DOCTOR OF PHILOSOPHY

AN IMAGE TRANSMISSION PROTOCOL FOR WIRELESS SENSOR NETWORKS

The Graduate School
of the University of Ulsan
Department of Electrical, Electronic,
and Computer Engineering

TA VAN KHOE

AN IMAGE TRANSMISSION PROTOCOL FOR WIRELESS SENSOR NETWORKS

Supervisor: Professor OH HOON

A Dissertation

Submitted to
the Graduate School of the University of Ulsan
In partial Fulfillment of the Requirements
for the Degree of

Doctor of Philosophy

by

TA VAN KHOE

Department of Electrical, Electronic,
and Computer Engineering
University of Ulsan, Korea
February 2022

An Image Transmission Protocol for Wireless Sensor Networks

This certifies that
the dissertation of TA VAN KHOE is approved.

Committee Chair Prof. Yoon, Seokhoon

Committee Member Prof. Oh Hoon (Advisor)

Committee Member Prof. Koo, Insoo

Committee Member Prof. Jo, Dongsik

Committee Member Dr. Yoo, Daesung

Department of Electrical, Electronic
and Computer Engineering,
University of Ulsan, Korea
February 2022

DEDICATED

To my family

Acknowledgments

First and foremost, I would like to express my sincere gratitude to my supervisor, Professor Dr. Oh Hoon, Department of Electrical, Electronic, and Computer Engineering, University of Ulsan, for his invaluable advice, motivating guidance, and support. Throughout the period of my Ph.D., he has provided me the kind support both in technical and non–technical matters. Without his help the vision of both this research work and the arc of the research direction would be hazy or to be apt short–sighted. I will always aspire to his knowledge, experience, and enthusiasm. Besides my advisor, I also would like to thank all the respected professors, lecturers, and the staves at the University of Ulsan for their kind support during my PhD program.

Secondly, I would like to thank the University of Ulsan for the AF1 scholarship and support during my PhD course. I would also like to thankfully acknowledge the gracious financial support of the BK21 plus program. I would be amiss if I did not mention my lab mates, who were always ready to help with any questions that I had.

Thirdly, my warm and heartfelt thanks go to my parents and friends for their tremendous support and hope they have given to me. Without that hope, this thesis would not have been possible. Thank you all for the strength you gave me.

Finally, I owe my deepest gratitude to my lovely wife, Nguyen Thi Thu, and my children. I am forever thankful for their unconditional love, continuous encouragement, and lovely support.

Ta Van Khoe

University of Ulsan, Ulsan, South Korea

December 2021

Contents

Acknowledgments i

Contents ii

List of Figures v

List of Tables vii

ABSTRACT viii

Chapter 1 INTRODUCTION 1

1.1 Wireless Multimedia Sensor Network 1

1.2 Image Transmission 2

1.2.1 Issues and Problems 2

1.2.2 Previous Approaches 3

1.3 Our Approach 7

1.4 Evaluation Methods 9

1.5 Thesis Organization and Contributions 10

1.5.1 Thesis Organization 10

1.5.2 Contributions 10

Chapter 2 PREVIOUS WORK 12

2.1 Image Transmission in WMSNs 12

2.1.1 Network Architecture 13

2.1.2 The Operation of WISNs 14

2.2 Work Related to Image Transmission 17

2.2.1 COM-MAC Protocol 17

2.2.2 Diff-MAC Protocol 18

2.2.3 RAIT Protocol 19

2.2.4 ERTP Protocol 20
2.2.5 EEIT Protocol 20
2.2.6 EMP Protocol 21
2.3 Applications 22
2.4 Research Topics 22
Chapter 3 PIPEPINED COOPERATIVE TRANSMISSION PROTOCOL 24
3.1 Network Model 24
3.2 Motivation 25
3.3 Notations and Messages 27
3.4 Protocol Design 27
3.4.1 Protocol Structure 28
3.4.2 Cooperative Path 29
3.4.3 Channel and Slot Scheduling 31
3.4.4 Image Transmission 34
3.4.5 The Lower Bound of Slot Length 36
3.5 Implementation 38
3.5.1 Testbed 38
3.5.2 Image Preparation 40
3.5.3 Image Transmission 41
3.5.4 Image Reconstruction 41
3.6 Performance Evaluation 42
3.6.1 Comparison of Features from Related Protocols 42
3.6.2 Analysis 45
3.6.3 Simulation 47
3.6.4 Experiment 52
Chapter 4 LoRa-WSN HYBRID NETWORK PROTOCOL 57

4.1 Network Model. 57
4.2 Motivation..... 58
4.3 Notations and Messages 60
4.4 Protocol Design..... 61
4.4.1 Protocol Structure 62
4.4.2 Command and Scalar Data Transmission 63
4.4.3 Image Transmission..... 66
4.5 Performance Evaluation 68
4.5.1 Comparison of Features from Recent Image Protocols 68
4.5.2 Analysis of End-to-End Delay 71
4.5.3 Discussion of Simulation Results 73
Chapter 5 CONCLUSIONS AND FUTURE STUDY 78
5.1 Remarks about Our Work..... 78
5.2 Future Research Directions..... 79
PUBLICATIONS 81
REFERENCES..... 82

List of Figures

Figure 1. Architectures of a wireless image sensor network 13

Figure 2. COM-MAC protocol structure 18

Figure 3. A model of a wireless image sensor network 24

Figure 4. Protocol structure 28

Figure 5. The operating principles of the cooperative path to improve reliability 31

Figure 6. Pipelined transmission with channel and slot scheduling, assuming that level L is 4k+4 34

Figure 7. Testbed for image transmission over WMSN 39

Figure 8. Multimedia node 39

Figure 9. A packet format 41

Figure 10. A packet that is reformatted by a sink 41

Figure 11. Comparison of the protocols for the lower bound of the end-to-end transmission delay of one image 45

Figure 12. Scenario S1 with a square dimension of 30 × 30 (m2) and random distribution of 1 sink and 30 nodes 49

Figure 13. Scenario S2 with a rectangular dimension of 15 × 100 (m2) and a two-way chained deployment of 1 sink and 21 nodes 49

Figure 14. Comparison of three protocols with varying k (scenario S1 and node 17 is the selected multimedia node) 50

Figure 15. Comparison of different protocols according to the hop distance of the multimedia node (Scenario = S2, k = 6 or k= 12) 51

Figure 16. The testbed of WMSN that consists one sink, ten nodes, and one source multimedia node to cover the part of the corridor and four rooms. 52

Figure 17. The sizes of the compressed images according to image resolutions with the

use of the JPEG 2000 standard 53
Figure 18. The clearance of reconstructed images according to the loss rate 54
Figure 19. Packet and Image delivery ratio according to the hop distance of the multimedia node 55
Figure 20. End-to-end delay of image transmission. 56
Figure 21. LoRa-WSN hybrid network model 57
Figure 22. Protocol Structure 62
Figure 23. The operation of hybrid LoRa-WSN protocol 63
Figure 24. An example of logical slot indexing with a frame of 16 slots 64
Figure 25. Command and scalar data transmission 65
Figure 26. The transmission of image packets from a SrcMN at level L to GW 66
Figure 27. The lower bound of the end-to-end transmission delay of one image 73
Figure 28. A scenario of a rectangular dimension of 15 × 100 (m^2) with 1 gateway and 19 nodes in QualNet simulator 73
Figure 29. The PDRs of hybrid protocol with different number of retransmissions (node 14 is the selected multimedia node) 74
Figure 30. The PDR and EC of three protocols with varying Ricean k factor, node 14 at level 8 is the selected multimedia node 75
Figure 31. The PDR and EC of three protocols with varying level of source multimedia node and the Ricean k = 8 76

List of Tables

Table 1. Acronyms and terminologies 27

Table 2. Transceiver channels according to the level of a node 32

Table 3. The comparison of key image transmission protocols.................................. 44

Table 4. Simulation parameters and values. 48

Table 5. Terms and terminologies... 61

Table 6. The features of image transmission protocols.. 70

ABSTRACT

The integration of low-power wireless networking technologies with low-cost hardware such as complementary metal-oxide semiconductor cameras and small microphones has fostered the development of Wireless Multimedia Sensor Networks (WMSNs) that are able to ubiquitously retrieve multimedia data such as still images or voices and scalar data from the environment. WMSNs have many resource constraints in terms of energy consumption, bandwidth, and processing capability, and wireless links are unstable. Furthermore, the transmission of large amounts of data over WMSNs has stringent requirements such as high reliability in packet transmission to recover high quality multimedia data, low end-to-end delay in data transmission to make timely decisions, and high efficiency in energy management to extend the lifetime of network. However, it is not easy to achieve these requirements due to low bandwidth and link instability because of signal fading, high interference, and a lot of obstacles. Therefore, the design of a cross-layer protocol to meet the requirements for both data transmission within the constraints becomes a concern and should be carefully considered.

We focus on the design and implementation of a pipelined cooperative transmission protocol for fast and reliable image delivery in wireless sensor networks (WSNs). In this approach, if a sink needs an image from a specific multimedia node, it establishes a cooperative path such that a path node, a node on the tree path from the multimedia node to the sink, selects a cooperating node from its neighboring nodes at the same tree level. Then, a distinct broadcast time slot is allocated to each tree level and the cooperating node assists its counterpart using the information obtained by means of overhearing message or data in order to forward a packet reliably to the nodes one level lower. This transmission mechanism improves the reliability of hop-by-hop transmission greatly without relying on retransmissions while making the packet move

fast. Furthermore, packets are transmitted in a pipelined manner using two channels thereby reducing the delay of image transmission greatly.

After that, we propose a simple LoRa-WSN hybrid network protocol to transmit both scalar and multimedia data over WSNs efficiently. This approach utilizes the long-range (LoRa) transmission of a LoRa network to deliver scalar data and control messages directly and utilizes the high data rate and high reliability of a WSN for the transmission of an urgent image or voice. The selected multimedia node transmits an image packet in every transmission interval without using control messages such as RTS and CTS. Upon receiving an image packet, a node forwards it immediately and continues in receive mode to receive other image packets, leading to a decrease in image transmission time. Furthermore, the selected multimedia node uses the LoRa network to get information about lost packets for path-wide retransmission in order to meet the requirement for high reliability in data transmission. The approach also can avoid the overhead problem caused by flooding control messages, and it can simplify the operation of sensor nodes by removing the requirement for time synchronization.

Chapter 1

INTRODUCTION

1.1 Wireless Multimedia Sensor Network

Wireless multimedia sensor networks (WMSNs) are composed of sensor nodes with processing, sensing, and radio communication capabilities at low power, where sensor data can be scalar or multimedia data. Sensor nodes that are battery-powered are deployed in a target area, and they work together to deliver sensory data to a sink or a server in multiple hops [1]. Depending on the ability to deliver multimedia data such as an image or voice, WMSNs can provide more precise and detailed information to a server. They also span a wide range of applications in recent years. However, WMSNs are always affected by time-varying link conditions caused by node mobility, internal and external interferences (internal interference indicates an interference caused by different data transmissions in the same network, while external interference is caused by data transmissions in other wireless networks using the same frequency band) [2], and various obstructions such as steel materials, concrete blocks, etc. Furthermore, WMSN applications often require high reliability in packet transmission, low end-to-end delay in data transmission, and high efficiency in energy management.

WMSN applications gain in popularity, and they are used in a wide range of fields. Considering one of the popular multimedia applications is safety surveillance for detection of disasters in which a server collects context data, such as the presence of flame, gas, oxygen, smoke, and/or extreme temperatures, from scalar sensors deployed in the target area, and the server determines whether an abnormal situation has occurred or not based on the analysis of those data. The judgments based on those scalar data may not be 100% reliable due to inherently erroneous sensor data. If the analysis is reliable to some extent, the server may request a still image from a multimedia node installed in the spot in order to make sure that a dangerous situation has really happened before taking proper measures. However, an image will not fit into the IEEE 802.15.4 medium access control (MAC) protocol data unit (MPDU) that can accommodate a header and trailer of 18 bytes and a payload of up to 110 bytes [3]. Thus, it has to be segmented into a series of packets and transmitted serially. For example, one image of 512×512 pixels with 24 bits per pixel corresponds to 786,432 bytes. Even if this image is compressed by using the JPEG2000 compression scheme with a compression ratio of 1:33.65 [4], it is still approximately 23,400 bytes, which corresponds to about 213 packets. If it takes about 100 milliseconds to deliver one packet, 21.3 seconds will be required to deliver the whole image. This delay is far beyond the time bound generally required for safety applications. Furthermore, those packets have to be delivered reliably so that the sink can recover an image of good quality. Another important issue in delivering an image is energy efficiency. If all nodes have to remain active during the delivery time of a big image, they can waste precious energy.

Many WMSN protocols dealing with the delivery of large amounts of multimedia data have been proposed so far for WMSN applications. Those ones may be classified into two categories: The *video streaming protocol* to transmit streaming data in a steady flow and the *image transmission protocol* to transmit a bulk data in an event-driven manner or as on-demand basis. In this thesis, we concentrate on image transmission protocols that have lower bandwidth requirements than video streaming protocols [5].

1.2 Image Transmission

1.2.1 Issues and Problems

Three stringent requirements for sending an image in WMSNs are high reliability in data transmission, low delay in image transmission, and high efficiency in energy management. The reliability of data transmission directly affects the quality of the image recovered at a server or sink. Time constraints for data acquisition are required to take timely measures and make timely decisions. Energy consumption must be managed efficiently because of the limited capacity of batteries. However, it is not easy to meet the requirements due to the low bandwidth and error-prone properties of WSNs.

Furthermore, source multimedia nodes produce large amount of data, and the nodes will generate a large number of packets because of the IEEE 802.15.4 medium access control (MAC) protocol data unit (MPDU) of 127 bytes [3]. In addition, different types of data such as scalar data, image data, and control messages exist simultaneously in WMSNs. As a result, the delivery of these data over WMSNs necessitates high bandwidth demand and it poses several challenges in terms of increased node power consumption and longer transmission delay [6]. However, there are many resource constraints in WMSNs such as irreplaceable battery-powered, a low data rate, and a limited processing capability.

The issues and requirements for designing WMSNs open many research directions, such as the design of new platforms and testbeds to process and manage efficiently multimedia data [7], [8], the design of novel protocols to satisfy the requirements for the various applications in real environments [9], [10], and the researches on source coding techniques and data processing to decrease the size of multimedia data delivered over WMSNs and enhance the correctness of received data [11], [12]. Therefore, the delivery of multimedia data over WSNs has become a concern and should be carefully considered and evaluated in a real environment.

1.2.2 Previous Approaches

Some protocols have been proposed to improve reliability in image transmission under end-to-end delay constraints. In [13], the authors proposed the reliable synchronous transport protocol (RSTP) to obtain images reliably from multiple multimedia sources while keeping the same level of quality for different images. They employed the receiver-controlled scheme for reliability instead of using the sender-controlled one [14] which is not suitable for WMSNs with low bandwidth and unstable links. RTSP uses a *path-wide retransmission scheme* such that a sink asks the source to retransmit the lost packets. In [15], the authors proposed the reliable asynchronous image transfer (RAIT) protocol that can maintain reliability against link failures and network congestion while multiple multimedia sources send a series of multimedia packets to a sink. They employ two sliding windows, one for the receiving queue to deal with link failures and another for the sending queue to control network congestion.

Some other approaches tried to reduce end-to-end delay in transmitting large multimedia data. The authors in [16] combined a massive transmission scheme (MTS) with the X-MAC protocol [17] to deliver large multimedia data fast while using the duty cycle mechanism to save energy. Meanwhile, a pipelined transmission approach was triggered by Flush [18] in which nodes transmit packets as long as parallel transmissions are not interfered, especially in a long chained wireless network. To maximize throughput with no interference, they estimate the minimum delay between two consecutive packet transmissions using an interference range. While Flush uses a single packet transmission, the authors in [19] used the *packet train*, a series of packet transmissions, to maximize throughput and enforce compliance to dynamic duty cycle limitations. Every node tries to maximize transfer throughput by tracking its duty cycles adaptively and determining the maximum number of packets it can receive and transmit in the train. These two approaches focused on asynchronous bulk transfers; they have evolved by incorporating the TDMA mechanism and using multiple channels for synchronous pipelining [20]. If all nodes on the path from a source to a destination use synchronized time slots, every node except the sink on the path can transmit data in every other slot using a distinct channel, thereby enabling a lot of parallel transmissions. However, if a single path is used, every source has to keep a *pipeline cycle time* that consists of one busy slot and one idle slot because its upstream node has to forward the received data in the idle slot. Moreover, wireless links on WMSNs are not assumed to be stable. Thus, the pipelined approach exposes two problems: the *bubble slot problem* that a source node skips one slot within the cycle time and the *pipeline stall problem* that a sink does not receive a packet during one cycle time if a packet is lost. Some protocols use a costly path-wide retransmission to recover the lost packet. Note that the probability of packet loss becomes higher with longer path.

In [21], they used *hop-by-hop retransmissions* such that a sender retransmits the same packet within the same cycle time if it detects the loss of the transmitted packet to mitigate the latter problem. In [22], the authors established *a dual-transmission path* such that two nodes are located at every hop distance except a source and a sink and then allowed two nodes to transmit the same packet simultaneously to exploit the notion of *the constructive interference* [23]. The protocol advances by using multiple dual-transmission paths to remove the bubble slot problem; however, it suffers from high energy consumption and still uses path-wide retransmissions. In [24], the authors proposed the efficient multi-path pipeline transmission (EMP) protocol. They not only removed the bubble slot problem with the use of multiple paths, but also they improved the pipeline stall problem by adjusting the pipeline cycle time such that slot length is increased to allow hop-by-hop retransmissions. However, it still suffers from image transmission delay due to the use of long pipeline cycle time and path-wide retransmission, and also from energy consumption due to the use of retransmission.

Some other approaches are more concerned with energy efficiency in addition to reliability. In [25], the authors proposed the energy-efficient and high-quality MAC protocol (EQ-MAC) in which a sender and a receiver can optimize idly listening times by using multiple sub-RTSs and sub-CTSs instead of a long preamble used in B-MAC [26]. Note that a node can know the intended destination only at the end of a long preamble. Furthermore, a sender and a receiver can avoid interference by blocking their neighboring nodes using sub-RTS and sub-CTS. In [27], the authors proposed the energy-efficient and reliable transport protocol (ERTP) that exploits trade-off between energy consumption and reliability by adjusting the maximum number of retransmissions for a lost packet according to the estimated link quality. In [28], the authors studied an energy-aware multipath routing technique to prolong the lifetime of the cluster based WMSNs in transmitting a high volume of multimedia data. They used a genetic algorithm to find the optimal routing path based on the minimum distance and the least energy dissipation.

Although the above-mentioned protocols have been proposed on WMSNs, the studies have mostly focused on the design of a new protocol and its evaluation by simulation. Therefore, it is worth studying to implement and experiment an image transmission protocol to see whether or not it can satisfy the stringent requirements of industrial applications in delivering an image on WMSNs.

In [29], the authors implemented a simple slotted transmission protocol that allows a multimedia node to transmit each image packet with a sufficient delay such that only one packet remains on the whole data path. This implies that it excludes multiple concurrent transmissions to avoid collision. Furthermore, it assigns a slot big enough for a node to perform retransmission in case of transmission failure. According to experiments, given that one slot time is limited to 11 milliseconds, the protocol could achieve packet delivery ratio (PDR) of 91.7% when a multimedia node is located two hops away from a server. Consequently, it is not determined that the protocol can satisfy the industrial requirements if the hop distance increases. Moreover, the use of a big slot can increase the end-to-end delay of image, and the retransmissions on the identical link may rather degrade reliability by increasing traffic.

In [30], the authors implemented a transmission control protocol (TCP) that uses congestion control and error control to achieve high reliability in data transmission. For congestion control, a sender adjusts its data sending rate adaptively according to the congestion level of a receiver that is determined based on buffer occupancy and buffer occupancy change rate. For error control, a node uses a hop-by-hop priority-based retransmission technique that allows retransmission only when a free channel is available and also gives higher priority to important I-frame packets than P-frame packets. Experiment results showed that the protocol could achieve a PDR of 98.7% for I-frame packets and a PDR of 95.8% for P-frame packets in a long-range topology of 6 hops. However, every sender on the path has to adjust a data sending rate according to the time-varying traffic of a receiver. This implies that every node has to send traffic information to its upstream node whenever necessary. This will not only degrade channel utilization, but also make it difficult to manage energy saving.

1.3 Our Approach

We address the above-mentioned issues and the disadvantages of recent protocols by designing *a new image transmission protocol* that can transmit image packets reliably without relying on retransmissions and can improve end-to-end delay significantly by employing a pipelined and cooperative transmission method. The protocol that works over a tree originating in a sink has two phases for image transmission: *path construction* and *image transmission*. If the sink wants to receive an image from a specified source multimedia node (*SrcMN*), it starts the path construction phase by broadcasting an *image request message* (*IREQ*) in order to establish a cooperative path from *SrcMN* to the sink. The cooperative path is constructed by including all nodes on the tree path from *SrcMN* to the sink and having each of the nodes select one of its neighbors at the same level to serve as *a cooperating node*. Then, a distinct transmission slot is allocated to each tree level, and two nodes at every tree level on the cooperative path collaborate to forward packets to nodes one level lower within the allocated slot. Packets are transmitted in a pipelined manner using multiple channels, such that nodes at every other level can transmit packets in parallel. The protocol saves energy by managing the duty cycles of the nodes on a cooperative path and by turning off the nodes that are not involved in image transmission. Upon receiving *IREQ* from the sink, a node can estimate the time by which it should receive an *image reply message* (*IREP*) from *SrcMN*. Then, it goes to sleep either if it does not receive IREP, or unless it is selected as a member of the cooperative path.

The superiority of the proposed protocol is proven in various ways. First, we analytically verified that the proposed protocol can reduce end-to-end delay significantly, compared to two recent and well-known multimedia protocols, EMP [24] and EEIT [29]. Second, we also proved by simulation that the protocol can greatly reduce energy consumption compared to the recent protocol, EMP, while achieving a high packet delivery ratio. Lastly, an experimental study was conducted to verify the soundness of the simulation studies on end-to-end delay, packet delivery ratio, and image delivery ratio and to examine the effect of using cooperating nodes. It was done in two environments that have different levels of the multipath fading effect.

Following that, we proposed a simple LoRa-WSN hybrid network protocol that can achieve the high reliability of image transmission and can transmit scalar data in a real-time manner. Nodes are assigned logical slot indices for scalar data transmission using a logical slot indexing algorithm that can satisfy time bounds in scalar data transmission. Then, nodes transmit scalar data in each assigned logical slot to achieve time constraints and avoid collisions. For image transmission, the selected multimedia node transmits an image packet in every transmission interval without using control messages such as RTS and CTS. Upon receiving an image packet, a node forwards it immediately and continues in receive mode to receive other image packets, leading to a decrease in image transmission time. Furthermore, the selected multimedia node uses the LoRa network to get information about lost packets for path-wide retransmission in order to meet the requirement for high reliability in data transmission. The operation of sensor nodes is simplified because it is triggered and controlled by the LoRa network, and it does not require time synchronization.

1.4 Evaluation Methods

Comparison of features from related protocols: We contrasted the features of the proposed protocols and those of other recent multimedia protocols for some performance metrics that are often used in image transmission, such as reliability in packet delivery, end-to-end delay in data transmission, and average energy consumption of network. The relative indicators such as high, medium, and low were used to compare the performance metrics of multimedia protocols.

Analysis: We calculated the lower bound of end-to-end delays (E2EDs) to transmit one image from the *SrcMN* at level *L* to a sink under the image transmission protocols and compared the E2EDs with varying level of *SrcMN*.

Simulation: The commercial simulator, QualNet 5.02, was used to evaluate the proposed protocols and to compare them with other recent protocols with different scenarios. In the simulation, the Ricean model for a multipath fading effect was used with driving parameter *k*, which is defined as the ratio of the receiving power in the direct path to that in other paths.

Experiment: We used the u*Mote* developed in our lab, running ContikiOS 2.6.

Testing scenarios were deployed and performed on campus and along the corridor of the computer engineering building at the University of Ulsan to verify the satisfaction of proposed protocols in environments with varying interference signals.

1.5 Thesis Organization and Contributions

1.5.1 Thesis Organization

Following a brief introduction to wireless multimedia sensor networks, the dissertation is organized as follows.

Chapter 2 presents the architecture and operation of WISNs. It also discusses recent image transmission protocols, WMSN applications, and research topics.

Chapter 3 provides a detailed description of an image transmission protocol that is proposed to address three issues related to the transmission of an image over WMSNs: Reliability in data transmission, image transmission delay, and energy efficiency.

Chapter 4 presents a simple hybrid network protocol that uses LoRa communication module to transmit scalar data and control messages and uses sensor communication module to transmit multimedia data.

Chapter 5 gives concluding remarks about our work and future research directions.

1.5.2 Contributions

First of all, we concentrate on reviewing the issues and challenges that come with designing a WMSN protocol and deploying a WMSN. We also discuss recent research on WMSNs and emphasize the advantages and disadvantages of each approach. This work provides a brief and comprehensive overview of the WMSN field and helps to identify research trends in this field.

Second, we shift our focus to previous work related to image transmission and present methodologies to evaluate a proposed protocol for image transmission in WISNs. After that, we summarize WISN applications and research topics.

Third, we propose and implement an image transmission protocol named a pipelined cooperative transmission (PCT) protocol for fast and reliable image delivery based on a tree topology originating from a sink, and allocation of a distinct time slot to each tree level. In this approach, if a sink needs an image from a specific multimedia node, it establishes a cooperative path such that a path node, a node on the tree path from the multimedia node to the sink, selects a cooperating node from its neighboring nodes at the same tree level. Then, the cooperating node assists its counterpart using the information obtained by means of overhearing message or data in order to forward a packet reliably to the nodes one level lower. This transmission mechanism improves the reliability of hop-by-hop transmission greatly without relying on retransmissions while making the packet move fast. Packets are transmitted in a pipelined manner using two channels such that different nodes at every other level send packets simultaneously.

Finally, we proposed a simple hybrid network protocol to deliver both scalar and image data efficiently, in which long range (*LoRa*) network is used to deliver control messages and scalar data at regular intervals, whereas WSN is used purely to deliver multimedia data from a *SrcMN* to a sink in an event-driven manner. This approach utilizes the long-range transmission of the LoRa network to deliver control messages directly from a sink to nodes. As a result, the hybrid network can decrease the overhead problem caused by flooding control messages in multi-hops. The approach also utilizes the high data rate and high reliability of a WSN for the transmission of an urgent image or voice. Moreover, the operation of sensor nodes is simplified because it is triggered and controlled by the LoRa network, and it does not require time synchronization.

Chapter 2

PREVIOUS WORK

2.1 Image Transmission in WMSNs

In WMSNs, nodes still have resource constraints such as processing, storage, and power limitations. Thus, the delivery of still images is more practical than the delivery of video streams, determining the context of wireless image sensor networks (WISN). In WISNs, multimedia nodes will capture, store, and process still images and scalar sensor nodes will measure scalar data such as pressure, humidity, and temperature. Nodes will collaborate to wirelessly deliver images and scalar data to a desired destination, referred to as a sink or server. Then, a sink will transmit collected data to the control center via a local wired link. Additionally, a sink can also send a command or control message to a particular node, or all nodes to control the operation of nodes or to request an image.

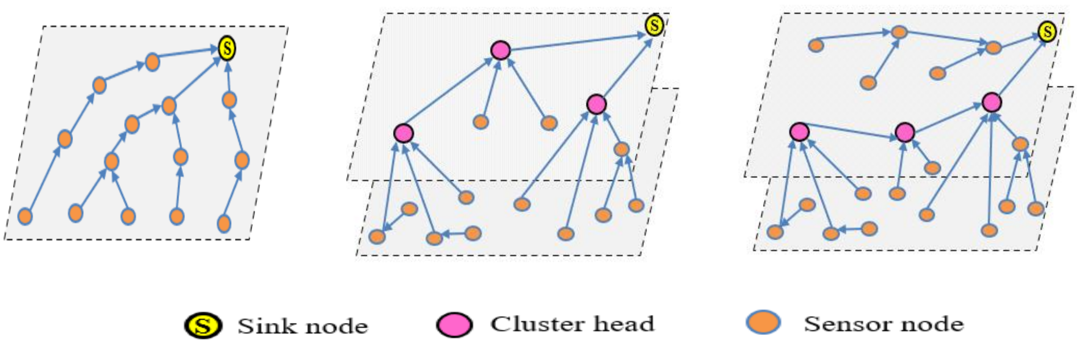
A scalar sensor node (*SN*) is composed of four major components: A sensing module, a processor module, a wireless communication module, and a power supply module. It has to operate autonomously and without requiring maintenance for a long time, even years. It is capable of measuring or collecting environmental information, performing some processing, and communicating with other network nodes [31]. However, the processing capacity of a SN is limited, and a node has an energy-limited battery power supply. Thus, the operation of a SN focuses on measuring environmental information, transmitting the data, and forwarding received data.

A multimedia sensor node (*MN*) is constructed by combining a multimedia module and a SN. The multimedia module may contain a processor that is designed specifically for image or video applications. Typical MNs are the Mesheye node, Panoptes node, Cyclops node, and CMU Camhe node [32]. The operation of multimedia module will aggravate the energy consumption problem of the MS. The most common methods of dealing with the problem are to turn the multimedia module on in an event-driven manner and to compress the multimedia data before transmitting it.

A sink is a device that is similar to sensor nodes but has more powerful features such as a stronger storage and processing capability. It is mainly responsible for packaging and sending commands to the nodes, collecting the data from SNs and MNs, and forwarding received data to the control center that interacts with users.

A control center is responsible for checking collected data that is delivered by the WISN, monitoring the data, and saving it on a computer for further activities. It also provides an interactive interface that allows users to supervise, evaluate, and give final decisions based on collected data.

2.1.1 Network Architecture



(a) Single-tier flat architecture (b) Single-tier heterogeneous architecture (c) Multi-tier heterogeneous architecture

Figure 1. Architectures of a wireless image sensor network

The WSN architecture is a flat architecture in which nodes are distributed homogeneously and are responsible for simple tasks such as physical information measurement. However, with the emergence of MNs besides SNs, it is necessary to categorize the network into different architectures to easily manage the operation of the network and efficiently utilize the available resources in the network. In general, there are three types of WISN architectures [31], as shown in Figure 1.

First, the *single-tier flat architecture* is composed of homogeneous nodes with the same functionalities and capabilities and a sink node, as shown in Figure 1 (a). In the architecture, a node can execute a lot of functions, such as capturing images, sensing data, and forwarding data. The architecture is simple to deploy, manage, and maintain because of the use of homogeneous nodes. It is, however, lack of resource sharing and inflexibility.

Second, the *single-tier heterogeneous architecture* is made up of heterogeneous nodes that are classified into two layers or more based on their capabilities or resources. In the two-layer, for example, the nodes are classified into member nodes and cluster heads, as illustrated in Figure 1 (b). The nodes within each cluster gather data and transmit it to the cluster head. The cluster head can communicate with a server directly or indirectly. This architecture has the advantage of being able to handle a variety of application scenarios.

Third, the *multi-tier heterogeneous architecture* is a hybrid of the single-tier homogeneous and multi-tier heterogeneous architectures, as shown in Figure 1 (c). In this architecture, a node communicates with the sink via the cluster head or via other nodes in multi-hops. The architecture is flexible, allowing it to be used for a variety of tasks with a better coverage, and high reliability. It is, however, a complex architecture with high maintenance costs.

2.1.2 The Operation of WISNs

The delivery of command or control messages from a sink to a specific node or group of nodes to control and manage the operation of nodes, and the delivery of both scalar and image data from nodes to a sink are two main operations in WISNs. The delivery of command messages requires high reliability and strict time constraints. Flooding is the most dependable method for delivering messages quickly, but it consumes a lot of network resources and battery power. The simultaneous transmission of a still image and scalar data is inherently contradictory. On the one hand, sensor nodes face resource constraints such as low data rates, limited battery life, and constrained processing capabilities. A still image, on the other hand, necessitates more complex processing methods and significantly higher bandwidth for delivery. The use of a compression technique at MNs that will effectively decrease the size of captured image and the use of a transmission protocol that can deliver an image over WSNs effectively are the best approaches to dealing with this contradiction. In fact, the delivery of an image and the image compression are not independent because the compression can impact on energy efficiency, E2ED of image delivery, and network lifetime. In this section, we describe the operation of WISNs that includes three functional parts: sensing, processing, and delivery to transmit an image from the source MN to a sink. Among these operations, the processing and delivery are the most power-intensive operations.

a. Processing of Capture Images

To decrease the total amounts of data that needs to be delivered in WISNs, processing of captured images is performed at a multimedia node (on-board processing). The processing can involve a simple or complex image processing algorithm. A simple processing algorithm such as background subtraction can provide MNs with basic information and assist them in determining whether to transmit the captured image or continue processing the image at a higher level. More complex algorithms such as feature extraction and object classification allow MNs to reason about the captured phenomena, such as classifying of the captured object. Therefore, the increased complexity of processing algorithms results in highly intelligent WISNs that can provide relevant information about the phenomena from captured images [33].

However, the power constraints of MNs require the use of simple image compression with a suitable compression ratio and acceptable image quality. There are two types of image compression techniques: Lossless and lossy image compression techniques. Lossless image compression methods use encoding techniques such as Run Length Encoding (RLE), Huffman encoding, and Lempel Ziv Welch (LZW) encoding directly, whereas lossy image compression methods use these techniques after quantization of the image or after some transformations such as a Discrete Cosine Transform (DCT), a Fast Fourier Transform (FFT), and a Wavelet Transform (WT). Many image compression standards such as GIF, JPEG2000, and MPEG-4 have been proposed and implemented, and each compression standard employs an encoding technique, such as JPEG images using Huffman encoding and GIF images using the LZW encoding technique.

Lossless compression differs from lossy compression in that it reduces the image size without sacrificing image quality and allows the original image to be reconstructed. However, the compression ratio of lossy compression techniques can exceed 50:1, while the compression ratio of lossless techniques can only reach up to 4 times the original image. By using lossless compression, a compressed image still has a large size, and the transmission of the image over WISNs requires high bandwidth demand. Therefore, lossless compression is used for applications that require a high quality of reconstructed image [34], and lossy compression is highly encouraged in WISNs.

b. Image Delivery

When a MN intends to transmit an image to a sink, it can either transmit its image directly to a sink (single-hop communication) or transmit the image to a sink through one or more intermediate nodes (multi-hop communication). Compared to single-hop communication, multi-hop communication can extend the coverage of a network and improve connectivity. Moreover, multi-hop communication gives higher energy efficiency and enables higher data rates, resulting in higher throughput and more efficient use of the wireless medium than single-hop communication [35]. Thus, multi-hop communication is the preferred communication method in WISNs.

Multi-hop communication can result in an increase in end-to-end delay, due to queueing and data processing at the intermediate nodes. Moreover, the reliability of data transmission over lossy wireless links will be decreased with the increase in the number of hops on the data path. Furthermore, data delivery is the most expensive in terms of energy consumption. Thus, to deliver an image effectively, WISNs try to reduce the size of the image by compressing captured images before delivering them over the wireless channel and exploit an energy-efficient protocol that can also fulfill the reliability and delay requirements of images.

2.2 Work Related to Image Transmission

2.2.1 COM-MAC Protocol

In [36], authors proposed a Clustered On-Demand Multichannel MAC (COM-MAC) protocol for cluster-based WMSNs. The COM-MAC operates in time intervals with each interval consisting of three consecutive phases: request phase, scheduling phase, and data transmission phase, as shown in Figure 2. During the request phase, each sensor node sends a request message (REQ) to the cluster head. The REQ message contains requirements for quality of % (QoS) such as the delivery deadline, priority information, and the size of multimedia data. During the scheduling phase, the cluster head uses a certain optimal scheduling algorithm and the information in REQ to adaptively schedule channels and time slots for cluster members, and then distributes the resulting schedule to nodes in the cluster to coordinate the data transmission. During the data transmission phase, sensor nodes use the assigned channel and send their data in the assigned slot without further contention.

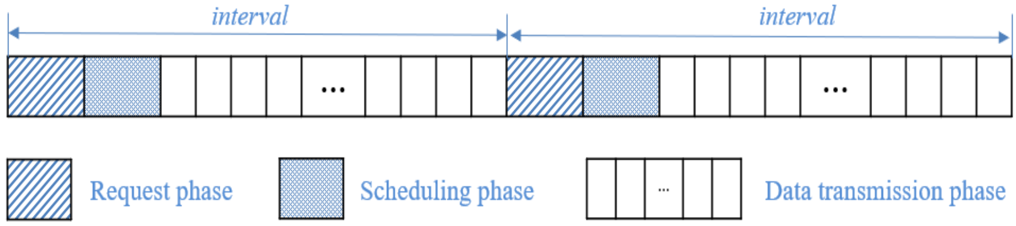


Figure 2. COM-MAC protocol structure

COM-MAC is the first scheduling-based multiple channel MAC protocol specifically designed for WMSN applications. It can achieve higher network throughput by utilizing multiple channels. However, it does not consider channel assignment of the whole network leading to high probability of collisions for data receiving at cluster heard.

2.2.2 Diff-MAC Protocol

In [37], authors designed and implemented a QoS-aware MAC protocol named Diff-MAC, that adjusts the contention window (CW) size and duty cycle (DC) dynamically based on a collection of relevant network statistics such as dominant packet classes and transmission failures. Diff-MAC differentiates the packets into different categories based on QoS requirements, such as best effort, non-real time, or real time. Then, it adapts its duty cycle (DC) relying on the network's dominant traffic class to balance delay and energy consumption. For example, if the flowing traffic is mainly real time data, sensor nodes adapt a higher duty cycle to meet the stringent delay requirements. Whereas if the flowing traffic has a best-effort characteristic, sensors adjust their duty cycles to lower levels to conserve energy. Diff-MAC also adjusts the CW size to achieve a tradeoff between the collision caused by simultaneous transmissions and the time wasted on waiting for the back-off counter to expire.

Moreover, the Diff-MAC maintains multiple priority queues instead of using one queue to save traffic classes in different priorities. It meets the real time requirement by employing a weighted fair queuing method to schedule packets on different queues and by employing a traversed hop count to schedule packets on the same queue. The approach is effective at ensuring fairness among different priority packets and it can provide lower latency and higher throughput. However, it suffers from the overhead problem caused by traffic class differentiation mechanisms and network monitoring statistics, and the cost of energy consumption caused by the use of large buffer sizes.

2.2.3 RAIT Protocol

In WISNs, a node has sensory data and must relay the packets from other nodes to a sink. Therefore, a node should consider the state of the sending queue and the state of the receiving queue at the network layer. In [15], authors proposed the reliable asynchronous image transfer (RAIT) protocol that can maintain reliability against link failures and network congestion while multiple multimedia sources send a series of multimedia packets to a sink. They employ a management scheme inside a sensor node and use a double sliding window method for node-to-node transmission, with one for the receiving queue to deal with link failures and another for the sending queue to control network congestion.

In addition, RAIT also uses a preemption scheme that allows an exclusive possession to guarantee that a node will preempt its parent at one level lower. By using the scheme, if a node, **A**, is transmitting packets corresponding to an image to its parent, **B**, other sibling nodes that want to send data to the common parent, **B**, have to wait for the node **A** to complete the transmission of packets. If the size of the receiving queue in the receiver is smaller than the image size, multiple receivers (nodes) will be used to receive an image at one. The use of preemption scheme can decrease the size of buffer in the sink node that collects and arranges image packets received from multimedia nodes and the buffer can be uniform.

2.2.4 ERTP Protocol

In [27], authors proposed an energy-efficient and reliable transport protocol (ERTP) to balance reliability and energy consumption of data transmission for streaming applications in WSNs by adapting the maximum number of hop-by-hop retransmissions in a distributed manner and employing implicit acknowledgement (iACK) based on the information obtained by means of overhearing. In WSNs, the transmitter can overhear forwarding transmissions at the receiver and interpret them as iACKs. If no acknowledgment is received after a certain timeout, the transmitter determines that the receiver failed to receive the packet. Thus, it retransmits the packet.

However, the determination of a timeout interval during which the transmitter has to wait for an iACK is non-trivial since the timeout interval is dependent on the time it takes the receiver at one level lower to forward the received packet. A short timeout interval will result in unnecessary retransmissions and wasted energy, while a long timeout interval will result in inefficient capacity utilization. In ERTP, the authors proposed a distributed algorithm to estimate the retransmission timeout that can save up to 50% of energy consumption when compared to other approaches.

ERTP is an energy-efficient transport protocol, and it achieves high reliability in data transmission by employing the information obtained by means of overhearing for the recovery of lost data. The weakness of the protocol is that it has to assume that the transmission rate is low and network congestion is negligible.

2.2.5 EEIT Protocol

In [29], authors proved that image transmission over multiple hop WSNs is feasible by using a combination of an energy efficient image processing architecture and a reliable transmission protocol named EEIT. They used a novel FPGA architecture to extract and transmit only updated objects from the background image, resulting in reduced data size and energy consumption for image transmission. Moreover, they used command messages to establish a private and reliable path for image transmission and they implemented a simple slotted transmission protocol that allows a multimedia node to transmit each image packet with a sufficient delay such that only one packet remains on the whole data path. This implies that it excludes multiple concurrent transmissions to avoid collision. Furthermore, a slot big enough for a node to perform hop-by-hop retransmission is used to maintain transmission reliability in case of transmission failure. However, the conservative packet transmission interval and hop-by-hop retransmission can increase the end-to-end delay of image and the retransmissions on the identical link may rather degrade reliability by increasing traffic.

2.2.6 EMP Protocol

In [24], the authors proposed an efficient multi-path pipeline transmission (EMP) protocol that uses a slot big enough for a node to perform hop-by-hop retransmission and employs channel hopping for pipeline transmission. They not only removed the bubble slot problem that a source node skips one slot within the cycle time by dispatching packets on multiple disjoint paths, but also, they improved the pipeline stall problem that a sink does not receive a packet during one cycle time if a packet is lost by adjusting the pipeline cycle time such that slot length is increased to allow hop-by-hop retransmissions. However, it still suffers from image transmission delay due to the use of long pipeline cycle time and path-wide retransmission, and from energy consumption due to the use of retransmission and multiple paths.

2.3 Applications

Multimedia surveillance sensor networks: WISNs have the potential to be used to prevent crime and terrorist attacks. Based on collected still images and the use of computer vision techniques, WISNs can identify missing people and determine criminals, or infer and record other relevant activities such as burglaries, car crashes, and traffic violations.

Traffic avoidance, enforcement, and control systems: In large cities, it can be possible to monitor and manager car traffic by deploying services that can provide traffic routing guidance to avoid congestion. Furthermore, intelligent parking guidance systems relied on WISNs will detect available parking slots and provide drivers with automated parking advice, leading to the improvement of mobility in urban areas. WISNs can also retrieve aggregate data such as a number of cars and average speed, detect violations, or save images in the event of accidents for later analysis.

Advanced health care delivery: A WISN and a cellular network can be combined to form a smart healthcare system that can provide high-quality medical services. The physiological data collected by WISNs may be stored for a long time at remote medical centers for medical investigations. In addition, the WISNS can also be used to monitor the condition of patients, detect the behavior of patients, and infer abnormal situations.

Environmental and structural monitoring: WISNs are used to keep an eye on the condition of bridges or other civilian structures. They are also used to determine the evolution of sandbars by oceanographers.

2.4 Research Topics

The design of a cross-layer protocol for WISNs: The correlation characteristics and functional inter-dependencies among the communication layers should be utilized to improve the performance of WISNs. The design of a cross-layer protocol appears to be the most promising replacement for inefficient traditional layered protocols. Recent research has shown that cross-layer design protocols that cover the routing, transport, and MAC layers lead to significant improvements in the reliability of data transmission and energy conservation.

Deployment and coverage control: Coverage control becomes critical in WISN because multimedia sensors can determine objects in a field of view (FoV) and capture images with direction-sensitivity. Thus, the research on the deployment of multimedia nodes and coverage control is more meaningful. Local information of neighboring MNs can be exchanged to identify the most beneficial orientations of their coverage and guarantee the coverage integrity and communication connectivity of WISNs.

In-network information processing: Delivering raw images in WISNs requires a large bandwidth and huge energy expenditure. Hence, it is required to perform data fusion or processing in-network. The use of data fusion and data processing at MNs that have high computational capability, can not only reduce the high bandwidth demand but also shorter end-to-end delay of image transmission. Encoding and decoding, target detection and tracking, object classification and identification are examples of general processing functions, and they can apply in WISNs.

Security in WISNs: WISNs are more vulnerable to attacks because they use the 2.4 GHz industrial scientific medical band. To ensure message authenticity, integrity, and confidentiality, security in WISNs should be considered during the network design phase, while maintaining the network's efficiency and scalability.

Chapter 3

PIPEPINED COOPERATIVE TRANSMISSION PROTOCOL

3.1 Network Model

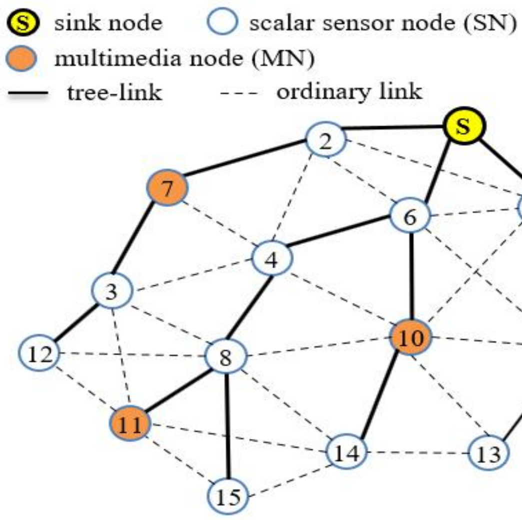


Figure 3. A model of a wireless image sensor network

The considered WISN consists of a sink or a server, a number of scalar sensor nodes (hereafter abbreviated as SNs), and some multimedia nodes (hereafter abbreviated as MNs). The sink is wall-powered, whereas SNs and MNs are battery-powered. All nodes form a tree originating from the sink. A node is said to be a *tree-node* if it belongs to a tree. Otherwise, it is an *orphan-node*. Two nodes that can directly communicate with each other are said to have an *ordinary link*. In particular, a link between a *tree-node* and its parent is called a *tree-link*.

Figure 3 shows a tree with five tree levels, which consists of one *sink*, 11 *SNs*, and 3 *MNs*. The solid lines and the dashed lines represent *tree-links* and *ordinary links*, respectively. A sink collects data from *SNs* at regular intervals while it requests images from *MNs* by broadcasting a command message on an on-demand basis.

3.2 Motivation

Three issues in sending an image in WMSNs include reliability in data transmission, image transmission delay, and energy efficiency. First, the reliability directly affects the quality of the image recovered at the receiver. Thus, it is a basic requirement to reduce a packet loss ratio. Most of the existing approaches rely on redundant transmission on multiple paths, or retransmissions of failed packets [24], [29]; however, they have to undergo energy consumption or increased error rate due to the increased traffic in low-bandwidth and error-prone WSNs.

Second, a server can involve time constraints for image acquisition. Application may fail to take timely measures if it gets an image over a required time bound. For example, let us consider a fire detection system. A server collects scalar data from sensor nodes periodically and judges whether or not the target field involves a fire. It is not possible to judge the situation with 100% confidence since sensor data inherently contain errors. Thus, a server can ask the camera sensor node to send a still image to verify the occurrence of a fire. In this case, the end-to-end delay of the still image has to be bounded to a certain time limit.

Third is to reduce the energy consumption of nodes and balance energy consumption among the nodes to increase network lifetime. Especially with surveillance applications that require frequent delivery of images, it is highly critical to manage energy consumption during image transmissions. In general, the transmission of scalar data is disabled during image transmission since it can disturb the transmission of an urgent image. Therefore, the nodes not involved in image transmission can be turned off. This derives another problem in that every node has to judge that it is not on the cooperative path and when to wake up after sleeping.

The proposed image transmission protocol named PCT can address the above three problems effectively. First, a cooperating node can improve the reliability of data transmission. Image packets have to be transmitted sequentially along the path. As in EEIT [29], if a single path is used in WSNs with a relatively high error rate, a packet can easily be lost while it is travelling to the sink. Under EMP [24], in which multiple disjoint paths can carry different packets, each path works independently. Thus, the possibility of a packet loss on each path is the same as EEIT. Furthermore, the simultaneous transmission of multiple packets along different paths can easily be exposed to collisions unless the packets are protected by different channels. For improvement of reliability, they often rely on *hop-by-hop* and/or *path-wide retransmission*. However, retransmissions may increase end-to-end delay and energy consumption, even with a pipelined transmission. The new approach deals with the reliability problem by using a cooperative path in which every node on the tree path from itself to the sink has a cooperating node that can salvage lost data. Then, the cooperating node assists its counterpart using the information obtained by means of overhearing messages or data in order to forward a packet reliably to the nodes one level lower. This does not use retransmission, thereby making a packet move fast toward a sink.

Second, the new approach employs the pipelined transmission technique. A node cannot receive and transmit data at the same time under the IEEE 802.15.4 standard. However, one node at level l and another node at level l-2 can transmit data simultaneously by using the *channel-assisted slot reuse technique* [38]. For example, in Figure 3, nodes 4 and 11 can transmit data simultaneously if they use different channels. Furthermore, according to one empirical study [21], the interference range is approximately twice as long as the transmission range. Thus, two nodes, four hops away from each other, can transmit data safely using the same channel, referred to as the *spatial slot reuse technique*.

3.3 Notations and Messages

Some acronyms and terminologies that will used throughout this thesis are presented in the Table 1.

Table 1. Acronyms and terminologies

SPS	A set of secondary parents of a node
Nbrs	A set of neighbors of a node
slotLen	The length of a time slot
path node	A node on a tree path from a source multimedia node to a sink
cooperating node	A node selected by a path node at the same tree level

Two messages used to establish a cooperative path are defined as follows.

- *IREQ*: A sink sends an *image request message* to a source multimedia node (*SrcMN*): *IREQ* = (*SrcMN*, *startIframe*, *level(SrcMN)*, *nPackets*) where *startIframe* is a global time when the sink issues *IREQ* to set up image transmission; *level(SrcMN)* is the level of *SrcMN*, and *nPackets* is the number of packets corresponding to one image that *SrcMN* has to transmit.
- *IREP: Upon* receiving *IREQ*, *SrcMN* replies with an *image reply message*: *IREP* = (*cn* = 0, *SPS*) where *cn* = 0 indicates that *SrcMN* does not have a cooperating node, and *SPS* indicates the set of its secondary parents. While *IREP* is traveling toward the sink, every forwarding node includes its own *cn* and *SPS* in *IREP* before forwarding it.

3.4 Protocol Design

The proposed pipelined cooperative transmission (PCT) protocol is described formally here, starting with an explanation of the protocol structure, followed by the path construction, channel and slot scheduling, and image transmission. The proposed protocol covers both a data link layer and a network layer, referred to as a *cross-layer* protocol, and lies between a physical layer and an application layer. It works with other cross-layer protocols: the slotted sense multiple access broadcast (SSMAb) protocol [39] for command transmission, and the slotted sense multiple access (SSMA) protocol [40] for scalar data collection. This study focuses on the fast and reliable image delivery, the issues regarding tree construction and time synchronization are performed once during the ITP and tree maintenance is performed during the CTP if necessary.

3.4.1 Protocol Structure

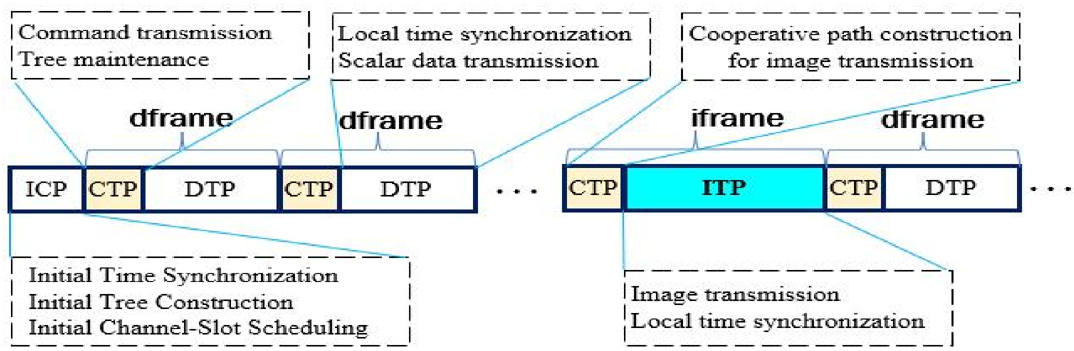


Figure 4. Protocol structure

The protocol structure consists of an initial control period (ICP) and a repeating data frame (*dframe*), with an image frame (*iframe*) interleaved on demand.

During the ICP, the protocol performs global time synchronization, tree construction, and the scheduling of channels and slots. The following frame can be either *dframe* or *iframe*, depending on the command that a sink sends at the start of the command transmission period (CTP).

During the CTP, a sink can send a message to a particular node or a group of nodes using a command transmission protocol, say SSMAb [39]. The message can be a warning message destined for a particular node, a command to ask nodes to perform tree maintenance, or *IREQ* to establish a cooperative path for image transmission. Therefore, at the beginning of the CTP, every node enters active mode, waiting for a message from the sink; if a node does not receive any message during a short time, it goes to sleep to save energy. If a sink broadcasts *IREQ* during the CTP, the current frame becomes the *iframe* in which the CTP is followed by an image transmission period (ITP).

If a current frame is the *dframe*, every node can send one sensor data to the sink using a data collection protocol, say SSMA [40]. If a current frame is the *iframe*, the cooperative path is established for image transmission during CTP and the following period becomes the ITP. Figure 4 illustrates the protocol structure with the brief explanation of protocol operation within each period. The rest of discussion in this chapter is narrowed down to the *iframe*.

3.4.2 Cooperative Path

This section explains how a cooperative path (CP) is constructed. A CP can be expressed as follows:

$$CP = \left((p^L, null), (p^{L-1}, c^{L-1}), \dots, (p^i, c^i), \dots, (p^2, c^2) \right)$$

where p^L is a source multimedia node, L = *level*(*SrcMN*), and p^i and c^i indicate a path node and a cooperating node at level i, respectively.

It is assumed that the sink has the address and the tree level of the selected multimedia node (as well as the number of image packets that has to be transmitted by the multimedia node) *a priori*. A sink can receive this information either when the WSN is deployed or by collecting information from multimedia nodes. To establish a cooperative path, the sink floods the network with the *IREQ*. Upon receiving the *IREQ*, the selected multimedia node sends *IREP* = (*0*, *SPS*) to its parent. Note that a source multimedia node does not need a cooperating node since it always broadcasts a packet. Therefore, *cn* = 0. Upon receiving the *IREP*, a path node selects a cooperating node that is common to both the *SPS* in the *IREP* and its neighbor set, *Nbrs*. This makes sure that the selected cooperating node connects to both the path node at the same level and the path node at one higher level, forming a stable triangle connection among the three nodes. Since every path node sends an *IREP* that includes its cooperating node, *cn*, a node that overhears the *IREP* can determine whether it was selected as a cooperating node or not. This process continues until the *IREP* reaches the sink. The cooperative path construction algorithm is detailed in Algorithm 1

Algorithm 1. Cooperative path construction

// *x.SPS*: The *SPS* of node *x*;
// *x.Nbrs*: The *Nbrs* of node *x*;
1: At multimedia node *x* that receives *IREQ*:
2:     send *IREP = (0, x.SPS)*;
3: At node *x* that receives *IREP = (*z*, y.SPS)*:
4:     node *x* becomes a path node
5:     **if** *y.SPS* = ϕ **then** // no secondary parent
6:         send *IREP = (0, x.SPS)*;
7:     **else**
8:         select *cn* $\in$ y.*SPS* $\cap$ x.*Nbrs*;
9:         send *IREP = (cn,* x.*SPS)*;
10:    **endif**
11: At node *x* that overhears *IREP = (*z*, y.SPS)*:
12:    **if** *x* = z and *level (x) = level (y)* **then**
13:        node x is a cooperating node;

Referring to Figure 3, let us see how Algorithm 1 establishes a cooperative path. Upon receiving *IREQ*, target multimedia node 11 transmits *IREP* = (*0, {3, 14}*) to its primary parent, 8 (line 2). Path node 8 selects one of the secondary parents, 3 or 14. Assume it chooses node 14 as a cooperating node (line 8). Then, it sends *IREP = (14, {10})* to node 4. When node 14 overhears *IREP*, it determines itself to be a cooperating node, since it is at the same level as node 8 (lines 12 and 13). This process continues level by level until the *IREP* reaches the sink. The resulting cooperative path: *CP* = ((*11*, *null*), (*8*, *14*), (*4*, *10*), (*6*, *2*)).

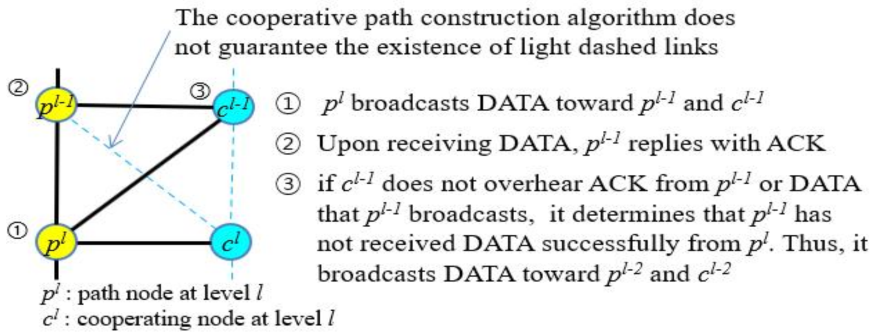


Figure 5. The operating principles of the cooperative path to improve reliability

There are two clear advantages with the use of a cooperative path. Referring to Figure 5, when a path node, p^l, broadcasts a packet, both path node p^{l-1} and its cooperating node, c^{l-1} at level *l-1*, receive the packet. Since c^{l-1} is surely connected to p^{l-1} according to the cooperative path construction algorithm, it can determine if p^{l-1} has received a packet successfully or not by checking the ACK or the packet that p^{l-1} replies with or broadcasts. If c^{l-1} judges that p^{l-1} failed to receive the packet, it can salvage the packet by broadcasting the previously saved packet. This improves the reliability of transmission. Furthermore, c^{l-1} broadcasts the same packet rather than relying on retransmission by p^l, thereby making the packet move faster toward the sink.

3.4.3 Channel and Slot Scheduling

One a cooperative path is established, a distinct slot is allocated to a tree level, and only the nodes at the same level cooperate in forwarding packets to the nodes one level lower within the allocated slot. This avoids packet collisions and eases energy management. Furthermore, if nodes at every other tree level use different channels, packets can be transmitted in parallel. If a spatial slot reuse technique [40] is employed, two channels are sufficient for parallel transmission. In this section, channel scheduling and slot allocation are discussed formally.

a. Channel Scheduling

Suppose that a multimedia node is located at tree level *L*. Since a sensor node works in half-duplex mode, it can either transmit or receive a packet. Furthermore, the nodes at even or odd levels can send packets simultaneously using different channels. Thus, $\lfloor L/2 \rfloor$ channels are required for maximum parallelism in data transmission. However, if the spatial slot reuse technique is employed, it is sufficient to use two channels, such that nodes at every other level use different channels, and nodes four hops away from each other use the same channel. Note that the signal interference range is approximately twice the transmission range [21]. The following formulas are derived to determine Ch_l^{Tx} and Ch_l^{Rx} that indicate a sending channel and a receiving channel, respectively, at level *l [41]*:

$$
\begin{aligned}
Ch_l^{Rx} &= ((Q(l) + R(l))\ MOD\ 2) + 1 \\
Ch_l^{Tx} &= ((Q(l-1) + R(l-1))\ MOD\ 2) + 1,\ l \neq 1
\end{aligned} \tag{1}
$$

where $Q(l) = DIV\,2, R(l) = l\ MOD\ 2$ and value 2 indicates two channels.

Table 2. Transceiver channels according to the level of a node

Level (*l*)	Ch_l^{Tx}	Ch_l^{Rx}
4k + 1	1	2
4k + 2	2	2
4k + 3	2	**1**
4k + 4	**1**	1

Table 2 summarizes channel allocations for four continuous levels, *4k+1*, *4k+2*, *4k+3*, and *4k+4*, $k \geq 0$, according to Eq. (1). Note that channel allocations are repeated for different *k* values. For example, a node at level 4k + 4 sends a packet on Channel 1, while a node at level 4k + 3 listens to the same channel, as underlined in the table.

b. Slot Scheduling

Given *level*(*SrcMN*) with *nPackets* as the number of packets to be transmitted, the number of slots (*nSlots*) required to deliver *nPackets* can be calculated as follows:

$$nSlots = (L-1) + 2\times(nPackets-1) = 2\times nPackets + L - 3 \tag{2}$$

In Eq. (2), the sink receives the first packet after latency of (L - 1) slots, and then a new packet per two slots for the remaining packets (*nPackets - 1*). Then, every node can calculate the length of the image transmission period as follows:

$$ITP = nSlots \times slotLen \tag{3}$$

When a node receives IREQ that includes *startIframe*, it can slice *ITP* into *nSlots* independently, starting with *startITP* (= *startIframe* + *CTP*). Then, every node, *x*, can derive $RxTime(x,k)$ and $TxTime(x,k)$, which indicate the receiving time and the sending time, respectively, of the k^{th} packet, and $EndTxTime(x)$, which indicates the ending time of the whole image transmission as follows:

for k =1 to *nPackets* do

$$\begin{aligned} RxTime(x,k) &= TxTime(x,k) - slotLen \\ TxTime(x,k) &= StartITP + (L - level(x)) \times slotLen + 2(k-1) \times slotLen \end{aligned} \tag{4}$$

$$EndTxTime(x,k) = StartITP + (L - level(x)) \times slotLen + (2 \times nPackets - 1) \times slotLen \tag{5}$$

Figure 6 illustrates an example of channel and slot scheduling, where ITP is sliced into *nSlots* corresponding to the number of slots required to deliver *nPackets* from a multimedia node at level *L*. It also shows channel allocation as the numbers in the slots when *L* = 4*k*+4.

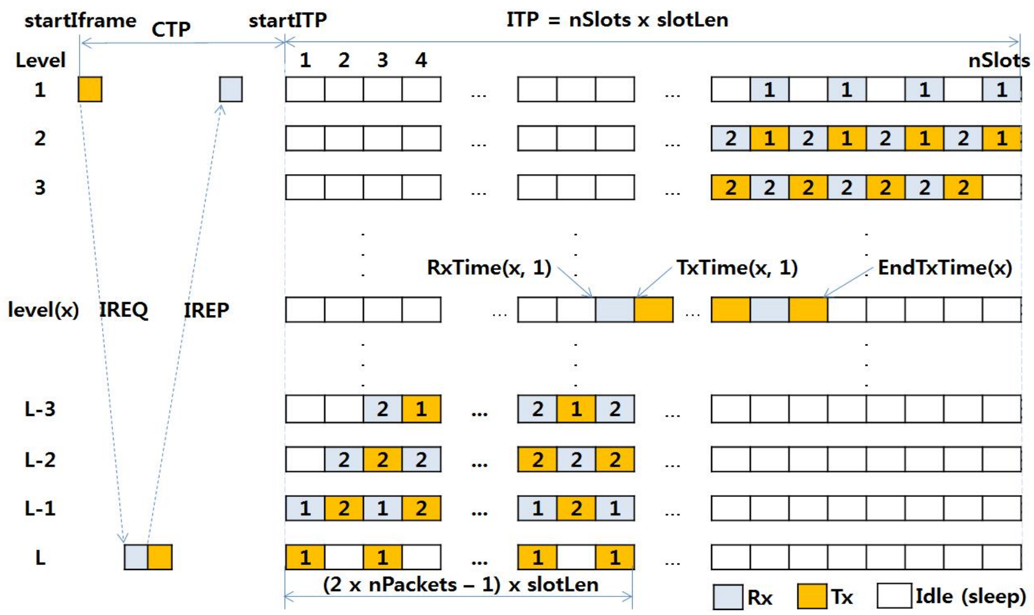


Figure 6. Pipelined transmission with channel and slot scheduling, assuming that level L is 4k+4

3.4.4 Image Transmission

a. Energy Management

If the sink needs an image from a specific multimedia node, say *SrcMN* at level *L*, it broadcasts *IREQ* to establish a forwarding cooperative path from *SrcMN* to the sink at the beginning of a command transmission period. Upon receiving *IREQ*, *SrcMN* responds with *IREP*, which enables path construction. Upon receiving *IREQ* = (*SrcMN*, *startIframe*, *level*(*SrcMN*), *nPackets*), a node calculates *startITP* as follows:

$$StartITP = StartIframe + CTP \tag{6}$$

Suppose that $L = level(SrcMN)$. After broadcasting *IREQ*, every node waits for IREP from *SrcMN*. Let *msgTxTime*(*M*) indicate the transmission time of message *M*. Then, *maxIREPWait*(*x*) as the maximum *IREP* waiting time of node *x* can be calculated as follows:

$$\begin{aligned} maxIREPWait(x) &= (L - level(x)) \times BSS \\ &\quad + (L - level(x) + 1) \times msgTxTime(IREP) + laxityTime \end{aligned} \tag{7}$$

where the first term is the delay from the time when node x broadcasts *IREQ* until *SrcMN* at level L receives *IREQ* using the command transmission protocol (SSMAb) [39]. BSS stands for *broadcast sharable slot*. The second term is the delay from the time when *SrcMN* sends *IREP* until cooperating node x overhears IREP issued by its counterpart path node to determine whether it is a cooperating node or not. The third term, *laxityTime*, indicates a guard time to reflect the time variation in processing a control message, since a cooperating node always has to overhear *IREP* transmitted by its counterpart path node, plus *unity* is required. Every node (say, for example, x) goes to sleep during the image transmission time, unless it is a path node or a cooperating node, until *maxIREPWait*(x) expires.

b. Reliable and Predictable Image Transmission

Node x starts the k^{th} *Rx slot* at *RxTime*(x, k) and the k^{th} *Tx slot* at *TxTime*(x, k). At the beginning of *Rx slot*, a path node and a cooperating node always clear a receiving buffer to save a packet that they receive or overhear. Upon receiving a packet, a path node responds with ACK, whereas a cooperating node just saves the received packet. In a *Tx slot*, the path node broadcasts the received packet immediately, taking priority over its cooperating node. Conversely, the cooperating node broadcasts the previously saved packet only if it does not overhear ACK or the packet that the path node replies with or broadcasts within a small fixed delay, *delayslot*, to avoid a collision:

$$delayslot \geq \text{ transmission time of one bit} \tag{8}$$

In this process, note that data collision can never occur, and a path node never rebroadcasts the packet. Furthermore, the ITP can be calculated based on the number of packets and the level of the source multimedia node, as expressed in Eq. (3). Thus, the maximum delay for image transmission can be predicted in advance. The PCT protocol for reliable image transmission is detailed in Algorithm 2.

Algorithm 2. The pipelined cooperative transmission protocol

Multimedia node	**Path node**	**Cooperating node**

Upon receiving *IREQ*:
k = 1;
set *TxTimer* = *TxTime*(*x*, *k*);
send *IREP* along tree path;
 	
When *TxTimer* expires:
broadcast DATA;
set *ACKTimer* = *WaitTime*(*ACK*);
k = k + 1;
set *TxTimer* = *TxTime*(*x*, *k*);
 	
When *ACKTimer* expires:
go to sleep;
 	
Upon receiving *ACK*:
do time synchronization;
clear *ACKTimer*;
go to sleep;
	When x becomes a path node:
	k = 1;
	set *RxTimer* = *RxTime*(*x*, *k*);
	
	When *RxTimer* expires:
	wait for DATA;
	save DATA;
	set *TxTimer* = *TxTime*(x, k);
	reply with *ACK*;
	
	When *TxTimer* expires:
	if there is DATA **then**
	broadcast DATA;
	set *ACKTimer* = *WaitTime*(*ACK*);
	endif
	k = k + 1;
	set *RxTimer* = *RxTime*(*x*, *k*);
	
	When *ACKTimer* expires:
	go to sleep;
	
	Upon receiving *ACK*:
	do time synchronization;
	clear *ACKTimer*;
	go to sleep;
		When x becomes a cooperating node:
		k = 1;
		set *RxTimer* = *RxTime*(*x*, *k*);
		
		When *RxTimer* expires:
		wait for DATA;
		save DATA;
		set *TxTimer* = *TxTime*(*x*, *k*);
		cn-txflag = **False** ;
		set *ACKTimer* = *WaitTime*(*ACK*);
		
		When *ACKtimer* expires:
		// if did not receive ACK
		set cn-txflag = **True**;
		Upon receiving *ACK*:
		do time synchronization;
		clear *ACKTimer*;
		go to sleep;
		
		When *TxTimer* expires:
		wait for *delayslot*;
		if (channel is idle) and (cn-txflag = **True**) **then**
		broadcast DATA;
		cn-txflag = **False**;
		endif
		k = k + 1;
		set *RxTimer* = *RxTime*(*x*, *k*);

3.4.5 The Lower Bound of Slot Length

In [42], $msgTxTime(x)$ is calculated as follows:

$$msgTxTime(x) = t_{mr}(x) + t_{turnon} + t_{CCA} + t_{ppd} + t_{Tx}(x) + t_{rm}(x) \tag{9}$$

where t_{mr} is the time to transfer a message from the microcontroller unit (MCU) to the radio chip buffer, t_{turnon} is the delay to turn on the radio chip, t_{CCA} is the clear channel assessment time, t_{ppd} is the physical layer processing delay, t_{Tx} is the time to transmit a message, and t_{rm} is the time to transfer a message from the radio chip to the MCU on the receiver side. The propagation delay is negligible, compared to the packet transmission time.

The length of a slot (*slotLen*) is the time taken to transmit packet x and to receive the ACK. Thus, the following inequality is used to determine *slotLen*:

$$slotLen \geq delayslot + msgTxTime(x) + msgTxTime(ACK) \tag{10}$$

where $delayslot$ corresponds to the length of 20 symbols as the sum of PPD (= the length of twelve symbols) and CCA time (= the length of eight symbols). Since the bandwidth is 250 kbps in the IEEE 802.15.4 physical layer standard [25], given that one symbol corresponds to four bits, $t_{CCA} = 0.128\ ms, t_{ppd} = 0.192\ ms$, and $delayslot = 0.32\ m$.

The transmission of packet x adds additional information of 16 bytes such as a synchronization header (SHR) of five bytes, a physical header (PHR) of one byte, a MAC header (MHR) of eight bytes (i.e., two bytes of frame control, 4 bytes of address information, one byte of timestamp and one byte of source tree level), and a MAC footer (MFR) of two bytes. Therefore, transmission time $t_{Tx}(x)$ of packet x is given as follows:

$$t_{Tx}(x) = 0.032 \times (16 + payload(x)) \quad (ms) \tag{11}$$

where 0.032 ms is one byte transmission time in 250Kbps and payload(x) indicates the payload of packet x.

In simulation, it is assumed that $t_{turnon} = 0$, $t_{mr}(x) = 0$, and $t_{rm}(x) = 0$ in Eq. (9). Thus, $msgTxTime(x)$for simulation is calculated as follows:

$$msgTxTime(x) = 0.832 + 0.032 \times payload(x) \quad (ms) \tag{12}$$

Since *payload*(ACK) is *zero*, $slotLen_{LB}(sim)$ as the lower bound of *slotLen* required to transmit message x in simulation can be obtained from Eq. (10) as follows:

$$slotLen_{LB}(sim) = 1.984 + 0.032 \times payload(x) \quad (ms) \tag{13}$$

Assuming that the payload of a message is 110 bytes long $slotLen_{LB}(sim) = 5.504$ ms.

Meanwhile, in experiment, the testbed was built using *uMote* developed in our laboratory using the CC2630 [43] that integrates an ARM Cortex M3 and the CC2420 radio chip. According to the datasheet [43], $t_{turnon} \approx 100$ µs, and in [39], t_{mr} and t_{rm} are given as follows:

$$t_{mr}(x) = (MHR + payload(x)) \times 2 \times 10^{-3} \quad (ms)$$
$$t_{rm}(x) = (MHR + MFR + payload(x)) \times 2 \times 10^{-3} \quad (ms)$$

By applying these values to Eq. (9), *msgTxTime*(x) for experiment is calculated as follows:

$$msgTxTime(x) = 0.868 + 0.036 \times payload(x) \quad (ms) \tag{14}$$

Then, $slotLen_{LB}(exp)$ as the lower bound of *slotLen* for experiment is given as follows:

$$slotLen_{LB}(exp) = 2.056 + 0.036 \times payload(x) \quad (ms) \tag{15}$$

for the same payload of 110 bytes that were used in simulation, $slotLen_{LB}(exp) = 6.016$ *ms*.

3.5 Implementation

3.5.1 Testbed

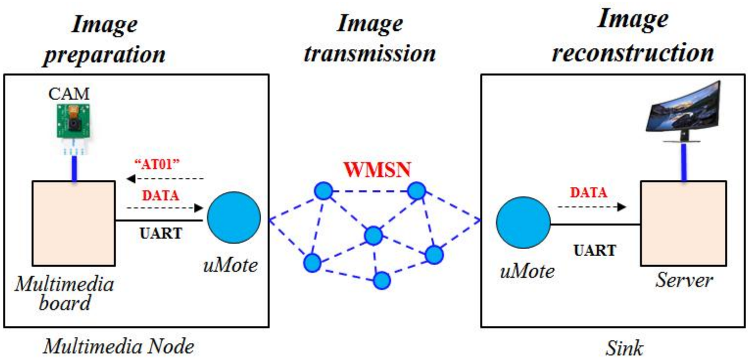


Figure 7. Testbed for image transmission over WMSN

Figure 8. Multimedia node

The testbed consists of three functional parts: Image preparation (IP), image transmission (IT), and image reconstruction (IR). The IP part, the IT part, and the IR part deal with the preprocessing of an image that includes image capture, compression, and the generation of image packets, the transmission of data on WMSN that employs the PCT protocol, and the image recovery, respectively as illustrated in Figure 7. The IP part includes multimedia board, *uMote*, and camera module (*CAM*). The multimedia board should have low power consumption and high-speed calculation since it deals with image processing. Thus, we used Raspberry pi 3 model B (*RPi*) that includes 1GB of RAM and a powerful 1.2 GHz quad-core ARM Cortex-A53 (64Bit) CPU. It has a HDMI socket, SD slot card memory, a couple of USB connectors, an ethernet connector, and a camera serial interface (CSI). The CAM module and *uMote* are connected to the multimedia board by a camera serial interface (*CSI*) port and universal asynchronous receiver-transmitter (UART), respectively. The multimedia board captures an image from the CAM and forwards the image to *uMote* that in turn sends it towards a server using WMSN. The CAM uses Raspberry Pi Camera Module V2 and the *uMote* was implemented with the CC2630 chip that integrates ARM Cortex M3 and CC2420 radio chip. The *RPi* uses Raspbian, a Debian-based operating system, and the *Qt* creator version 4.14.0 [43] to control the CAM module and process the image data. A multimedia node is shown in Figure 8.

The network for image transmission consists of a number of *uMotes* that form a multi-hop wireless multimedia sensor network based on the IEEE 802.15.4 standard [3]. Every node has a routing capability. A server has a sink node that receives the image using WMSN and then forwards it to a server. It reassembles the image packets to recover the original image.

3.5.2 Image Preparation

A server collects data from scalar nodes in the target monitoring zone at regular intervals. If it judges that an abnormal situation has occurred in the target zone based on the analysis of collected scalar data, it requests an image from the multimedia node, referred to as a *source MN* (*SrcMN*), by sending an *image request* (*IREQ*) message. Upon receiving *IREQ*, the *uMote* in *SrcMN* sends a command, *AT01*, to the multimedia board. Upon receiving the *AT01* command from the *uMote*, the multimedia board captures an image using the CAM module. In our testbed, the image has a resolution of 100 pixels in width and 80 pixels in height. The resolution of the captured image can be changed by adjusting the image resolution. The multimedia board compresses and encodes the captured image by using the *JPEG 2000* standard that is used popularly for WSNs [4]. Then, it saves the compressed image in the buffer, segments the image into 100-byte data units and formats each data unit to produce a packet or frame. We use a packet and a frame interchangeably for convenience. The packets are transmitted to the *uMote* via UART.

A packet consists of the start of frame (2 bytes), sequence of a frame (1 byte), length of payload (1 byte), payload (100 bytes), and end of a frame (2 bytes), as shown in Figure 9. Note that if only 1 byte is used as the start or the end of the frame, an error in frame detection can be made because the pixel values of an image range from 0x00 to 0xFF.

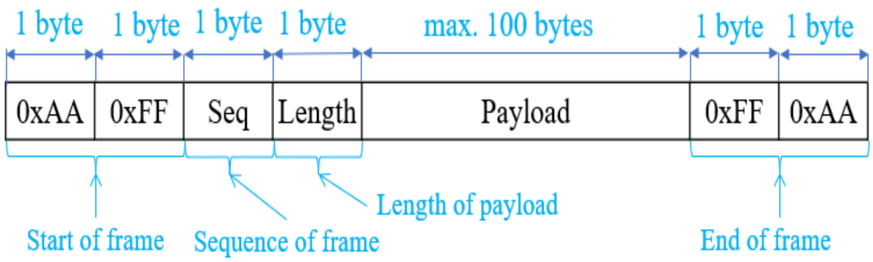


Figure 9. A packet format

3.5.3 Image Transmission

Upon receiving an image packet, the *uMote* transmits it towards the server using WMSN, in which the PCT protocol is implemented. The operation of the PCT protocol is described as follows. First, the PCT establishes a *cooperative path* that consists of *path nodes* residing on the tree path and *cooperating nodes* that assist the path nodes to improve reliability of data transmission. Each path node selects one cooperating node during path establishment. Then, a *path node* cooperates with its *cooperating node* to deliver packets reliably to nodes at one level lower within the distinct slot allocated to each tree level. Finally, two channels are used and scheduled to allow the transmission of packets in a parallel pipelined manner.

3.5.4 Image Reconstruction

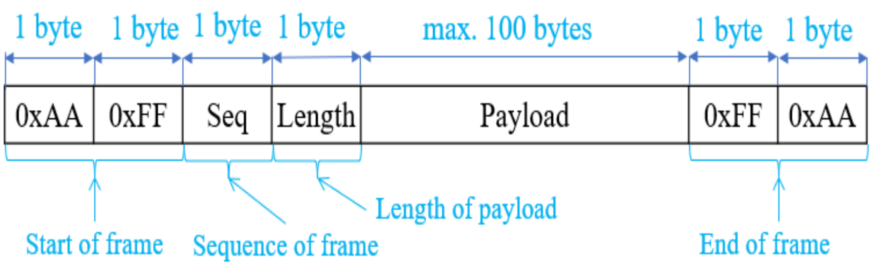


Figure 10. A packet that is reformatted by a sink

When a sink receives a packet, it saves only the payload of the packet on its buffer, and reformats the payload as shown in Figure 10. The format includes the start of a frame, payload, and the end of the frame. Then, the sink forwards each packet to the server that decompresses and saves the reconstructed image.

3.6 Performance Evaluation

In this section, the PCT protocol is compared with two recent multimedia protocols: EMP [24] and EEIT [29]. First, the key features of those three protocols are contrasted. The total time to deliver one image from a *SrcMN* to the sink is analyzed, and then, the average energy consumption and the packet delivery rate for sending one image are evaluated comparatively, and the simulation and experimental data for PCT are compared.

3.6.1 Comparison of Features from Related Protocols

In this subsection, we contrast the features of the proposed protocol PCT and other two protocols EMP and EEIT for some performance metrics that are often used in image transmission and compare their performance using the relative indicators such as high, medium, and low.

EMP allows hop-by-hop retransmissions by increasing slot length to alleviate the pipeline stall problem and also can remove the bubble slot problem by dispatching packets on multiple disjoint paths. However, each path may not be stable due to the multipath fading effect and/or internal and external interferences. Therefore, EMP is more likely to experience packet loss, thereby increasing hop-by-hop retransmissions and/or path-wide retransmissions for reliability. This will increase end-to-end delay. On the contrary, PCT allows the bubble slot problem by dispatching a new packet in every other slot; however, it uses a robust cooperative path that enables the removal of retransmissions, thereby making a packet move fast. Furthermore, PCT is free from internal interference by using the channel assisted slot reuse and spatial slot reuse techniques in pipelined transmission.

Meanwhile, EEIT achieves high energy efficiency by disabling the nodes that are not on the path during image transmission. Furthermore, it allows only one packet to be transmitted on the path, thereby removing internal interference completely; however, this tends to increase the image transmission time. EMP can improve network lifetime by balancing energy consumption on multiple paths; however, hop-by-hop or path-wide retransmission can increase the energy consumption of nodes, although it improves transmission reliability. PCT saves energy by disallowing retransmissions. It also manages duty cycles within the scheduled slots and turns off the nodes that are not on the cooperative path.

EEIT tries to improve reliability by using hop-by-hop retransmission and by removing internal interference. However, if the path is not stable, EEIT can still suffer from packet loss. In comparison to EEIT, EMP improves reliability by using both path-wide and hop-by-hop retransmissions. It may overcome the reliability problem to some extent; however, it can increase image transmission time and the energy consumption of nodes. Conversely, PCT improves reliability without sacrificing a delay in image transmission by using a cooperative path.

The discussion is summarized in Table 3. You can find the quantitative comparison for the same performance metrics in the following subsections: 3.6.2, 3.6.3, and 3.6.4.

Table 3. The comparison of key image transmission protocols

Features	EMP	EEIT	PCT
Image transmission method	- Removes the bubble slot problem using multiple paths - Reduces the pipeline stall problem by allowing hop-by-hop retransmissions within the same pipeline cycle time	- Does not use a pipelined transmission - Estimates the end-to-end delay of a packet and adjusts packet transmission interval such that only one packet resides on a path	- Allocates time slots and channels to a tree level for secured transmission - MN transmits a new packet *in every other slot* in a pipelined manner
Reliability	- Tries to avoid collisions using channel hopping on each path against pipelined transmission - Uses hop-by-hop and path-wide retransmission for the lost packets	- Allows only one packet on a single path for secured transmission - Uses hop-by-hop retransmission for reliability	- Avoids collisions using channel and slot scheduling in pipelined transmission - Improves the reliability of hop-by-hop transmission by using a cooperating node
	High	**Medium**	**High**
Energy consumption	- Disables the nodes not on multiple paths and achieves energy balancing by distributing packets on multiple paths - Reduces the number of path-wide and hop-by-hop retransmissions by avoiding collisions using channel hopping	- Disables the nodes not on the path - Reduces the number of hop-by-hop retransmissions by removing internal interference	- Uses a single path and disables the nodes not on the cooperative path - Manages the duty cycles of nodes on a cooperative path based on assigned slots - Does not allow retransmissions
	High	**Low**	**Low**
End-to-end delay	- Removes the bubble slot problem using multiple paths; however, uses path-wide retransmissions and the long pipeline cycle time to allow hop-by-hop retransmissions - Thus, depending on link quality, end-to-end delay can increase with the increase of the hop distance of MN to a sink.	- Does not use pipelined transmission - Allows hop-by-hop retransmissions - Therefore, end-to-end delay can increase with the increase of the hop distance of MN to a sink.	- Allows the bubble slot problem in the pipelined transmission - Make a packet move fast by disallowing hop-by-hop retransmissions - Avoids path-wide retransmissions by achieving the high reliability of hop-by-hop transmission
	High	**High**	**Low**

3.6.2 Analysis

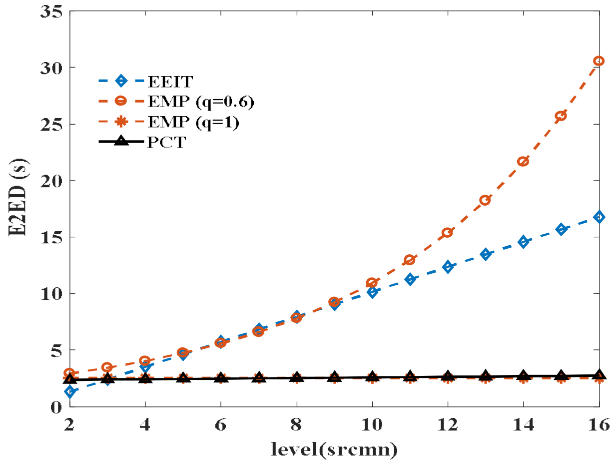


Figure 11. Comparison of the protocols for the lower bound of the end-to-end transmission delay of one image

In this section, we calculate the lower bound of the end-to-end delay (E2ED) to transmit one image from the *SrcMN* at level L to the sink under the three image transmission protocols: EEIT, EMP, and PCT. EEIT works as follows. First, the sink broadcasts a message to establish a path and inform all nodes of the start of image transmission. This delay is denoted by $T_{StartTx}$. Then, the nodes not on the path can go to sleep. Second, the MN transmits packets to the sink along the path, denoted as *ITP*(*EEIT*). Lastly, the sink broadcasts a message to inform nodes of the end of image transmission. This delay is denoted by T_{EndTx}. Thus, E2ED(EEIT) is given as follows:

$$E2ED(EEIT) = T_{StartTx} + ITP(EEIT) + T_{EndTx} \tag{16}$$

In this approach, since the MN transmits a new packet only if the preceding packet arrives at the sink, and every node on the path uses ACK for secured hop-by-hop transmission, ITP(EEIT) is expressed as follows:

$$ITP(EEIT) = (msgTxTime(x) + msgTxTime(ACK)) \times nPackets \times (L-1) \tag{17}$$

where $msgTxTime(x)$ and $msgTxTime(ACK)$ are calculated by using Eq. (12). Therefore, with a payload of 110 bytes and one image of 220 packets, ITP(EEIT) is calculated as follows:

$$ITP(EEIT) = 1140.48 \times (L-1) \tag{18}$$

Assuming that $T_{StartTx} = T_{EndTx} = 100\ ms$, E2ED(EEIT) is given as follows:

$$E2ED(EEIT) = 1140.48 \times L - 940.48\ (ms) \tag{19}$$

Meanwhile, EMP takes four phases. First, it establishes multiple paths from the MN to the sink (T_{Path}). Second, it transmits packets along the multiple paths using pipelined transmission (ITP). Third, the sink requests retransmission of lost packets. The path-wide retransmission delay is denoted by $T_{LostPacket}$. Lastly, it closes the transmission by broadcasting a message. Thus, E2ED(EMP) is calculated as follows:

$$E2ED(EMP) = T_{Path} + ITP(EMP) + T_{LostPacket} + T_{EndTx} \tag{20}$$

Assuming that the lengths of different paths and the link quality value, q, are the same, the value for the ITP is calculated as follows [24]:

$$ITP(EMP) = \frac{n \times (msgTxTime(x) + msgTxTime(ACK)) \times nPackets}{(1-(1-q)^n)^{L-1}} \tag{21}$$

where, in general, $n = 2$ as the number of transmissions to send one packet in each wireless hop. Assuming that each of the three phases (except the second phase) takes 100 ms, E2ED(EMP) is calculated as follows:

$$E2ED(EMP) = 300 + \frac{2 \times 1140.48}{(1-(1-q)^2)^{L-1}}\quad (ms) \tag{22}$$

Meanwhile, PCT has two phases: building a cooperative path (CTP) and sending an image (ITP) in a pipelined manner. Thus, E2ED(PCT) is calculated as follows:

$$\begin{aligned} E2ED(PCT) = CTP + ITP(PCT) = {} & (msgTxTime(IREQ) \\ & + msgTxTime(IREP)) \times (L-1) + nSlots \times slotLen \end{aligned} \tag{23}$$

where $msgTxTime(\text{IREQ}) = \text{size (BSS)} = 20\ ms$ [39]. From Eq. (9), $msgTxTime(IREP) = 0.99\ ms$ with $IREQ = 5$ bytes. From Subsection 3.4.5, since $slotLen_{LB}(\text{sim}) = 5.504\ ms$, E2ED(PCT) is calculated as follows:

$$E2ED(PCT) = 20.99 \times (L-1) + (2 \times nPackets + L - 3) \times 5.504$$
$$= 26.467 \times L + 2384.285\ (ms) \tag{24}$$

Figure 11 compares *E2ED* under the three approaches with an increase in the level of the *SrcMN*, *level*(*SrcMN*). E2ED(EMP) is calculated with two different values for link quality: $q = 0.6$ and $q = 1.0$. It is clear that as *level*(*SrcMN*) increases, E2ED(EEIT) and E2ED(EMP) with $q = 0.6$ increase rapidly, whereas E2ED(PCT) remains constant. E2ED(EMP) with the best link quality ($q = 1$) and E2ED(PCT) remained almost constant (< 2.5 seconds) against the change in *level*(*SrcMN*) because they both use the pipelined transmission.

In conclusion, PCT can achieve highly dependable end-to-end delay with low energy consumption in acquiring an image. However, the end-to-end delay of 2.5 seconds in image acquisition may not be sufficient for the fire detection system, an example discussed in the Motivation part. However, this delay is meaningful compared to the delay of 8 seconds achieved with EMP when the *SrcMN* is located at level 8. Furthermore, the delay can be greatly improved with the use of either two distinct cooperative paths or the better compression technology.

3.6.3 Simulation

a. Simulation Setup

The commercial QualNet simulator (version 5.0.2) was used with the key simulation parameters and values given in Table 4. PCT was compared with EEIT and EMP using scenarios S1 and S2. Scenario S1 is a square with dimensions of 30×30 m^2 with a random node distribution of 30 nodes and the sink as shown in Figure 12, where the MN is selected randomly. Scenario S2 is a long rectangular shape at 15×100 m^2 with a two-way chained deployment of one sink and 21 nodes, as in Figure 13. From Subsection

3.4.5, *slotLen* in the simulation was set to 5.6 *ms*. The Ricean model for a multipath fading effect was used with changing *k*, defined as the ratio of the receiving power in the direct path to that in other paths.

Table 4. Simulation parameters and values.

Parameters	Value
Number of packets per one image	220
Transmission range	-30dBm (≈10 m)
Channel frequency	2.4 and 2.43 GHz
Path loss model	two-ray
Shadowing model	Constant (Mean = 4dB)
Noise factor	10 dB
k value in Ricean fading model	3 ~ 18
Sensor energy model	MicaZ
Battery model	Linear

Three evaluation metrics were used. Firstly, the *packet delivery ratio* (*PDR*) indicates the ratio of the number of packets received at a sink to the total number of packets transmitted by the MN. Second, t*he image delivery ratio* (*IDR*) is the ratio of the number of lossless reconstructed images at the server to the total number of images transmitted by the MN. Note that IDR does not count any blurred images reconstructed with the loss of any packets since an object may not be identified from the blurred images. Third, the *end-to-end delay (E2ED)* of an image is the time span to deliver an image from the *SrcMN* to a sink. Note that the time to turn on the camera and capture an image is deliberately ignored since it varies largely according to the quality of a camera module and also it can be shortened if the additional sensor is used to detect an ambient event and turn on the camera autonomously. The time to compress a captured image at a multimedia node and reconstruct the image at a server are ignored since they are negligible compared to the delay of image delivery.

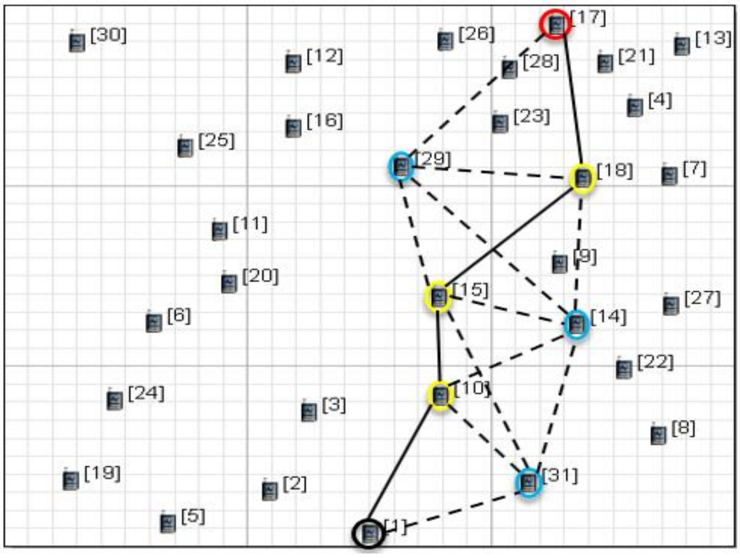


Figure 12. Scenario S1 with a square dimension of 30 × 30 (m2) and random distribution of 1 sink and 30 nodes

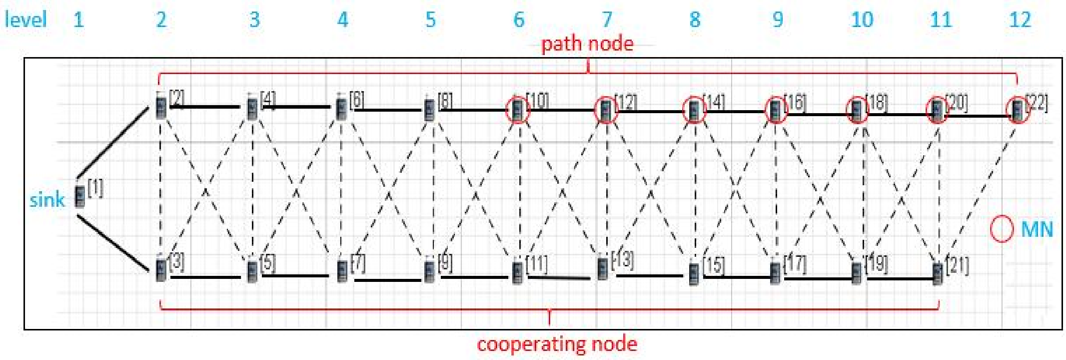
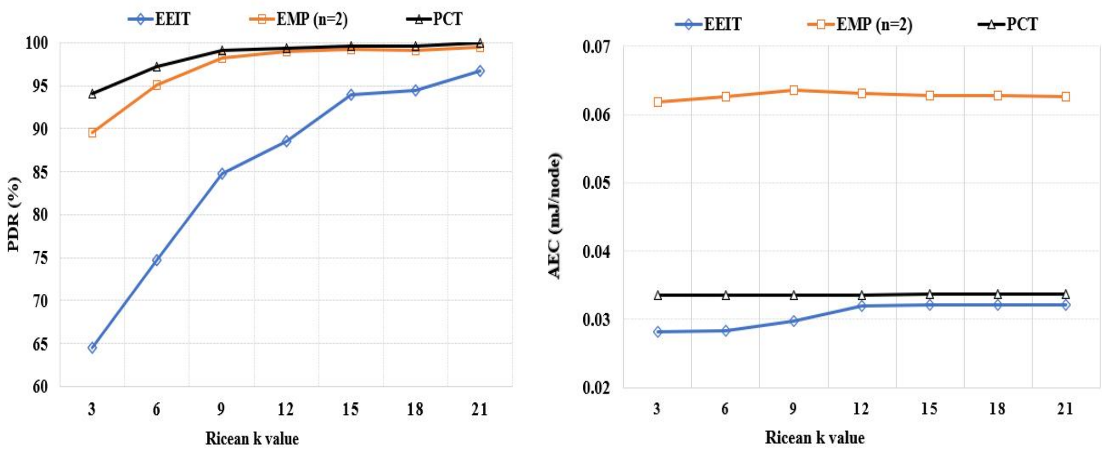


Figure 13. Scenario S2 with a rectangular dimension of 15 × 100 (m2) and a two-way chained deployment of 1 sink and 21 nodes

b. Sensitivity to Multipath Fading



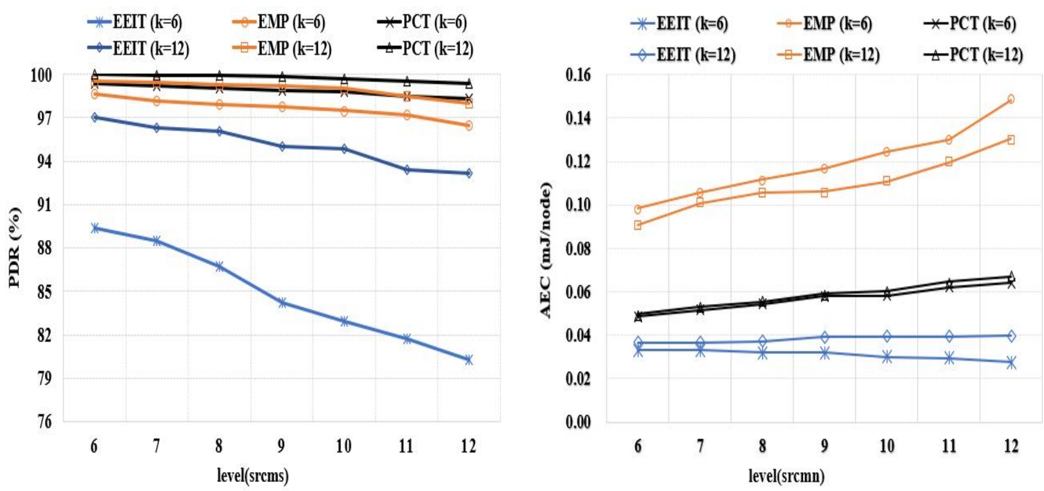
(a) Packet delivery ratio (b) Average energy consumption

Figure 14. Comparison of three protocols with varying k (scenario S1 and node 17 is the selected multimedia node)

In Figure 14, three protocols, EEIT, EMP(n=2) allowing a maximum of two transmissions per packet, and PCT, are compared in terms of packet delivery ratio and average energy consumption in Scenario S1. Both PCT and EMP showed excellent PDRs, since PCT uses a cooperative path, and EMP uses retransmissions for reliability, while PCT achieved a slightly higher PDR than EMP. However, PCT showed very low energy consumption, comparable to EEIT that allows only hop-by-hop retransmission on a single path. For the same reason, both PCT and EMP were less sensitive to the decrease in the *k* value while their PDRs showed the increasing gap. This implies that PCT is less sensitive to the high multipath fading effect. However, the PDR under EEIT decreased sharply as the *k* value decreased because it uses a single path and only hop-by-hop retransmission for reliability on the single path. On the other hand, this is evidence that path-wide retransmission under EMP greatly improves PDR.

Figure 14 (b) shows that retransmissions cause high consumption of energy. Meanwhile, it can be shown that the AEC under PCT is kept low, while it remains high under PDR. Note that AEC under EEIT decreases as the *k* value goes below 12. This is because more packets are lost with the increase in the multipath fading effect. From this discussion, we can conclude that PCT based on the cooperative path is a highly dependable image transmission protocol, since it can achieve high PDR comparable to EMP, and low energy consumption comparable to EEIT, and it is also less sensitive to the multipath fading effect.

c. Sensitivity to The Hop Distance of The Multimedia Node



(a) Packet delivery ratio (b) Average energy consumption

Figure 15. Comparison of different protocols according to the hop distance of the multimedia node (Scenario = S2, k = 6 or k= 12)

In order to see the sensitivity to the hop distance between the MN and the sink, the three protocols were compared in terms of packet delivery ratio and average energy consumption under Scenario S2. As shown in Figure 15 (a), both EMP and PCT maintained high PDRs, but were less sensitive to an increase in the hop distance of the MN to the sink, whereas EEIT showed high sensitivity to hop distance. Note that the gap under EEIT (k=6) and EEIT (k=12) at a hop distance of 11 reached about 13. This gap is almost twice as big as the gap (about 7) at a hop distance of 5. Furthermore, EMP is a little more sensitive than PCT to the increase in the hop distance, since path-wide retransmission will become less effective for a longer hop distance from the MN to the sink.

Figure 15 (b) shows that PCT consumes much less energy than EMP, since it does not allow retransmissions. Furthermore, EMP shows a constantly increasing pattern in AEC with increases in hop distance. This is because it uses path-wide retransmission. As expected, EEIT consistently showed the lowest AEC and had a slightly decreasing AEC pattern with increases in hop distance. This is because the lost packets do not incur energy consumption. Note that EEIT loses more packets based on the increase in hop distance, as shown in Figure 15 (a).

3.6.4 Experiment

a. Experiment Setup

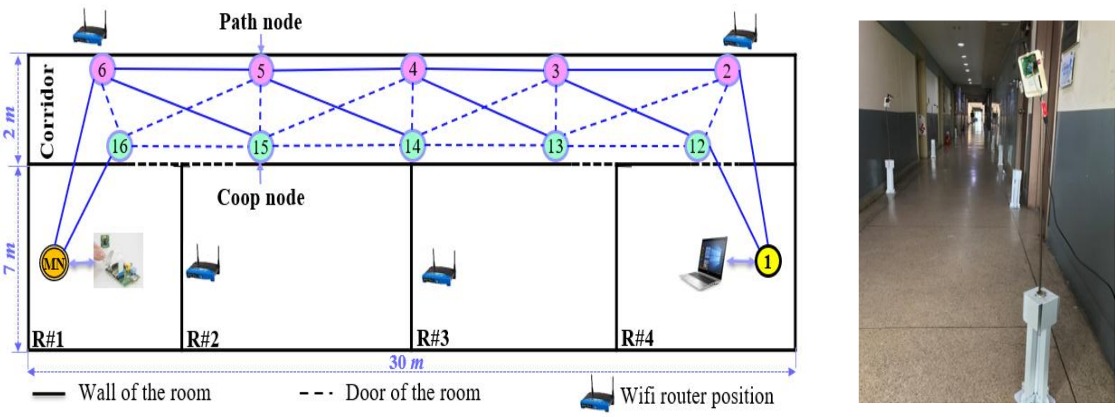


Figure 16. The testbed of WMSN that consists one sink, ten nodes, and one source multimedia node to cover the part of the corridor and four rooms.

A number of routing sensor nodes were manually deployed along the corridor of the computer engineering building in University of Ulsan, with a hop-distance of about 7 meters. An MN is located in room number 1, *R#1*, while a sink is located in *R#4* as illustrated in Figure 16. Note that two WiFi access points (APs) located along the corridor and two WiFi APs in R#2 and R#3 can interfere the transmission of image packets.

Three evaluation metrics were used. Firstly, the *packet delivery ratio* (*PDR*) indicates the ratio of the number of packets received at a sink to the total number of packets transmitted by the MN. Second, t*he image delivery ratio* (*IDR*) is the ratio of the number of lossless reconstructed images at the server to the total number of images transmitted by the MN. Note that IDR does not count any blurred images reconstructed with the loss of any packets since an object may not be identified from the blurred images. Third, the *end-to-end delay (E2ED)* of an image is the time span to deliver an image from the *SrcMN* to a sink. Note that the time to turn on the camera and capture an image is deliberately ignored since it varies largely according to the quality of a camera module and also it can be shortened if the additional sensor is used to detect an ambient event and turn on the camera autonomously. The time to compress a captured image at a multimedia node and reconstruct the image at a server are ignored since they are negligible compared to the delay of image delivery.

b. The Size of Compressed Images

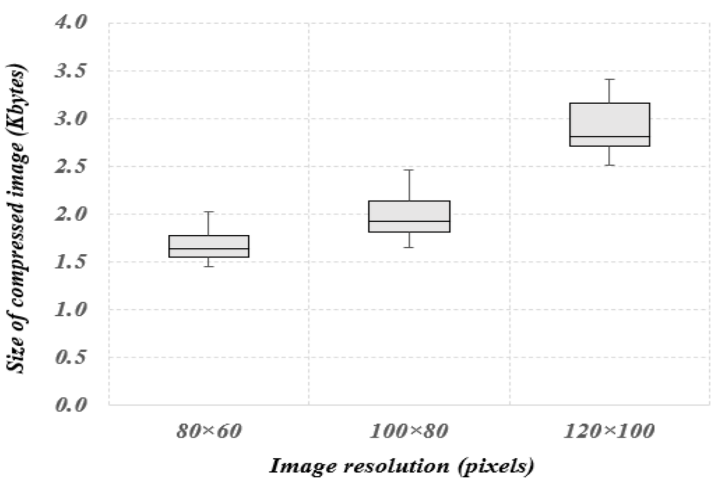


Figure 17. The sizes of the compressed images according to image resolutions with the use of the JPEG 2000 standard

The captured images are of an identical size; however, the compressed images are different in size. Thus, the ITP has to be adjusted according to the size of the compressed image. Figure 17 shows a variation in the sizes of compressed images with different resolutions. Note that the compressed images with the resolution of 100 × 80 pixels range in size between 1.65 and 2.5 Kbytes (indicated by a vertical line) and mostly between 1.8 Kbytes and 2.2 Kbytes (indicated by a box). The median value of the sizes is 1.93Kbytes (indicated by a horizontal line in the box).

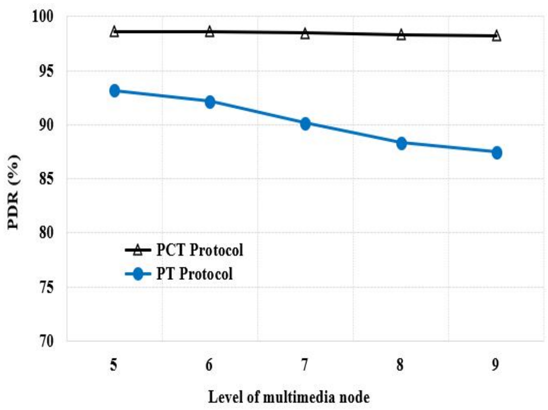
c. Experiment Results

Packet and Image Delivery Ratio

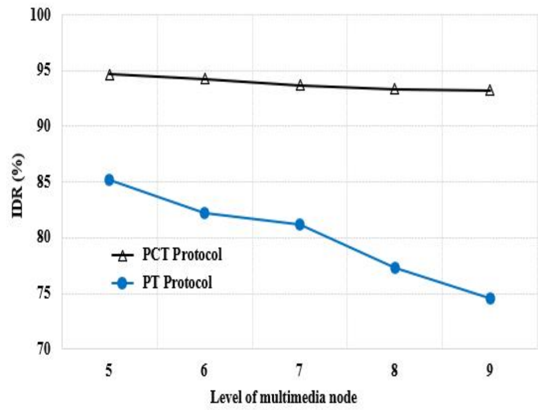
(a) Captured image | (b) Image without data loss | (c) Image with 4% data loss | (d) Image with 8% data loss | (e) Image with 12% data loss

Figure 18. The clearance of reconstructed images according to the loss rate of image data

Figure 18 shows the effect of data loss on the quality of a reconstructed image at a sink. The original image that is captured by CAM is shown in Figure 18 (a). The lossless reconstructed image is shown in Figure 18 (b), while the blurred images with some data loss, of 4%, 8%, and 12% are shown in Figure 18 (c), 18 (d), and 18 (e), respectively. As the rate of data loss increases, the images get blurred more and more. It is known that the last image with 12% data loss is considerably destroyed in its lower part so that the trademark of the shirt cannot be identified.



(a) Paket delivery ratio



(b) Image delivery ratio

Figure 19. Packet and Image delivery ratio according to the hop distance of the multimedia node

Figure 19 shows the PDR and IDR from the experiments in the testbed. In Figure 19 (a), we compare the PDRs of two approaches: The PCT protocol that uses a cooperative path and the PT protocol that uses pipelined transmission along a simple path by allowing the transmission of a packet every transmission interval (*TxInt*). As the hop distance of MN increases, the performance gap between PCT and PT increases gradually. It is shown that when MN is located at the tree level of 9, the PDR of PCT improves that of PT by 12%. Figure 19 (b) compares the IDRs of the two approaches. In this comparison, the image with the loss of a single packet is discarded without counting. The MN sends one image to a sink every 6 *seconds* and 300 images totally. According to experiments, a sink with the PCT protocol could receive 94% of images without losing any data. On the contrary, the IDR of PT decreases sharply as the tree level of the MN increases, down to 75% at the tree level of 9.

From the experiments, it can be concluded that the PCT protocol is highly reliable in delivering images even in a harsh environment with four heavily interfering WiFi APs.

End-to-End Delay

In this section, the E2ED of image transmission is examined for the PCT protocol and the CT protocol that uses a cooperative path but does not use a pipelined transmission. Note that the E2ED depends on the number of image packets generated from a compressed image. The original images of 34 Kbytes with a resolution of 100×80 pixels were compressed to the images of varying sizes between 1.65 and 2.5 Kbytes that correspond to 17 packets and 25 packets of 100 bytes, respectively.

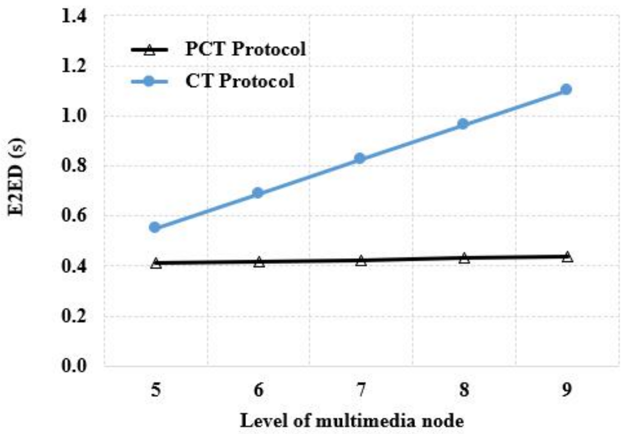


Figure 20. End-to-end delay of image transmission.

Figure 20 compares the E2ED of the PCT protocol and that of the CT protocol. PCT could achieve 0.4 seconds to transmit each image on average and slightly increasing times as the level of MN increases. This is the reason that the pipelined transmission just increases one hop transmission time with the increase of one level. Note that the latency by one hop is just 6 milliseconds. So, the graph pattern almost looks like a flat. On the contrary, CT increases E2ED linearly to the number of hops since the interval of two consecutive data transmissions increases.

Chapter 4

LoRa-WSN HYBRID NETWORK PROTOCOL

4.1 Network Model.

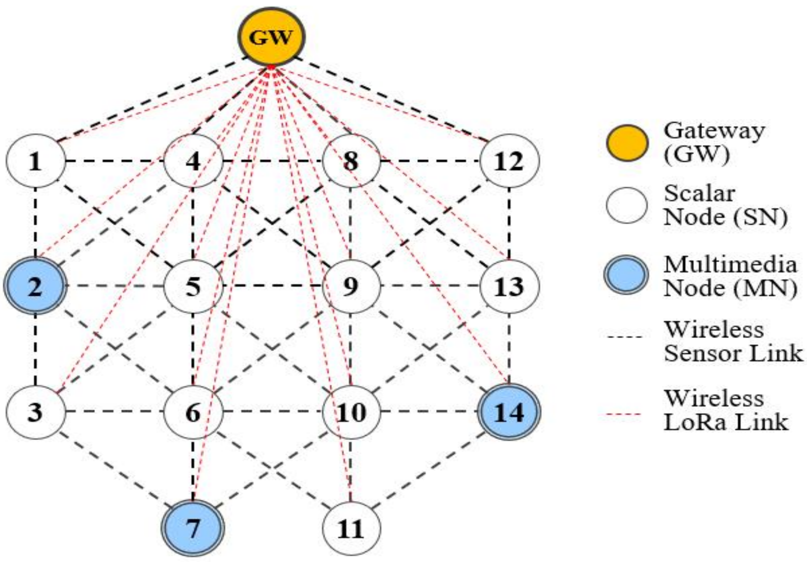


Figure 21. LoRa-WSN hybrid network model

The considered LoRa-WSN hybrid network consists of a gateway or a server (*GW*), a number of *scalar nodes* (*SNs*), and some *multimedia nodes* (*MNs*). The sink is wall-powered, whereas the SNs and MNs are battery-powered. A node can transmit its data to a sink in one hop using a LoRa communication module (LCM) referred to as a *wireless LoRa link,* or in multiple hops using a sensor communication module (SCM) referred to as a *wireless sensor link*. Every node is required to transmit scalar data to a server periodically. When a server judges that an abnormal event has occurred based on analyzing the collected scalar data, it will request the transmission of voice or image from a source *MN* (*SrcMN*) in the spot.

Figure 21 shows a LoRa-WSN hybrid network, which consists of one *gateway*, 11

SNs, and 3 *MNs*. The black dashed lines and the red dashed lines represent the *wireless sensor links* and the *wireless LoRa links*, respectively.

4.2 Motivation

In wireless data collection systems, scalar data and multimedia data such as image or voice are delivered from nodes to a server via a wireless network. We consider the disadvantages of using a WSN or a LoRa network to deliver both the data simultaneously.

The use of a WSN to deliver both the data has four problems as follows. The f*irst* is the overhead problem by flooding routing messages to establish a short and reliable path for data transmission and by broadcasting command messages to control the network operation, such as the broadcast of an *image request message* (*IREQ*) to request an image from the *SrcMN*. The *second* is the inefficient channel utilization by using of control messages such as RTS, CTS, and ACK to secure the reliability of data transmission and by pausing momentarily the transmission of scalar data in multimedia transmission time to avoid internal interference and collisions for the delivery of an urgent image or voice. The *third* is the requirement of precise time synchronization for the operation of network. The requirement is more complicated in a WSN because it includes one hundred or even a thousand nodes and uses multiple hop transmission. The *fourth* is the difficulties in managing the network operation once the network is deployed, such as a node must know that it will take part in image transmission or not and when to wake up after sleeping. Additionally, a conflict-free schedule for network operation within a time constraint is also a challenge for the WSNs.

There are three disadvantages when using a LoRa network to deliver large amount of data as follows. *First*, the data transmission rate supported by LoRa can achieve between 0.3 kbps and 50 kbps depending on the settings of spreading factor (SF) and channel bandwidth (BW) [44], leading to a long time on air (ToA) and the high latency of data transmission. For example, the maximum of data transmission rate of LoRaWAN is 27 kbps when SF = 7 and BW = 500 kHz [45]. *Second*, a LoRa network faces with high possibility of packet loss because of the long ToA such as the average of packet loss when sending the packets with a payload of 55 bytes between two LoRa nodes on different floors is about 20% [46]. *Third*, the energy consumption of a LoRa network is higher than it of a sensor network. In [47], authors showed that when delivering of packets over a distance of one kilometer by using LoRaWAN and ZigBee, the energy consumption of LoRaWAN is 2.94 times of ZigBee's. Thus, the use of a LoRa network to deliver large amount of data such as an image or voice in a real-time and reliable manner needs more attention from researchers.

This chapter presents a simple LoRa-WSN hybrid network protocol to transmit both scalar and multimedia data efficiently, in which LoRa network is used to deliver control messages and scalar data at regular intervals, whereas WSN is used dedicatedly to deliver multimedia data from a *SrcMN* to a sink in an event-driven manner. This approach utilizes the long-range transmission of the LoRa network to deliver control messages directly from a sink to nodes. As a result, the LoRa-WSN hybrid network can decrease the overhead problem caused by flooding control messages in multi-hops. The approach also utilizes the high data rate and high reliability of a WSN for the transmission of an urgent image or voice. Moreover, the operation of sensor nodes is simplified because it is triggered and controlled by the LoRa network, and it does not require time synchronization.

The operation of the network in Figure 21 that employs the LoRa-WSN hybrid protocol is as follows. Every node sends its scalar data periodically to a sink via the *wireless LoRa links.* Based on analyzing the collected scalar data, if the server judges that an abnormal event occurs it will require the transmission of an image or voice from a *SrcMN* such as node 7 by broadcasting a command message (*CM*) via the *wireless LoRa links*. Upon receiving the *CM*, the *SrcMN* will transmit the packets of an image or voice to a server along a data path with *wireless sensor links* such as the *data path* 7 → 6 → 5 → 4 → Sink. When a sink realizes that there are lost packets, it will broadcast a retransmission request (*RR*) message to the *SrcMN* via wireless LoRa links. Upon receiving the RR, the *SrcMN* retransmits the lost packets along the data path.

The advantages of using the LoRa-WSN hybrid network protocol to deliver both scalar and multimedia data are five-fold:

(1) Shorten the time it takes for an image or voice to be delivered by removing control messages and quickly activating the multimedia delivery.

(2) Simplify the management of the network by using one-hop transmission for control messages.

(3) Improve the reliability of data transmission by using path-wide retransmission for multimedia data transmission and using a dedicated time slot for scalar data transmission.

(4) Increase network traffic considerably by allowing simultaneous transmission of both scalar and multimedia data.

(5) Mitigate the traffic congestion caused by the high bandwidth demand of data transmission.

4.3 Notations and Messages

Several terms and notations used in this chapter are defined as Table 5.

Table 5. Terms and terminologies

TxInt	The transmission interval between two continuous image packets
LCM	The LoRa communication module
SCM	The sensor communication module
comCh	The common channel of LoRa communication module
reCh	The channel to broadcast *RR* message
N_m	The number of missing packets at the gateway
MaxNumReTx	The maximum number of retransmissions
t(RREQ)	The time span to transmit *RREQ* message from gateway to the *SrcMN*

Three messages used to control image transmission are defined as follows:

- *CMD*: The gateway sends a command message to nodes using *wireless LoRa links*: *CMD* = (*reSch*, *TPI*, *SrcMN*, *mFlag*, *reCh*, [*pathReSetup*]) where if *reSch* = 1, nodes are required to reschedule slots for scalar data transmission using task profile information *TPI, SrcMN* is source MN address, *mFlag* = 0 for image and *mFlag* = 1 for voice, *reCh* is channel to deliver retransmission request message, if *pathReSetup* = 0, *SrcMN* sends voice/image immediately.
- *RR*: The gateway sends a retransmission request message $RR = (N_m, (m_1, ..., m_k))$ to the *SrcMN* where N_m indicates number of lost packets and $m_1, ..., m_k$ are sequence of lost packets.
- *RREQ*: Gateway sends a routing request message (*RREQ*) to construct a new data path for multimedia transmission.

4.4 Protocol Design

4.4.1 Protocol Structure

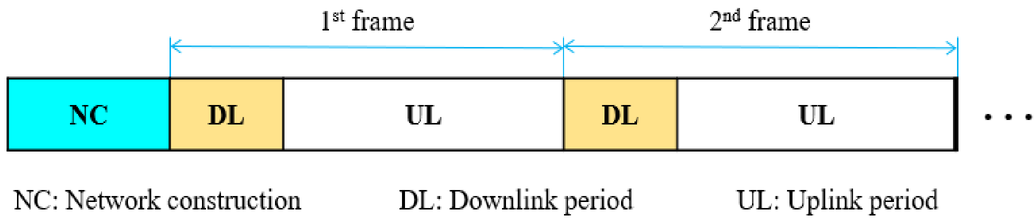


Figure 22. Protocol Structure

As shown in Figure 22, the protocol starts with the *network construction* (NC) period during which a one-hop LoRa network topology and a multi-hop WSN topology are constructed and then repeats a frame that consists of a downlink (DL) period and an uplink (UL) period. The UL period is divided into a number of data transmission slots, 2^N, where N is a frame factor ($N \geq 0$). During the UL period, scalar data is transmitted from all the nodes periodically, and a still image is transmitted from the *SrcMN* in an event-driven manner.

During the DL period, GW sends a command message, CMD, on the common channel to control the transmission of scalar and multimedia data. Upon receiving CMD, a node schedules transmission slots using task profile information (*TPI*) for scalar data transmission and wakes sensor communication module for multimedia transmission. GW can broadcast a routing request message (*RREQ*) to establish a new data path for multimedia transmission. During the UL period, scalar data and multimedia data are transmitted to GW using the LoRa communication module (*LCM*) and sensor communication module (*SCM*), respectively, as shown in Figure 23. The following section will clearly present the transmission of both scalar and multimedia data.

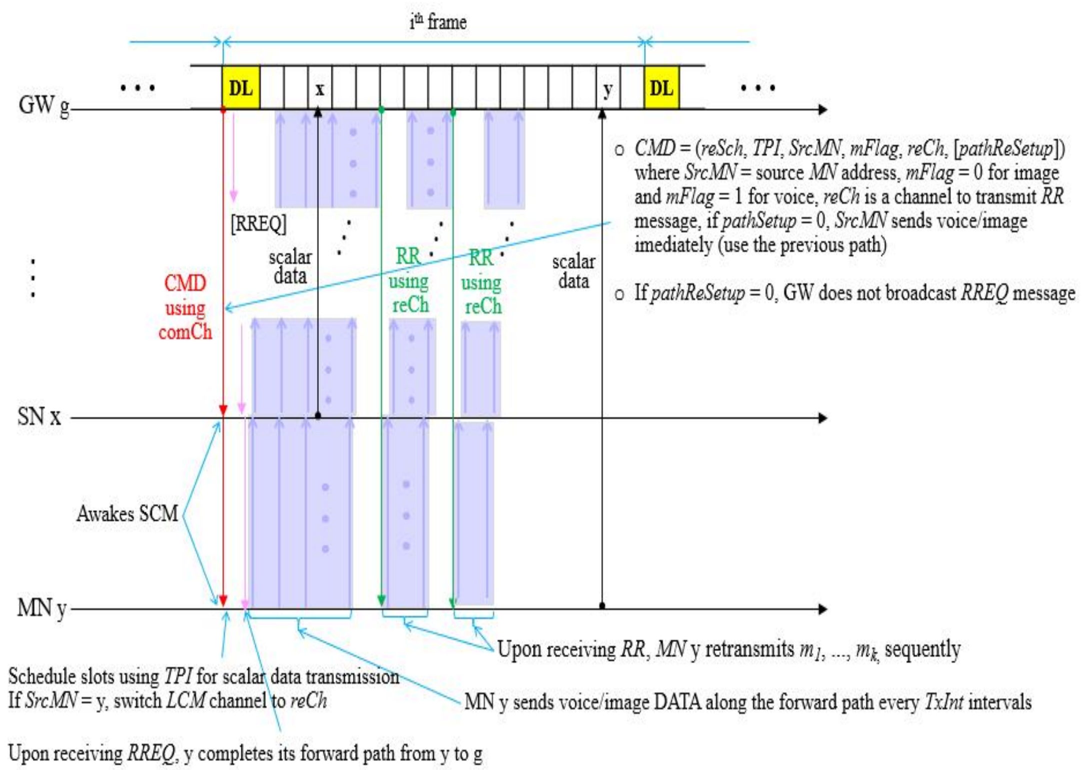


Figure 23. The operation of hybrid LoRa-WSN protocol

4.4.2 Command and Scalar Data Transmission

a. Slot Scheduling

During UL period, nodes transmit their sensory data to the GW by using wireless LoRa links. Assumption that, all participating nodes can connect to the GW directly. Each node is modelled as a *task* as an active entity. Each task is required to send one packet periodically to the GW. We utilized a real-time LoRa protocol [48] that based on a frame-slot architecture for scalar data transmission.

For a frame of 2^N slots, a task of node *x*, denoted by τ_x, is characterized by its *class* *c* as $\tau_x = (c)$ such that its *transmission interval*, $TI(\tau_x)$, is defined as follows:

$$TI(\tau_x) = 2^N / 2^c \tag{25}$$

this indicates that τ_x, is required to transmit 2^c packets during one frame period.

Given a set of tasks, the schedule to satisfy the data transmission requirements for each task in the set is generated by a task scheduling algorithm. Task slot scheduling relies on the *logical slot indexing* (LSI) algorithm [48] that assigns a logical slot index to each of 2^N UL slots such that if task τ of class c is assigned 2^c slots sequentially starting with any logical slot index and transmits data in each assigned slot, it can satisfy its time constraints in data transmission. The example of the logical slot indices assigned by the LSI algorithm for the UL period of 16 slots is illustrated in Figure 24.

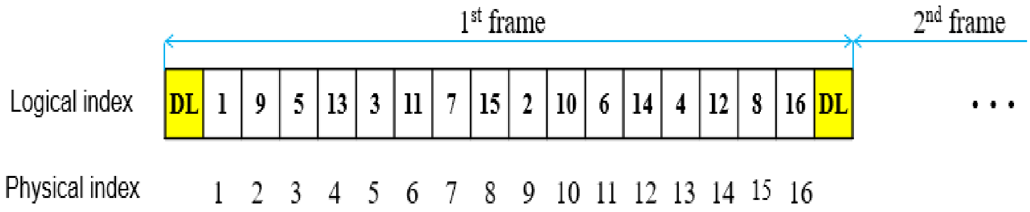


Figure 24. An example of logical slot indexing with a frame of 16 slots

For convenience, let us define the *slot demand* (*SD*) of *node i*, denoted as *SD(i)*, as the number of slots that it requires within one frame. Then, *SD*(*i*) is expressed as follows:

$$SD(i) = 2^{class(i)} \tag{26}$$

where *class*(*i*) indicates the class of node *i*. Every node *i* is required to register its *task profile information*, *TPI*(*i*), with GW as follows:

$$TPI(i) = (i, class(i)) \tag{27}$$

Then, a GW manages the task profile information of network (TPI) for the participating nodes as follows:

$$TPI = (TPI(1),\ TPI(2), TPI(3), \ldots, TPI(n)) \tag{28}$$

where *n* is the number of nodes in the network.

During *DL* period, a GW broadcasts CMD message that contains *TPI* to every node.

Upon receiving *TPI*, a node can determine the total slot demands of all preceding nodes in *TPI* and can calculate the *start of the logical slot index* (*startLSI*). Then, it can determine its *slot allocation* (*SlotAll*) as a set of assigned logical slot indices for itself. Based on its *SlotAll* and the given slotted frame, the node can determine its *transmitting slots* (*TxSlots*) to transmit its sensory data. For detail, readers are referred to the [48].

Let us give an example for task scheduling for a simple network including node *x* and node *y*, and both having class 1. Then, *TPI* (*x*) = (*x*, 1) and *TPI* (*y*) = (*y*, 1) and *SD* (*x*) = *SD* (*y*) = 2. Suppose that the logical slot indices from 1 to 4 have been scheduled for other tasks. Thus, node *x* has *startLSI* = 5 and *SlotAll*(*x*) = (5, 6) that correspond to the physical indices (3, 11). Node y has *startLSI* = 7 and *SlotAll* (*y*) = (7, 8) that correspond to the physical indices (7, 15).

b. Command and Scalar Data Transmission

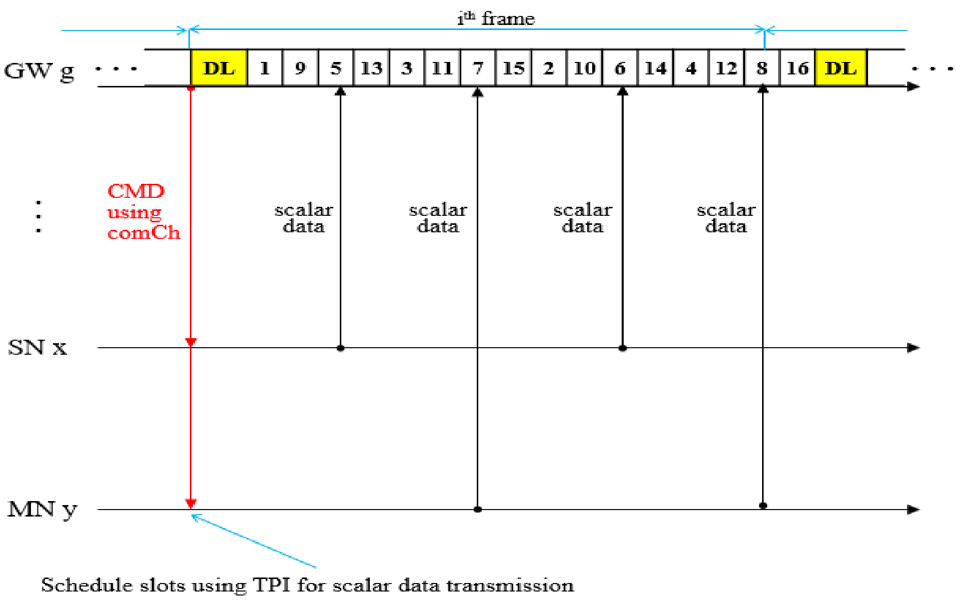


Figure 25. Command and scalar data transmission

GW can send a command message, *CMD = (reSch, TPI, SrcMN, mFlag, reCh, [pathReSetup])*, to end nodes during the *DL* period on the common channel (*comCh*) using *LCM*. Upon receiving IREQ, every SN transmits its sensory data at assigned slots.

If *reSch* of one, nodes must reschedule slots using the task profile information, *TPI,* as mentioned above. The *SrcMN* can capture a still image or record a voice file, depending on the value of the multimedia flag (*mFlag*). The study of this thesis concentrates on the requirement of an image from the *SrcMN*. The transmission of image data from the *SrcMn* to a GW using multiple hops via *wireless sensor links*.

Every node sends data to GW in each of the physical slots that correspond to the assigned logical slot indices. For example, in Figure 25, sensor node (SN) *x* sends data in the physical slots of 3 and 11 that correspond to the logical slot indices 5 and 6, and multimedia node (MN) *y* sends data in the physical slots of 7 and 15 that correspond to the logical slot indices 7 and 8.

4.4.3 Image Transmission

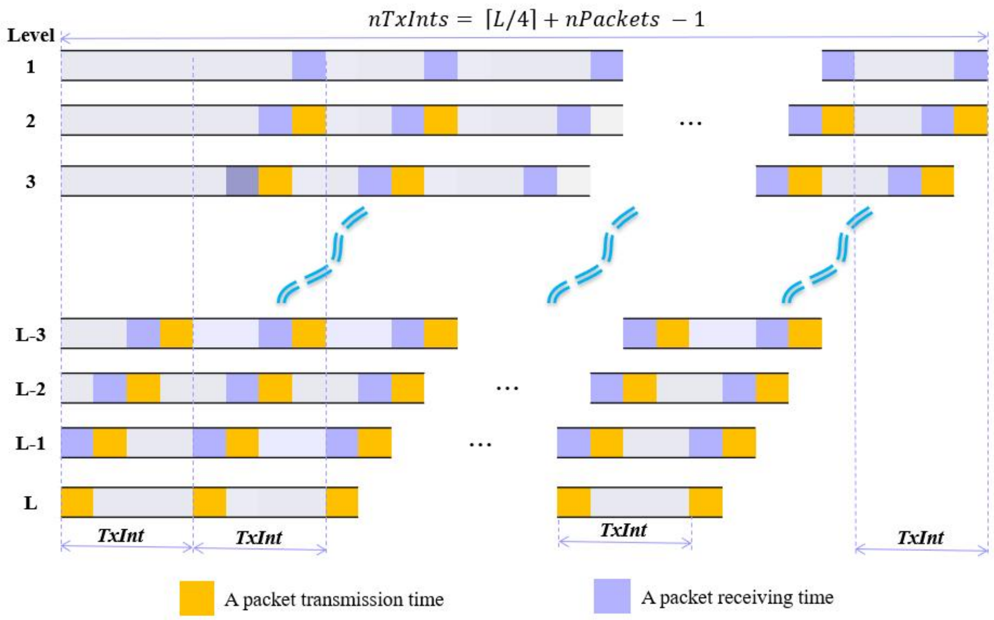


Figure 26. The transmission of image packets from a SrcMN at level L to GW

If GW requires an image from a specific multimedia node, say *SrcMN* at level *L*, it attaches the address of *SrcMN* in the CMD message. Upon receiving CMD, a node awakes its SCM. When *pathReSetup* is set to one, the GW sends a *RREQ* message on

SCM to establish a new path for image transmission. Upon receiving RREQ, the *SrcMN* completes its forward path to GW before sending image packets. When *pathReSetup* is set to zero, the *SrcMN* will capture an image and send it immediately using the previous path. Furthermore, it can determine the channel that GW uses to send RR messages and it switches LCM channel to *reCh* to receive them.

A captured image will be delivered from the *SrcMN* to GW via wireless sensor links in multiple hops. A captured image needs to be compressed and divided into a number of packets as described in Chapter 3. Since the interference range is approximately twice as long as the transmission range [21], two nodes on the data path that are four hops away, can safely transmit data using the same channel. It means the *SrcMN* can send an image packet in every transmission interval (*TxInt*) that equals four times the transmission time of a packet. Suppose that L = *level*(*SrcMN*) and *nPackets* is the number of packets corresponding to one image that *SrcMN* has to transmit. The number of transmission intervals *(nTxInts)* required to deliver *nPackets* can be calculated as follows:

$$nTxInts = \left\lceil \frac{L}{4} \right\rceil + nPackets - 1 \tag{29}$$

Path nodes will keep SCM in receive modes in image transmission time and forward the received packet immediately. Figure 26 illustrates the transmission of an image divided to *nPackets* from a *SrcMN* at level L to GW.

Since nodes use SCM purely for the transmission of image packets (*IPackets*) and a node does not compete with other nodes, control messages such as RTS and CTS are removed. This approach not only reduces the one-hop transmission time of a packet, but also utilizes the high data rate and high reliability of a WSN for the transmission of an image in an urgent case. Furthermore, path-wide retransmission is employed to meet the requirement for high reliability in data transmission. By using *LCM* to deliver RR message from GW to *SrcMN*, we can simplify the control of path-wide retransmission.

Based on the transmission interval and number of packets required to be transmitted from *SrcMN* to GW, an image timer (*ITimer*) is calculated to determine the time point of transmitting the RR message. Moreover, we use the maximum number of path-wide retransmissions (*MaxNumReTx*) to balance energy consumption and reliability of packet transmission. The LoRa-WSN hybrid network protocol for image transmission is detailed in Algorithm 3.

Algorithm 3. The LoRa-WSN hybrid transmission protocol

Gateway (*GW*):

After sending *CMD* on *comCh*
  $k = 0$;
  If (*pathReSetup* = 1) **then**
    Broadcast *RREQ* message;
    Set *ITimer* = *TxInt* ×($\lceil L/4 \rceil$+*nPacket* – *1)* + t(*RREQ*);
  else
    Set *ITimer* = *TxInt* ×($\lceil L/4 \rceil$+*nPacket* - *1);*
  endif
When *ITimer* expires:
  if ((k < *MaxNumReTx*) *and* (N_m!= 0)) ***then***
    Construct RR message;
    Broadcast RR message on *reCh*;
    $k = k + 1$;
    set *ITimer* = *TxInt* × ($\lceil L/4 \rceil$+N_m - *1);*
  else
    Transmit received data to a server;
  endif

Sensor node (SN)

Upon receiving *CMD*:
  Schedule slots using *TPI* in *CMD*;
  Awake SCM;
Upon receiving *Packet*:
  Forward the received *Packet*;
  Keep SCM in received mode
At scheduled slot:
  Transmit scalar data using *LCM*

Source multimedia node (SrcMN)

Upon receiving *CMD*:
  Schedule slots using *TPI* in *CMD;*
  Awake *SCM*;
  Switch LCM channel to *reCh*;
Upon receiving *CMD*
  if *(pathReSetup* = 1) **then**
    Wait *RREQ* message.
    Establish data path from *SrcMN* to GW;
    Transmit a *Packet* in every *TxInt*;
  else
    Transmit a *Packet* in every *TxInt*;
  endif
Upon receiving *RR*:
  Retransmit $m_1, ..., m_k$ sequentially.
At scheduled slot:
  Transmit scalar data using *LCM*;

4.5 Performance Evaluation

4.5.1 Comparison of Features from Recent Image Protocols

The features of the proposed LoRa-WSN hybrid protocol are contrasted with those of other three protocols, EMP, PCT, and LoRaCP on three performance metrics: reliability, energy consumption, and end-to-end delay.

EMP can improve the reliability of packet transmission by using not only retransmission (hop-by-hop and path-wide retransmissions) but also channel hopping. However, if the path is not stable, EMP can suffer from the loss of packets. On the contrary, PCT can improve the reliability of packet transmission by using a cooperating node to salvage a lost packet without retransmission and by using slot scheduling to reduce collision. LoRaCP tries to get a high packet delivery ratio by updating link quality to establish an optimal routing path. In comparison to EMP, PCT, and LoRaCP, LoRa-WSN hybrid protocol can establish a new path for reliable data transmission if necessary, and it also uses path-wide retransmission flexibly to meet the requirement of high reliability in packet transmission.

EMP can balance energy consumption by distributing packets on multiple paths; however, the use of hop-by-hop or path-wide retransmission will increase the energy consumption of nodes. Meanwhile, PCT saves energy by using slot scheduling to manage the operation of nodes on cooperative path and by disallowing retransmissions. LoRaCP reduces energy consumption by managing the activity time of the radio strictly. LoRa-WSN hybrid protocol reduces energy consumption by removing control messages such as RTS and CTS, and it eliminates the overhead problem by utilizing the LoRa network to deliver RR messages.

Table 6. The features of image transmission protocols

<table>
<tr><th>Features</th><th>EMP</th><th>PCT</th><th>LoRaCP</th><th>Hybrid Protocol</th></tr>
<tr><td rowspan="2">Approach</td><td>- Builds a separated multiple paths from a MN to a gateway
- Allows parallel transmission by using time slots, channel hopping and multiple paths</td><td>- Establishes a cooperative path included path nodes and cooperating nodes for reliable data transmission.
- Assigns time slots and channels to a tree level to enable pipelined transmission</td><td>- Uses LoRa network to transmit control messages that include information about the link quality to a server and routing message from server to nodes
- Uses Zigbee based Kmote to form data plane</td><td>- Uses LoRa network to deliver control messages and scalar data in a real-time and reliable manner by using slot scheduling
- Dedicates SCM for the transmission of an image</td></tr>
<tr><td colspan="2">- Does not allow the transmission of scalar data in multimedia data transmission time and requires time synchronization for slot scheduling</td><td colspan="2">- Separates the control plane from data plane for network operation</td></tr>
<tr><td>Reliability</td><td>- Uses hop-by-hop retransmission relying on ACK message and path-wide retransmission by flooding RR messages over WSNs
- Tries to remove collision using channel hopping</td><td>- Uses the SSMAb protocol for command delivery over WSNs
- Avoids collisions using slot scheduling and channel-assisted slot reuse technique
- Uses a cooperating node to salvage a lost packet without retransmission</td><td>- Allocates time slots and channels in a round-robin fashion to form high reliability one-hop control plane
- Uses an optimal routing path for reliable data transmission of the Zigbee network</td><td>- Uses LCM to broadcast control messages.
- Uses path-wide retransmission and RR messages are delivered via LCM
- Establishes a new path for reliable data transmission if necessary</td></tr>
<tr><td>Energy consumption</td><td>- Disables the nodes not on multiple paths and balances energy consumption by distributing packets on different paths
- Uses channel hopping to reduce the number of path-wide and hop-by-hop retransmissions</td><td>- Uses slot scheduling to manage the operation of nodes on cooperative path.
- Does not use retransmissions</td><td>- Reduces overhead problem and collision by allocating a distinct slot and channel to a LoRa node.
- Minimizes the activity time of the radio for data transmission, so as to reduce energy consumption.</td><td>- Removes control messages such as RTS and CTS and eliminates overhead problem by utilizing LoRa network
- Reduces number of nodes on data path and uses SCM only for image transmission</td></tr>
<tr><td>End-to-end delay</td><td>- Uses pure one-slot pipelined transmission
- Can have high the number of retransmissions, relying on link quality</td><td>- Uses two-slot pipelined transmission
- Does not allow hop-by-hop and path-wide retransmissions</td><td>- Uses time slots and carrier-sense multiple access with collision avoidance (CSM/CA)</td><td>- Transmits a new packet in every TxInt
- Optimizes one-hop delivery time of a packet by removing control messages
- Achieves real time in scalar data transmission</td></tr>
</table>

In EMP, a MN can transmit a packet in every slot continuously using different disjoint paths and pipelined transmission, while in PCT, it can transmit packets in every other slot using channel-assisted and spatial slot reuse techniques. However, EMP can have a high number of retransmissions since multiple paths are independent, and an individual path may not be stable due to multipath fading effect and/or internal and external interferences. PCT uses a cooperative path on which a cooperative node can salvage a packet instantly by judging whether a path node transmits a packet or not. Since PCT removes the possibility of collision, it does not allow a path node to retransmit the same packet. Therefore, it can be expected that PCT can reduce image transmission time compared to EMP. Meanwhile, in LoRa-WSN hybrid protocol, scalar data is delivered in a real-time manner by using the LSI algorithm, and end-to-end delay of an image can be reduced by optimizing the one-hop delivery time of a packet and by removing control messages. The discussion on the features of image transmission protocols is presented in Table 6.

4.5.2 Analysis of End-to-End Delay

In this section, we calculate the lower bound of the end-to-end delay to transmit one image from the *SrcMN* at level L to a sink under the three image transmission protocols: EMP, PCT and LoRa-WSN hybrid protocol. The E2ED(EMP) and E2ED(PCT) are taken from the calculation in Subsection 3.6.2 as follows:

$$E2ED(EMP) = 300 + \frac{2 \times 1140.48}{(1-(1-q)^2)^{L-1}} \quad (ms)$$

$$\begin{aligned} E2ED(PCT) &= 20.99 \times (L-1) + (2 \times nPackets + L - 3) \times 5.504 \\ &= 26.467 \times L + 2384.285 \quad (ms) \end{aligned}$$

Meanwhile, LoRa-WSN hybrid protocol works as follows. First, GW broadcasts a command message to inform all nodes of the start of image transmission using LoRa communication. This delay is denoted by T_{LoRa}. Then, the SrcMN transmits packets to the GW along the path, denoted as *ITP*(*Hybrid*). Finally, the GW transmits a RR message to inform lost packets and the SrcMN retransmits lost packets. Thus, E2ED(Hybrid) is given as follows:

$$E2ED(Hybrid) = 2 \times T_{LoRa} + ITP(Hybrid) \tag{30}$$

Suppose that LoRa uses SF7 and 250 kHz of bandwidth to deliver a command message of 10 bytes. T_{LoRa} is 30.8 milliseconds. The SrcMN transmits a new packet in every *TxInt*, *ITP*(*Hybrid*) is expressed as follows:

$$ITP(Hybrid) = (\left\lceil \frac{L}{4} \right\rceil + nPackets - 1) \times TxInt \tag{31}$$

where $TxInt$ is four times of $msgTxTime$ that is calculated by using the Eq. (12). With a payload of 110 bytes, $msgTxTime(x) = 4.352\ ms.$

E2ED$(Hybrid)$ is calculated as follows:

$$E2ED(Hybrid) = \ 2 \times 30.8 + (\left\lceil \frac{L}{4} \right\rceil + 219) \times 17.4 = 17.4 \left\lceil \frac{L}{4} \right\rceil + 3872.2 \quad (ms) \tag{32}$$

Figure 27 compares the lower bound of the end-to-end delay of three protocols: EMP, PCT, and LoRa-WSN hybrid protocol with the increase in the level of *SrcMN*, *level*(*SrcMN*). E2ED(EMP) is calculated in two different cases of link quality, $q = 0.6$ and $q = 1$. It is shown that, E2ED(EMP) in the case of low link quality, $\mathrm{q} = 0.6$, is highly sensitive to the increase in the *level*(*SrcMN*). Whereas E2ED(PCT), E2ED(Hybrid), and E2ED(EMP) with the best link quality, $q = 1$, are nearly constant with the increase of *level*(*SrcMN*). E2ED(PCT) and E2ED(EMP) are smaller than E2ED(Hybrid) because PCT and EMP allow pipelined transmission.

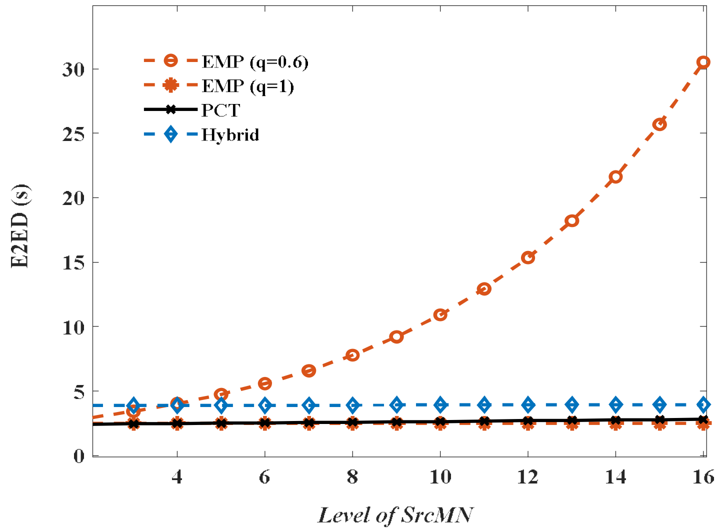


Figure 27. The lower bound of the end-to-end transmission delay of one image

4.5.3 Discussion of Simulation Results

LoRa-WSN hybrid protocol was evaluated and compared with EMP and PCT using the QualNet simulator with scenario of a rectangular dimensions of 15 × 100 m^2 with a GW and 19 nodes, as in Figure 28. The Ricean model for a multipath fading effect was also used with varying Ricean *k* factor. Two metrics were used. One is *energy consumption* (*EC*), which is the energy consumed by the network to deliver an image from the *SrcMN* to the GW, and another is the *packet delivery ratio* (*PDR*).

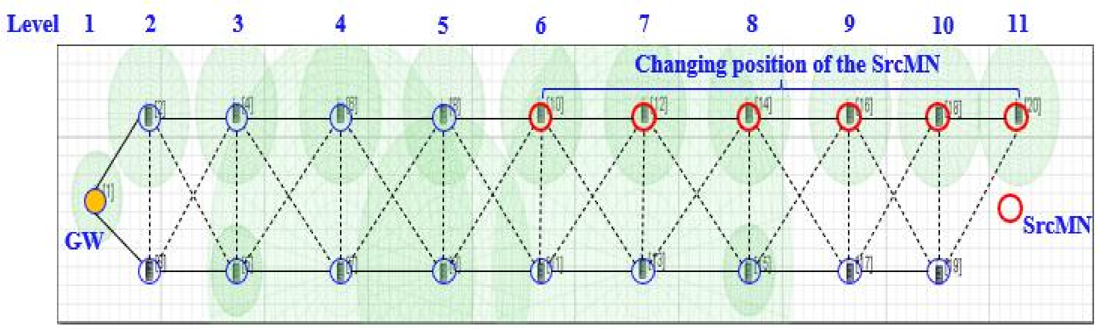


Figure 28. A scenario of a rectangular dimension of 15 × 100 (m^2) with 1 gateway and 19 nodes in QualNet simulator

a. Sensitivity to Number of Retransmission

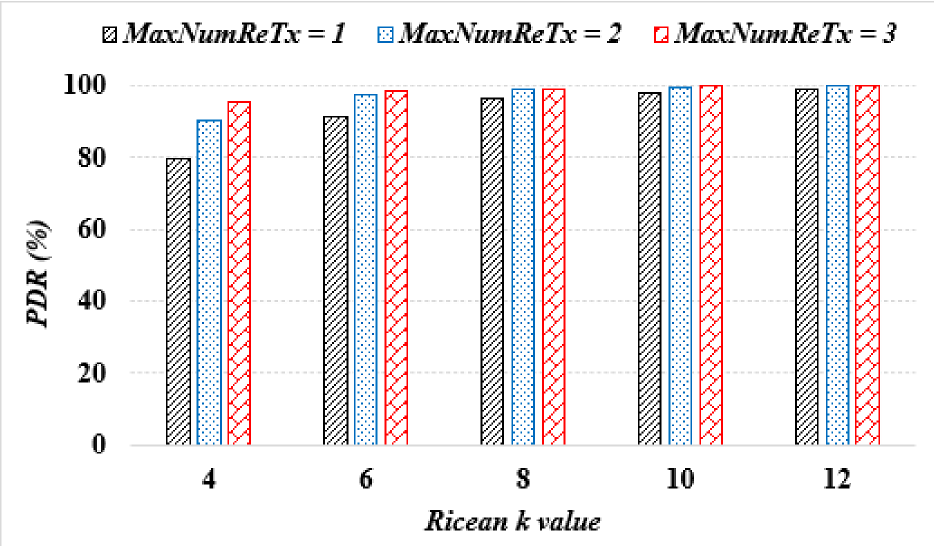
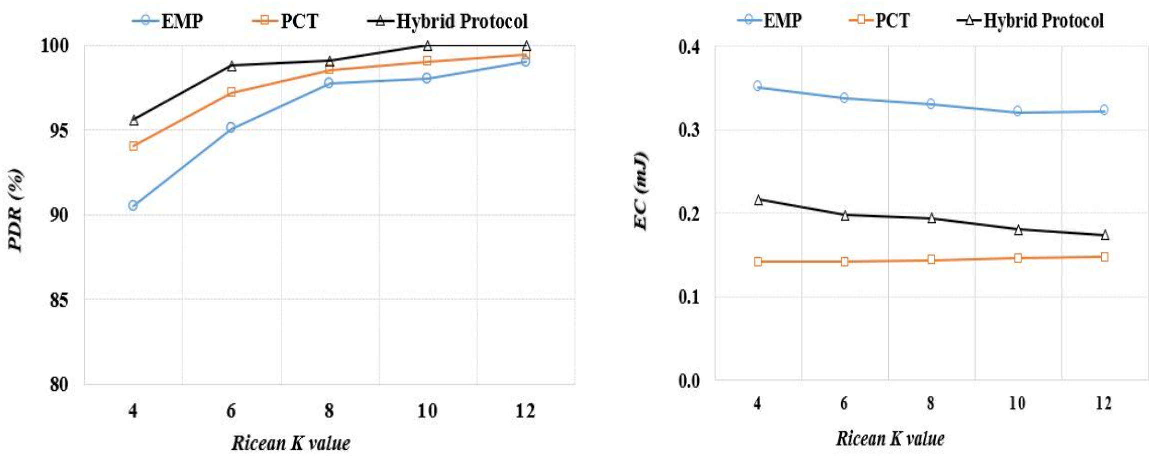


Figure 29. The PDRs of hybrid protocol *with different number of retransmissions (node 14 is the selected multimedia node)*

In Figure 29, the PDRs of the LoRa-WSN hybrid protocol with different maximum number of path-wide retransmissions, *MaxNumReTx*, are examined by changing the fading *k* value from 4 to 12. It is clearly shown that the proposed protocol achieves higher PDRs as the fading *k* value goes up, because fewer packets are lost with the decrease in the multipath fading effect. In the case of *MaxNumReTx* of three*,* the proposed protocol maintained a higher PDR throughout *k* factors, and the PDR was always over 96% at the level of *SrcMn* of eight. Furthermore, the protocol achieved a high PDR of more than 97% regardless of the value of *MaxNumReTx* in the environment with a normal level of multipath fading effect (*k* of around 10). Those results proved that path-wide retransmission under the LoRa-WSN hybrid protocol greatly improves PDR. The protocol with a *MaxNumReTx* of three can achieve a high PDR in a high multipath fading environment, and the value of *MaxNumReTx* will be used for the hybrid protocol in comparison to other protocols.

b. Comparison of Protocols with Varying Fading Effects



(a) Packet delivery ratio (b) Energy consumption

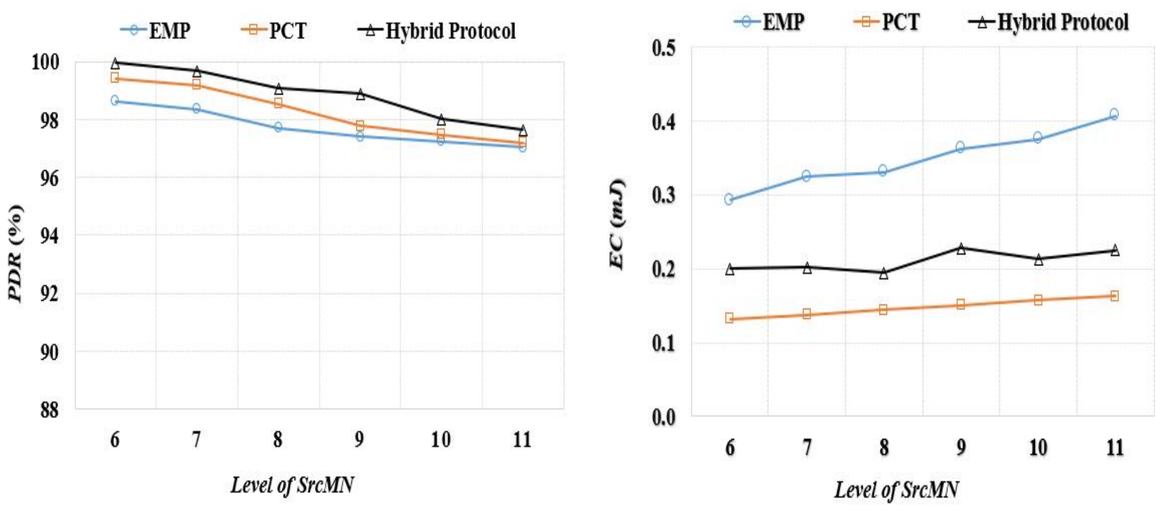
Figure 30. The PDR and EC of three protocols with varying Ricean k factor, node 14 at level 8 is the selected multimedia node

Figure 30 compares the packet delivery ratio and energy consumption of three protocols: EMP, PCT, and LoRa-WSN hybrid protocol, according to the increase in the degree of fading effect. It is clearly shown that the hybrid protocol consistently achieved the highest PDR regardless of fading k value, and it was very reliable (its PDR of over 96%) even in a harsh environment ($k = 4$). Furthermore, three protocols showed high PDRs of over 90% because they use retransmission or a cooperative path. The hybrid protocol is less sensitive to changes in Ricean k value; however, both PCT and EMP tend to lower PDR quite sensitively as Ricean k value decreases. Since the decrease in k indicates the high degree of multipath fading effect, EMP that uses multiple paths is more sensitive to the decrease of in Ricean k value than PCT, which uses a cooperative path.

Figure 30 (b) shows that ECs decreases as the k value goes up because more packets are lost with the increase in the multipath fading effect. Furthermore, the hybrid protocol lowers energy consumption considerably compared to EMP, down to the slightly higher energy consumption of PCT.

Based on this discussion, we can conclude that the LoRa-WSN hybrid protocol is a highly dependable image transmission protocol because it achieves high PDR comparable to EMP, low energy consumption comparable to PCT, and it is also less sensitive to the multipath fading effect.

c. Comparison of Protocols with Varying Level of SrcMN



(a) Packet delivery ratio (b) Energy consumption

Figure 31. The PDR and EC of three protocols with varying level of source multimedia node and the Ricean k = 8

Figure 31 compares the packet delivery ratio and energy consumption of the three protocols according to the level of the *SrcMN*, *level*(*SrcMN*). It is obvious that the three protocols achieve high PDRs of over 97% and they are less sensitive to an increase in the level of the *SrcMN*. This is because three protocols concentrate on improving the reliability of packet transmission by using retransmission or cooperating nodes to assist path nodes. The PDRs are reduced by about 2% as the increases in *level*(*SrcMN*) from 6 to 11. The LoRa-WSN hybrid protocol consistently maintains the best PDR regardless of fading *k* value, while the PDR under PCT is slightly higher than that under EMP.

Figure 31 (b) shows that the PCT consistently shows the lowest EC and has a slightly decreasing EC pattern with the increases in level of *SrcMN*. This is because it uses a cooperative path without retransmission and slot scheduling. Meanwhile, the EC patterns of the EMP and hybrid protocol increase constantly with the increases in level of the *SrcMN*. This is because they use retransmission. Note that the number of lost packets under EMP is the highest, as shown in Figure 30 (a). Thus, EMP consumes the most energy to deliver an image to the gateway.

Chapter 5

CONCLUSIONS AND FUTURE STUDY

5.1 Remarks about Our Work

The transmission of an image in WSN must meet several requirements, including high reliability in data transmission, low delay in image transmission, and high efficiency in energy management. However, unstable wireless links, low bandwidth, and tight resource constraints are heuristic characteristics of WISN, leading to a lot of difficulties in delivering an image. Hence, some new approaches are proposed to meet the requirements for image delivery in WISNs. These approaches need to concentrate not only on the design of a new transmission protocol and its evaluation by simulation, but also on the implementation and deployment of the proposed network to see if it can satisfy the stringent requirements of WISN applications.

In this thesis, A *pipelined cooperative transmission* (PCT) protocol was proposed to satisfy bounds for end-to-end delay and the packet loss ratio in delivering an image in WISNs. PCT is characterized by the use of a cooperative path and the pipelined transmission. The cooperative path contributes to not only improving reliability in packet transmission, but also shortening image transmission delay significantly by removing retransmissions. It could deliver an image over 11 hops to the sink in a reliable manner, even in the high degree of fading effect environment.

According to experiments, the PCT protocol could deliver an image of 34 Kbytes with image transmission delay of less than 0.4 seconds without any loss of packet and also about 94% of totally generated images, even with a long-distance multi-hop topology and the high interference of WiFi signals. Furthermore, PCT achieved high energy efficiency and low end-to-end delay, compared with the recent EMP approach. The results proves that the PCT protocol is highly dependable to satisfy the most industry requirements for image delivery, in terms of reliability in packet and image transmission, and the end-to-end delay of image delivery.

Moreover, a simple LoRa-WSN hybrid network protocol was designed to improve the reliability of image packet transmission and to deliver scalar data in a real-time manner. The proposed protocol is characterized by the combination of the LoRa network and WSN to deliver scalar data and multimedia data, respectively. In scalar data transmission, nodes are assigned logical slot indices by using a logical slot indexing algorithm that can satisfy time bounds in scalar data transmission and they transmit scalar data in each assigned slot. In image transmission, the selected multimedia node transmits an image packet in every transmission interval and uses LCM to get information about lost packets for path-wide retransmission.

Simulation results show that the LoRa-WSN hybrid protocol could deliver an image of 220 packets from the *SrcMN* at a long hop distance in a reliable manner. It maintained a packet delivery ratio of more than 96% even in a harsh environment of multipath fading ($k = 4$). The hybrid protocol outperforms PCT and EMP in terms of packet delivery ratio under different levels of multipath fading. Furthermore, the hybrid protocol achieved high energy efficiency that was comparable to PCT and much better than the EMP approach. Thus, it is believed that the LoRa-WSN hybrid protocol can meet the requirements for image transmission in WSN.

To summarize, we believe that this thesis fills a need on the topic of image transmission in WSNs, and we hope it will be useful for researchers who want to begin contributing or further exploring this promising research domain.

5.2 Future Research Directions

In further researches, the PCT protocol can be improved by using multiple cooperative paths to remove the bubble slot problem. Furthermore, the LoRa-WSN protocol will be evaluated systematically and examined for its applicability to applications in real-industrial environments such as factories using a hybrid sensor node with a LoRa communication module, a sensor communication module, and a scalar sensor module, as well as a multimedia hybrid node with an additional multimedia module. Moreover, the hybrid protocol can shorten the end-to-end delay of image transmission by using multiple channels and multiple paths.

PUBLICATIONS

[1] D. S. Yoo, V. K. Ta, B. T. Jang, and H. Oh, "An Energy-Efficient Slotted Sense Multiple Access Broadcast Protocol for Reliable Command Delivery in Dynamic Wireless Sensor Networks," vol. 19, no. 5, *Sensors* (Basel), March 2019, pp. 1236-1256.

[2] V. K. Ta and H. Oh, "A Pipelined Cooperative Transmission Protocol for Fast and Reliable Image Delivery in Wireless Sensor Networks," *IEEE Access*, vol. 8, pp. 142758-142771, 2020, doi: 10.1109/ACCESS.2020.3013738.

[3] V. K. Ta and H. Oh, "Implementation and evaluation of image transmission protocol in wireless multimedia sensor networks," *in 2021 4th international conference on sensors, signal, and image processing* (SSIP 2021), 15-17 Oct 2021, doi: https://doi.org/10.1145/3502814.3502823.

REFERENCES

[1] W. Zhang *et al.*, "Adaption Resizing Communication Buffer to Maximize Lifetime and Reduce Delay for WVSNs," *IEEE Access,* vol. 7, pp. 48266-48287, 2019, doi: 10.1109/access.2019.2909869.

[2] A. Sikora and V. F. Groza, "Coexistence of IEEE802.15.4 with other Systems in the 2.4 GHz-ISM-Band," in *2005 IEEE Instrumentationand Measurement Technology Conference Proceedings*, 16-19 May 2005 2005, vol. 3, pp. 1786-1791, doi: 10.1109/imtc.2005.1604479.

[3] V. Jeličić and V. Bilas, "Reducing power consumption of image transmission over IEEE 802.15.4/ZigBee sensor network," in *2010 IEEE Instrumentation & Measurement Technology Conference Proceedings*, 3-6 May 2010 2010, pp. 1211-1215, doi: 10.1109/imtc.2010.5488204.

[4] H. Wu and A. A. Abouzeid, "Energy efficient distributed image compression in resource-constrained multihop wireless networks," *Computer Communications,* vol. 28, no. 14, pp. 1658-1668, 2005, doi: https://doi.org/10.1016/j.comcom.2005.02.018.

[5] I. F. Akyildiz, T. Melodia, and K. R. Chowdury, "Wireless multimedia sensor networks: A survey," *IEEE Wireless Communications,* vol. 14, no. 6, pp. 32-39, 2007, doi: 10.1109/MWC.2007.4407225.

[6] A. Pinto, Z. Zhang, X. Dong, S. Velipasalar, M. C. Vuran, and M. C. Gursoy, "Energy Consumption and Latency Analysis for Wireless Multimedia Sensor Networks," in *2010 IEEE Global Telecommunications Conference GLOBECOM 2010*, 6-10 Dec. 2010 2010, pp. 1-5, doi: 10.1109/GLOCOM.2010.5683898.

[7] A. Rowe, D. Goel, and R. Rajkumar, "FireFly Mosaic: A Vision-Enabled Wireless Sensor Networking System," in *28th IEEE International Real-Time Systems Symposium (RTSS 2007)*, 3-6 Dec. 2007 2007, pp. 459-468, doi: 10.1109/RTSS.2007.50.

[8] R. Kleihorst, A. Abbo, B. Schueler, and A. Danilin, "Camera Mote with a High-Performance Parallel Processor for Real-Time Frame-Based Video Processing," in *2007 First ACM/IEEE International Conference on Distributed Smart Cameras*, 25-28 Sept. 2007 2007, pp. 109-116, doi: 10.1109/ICDSC.2007.4357513.

[9] T. Melodia and I. F. Akyildiz, "Cross-Layer Quality of Service Support for UWB Wireless Multimedia Sensor Networks," in *IEEE INFOCOM 2008 - The 27th Conference on Computer Communications*, 13-18 April 2008 2008, pp. 2038-2046, doi: 10.1109/INFOCOM.2008.269.

[10] Y. Sun, H. Ma, L. Liu, and Y. e. Zheng, "ASAR: An ant-based service-aware routing algorithm for multimedia sensor networks," *Frontiers of Electrical and Electronic Engineering in China,* vol. 3, no. 1, pp. 25-33, 2008/01/01 2008, doi: 10.1007/s11460-008-0013-7.

[11] W. Wang, D. Peng, H. Wang, H. Sharif, and H. Chen, "Energy-Constrained Distortion Reduction Optimization for Wavelet-Based Coded Image Transmission in Wireless Sensor Networks," *IEEE Transactions on Multimedia,* vol. 10, no. 6, pp. 1169-1180, 2008, doi: 10.1109/TMM.2008.2001354.

[12] J. J. Ahmad, H. A. Khan, and S. A. Khayam, "Energy efficient video compression for wireless sensor networks," in *2009 43rd Annual Conference on Information Sciences and Systems*, 18-20 March 2009 2009, pp. 629-634, doi: 10.1109/CISS.2009.5054795.

[13] A. Boukerche, D. Yan, F. Jing, and R. Pazzi, "A reliable synchronous transport protocol for wireless image sensor networks," in *2008 IEEE Symposium on Computers and Communications*, 6-9 July 2008 2008, pp. 1083-1089, doi: 10.1109/iscc.2008.4625679.

[14] C. Casetti, M. Gerla, S. Mascolo, M. Y. Sanadidi, and R. Wang, "TCP Westwood: End-to-End Congestion Control for Wired/Wireless Networks," *Wireless Networks,* vol. 8, no. 5, pp. 467-479, 2002, doi: 10.1023/a:1016590112381.

[15] J.-H. Lee and I.-B. Jung, "Reliable asynchronous image transfer protocol in wireless multimedia sensor networks," *Sensors (Basel, Switzerland),* vol. 10, no. 3, pp. 1486-1510, 2010, doi: 10.3390/s100301487.

[16] J.-H. Lee, "A Massive Transmission Scheme in Contention-Based MAC for Wireless Multimedia Sensor Networks," *Wireless Personal Communications,* vol. 71, no. 3, pp. 2079-2095, 2013, doi: 10.1007/s11277-012-0924-8.

[17] M. Buettner, G. Yee, E. Anderson, and R. Han, *X-MAC: A short preamble MAC protocol for duty-cycled wireless sensor networks*. Proceedings of the 4th International Conference on Embedded Networked Sensor Systems, 2006.

[18] S. Kim *et al.*, "Flush: a reliable bulk transport protocol for multihop wireless networks," in *SenSys '07*, 2007.

[19] A. C. Valera, W.-S. Soh, and H.-P. Tan, "Enabling sustainable bulk transfer in environmentally-powered wireless sensor networks," *Ad Hoc Networks,* vol. 54, pp. 85-98, 2017/01/01/ 2017, doi: https://doi.org/10.1016/j.adhoc.2016.10.008.

[20] F. Österlind and A. Dunkels, "Approaching the maximum 802.15.4 multi-hop throughput," 2008.

[21] V. Gabale, K. Chebrolu, B. Raman, and S. Bijwe, "PIP: A multichannel, TDMA-based MAC for efficient and scalable bulk transfer in sensor networks,"

ACM Transactions on Sensor Networks (TOSN), vol. 8, 2012, doi: 10.1145/2240116.2240117.

[22] M. Doddavenkatappa and M. Choon, "P3: A Practical Packet Pipeline using synchronous transmissions for wireless sensor networks," in *IPSN-14 Proceedings of the 13th International Symposium on Information Processing in Sensor Networks*, 15-17 April 2014 2014, pp. 203-214, doi: 10.1109/ipsn.2014.6846753.

[23] F. Ferrari, M. Zimmerling, L. Thiele, and O. Saukh, "Efficient network flooding and time synchronization with Glossy," in *Proceedings of the 10th ACM/IEEE International Conference on Information Processing in Sensor Networks*, 12-14 April 2011 2011, pp. 73-84.

[24] D. Pyeon and H. Yoon, "An efficient multi-path pipeline transmission for a bulk data transfer in IEEE 802.15.4 multi-hop networks," *Wireless Networks,* vol. 25, no. 1, pp. 117-130, 2019, doi: 10.1007/s11276-017-1542-x.

[25] H. S. Aghdasi, M. Abbaspour, and M. E. Moghadam, "An Energy-Efficient and High-Quality MAC Protocol for Image Transmission in Wireless Sensor Networks," in *2008 4th IEEE International Conference on Circuits and Systems for Communications*, 26-28 May 2008, pp. 838-842, doi: 10.1109/iccsc.2008.183.

[26] J. Polastre, J. L. Hill, and D. E. Culler, "Versatile low power media access for wireless sensor networks," in *SenSys '04 - Proceedings of the Second International Conference on Embedded Networked Sensor Systems*, 2004, pp. 95-107.

[27] H. Le, W. Hu, P. Corke, and S. Jha, *"ERTP: Energy-efficient and Reliable Transport Protocol for data streaming in Wireless Sensor Networks"*. 2009, pp. 1154-1171.

[28] A. Genta, K.Lobiyal, and J. Abawajy, "Energy Efficient Multipath Routing Algorithm for Wireless Multimedia Sensor Network," *Sensors,* vol. 19, p. 3642, 2019, doi: 10.3390/s19173642.

[29] S. M. Aziz and D. M. Pham, "Energy Efficient Image Transmission in Wireless Multimedia Sensor Networks," *IEEE Communications Letters,* vol. 17, no. 6, pp. 1084-1087, 2013, doi: 10.1109/lcomm.2013.050313.121933.

[30] N. Abbas, F. Yu, and Y. Fan, "Intelligent Video Surveillance Platform for Wireless Multimedia Sensor Networks," *Applied Sciences,* vol. 8, no. 3, 2018, doi: 10.3390/app8030348.

[31] H. Ma, L. Liu, and H. Luo, "Introduction to Multimedia Sensor Networks," in *Multimedia Sensor Networks*, (Advances in Computer Science and Technology, 2021, ch. Chapter 1, pp. 1-10.

[32] B. Tavli, K. Bicakci, R. Zilan, and J. M. Barcelo-Ordinas, "A survey of visual sensor network platforms," *Multimedia Tools and Applications,* vol. 60, no. 3, pp. 689-726, 2012/10/01 2012, doi: 10.1007/s11042-011-0840-z.

[33] S. Hengstler, D. Prashanth, S. Fong, and H. Aghajan, "MeshEye: A Hybrid-Resolution Smart Camera Mote for Applications in Distributed Intelligent Surveillance," in *2007 6th International Symposium on Information Processing in Sensor Networks*, 25-27 April 2007 2007, pp. 360-369, doi: 10.1109/IPSN.2007.4379696.

[34] K. Sayood, "Chapter 1 - Introduction," in *Introduction to Data Compression (Fifth Edition)*, K. Sayood Ed.: Morgan Kaufmann, 2018, pp. 1-10.

[35] J.-L. Z. a. L. Nie, *Energy efficiency of multi-hop communication in wireless sensor network*. 2015 International Workshop on Wireless Communication and Network: https://doi.org/10.1142/9789814733663_0051, 2015.

[36] C. Li, P. Wang, H. Chen, and M. Guizani, "A Cluster Based On-demand Multi-Channel MAC Protocol for Wireless Multimedia Sensor Networks," in *2008 IEEE International Conference on Communications*, 19-23 May 2008 2008, pp. 2371-2376, doi: 10.1109/icc.2008.451.

[37] M. Yigitel, O. Incel, and C. Ersoy, *Design and implementation of a QoS-aware MAC protocol for Wireless Multimedia Sensor Networks*. 2011, pp. 1991-2001.

[38] O. Incel, L. Hoesel, P. Jansen, and P. Havinga, "MC-LMAC: A multi-channel MAC protocol for wireless sensor networks," *Ad Hoc Networks,* vol. 9, pp. 73-94, 2011, doi: 10.1016/j.adhoc.2010.05.003.

[39] D. S. Yoo, V. K. Ta, B. T. Jang, and H. Oh, "An Energy-Efficient Slotted Sense Multiple Access Broadcast Protocol for Reliable Command Delivery in Dynamic Wireless Sensor Networks," vol. 19, no. 5): Sensors (Basel), March, 2019, pp. 1236-1256.

[40] H. Oh and C. T. Ngo, "A Slotted Sense Multiple Access Protocol for Timely and Reliable Data Transmission in Dynamic Wireless Sensor Networks," *IEEE Sensors Journal,* vol. 18, no. 5, pp. 2184-2194, 2018, doi: 10.1109/jsen.2018.2790422.

[41] M. A. K. Azad, C. T. Ngo, and H. Oh, "A two-channel slotted sense multiple access protocol for timely and reliable data transmission in industrial wireless sensor networks," *International Journal of Distributed Sensor Networks,* vol. 16, no. 4, 2020, doi: 10.1177/1550147720902007.

[42] P. Vinh and H. Oh, "Optimized Sharable-Slot Allocation Using Multiple Channels to Reduce Data-Gathering Delay in Wireless Sensor Networks," *Sensors,* vol. 16, pp. 505-526, 2016, doi: 10.3390/s16040505.

[43] *https://www.qt.io/product/development-tools*.

[44] B. Chaudhari and S. Borkar, "2 - Design considerations and network architectures for low-power wide-area networks," in *LPWAN Technologies for IoT and M2M Applications*, B. S. Chaudhari and M. Zennaro Eds.: Academic Press, 2020, pp. 15-35.

[45] F. Adelantado, X. Vilajosana, P. Tuset-Peiro, B. Martinez, J. Melia-Segui, and T. Watteyne, "Understanding the Limits of LoRaWAN," *IEEE Communications Magazine,* vol. 55, no. 9, pp. 34-40, 2017, doi: 10.1109/MCOM.2017.1600613.

[46] R. Liang, L. Zhao, and P. Wang, "Performance Evaluations of LoRa Wireless Communication in Building Environments," *Sensors,* vol. 20, no. 14, 2020, doi: 10.3390/s20143828.

[47] C. Gu, R. Tan, X. Lou, and D. Niyato, "One-Hop Out-of-Band Control Planes for Low-Power Multi-Hop Wireless Networks," in *IEEE INFOCOM 2018 - IEEE Conference on Computer Communications*, 16-19 April 2018 2018, pp. 1187-1195, doi: 10.1109/INFOCOM.2018.8486301.

[48] Q. L. Hoang, W. Jung, T. Yoon, D. Yoo, and H. Oh, "A Real-Time LoRa Protocol for Industrial Monitoring and Control Systems," *IEEE Access,* vol. 8, pp. 44727-44738, 2020, doi: 10.1109/ACCESS.2020.2977659.