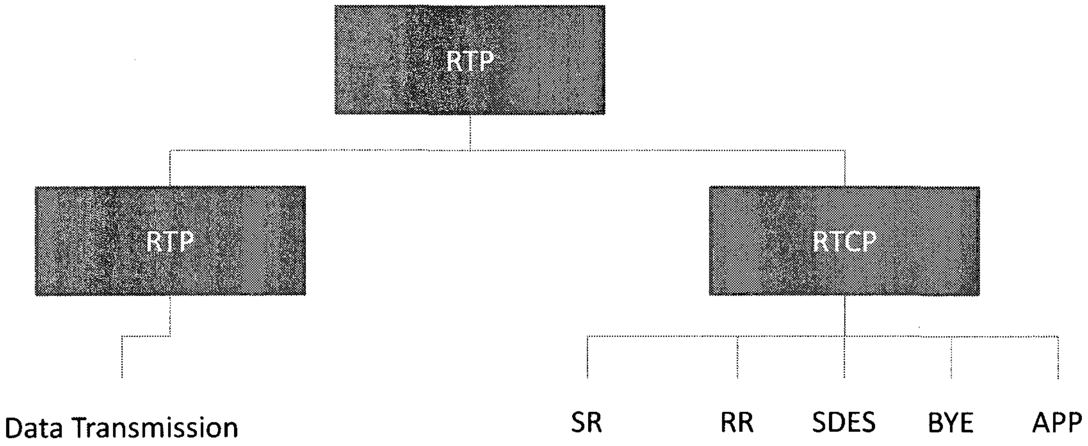
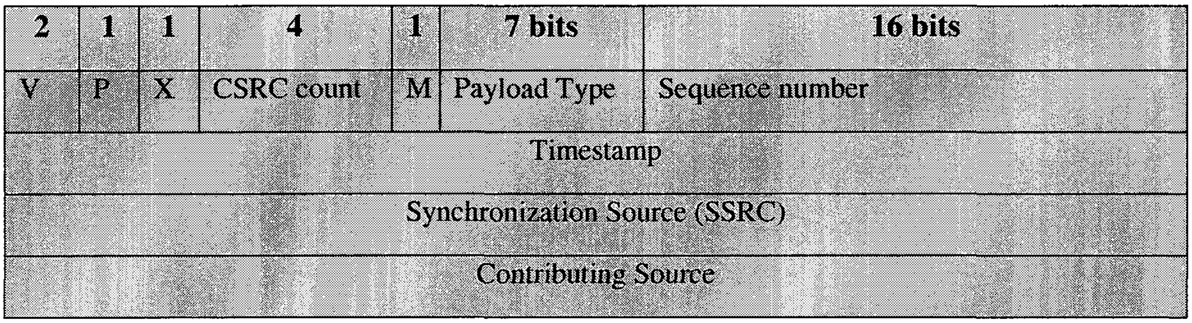


Figure 5: RTP protocol divided into RTP data transmission and RTCP control protocol.

RTP consists of two collaborating protocols: (1) RTP for real-time data transportation and (2) Real-time Control Protocol (RTCP) for controlling the session. RTP data transfer protocol assists in sequencing, payload identification, and play back of media.

Each RTP data packet contains a header and a payload data. Figure 6 shows the standard RTP header format and the important fields are explained below:



2	1	1	4	1	7 bits	16 bits
V	P	X	CSRC count	M	Payload Type	Sequence number
Timestamp						
Synchronization Source (SSRC)						
Contributing Source						

Figure 6: RTP header format [SCFJ03]

Version (V): identifies the used version of RTP protocol.

Marker (M): marks the important events in the packet stream.

Payload type (PT): identifies the format of RTP.

Sequence number: an initial random number generated by the source. It gets incremented by one for each sent RTP data packet. The receivers can use these numbers to calculate packet loss and repair received packet sequence.

Timestamp: represents the sampling instant of the first octet in RTP data packet.

SSRC: A unique random number which identifies the synchronization source (i.e. camera or microphone).

The RTP data transport is improved by the RTCP control protocol. RTCP provides QoS monitoring, bandwidth scaling, and source identification functionalities by sending different type of control packets periodically to all the participants in the real-time multimedia session. The main RTCP packet types are as follows:

- **RR:** Receiver Report packets are sent by receivers and contain statistical information about the running multimedia session (e.g. fraction of packet loss).
- **SR:** Sender Report packets are the same as RR but sent by a member which is a sender and a receiver in the session.
- **SDES:** Source description information (e.g. CNAME, e-mail, address, etc…)
- **BYE:** Used to indicate end of participation.
- **APP:** Application-specific functions.

Monitoring the QoS of the session

The most important RTCP packets are the SR and the RR packets. The packet contains statistical information such as the fraction of packet loss, highest sequence number received, jitter, and other information to compute round trip time delay. The SR packet contains more statistics, i.e. timestamp, count of data packets, and number of payload octets sent. The current session performance can be assessed by exchanging and analyzing these SR and RR feedback packets between the session members. This allows the data sender to adjust the transmission rate or change the data encoding according to the current network state. In bad network condition states, reducing the transmission rate can alleviate congestion and improve QoS.

Source Identification

As participants enter and leave the session without membership, the Source DEScription (SDES) feedback packets allow keeping track of each participant in the session. This allows synchronization of the received audio and video data. Each SDES packet contains at least the canonical name (CNAME) and may contain other identity information such as e-mail or telephone number [SCFJ03].

Bandwidth Scaling

RTCP is limited to use only 5% of the bandwidth for exchanging control packets shared by all participants. The other 95% is used for RTP data transmission. However, to provide scalable operation over a limited bandwidth, the RTCP transmission rate of feedback packets should be controlled. For doing this, each participant keeps track of the number of participants in the session independently to compute and to adjust its feedback control transmission rate.

2.5 Summary

Although wireless communications is not a new technology, we are witnessing a wide spread of wireless enabled devices around the globe. New type of applications necessitates different type of wireless network architecture. In contrast to the stable and widely deployed infrastructure mode architecture, ad hoc mode (e.g. MANET, WSN) is attracting many researchers. However, it is important to realize the unique requirements of WSNs for wide implementation in the near future. Limitations such as scalability, cost, fault tolerance, topology changes, and low power consumption should be considered. While many researchers are currently engaged in developing schemes that fulfill WSNs requirements, less effort has been done for supporting real-time multimedia over WSNs. Next chapter is related work. We present some of the protocols and algorithms proposed to date for sensor networks and highlight the importance and the added functionality of our proposed solution.

3. RELATED WORK

Despite the existence of several reliable and energy efficient dissemination protocols for WSNs, to the best of our knowledge, none of these protocols address reliable real-time multimedia communication support with strict delay bounds. For example, in [WCK02], Wan et al proposed the Pump slowly Fetch Quickly (PSFQ) protocol to allow reliable programming/ re-tasking sensors in WSNs (i.e. sink to sensor nodes communication). The Reliable Multi Segment Transport protocol (RMST) [SH03] can provide reliable sensor to sink conventional data transport service. However, most of these routing protocols and data dissemination frameworks are targeting many-to-one applications which are delay tolerant (e.g. fire detection, motion detection, etc...). The delay tolerance of these applications allows using data caching, aggregation, and in-network processing techniques to provide hop-by-hop recovery and power efficiency. The use of a data cache is required to buffer messages to ensure successful delivery. As the application data are delay tolerant, recovery from packet loss can be achieved by retransmitting the data packets repeatedly until successful reception. In addition, by caching data packets at intermediate nodes, multiple packets can be aggregated together to allow a single transmission to the next hop, thus reducing power consumption. Furthermore, by processing multiple data packets coming from different sources (e.g. by computing average or maximum of several temperature readings), the amount of data flowing to the sink node is reduced, thus reducing power consumption.

However, real time multimedia applications often have different characteristics and stringent latency requirements. Video segments are massively large in size and can't be cached due to their strict play out schedule and extremely small sensor memory size. It is also impractical to decode compressed video frames at intermediate nodes, combine them with other frames from other sources, and then compress them again before sending them to the next hop towards the destination sink. This is because of the complexity of such coding technique, the limited processing and storage power of sensor nodes, and the imposed delay overhead by such process. In addition, as the size of a single video frame often exceeds the Maximum Transmission Unit (MTU), aggregation of multiple video segments is inapplicable. Furthermore, a hundred percent packet loss free reliability is not mandatory. Multimedia applications can tolerate some packet loss, thus successful delivery of all packets is not required.

For all these reasons, we adopted a different approach in providing reliable real-time multimedia delivery in WSNs. Using flooding techniques, reliability can be achieved by concurrent and redundant data transmission and avoiding any delaying mechanism to ensure successful and in time delivery. While this concept is not new (i.e. flooding), this is the first attempt to use flooding for reliable and energy efficient real-time multimedia delivery in WSNs. That is because flooding of real-time multimedia over WSNs consumes large amount of power, while reducing quality because of collisions and contention of the shared wireless channel.

In this thesis, we are not introducing a new flooding protocol. Rather, we are introducing FZIP, a novel setup protocol which can constrain the multimedia flooding in a small size flooding zone. In addition, the size of the flooding zone is controlled by FZCP to balance between reliability, latency, and power consumption. The mechanism of constraining and controlling the flooding storm provided by FZIP and FZCP enables reliable and power efficient multimedia flooding over WSNs. In what follows, we review some of the previous work and compare it with our proposed protocols.

3.1 Flooding

Flooding is an old technique used for data communication which doesn't mandate previous knowledge of network topology or complex routing algorithm [AK04] [HL88]. When a node needs to send a data packet, it simply broadcasts it to all of its neighbors. Each neighboring node in turn rebroadcasts the message to its neighbors, exactly one time, until the packet is reached to the desired destination(s). In addition, flooding nodes have to detect a duplicate received message which is essential to prevent broadcasting messages between neighbors back and forth endlessly. To do this, flooding techniques use one or more associated header fields (e.g. sequence numbers) to detect and eliminate these duplicate messages.

Flooding (or Broadcasting) is an essential and common operation in communication to resolve many issues (e.g. hello messages between neighbors, hosts paging). Moreover, flooding in flat multi-hop wireless networks is widely used by routing protocols during its setup phase in order to establish knowledge about the current network topology. Flooding is simple because it doesn't require previous knowledge about network topology and can work in situations where nodes lack to have ID addresses (e.g. WSNs). In addition to its support of one-to-all (broadcast) or one-to-many (multicast) data delivery communications, flooding is also used for one-to-one communications benefiting from its robust and fault tolerant nature.

However, flooding has several disadvantages especially when used in multi-hop wireless networks. Since radio signals of a node overlaps with other neighboring nodes signals, it is likely to have serious redundancy, contention, and collision when simple flooding technique is used. In [NTCS99], Sze-Yao Ni et al studied this phenomenon in multi-hop wireless networks which they refer to it as the "broadcast storm problem". Redundancy occur when a node receives a message several times, or more specifically when a node tries to broadcast a message while all of its neighbors have received the same message already. The possibility of having many neighboring nodes to rebroadcast a message can severely content with each other and increases collision possibilities. This problem has been studied extensively in the literature and many algorithms and broadcast schedules were proposed to reduce the possibility of broadcast collisions (e.g. [CK85] [LAB93] [CW85]). In [HKB99], Heinzelman et al stated similar problem of flooding in WSNs and listed implosion, overlap, and power blindness problems.

In our FZIP proposed protocol, and in order to reduce the power inefficiency of flooding, we constrain the flooding storm within a small area between source and destination. Thus, we limited the flooding storm in a small area which minimizes the problem scope significantly. The flooding zone also alleviates flooding contention and collision problem by limiting flooding packets to the flooding zone members. In addition, by monitoring data delivery rate at destination sink node, the number of nodes participating in flooding changes dynamically according to the current network state.

3.2 Directed Flooding

Farivar et al [FM05] proposed "Directed Flooding" (DF), a fault tolerant and energy efficient flooding based routing protocol for wireless sensor networks. The functionality of this proposed protocol is built on a very important assumption in which nodes are aware of their geographical location (i.e. x and y coordinates). When a node decides to send a data packet it uses the x and y coordinates to compute a pie-shaped space called "virtual aperture" and encapsulates this computed value with the data packet (see Figure 7). Once the packet is broadcasted, each receiving node checks the computed aperture. If the node's position is within the aperture, it sends an acknowledgment message back to the sending node before repeating the same forwarding packet step (i.e. computes a new "virtual aperture") until the destination node (sink) is reached.

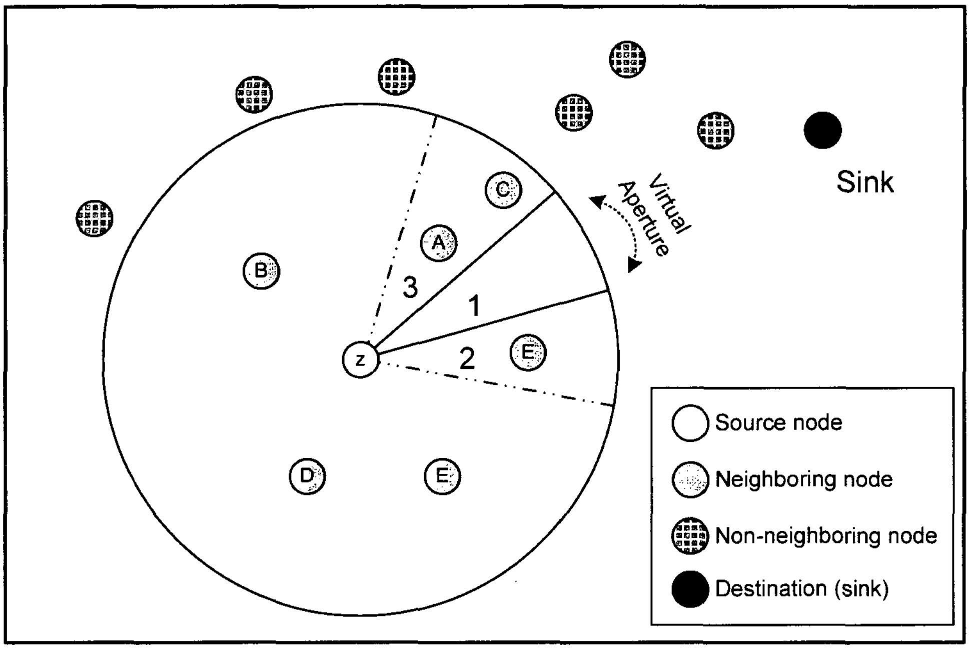


Figure 7: Directed flooding showing different computed virtual apertures.

If the sending node did not receive an acknowledgment within a certain period of time (i.e. when none of the neighboring nodes lie in the virtual aperture), the sending node computes a new virtual aperture clockwise then counterclockwise and repeat the process until an acknowledgment is received.

It is important to emphasize the impact of the "virtual aperture" size on communication efficiency. Computing a large aperture size increases the possible number of nodes lying in it thus increases redundancy and successful packet delivery on the price of more energy consumption. Finding an optimum or suboptimum virtual aperture is a difficult task and depends on the network density. The protocol in [FM05] has not provided any adaptive way for selecting suitable virtual aperture size. Moreover, the used send and acknowledgement mechanism doesn't suit delay sensitive applications such as real-time multimedia. A sending node might repeat the process of virtual aperture computation several time until a neighboring node is reached. This communication mechanism increases latency, jitter (delay variability), and may lead to packet loss due to late arrival. It is clear that such communication mechanism doesn't provide support for real-time multimedia applications. In addition, Directed Flooding may fail to bypass network holes. A hole occurs when a node doesn't have a neighboring node closer to the targeting region. In addition, the ability of the sensor node to have location awareness requires GPS capabilities. Nevertheless, supplying all sensor nodes with GPS hardware may not be feasible for sensor networks due to production cost and energy consumption issues [SHS01].

In our proposed FZIP protocol we have used similar concept of constraining the flooding process, within a limited area (i.e. FZ), but without the need to have geographical node location awareness (i.e. expensive GPS capabilities). The FZ connects the source and the sink over the shortest possible path/paths, as long as they are reachable, thus bypassing any network holes. The flooding zone size is similar to the virtual aperture in limiting nodes which will relay data packets redundantly. However, the FZ size is computed and at the sink node during the FZIP initialization process, and can be controlled (i.e. increased or decreased) by the FZCP protocol based on the current performance and the network state. [FM05] did not provide any adaptive mechanism for selecting suitable virtual aperture size. Moreover, FZIP and FZCP work in a real-time multimedia session, while the Directed Flooding protocol is not suitable for real-time multimedia data as discussed before.

3.3 Video Diffusion

In [ZH06], Zhang et al presented Video Diffusion (VD) as a route-failure resilient and multi-path mechanism to improve wireless video streaming for ad hoc networks. Video Diffusion mechanism tries to improve video data transportation in two ways: correct interpretation of interference-resulted routing failure and multi-path redundant delivery. The authors observed that in MAC protocols based on CSMA/CA (e.g. IEEE 802.11), end-to-end packet delivery often suffers from delivery failure due to contention of the shared wireless channel. In situations of interference and channel contention, routing protocol may wrongly assume that the next hop is unavailable. As a result, the used routing protocol triggers new route exploration process which worsens video transportation quality by temporary blocking transmission until a new route path is discovered. The authors refer to this phenomenon as "misinterpretation for interference-resulted routing failure". The overhead of such process is unacceptable for real-time multimedia data with strict latency requirements. VD avoids this phenomenon by multi-path redundant delivery, where another copy of this packet instant is expected to reach destination over a different path.

Video Diffusion, as proposed by the authors, incorporates a two-phase process, "View Request" and "Video Relay" processes. When the destination node (D) desires to receive video stream from the source node (S), it floods a "View Request" to all nodes including the destination node (D). Relaying nodes avoid broadcasting "View Request" packet more than one time by caching the "View Request" id. Once the source node (S) receives the "View Request" packet, it starts broadcasting video packet to node (D). Every node relay such video packet if, and only if, this node saw a "View Request" and never forwarded this packet before. The conducted simulation and performance comparison of transforming video using UDP and VD protocols showed that VD outperforms UDP by about 52% in the simulated grid topology.

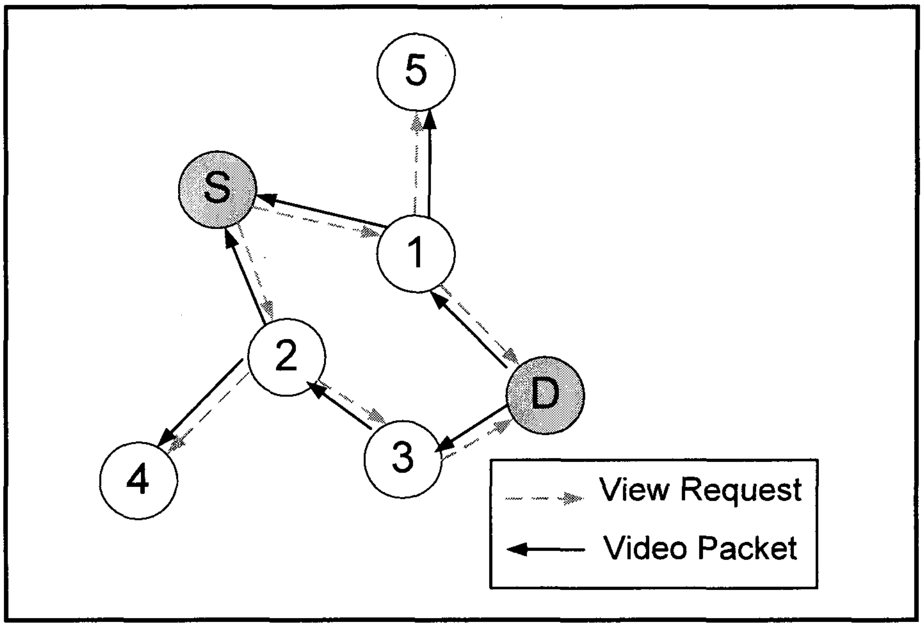


Figure 8: Video Diffusion

Video diffusion is a simple flooding mechanism which works well for small size ad hoc networks. Nevertheless, VD mechanism doesn't scale well for wireless sensor networks or large ad hoc networks. The participation of all nodes in relaying packet has a negative impact on power consumption and video quality. Nodes which are not located in the area between source and destination don't positively contribute to the session. These nodes will increase channel contention and thus reduce bandwidth. This blind nature of all node participation creates large number of redundant paths, where multiple copies of the same packet flow from source node to destination node causing sever channel contention, unnecessary redundancy, higher end-to-end packet loss, and thus poor quality.

Furthermore, the unnecessary participation of all nodes drains sensor nodes power quickly. It is clear that in VD nodes need location awareness in order to accept or to reject participation in a specific session. This is exactly what our proposed FZIP protocol does. Instead of the simple "View Request" process in VD which provides nodes with knowledge about the ongoing session, FZIP uses a two way handshake process. During the exchange of the Init-msg and Ack-msg of FZIP, each node learns its distance from both session end points (i.e. sender and sink). This enables nodes to decide whether to join or to reject session participation based on the desired level of reliability computed at the sink node. The number of redundant paths is controlled by limiting the Flooding Zone size. FZCP monitors the incoming multimedia traffic quality and enlarge or reduce the FZ size to balance between quality and power consumption. Our experiments show that the VD, which we modified to run over large WSNs, achieves much better performance when used with our proposed FZIP and FZCP protocols.

3.4 Constrained Flooding

In Constrained Flooding (CF), Zhang et al [ZF06] proposed a robust and energy efficient routing framework for wireless sensor networks. The earliest appearance of constrained flooding was as a meta-routing strategy in constraint based routing. [ZFK04]. Constrained Flooding takes the advantages of flooding robustness and uses retransmission policies to minimize energy expenditure. It incorporates a real-time reinforcement kernel supported by other meta-routing strategies such as probabilistic constrained retransmission policy and differential delays mechanism. The incorporated real-time learning kernel is important to maintain and reinforce a previously setup potential/cost field. The potential fields allow data to flow from nodes with higher potential fields downward to the sink. However, the maintenance of the potential field is necessary for dynamic networks. In constrained flooding, the real-time kernel allows dynamic adaptation and maintenance of these potential fields without additional management messages by hearing neighboring nodes potential fields integrated with each broadcasted data message. Constrained flooding also incorporates a differential delay and probabilistic retransmission policy to reduce redundancy and collision. The delay mechanism holds the received packet for some time before retransmission. The delay period a packet takes before being retransmitted varies depending on the difference of potential/cost fields. The less the difference is, the sooner the retransmission will be. Moreover, in order to reduce redundancy, each packet has a specific probability (P) of being transmitted, where 0<P<1. For example, If P is 0.9, then the packet has a 90% probability of being retransmitted. The probabilistic retransmission allows the node to set P value based on the number of times this packet has been heard. The more the same message has been heard before, the less possibly the message will be transmitted. Other elements such as aggregation and duplicate retransmission are also suggested to reduce transmission and increase success packet delivery.

Constrained Flooding framework works well for applications which require transmission of mall size data periodically from all nodes to a single base station (sink). This all-to-one convergecast data communication may not work well for real-time multimedia applications because of several reasons. First, data aggregation can't be applied. Multimedia data packets are massive in size and normally exceed the Maximum allowed Transmission Unit (MTU), for example a video frame is normally fragmented into multiple packets. In addition, the differential delay mechanism may not work well for delay intolerant real-time multimedia data. If packets are delayed, the may reach destination after their play-out schedule in which they are considered lost. The differential delay mechanism also increases packet latency variability (Jitter) which is not desirable for smooth media play. Furthermore, storing continuous packets of a multimedia stream in sensors limited memory increases the probability of node congestion. Additionally, the potential field learning kernel might not work correctly for event based applications. As discussed above, the learning kernel enables nodes to adjust their potential fields by hearing the ongoing communication messages between neighboring nodes. This adaptive mechanism works well for applications where sensors are periodically sending data. In event-driven application, multimedia sensors may stay idle for long time until an event is detected. The topology may change during this idle-state and the previously maintained potential field may not match current expected values. This can prevent base station from receiving the streamed multimedia data. Finally, the cross-layer design of CF prevents interoperability and integration in heterogonous architecture which is quite often for multimedia enabled WSNs.

3.5 Real-Time Control Protocol (RTCP)

The Real-Time Control Protocol (RTCP) is used to monitor the QoS of an RTP multimedia session in the Internet. RTCP provides feedback information about the quality of the ongoing multimedia session. This feedback information helps the sender to change its sending rate according to current network state. RTCP sends different type of reports to all the session's member. Most importantly, the Receiver Reports (RRs). RRs feedback reports are sent, as multicast, periodically to all session members. The RR feedback reports contain statistics about the ongoing session such as the fraction packet loss, total number of packet loss, jitter, etc... We used a similar concept of monitoring the session quality and adapting it to the current network state in our proposed FZCP. However, we have made major changes to make it more practical for WSNs. In RTCP, the interval between two RTCP feedback reports from the same member increases with the increase of session members. When the number of session members increases, the feedback reports are sent infrequently and thus become useless as it doesn't reflect current network condition. In addition, sending these feedback reports consumes bandwidth and increases power consumption. In our proposed FZCP, instead of sending feedback reports periodically to the sender to take action, the sink monitors the incoming real-time multimedia traffic, detects quality poverty issues, and decides on the corrective action to be taken. This shifts the complexity of detecting performance issues from sensor nodes (which are energy scarce) to the sink node with less power and memory concerns. In our FZCP, the session is monitored at fixed time intervals. FZCP, at the sink node, computes the fraction of packet loss every 1 sec, to provide more accurate indication of current network state and faster response. In addition, the sink needs not to send (FZ-resize) feedback messages except when bad quality is detected. In the case of bad performance, FZCP initiates a FZ resize process to alleviate current condition. In this way, FZCP saves bandwidth and power from sending periodical feedback report messages.

4. FLOODING ZONE INITALIZATION AND CONTROL PROTOCOLS

This chapter explains our proposed protocols to enable reliable and power efficient real-time multimedia communication in wireless sensor network (WSNs). The chapter is divided into three main sections. In the first section, we introduce, define, and analyze the concept of flooding zone which is a key element in our proposed protocols. In the second section, we present detailed description of our Flooding Zone Initialization Protocol (FZIP). FZIP role is to construct the FZ before the flooding session starts. This enables multimedia flooding protocols (e.g. Video Diffusion) to achieve better performance in terms of energy consumption and reliable delivery. At last, the third section presents and describes in details the Flooding Zone Control Protocol (FZCP). FZCP helps to control the flooding zone size (depth) after being initialized by FZIP. In FZCP, the multimedia traffic is monitored to detect performance issues (i.e. bad quality and high energy consumption) and to take a corrective action by switching the flooding zone size accordingly. This is enables FZCP to maintain and optimize the flooding performance during the multimedia session even under very bad sensor network conditions (e.g. high transmission errors).

4.1 Flooding Zones

In this section, we introduce the concept of flooding zone which is a key element in our proposed protocol. We highlight the motive behind flooding zone by studying the impact of flooding zone size on flooding efficiency in a simulated example. Next, we propose and define our optimal flooding zone concept.

4.1.1 Flooding and Flooding Zones

Flooding is a simple and old method, used in multi-hop wireless ad hoc and sensor networks, which doesn't require complex routing algorithms [ZF06, ROG04, and FM06]. In flooding, a node sends broadcast messages basically to the surrounding neighbors covered by its signal radius. Neighboring nodes in turn rebroadcast the message till it reaches the specified destination or drop it when exceeding a predefined Time-To-Live (TTL[5]) value. In this way, multiple copies of the same packet travel from source to all nodes, including the destination, over multiple and different paths. Every node, including the destination node, may receive multiple instances of the same packet propagating through different joint or disjoint paths. Therefore, we use the term "zone" rather than "path" in describing it. The redundant instances of a single packet provide a level of reliability. That is to say, if an instance of a packet is lost due to any reason (e.g. transmission error, congestion, etc...), another instance of the packet still has the chance to arrive successfully.

Flooding mechanisms are considered to be connectionless and don't use session establishment before transmission. Such open and unrestricted open flooding zone performs poorly in WSNs, which is often large and densely deployed. In such situations, packets collisions become more often which degrade reliable and efficient delivery. In addition, the participation of all nodes in delivering flooded packets will increase resource consumption. This includes the energy used for wireless transmission which will drain sensors limited power quickly. WSNs using flooding protocol has shorter life cycle.

4.1.2 Motivation: Impact of Flooding Zone Size on Efficiency

When using flooding in static wireless ad hoc or sensor network topology, it is expected that not all wireless nodes will always aid in delivering packets between the end-points. The physical location of wireless nodes has a direct impact on such delivery. Avoiding the participation of such useless nodes would not only reduce power expenditure, but also can enhance delivery.

To clarify this issue, consider the simple wireless topology of 100 wireless nodes in Figure 9. Each node is able to communicate with 4 of its neighbors (shown as a solid line). We simulated this topology using NS2 to transfer a video stream of 1000 packets of

[5] TTL specifies how long a datagram can stay in the internet, it is set by the sender, and is decremented by the routers and hosts who process it. A datagram will be discarded if the TTL becomes zero and it still hasn't reached its destination.

equal size. We considered node 25 as a source which sends the packets to node 21 (destination) at a rate of 1 packet every 33ms. We used a simple flooding protocol with a simple transmission policy to broadcast the video stream. This flooding policy prevents broadcasting the same packet except one time by caching sent packets sequence numbers. We did not use any delay retransmission policy for flooding and evenly introduce random traffic and error models between the different nodes.

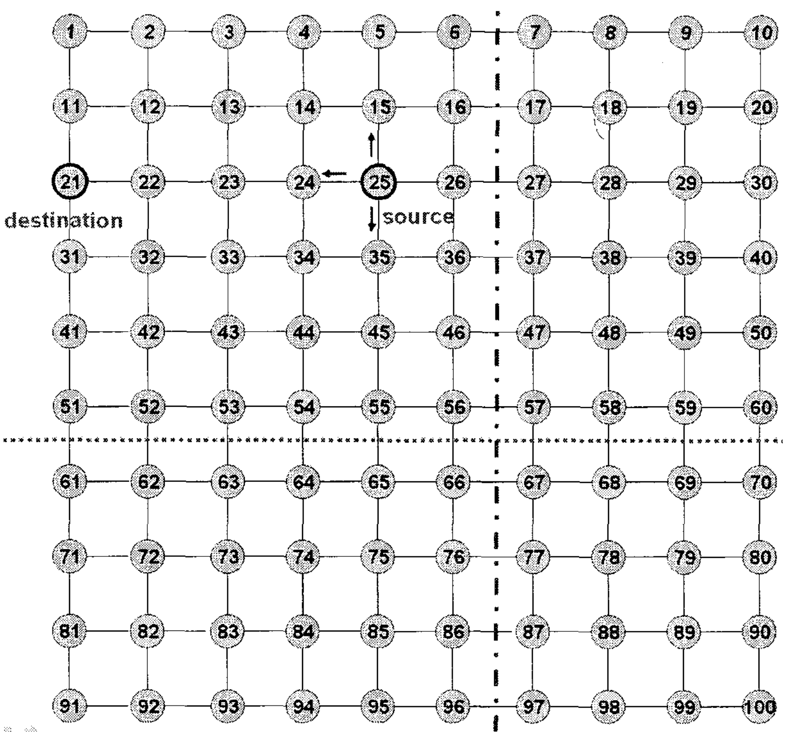


Figure 9: 100 node grid multi-hop network.

We ran the simulation in two different scenarios. In the first scenario, we configured all the nodes in the topology to participate in the flooding (Open Flooding Zone), while in the second scenario, we configured only the nodes at the left of the dashed line and above the dotted line, see figure 1, to be flooding members. All the remaining non configured nodes drop silently any received video packets. We ran the simulation 100 times and took the average packet delivery at the destination node. As expected, we found that the results of the second scenario outperformed the results of the first one in terms of delivery ratio, energy consumption, and latency.

This particular experiment forced important questions. How can we select a suitable set of nodes for flooding communication between any arbitrary session end points? What criteria should be used to select the flooding zone members (flooding nodes)? To answer these questions, we introduce the flooding zone depth principle.

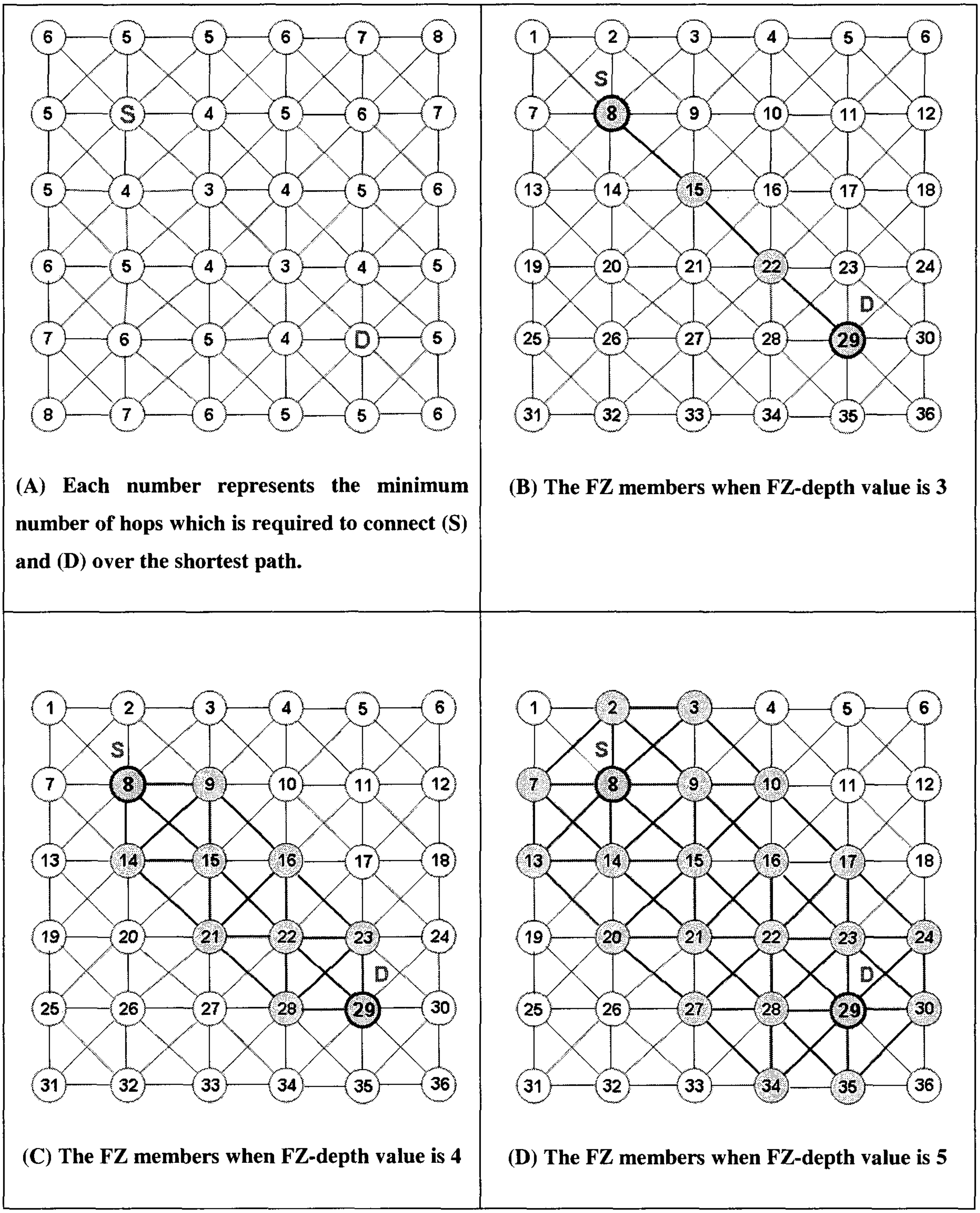
4.1.3 Flooding Zone Depth

In order to determine an efficient flooding zone for a specific session, we need to determine its dimensions. More specifically, we are trying to determine the flooding zone depth. Flooding zone depth represents the number of hops that a packet should not exceed during retransmission. In this way, flooding zone would look like a set of multiple paths connecting the source and destination nodes. The maximum length of these paths should not exceed the flooding zone depth (FZ-depth). For example, figure 10 represents a 36 node WSN. The number shown in each node represents the minimum number of hops needed to forward a packet between source node (S) and destination node (D) while passing this node. If the FZ-depth chosen is 3, then the flooding zone will have only a single path. If the FZ-depth is chosen to be 4 or larger, then the number of paths will increase accordingly.

The Flooding Zone depth (FZ-depth) is bounded by lower and upper bounds. The lower bound is forced by the topology of the wireless nodes and the position of source and destination nodes. FZ-depth can't be less than the required number of hop-by-hop forwarding over the shortest possible path between source and destination. For example, in Figure 9, a packet from node 11 needs to be forwarded at least 3 hops to reach node 14. However, the upper bound of the session depth is enforced by several factors. Different applications have different latency requirements. The hop-by-hop forwarding adds extra queuing and processing delays which affect packet arrival and can't be ignored. For example, in a real-time multimedia application, there is a strict latency and jitter requirements for playing out audio/ video frames. In such situation, session depth can't exceed a certain value to provide smooth media play. In addition, a larger session depth increases the number of flooding nodes members. This in turn increases both power consumption and successful packet delivery. Consequently, choosing session depth should be adaptive in a way to balance between the required application's latency, level of reliability, and energy consumption.

By determining the FZ-depth, we can construct a flooding zone. A flooding Zone with FZ-depth chosen as **n** between sender and receiver is the set of nodes where each node can be part of at least one **n**-hops long path between session end points.

For example, in figure 10, the flooding zone between sources node (S) and destination node (D) with different FZ depths is illustrated. In Figure 10-a, each node is marked by the minimum number of hops required to deliver a packet from (S) to (D) over the shortest possible path. In Figures 10-b, 10-c, and 10-d, the shaded nodes represent the flooding zone members when FZ-depth is chosen as three, four, and five respectively. These nodes are the only nodes which can deliver packets from S to D according to the chosen FZ-depth. In figure 10-b, if the FZ-depth is selected to be 3 (i.e. equals to the lower bound), only nodes 15 and 22 are members of the Flooding Zone. It is trivial to say this is the most efficient FZ in terms of energy since it has the least number of nodes. However, it is obvious that this flooding zone provides the lowest reliability level since it is a single shortest path. If a packet is lost at any hop, then this packet will never reach the destination as there is no redundant copy over other paths. However, with FZ-depth equals to 4 or 5 (Figures 10-c and 2-d respectively), there are multiple paths in the FZ connecting S and D. If a packet gets lost at any node, then there is still a chance that another instance of this packet will make it to destination over a different path. Nevertheless, this reliability costs extra power as a consequence.



(A) Each number represents the minimum number of hops which is required to connect (S) and (D) over the shortest path.

(B) The FZ members when FZ-depth value is 3

(C) The FZ members when FZ-depth value is 4

(D) The FZ members when FZ-depth value is 5

Figure 10: a 36 node Gird topology WSN

4.2 FZIP: Flooding Zone Initialization Protocol

4.2.1 Overview

The Flooding Zone Initialization Protocol (FZIP) is a novel protocol [EE08] which initializes an efficient flooding zone. FZIP role is only to construct the FZ before the flooding session starts. This enables multimedia protocols (e.g. Video Diffusion [ZH06]) to achieve better performance in terms of energy consumption, bandwidth, and successful source to destination packet delivery.

Before the actual process of flooding multimedia packets, FZIP initializes a suitable set of intermediate nodes (i.e. Flooding Zone) which will participate in relaying flooded packets between the source and the destination. The initialized flooding zone will constrain the flooding storm within the FZ, thus limiting the number of nodes which will transmit and receive the flooded packets. FZIP sets up a suitable Flooding Zone (FZ) by a two-way handshake process. The process starts by broadcasting an initialization message (Init-msg) from the source node towards the destination. The destination replies back by another single acknowledgment message (Ack-msg) towards the source. During the exchange of Initialization and acknowledgment messages, intermediate nodes learn their relative hop distance from both sending and receiving nodes. Based on a heuristic function, a suitable FZ-depth is estimated. The measured FZ-depth enables intermediate nodes to decide whether to join the FZ or not. FZIP role ends after the initialization of the flooding zone. Once the FZ is initialized, the source node starts the multimedia session by flooding the data packets to the network. Only flooding zone members will relay these packets hop by hop towards the sink. The actual data transportation is handled by another multimedia flooding protocol (e.g. Video Diffusion). FZIP doesn't depend on geographical nodes' position nor the underlying MAC and routing protocols. This flexibility and interoperability allows FZIP to work in heterogeneous architectures and support many other flooding protocols.

4.2.2 FZIP Protocol Description

4.2.2.1 FZIP Message Layout

To better understand the functionality of the FZIP protocol, it is essential to understand the message layout of the Init-msg and the Ack-msg. The Init-msg and the Ack-msg used by FZIP share the same 6 packet header fields (see Figure 11):

Type	Session-id	Src-id	Des-id	Hop-id	FZ-depth

Figure 11: FZIP Packet Header Format (Init-Msg and Ack-msg)

- **Type**: message type (i.e. Init-msg or Ack-msg)
- **Session-Id**: to distinguish between different sessions
- **Src-Id, Des-Id**: Source and Destination identification
- **Hop-Id**: represents number of times this message is forwarded, starts with zero and gets incremented by 1each time the packet is forwarded (broadcasted).
- **FZ-depth**: limits the number of times a packet can be forwarded before getting dropped (FZIP packets will be discarded if its Hop-Id value exceeds FZ-depth)

The Init-msg and the Ack-msg have the same fixed header values except Hop-Id and FZ-depth fields. The Hop-Id field represents the current number of times this message has been broadcasted. The Hop-Id is dynamically changing as the message is broadcasted. It starts with zero and gets incremented by each node before forwarding it. In this way, multiple instances of the same message can reach a node with different Hop-Id values over different paths. The smallest received Hop-Id field value represents the minimum number of hops required to reach original message initiator (i.e. the source node).

4.2.2.2 Estimating suitable FZ-depth value

FZ-depth, as explained in subsection 4.1.3, plays an important role in the process of flooding zone initialization. FZ-depth value determines the size of the FZ (i.e. the number of zone's nodes) which in turn determines reliability level, latency, and power consumption. Based on the application's requirements, different FZ-depth value can be chosen. Increasing the FZ-depth value will enhance reliability on the price of energy consumption as the number of nodes increases. The value of flooding zone depth in Init-msg and Ack-msg are chosen as the following:

FZ-depth for Init-msg

In the Initialization Message process, the source node sets the FZ-depth field with a predefined value (e.g. default TTL). The source node has no previous knowledge about the hop distance from the destination node. Consequently, the FZ-depth value in the Init-msg is used to prevent the message from looping back within the network. This value can be adjusted according to the maximum allowed number of hops between the source and the destination depending on the deployed network size and density. In NS-2 [NS2], the default TTL value for ad hoc routing protocols is 32 and can be used as the FZ-depth value in the Init-msg.

FZ-depth for Ack-msg

Before the destination node sends the Acknowledgment Message (Ack-msg), the value of the FZ-depth field is computed. It is hard to generalize a function which fits every application. However, this can be customized according to the used WSN application requirements including its priority, level of fault tolerance, and deployed network size.

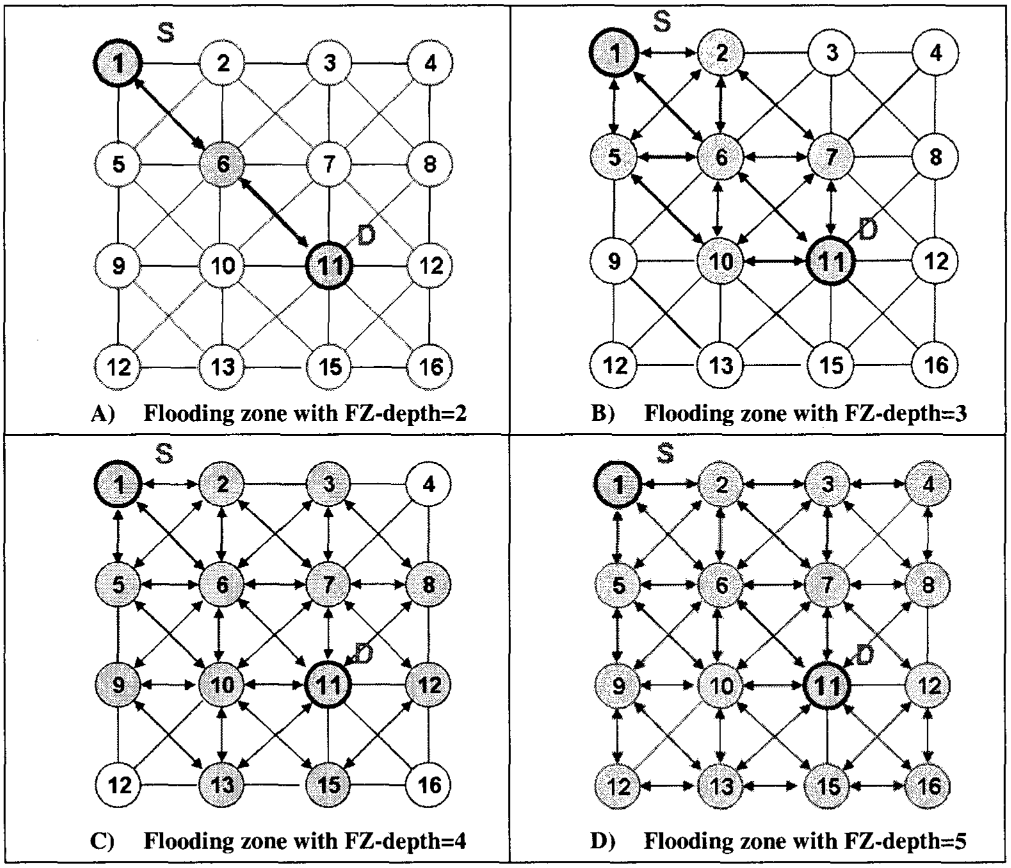
The destination node receives usually multiple instances (copy) of the Init-msg from different paths carrying different Hop-Id counter values. The destination node waits for a period of time before replying back with an Ack-msg. This will allow the destination node (sink) to record and sort all the received Hop-Id counter values within these incoming Initialization messages in a table (FZ-depth Table). It also records the frequency (i.e. number of Init-msg messages) of received messages carrying the same Hop-Id. The smallest value in the table (see Table 1) represents the shortest path distance between the source and destination (i.e. lower bound). The estimated FZ-depth can't be less than this smallest value.

Before sending the Acknowledgment Message to the source node, the destination node uses the Depth Table to select a suitable FZ-depth value based on a pre-configured priority. This can be a simple function which can be customized according to the type of application data, loss tolerance level, and priority. For least power consumption but least level of reliability, FZ-depth can be chosen as the lower bound (i.e. least value in the depth table). Increasing the FZ-depth will provide more reliable service but on the price of energy.

For example, assume that the required application needs the FZ-depth to be selected in a way to ensure there is at least 3 different paths to the source node. This can be easily selected from the depth table by looking at the frequency column. If the application requires having at least 3 redundant paths to ensure acceptable and efficient quality, it can select the Hop-Id with frequency column having the value of 3 or more. For example in Figure 12, the FZ-depth value will be chosen as 3 (second smallest path) because it provides 3 different paths between the source and the destination nodes.

Hop-Id	**Frequency**	
2	**1**	**// (Lower Bound) Least reliability, Least power consumption**
3	**3**	
4	**7**	
5	**8**	**// Best Reliability, Highest power consumption**

Table 1: FZ-depth table



A) Flooding zone with FZ-depth=2

B) Flooding zone with FZ-depth=3

C) Flooding zone with FZ-depth=4

D) Flooding zone with FZ-depth=5

Figure 12: FZ with different FZ-depth values

4.2.2.3 Initialization Message (Init-msg)

The source initiates the FZ by broadcasting an Init-msg. Each intermediate node receiving the Init-msg acts as follows. First, the node checks the header fields and makes sure it is not the destination node and the Hop-Id doesn't exceed the FZ-depth value (i.e. TTL in Init-msg). Second, it looks in the cache for previously received Init-msg. If the node receives this Init-msg for the first time, or the newly received Init-msg carries Hop-Id counter with lower value than the cached one, then it caches this Hop-Id along with the Session-Id number. Finally, the node increments the Hop-Id field, then broadcasts the modified Init-msg to its neighbors. If the received Init-msg has a higher Hop-Id than the cached one, the Init-msg is dropped silently. This process is repeated until the Init-msg reaches the destination or the Hop-Id field exceeds the FZ-depth. At the end of this process, each participating node has cached an Init-Hop-Id representing the smallest number of hop-by-hop distance from the source node.

The following pseudo code explains the Init-msg process:

FZIP Init-msg Pseudo Code

```
IF      (Session-Id not in session-seen) OR ( Hop-Id < Init-Hop-Id stored at node) THEN {
Init-Hop-Id = Hop-Id
Append Session-Id to session-seen
Increment Hop-Id
        IF ( Hop-Id < TTL ) THEN {
            broadcast message to neighbors
        }
}
```

Table 2: FZIP Init-msg pseudo code

4.2.2.4 Acknowledgment-Message (Ack-msg)

Once the destination node receives the first instance of the Init-msg, it waits for a period of time (e.g. 50ms) before replying back by initiating an Ack-msg. This delay allows other Init-msg instances to be delivered and the FZ-depth-Table to be constructed. As discussed in sub-section 4.2.2.2, different methods can be used to measure the required FZ-depth depending on the application used. Afterwards, the destination node constructs and broadcasts an Ack-msg back to the source node carrying the estimated FZ-depth. Each intermediate node receiving the Ack-msg checks its cache for previously seen Init-msg. If the node has no cache record for this session, then the node drops the message. Otherwise, the node decides to join the flooding zone by computing the following equation:

$$\text{Init-Hop-Id} + \text{Ack-Hop-Id} \leq \text{FZ-depth}$$

Where Init-Hop-Id is the cached Init-msg Hop-Id, while Ack-Hop-Id represents the received Ack-msg Hop-Id. If the equation yields true, then the node joins the FZ session, increments the Ack-msg Hop-Id and broadcasts it to its neighboring nodes. This process is repeated at each node until the Ack-msg reaches the source or the Hop-Id field, in the Ack-msg, exceeds the FZ-depth value so the node will drop the Ack-msg.

The following pseudo code explains the Ack-msg process:

FZIP Ack-msg Pseudo Code

```
IF      ( Session-Id is not in session-seen) OR ( Hop-Id > Ack-Hop-Id ) THEN {
Drop Ack-msg and exit
} ELSE {  // This node has seen an Init-msg for this Ack-msg or the new Hop-Id is smaller than
          // the currently cached Ack-msg
          Set Ack-Hop-Id=Hop-Id
          Append Session-Id to session-seen
              IF ( $\text{Init-Hop-Id} + \text{Ack-Hop-Id} \leq \text{FZ-Depth}$ ) THEN {
                  Set Node as a member of this FZIP session  // the node joins the FZ session
              }
              IF ( Hop-Id < TTL ) THEN {
                  broadcast message to neighbors
               } ELSE {
                      Drop Ack-msg and exit
                      }
}
```

Table 3: FZIP Ack-msg Pseudo Code

Figures 13, 14, and 15 illustrate the process of FZIP Init-msg and Ack-msg exchange between session source (S) and destination (D). In Figure 13, S broadcasts the Init-msg to its neighbors. Nodes (a) and (d) receive the Init-msg with Hop-Id=1. They cache it in their buffer, increment the Hop-Id field, and then forward the Init-msg to their neighbors. Each node's number in Figure 13 shows the smallest Init-msg's Hop-Id value it received. D replies back with an Ack-msg. Each node's number in Figure 14 shows the smallest Ack-msg Hop-Id value it received. Each node checks if the sum of Init-Hop-Id and Ack-Hop-Id is greater than FZ-depth or not. Figure 15 shows the initialized nodes as shaded circles with FZ-depth chosen as 4. For example, node (a) joins the session because the sum of received Init and Ack Hop-Ids (i.e. 1 and 3 respectively) is 4. Node (k) does not join the session because the sum of received Init and Ack Hop-Ids (i.e. 1 and 5) is 6 (i.e. greater than the chosen FZ-depth).

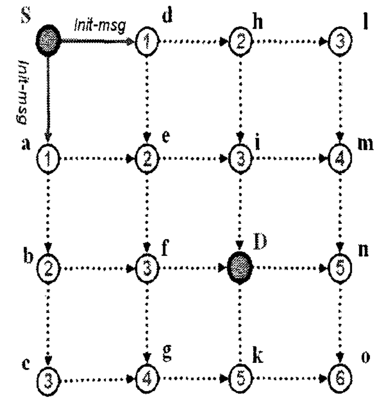


Figure 13: Init-msg flow.

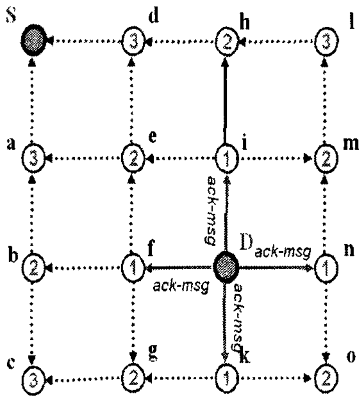


Figure 14: Ack-msg flow.

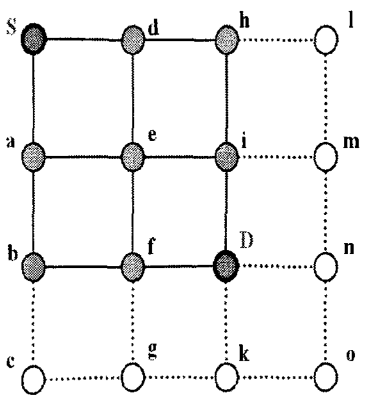


Figure 15: Initialized FZ.

The following subsection, 4.3, describes the Flooding Zone Control Protocol (FZCP).

4.3 FZCP: Flooding Zone Control Protocol (FZCP)

4.3.1 Overview of FZCP

Fixing the FZ-depth size to specific size will not ensure optimum performance especially in a highly dynamic WSN. During the session, the FZ-depth may need to be changed several times to ensure same level of service. The goal of the Flooding Zone Control Protocol (FZCP) is to control the flooding zone size (depth), after being initialized by FZIP. This enables FZCP to maintain and optimize the performance of flooding real-time multimedia data. FZCP monitors the multimedia session to detect performance issues (i.e. bad quality and unnecessary high energy consumption) and to take a corrective action by switching the flooding zone size accordingly. FZCP helps maintaining good multimedia quality and less power consumption for flooding multimedia data over WSNs. This is achieved by incorporating new functionalities in FZCP including: performance monitoring, trade-off, adaptation function, and flooding zone switching operations (FZ resize operation). FZCP at the sink node monitors the session performance by keeping statistics of the received and lost packets. This helps detecting performance issues as they happen. When performance deteriorates, the adaptation algorithm takes a corrective decision by computing a new FZ size. FZCP sends flooding zone resize messages to switch flooding multimedia traffic over a different flooding zone size and adjust performance accordingly. For example, if packet loss is high, FZCP will increase the FZ size to reduce packet loss and improve quality. In this way, the flooding zone size during the session is no longer fixed and can flexibly change over time according to current network conditions. Our simulation shows that FZCP with the newly applied functionalities not only reduce energy and packet loss of multimedia flooding, but also provides soft QoS assurance for flooding multimedia data over highly changing WSNs conditions.

4.3.2 Problem and Motivation

In section 4.1, we introduced the concept of flooding zone for multi-hop wireless networks as a mechanism to optimize power consumption for flooding protocol. We have shown that flooding can achieve better delivery rate and lower energy consumption when constrained in a suitable flooding zone. Accordingly, our proposed Flooding Zone Initialization Protocol (FZIP) is presented to initialize a suitable flooding zone between any two arbitrary nodes. We have discussed the impact of flooding zone size on flooding performance. It is shown that flooding over the smallest possible flooding zone (i.e. shortest possible path) will ensure lowest energy consumption, but might fail to deliver required level of reliability (i.e. acceptable packet loss). This is expected as the smallest flooding zone provides less redundancy for the propagated packets. However, increasing the flooding zone size increases packet redundancy providing lower packet loss on the price of extra energy overhead. Consequently, FZ size must be carefully selected in order to have good quality without consuming unnecessary power.

For a certain WSN, it is possible to physically test and choose the best flooding zone size which can provide acceptable packet loss rate with least energy overhead. However, how to choose an optimized FZ size poses a tradeoff between quality and power efficiency under different WSN sizes, densities, radio channel conditions, and current network utilization level. This dynamic nature of WSNs makes it hard to generalize a schema which can be optimized for every network. In addition, the network conditions could change during the multimedia session and after the flooding zone is initialized. Therefore, the best flooding zone size (i.e. deliver good quality with least power) for the whole multimedia session duration can't be statically fixed to one size. When network conditions get changed (i.e. to better or worse state), the previously initialized FZ size can be smaller than what it should be currently and thus not delivering the multimedia data with required level of quality, or the FZ can be larger than what it should be and thus consuming unnecessary more power. For all these reasons, it is clear that a static FZ size estimation, which doesn't take into consideration dynamic current network state, leads to less efficient network utilization and performance. It is also clear that any multimedia communication protocol for WSNs which is not adaptive to changing network conditions and does not have a notion of performance awareness would either be spending unnecessary extra overhead or fail to provide the expected level of reliability.

4.3.3 FZCP Protocol Description

When a multimedia wireless sensor node needs to starts a real-time multimedia session, it triggers FZIP to initialize a flooding zone before starting flooding multimedia data packets to the sink. During the FZIP initialization phase, the sink and the source nodes exchange initialization & acknowledgment messages in which intermediate nodes compute their hop distances. According to the current wireless network condition and sensor location, an initial flooding zone size is computed and initialized. The source node starts flooding the multimedia packets over the initialized FZ to the sink destination node. Only intermediate nodes, which are members of the initialized FZ, relay flooded multimedia packets hop-by-hop towards the sink. FZCP starts after the flooding zone gets initialized. FZCP monitors the quality of the incoming multimedia data arriving at the sink node. If the incoming multimedia quality is unsatisfactory, FZCP will resize the FZ accordingly (larger/smaller) to avoid performance degradation.

In order to achieve this, FZCP performs different protocol functionalities including: performance monitoring operation, trade-off and decision taking function, and flooding zone resize process. The sink monitors the session performance by computing statistics of the delivered multimedia data (i.e. packet loss). Obtaining these statistics about the multimedia traffic is important to detect performance deterioration and subsequent decision-making. When a performance issue is detected (e.g. high packet loss, high power consumption), FZCP applies a trade-off algorithm to decide on a corrective action to be taken according to the current network conditions. If the taken decision is to improve multimedia quality, FZCP enlarges the FZ size to help reducing packet loss. However, if the taken decision is to reduce power consumption, FZCP decreases the FZ size to reduce overhead. When the FZCP at the sink node decides to change the FZ size, it floods FZ-resize messages to reach intermediate nodes to act on it. Intermediate nodes relay the FZ-resize message after modifying their FZ membership accordingly (i.e. joining or disjoining the new FZ). According to their new membership, intermediate nodes resume or stop flooding multimedia data and thus adjusting delivery performance. In what follows, we describe in details the functionalities of FZCP.

4.3.3.1 Monitoring Flooding Performance (problem detection)

The monitoring operation is complex and requires continuous computation, feedback messaging, and memory resources (e.g. RTP and RTCP). Consequently, applying such operation for resource and bandwidth limited WSN's nodes is not practical. To overcome this problem for FZCP, the monitoring operation is only performed at the receiving node (sink). Instead of sending the feedback report periodically to the sender to take action, the sink monitors the incoming real-time multimedia traffic, detect quality deterioration issues, and decide on the corrective action to be taken. This shifts the complexity of detecting performance issues from sensor nodes , which are energy scarce, to the sink node with less power and memory concerns. In this way, no feedback reports need to be sent to source node periodically. In addition, FZ resize messages are sent only when the multimedia session exhibits some performance issues and need to take a corrective action. This scheme minimizes the overhead of FZCP operation which goes well for resource limited WSNs. For example, in a situation with a stable network condition, FZCP sends no FZ resize messages and thus has no associated overhead on the sensor network. In what follows, our monitoring operation is presented:

Identifying performance problem and deciding on correction action

Monitoring the session performance is important to detect network state changes and deteriorated performance during the lifetime of the real-time multimedia session. This awareness allows FZCP to take the correct subsequent adaptation decisions. This is achieved at the sink node where FZCP monitors the incoming data. The monitoring operation builds performance statistics by computing the number of packets successfully received at the sink node periodically as a measure of quality. At every T_i time interval (e.g. every 1 second), the monitoring function computes Packet Loss Fraction (PL) using the following formula [RFC3550]:

$$\boxed{\begin{array}{c}\textit{Packet Loss Fraction } (PL)\\ \textit{during interval } T(i)\end{array} = \frac{\begin{array}{c}\textit{Expected Number of Packets}\\ \textit{during interval } T(i)\end{array} - \begin{array}{c}\textit{Number of Packets Received}\\ \textit{during interval } T(i)\end{array}}{\begin{array}{c}\textit{Expected Number of Packets}\\ \textit{during interval } T(i)\end{array}}}$$

We assume constant bit rate multimedia session in which multimedia data is sampled at a constant rate and therefore the expected number of packets can be computed for any interval. We also assume that each multimedia packet is encapsulated with a header containing several fields such as session number, sequence no, timestamps, etc...(i.e. similar to RTP packets). The session number field helps to distinguish between different sessions. The sequence number helps to filter duplicate received data packets and helps the application to reconstruct the playback sequence. The timestamp field is used to calculate the time this packet took from source node to destination. Other header fields can be added or customized according to the used multimedia flooding protocol.

The monitoring function uses several thresholds to help the protocol detect undesired performance. The Maximum Packet Loss Threshold (MAX_PLTH) represents the maximum packet loss percentage during an interval, which can be tolerated by the multimedia application (e.g. Voice can tolerate up to 20%)[6]. This threshold represents the maximum tolerable packet loss by the application. If the packet loss fraction PL_i at interval T_i exceeds MAX_PLTH then the quality of the multimedia session is considered bad. In such case, the monitoring function sends a FZ size increase (+FZ) message demanding better service by switching to a larger FZ size.

However, the Minimum Packet Loss Threshold (MIN_PLTH) is used to prevent unnecessary power consumption. Given that multimedia applications are packet loss tolerant up to a limit, it is more important to save power as long as the quality is in an acceptable state (i.e. packet loss between MIN_PLTH and MAX_PLTH). If the packet loss is lower than (MIN_PLTH) then the current network state is very good and worth trying reducing the quality a bit to save more power. In this case, FZCP initiates a FZ size decrease message (-FZ) requesting lower power consumption by switching to a smaller FZ.

When inefficient performance is detected and FZCP decides to modify the FZ size, it checks two more thresholds before sending the +FZ or -FZ messages.

[6] The process of selecting the MIN_PLTH and MAX_PLTH depends on the type of multimedia (i.e. audio or video), the used multimedia encoding, and the level of packet loss tolerance (i.e. level of sensitivity and maximum allowed latency).

Minimum Flooding Zone Threshold (MIN_FZTH) represents the Flooding Zone lower bound (i.e. shortest path between source and destination sink) and is computed from the current FZ depth table initiated by FZIP[7]. MIN_FZTH is fixed for the whole session and is set with the smallest hop id in the flooding zone depth table. FZCP will not send a -FZ message if the current FZ depth equals the MIN_FZTH.

However, Maximum Flooding Zone Threshold (MAX_FZTH) represents the maximum bound and is computed dynamically. FZCP will not send a +FZ message if the current FZ depth equals the MAX_FZTH. MAX_FZTH is primarily configured with a very large number (infinity) as the maximum FZ depth is not known yet. After each +FZ operation, FZCP compares the new packet loss fraction with the packet loss fraction before switching to a larger FZ size. If quality worse than before increasing the FZ size, FZCP sets the MAX_FZTH with the previous FZ size and decreases the FZ size by sending a –FZ message. Enlarging the FZ size in this case will not benefit the quality (i.e. will not reduce packet loss). In this case, rate adaptation can be performed if supported[8]. FZCP can send a feedback message to the source node to switch the multimedia encoding to lower rate and thus the packet loss fraction can be reduced.

The following pseudo code describes the decision making algorithm:

Corrective Action Decision Making Algorithm

```
At every  time interval ($T_i$ second)
IF ( PL > MAX_PLTH & Current FZ-Depth < MAX_FZTH) THEN    // Quality is Bad
{
   Send  Resize-msg to increase FZ Depth (+FZ)      // Switch to Larger Flooding Zone to reduce
                                                    // packet loss
}
ELSE                                                          // Quality is Very Good
```

[7] In a static WSN, the FZ depth table is fixed for the whole session and is populated during the FZIP process.

[8] Rate adaptation is also possible but is not discussed or implemented in this work.

```
IF ( PL < MIN_PLTH & MIN_FZTH > Lower bound) THEN
{
Send Resize-msg to decrease FZ Depth (-FZ)          // Switch to smaller FZ to preserve power
}
exit                                                //Quality is acceptable, Keep current FZ
```

Table 4: Decision making algorithm pseudo code

4.3.3.2 Adjusting Flooding Performance (FZ resize operation)

When inappropriate performance is detected and a corrective decision is taken, FZCP initiates a flooding zone resize process to accommodate this performance degradation. The resize process aims to increase or decrease the flooding zone size in accordance with the corrective action decision taken. The sink node broadcasts a Resize Message (Resize-msg) demanding intermediate nodes to modify their FZ membership (i.e. join/disjoin the FZ). Based on their new FZ membership state, intermediate nodes resume or stop forwarding multimedia data and thus adjust flooding performance.

FZCP Message Layout

FZCP uses a small size packet header fields for its Resize-msg messages .The Resize-msg share the same packet header format and values of FZIP's Initialization and Acknowledgment messages (See Figure 16). Before sending the Resize-msg, FZCP sets these fields with the same values of the current running FZ. The Hop-Id starts with zero and get incremented after each broadcast. However, an FZCP message carries a different type code to distinguish it from Init and Ack FZIP messages. Furthermore, it sets FZ-depth field with a different value decided by the performance monitoring function. A Flooding Zone increase request (+FZ Resize-msg) holds an incremented current FZ-depth value, while a Flooding Zone decrease request (-FZ Resize-msg) holds a decremented FZ-depth value. The incremented or decremented values are selected from FZ depth Table.

Type	Session-id	Src-id	Des-id	Hop-id	FZ-depth

Figure 16: FZCP Packet Header Format

- Type: message type (i.e. +FZ, or –FZ or Rate Adaptation)
- Session-Id: same session number used by FZIP
- Src-Id, Des-Id: Curent Source and Destination identification
- Hop-Id: starts with Zero and gets incremented after each broadcast by a node
- FZ-depth: newly computed FZ-depth by the correction decision making algorithm

Flooding Zone Resize Message (FZ Resize-msg)

The sink starts the FZ resizing process by broadcasting a Resize-msg with the newly computed FZ-depth value. Each intermediate node receiving the Resize-msg acts as follow. First, the node checks if it was previously initialized by FZIP by verifying the Session-Id, Src-Id, and Des-Id. A node not previously initialized by FZIP will abort this FZCP process (i.e. no cached session information, Ack-Hop-Id, or Init-Hop-Id). Second, the node decides on joining the flooding zone by comparing its previously cached Init-Hop-Id and Ack-Hop-Id using the following equation: Init-Hop-Id + Ack-Hop-Id $\leq$ FZ-depth. If the equation yields true, then the node joins the FZ session, increments the Hop-Id and rebroadcasts it to its neighboring nodes. Otherwise, the node aborts the operation. This process is repeated at each node as long as the Resize-msg Hop-Id field does not exceed the FZ-depth value.

The following presents pseudo code explaining the Resize-msg process:

FZCP Resize-msg Pseudo Code

```
IF      (Session is not seen) OR ( Hop-Id > FZ-depth)
        THEN
```

```
    Exit Resize-msg Operation
END IF
IF    $(Init\text{-}Hop\text{-}Id + Ack\text{-}Hop\text{-}Id \leq FZ\text{-}depth)$ THEN
    Set FZ flag ON                 (node become a member of this FZ)
ELSE
    Set FZ flag OFF                (node doesn't become a member of this FZ)
END IF
Increment Hop-Id
Broadcast Resize-Msg
```

Table 5: FZCP Resize-msg Pseudo Code

4.4 Summary

This chapter explains our proposed protocols to enable reliable and power efficient real-time multimedia delivery over Wireless S Sensor Network (WSNs). It introduces, defines, and analyzes the concept of flooding zone which is a key element in our proposed protocols. Then, detailed description of our Flooding Zone Initialization Protocol (FZIP) is presented. At last, the Flooding Zone Control Protocol (FZCP) is described in details..

5. EXPERIMENTAL SIMULATION

Discrete packet-level simulation is used to examine and analyze the performance of our proposed Flooding Zone Initialization Protocol (FZIP) and Flooding zone Control protocol (FZCP). The two protocols have been coded and experimental simulations were conducted with the Network Simulator (NS) [NS2]. While many network simulators such as NS-2, Opnet[9], QualNet[10] , etc..., are widely available, we use NS-2 in this thesis. NS-2 enables us to evaluate the behavior of our proposed protocols in large-scale and complex WSNs at low cost. The goal of the simulated experiments is to study the performance of the proposed protocols in relation to several evaluation metrics and highlight the advantages of our design choices. Simulation results confirm the ability of our protocols to enhance the quality of real-time multimedia packet delivery and achieve lower energy consumption.

This chapter starts with an overview of the NS-2 simulator. This is followed by a description of our simulation environment and configuration. Afterwards, different simulation experiments are described and discussed.

In section 5.3, we present two simulation scenarios to evaluate the setup FZIP protocol. In the first FZIP scenario, subsection 5.3.2, we study the performance of transferring a real-time multimedia stream using different cases (UDP, VD, and VD+FZIP) over WSNs. This experiment emphasizes the effect of the distance, in terms of hops, between the source and the sink nodes, on the performance of the different cases. In the second scenario, subsection 5.3.3, we study the performance of UDP, VD, and three different VD+FZIP cases (using different flooding zone sizes) to transfer real-time multimedia streams under various wireless channel conditions. This emphasizes the performance variation of the different protocols in relation to changing network conditions, as well as, in relation to the flooding zone size.

[9] http://www.opnet.com
[10] http://www.scalable-networks.com

Furthermore, in section 5.4, we present our two simulation scenarios to validate the ability of the FZCP protocol to control the FZ size in order to maintain good quality reduce power consumption. We study the performance of transferring a real-time multimedia stream using VD+FZIP with VD+FZIP+FZCP over WSNs. In each scenario, we use different cases of VD+FZIP, configured to use different flooding zone sizes. In the first FZCP scenario, subsection 5.4.2, we study the performance of transferring a real-time multimedia stream in a static network condition. This leads to a comparison of the best achieved performance from the different used VD+FZIP cases with the performance of VD+FZIP+FZCP case. In the second scenario, subsection 5.4.3, we study the performance of transferring a real-time multimedia stream in a dynamically changing network condition during the session's lifetime. This highlights the need to have a dynamically changing FZ size during the session lifetime to maintain good quality and reduce power consumption. We conclude this chapter with a summary.

5.1 Overview of the Network Simulator (NS-2)

The Network Simulator (NS-2) [NS2], widely known as NS-2, is a discrete event simulator designed for networking research. NS-2 provides essential support for studying the dynamic nature of communication protocols at low cost. Different network protocols (e.g. TCP, UDP, routing protocols, and multicast protocols) can be simulated by NS-2. In addition, simulation for wired and wireless (local and satellite) networks is also supported. NS-2 is a free network simulation application that can be downloaded from the web at no cost. In addition, NS is compatible with Windows and UNIX operating systems.

Due to its modularity and flexibility, NS-2 has gained great popularity in the networking research community. One of the most important features of NS-2 is its object-oriented paradigm. NS-2 comes with many built-in objects and modules for direct use. However, advanced users may need to write their own C++ modules. For this, NS-2 has an open architecture that allows users to edit existing modules or to add new functionalities. NS-2 is written in C++ language, with an OTcl interpreter shell. The OTcl shell acts as a user interface that allows the input model files (OTcl scripts) to be executed. New simulator objects can be created through the OTcl interpreter. These new simulator objects are then mapped to other corresponding C++ objects at simulation runtime.

We simulate WSNs topologies using the NS-2 wireless module. The wireless model supports essential features that allow the simulations of mobile and static multi-hop ad-hoc networks. In addition, the wireless model provides several routing protocols such as DSDV, AODV, TORA, and DSR. An implementation module for the Directed Diffusion protocol is also supported.

NS-2 can be run from the command shell prompt. The generated trace files contain detailed information of the network communication at the different layers of the protocol stack (i.e. MAC, Network, and Transport levels). Simulation scenarios can be visualized using the Network AniMator (NAM) [NS2], which is a complementary animation tool for NS-2.

5.2 Simulation Environment and Configuration

We used the Network Simulator (NS-2) version 2.3.1 for conducting all our experiments. We installed NS-2 on an Ubunto version 7.1 Linux system on a Sun AMD Optron 64-bits workstation. This system has a 2 processor and a 6 GB of RAM.

NS-2 supports many wireless Medium Access Control (MAC) protocols including IEEE 802.11, SimplMAC, and sMAC. As FZIP and FZCP are independent of the link layer (MAC), any of the above mentioned MAC protocols can be used for simulation. However, we limited our simulation experiments on IEEE MAC 802.11 (in DFC mode). This is because IEEE 802.11 is the dominant and most widely used MAC protocol for wireless networks. In addition, it supports the MAC layer RTS/CTS/DATA/ACK reliability pattern for all unicast communications and simply sends out DATA for all broadcast communications. Furthermore, the supported sMAC module by NS-2 did not scale well for our large simulated WSNs topologies. While we were able to simulate small size WSNs topology (36 nodes) with sMAC, larger topologies of 100 and 400 nodes generated core dump bug errors.

We used the default features of IEEE 802.11 in all of our different topologies. This includes the default configuration of the wireless communication range (250 meters) and CTS/RTS/DATA/ACK link layer reliability features.

NS-2 implements three different propagation models: free space model, two-ray ground model, and the shadowing model. However, all of our simulations are based on the TwoRayGround model which gives more accurate prediction at a long distance than the free space model.

At the network layer level, we have used Ad hoc On-Demand Distance Vector (AODV) [RFC 3561]. AODV is one of the most widely accepted routing protocols for ad hoc networks and AODV model is well implemented in the NS-2 simulator. As flooding protocols do not require routing, the on demand feature of AODV ensures that no additional control traffic routing is generated during the simulation except when used for unicast communication protocols (e.g. UDP).

The default energy model of NS-2 is used to compute power consumption. All nodes at the beginning of the simulation are set with 1000 joules power source. Wireless transmission of a packet is configured to consume twice the power for receiving a packet.

Each simulation output is recorded in a trace file including all protocol stack layers communications (i.e. MAC, Network, and Transport layers). We also implemented our own monitoring agents to record some other important statistical information at certain intervals (i.e. fraction of packet loss and latency).

The following, Table 6, summarizes the wireless model configuration used in our simulation.

	Wireless Model Category	**Simulation Settings**
1	Channel Type	WirelessChannel
2	Radio-propagation model	TwoRayGround
3	Network interface type	WirelessPhy
4	MAC type	Mac/802_11 with default 1 Mbps bandwidth

5	Routing Protocol	AODV
6	Interface queue type	Queue/DropTail/PriQueue
7	Max packet in interface queue	50
8	Antenna model	Antenna/OmniAntenna

Table 6: Used wireless module configuration.

For conducting our simulation, several agents were coded. One of the good features of FZIP and FZCP is their simplicity. This enabled us to implement them using oTcl and not need to write a separate new model using C++. Instead, FZIP and FZCP agents were coded in oTcl by modifying the built-in Messaging Passing module according to the pseudo code presented in chapter 4. We also implemented the Video Diffusion protocol (VD) [ZH06] using oTcl. We also coded a multimedia traffic generator for simulating a multimedia stream. This generator creates multimedia packets by allocating the required packet size, initializes some header fields, and then passes these packets to the transport protocol (e.g. UDP, VD, etc...). A multimedia packet constructed by this generator consists of the following header fields: Session-ID, Sequence Number, and Timestamp.

We have chosen different metrics to emphasize the motivation for our proposed protocols (i.e. FZIP and FZCP) during our simulation. These metrics are:

- **Packet Loss Rate:** which measures the ratio of the number of packets lost, calculated at the sink, to the actual number of packets a source node sent. This metric reflects the quality of the received real-time multimedia stream.

$$Packet\ Loss\ Ratio = 1 - \frac{Total\ number\ of\ packets\ received\ at\ sink\ node}{Total\ number\ of\ packets\ sent\ by\ source\ node}$$

- **Average energy consumption:** which indicates the average consumed power per node during the multimedia transfer and is calculated as the ratio of total consumed power by all nodes to the total number of nodes in the network. The average energy consumption includes all MAC layer wireless transmission during the simulation including AODV and CTS/RTS/ACK messages.

$$Average\ energy\ consumption = \frac{Total\ consumed\ power\ by\ all\ nodes}{Number\ of\ nodes\ in\ the\ topology}$$

- **Average Delivery Overhead:** This metric is used to examine the associated overhead of each simulation. It measures the average number of wireless transmissions (Tx) required to deliver a message from the source node to the sink. Wireless transmissions include all MAC layer messages recorded in the trace file of the simulation. As wireless transmissions are the main source of power consumption, average overhead gives another indication of the associated power consumption of a protocol.

$$\text{Average Delivery Overhead} = \frac{Total\ number\ of\ wireless\ transmissions}{Number\ of\ multimedia\ packets\ sent\ by\ source\ node}$$

- **Average Latency:** which measures the average time a packet takes during transmission from the source node until the reception of the packet by the destination node (sink). This metric examines the delay bound performance for a simulated case.

5.3 FZIP Simulation

The goal of the two simulated scenarios in subsections 5.3.2 and 5.3.3 is to justify the flooding zone concept and highlight the advantages of FZIP. More specifically, we want to confirm, by simulation, that FZIP helps multimedia flooding protocol to:

- Enhance real-time multimedia delivery service (i.e. reduce packet loss)
- Reduce power consumption.

5.3.1 Simulation Approach

To evaluate FZIP, we compare its performance with other protocols in two different scenarios. In each scenario, in subsection 5.3.2 and 5.3.3, we study the performance of the different protocols in terms of average packet loss and average power consumption. In subsection 5.3.3, we also measure the average latency metric. In the first scenario, subsection 5.3.2, we study the performance of transferring a real-time multimedia stream using different cases (i.e. UDP, VD, VD+FZIP) as a function of end-to-end distance (i.e. number of hops between the multimedia source node and the receiving sink node). This will emphasize the performance variation as the distance, in terms of hops, between the source and the sink node increases. While in the second scenario, subsection 5.3.3, we study the performance of UDP, VD, and three different VD+FZIP cases (using different flooding zone sizes) to transfer real-time multimedia streams under various wireless channel conditions. This will emphasize the performance variation of the different cases in relation to network condition (i.e. channel error rate %), as well as, in relation to the flooding zone size (depth).

In order to highlight the design motivation of FZIP, we compare the performance of transferring a multimedia stream in three approaches: a single path communication (using UDP), flooding over all possible paths (using VD), and flooding using the proposed setup Flooding Zone Initialization Protocol (using VD+FZIP).

UDP

The User Datagram Protocol (UDP) [RFC768] is the most widely used standard protocol for multimedia transport in IP networks. The purpose of using the UDP protocol is to compare the performance of single path delivery against multi-path delivery using flooding. It is trivial to notice that in a perfect situation (low sending rate, no transmission errors, and short source to sink distance), UDP with single path consumes less total power in comparison with multi-paths delivery. This is because, in multi-paths, the number of packet transmissions would be in multiples of the number of used paths. However, this might not be true when the energy and latency overhead of routing protocol and MAC reliability mechanism is measured. That is to say, the overhead of the CST/RST/ACK MAC layer reliability mechanism as well as new route exploration and update messaging are also included in our evaluation. In our experiment, we have selected the Ad hoc On-Demand Distance Vector (AODV) [RFC3561] routing protocol to support UDP communications. AODV is one of the most widely accepted routing protocols for ad hoc networks and the AODV model is well implemented in the NS-2 simulator. In addition, AODV is designed to support various ad hoc networks size (e.g. WSNs) ranging from tens to thousands of mobile nodes. Moreover, AODV can handle a variety of data traffic levels and mobility rates. However, we will limit the network size in our simulation to 100 and 400 static nodes only.

Video Diffusion (VD)

The second approach represents a simple open flooding protocol. While many advanced flooding policies (e.g. randomized delay retransmission) can reduce collision, we preferred to use classical flooding technique which doesn't employ delay retransmission to achieve best latency bounds for real-time multimedia traffic delivery. Owing to this, we have chosen Video Diffusion (VD) [ZH06] as the best possible candidate to represent classical flooding technique in our experiment. VD support real-time multimedia flooding as it doesn't apply any delaying mechanisms. In VD, as explained in chapter 3 subsection 3.3, the destination node (sink) floods a "View Request" to all network nodes until it reaches the multimedia stream source node. Forwarding nodes cache the session information (i.e. Session ID) and wait for stream packets to be forwarded. Each forwarding node tracks previously seen data packets, sent by the source, by caching their sequence numbers and prevents forwarding a packet except one time. It is trivial to notice that VD represents an open flooding technique in which all nodes participate in data delivery over all possible paths. Each sent multimedia packet from the source node carries the same Session ID number used in the "View Request" of VD.

VD+FZIP

To be fair in evaluating FZIP, we evaluated its performance using VD also. As explained before, in chapter 3 subsection 3.2, the function of FZIP is to construct the flooding zone only which allows the multimedia flooding protocols to enhance performance. For that, we "re-tooled" the Video Diffusion protocol to operate with FZIP where the "View Request" process is replaced by FZIP Initialization and Acknowledgment messages. In addition, as FZIP can be configured to use different FZ sizes, we use different FZIP cases with different configurations. For example FZIP-I represents an FZIP protocol configured to use the smallest FZ-depth value in the sorted FZ Depth Table during the initialization process. In a similar way, FZIP-II, FZIP-III, FZIP-IV are configured to use second, third, and fourth value in the sorted FZ-depth Table as its FZ-depth size. Note that the FZ Depth Table is sorted in an ascending order.

5.3.2 FZIP Simulation Scenario 1 (end-to-end hop distance)

In this scenario, we simulate transferring a real-time multimedia traffic using UDP, VD, and VD+FZIP. The goal is to study the different performance for each case in terms of average packet loss and energy consumption, regarding source-to-sink hop distance (i.e. path size). We simulated a medium size grid topology of 100 stationary nodes evenly distributed in 10 rows by 10 columns (see Figure 17). Each node is 170m away from its neighbors allowing each one to communicate with 8 nodes within the default 250m signal radius. Simulation results are based on the IEEE MAC 802.11 (in DFC mode) and the AODV [RFC3561] routing protocol.

In all simulated cases (UDP, VD, and VD + FZIP), the sink node is fixed at node 42. Simulation started with a source (S1) at node 43 located at 1 hop away from the sink. In each case, we transferred 1000 packets of 512 Bytes each between source and destination nodes at a rate of 66 ms/packet (60 Kbps). We repeated the simulation for each case seven times, by shifting the multimedia data sender one hop further from the receiver up to 7 hops away (i.e. source node number is 43 to 49 as S1 to S7 respectively). Each simulation experiment was further repeated 10 time and we took the average result. In each simulation experiment, we measured the packet loss rate and average energy consumption.

The destination node (i.e. the sink) is configured to wait 50 ms after receiving the first Init-msg and before it can reply back with the Ack-msg (i.e. to construct the FZ depth Table). During this 50ms, FZIP is configured to use the second smallest received Hop-Id, stored in the FZ depth table, for setting the FZ-depth in the Ack-msg. The simulation uses the default supported NS-2 energy model where the initial power for each node is set to 1000 joules. Transmitting a packet is configured to consume twice the power of receiving.

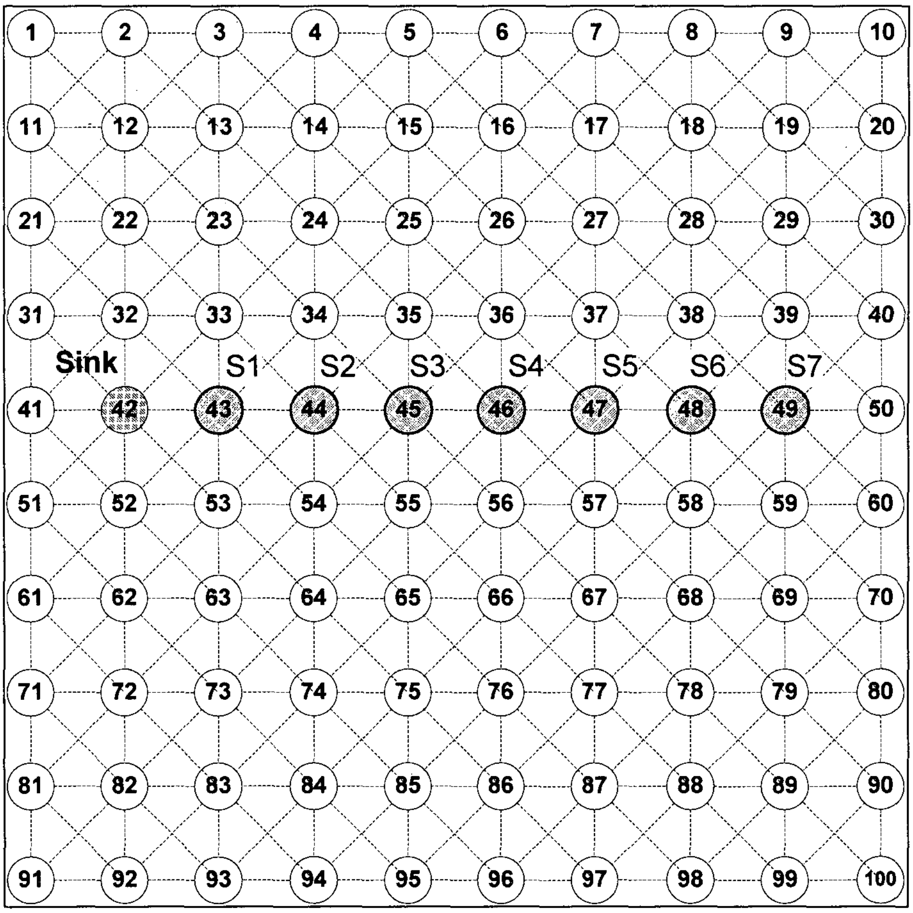


Figure 17: 100 node flat grid topology

Simulation Results (Scenario 1)

In Figure 18, the results of transferring the 1000 packets by the three different cases (UDP, VD, and VD+FZIP) are presented. As one might expect, the average packet loss ratio of all cases increases as the distance, in terms of hops, between the sources and the sink increases. However, it is observed that the packet loss ratio for UDP starts to increase rapidly after 4 hops distance. One good lesson here is the ineffectiveness of UDP, representing transportation over a single path, for real-time multimedia transportation under large scale multi-hop WSNs. The MAC layer transmission reliability and recovery mechanisms such as CTS/RST/ACK and ARQ are ineffective in such situations. In contrast to a single path scheme, VD and VD+FZIP maintained a low packet loss rate in all cases. The lower packet loss ratio by VD+FZIP and VD in comparison to UDP justifies the design paradigm of using redundant packet delivery over diverse paths as a mechanism to reduce packet loss (i.e. provide reliable delivery).

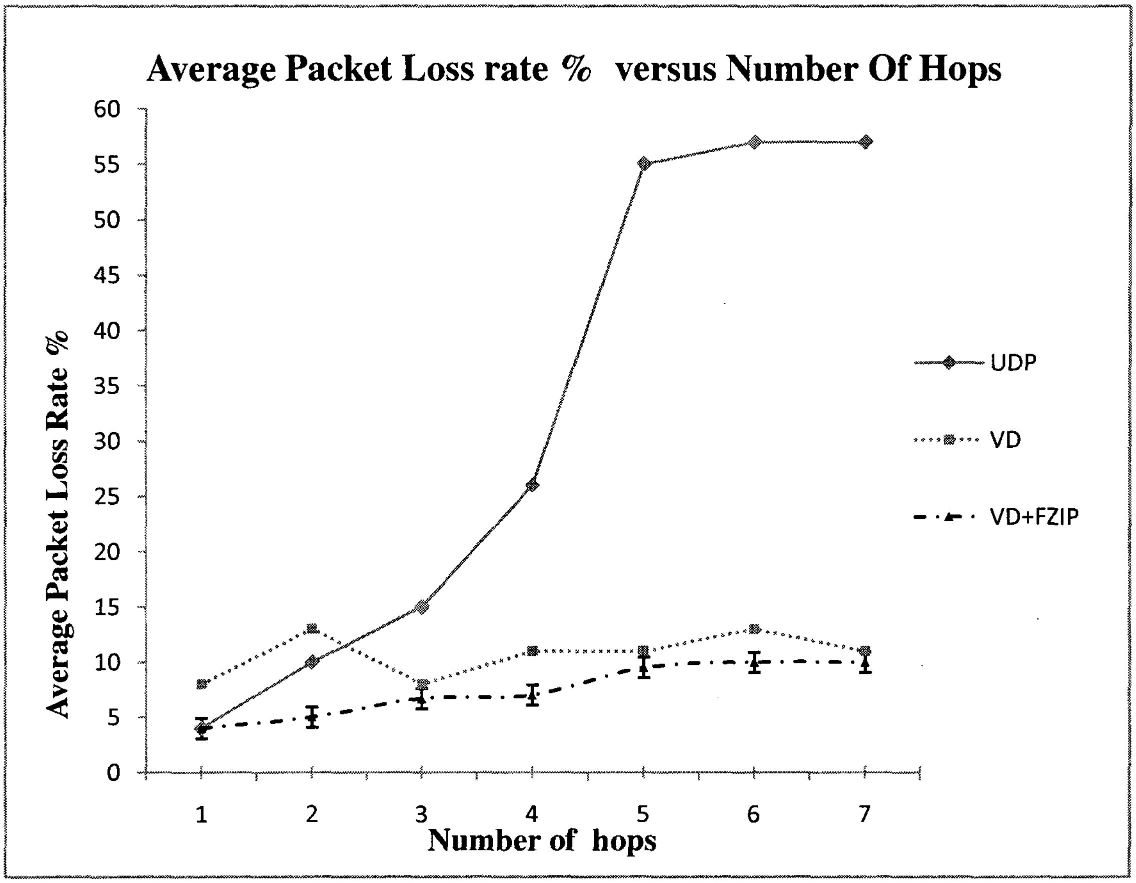


Figure 18: Average packet loss rate versus number of hops

It is also noticeable that the average packet loss rate of the VD case for the source node in close proximity to a destination nodes (i.e. 1 or 2 hops distance away) is worse than UDP (single path delivery). This particular observation makes clear that, flooding is unnecessary for delivering packets to a destination node which is few hops away. In other word, why to let all network nodes participate in delivering packets to a node which is 1 or 2 hops away? However, this blind behavior of flooding is avoided using FZIP. As shown in Figure 18, the VD+FZIP case outperform all other cases in terms of packet loss ratio by not exceeding 10% even for distance of 7 hops. The effect of the random noise and traffic on the performance of VD+FZIP has been small. The 95% confidence interval, shown as vertical error bars in Figure 18, did not exceed 1.15%. The lower packet loss ratio by VD+FZIP in comparison with VD justifies the design paradigm of

FZIP to enhance flooding reliability by constraining the broadcast storm within the flooding zone.

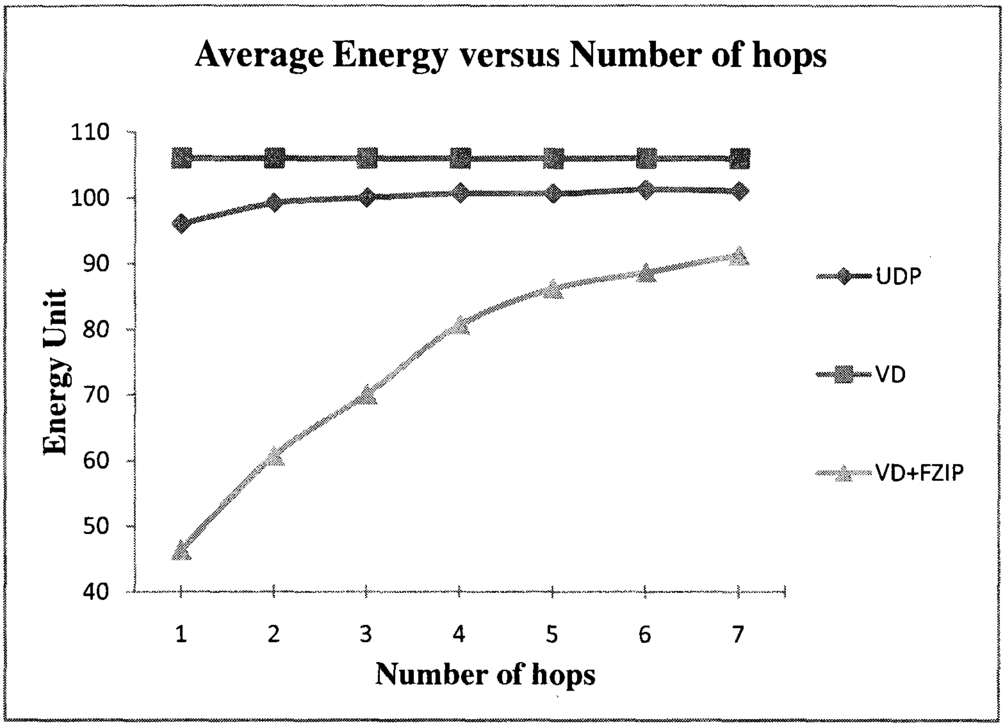


Figure 19: Average energy spending versus number of hops

Figure 19 shows the average power consumption per node at the end of each simulation. We can observe that the average energy consumption of VD+FZIP is consistently smaller than UDP and VD. In contrast to VD, where the average energy consumption is almost constant in all cases, it is shown that the average energy consumption by VD+FZIP increases linearly as the number of hops distance between the source and the sink increases. However, VD with open flooding zone performs poorly by consuming a constant large amount of power regardless of whether the source node is one or several hops away from the sink. This is due to the participation of all nodes in flooding regardless of the distance between the source and the sink nodes. The lower energy consumption by VD+FZIP in comparison with VD justifies the design paradigm of the flooding zone to minimize energy overhead without sacrificing delivery quality. It is clear that constraining the flooding storm within the carefully selected flooding zone helps flooding protocols to perform more efficiently in term of reliability and energy consumption.

Furthermore, the unexpected huge power amount consumed by UDP is due to the overhead caused by the AODV control messages and the MAC layer error recovery signaling messages. By analyzing the simulation trace files, we observe that temporary packet loss and transmission failure causes AODV to assume link failure and frequently exercise path exploration and maintenance processes. This particular result justifies our expectation that, the overhead of the data link and the network layers in case of UDP usage can be expensive and inefficient for providing multimedia delivery with good quality in WSNs. On the other hand, the multipath redundant transmission provided by the FZ can provide reliable delivery. In addition, by controlling this redundancy within suitable FZ size, the overhead of this redundancy can be less than the accumulated overhead of single path transmission. By flooding multimedia packets over the flooding zone avoid the power and the delay overheads of the data link and network layers reliability mechanisms.

5.3.3 FZIP Simulation Scenario 2 (Channel Error Rate)

In scenario 1 (see subsection 5.3.2), we studied the effect of the source-to-sink hop distance on the performance of transferring real-time multimedia data using UDP, VD, and VD+FZIP cases. In scenario 2, we study the effect of various network conditions, represented by the wireless channel error rate, on the performance of these protocols. The goal is to validate the effectiveness of our proposed FZIP protocol in providing reliable and power efficient real-time multimedia delivery even under bad WSNs network conditions. In [ZG03], an experiment for an indoor sensor network shows that, at the physical layer, half of the wireless links exhibited error rates of 10% and another third of the wireless links experienced more than 30%. Taking into consideration the deeper impact of outdoor factors on WSN deployment (e.g. military surveillance), we believe this error rate can be even higher. (e.g. weather effects, interference from other sources, etc...).

In addition, instead of using single flooding zone size for VD+FZIP in scenario 1, we study the performance of VD+FZIP using different flooding zone sizes in this scenario. Based on our theoretically analysis, presented in chapter 4, subsection 4.1.3, we claimed that as the flooding zone size increases, the power consumption and latency increase whereas the packet loss rate decreases. For this, we use 3 FZIP cases namely VD+FZIP-I, VD+FZIP-II, and VD+FZIP-III configured to use first, second, and third flooding zone sizes respectively from the FZ-Depth Table. Note that the FZ-Depth values are sorted in an ascending order in the FZ-Depth Table. The goal is to validate our expected impact of different flooding zone sizes and varying network conditions on the performance of flooding in terms of packet loss, power consumption, and latency.

In this scenario, we transferred 1000 packets of 512 Bytes each with a rate of 33 ms/packet (120 Kbps) using the five different cases: UDP, VD, VD+FZIP-I, VD+FZIP-II, and VD+FZIP-III. The remaining configurations of scenario 2 are similar to what was presented in scenario 1, subsection 5.3.2, with the following exceptions. First, we simulate a larger size grid topology of 400 stationary nodes evenly distributed in 20 rows by 20 columns. We used similar node distance same MAC configuration as discussed in scenario 1 (see Figure 20). Second, instead of shifting the source node position each time, we fixed the sink in the middle of the topology at node 208 and the source node 7 hops away from the sink at node 201. Third, we repeat the experiment for each case with increasing wireless channel error rates ranging from 0%, 5%, 10%... up to 50%. The purpose of this, as mentioned previously, is to study the performance variation against different network conditions (i.e. good, moderate, and bad network conditions). We ran the simulation in each case (i.e. UDP, VD, VD+FZIP-I, II, and III) 100 times and took the average result of average packet loss rate, average energy cost, and average latency.

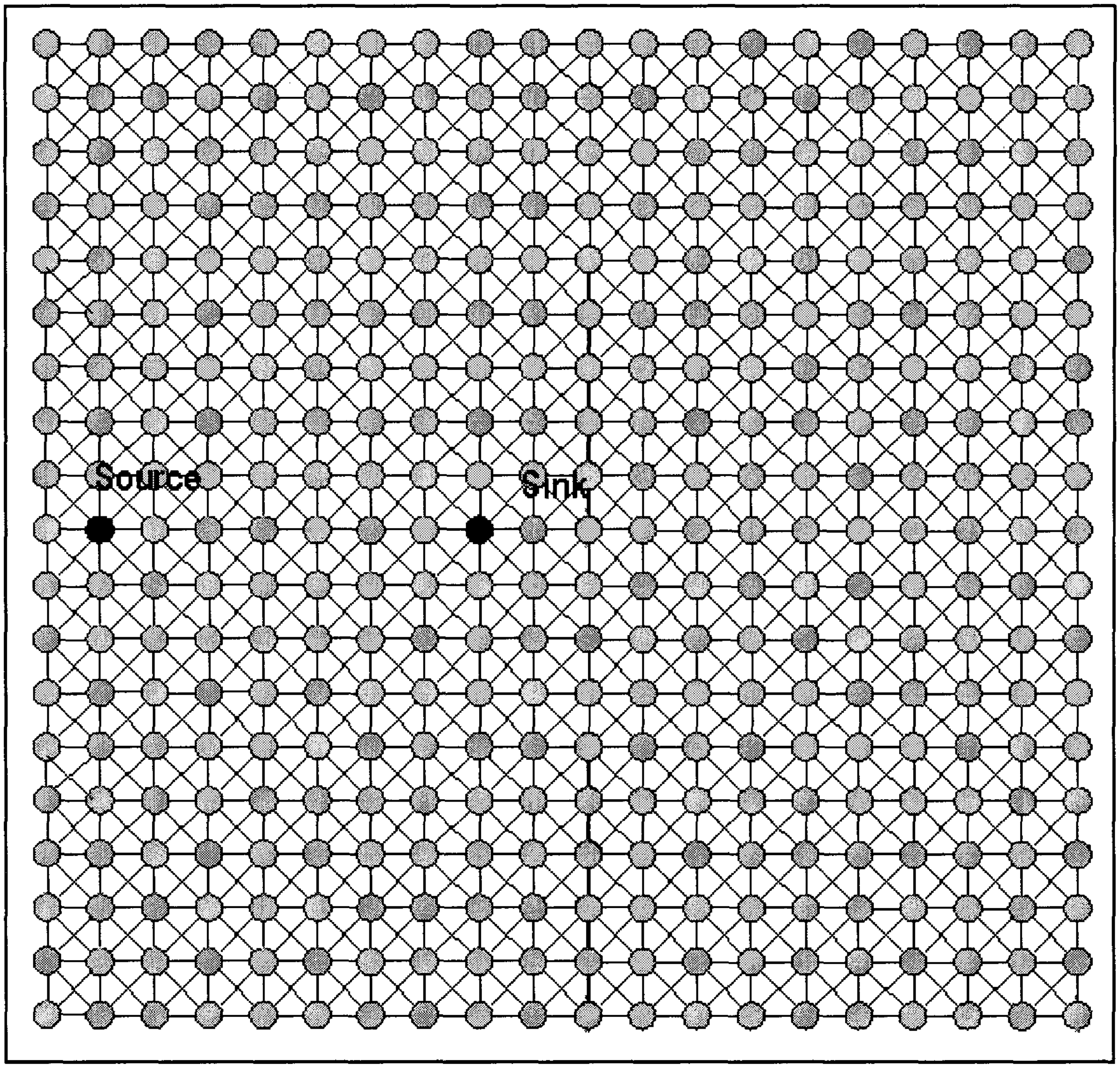


Figure 20: A 20 x 20 flat grid topology

Simulation Results (Scenario 2)

Average Packet Loss Rate

In Figure 21, we present the results of multimedia stream transfer for the five cases (UDP, VD, VD+FZIP-I, VD+FZIP-II, and VD+FZIP-III) with increasing wireless channel error rates. As predicted, the packet loss rate increases as the network channel error rate increases. For UDP, the packet loss rate increases linearly up to channel error rate of 10%. For channel error rates larger than 10%, the packet loss rate starts to increase rapidly reaching 50% loss for channel error rate of 15%. In situations of high channel error rates of 35% and higher, the packet loss of UDP is 99% and fails to reach destination for error rates of 40% and more. This inefficient performance of UDP confirms our argument that UDP is not suitable to be used for multimedia transportation in large and error prone multi-hop wireless networks. The MAC layer reliability mechanisms are not efficient enough in providing good packet delivery over large networks with high channel error rates.

In contrast to a single path delivery by UDP, flooding over multipath appears to be resilient to wireless channel errors. The average packet loss of VD is only 12% in networks with channel error rates ranging between 0% to 20%, and slightly increases to 14% even under very high channel error of 40%. Under extreme channel error rates of 50%, VD experiences packet loss of 24%.

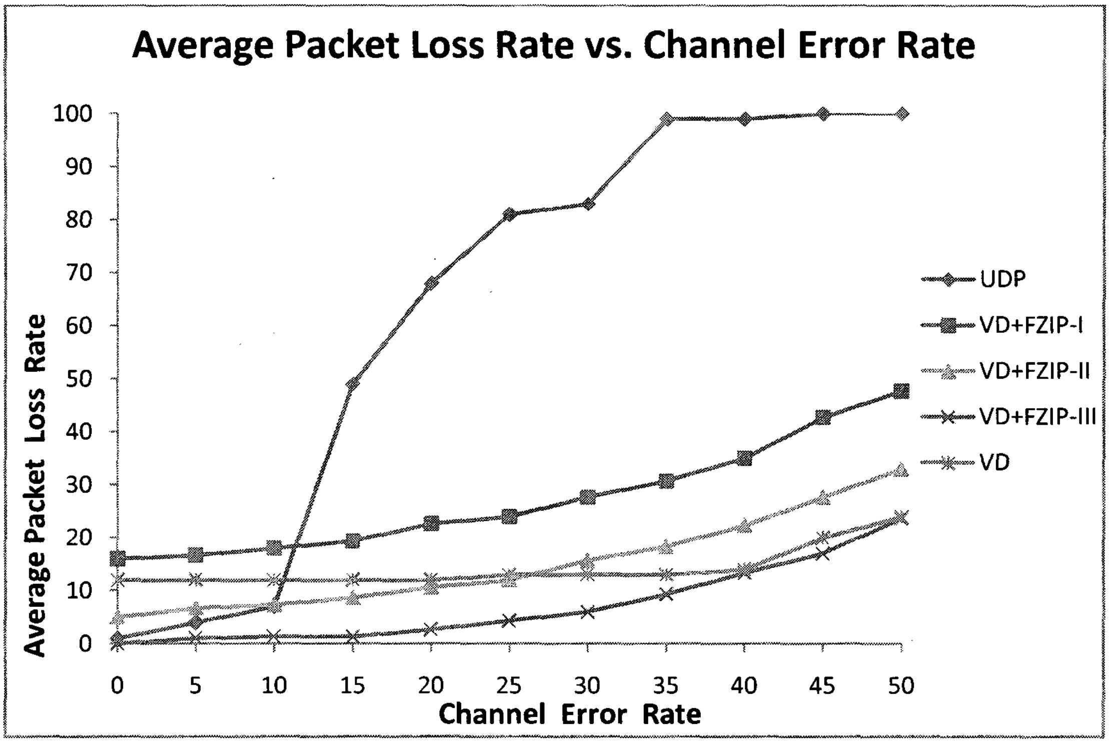


Figure 21: Average packet loss rate as a function of Channel Error Rate

As for VD+FZIP-I, representing flooding over the smallest possible flooding zone, the packet loss rate with channel error 0 is 16% which is relatively higher than all other cases. Note that average packet loss rate for VD is 12% at channel error rate 0. That is expected since the size of the flooding zone is small and the redundancy is minimal, while VD uses all nodes in flooding. However, the packet loss rates of VD+FZIP-I did not exceed 20% with moderate wireless channel error rate of 15% and still delivers 70% (i.e. 30% average packet loss rate) of the stream in networks with high channel error of 35%.

By increasing the flooding zone depth size, it is shown that some lower average packet loss rates of VD+FZIP-II and VD+FZIP-III are achieved compared with VD. VD+FZIP-II outperforms VD by achieving lower packet loss rate up to channel error rate of 25% and VD+FZIP-III outperforms VD in all cases even with channel error rate of 50%. VD+FZIP-III achieved very low packet loss rate of 1.33% at channel error rates up to 15% and less than 5% packet loss rate under channel error rate of 25%. For extreme channel error rate of 50%, VD+FZIP-III achieved less than 24% packet loss rate.

It is shown from Figure 21 that VD gives more constant packet loss rate desipte the increase of channel error rate, while VD+FZIP I, VD+FZIP II, and VD+FZIP III packet loss rates increase relatively at a higher pace. Nevertheless, the packet loss rate for VD starts with packet loss error rate of 12% which is relatively higher in comparison to UDP, VD+FZIP-II, and VD+FZIP-III up to channel error of 10%. This limitation makes VD (i.e. open flooding) less efficient in terms of packet loss for networks with low channel error rates (i.e. good network conditions). This is because, in good network conditions, packet loss is low and UDP can achieve better delivery rates. The associated overhead of the redundant delivery of VD is not necessary.

Power Overhead

Figure 22, shows the average power consumption per node at the end of each simulation. In perfect wireless channel conditions (channel error rate 0%), UDP consumes least energy in comparison with all other protocol cases. That is expected as the perfect network condition requires no link-layer recovery control messaging. However, average power consumption for UDP spikes sharply at channel error rate of 5%, over 171 joules, exceeding all VD+FZIP cases (i.e. VD+FZIP-I, II, and III). The average power consumption for UDP in all wireless channel errors of 10% and more is way higher than the average power consumption of all VD+FZIP cases, and somewhat less than VD. As the channel error rate increases, the associated power cost of UDP increases. The inefficient average power consumption of UDP under channel error rate of 5% and higher is due to the associated cost of the link-layer loss recovery mechanisms (i.e. RTS/CTS/ACK and ARQ) and routing protocol control messaging[11].

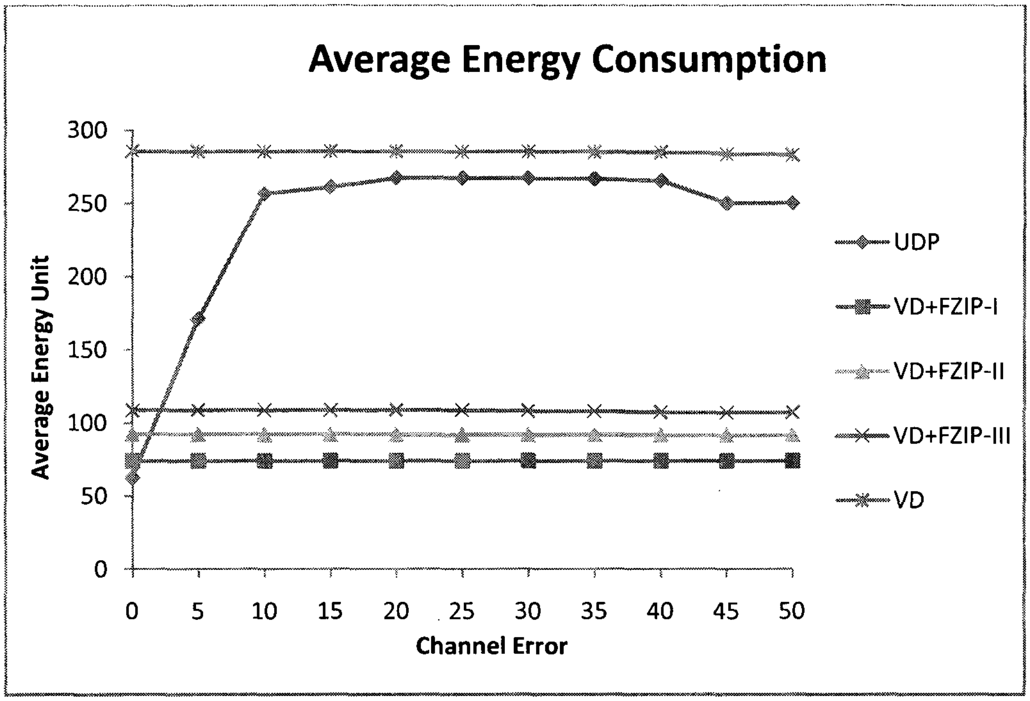


Figure 22: Average Energy consumption

In contrast, average power consumption associated with all VD and VD+FZIP cases consume, to some extent, constant level of energy. However, it is noticeable that as the FZ depth increases, the power consumption increases. VD+FZIP-I representing smallest flooding zone, consumes least power in comparison with VD+FZIP-II and VD+FZIP-III (i.e. with larger flooding zone sizes). In the same way, VD+FZIP-II consumes less power in comparison with VD+FZIP-III and VD without FZ.

The lower energy consumption by VD+FZIP-I, II, and III in comparison with VD justifies the design paradigm of flooding zone to minimize energy.

Latency (delay)

[11] When the link layer, at a specific node, fails to deliver the packet correctly due to the channel errors, the AODV routing protocol assumes that the node no longer exists, therefore AODV initiates a new path exploration process to find an alternative route.

Figure 23 shows the average latency results for simulation with channel error of 0%. VD+FZIP-I has the smallest latency factor. It is shown in Figure 23 that as the flooding zone size increases, the latency of packet delivery increases. This is because of the increase of the paths length in the larger FZs (i.e. number of hop-by-hop source to sink path). For example, a received packet at a sink node over a 10 hop paths will take more time to reach the destination than a packet flowing over a shorter number of hops path. VD+FZIP-I (i.e. the smallest flooding zone) delivers packets in less time than all other protocols cases including UDP. In UDP, the link layer CTS/RTS/DATA/ACK communication pattern adds extra time in comparison with direct broadcast.

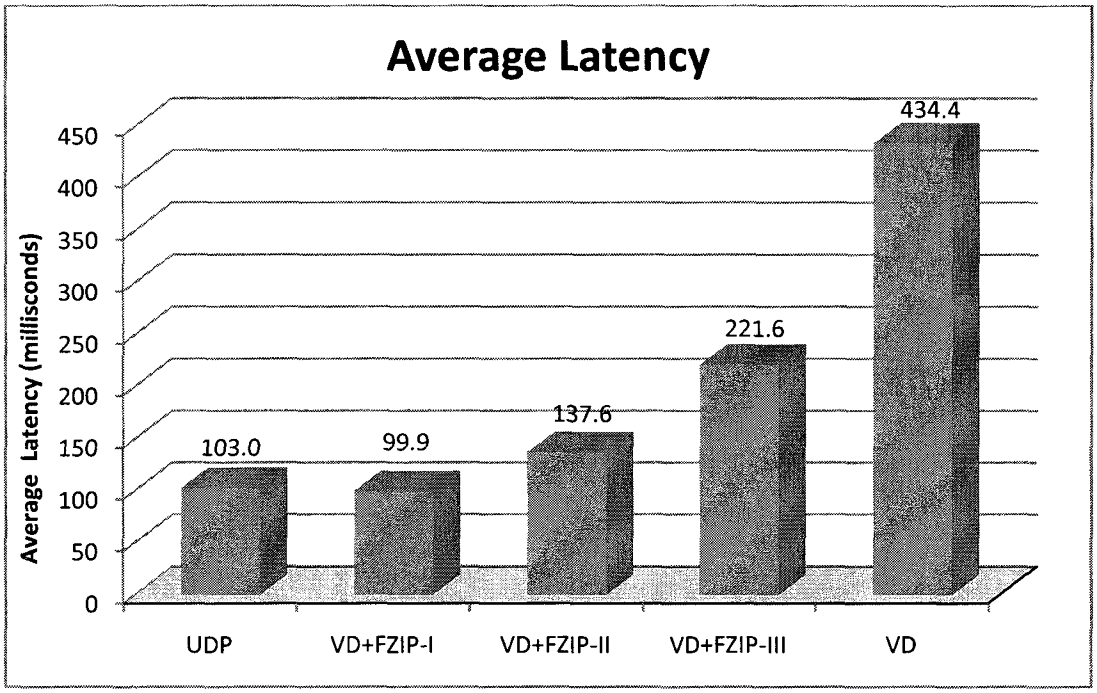


Figure 23: Average packet delivery latency by the simulated protocols

In summary, it is observed that the FZ size has an impact on the performance of data flooding. By comparing the performance of flooding using the 3 different flooding zone sizes (i.e. VD+FZIP-I, II, III), it is shown that the VD flooding protocol achieves different performance levels in terms of latency, power and reliability (i.e. packet loss). As the flooding zone size increases, power overhead and latency increases while packet loss decreases. This validates our claims during the design phase.

In addition, it is observed that in order to achieve best desired performance, a tradeoff of these performance metrics is required. For achieving this, we need to answer the question "How can FZIP select a suitable FZ size?" Moreover, in situations of dynamically changing network conditions, having a fixed FZ size throughout the duration of the session will lead to less efficient performance. These two particular observations are the motivation of our complementary FZCP protocol. In next subsection 5.4, FZCP simulations are presented to justify this.

5.4 FZCP Simulation

Recall that the FZIP role finishes after initializing the FZ. Therefore, the initialized FZ size remains fixed for the whole session period. If the selected FZ is small, the packet delivery quality might not be sufficient for the application (i.e. packet loss is high). However, if the selected FZ size is large, unnecessary extra power may be consumed. In addition, in situations where the network conditions dynamically change, having a fixed FZ size will not provide an acceptable level of quality along the session lifetime. FZCP aims to solve these two problems by continuously monitoring the performance and switching to the suitable FZ size according to current network state. The selected FZ size is determined based on a tradeoff of good level of quality and less power consumption.

The goal of the simulated experiments in this subsection is to justify the FZCP design paradigm. More specifically, we want to examine, by simulation, the ability of FZCP to:

- Compute the suitable flooding zone size which delivers acceptable level of quality and less power consumption.
- Maintain an acceptable level of quality during the lifetime of the multimedia session.

5.4.1 FZCP Simulation Approach

In order to justify the above mentioned goals, we study the performance of transferring a real-time multimedia stream using VD+FZIP and VD+FZIP+FZCP[12] in two different scenarios. In each scenario, we use different cases of VD+FZIP, configured to use different flooding zone sizes (i.e. VD+FZIP-I, II, III, and IV). In each scenario, we study the performance of the different cases in terms of average packet loss and average power consumption.

In the first scenario (subsection 5.4.2) we compare the performance of VD+FZIP and VD+FZIP+FZCP in a real-time multimedia session running over a network with stable (i.e. not dynamic) network condition. This will simplify the comparison of the best achieved performance by the different used VD+FZIP cases to the performance of VD+FZIP+FZCP case. In the second scenario (see subsection 5.4.3), we compare the performance of VD+FZIP and VD+FZIP+FZCP for running a real-time multimedia session over dynamically changing network condition. This will highlight the need to have a dynamically changing FZ size during the session lifetime to reduce power consumption.

5.4.2 FZCP Simulation Scenario 1 (Unchanging Network Conditions)

In this experiment, we evaluate the ability of VD+FZIP+FZCP to achieve similar performance in comparison to the best result among the performance achieved by the different VD+FZIP cases (i.e. VD+FZIP-I, II, III, and IV). That is to say, the goal is to validate the ability of FZCP to choose the best flooding zone size to deliver good level of quality (i.e. packet loss rate doesn't exceed certain level) and lowest associated power consumption.

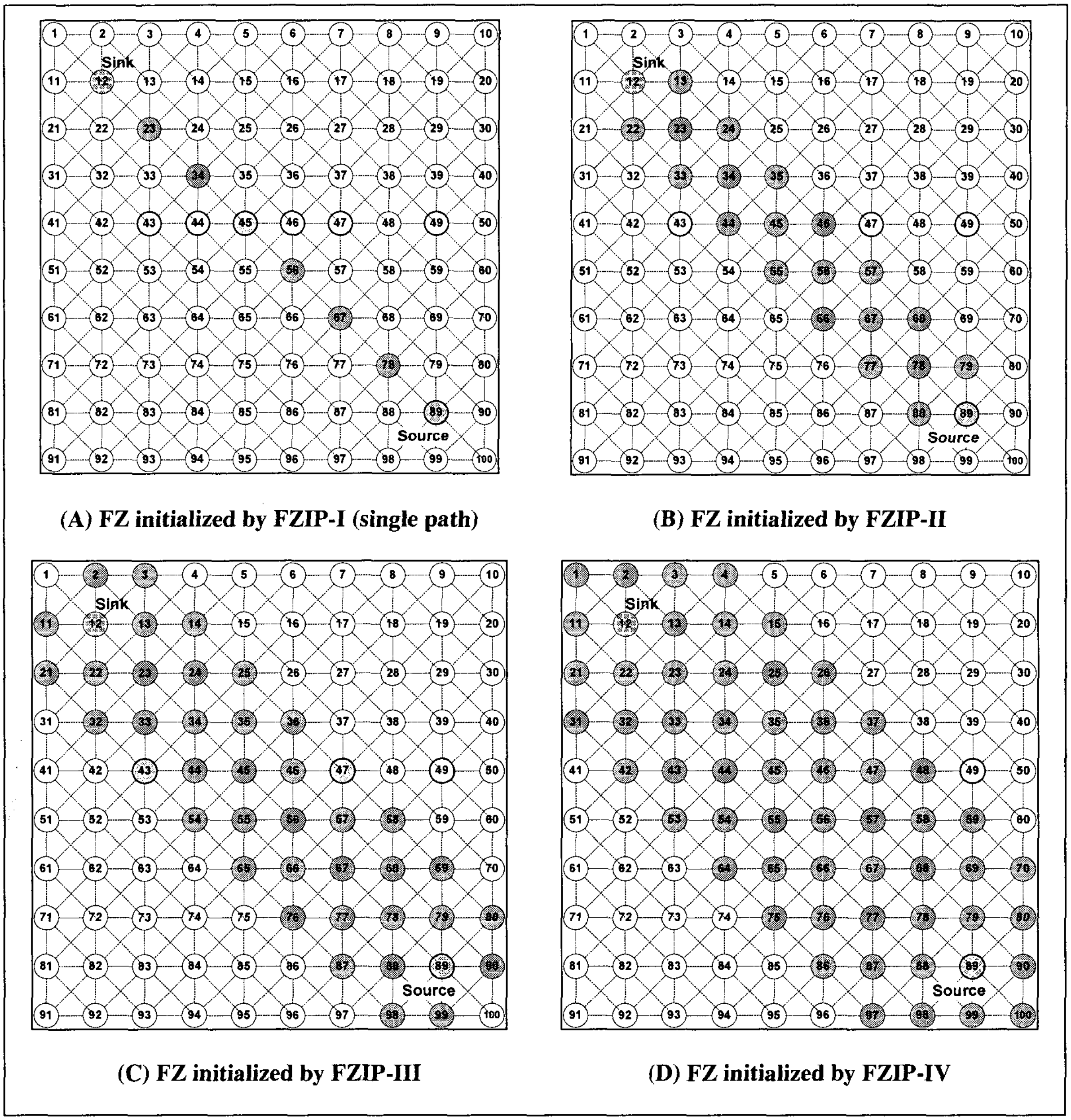
In order to do this, we break up the simulation in this scenario into two stages. In stage 1, we compare the performance of VD+FZIP using different flooding zone sizes to identify the best possible performance that can be achieved. This will demonstrate the

[12] Note that VD+FZIP+FZCP means the session is using the VD flooding protocol and FZIP setup protocol at the beginning. The session runs FZCP after FZIP setup until the end of the session.

problem of how to choose the best possible flooding zone size to deliver good quality while consuming less power. To do this, we manually examine the performance achieved by different cases of VD+FZIP configured to use different flooding zone sizes namely: VD+FZIP-I, VD+FZIP-II, VD+FZIP-III, FZIP-IV. The VD+FZIP-I, VD+FZIP-II, VD+FZIP-III, and VD+FZIP-IV are configured to use the first, the second, the third, and the fourth possible flooding zone sizes respectively (i.e. the first, second, third, and fourth hop field values in the sorted FZ-depth table respectively). By comparing the performance metrics of these different VD+FZIP cases, we can manually identify the flooding zone size which provides the best possible performance. In stage 2, we evaluate VD+FZIP+FZCP performance according to the best result achieved in stage 1. This will justify the ability of FZCP in providing efficient tradeoff of good quality and less power consumption.

We simulated, in this scenario, a medium size grid topology of 100 stationary nodes evenly distributed in 10 rows by 10 columns. Each simulation lasts for 60 seconds. In all simulated cases, the sink node is fixed at node 12 and the source node is 7 hops away at node 89 (see Figure 24). For each case, we transferred 3600 packets of 370 Bytes each between the source and the destination nodes at a rate of 1 packet every 16.666 milliseconds.

As for FZIP configuration parameters, the destination node is configured to wait 50 ms after receiving the first Init-msg and before it can reply back with the Ack-msg. During the 50 ms, the destination node constructs and sorts the FZ Depth Table in an ascending order. The different VD+FZIP cases select the respective FZ-depth value from the sorted FZ Depth Table. VD+FZIP-I represents the smallest FZ size with least reliability as the FZ size in this case represents a single path (see Figure 24-A). Figure 24 shows the different flooding zone sizes initialized by the different FZIP versions.



(A) FZ initialized by FZIP-I (single path)

(B) FZ initialized by FZIP-II

(C) FZ initialized by FZIP-III

(D) FZ initialized by FZIP-IV

Figure 24: A 100 node flat grid topology

Remember that FZCP role is to control the FZ size. FZCP performs three functionalities: performance monitoring, tradeoff adaptation function, and flooding zone switching operations. The monitoring function, running at the sink node, calculates the fraction of packet loss every T time interval. In our FZCP simulation case, we set the time interval with 1 second. If the calculated fraction of packet loss exceeds the Maximum Packet Loss Threshold (MAX_PLTH), then the tradeoff and adaptation function decides on correcting the bad performance by switching to larger FZ-size.

However, if the fraction of packet loss is less than Minimum Packet Loss Threshold (MIN_PLTH), then the adaptation function decides to switch to a smaller FZ size, thus reducing power.

In our VD+FZIP+FZCP case simulation, we set the MAX_PLTH to be 20% while the MIN_PLTH is set to 0%. That is to say, as long as the fraction packet loss of the incoming multimedia traffic is less than 20%, the multimedia quality is considered good. If the fraction packet loss rate exceeds 20%, the multimedia quality is considered bad and FZCP will send +FZ Resize message to enlarge the FZ size. In addition, FZCP is configured to start with the smallest FZ depth used during the FZIP initialization phase.

To simulate bad network conditions, each simulated case is configured with fixed channel error rate of 30% in each simulation. We measured the average packet loss rate, average energy consumption, and average latency for each simulation. We recorded the packet loss rate of the delivered multimedia traffic every 1 second to a trace file. We ran the simulation for each case 10 times and took the average result.

Simulation Results (Scenario 1)

Table 7 shows the simulation results of average packet loss, average power consumption, and average latency for all simulated VD+FZIP cases. It is shown in Table 7 that, as the FZ size increases, the average power consumption and latency increase. As VD+FZIP-I has the smallest FZ size, it consumes less power and delivers packets sooner than larger FZ sizes (i.e. VD+FZIP-I provides smaller delivery latency). In addition, as the FZ size increases, average packet loss decreases (i.e. quality increases). However, the increase of the FZ size will reduce packet loss up to a certain limit, after which, the packet loss will relatively increase. As shown in Table 7, average packet loss decreases as the FZ size increases until it reaches FZIP-IV. That is to say, increasing the FZ to a large size, as in VD+FZIP-IV, result in a FZ size relatively equivalent to open flooding. It is clear from Figure 24-D that in VD+FZIP-IV, the sink has 8 neighboring nodes. This will increase contention of the shared wireless channel and increase collisions.

	Average Packet loss %	Average Power (J)	Average Latency (ms)
VD+FZIP-I	91.1	168.87	62.2
VD+FZIP-II	11.1	195.02	86.3
VD+FZIP-III	3.4	216.57	124.8
VD+FZIP-IV	7.4	232.77	186.1

Table 7: Average packet loss, average power, and average latency simulation results

As show in Table 7, VD+FZIP-I, representing the smallest flooding zone size, failed to deliver good quality. VD+FZIP-I delivered only around 10% of the data packets successfully (i.e. average packet loss is 91%). The bad reliability of FZIP-I is due to the small flooding zone size, which is in this specific situation, a single path (i.e. no redundancy). For this reason, VD+FZIP-I can't represent the best possible FZ size as the average packet loss exceeds 20% which is the MAX_PLTH. However, all the other cases achieved an average packet loss rate lower than 20% (i.e. good quality). It is shown in Table 7 that, VD+FZIP-II achieves best possible trade-off performance in comparison to all other cases. From table 7, we find that VD+FZIP-II, VD+FZIP-III, and VD+FZIP-IV all achieve good quality (i.e. less than 20%). However, VD+FZIP-II consumes the least power and latency in comparison with VD+FZIP-III and VD+FZIP-IV. Accordingly, the second smallest flooding zone initialized by VD+FZIP-II is considered the best possible Flooding Zone for this scenario.

	Avg. Packet loss %	Avg. Power Spending (J)	Avg. Latency (ms)
VD+FZIP+FZCP	13.6%	196.57	84.5
VD+FZIP-II (best performance)	11.1%	195.02	86.3
Difference	2.5	1.55	-1.8

Table 8: Comparison of VD+FZIP-II and VD+FZIP+FZCP simulation results

By comparing the performance result of VD+FZIP+FZCP with the performance of the best possible FZ we can evaluate its effectiveness. Table 8 shows the result of VD+FZIP+FZCP simulation in comparison with VD+FZIP-II (i.e. best possible FZ performance). As shown in Table 8, by using VD+FZIP+FZCP, close performance is achieved.

Table 8 shows that VD+FZIP+FZCP achieved close performance metrics in comparison with the best possible results by VD+FZIP-II. The average packet loss of VD+FZIP+FZCP is 13.6% which is only 2.5% more than the optimal value of 11.1% by FZIP-II. The difference in average power consumption is only 1.55 joule more and the average latency is 1.8 ms less. The small performance difference of VD+FZIP+FZCP in comparison with the best possible performance is due to the FZ resize process overhead. Figure 25 shows the fraction of packet loss at every 1 second of the simulation lifetime for VD+FZIP-I, VD+FZIP-II, and VD+FZIP+FZCP. As shown, VD+FZIP+FZCP started with the smallest FZ size, and then switched at end of second 1, (see Figure 25), to the next FZ size in the FZ Depth Table (i.e. same FZ size as used by VD+FZIP-II). By excluding the 1st second result from our comparison, identical performance is achieved. That is to say, from second 2 to end of session lifetime, VD+FZIP+FZCP and VD+FZIP-II performance is identical.

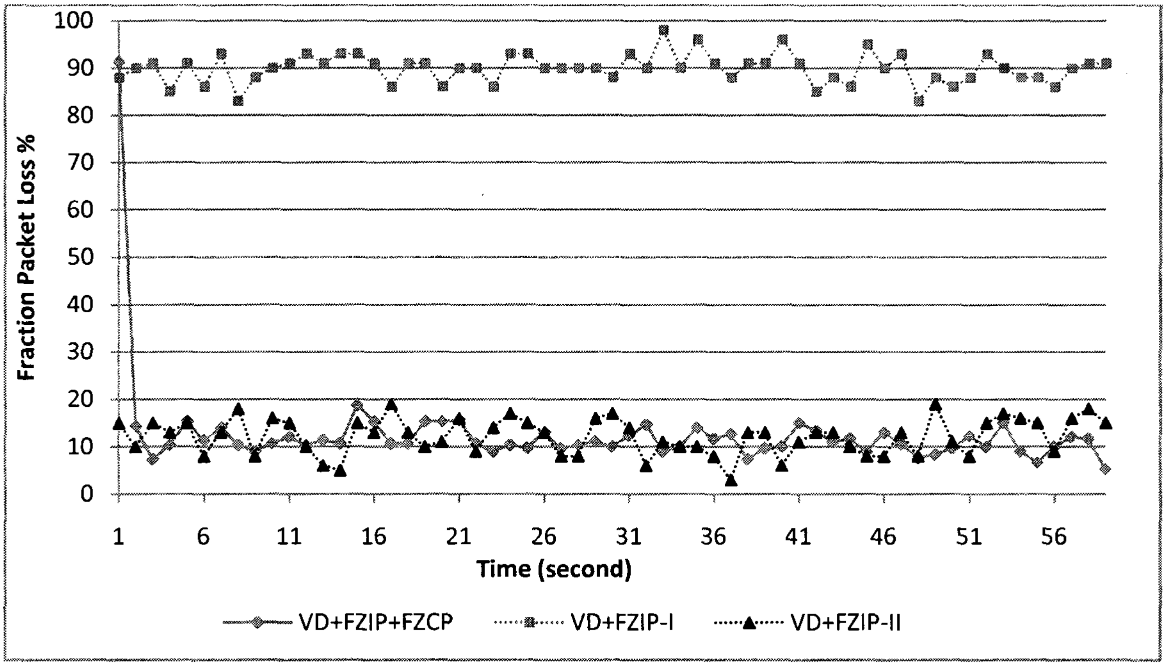


Figure 25: Fraction of packet loss % versus time intervals (seconds)

In Figure 25, it is shown that the fraction of packet loss of VD+FZIP-II and VD+FZIP+FZCP is close and not exceeding 20% except at second 1. During second 1, both VD+FZIP-I and VD+FZIP+FZCP performed poorly by not delivering 10% of the multimedia data as both of the protocols are using the smallest FZ size (i.e. single path). However, the monitoring ability of VD+FZIP+FZCP detected this bad performance at the end of second 1. To overcome this bad quality, VD+FZIP+FZCP initiates a +FZ Resize message to enlarge the FZ from smallest FZ size to second smallest FZ size. At second 2 and all subsequent time intervals, VD+FZIP+FZCP did not detect any bad quality. Thus, VD+FZIP+FZCP did not need to change the FZ size until the end of session lifetime. The high packet loss at second 1 and the associated overhead of the +FZ Resize-msg explain the little performance difference between VD+FZIP+FZCP and VD+FZIP-II. This difference will decrease for longer multimedia session lifetime.

In this scenario, we fixed the channel error rate to 30% during the session lifetime. The best possible flooding zone size changes because of several factors (e.g. changing wireless channel error rate during the session, source-to-sink hop distance, and the density). Having a fixed FZ size will lead to less optimized performance without FZCP. In the next scenario, subsection 5.4.3, we evaluate FZCP effectiveness in dynamically changing network conditions.

5.4.3 FZCP Simulation Scenario 2 (Changing Network Conditions)

The purpose of this simulation scenario is to evaluate the ability of the FZCP protocol to save power while maintaining an acceptable level of quality for real-time multimedia delivery. In order to do that, we compare four different cases: VD+FZIP-I, VD+FZIP-II, VD+FZIP-III and VD+FZIP+FZCP. We simulated a medium size grid topology of 100 stationary nodes evenly distributed in 10 rows by 10 columns (see Figure 26).

In each simulation, we transferred 3600 packets of 370 Bytes each between the source and the destination nodes at a rate of 1 packet every 16.666 ms/packet (103 Kbps). Each simulation lasts for 60 seconds. In all simulated cases, sink node is fixed at node 42, while the source node is located 6 hops away at node 48 (see Figure 26).

The destination node is configured to wait 50 ms after receiving the first Init-msg and before it can reply back with the Ack-msg. During this 50 ms, the FZ depth Table is constructed. VD+FZIP-I, VD+FZIP-II, VD+FZIP-III are configured to use the first, the second, and the third values stored in the sorted FZ depth table respectively.

For FZCP, we set the MAX_PLTH to 9% and the MIN_PLTH to 3%.That is to say, as long as the fraction of packet loss of the incoming multimedia traffic is between 3% and 9%, the multimedia quality is considered "good". If the fraction of packet loss rate exceeds 9%, the multimedia quality is considered "bad".[13] If the fraction of packet loss drops below 3%, the quality is considered "very good". Saving power, in this case, would be desirable.

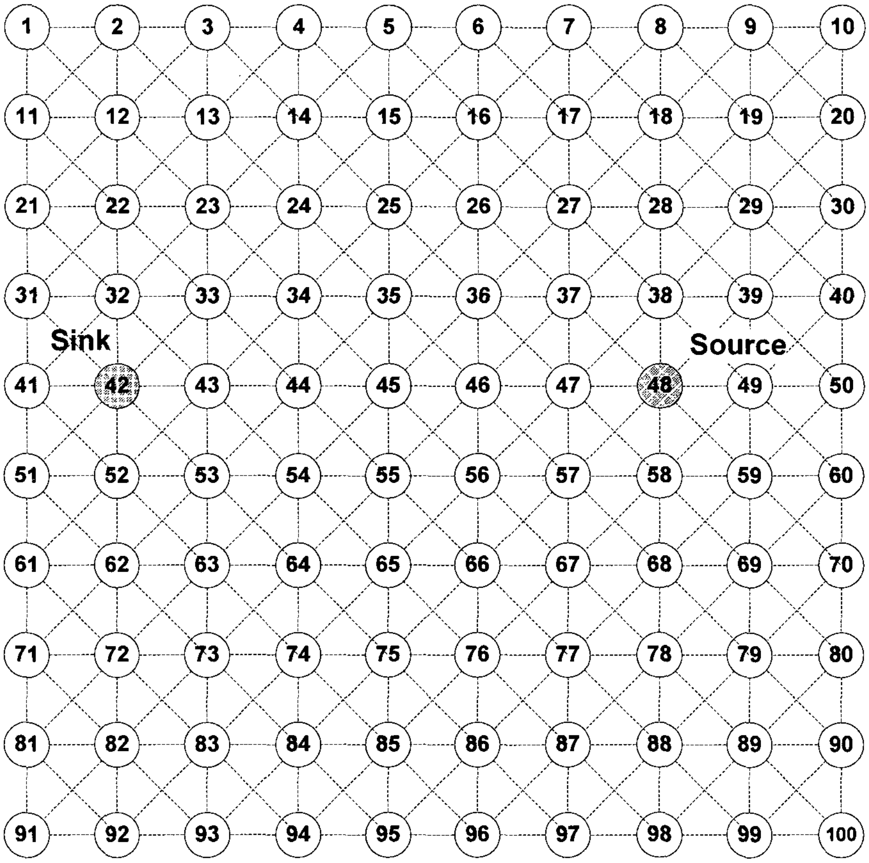


Figure 26: A 100 node flat grid topology

[13] Packet loss rates up to 20% can be tolerated for audio [KR03]. We have used 9% for MAX_PLTH as an example to test FZCP performance with, somewhat, more strict requirements.

To examine the FZCP design and validate its ability to maintain good quality throughout the lifetime of the real-time multimedia traffic, we study its performance under changing network conditions. To simulate dynamic network conditions we configure different wireless channel error rates during the lifetime of a session. The network condition during the 60 seconds session life time is divided into three states. The simulation starts with moderate channel error of 10% until second 9 of the simulation. At second 10, we increase the channel error to 30% to simulate bad network conditions and test the ability of FZCP to respond correctly. The channel error rate remains at 30% for 20 seconds (i.e. second 10 - second 29). At second 30, the channel error rate drops back to 10% and remains so till the end of the simulation time (i.e. second 30- second 60).

This fluctuation in wireless channel error rate as well as the tight good quality range (i.e. fraction of packet loss between 3% - 9%) gives a challenging simulation platform to simulate FZCP performance in real life scenarios. We ran the simulation for each case (VD+FZIP-I, VD+FZIP-II, VD+FZIP-III, and VD+FZIP+FZCP) 10 times and took the average result. In each simulation experiment, we measure the average fraction of packet loss rate at each 1 second time interval, the average power consumption, and the average delivery overhead (explained in subsection 5.2).

Simulation Results (Scenario 2)

In order to simplify the demonstration of the results, we first present the simulation results of the three VD+FZIP cases (i.e. VD+FZIP-I, VD+FZIP-II, and VD+FZIP-III). Figure 27 shows the average fraction of packet loss rate of the different cases at every 1 second time interval up to second 40 of the session lifetime. The upper and lower dashed lines represent the maximum and minimum packet loss thresholds (i.e. MAX_PLTH and MIN_PLTH) respectively.

To better interpret the plotted chart, recall that the network conditions become bad between 10 and 29 second time interval where the channel error rate becomes 30%. During this bad network condition, it is clear the increase in fraction of packet loss incurred by the different cases.

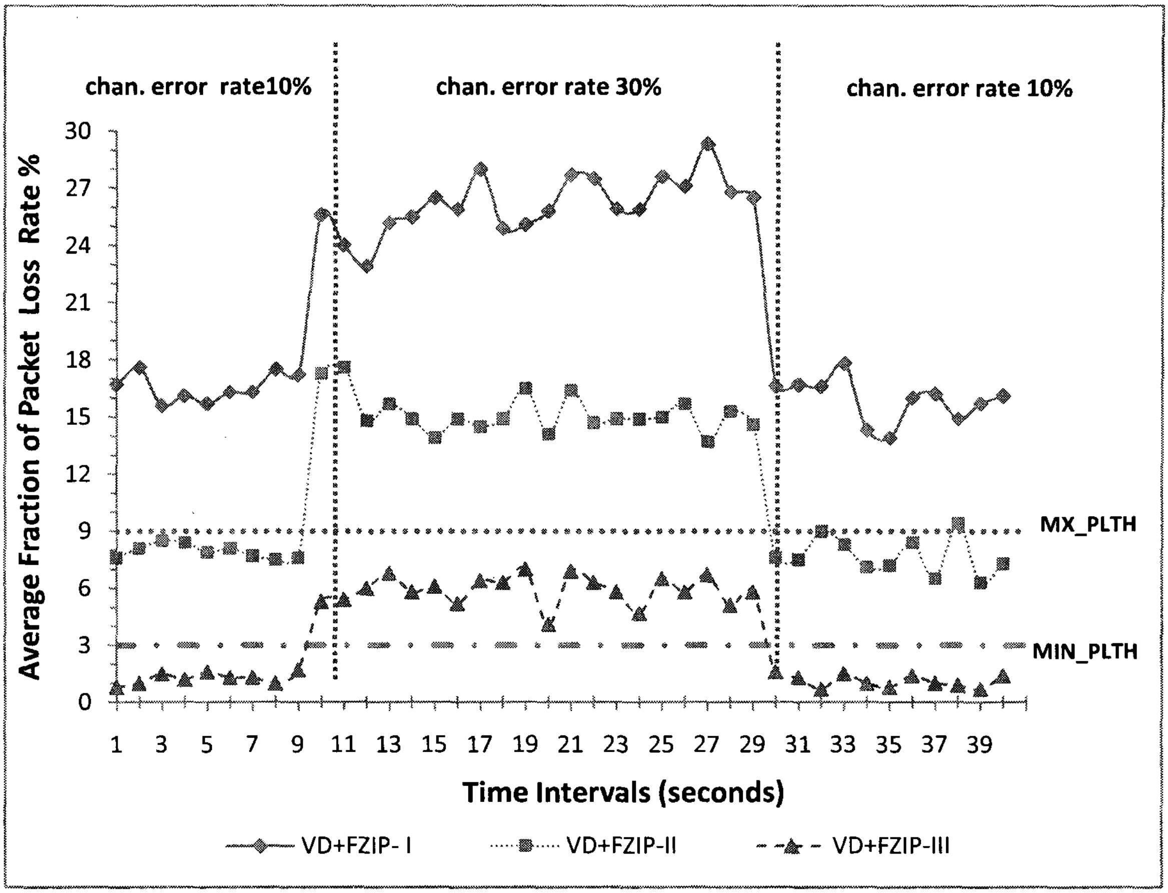


Figure 27: Average fraction of packet loss at every 1 second time intervals

As shown in Figure 27, up to second 9 (i.e. during the moderate channel error rate of 10%), VD+FZIP-I provides fraction of packet loss ranging between 15 and 18 percent and exceeding the maximum packet loss threshold. This poor performance gets worse during the bad network condition period (i.e. time intervals sec 10-sec 29) by exceeding 22%. For the remaining session lifetime, second 30 - second 60 time interval, (Figure 27 is showing only results up to second 40), packet loss of FZIP-I drops back in the range between 13% and 18%. It is clear that VD+FZIP-I cannot deliver acceptable level of quality for the whole session period. This is expected as VD+FZIP-I uses the smallest FZ size and the network condition is not good (i.e. wireless channel error rate is 10% or higher).

VD+FZIP-II, as shown in Figure 27, provides better delivery quality than VD+FZIP-I. For the time duration, second 1 to second 9 time interval, VFZIP-II delivers the required level of quality by not exceeding the maximum packet loss threshold of 9% most of the time. However, at second 10, when the network condition becomes bad (i.e. channel error rate of 30%), VD+FZIP-II performs poorly and exceeds the maximum packet loss threshold. This performance continues till second 29. For time duration of second 30 to second 60, when the channel error rate goes back to 10%, the average fraction of packet loss drops back to the good quality level of 9% most of the time. Overall, VFZIP-II could not honor good level of quality for 20 seconds continuously.

For the third case, VD+FZIP-III provides good quality for the whole session lifetime. During the moderate network condition of 10% channel error rate (i.e. second 0 - second 9 and second 30- second 60), the average fraction of packet loss of VD+FZIP-III is around 2%. Although, this average fraction of packet loss increases during the bad network condition (i.e. between seconds 10-29), the obtained quality doesn't exceed the 9% of maximum packet loss threshold. Thus, VD+FZIP-III is the only case which can deliver good quality for the whole session lifetime.

Figure 28, presents the average fraction of packet loss for FZCP. The upper and lower dashed lines represent the maximum and minimum packet loss thresholds (MAX_PLTH and MIN_PLTH) respectively. Recall that FZCP starts with smallest FZ size. At second 1, FZCP monitoring function detects that the fraction of packet loss is high, around 16%, and exceeds the maximum fraction packet loss of 9%. In order to enhance quality, FZCP initiates a +FZ Resize-msg to increase the FZ size. At second 2 and after the FZ size enlargement, the fraction of packet loss becomes lower than 9% and thus VD+FZIP+FZCP provides good quality. At second 10, where the bad network condition period starts, packet loss spikes to 12% causing bad quality. As a result, FZCP tries to enhance the quality by initiating another +FZ Resize-msg to enlarge the FZ size. As a result, during the time from second 10 to second 29, FZCP maintains good quality where the average fraction of packet loss does not exceed the maximum packet loss threshold. After the end of the bad network condition period (i.e. at second 30), the average fraction of packet loss of FZCP drops down below the minimum packet loss threshold of 3%. As a result, FZCP tries to save power by initiating a –FZ Resize-msg, thus reducing the FZ size. From second 30 to the end of the session lifetime, FZCP maintains good quality with the current FZ size where the average fraction of packet loss doesn't exceed the minimum and maximum packet loss thresholds.

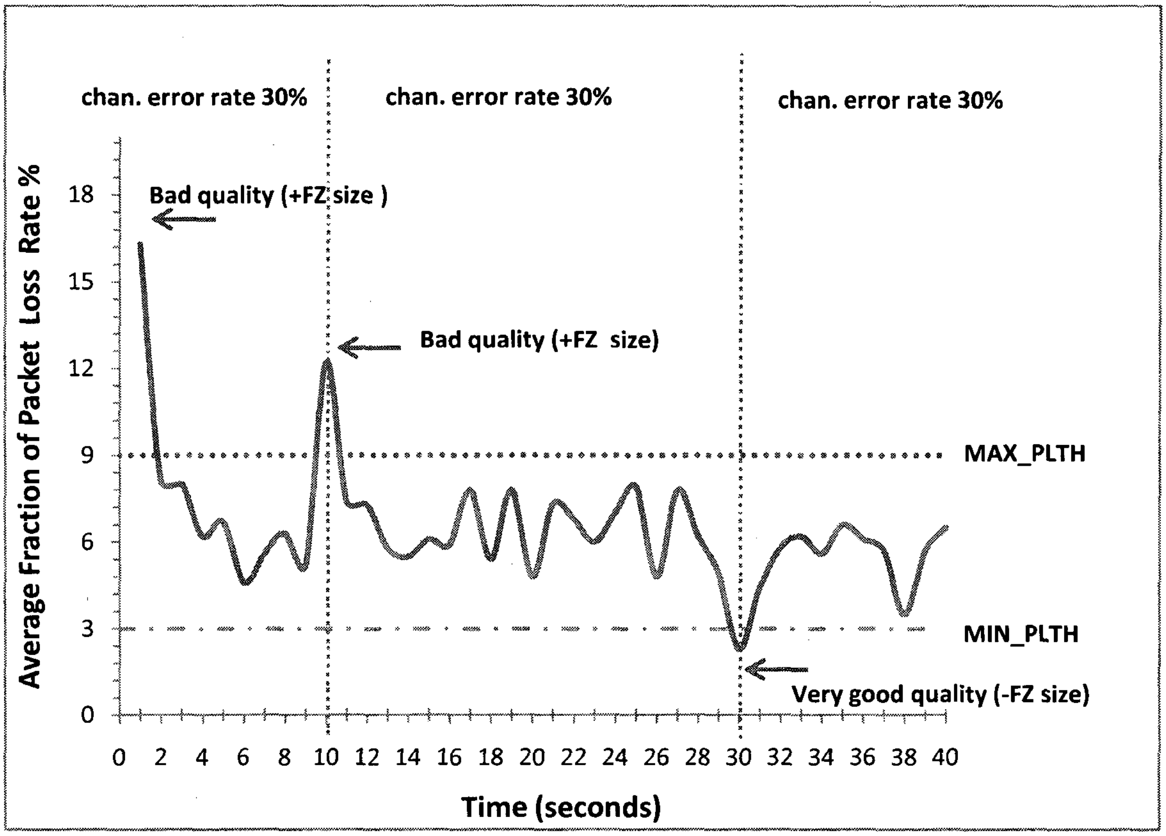


Figure 28 : Average fraction of packet loss for VD+FZIP+FZCP

Figure 29 shows the average fraction of packet loss rate of all the different cases, presented in Figures 27 and 28, at every 1 second time interval up to the end of the session lifetime. It is shown in Figure 29 that, except at the 1st and the 10th second, VD+FZIP+FZCP generally honors the required level of quality during the session lifetime. The occasional drops in quality level, at second 1 and second 10, causes minor glitches while playing the received real-time multimedia traffic. It is shown in Figure 29 that VD+FZIP-I did not provide good quality level for the whole 60 second session lifetime. VD+FZIP-II provides good quality in the moderate wireless channel error rate conditions of 10%. However, it doesn't provide good quality during the bad network condition (i.e. sec 10 – sec 29 with channel error rate of 30%). VD+FZIP-III provided better quality than all other cases for the whole session lifetime.

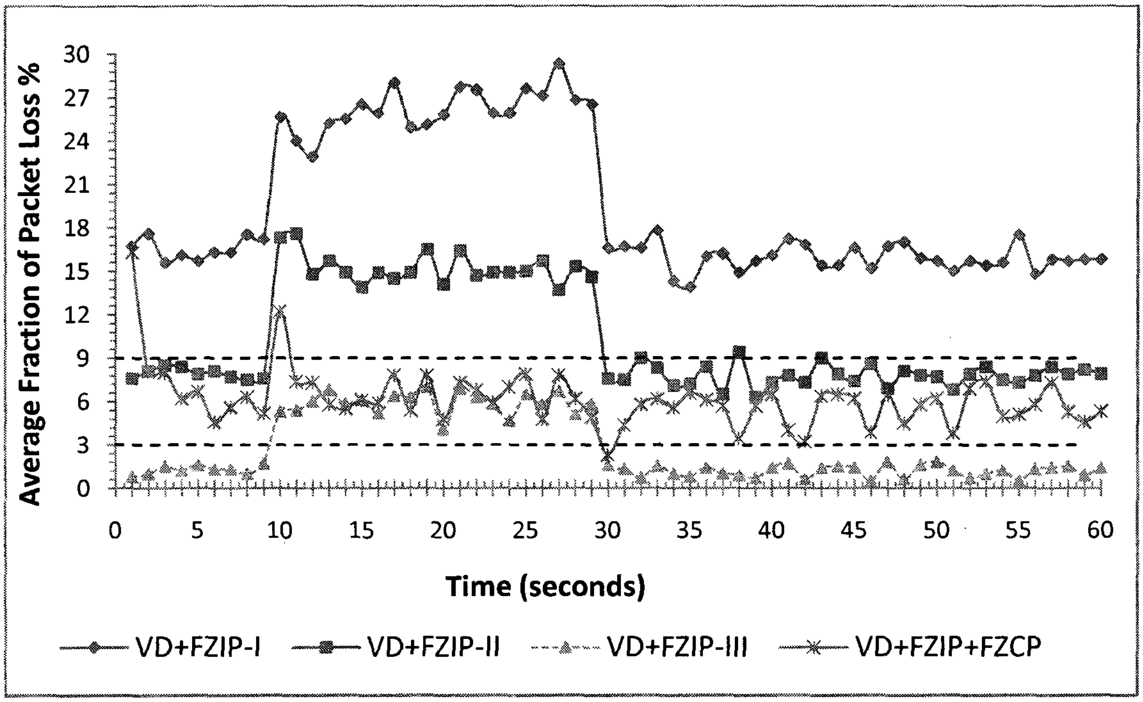


Figure 29 : Average fraction of packet loss for all the simulated cases.

It is worth noting that the goal of FZCP is to achieve a tradeoff of good quality and less power consumption. A protocol achieves good quality when the fraction of packet loss doesn't exceed the MAX_PLTH, and spend less power when the fraction packet loss doesn't exceed the MIN_PLTH. As the average packet loss of VD+FZIP-III doesn't exceed the MAX_PLTH at all time intervals, it provides better quality than VD+FZIP+FZCP. However, the average packet loss of VD+FZIP-III is less than the MIN_PLTH for 40 seconds, thus it is consuming more power. In Figure29, it is shown that the average fraction of packet less by VD+FZIP+FZCP generally lies between the MAX_PLTH and MIN_PLTH, thus provides good quality while not spending unnecessary power.

VD+FZIP+FZCP achieves lower average delivery overhead and lower average power consumption in comparison with VD+FZIP-III (i.e. the only VD+FZIP case which delivered good quality along the session lifetime). Figure 30 compares the average delivery overhead and the average power consumption for VD+FZIP-III and

VD+FZIP+FZCP cases. The VD+FZIP-I and VD+FZIP-II results are omitted as both of these cases failed to provide good quality for the whole session lifetime (see Figure 29). The Average delivery overhead measures the average number of wireless transmission operations required to deliver the message from the source node to the sink. It includes all MAC layer transmissions for VD multimedia data messages, FZIP initialization messages, and FZCP resize messages. The average delivery overhead gives an indication of the average number of nodes participating in the FZ[14].

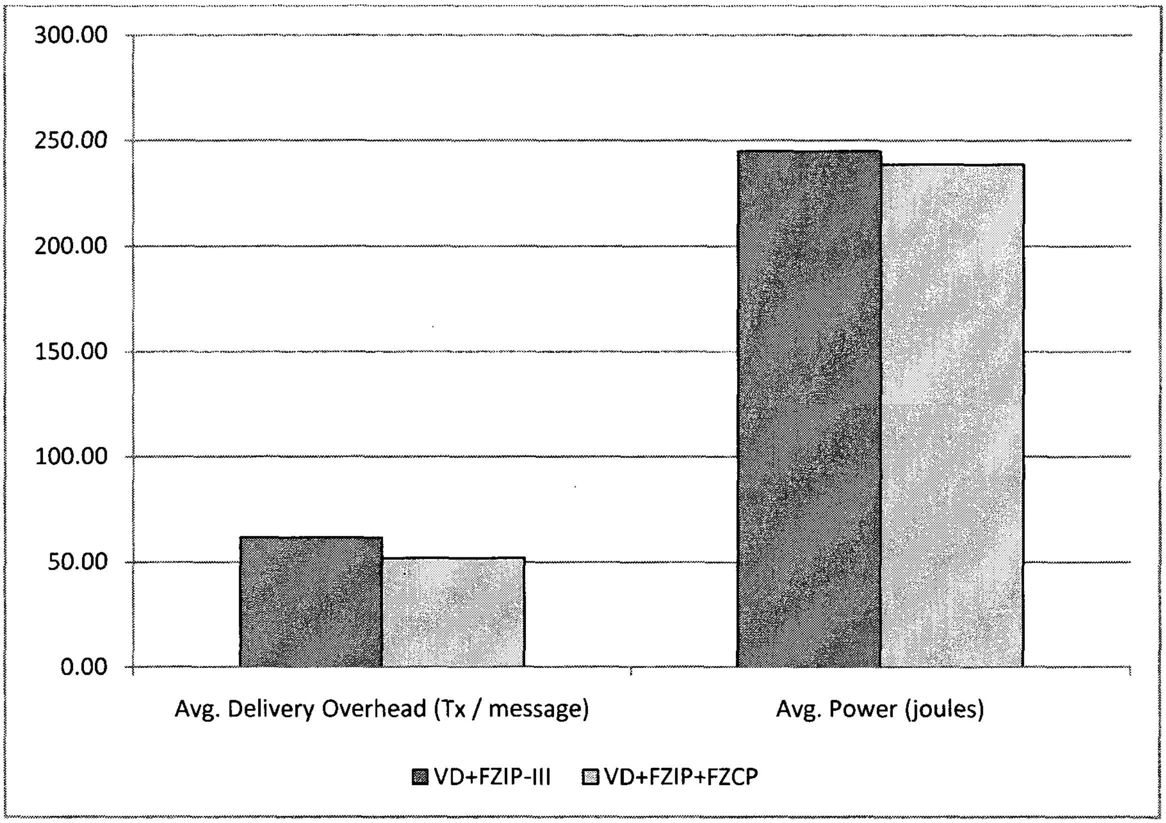


Figure 30 : Average delivery overhead and Average Power consumption

It is shown in Figure 30 and table 9, that the average overhead of FZCP is 51.7, which is about 16% less than 61.51 average overhead of FZIP-III. In addition, VD+FZIP+FZCP consumed 6.171 joules less than VD+FZIP-III. In the VD+FZIP+FZCP case, the flooding zone size was decreased, according to the network condition at that time, and thus saved some power and overhead during the moderate channel condition

[14] Recall that each node in the flooding zone transmits a packet one time only.

periods running with a smaller FZ size and less number of members. Moreover, it is worth noting that although FZCP flooding zone resize messages consumes power and resources, the overhead of these resizing processes is lower than the gained benefit in power consumption.

	Avg. Delivery Overhead	Avg. Power Consumption
VD+FZIP-I	26.81	211.820
VD+FZIP-II	41.38	232.596
VD+FZIP-III	61.51	244.938
VD+FZIP+FZCP	51.70	238.767

Table 9: Average delivery overhead for all cases

In Table 9, it is clear that VD+FZIP+FZCP achieved fair average deliver overhead and average power consumption in comparison to VD+FZIP-II and VD+FZIP-III. The extra power consumption of VD+FZIP+FZCP in comparison to VD+FZIP-II was necessary for obtaining good quality during the 20 seconds bad network conditions. While the less power consumption by VD+FZIP-FZCP in comparison with VD+FZIP-III was achieved by reducing the FZ size during the moderate network conditions (i.e. sec 0-sec 9 and sec 30- sec 60). This reduction in average power consumption and average delivery overhead would be even more in case FZIP was configured to use a larger FZ size (e.g. VD+FZIP with FZ size of 4 or more).

In conclusion, for the FZCP simulated scenarios in section 5.3.2 and 5.3.3, it is clear that FZCP adds an important advantage for flooding multimedia packet over the flooding zone initialized by the set up FZIP protocol. The FZIP protocol uses a fixed FZ size for setting up the flooding zone. Depending on the chosen flooding zone size, different performance metrics can be achieved. As WSNs are dynamic, previous knowledge of current network conditions is hard to expect. If FZIP is configured to use a large FZ size to deliver good quality, then this will lead to unnecessary power consumption at times of good wireless network conditions. However, if FZIP is configured to use a small FZ size to save power, then this will lead to poor and unacceptable quality at time of bad wireless network conditions. FZCP requires no previous knowledge of the current network conditions to provide good performance. The ability of FZCP to monitor the incoming multimedia quality and to change the FZ size accordingly helps to avoid unnecessary power consumption while providing good quality.

5.5 Summary

This chapter explains the conducted simulation experiments to validate our proposed protocols. An overview of the used Network Simulator NS-2 along with an explanation of the simulated environment are presented. Our simulation results show that FZIP and FZCP can efficiently help in reducing power consumption and providing good quality for transporting real-time multimedia data. We have shown that, the Video Diffusion (VD) flooding protocol achieves lower packet loss rates (i.e. better quality) and lower power consumption when used with FZIP. Instead of flooding over all network nodes, VD can use FZIP to construct a flooding zone with different sizes to reduce power and packet loss. In addition, by letting FZCP control the FZ size during the multimedia session lifetime, the delivery performance can generally be enhanced further by switching to the most suitable flooding zone. Thus, FZIP and FZCP enable a simple flooding protocol (e.g. VD) to provide a more reliable and power efficient delivery of real-time multimedia even under error prone and varying network conditions of WSNs.

6. CONCLUSIONS AND RECOMMENDATIONS

In Wireless Sensor Networks (WSNs), reliable delivery of real-time multimedia data over wireless sensors with limited power is a challenging issue. Reliability is important as packet loss is more often in WSNs and occurs due to several reasons at the different levels of the protocol stack (i.e. physical, data link, network, and application layers). Reliable real-time multimedia delivery can be achieved using flooding. In flooding, several copies of the same packet are transferred by all nodes over all paths to the destination node redundantly. In this way, if a packet is lost for any reason (e.g. congestion, fading, interference, route failure, etc...), another copy or copies will have the chance to reach the destination "in time" over different path(s). However, flooding real-time multimedia data in large scale WSNs consumes huge amount of power and can lead to severe broadcast collision and channel contention which degrades the overall reliability and scalability.

In this thesis, we propose the novel Flooding Zone (FZ) concept to enable reliable and power efficient real-time multimedia delivery service for WSNs. The reliable delivery is achieved by redundant data transmission over the different paths in the flooding zone. That is to say, instead of flooding over all nodes in the network, the broadcast storm is constrained in a small zone connecting the source and the sink nodes. Thus, power consumption is way less than normal multimedia flooding. Furthermore, depending on the flooding zone size, different performance metrics can be achieved in terms of delivery rate, latency, and power consumption. The flooding zone concept mitigates the huge power consumption disadvantage of flooding real-time multimedia data while providing reliable and in time delivery.

Accordingly, we propose the Flooding Zone Initialization Protocol (FZIP). FZIP initializes a suitable flooding zone, between a source node and the sink, before flooding the real-time multimedia data traffic. FZIP is simple, light, hardware independent, easy to customize, and MAC and Network layer independent. These features conform to sensors' limitations and facilitate straightforward integration into diverse WSNs architectures and applications.

In addition, we propose the Flooding Zone Control Protocol (FZCP). FZCP is a complimentary protocol which helps to maintain delivering good quality the real-time multimedia session lifetime. As wireless sensors are implanted in the real world and get affected by the dynamically changing physical environment in which they reside, the wireless sensor network conditions keep changing over time. In such situations, the FZ size needs to be changed in order to maintain good quality and less power consumption. FZCP monitors the multimedia delivery, at the sink node, and dynamically changes the FZ size to adapt to current network conditions. FZCP increases the FZ size in situations of deteriorated multimedia session quality, while it reduces the FZ size to reduce unnecessary power consumption when possible.

We conducted several simulations, using the NS-2 simulator, to evaluate our proposed protocols. We evaluated the performance of transferring a real-time multimedia stream under different WSNs sizes and varying network conditions. We compared our proposed FZIP protocol with normal flooding over all paths in the network and with single path communication. Our simulation results show that by using FZIP, multimedia flooding delivers good quality and much lower power consumption in comparison to multimedia flooding. In addition, while single path communication fails to deliver good quality as the distance (in terms of hops) between the source and the sink increases, our proposed protocol delivers good quality successfully regardless of the nodes' distance. Furthermore, the results show that FZIP achieves good performance even at times of severe channel error rates. For FZCP, we evaluated its ability to control and change the flooding zone size in order to optimize the performance of multimedia flooding in stable as well as dynamically changing network conditions. Our results show that, FZCP helps compute the best flooding zone size in stable network conditions and helps maintain good session quality under dynamically changing network conditions, while reducing the power consumption.

Some enhancements can be added to the flooding zone concept for further improvement. While we have focused on static WSNs with non-mobile sensor nodes, the FZIP protocol can be extended to support MANET or WSNs with mobile wireless sensor and sink nodes. By monitoring the delivery performance at the sink, the flooding zone can be re-initialized when bad quality, due to node mobility, is detected. However, the overhead of the re-initialization process and the ability of the monitoring operation to decide when to re-initialize the FZ need to be further investigated.

The goal of FZIP and FZCP is to develop independent protocols which can help multimedia flooding protocol to achieve better reliability and power efficiency. FZIP and FZCP are not responsible for delivering the multimedia data. FZIP sets up the FZ, while FZCP controls its size during the session. The actual packet relaying is handled by the used multimedia flooding protocol, and not by FZIP or FZCP. Although using a very simple multimedia flooding protocol delivers good results, advanced multimedia flooding techniques and policies are expected to enhance the performance significantly. Apart from the simple and straight forward multimedia flooding protocol, where each node broadcasts a received message exactly one time, advanced flooding policies allow the intermediate node to act more proactively. For example, avoiding forwarding a late packet whose expected play-out time has expired would reduce unnecessary overhead. Another example is energy awareness policy. A flooding zone node with limited remaining power can disjoin the flooding zone to extend the overall lifetime of the network.

REFERENCES

[AK04] J. Al-Karaki and A. Kamal. “Routing Techniques in Wireless Sensor Networks: A Survey”. IEEE Wireless Communications, 11(6):6–28, 2004.

[AMC07] I. Akyildiz, T. Melodia, and K. Chowdhury. "A survey on wireless multimedia sensor networks". The International Journal of Computer and Telecommunications Networking, 51(4): 921–960, 2007.

[ASSC02] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. “Wireless sensor networks: a survey”. Computer Networks, 38(4): 393–422, 2002.

[AX05] I. Akyildiz , X. Wang , and W. Wang. “Wireless mesh networks: a survey”. Computer Networks and ISDN Systems, 47(4):445–487, 2005.

[CK85] Chlamtac, I., and Kutten, S., “On broadcasting in radio networks - problem analysis and protocol design”. IEEE Transactions on Communications COMM- 33, pp: 1240–1246, 1985.

[CMY06] C. Chen, J. Ma, and K. Yu. Designing Energy-Efficient Wireless Sensor Networks with Mobile Sinks. In the 4th ACM Conference on Embedded Networked Sensor Systems (SenSys 2006), 2006.

[CW85] I. Chlamtac, and O. Weinstein. “The wave expansion approach to broadcasting in multihop radio networks”. In Proceeding ofINFOCOM, pp: 874–881, 1987.

[DBN03] B. Deb, S. Bhatnagar, and B. Nath. “ReInForM: Reliable Information Forwarding using Multiple Paths in Sensor Networks”. In Proceeding of28th Annual IEEE International Conference on Local Computer Networks, pp: 406–415, 2003.

[EE08] T. Elamsy, and R. El-Marakby. “Flooding Zone Initialization Protocol (FZIP): Enabling efficient multimedia diffusion for multi-hop wireless networks”. IEEE Symposium on Computers and Communications (ISCC08). pp: 1056–1061, 2008.

[FJL00] M. Frodigh, P. Johansson, and P. Larsson. “Wireless ad hoc networking: the art of networking without a network”. Ericsson review, pp: 248–263, 2000.

[FM05] R. Farivar, M. Fazeli, and S. Miremadi. “Directed Flooding: A Fault-Tolerant Routing Protocol for Wireless Sensor Networks”. In Proceeding of2005 Systems Communications (ICW'05), pp: 395–399, 2005.

[HHS08] Y. Li, M. Thai, and W. Wu. “Wireless Sensor Networks and Applications”. 2008.

[HKB99] W. Heinzelman, J. Kulik, and H. Balakrishnan. “Adaptive protocols for information dissemination in wireless sensor networks”. In Proceeding of the 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom99), pp: 174–185, 1999.

[HL88] S. Hedetniemi, and A. Liestman. ”A survey of gossiping and broadcasting in communication networks”. Networks, 18(4):319–349, 1988.

[IGE00] C. Intanagonwiwat, R. Govindan, and D. Estrin. “Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks”. In Proceeding of6th Annual international conference on Mobile computing and networking, pp: 56– 67, 2000.

[IPK05] M. Islam, R. Pose, and C. Kopp. “A Link Layer Security Protocol for Suburban Ad-Hoc Networks”. Australian Telecommunication Networks and Applications Conf. (ATNAC05), pp: 174–177, 2005.

[KR03] J. Kurose, and K. Ross. "Computer Networking: A top down approach featuring the internet". Third Edition, 2003.

[LAB93] C. Lee, M. Ammar, and J. Burns. “Improved Randomized Broadcast Protocols in Multi-hop Radio Networks”. Technical report. 1993.

[Lon01] A. Longman. ”Ad hoc networking”. Book. 2001.

[MBNP03] S. Mao, D. Bushmitch, S. Narayanan, and S. Panwar, “MRTP: A Multi-Flow Realtime Transport Protocol for Ad Hoc Networks”. In Proceeding ofIEEE VTC 2003, pp: 2629–2634, 2003.

[MRX08] S. Mishra, M. Reisslein, and G. Xue. “A Survey of Multimedia Streaming in Wireless Sensor Networks”. IEEE Communications Surveys and Tutorials, 10(4):18–39, 2008.

[MZP01] M. Pearlman, Z. Haas, P. Samar, "The zone routing protocol (zrp) for ad hoc networks", in Internet Draft - Mobile Ad Hoc Networking (MANET) Working Group of the Internet Engineering Task Force (IETF),2001.

[NS2] L. Breslau, D. Estrin, K. Fall, S. Floyd, J. Heidemann, A. Helmy, P. Huang, S. McCanne, K. Varadhan, Y. Xu, H. Yu, "Advances in network simulation", IEEE Computer, 53(5):59-67, 2000.

[NTCS99] S. Ni, Y. Tseng, Y. Chen, and J. Sheu. "The Broadcast Storm Problem in a Mobile Ad Hoc Network". In Proceeding of5th annual ACM/IEEE international conference on Mobile computing and networking, pp: 151–162

[PK00] G. Pottie, and W. Kaiser. "Wireless integrated network sensors". Communications of the ACM, 43(5): 51–58, 2000.

[RFC3561] RFC 3561 - Ad hoc On-Demand Distance Vector (AODV) Routing.

[RFC768] RFC 768 - User Datagram Protocol.

[ROG04] A. Rahman, W. Olesinski, and P. Gburzynski. "Controlled flooding in wireless ad-hoc networks". International Workshop on Wireless Ad-Hoc Networks, pp: 73–78, 2004.

[RR02] R. Ramanathan, and J. Redi. "A Brief Overview of Ad Hoc Networks: Challenges and Directions". IEEE Communications Magazine, 40(5):20–22, 2002.

[RSZ04] C. Raghavendra, K. Sivalingam, and T. Znati. "Wireless Sensor Networks". Springer, Book, chapter 2, pp: 21–50, 2004.

[SACL03] J. Stankovic, T. Abdelzaher, L. Chenyang, and S. Lui. "Real-time communication and coordination in embedded sensor networks". In Proceeding ofIEEE 91(7):1002–1022, 2003.

[SCFJ03] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications". RFC 3550, Network Working Group, 2003.

[SH03] F. Stann, and J. Heidemann. "RMST: reliable data transport in sensor networks". In Proceeding ofFirst IEEE International Workshop on Sensor Network Protocols and

Applications, pp: 102–112, 2003.

[SHS01] A. Savvides, C. Han, and M. Srivastava. "Dynamic fine-grained localization in adhoc networks of sensors", In Proceeding of the ACM MobiCom'01, pp: 166–179, 2001.

[SKK08] S. Sharma, D. Kumar, and R. Kumar. "QoS-Based Routing Protocol in WSN". Advances in Wireless and Mobile Communications. 1(3): 51–57, 2008.

[Sye08] M. Syed. "Multimedia Technologies: Concepts, Methodologies, Tools, and Applications". 2008.

[WCK02] C. Wan, A. Campbell, and L. Krishnamurthy. "PSFQ: a reliable transport protocol for wireless sensor networks". In Proceeding of the 1st ACM international workshop on Wireless sensor networks and applications (WSNA 02), pp: 1–11, 2002.

[WL06] Y. WANG, and H. LIN. "Multipath QoS routing with interference provision in Ad Hoc wireless network. Journal of information science and engineering, 22(6):1325–1338, 2006.

[ZF06] Y. Zhang, and M. Fromherz. "Constrained flooding: a robust and efficient routing framework for wireless sensor networks". In Proceeding of the 20th International Conference on Advanced Information Networking and Applications, pp: 1–6, 2006.

[ZFK04] Y. Zhang, M. Fromherz, and L. Kuhn. "Smart routing with learning-based QoS-aware meta-strategies". In Proceeding of the Quality of Service in the Emerging Networking Panorama, Lecture Notes in Computer Science 3266/2004, pp: 298-307, 2004.

[ZG03] J. Zhao, and R. Govindan. "Understanding packet delivery performance in dense wireless sensor networks". In Proceeding of the 1st ACM SenSys Conference, pp: 1–13, 2003.

[ZH06] J. Zhang, and A. Helal. "Video Diffusion: A Routing Failure Resilient, Multi-Path Mechanism to Improve Wireless Video Transport". In Proceeding of the European Symposium on Mobile Media Delivery (EuMob), pp: 1–5, 2006.

APPENDIX A - ABBREVIATIONS

Abbreviation	Full Title
ACK	Acknowledgement
AODV	Ad hoc On-Demand Distance Vector
CF	Constrained Flooding
Ack-msg	Acknowledgment Message
CNAME	Canonical Name
CTS	Clear To Send
DiffServ	Differentiated Services
DSDV	Destination Sequenced Distance Vector
DSR	Dynamic Source Routing
FEC	Forward Error Correction
FZ	Flooding Zone
FZ-Resize-msg	Flooding Zone Resize Message
FZCP	Flooding Zone Control Protocol
FZIP	Flooding Zone Initialization Protocol
Init-msg	Initialization Message
MAC	Media Access Control
MAX_PLTH	Maximum Packet Loss Threshold
MIN_PLTH	Minimum Packet Loss Threshold
NS	Network Simulator
QoS	Quality of Service
RFC	Request For Comments
RMST	Reliable Multi- Segment Transport
RR	Receiver Report

RTCP	RTP Control Protocol
RTP	Real-time Transport Protocol
RTS	Ready To Send
SR	Sender Report
TORA	Temporally Ordered Routing Algorithm
UDP	User Datagram Protocol
VD	Video Diffusion protocol
WLAN	Wireless Local Area Network
WSN	Wireless Sensor Network

VITA AUCTORIS

NAME:	TARIK ABDUL-RAHMAN EL-AMSY
D.O.B:	1974
PLACE OF BITH:	Kuwait
EDUCATION	M.Sc. Computer Science, University of Windsor, Windsor, Ontario, Canada. 2009
	B.Sc. Computer Science, Ajman University of Science and Technology, Ajman, United Arab Emirates. 1997