RELIABLE LOW POWER NETWORK PROTOCOL IN IOT FOR AGRICULTURE FIELD

BY

CHRISTOPHER WONG JIN SOON

A REPORT
SUBMITTED TO
Universiti Tunku Abdul Rahman
in partial fulfillment of the requirements
for the degree of
BACHELOR OF INFORMATION TECHNOLOGY (HONOURS) COMMUNICATIONS AND NETWORKING
Faculty of Information and Communication Technology
(Kampar Campus)

JUN 2024

UNIVERSITI TUNKU ABDUL RAHMAN

REPORT STATUS DECLARATION FORM

Title: RELIABLE LOW POWER NETWORK PROTOCOL
IN IOT FOR AGRICULTURE FIELD
Academic Session: June 2024

I CHRISTOPHER WONG JIN SOON
(CAPITAL LETTER)

declare that I allow this Final Year Project Report to be kept in
Universiti Tunku Abdul Rahman Library subject to the regulations as follows:

1. The dissertation is a property of the Library.
2. The Library is allowed to make copies of this dissertation for academic purposes.

Verified by,

(Author's signature)

(Supervisor's signature)

Address:
108 Laluan Perajurit 1,
Taman Ipoh Timur,
31400, Ipoh, Perak

Goh Hock Guan
Supervisor's name

Date: 12/9/2024

Date: 13/9/2024

Universiti Tunku Abdul Rahman			
Form Title : **Sample of Submission Sheet for FYP/Dissertation/Thesis**			
Form Number: **FM-IAD-004**	Rev No.: **0**	Effective Date: **21 JUNE 2011**	Page No.: **1 of 1**

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY

UNIVERSITI TUNKU ABDUL RAHMAN

Date: 12/9/2024

SUBMISSION OF FINAL YEAR PROJECT /DISSERTATION/THESIS

It is hereby certified that ______***Christopher Wong Jin Soon***_______ (ID No: __***20ACB01539***__) has completed this final year project entitled "RELIABLE LOW POWER NETWORK PROTOCOL IN IOT FOR AGRICULTURE FIELD" under the supervision of Ts Dr Goh Hock Guan (Supervisor) from the Department of Computer and Communication Technology, Faculty of Information and Communication Technology

I understand that University will upload softcopy of my final year project in pdf format into UTAR Institutional Repository, which may be made accessible to UTAR community and public.

Yours truly,

Christopher Wong Jin Soon
(*Student Name*)

*Delete whichever not applicable

DECLARATION OF ORIGINALITY

I declare that this report entitled "**RELIABLE LOW POWER NETWORK PROTOCOL IN IOT FOR AGRICULTURE FIELD**" is my own work except as cited in the references. The report has not been accepted for any degree and is not being submitted concurrently in candidature for any degree or other award.

Signature :

Name : Christopher Wong Jin Soon

Date : 12/9/2024

ACKNOWLEDGEMENTS

I would like to express my sincere thanks and appreciation to my supervisor, Dr Goh Hock Guan who has given me this opportunity to carry out a network protocol design project. It has provided me with a glimpse into the R&D required to develop a functional network protocol and will be a useful tool for me to enter the networking R&D field in the future.

I must also thank to my parents for their love, support, and continuous encouragement throughout the course. Their unconditional support and sacrifices have been the cornerstone of my academic journey.

ABSTRACT

An Internet of Things (IoT) sensor network consists of sensor nodes which sample and send data to the base station. Agriculture is one sector which makes use of IoT sensor networks for collecting data on the growing environment of crops for farmers to perform adjustments for optimal crop growth. Deployment of IoT networks in agriculture use cases face multiple constraints, including power constraints. Thus, IoT systems from other industrial sectors cannot be used since they do not have the same constraints or requirements. Existing research in energy efficient network protocols have problems with reliability or weaknesses that make them unsuitable for direct use in agriculture use cases. This project suggests the design of a new reliable and energy-efficient network protocol for deployment in a simulated agriculture use case. This network protocol can be further split into three separate protocols, which are the application layer protocol, network layer protocol and data link layer protocol. The application layer protocol handles the sampling and caching of data, which saves energy by maximizing the data segment of packets sent. The network layer protocol simplifies the routing process based on connection tables from BNT. The data link layer protocol handles the Medium Access Control (MAC) of the mesh network efficiently using algorithms based on the Scheduling sub-cycle of the Fast, Adaptive, and Energy-efficient Multi-channel-multi-path (FAEM) protocol by Liew et al. The new network protocol will be tested in a simulation and benchmarked against other similar energy-efficient protocols to determine its capability.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

IoT	Internet of Things
E-PEGASIS	Extended Power Efficient Gathering in Sensor Information Systems
PEGASIS	Power Efficient Gathering in Sensor Information Systems
FAEM protocol	Fast, Adaptive, and Energy-efficient Multi-channel-multi-path protocol
BNT	Basketball Net Topology
MAC	Medium Access Control
IM-P	Iterative Matching as Parent
IM-C	Iterative Matching as Child
CSMA/CA	Carrier-sense multiple access with collision avoidance
LEER	Low Energy nodes Efficient Routing
OSI	Open Systems Interconnection
IP	Internet Protocol
MTU	Maximum Transmission Unit

Chapter 1

Introduction

The Internet of Things (IoT) refers to the concept of devices connected over a network or the Internet that are used to perform data collection, analysis and decision-making [1]. Gartner's Hype Cycle states that the "hype", which means expectation or interest, for IoT peaked in 2016 [2]. The Hype Cycle has also shown that interest in IoT now has dropped significantly compared to its peak, but IoT shows stable signs of growth as more industries and sectors integrate IoT into their workflows and systems. Example use cases of IoT include fleet management of transport networks [3], smart manufacturing [4], and smart cities [5].

A less commonly mentioned but still significant sector that utilizes the IoT is agriculture. The use of IoT networks in this sector includes sensor networks which sample and record various factors needed for optimal crop yields, such as weather monitoring, soil conditions monitoring, disease monitoring and irrigation monitoring [6]. The sensor nodes which make up these wireless sensor networks consist of a sensor module (e.g., temperature sensor, ground humidity sensor), a computing unit (microcontroller), transceiver modules (e.g., Wi-Fi, Zigbee) and a power unit (e.g., battery, solar cell) [7].

The agriculture sector is not a monolithic and homogeneous sector. The different subsectors within the agriculture sector each have their own needs and deploy their IoT networks in many different environments. For example, a palm oil plantation and a paddy field have different environmental conditions that affect the deployment of IoT sensor networks. To the best of my knowledge, there is no one-size-fits-all IoT solution available in the market. Farmers must specify their requirements so that IoT system providers can help customize their software and hardware accordingly before deployment.

While the difference between subsectors of the agriculture sector makes the deployment of IoT networks harder, this difference is even more pronounced when compared with other industrial uses of IoT networks. For example, limitations of the agriculture sector such as the lack of power infrastructure and communications infrastructure might not be addressed by IoT systems designed for smart manufacturing. The agriculture and manufacturing sectors both have different priorities and demands for their IoT systems and thus their IoT systems are not interchangeable.

1.1 Problem Statement and Motivation

Motivation

The motivation for this project is the difficulty of setting up and maintaining an IoT sensor network in agricultural environments. The agriculture sector is often located in suburban or rural areas, which can be hard to traverse especially in certain agriculture subsectors like palm oil plantations or paddy fields.

Various constraints and factors must be taken into consideration when designing an IoT network, including but not limited to power, communication and location constraints.

The power constraints of an IoT network stem from the fact that sometimes a convenient and consistent source of power is unavailable. This problem is especially prevalent in the agriculture sector where most fields or farms do not have complete electricity infrastructure. To deal with this issue, IoT systems use energy storage (e.g., Lithium-ion batteries) and energy harvesting from alternative power sources (e.g., solar cells) [8]. Even then, these energy sources either provide a limited pool of energy or a non-continuous source of energy.

If a deployed sensor network encounters frequent failures that require a technician to physically locate and fix the problem, including network outages and battery replacements, the costs incurred for the operation and maintenance of the network will be high. Besides costs, the downtime of the network might disrupt the workflow of farmers, potentially wasting precious work hours.

Problem Statement

For the IoT sensor nodes, the transmission and reception of data is the process that consumes the most power for IoT devices [9]. If prolonged periods of sending and receiving data packets are done using an inefficient network protocol, the sensor node’s power budget will be put under strain, resulting in the need for constant replacements of batteries or even result in abrupt network outages for sensor nodes relying on alternative power sources.

If the network protocols used by an IoT sensor network are not designed with reliability or redundancy in mind, a lot of data might be lost if a sensor node fails. Each sensor node that is not a leaf node will act as a router for outgoing data that go from the sensor node’s children towards the base station, and for incoming instructions that go from the base station towards a specific sensor node. If a sensor node only has one parent node (e.g., in a tree topology) that has failed and no other backup connections exist, all data collected by that sensor node and received from its children will be discarded due to not being able to be sent towards the base station. Besides node failure, data loss might occur during transmission due to the network protocol not having mechanisms for detecting data loss and requesting retransmission. Such network protocols are described as “unreliable” network protocols.

1.2 Objectives

The aims of the final year project are:

1. To design a new energy-efficient network protocol. The network protocol must have the same level of energy efficiency as the FAEM protocol without certain weaknesses such as the lack of “hot joining” mechanisms.
2. To improve the reliability of a WSN in terms of data transmissions and network topology. The number of retransmissions needs to be low to save energy and ensure as much data as possible reaches the base station. Redundancy in the network topology is to reduce the effects of single-node failures on the network.

1.3 Project Scope

This project will be a research-based project in which a network protocol will be designed which will encompass multiple layers in the OSI (Open Systems Interconnection) model, specifically the application layer, the network layer, and the data link layer. Some assumptions will be made about the environmental conditions of the simulated deployment environment, the energy consumption of the IoT sensor nodes and the occupancy of the transmission medium. The following are the scopes of the project:

1. The development of an energy-efficient network protocol that will be used and tested in a simulated paddy field using multiple different topologies.
2. The modelling of power consumption of the network protocol by measuring the duration of receiving and transmitting data of each sensor node.
3. The measurement of throughput of the network protocol by measuring the amount of data sent and received per unit of time.
4. The measurement of reliability of the network protocol by measuring the percentage of packet loss during transmission and reception of data as well as monitoring the stability of the network topology.
5. The benchmarking of the new network protocol against other modelled network protocols, to determine its capabilities and suitability as an energy-efficient network protocol.

1.4 Contributions

The contributions of the project include the design of a new network protocol that improves the energy efficiency of a network. This increased energy efficiency prolongs the lifetime of the network, resulting in fewer downtimes caused by nodes running out of energy and reducing the frequency of battery replacements. By using this energy-efficient network protocol, farmers can reduce the frequency of trips needed for the replacement of battery replacements, which will save costs and time. Furthermore, this energy-efficient network protocol can be further adapted into other practical use cases by future research works or act as a reference for future research into energy-efficient network protocols.

The new network protocol will also be reliable in terms of data transmission and network topology. This cuts down the percentage of retransmissions, which saves more energy, and reduces the percentage of data loss, which results in fewer data gaps in a database. Farmers will have more data to rely on which enables them to make more accurate decisions in the field. The improved reliability of network topology also reduces the possibility for parts of a network to be disconnected from the base station because of a single node failure. This reduction in major network outages is crucial for farmers who spend most of their workday in the field, only spending a smaller portion of work hours doing office work such as interacting with the sensor network and retrieving data from the sensor network database. By the time the farmer has identified the occurrence of a major network outage, a lot of data would have already been lost, potentially affecting the next decision-making phase thus the reduction in major network outages can prevent inaccurate decision-making. Besides that, the demonstration of a network protocol with reliability features can help future research works implement similar features to improve their network protocols’ reliability.

1.5 Report Organization

This FYP report consists of seven chapters.

Chapter 1 introduces the project by highlighting the motivation of the project and the problems the project aims to address.

Chapter 2 involves reviewing existing literature for knowledge relevant to WSNs and WSN applications in the agriculture sector. Existing data collection or data aggregation protocols used in WSNs are reviewed in a section of this chapter. Additionally, energy-efficient data collection protocols and reliable data collection protocols are reviewed to identify potential improvements to be implemented in the new network protocol or potential weaknesses that the new network protocol can aim to address.

Chapter 3 explains the models used to model and test the new network protocol. This includes the topology of the tested network, radio model, battery model and reliability models.

Chapter 4 explains the new network protocol that is being developed as part of this project. The chapter will explain its functionality and features in terms of achieving energy efficiency and reliability.

Chapter 5 explains the hardware and software setup, as well as the simulation settings that were used to run the simulation. This chapter also discusses the challenges encountered during the implementation of the network protocol.

Chapter 6 is for the evaluation of the simulation results and the objectives of this project. This chapter is also used to reflect upon the challenges encountered during this project.

Chapter 7 serves as the conclusion in which the results of the development and simulation of the new network protocol will be summarized. The chapter also contains recommendations on potential improvements to the network protocol in the future.

Chapter 2

Literature Review

2.1 Wireless Sensor Network (WSN)

Kandris et al [10] describe WSN as a group of sensor nodes linked together through wireless communication. A sensor node typically consists of a processor, storage medium, a transceiver, one or more sensors, an analog-to-digital converter (ADC), and a battery or other similar power sources. Each sensor node uses its sensors to detect changes in the surrounding environment and after being processed by the node's processors, the data is then sent to the base station or to other intermediate nodes.

Kandris et al states that the six most popular uses for WSNs are military applications, health applications, environmental applications, flora and fauna applications, industrial applications, and urban applications.

Within military applications, WSNs have been developed and used to perform tasks such as battlefield surveillance and combat monitoring. Sensor nodes equipped with magnetic and acoustic sensors can be deployed on battlefields to help militaries detect the direction of weapon fire and locate their source. This works even for locating indirect fire weapons such as mortars and artillery which are usually not within the line of sight of sensors or military observers.

For health applications, WSNs are used with medical sensors such as ECG sensors, blood pressure sensors and blood glucose sensors so that real-time information about a patient's health can be provided to the healthcare provider. These usages of WSNs in the medical field provide patients with various benefits such as remote health monitoring and health alert systems.

Environmental applications of WSNs include water and air monitoring are used to detect the quality of air or water and the presence of specific substances or impurities in the place of interest. Drinking water quality and pipe leakage are some of the tasks that water monitoring WSNs perform. Another type of environmental application of WSNs is emergency alerting in which WSNs are deployed to provide early warning alerts about potential natural disasters such as earthquakes, forest fires and tsunamis.

Flora & fauna applications of WSNs include greenhouse monitoring, in which WSNs are used to monitor environmental factors inside a greenhouse which help in performing precise climate control in the greenhouse. Besides greenhouse monitoring, crops and livestock can also be monitored through the deployment of WSNs. Information about soil conditions and livestock behaviour can help farmers optimize their decisions and perform emergency corrections.

Industrial applications of WSN include logistic use cases and machinery health monitoring. In logistics, WSNs can be used to monitor cargo containers to ensure better transportation conditions for the goods within. For machinery health monitoring, WSNs are deployed to monitor machinery and pipelines to determine their condition and efficiency non-intrusively.

Urban applications of WSN help improve the quality of life of citizens. Projects for such systems include a system for road traffic control near flooded tunnels. Sensor nodes located inside flood-prone tunnels continuously monitor water levels so that central control systems can manage the use of water pumps while also diverting traffic away from tunnel entrances. Structural monitoring is another example of urban applications of WSN in which sensor nodes are placed throughout a structure such as a building or a bridge to detect vibrations and stresses to infer the structural integrity of the structure being monitored.

2.2 WSN in Agriculture

Kiani and Seyyedabbasi [11] suggest the usage of WSN and IoT in precision agriculture to improve the production of crops in small farms. Their proposed system architecture deploys sensor nodes equipped with humidity, temperature, and soil moisture sensors in the farm environment. These sensor nodes collect data on an hourly basis and transmit it to the nearest gateway either directly or by being passed on through other sensor nodes. The server stores this information and provides it to the farmers in the form of tables and graphs. This historical data, combined with weather forecasts provided by the server helps farmers make better informed decisions. The paper also suggests the combination of this system with remotely controlled actuators that allow the farmer to perform some of the work such as watering the plants.

Ojha et al. [12] provide a few case studies on real-world implementations of WSNs in agriculture. A 2011 case study on an alfalfa crop irrigation cut-off system highlights the inefficiency of the flood-irrigation method that is used for water alfalfa which causes water runoffs. The new irrigation system that was developed involves planting the alfalfa on a slightly sloped field with an irrigator at the top of the field and water detection sensor nodes at the bottom of the slope. During an irrigation cycle, an operator will activate the irrigators at the top of the slope and let water trickle down the slope. When the water reaches the bottom of the slope, the water detection sensor nodes will send an SMS message to the operator to notify them to turn off the irrigator. Through this method, sufficient water will be provided to the whole alfalfa field with minimal wastage. Another 2009 case study involved the development of a prediction system for Downy Mildew disease which is a weather-related disease that affects grape plants. Sensor nodes equipped with temperature, humidity and leaf wetness duration sensors were deployed. The data collected by the sensors were put through two probability models to calculate the "Infection Index", which likely meant the probability of a disease outbreak.

2.3 Energy-Efficiency and Reliability in Data Collection

2.3.1 Enhanced Energy Efficient Routing for Wireless Sensor Network Using Extended Power Efficient Gathering in Sensor Information Systems (E-PEGASIS) Protocol

Extended Power Efficient Gathering in Sensor Information Systems (E-PEGASIS) [13] is a chain-based network protocol built on the Power Efficient Gathering in Sensor Information Systems (PEGASIS) chain-based routing protocol.

Every cycle in E-PEGASIS consists of three phases which are the Construction of Chain Phase, Enhanced Leader Selection Phase and Data Transmission Phase.

During the Construction of Chain Phase, the nodes in a sensor network are linked into chains by the application of a greedy protocol. The algorithm used for the construction of chains is shown in Figure 2.2.

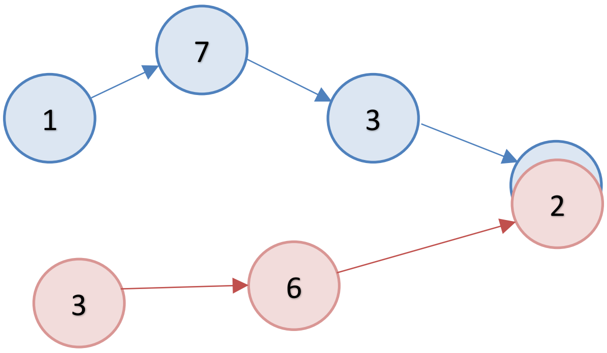


Figure 2.1 An Example E-PEGASIS Topology with Two Chains

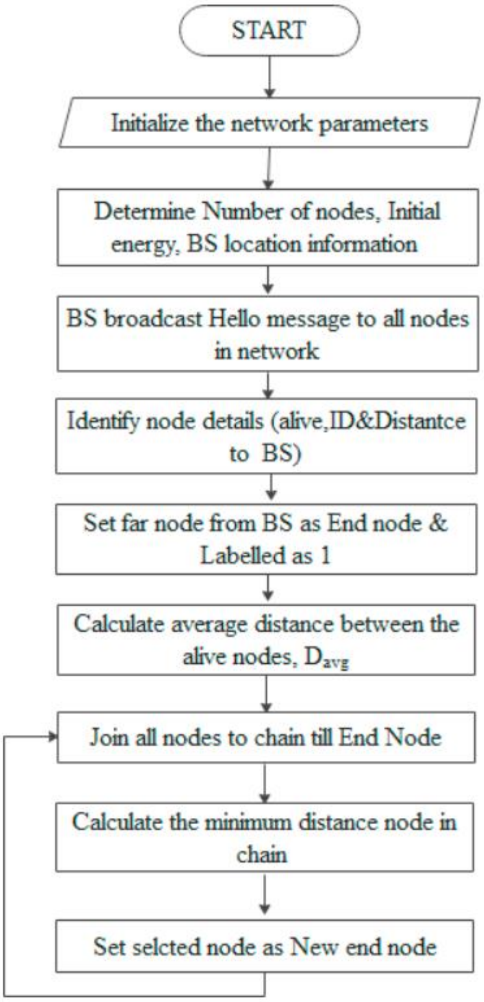


Figure 2.2 Chain Construction Algorithm as seen in [13]

During the Enhanced Leader Selection Phase, each chain selects a leader based on the weight of every node in the chain, which is calculated using the formula:

$W_j = RE_j / d_j$

W_j is the weightage of node j.

RE_j is the remaining energy budget of node j.

d_j is the distance between node j and the base station.

This is done to ensure that the same node is not constantly selected as the leader due to energy constraints.

After the chain leader has been selected, secondary leaders will be selected. If a node is closer to the base station than the chain leader, it is selected as a secondary leader. The secondary leader forwards all collected information directly to the base station instead of to the chain leader.

In the Data Transmission Phase, data collection occurs via the token passing method. The chain leader sends a token to the leaf node of a chain to start the data transmission. Starting from the leaf node, each node forwards its collected data to the next node in the chain. Each node performs data fusion before forwarding the data packets, in which data collected by self is combined with the data packet received from its children. Chain leaders collect data from all its chains before fusing them and forwarding them to the base station.

When all nodes have forwarded their data to the base station, the cycle repeats itself starting again at the Construction of Chain Phase.

Strengths

E-PEGASIS considers the remaining power budget of every node in the network. This is important for selecting leaders because chain leaders are sometimes responsible for collecting and forwarding data for multiple chains which can drain their energy budget very quickly. The weightage system of the E-PEGASIS protocol balances the energy load as evenly as possible, ensuring that each node lasts as long as possible.

E-PEGASIS reduces protocol overhead by performing data fusion before forwarding data packets. Instead of sending two packets each with its packet headers, each node combines them whenever possible to form one packet. This reduces the transmission time which in turn reduces the energy spent on the transmission and reception of data.

Weaknesses

The Construction of Chain Phase that is repeated for every cycle involves the base station broadcasting a Hello Message to all nodes in the network. The paper does not mention how this Hello message reaches nodes outside of the base station’s transmission range, so it is assumed that some nodes were assigned ahead of time to rebroadcast the Hello message. It is also not mentioned how the base station receives node details from each of the nodes in the network, so it is also assumed that all nodes reply to the Hello message with their own Hello message containing node details (alive, ID and distance to base station). Again, this might require multiple assigned forwarder nodes to forward the nodes’ Hello messages to the base station. This is very energy-inefficient for the forwarder nodes.

If a node in a chain dies or fails during the Data Transmission Phase, data from its children will not be forwarded to the base station because there is no contingency or alternative path to the base station. This can potentially lead to data loss or synchronization issues.

Critical Comments

The energy balancing done by the E-PEGASIS protocol to extend the battery life of every node in the network is very useful for ensuring a sensor network lasts as long as possible before battery replacements are required. However, the methods for constructing the chains which involve the base station broadcasting a Hello message and the children replying with the details of itself is very power inefficient for the nodes in charge of forwarding the messages. Instead of broadcasting the hello messages and receiving replies, the details of a node could be included in the data packets sent during the Data Transmission Phase.

The weakness of data loss and synchronization issues due to node failure could be addressed by using a mesh network topology. However, due to the algorithms used in the E-PEGASIS protocol which establishes a temporary tree topology, the implementation of alternative paths might be difficult.

The data fusion used in E-PEGASIS is a useful method in reducing the overhead of data packets sent, thereby reducing the time spent transmitting and receiving data. In the network protocol that will be designed in this project, a similar approach will be adopted at the individual node level via the caching of data. Nodes will not send data packets until it has collected enough data to fully occupy one data packet. This might introduce a lot of latency but saves energy by not spending time unpacking the received data packet, combining the data, and repacking the data packet.

2.3.2 A Fast, Adaptive, and Energy-Efficient Data Collection Protocol in Multi-Channel-Multi-Path Wireless Sensor Networks

Liew et al. [14] propose the Fast, Adaptive, and Energy-efficient Multi-channel-multi-path (FAEM) protocol which is built on top of the Basketball Net Topology (BNT). The BNT is a network topology that categorizes nodes into levels, with each node maintaining a connection table used for routing and Medium Access Control (MAC) scheduling. The level of a node is determined by the lowest level (lowest-numbered) node within its connection range. That lowest level node in range will be registered as the node's parent. It will then assign itself the level of its parent plus one. All nodes with one level lower than itself (level n-1) will be registered as the node's parents and recorded in the connection table. Similarly, nodes with the same level are registered as peers, and nodes one level higher than self (level n+1) are registered as children, all recorded in the connection table.

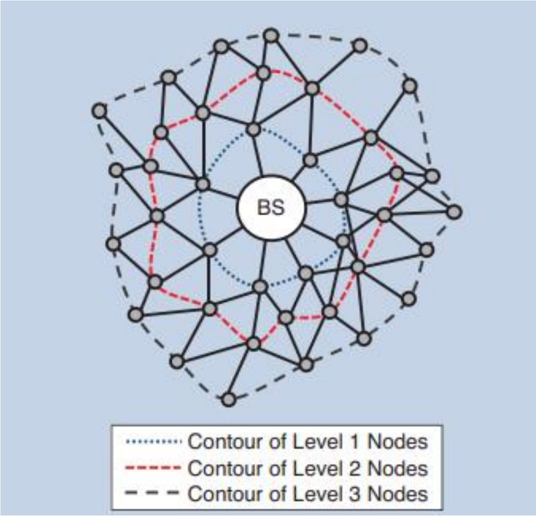


Figure 2.3 Example of BNT as seen in [14]

The FAEM protocol consists of two phases, the Node-Channel Assignment Phase, and the Scheduling and Packet Forwarding Phase.

In the Node-Channel Assignment Phase, every node is assigned a specific receiving channel that is different from those used by all its neighbours within two hops, to prevent interference during data transmission and reception. To initiate this phase, all nodes send their connection tables to the base station. Then, the base station uses all the connection tables to generate a two-hop neighbouring graph which shows the connections between a node and its neighbours, as well as connections for neighbours two hops away. Then, the base station will assign a unique receiving channel to each node which is not used by the node's one-hop and two-hop neighbours.

Upon completion of the Node-Channel Assignment Phase, the Scheduling and Packet Forwarding Phase can begin. One unit of time in this phase is called a cycle, which is made up of the Scheduling sub-cycle and the Packet Forwarding sub-cycle.

During the Scheduling sub-cycle, nodes negotiate with their parents and children to allocate or be allocated time slots for sending or receiving data, with the remaining unused time slots being used for sleeping. This sub-cycle can be further broken down into two parts, which are Iterative Matching as Parent (IM-P) and Iterative Matching as Child (IM-C).

During the IM-P, the node negotiates with its children to allocate time slots for receiving, while during the IM-C part, the node negotiates with its parents to be allocated time slots for sending data. Parents assign more time slots to children with larger data buffers. The order of IM-P and IM-C in the Scheduling sub-cycle depends on the level of the node. For example, level 1 nodes will spend the first half of the Scheduling sub-cycle performing IM-P and the second half performing IM-C, while level 2 nodes will do the opposite, spending the first half performing IM-C and the second half performing IM-P

The Packet Forwarding Phase consists of M frames and each frame contains N slots. Each slot represents a specific time slot in which a child node can transmit a packet to its parent node and receive an ACK in return. A child node waits until its assigned time slots to transmit data packets to the parent that assigned the time slots, and the parent is also ready to receive data packets during the assigned time slots. The same time slot assignments are repeated in each frame until the start of the next cycle.

Strengths

The FAEM protocol offers higher throughput compared to contention-based MAC protocols such as Carrier-sense multiple access with collision avoidance (CSMA/CA). Although the Scheduling sub-cycle within the Scheduling and Packet Forward phase is carried out using CSMA/CA to avoid collisions, it produces a collision-free data transmission and reception schedule that nodes can use to efficiently transmit data. The higher throughput of the FAEM protocol can be attributed to the collision-free data transmissions and reduced overhead during transmission. CSMA/CA must reserve a transmission medium by sending RTS and CTS packets, during which it is vulnerable to collisions resulting in backoffs, and then complete the data transmission with an ACK packet. On the other hand, FAEM has already reserved time slots for data transmission in advance, so no RTS and CTS packets are required. Nodes just send data packets during their assigned time slot and receive an ACK packet in reply.

The BNT provides redundancy for each node through the connection table. Each node can have multiple parents which means there are multiple paths that a data packet can travel through to reach the base station. This makes the network more resilient to node failures.

The FAEM protocol provides load balancing for data transmission. When a child node broadcasts a Request packet during the Scheduling sub-cycle, it receives replies in the form of a Grant packet from each of its parent nodes. This Grant packet contains a parent node's offered time slots. The child node can accept multiple Grant packets from different parent nodes, spreading the load across multiple parents instead of overburdening only one parent.

Weaknesses

The FAEM does not contain any mechanisms for "hot joining", a term used in games where players are allowed to join an ongoing game, or "hot plugging". For the FAEM protocol, once the network is initialized and the Node-Channel Assignment Phase is complete, there is no way to add new nodes to the network without first shutting down the whole network.

Critical Comments

The BNT and FAEM protocols contain key features that fulfil the objectives of this project, which is the design of a network protocol that is energy-efficient and reliable. Additionally, it provides higher throughput than other protocols which improves its overall performance. Thus, some protocols and algorithms designed in this project will be based on the BNT and FAEM protocols. For example, the network-layer protocol will use the concept of connection tables from BNT, and the scheduling protocol of the data link layer will use the FAEM protocol's Scheduling and Packet Forwarding phase.

However, the FAEM protocol has some weaknesses, namely the lack of "hot joining" mechanisms and the use of a less common wireless standard. These must be addressed to make it more suitable in the context of deployment in an agriculture IoT network.

2.3.3 Low-Power, End-to-End Reliable Collection using Glossy for Wireless Sensor Networks

Suzuki et al. [15] propose Choco, a low-power, end-to-end reliable collection protocol for WSN, which is built on Glossy, an efficient flooding scheme utilizing precisely timed constructive interference for reliable delivery.

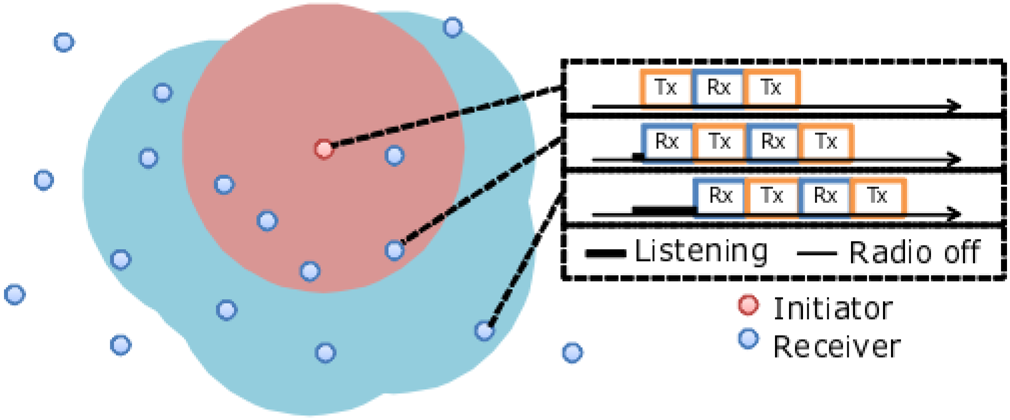


Figure 2.4 Example of Glossy flooding as seen in [15]

Nodes with Glossy retransmit received packets as soon as they are received, with no backoff. Every node that is not asleep will participate in the pattern of transmitting and receiving until each node has reached its number of relays, a predefined value. This entire cycle is called a Glossy phase. Participating nodes are classified as either the initiator, a single node which starts the transmission of their packet, and receivers, other nodes which turn on their radios and wait to receive the packet. The initiator sends their packet and the alternating transmit and receive repeats until the number of relays is reached. Since all nodes with Choco will participate in the Glossy phase, transmissions can be carried out in both the upstream and downstream directions, without the need for a complicated multi-hop routing protocol.

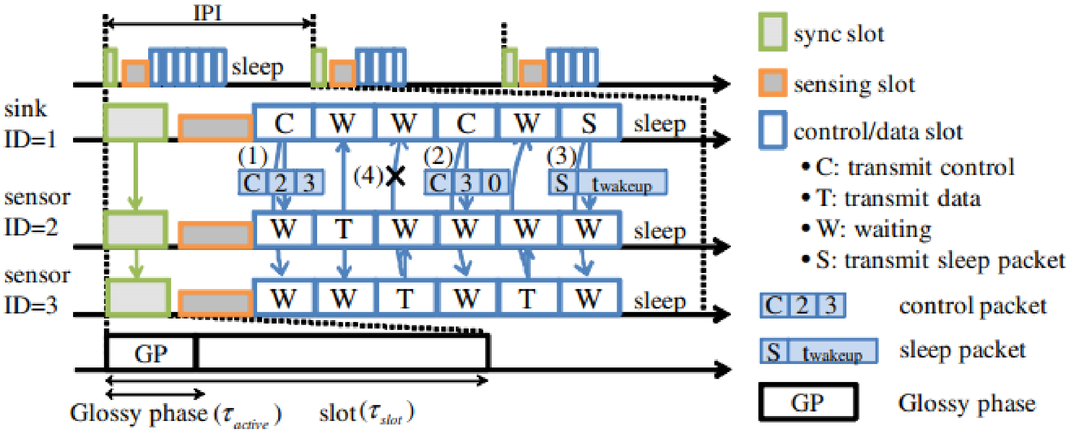


Figure 2.5 Choco slot assignment as seen in [15]

The Choco protocol communicates in slots. The sink node periodically performs time synchronization by broadcasting a time synchronization packet using Glossy. Before starting the communication cycle, the sink node transmits a control packet which contains the transmission schedule. The schedule contains information about which nodes are allowed to transmit at allocated slots. The number of slots contained within the control packet is determined by the maximum packet length set by Glossy. After the control packet is transmitted, the nodes specified in the control packet will initiate a Glossy phase during their assigned slot, transmitting their data packets towards the sink node.

The number of slots the sink node assigns to a slot can be determined by traffic and packet loss, where packet loss is determined by Glossy whenever the sink node does not receive a packet during a scheduled slot. Traffic is predictable in the case of WSNs and thus can be set as a predefined rate configured in the base station. By using predictable traffic and packet loss detection, the sink node can reliably schedule slots for transmissions and retransmissions until all nodes have sent in their data to the base station. Then, the sink node will send out a sleep packet informing all nodes when to sleep and when to wake up. Nodes using Choco can only sleep if it has received at least one sleep packet. The implication of this is that nodes that never receive any sleep packets will keep their radios on, causing a lot of the node’s energy reserves to be wasted idling. Although not illustrated in the diagram, Choco avoids this problem by transmitting the sleep packet multiple times before the actual sleep period begins.

Strength

Choco provides reliable data collection through the scheduling of retransmissions and the constructive interference of Glossy. Any packet loss detected by Glossy during a node's assigned slot will be scheduled for a retransmission until the sink node has received all available data from its nodes. The constructive interference caused by Glossy's floods will boost the signal strength of transmission at the receivers, making it more resistant to destructive interference from background sources.

Choco provides reliable downstream (base station to node) without the need for complicated multi-hop routing protocols. Since all nodes participate in the Gloss phase of packet transmission, all nodes will be involved in transmitting the packet from its source to its destination. This eliminates the possibility of a node's buffer overflowing and discarding the packet or a node crashing with a packet in its buffer, which is a problem in some multi-hop WSN protocols.

Weakness

Choco relies on centralized scheduling from the sink node. Each node's energy consumption depends on both its transmission time and its duty cycle (time spent active divided by time per cycle). As more nodes are added to the network, the sink node needs to schedule more transmissions per cycle. Since all nodes are involved in the Glossy phase, each node needs to participate in more transmission and receiving of data packets. Besides that, each node spends more time active per cycle since it takes a longer time for all data packets to be delivered to the sink node, only after which the sleep packet will be transmitted.

Choco does not contain any mechanism that allows it to adapt to dead nodes which are nodes that have crashed due to errors or a lack of energy. This means that it is possible that node crashes will cause the sink node to continuously schedule for retransmissions which have to be flooded through Glossy and keep all nodes in the network active all the time since the sink node has not collected all the data from its nodes, meaning that it cannot send out a sleep packet. This causes all nodes in the WSN to waste a lot of energy idling and transmitting control packets.

Critical Comments

The Choco on Glossy protocol uses the Glossy protocol to improve reliability for both upstream and downstream transmissions is very useful in small networks. However, its lack of scalability and adaptability make it unsuitable for large WSNs, where there is a reasonably high chance that at least one node will crash or fail in between node battery replacements.

Another possible scenario not addressed by the literature is that in large networks, it is possible that the transmission delay of a few microseconds due to the varying distance between nodes could be amplified the further it has to travel. This might cause the multiple simultaneous transmissions to be offset from each other by a few hundred microseconds. Instead of causing constructive interference at the receiver, multiple offset transmissions may cause inter-symbol interference, which makes it difficult for the receiver to interpret the signals correctly.

2.4 Concluding Remarks

In this chapter, a comprehensive review of existing literature on WSNs was carried out, with a focus on energy efficiency and reliability in data collection. Three data collection protocols, E-PEGASIS, FAEM and Choco on Glossy, were analysed. Each of the analysed protocols offered its own novel approach to improving energy efficiency and reliability.

E-PEGASIS improves energy efficiency by increasing the payload per packet sent, which in turn increases the payload-to-overhead ratio, resulting in lower energy consumption for packet overhead. However, its chain-based nature makes it possible that the token and data from all children nodes could be lost when a node fails, and the vagueness of certain mechanisms such as the broadcasting of Hello Messages makes it unsuitable for real-world scenarios.

The FAEM collision-free transmission and reception which is achieved by the Scheduling sub-cycle reduces the energy spent on retransmissions. This in combination with its mesh-like topology makes it a reliable and energy-efficient protocol and is the basis on which this project's new network protocol will be built.

The Choco on Glossy protocol uses a unique flooding mechanism which utilizes constructive interference to boost signal strength at receivers. However, the centralized nature of its scheduling and flooding nature makes it unsuitable for larger WSNs typically found in the agriculture industry.

Based on the findings of the literature review, the new network protocol will be largely based on the FAEM protocol with additional improvements such as a hot-joining protocol which allows new nodes to join an initialized network. Ideas such as the data fusion from E-PEGASIS will also be utilized in the new network protocol to increase the payload-to-overhead ratio.

Chapter 3

System Model

3.1 Topology

The network will be modelled and tested using multiple topologies, each consisting of 250 nodes. The topologies that will be used are:

i) Uniform Grid topology

This topology consists of nodes arranged in a grid formation with each node being equidistant to up to four neighbouring nodes.

ii) Random Even Distribution topology

This topology consists of nodes being distributed evenly within a 25 km^2 area centred around the base station.

iii) Random Uneven Distribution topology

This topology consists of nodes being distributed unevenly within a 25 km^2 area centred around the base station. This topology will result in some areas of high node density and some areas of low node density.

3.2 Network + Radio Model

The network will be configured as a mesh network. This helps enhance the reliability factor of the new network protocol because node failures can be circumvented easily. It also helps with load balancing, allowing each node to maximize its battery life.

The original plans for the radio model were to use the INET library's UnitDiskRadio and UnitDiskRadioMedium classes to simulate simple wireless transmission. Each node would have its own communication range and interference range, allowing each node to communicate with others within their communication range and interfere with transmissions between other nodes where the receiver is in range of the node's interference range.

However, due to the complexity of understanding, inheriting and modifying the built-in classes, a simplified pseudo-wireless radio model was devised. During the simulation setup, the program iterates through each node and calculates its distance from every other node in the simulation. If the distance of the node to another is less than or equal to the communication range, a standard wired connection is established between the two nodes, allowing for communication to be carried out. If the distance of the node to another is greater than the communication range but within the interference range, a separate wired connection will be established between the two nodes, allowing for each node to interfere with each other during their respective transmissions. To simulate wireless transmissions, nodes will send packets through all their wired connections, both communication and interference connections. All these wired connections will be assigned a DatarateChannel class to assign their bitrate, propagation delay, and bit error rate. Note that the DatarateChannel class does not have any relation to the wireless channels mentioned in the following paragraph.

Since the network protocol needs to simulate multiple wireless channels and interference, a channel value is added to every packet before transmission. When a node starts to receive a transmission, it will check the channel value attached to the packet and compares it to its current listening channel, the channel which is currently listening on. If the packet's channel value is equal to the node listening channel, the node will accept the packet transmission and process the packet when the transmission is complete, otherwise, it will allow the transmission to complete and drop the packet. However, if there is already an ongoing transmission on a specific channel and another transmission is received on the same channel, all ongoing transmissions on that channel will be marked with a collision flag, causing the node to discard the collided packets during packet processing.

3.3 Battery + Power Consumption Model

The battery model used in the modelling of the new network protocol keeps track of the residual energy charge left in each node, which can be measured in Coulombs (C). The battery will keep the node running until its residual energy charge reaches zero after which the node will crash. This battery model is not affected by real-world battery characteristics such as self-discharge and temperature dependence since the focus of the battery model is to track the energy consumption of the new network protocol.

The power consumption model used is a state-based energy consumption model. The node will place a current drain, measured in Milliamps (mA), on its battery depending on the current state of its "radio". The node consumes the most current when its radio is transmitting and receiving data, but other radio states such as idling and sleeping also consume a small amount of current.

The formula to calculate the remaining charge of a node is as follows:

$$E_{remaining} = E_{prev} - (I_{prev} \times (t_{current} - t_{prev}) \times V)$$

where:

$E_{remaining}$ is the remaining energy charge of a node

E_{prev} is the energy charge of a node recorded during the previous poll

I_{prev} is the current draw (milliamps) of the previous radio state

$t_{current}$ is the current timestamp (s, simulation time)

t_{prev} is the timestamp of the previous poll

Figure 3.1 Formula for the Calculation of Remaining Charge in A Node

Every time a node changes its radio state due to transmissions and scheduling timings, the above formula will be run to deduct the energy charge spent since the last poll.

3.4 Reliability Model (Node Failure Model + Packet Loss Model)

The node failure model used in the modelling of the new network is a global node failure controller. This controller will use a bell curve to cause random nodes to fail. The x-axis of the bell curve is the time component of the network simulation, and the y-axis represents the probability of node failure. Every X seconds, the node failure controller will use the bell curve's probability at the current time to determine if a node should fail. If the calculation determines that a node is to fail, the node failure controller will set a random working node to fail and crash.

The packet loss model used in modelling the new network is a constant bit error rate. This means that every bit in the data transmission frame has an X% chance of being flipped, turning the packet into an erroneous packet. This model is used as opposed to a simple packet loss rate model because it tests if the new network protocols approach of aggregating multiple data entries in one packet is more efficient compared to the more common method of sending data as soon as it is produced or ready.

3.5 Concluding Remarks

In this chapter, the various models that will be used to model the new network protocol have been explained.

Three topologies that will be tested, which are the uniform grid, random even distribution, and random uneven distribution will be used to test the new network protocol’s performance under different network conditions.

The pseudo-wireless radio model will be used to simulate a multi-channel wireless communication with interference.

The battery model and power consumption model focus on the current draw imposed by the state of the node’s radio. This will be used to measure the energy usage of the new network model and to simulate the node running out of energy and crashing.

The global node failure controller will be used to model random failures in the network. The random node failures combined with node crashes from lack of energy can represent the real world’s random node failures and nodes running out of energy.

The bit error rate which was chosen as the packet loss model has a chance to cause every packet to be lost which is increased by the length of the packet in bytes. This is done to test whether the data aggregation implemented by the new network protocol will bring better energy efficiency and reliability.

Chapter 4

System Design

4.1 Extended FAEM

The new network protocol developed in this project is called Extended FAEM. It is loosely based on the energy-efficient network protocol Fast, Adaptive, and Energy-efficient Multi-channel-multi-path (FAEM) by Liew et al.

The new network protocol, which occupies the application layer, network layer and link layer of the OSI model, was originally planned to be compliant with the OSI model. However, due to the complex nature of layered protocols in OMNeT++, all the layers were combined into a single layer during implementation to simplify the coordination and data sharing between them.

Application Layer Protocol

The application layer protocol aggregates outgoing data entries into one single large payload. It collects data entries until the buffer exceeds a predefined limit or when a time limit is exceeded, after which it will send the data entries. This is done to make full use of the maximum segment size, resulting in a higher data-to-overhead ratio. In theory, this should mean that in total less time is spent sending the headers of a data frame, which in turn means that less energy is spent sending the data frame headers.

In addition to data aggregation, the application layer protocol also implements data trimming. The data entries that need to be sent are sorted based on the timestamp of the data entry and arranged accordingly into the segment of the data packet. Then, if the data entries are identified to be uniformly periodic, the application layer protocol will trim away all the timestamps except for the timestamp of the first two data entries. The missing timestamps are recalculated when the data packet reaches its destination. This helps increase the data density of each packet sent out, meaning that less energy is spent in sending out the same number of data entries.

Network and Link Layer Protocol

The network layer protocol and the link layer protocol have been combined into one simplified layer to reduce the size of the frame sent out. Instead of encapsulating the data packet inside a frame, the data packet that is sent out by the node only has one unified header. Through this unified header approach, the size of the header is significantly reduced using short type variables (2 bytes long) as unique node IDs, replacing the IP addresses and MAC addresses that would normally be used in a normal network.

Table 4.1 Combined layer header format

Destination Node ID (2 bytes)	Source Node ID (2 bytes)	Direction (1 byte)	Type (1 byte)	Length (2 bytes)	Header Checksum (2 bytes)	Sequence Number (2 bytes)

Network Layer Protocol

The network layer protocol simplifies the data routing process using a connection table system adapted from the FAEM's Basketball Net Topology. Most upstream transmissions (node to base station) do not require a complex routing system. The node only needs to send data to one of its parents, which it can find from its connection table. In this way, the bulky IP header, which occupies a minimum of 20 bytes, is replaced with a much slimmer header.

The connection table of the network layer protocol will keep information about each neighbouring node that the current node has established a connection with, including information such as Node ID and routing metrics.

Table 4.2 Example of connection table

Parent (Node Level - 1)	Node 1	Node 2	Node 6
Peer (Node Level)	Node 4	Node 8	
Child (Node Level + 1)	Node 5	Node 7	Node 4

The routing metrics will be used by the network layer protocol to determine which node is selected when forwarding packets in a particular direction, either upstream or downstream. This metric is affected by how often the neighbour node requests, accepts, or declines during the Scheduling sub-cycle of the link layer protocol.

Link Layer Protocol

The link layer protocol will handle the MAC of the BNT by implementing the Node-Channel assignment phase and Scheduling sub-cycle of the FAEM protocol in order to achieve collision-free forwarding during the Packet Forwarding sub-phase.

The Node-Channel Assignment Phase of the extended FAEM protocol works similarly to that of the FAEM protocol it is based on. All nodes will send their connection tables to the base station. When the base station has received all the connection tables, it will assign each node a receiving channel that is unique within a two-hop range of that node. This is done to minimize the interference caused by multiple nearby nodes using the same channels for the Packet Forwarding phase.

The Node-Channel Assignment phase was partially bypassed during the implementation: the base station has direct access to each of each node’s connection tables, bypassing the need for a multi-hop routing system to propagate each node’s connection table. This approach was chosen to avoid the high rate of collisions that resulted from a lack of CSMA/CA integration. The INET library’s CSMA/CA implementation had a strict implementation method which required the use of a fixed packet format, which was not followed during the development of this network protocol.

4.2 Node-Channel Assignment Phase

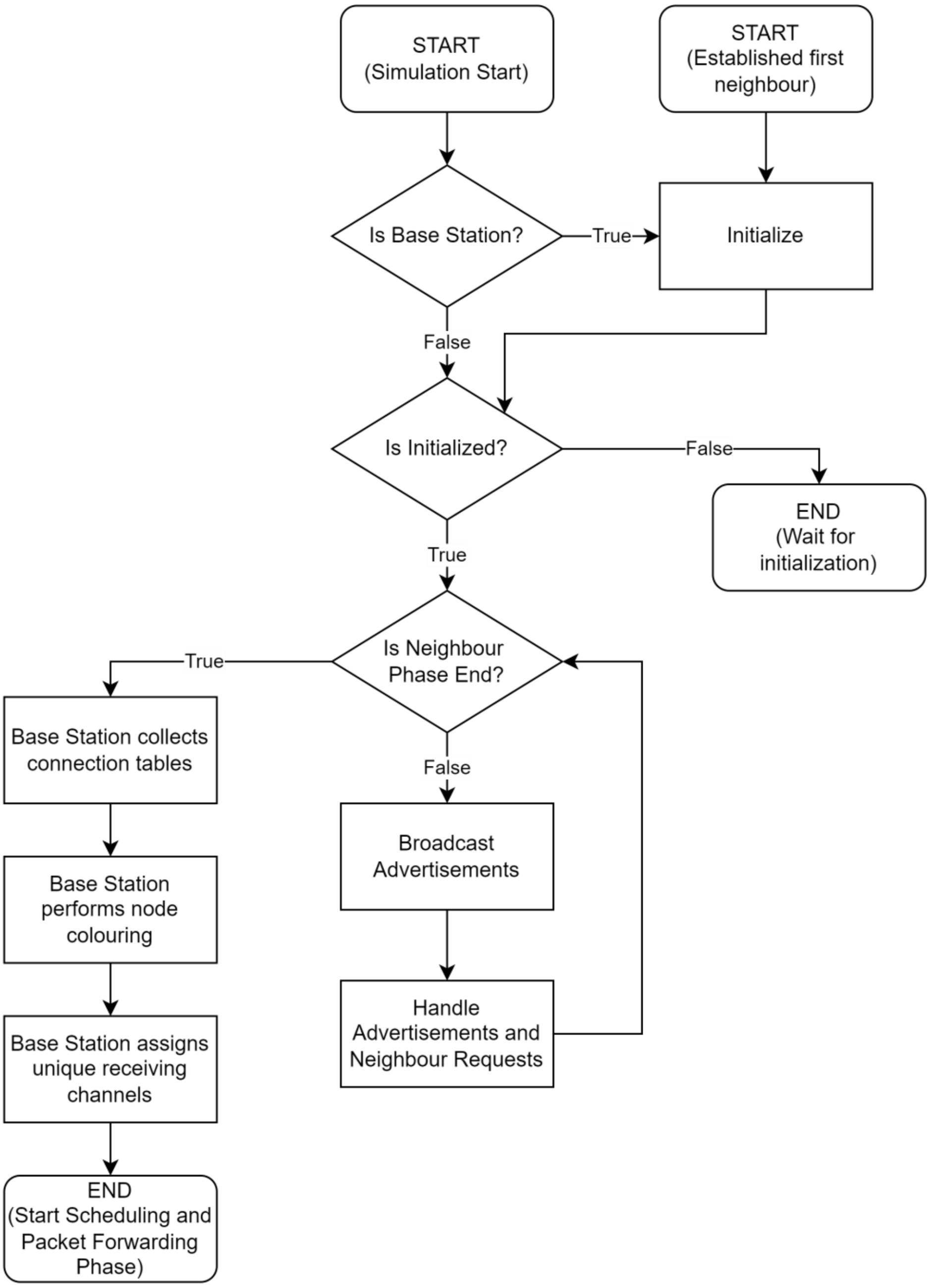


Figure 4.1 Flowchart of Node-Channel Assignment Phase

The above figure shows a summarized flow chart of the node-channel assignment phase. At the start of the simulation, each node checks whether it is the base station. If the node is a base station, it will initialize with base station starting parameters and start regularly broadcasting Advertisement packets as part of the node-channel assignment phase. Otherwise, the node will idle until it receives an Advertisement packet from an already initialized node.

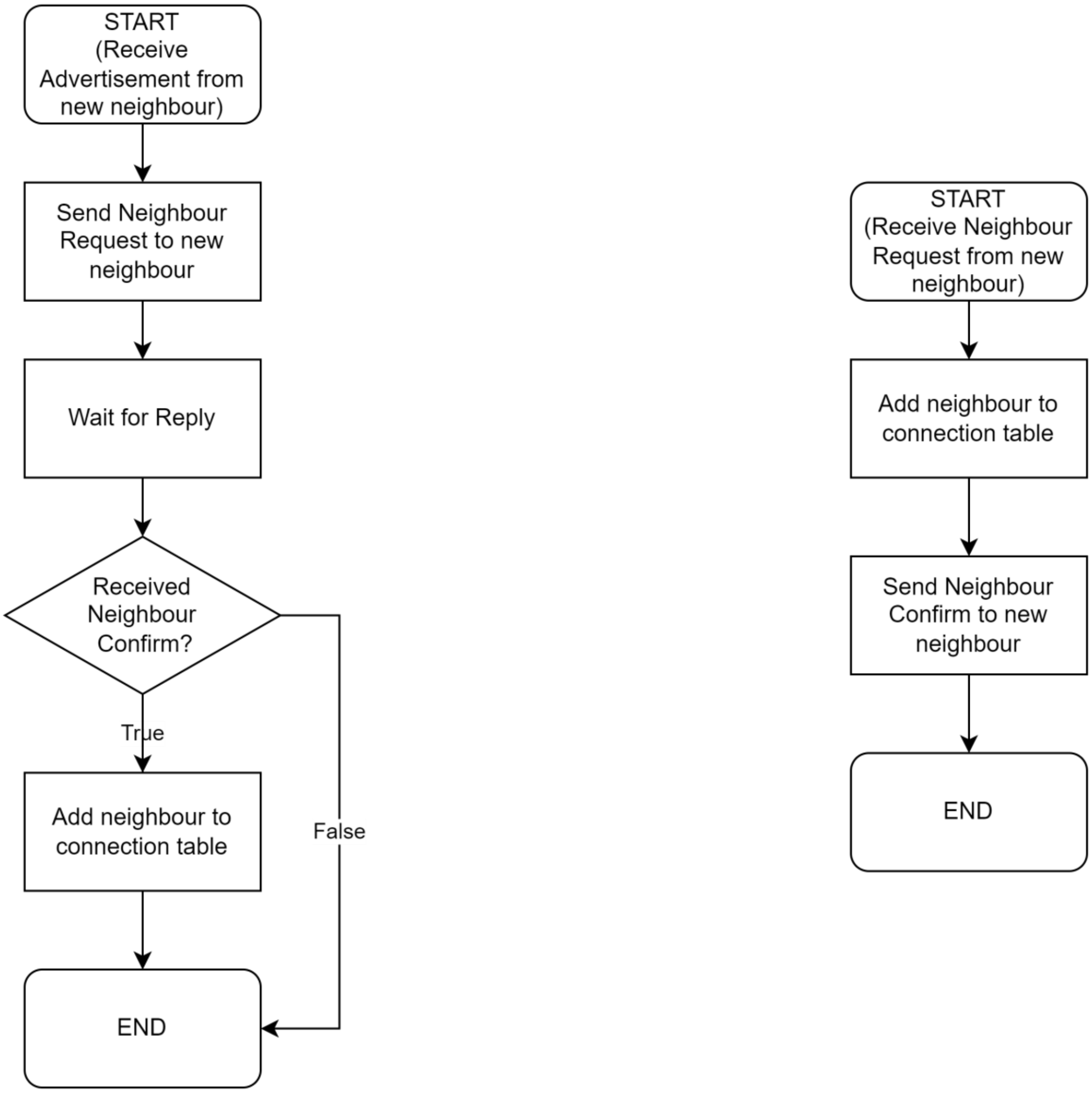


Figure 4.2 Flowchart of Establishing Neighbour Relation as Initiator (Left) and as Advertiser (Right)

The above diagram shows the process of establishing a neighbour relation with a new neighbour. Every initialized node will frequently broadcast Advertisement packets containing information about itself to nearby nodes. When a node discovers a new neighbour which they have not recorded in the connection table through these Advertisement broadcasts, the node will send out a Neighbour Request packet to that new neighbour with information about itself.

When the advertising node receives a Neighbour Request packet, it will add a new entry into its connection table based on the information received, and reply with a Neighbour Confirm packet. Upon receiving the Neighbour Confirm packet, the node will record the new neighbour into its connection table and continue with the node-channel assignment phase. If the establishment of the neighbour relation results in a node’s first neighbour, it will initialize and start broadcasting Advertisement packets. Initialized nodes will start collecting data samples.

The node-channel assignment phase ends when a synchronized node-channel assignment phase timer expires. The base station will collect the connection tables from all nodes, perform node colouring, and then assign a two-hop unique receiving channel to each node. Then, the first Scheduling and Packet Forwarding phase begins.

4.3 Scheduling and Packet Forwarding Phase

The extended FAEM's Scheduling and Packet Forwarding phase consists of the Scheduling sub-cycle and the Packet Forwarding sub-cycle. This phase is repeated until the end of the simulation.

Table 4.3 Example of Level 1 node's Scheduling and Packet Forwarding Phase

Duration (ms)	t/2	t/2	3	3	3	3	3	0.005	3
Action	IM-P	IM-C	RDAT	RDAT	RDAT	RDAT	RDAT	GUARD	S_ACK
Slot	N/A		1	2	3	4	5	N/A	N/A
Session	N/A		1						

Table 4.4 Example of Level 2 node's Scheduling and Packet Forwarding Phase

Duration (ms)	t/2	t/2	3	3	3	3	3	0.005	3
Action	IM-C	IM-P	SDAT	SDAT	SDAT	SDAT	SDAT	GUARD	R_ACK
Slot	N/A		1	2	3	4	5	N/A	6
Session	N/A		1						

Actions

IM-C – Iterative Matching as Child
IM-P – Iterative Matching as Parent
SDAT – Send Data Packet
RDAT – Receive Data Packet
S_ACK – Send Acknowledgement
R_ACK – Receive Acknowledgement
GUARD – Guard Space between Data Sending and ACK

Table 4.3 and Table 4.4 represent examples of the timing of Scheduling and Packet Forwarding phase, which starts with the Scheduling sub-cycle (IM-C and IM-P).

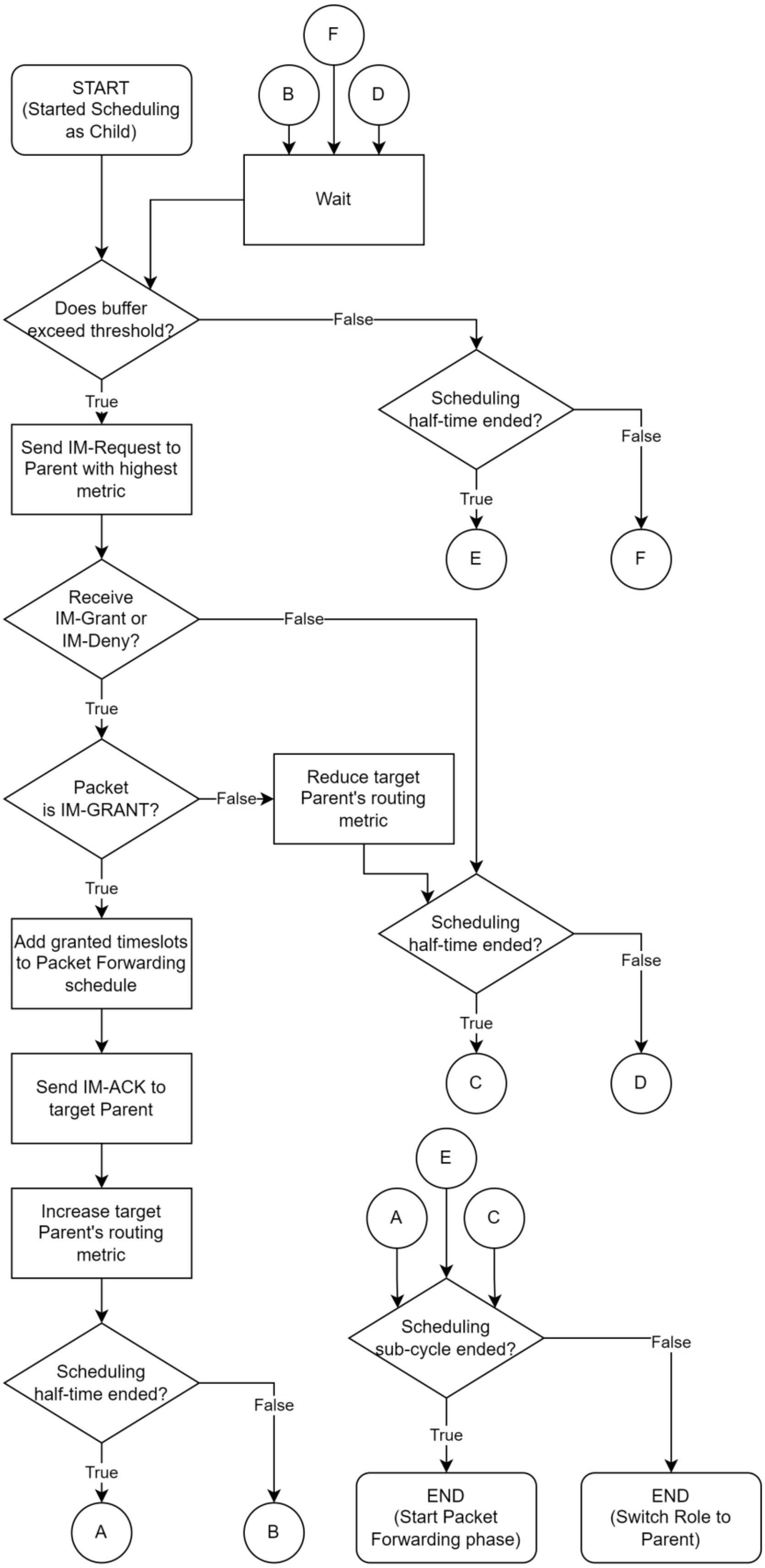


Figure 4.3 Scheduling Sub-cycle as Child

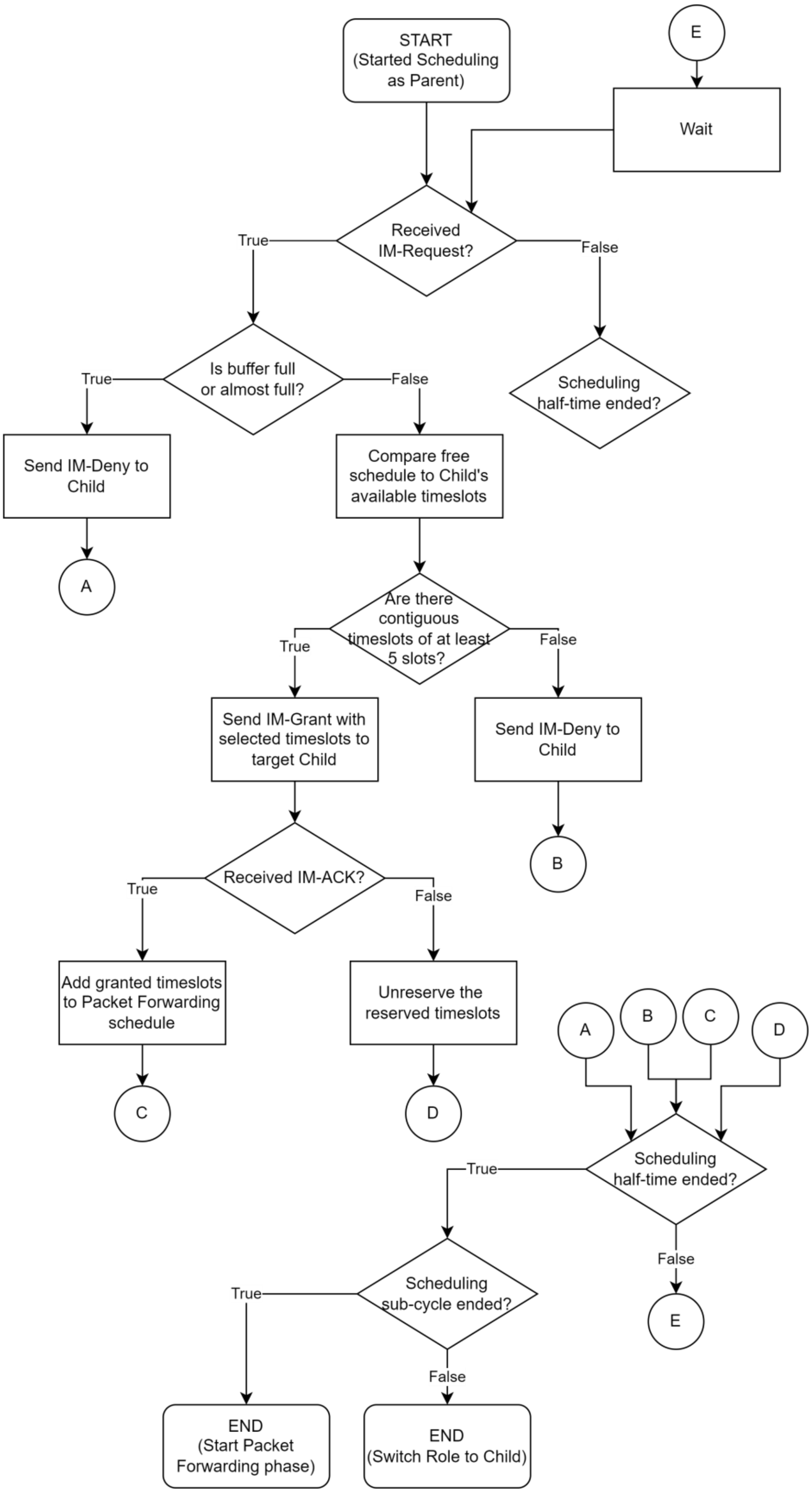


Figure 4.4 Scheduling Sub-cycle as Parent

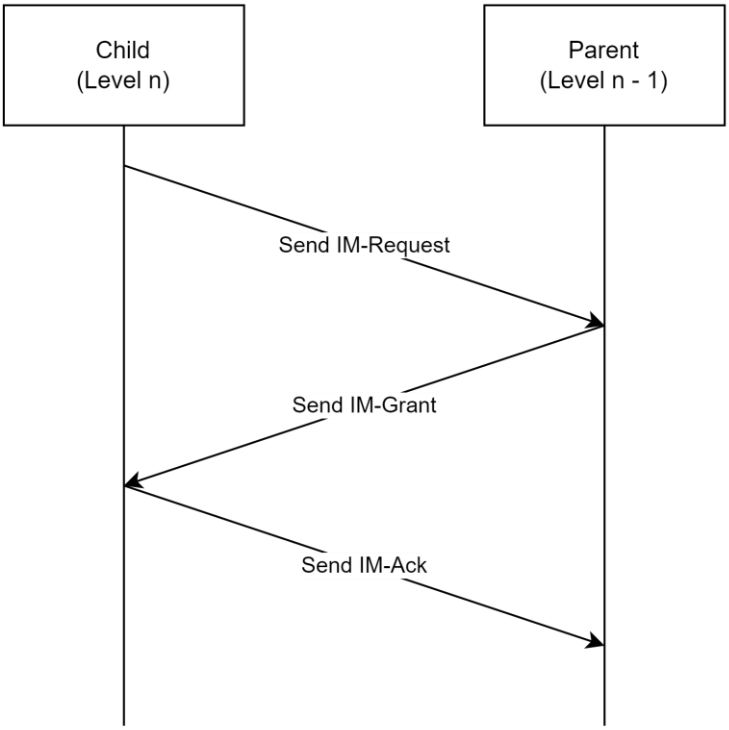


Figure 4.5 Example of Successful Iterative Matching Exchange

During the Scheduling sub-cycle, each node will spend half the time performing the Iterative Matching as Child stage and the other half performing the Iterative Matching as Parent stage. The duration of the Scheduling sub-cycle is predefined as duration t.

While performing the Iterative Matching as Child, a node with buffer exceeding a threshold percentage will send an IM-Request message to one of its Parent nodes. The IM-Request message contains its Node ID, the Parent's Node ID, the size of its packet buffer and the indices of the two longest contiguous timeslots in its schedule where it is free. Based on the node's connection table, the Parent node with the highest routing metric will be selected as the target.

Upon receiving the IM-Request packet, the Parent node will check if it has a sufficient free buffer. If it does not have sufficient free buffer, it will reply to the Child node with an IM-Deny message. The Child node receiving the IM-Deny packet will reduce the routing metric of the target Parent node.

If it has sufficient free buffer, the Parent node will compare the available timeslots of the Child node to its own schedule to get the largest two contiguous timeframes where both the Parent and Child nodes are free. The Parent node will reduce the length of contiguous timeslots based on the Child node's current buffer and its own free buffer. Contiguous timeslots shorter than five timeslots are ignored for energy efficiency, as shorter contiguous timeslots have a higher overhead (ACK packets) to data ratio. The Parent node will then send an IM-Grant message which contains its Node ID, the child node's Node ID, its receiving channel, and the indices of the allocated contiguous timeslots for the Child node.

The Child node that has received an IM-Grant message will reserve the timeslot in its next Packet Forwarding sub-cycle to send packets to the parent that has granted it and respond to the IM-Grant message by sending back an IM-ACK message. The Parent node will in turn reserve the timeslots returned in the IM-ACK message. However, if the Parent node waits too long without any replies, it will deallocate the timeslots that were tentatively reserved in the IM-Grant message.

Node\Timeslot	1	2	3	4	5	6	7	8	9	10
Node1 (level 1)	Sp	Sp	Sp	Sp	Sp	Rd	Rd	Rd	Rd	Sa
Node2 (level 2)	Rd	Rd	Rd	Rd	Sa	Sd	Sd	Sd	Sd	Ra
Node3 (level 3)	Sd	Sd	Sd	Sd	Ra	Sp	Sp	Sp	Sp	Sp

Table 4.5 Example Schedule for Packet Forwarding Sub-cycle

Keywords
Sp – Sleep
Sd – Sending Data
Rd – Receiving Data
Sa – Sending Ack
Ra – Receiving Ack

During the Packet Forwarding sub-cycle, every node will periodically check its schedule for the current timeslot. If the node has been scheduled to send data, it will change its radio state to transmitting and send out a data packet to the target node. Likewise, if the node has been scheduled to receive data, it will change its radio state to receiving and be ready to receive a data packet from the target node. If a node has nothing scheduled for a particular timeslot, it will put its radio to sleep state to save energy.

The Packet Forwarding sub-cycle is carried out in sessions. A session refers to the contiguous timeslots agreed upon by the Child and Parent nodes during the Scheduling sub-cycle. A session will begin with the Child node sending data to its Parent node and this is repeated until the second last timeslot in the session. During the last timeslot, the Parent node will send an ACK packet back which acts as acknowledgement for the session and negative acknowledgement for missing packets. The Child node will keep the data packets listed in the negative acknowledgement list and free up the buffer for the packets that were successfully sent.

4.4 Concluding Remarks

In this chapter, the multiple protocol layers that make up the new network protocol have been detailed and justified. The flow of the Node-Channel Assignment phase, Scheduling and Packet Forward sub-cycles have been illustrated in the form of flowcharts and schedule tables.

The Node-Channel Assignment phase will establish neighbour relations between nearby nodes as well as assign each node a unique receiving channel.

The Scheduling sub-phase carries out negotiations and scheduling to prepare a collision-free packet forwarding schedule while the Packet Forward sub-phase follows that schedule to ensure reliable data transmission.

Chapter 5

Simulation

5.1 Hardware Setup

The simulation was run on an Asus ROG Strix Laptop. However, the specification of the computer should have minimal impact on the timings and accuracy of the simulation. Since the OMNeT++ simulation program is single-threaded, the simulation speed is largely dependent on the processor’s single-core performance.

Table 5.1 Specifications of Laptop

Description	Specification
Model	Asus ROG Strix G512LU
Processor	Intel Core i7-10750H
Operating System	Windows 11
Graphic Card	NVIDIA GeForce GTX 1660 Ti
Memory	2*8GB DDR4 SODIMM RAM
Storage	1TB NVME SSD

5.2 Software Setup

The extended FAEM network protocol was developed and simulated in OMNeT++ using the INET framework.

Table 5.2 Specifications of Software

Description	Specification
Software	OMNeT++ 6.0.1
Framework	INET 4.5

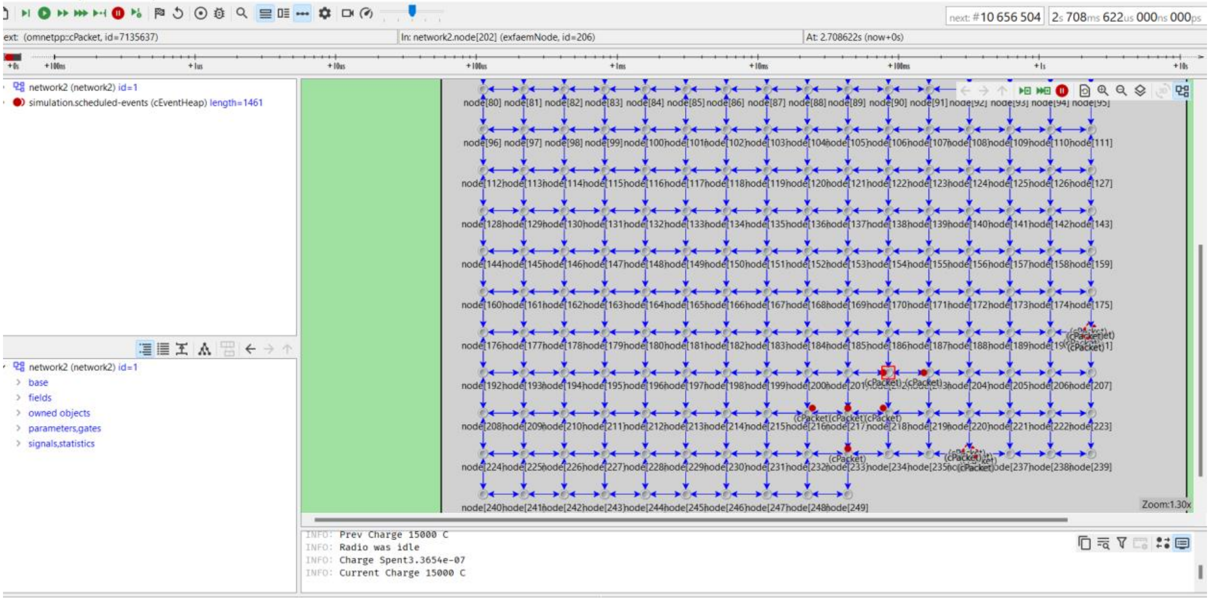
5.3 Simulation Settings

Table 5.3 Settings of Simulation

Parameter	**Value**	**Description**
Sample Size	30 bytes	The size of the data sample. Buffer is increased by this amount during data sampling.
Timestamp Size	7 bytes	The size of the timestamp for each data sample. The buffer is increased by this amount during data sampling. It can be omitted during transmission and recalculated at the receiving end.
Maximum Transmission Unit	128 bytes	Maximum size of packet payload.
Frame Header Size	12 bytes	The size of the header of a frame.
Rate of Data Sampling	2 per second	The rate at each data is sampled and the buffer is increased.
Byte Buffer Size	6400 bytes	The size of the byte buffer of each node.
Scheduling Buffer Threshold	5%	The threshold for a node to begin requesting during the scheduling sub-cycle.
BER	0.01%	Bit Error Rate
Radio Bitrate	250 Kbps	Transmission rate of radio
Transmission Range	50m	Radio transmission range
Interference Range	100m	Radio interference range
Transmit Current	17.4 mA	Current draw when transmitting a packet
Receive Current	18.8 mA	Current draw when receiving a packet
Idle Current	0.426 mA	Current draw when radio is idling
Sleep Current	0.021 µA	Current draw when radio is in sleep state
Battery Initial Charge Capacity	15000 C	The amount of charge the battery in the node starts with

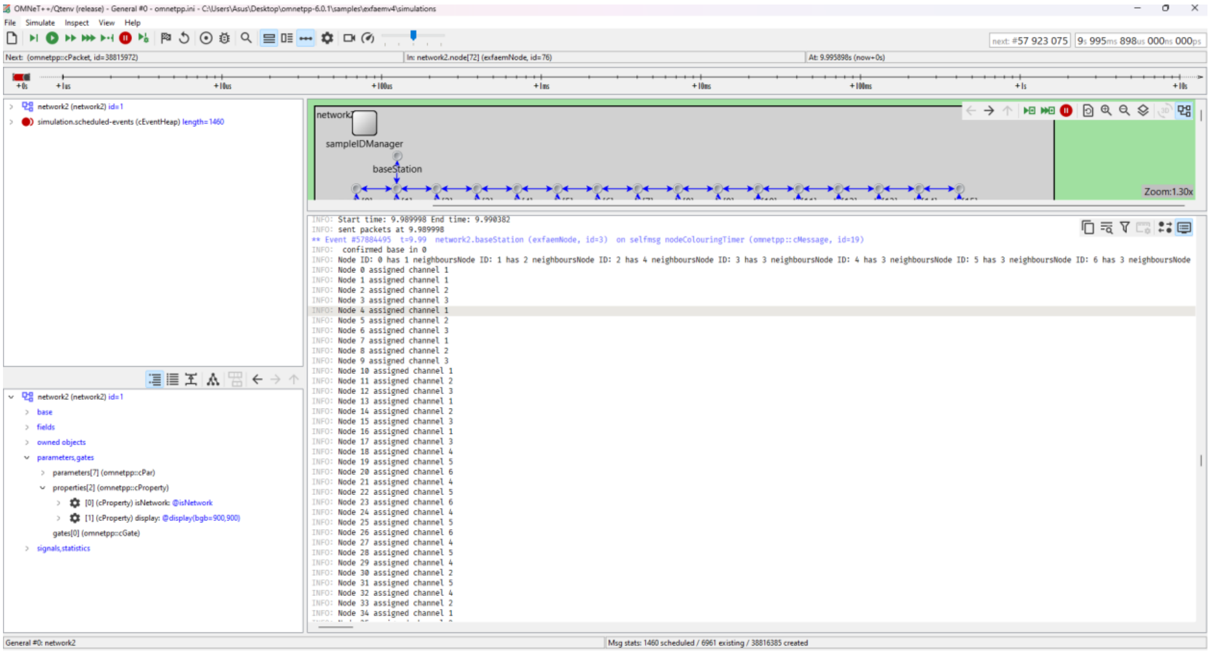
The sample size is based on the output values of an NPK sensor (three floats), temperature sensor (one float), soil pH sensor (one float), soil humidity sensor (one float) and air humidity sensor (one float). No actual raw data is being generated during data sampling.

5.4 Simulation Operation



```
INFO: Prev Charge 15000 C
INFO: Radio was idle
INFO: Charge Spent3.3654e-07
INFO: Current Charge 15000 C
```

Figure 5.1 Screenshot of Simulation in Operation



```
INFO: Start time: 9.989998 End time: 9.990382
INFO: sent packets at 9.989998
** Event #57884495  t=9.99  network2.baseStation (exfaemNode, id=3)  on selfmsg nodeColouringTimer (omnetpp::cMessage, id=19)
INFO:  confirmed base in 0
INFO: Node ID: 0 has 1 neighboursNode ID: 1 has 2 neighboursNode ID: 2 has 4 neighboursNode ID: 3 has 3 neighboursNode ID: 4 has 3 neighboursNode ID: 5 has 3 neighboursNode ID: 6 has 3 neighboursNode
INFO: Node 0 assigned channel 1
INFO: Node 1 assigned channel 1
INFO: Node 2 assigned channel 2
INFO: Node 3 assigned channel 3
INFO: Node 4 assigned channel 1
INFO: Node 5 assigned channel 2
INFO: Node 6 assigned channel 3
INFO: Node 7 assigned channel 1
INFO: Node 8 assigned channel 2
INFO: Node 9 assigned channel 3
INFO: Node 10 assigned channel 1
INFO: Node 11 assigned channel 2
INFO: Node 12 assigned channel 3
INFO: Node 13 assigned channel 1
INFO: Node 14 assigned channel 2
INFO: Node 15 assigned channel 3
INFO: Node 16 assigned channel 1
INFO: Node 17 assigned channel 3
INFO: Node 18 assigned channel 4
INFO: Node 19 assigned channel 5
INFO: Node 20 assigned channel 6
INFO: Node 21 assigned channel 4
INFO: Node 22 assigned channel 5
INFO: Node 23 assigned channel 6
INFO: Node 24 assigned channel 4
INFO: Node 25 assigned channel 5
INFO: Node 26 assigned channel 6
INFO: Node 27 assigned channel 4
INFO: Node 28 assigned channel 5
INFO: Node 29 assigned channel 4
INFO: Node 30 assigned channel 2
INFO: Node 31 assigned channel 5
INFO: Node 32 assigned channel 4
INFO: Node 33 assigned channel 2
INFO: Node 34 assigned channel 1
```

Figure 5.2 Screenshot of Event Logs indicating a Successful Node-Channel Assign Phase

The previous figures show the simulation environment in operation.

5.5 Implementation Issues and Challenges

It is difficult to understand the INET framework. The INET framework consists of many interconnected classes, which made picking up and modifying the classes a significant learning challenge. A lot of time had been spent following the layered protocol standard and the standard host class, but a working prototype compliant with those concepts could not be created, resulting in the development of a simplified protocol which deviated from the OSI model and INET normalcy.

It is difficult to replicate or emulate the performance and functionality of modern network protocols. The scope and objectives of this project involve benchmarking against and improving certain aspects of existing network protocols, but due to a lack of experience and deviation from network protocol norms, it became hard to compare the new protocol to existing ones.

It was difficult to balance simplicity and realism. At many points during the development of this network protocol, excessive effort was being put into developing code which was realistic to real-world scenarios, resulting in significant delays.

The network protocol project was initially developed on OMNeT++ 6.0.0, which had a bug which caused build errors. The researcher was unaware of this at the time and had spent considerable time debugging their code and fixing the IDE, leading to future delays.

Some of the scope and objectives of the project could not be achieved due to certain design compromises made during the development of the project. Features such as the “hot joining” mechanism could not be implemented because the simulation does not handle raw data, and therefore, multi-hop upstream and downstream routing could not be fully replicated.

5.6 Concluding Remark

In this chapter, the software, hardware and simulation setup have been detailed for potential replication. The challenges encountered during the implementation of the simulation have also been discussed in this chapter.

Chapter 6

System Evaluation and Discussion

6.1 Simulation Results

Due to the unconventional structure of the new network protocol, no accurate comparisons could be made with other energy-efficient network protocols. Instead, network protocol parameters were varied to test how varying parameters affected the reliability of transmission and energy efficiency of the network protocol.

Grid Topology

Packet Loss

Table 6.1 Packet Loss Result

MTU	37 bytes	128 bytes	256 bytes
Percentage of Packet Lost	7.10%	14.54%	10.60%

Packet loss is determined by the total number of packets dropped by all nodes divided by the total number of non-broadcast packets sent. In this simulation, only the packets sent and received during the Packet Forwarding sub-cycle will be tracked. Due to the simplicity of the backoff mechanism implemented in the protocol, it results in a high number of collisions and causes nodes to experience starvation. As a result, interference and statistics from the Scheduling sub-cycle will be disregarded.

Header Overhead Ratio

Table 6.2 Header Overhead Ratio Calculation

MTU	37 bytes	128 bytes	256 bytes
Header Overhead ratio	32.43%	9.38%	4.69%

The header overhead ratio is a calculated value of header size (12 bytes) divided by the maximum size of the packet. This indicates how much energy is used for sending the header compared to the data portion of the packet.

Sample Loss

Table 6.3 Sample Loss Result

MTU	37 bytes	128 bytes	256 bytes
Percentage Samples Lost	99.34%	97.85%	94.30%

Percentage of samples lost is quantified by the percentage of samples that were permanently lost in transmission, lost in dead nodes or buffer limitations. This is the key statistic for defining data collection reliability.

Throughput

Table 6.4 Throughput Result

MTU	37 bytes	128 bytes	256 bytes
Base Station Throughput (samples per second)	0.193	0.580	1.547

Throughput is quantified by taking the number of samples received by the base station at the end of the simulation divided by the simulation time (300s).

Energy Consumption

Table 6.5 Energy Consumption Result

MTU	37 bytes	128 bytes	256 bytes
Average Energy Consumption	0.2283C	0.1953C	0.1852C

Average energy consumption is the key metric to determining energy efficiency and the value is retrieved from each node by deducting its final energy charge from the starting energy charge. Note that the values are extremely low, likely due to flaws in the energy consumption model or the short time of simulation (300s).

6.2 Result Analysis

The packet loss statistics show that the larger the packet the more likely it is to be dropped. This is true due to the simulation’s Bit Error Rate model where every bit has a 0.01% chance to flip to become a faulty bit, causing the packet to be corrupted and dropped. However, the 256-byte MTU simulation deviates from the trend, likely because the lower number of packets being sent was not being statistically significant enough to follow the law of large numbers.

The header overhead ratio calculation shows that even though smaller packets encounter less packet loss, they end up spending more energy on transmitting packet headers. The 128-byte MTU simulation has about double the packet loss of the 37-byte MTU simulation, but the smaller 37-byte MTU simulation uses roughly four times the energy sending packet headers as the 128-byte MTU simulation.

The sample loss for all simulations is extremely high due to the low throughput of the network protocol. By the end of the simulation, most samples were still stuck inside the nodes and could not make it to the base station in time.

The throughput result shows that nodes which were able to send more samples per packet during a single timeslot achieved a higher throughput in terms of samples per second.

The energy consumption of the simulations confirms the fact that larger packets tend to save more energy. However, it can be inferred that the optimum packet size is dependent on the error rate of the simulation. A larger MTU increases the likelihood that the packet becomes faulty and discarded, resulting in the need to spend more energy on retransmissions.

6.3 Project Challenges

The researcher encountered problems with poor time management, leading to a last-minute rush to complete the final year project. This has led to a notable decrease in the quality of the final deliverable compared to the initial plans.

The researcher chose to take on an overly ambitious project. The researcher's area of specialization lies in communication and networking. However, most of the teaching materials in the course focused on the networking aspect rather than the communication aspect, which made the project more challenging.

The researcher faced challenges due to limitations in expertise. The researcher's main field of study was communications and networking. Although programming was taught as part of the course materials, networking remained the primary focus. This meant that although the researcher had knowledge of programming concepts like software engineering principles and object-oriented programming, which are concepts heavily applied to the OMNeT++ simulator and INET library, they had little practical experience outside of guided tutorials and assignments.

The researcher had encountered significant delays resulting from technical debt. Due to the previously mentioned expertise limitations, the OMNeT++ project accrued significant technical debt over the course of its development, resulting from messy implementation and a lack of structured planning. This led to significant delays to fix the problems.

6.4 Objectives Evaluation

Objective 1:

To design a new energy-efficient network protocol. The network protocol must have the same level of energy efficiency as the FAEM protocol without certain weaknesses such as the lack of “hot joining” mechanisms.

Result:

Partial success, the project produced a functioning network protocol with a focus on energy efficiency. However, the project was not able to achieve the ambitious objective of matching existing network protocols developed by industry experts. The proposed improvements such as “hot joining” mechanisms could not be implemented. The data trimming method has been shown to be a promising way of compressing data to be sent through packets.

Challenges:

Lack of expertise and practical experience in the development of network protocols and software led to significant delays and bad design choices.

Lessons Learned:

Future projects must have more realistic goals and detailed planning to ensure that objectives can be met.

Objective 2:

To improve the reliability of a WSN in terms of data transmissions and network topology. The number of retransmissions needs to be low to save energy and ensure as much data as possible reaches the base station. Redundancy in the network topology is to reduce the effects of single-node failures on the network.

Result:

Partial success, node-to-node transmission proved to be somewhat reliable. However, the low throughput of the network protocol resulted in severe data loss. The reliability of network topology could not be verified.

Challenges:

Timings and order of communication in a complex system are hard to manage and lead to many different unforeseen interactions.

Lessons Learned:

Timings and collisions are very important factors that need to be considered when improving energy efficiency.

6.5 Concluding Remark

This chapter discusses the results of the simulation and their implications. The results are not fully reliable due to the improper simulation settings such as the simulation time and packet forwarding intervals, which caused certain results to be too similar or statistically insignificant. The simulations should have been run again with newer values, but this was not possible due to time constraints for completing this report.

An evaluation of the challenges encountered during the development of the network protocol and the success of project objectives are also made.

Chapter 7

Conclusion and Recommendation

7.1 Conclusion

This project aimed to develop a reliable, energy-efficient network protocol loosely based on the FAEM protocol by Liew et al.

It aimed to incorporate elements of both the FAEM and E-PEGASIS protocols while also implementing mechanisms to address their weaknesses. The planned features were the inclusion of a "hot joining" mechanism and data trimming functionality.

The project achieved partial success. It was able to produce a new network protocol focused on energy efficiency but was unable to implement all the improvements. Only the data trimming feature was implemented. Benchmarking against existing network protocols unfeasible due to the unconventional design of the network protocol.

7.2 Recommendation

The new network protocol should be reworked in compliance with the OMNeT++ layered protocol standard and the OSI model. It makes it easier to be benchmarked with existing network protocols already present in the INET framework, while also allowing other networks to be simulated through the replacement of certain protocol layers.

The new network protocol should be implemented in a simulation capable of gathering and transmitting raw data. This would enable realistic simulation of upstream and downstream multi-hop routing, which makes it more feasible to implement and test the "hot joining" mechanism. It would also support creative uses of data trimming, such as trimming data that are similar in value to further reduce the byte size of individual data samples in a packet, thereby increasing data density without affecting packet size.

REFERENCES

[1] C. Sharma and N. K. Gondhi, “Communication Protocol Stack for Constrained IoT Systems,” in *2018 3rd International Conference On Internet of Things: Smart Innovation and Usages (IoT-SIU)*, Feb. 2018, pp. 1–6. doi: 10.1109/IoT-SIU.2018.8519904.

[2] M. Asemani, F. Abdollahei, and F. Jabbari, “Understanding IoT Platforms : Towards a comprehensive definition and main characteristic description,” in *2019 5th International Conference on Web Research (ICWR)*, Apr. 2019, pp. 172–177. doi: 10.1109/ICWR.2019.8765259.

[3] M. S. Punith, M. Nithya, and K. Deepa, “IoT Enabled Smart Fleet Management,” in *2022 IEEE 4th International Conference on Cybernetics, Cognition and Machine Learning Applications (ICCCMLA)*, Oct. 2022, pp. 256–260. doi: 10.1109/ICCCMLA56841.2022.9989097.

[4] X. Shi, T. Baba, D. Osagawa, M. Fujishima, and T. Ito, “Maturity Assessment: A Case Study toward Sustainable Smart Manufacturing Implementation,” in *2019 IEEE International Conference on Smart Manufacturing, Industrial & Logistics Engineering (SMILE)*, Apr. 2019, pp. 155–158. doi: 10.1109/SMILE45626.2019.8965284.

[5] S. S. Hajam and S. A. Sofi, “IoT-Fog architectures in smart city applications: A survey,” *China Communications*, vol. 18, no. 11, pp. 117–140, Nov. 2021, doi: 10.23919/JCC.2021.11.009.

[6] M. R. M. Kassim, “IoT Applications in Smart Agriculture: Issues and Challenges,” in *2020 IEEE Conference on Open Systems (ICOS)*, Nov. 2020, pp. 19–24. doi: 10.1109/ICOS50156.2020.9293672.

[7] M. Kocakulak and I. Butun, “An overview of Wireless Sensor Networks towards internet of things,” in *2017 IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC)*, Jan. 2017, pp. 1–6. doi: 10.1109/CCWC.2017.7868374.

[8] A. Raj and D. Steingart, “Review—Power Sources for the Internet of Things,” *J. Electrochem. Soc.*, vol. 165, no. 8, pp. B3130–B3136, 2018, doi: 10.1149/2.0181808jes.

[9] A. M. K. Abdulzahra, A. K. M. Al-Qurabat, and S. A. Abdulzahra, “Optimizing energy consumption in WSN-based IoT using unequal clustering and sleep scheduling methods,” *Internet of Things*, vol. 22, p. 100765, Jul. 2023, doi: 10.1016/j.iot.2023.100765.

[10] D. Kandris, C. Nakas, D. Vomvas, and G. Koulouras, “Applications of wireless sensor networks: An up-to-date survey,” *Applied System Innovation*, vol. 3, no. 1, p. 14, Feb. 2020. doi:10.3390/asi3010014

[11] F. Kiani and A. Seyyedabbasi, “Wireless Sensor Network and internet of things in Precision Agriculture,” *International Journal of Advanced Computer Science and Applications*, vol. 9, no. 6, 2018. doi:10.14569/ijacsa.2018.090614

[12] T. Ojha, S. Misra, and N. S. Raghuwanshi, “Wireless Sensor Networks for agriculture: The state-of-the-art in practice and future challenges,” *Computers and Electronics in Agriculture*, vol. 118, pp. 66–84, Oct. 2015. doi:10.1016/j.compag.2015.08.011

[13] S. Sadhana, E. Sivaraman, and D. Daniel, “Enhanced Energy Efficient Routing for Wireless Sensor Network Using Extended Power Efficient Gathering in Sensor Information Systems (E-PEGASIS) Protocol,” *Procedia Computer Science*, vol. 194, pp. 89–101, 2021, doi: 10.1016/j.procs.2021.10.062.

[14] S.-Y. Liew, C.-K. Tan, M.-L. Gan, and H. G. Goh, “A Fast, Adaptive, and Energy-Efficient Data Collection Protocol in Multi-Channel-Multi-Path Wireless Sensor Networks,” *IEEE Comput. Intell. Mag.*, vol. 13, no. 1, pp. 30–40, Feb. 2018, doi: 10.1109/MCI.2017.2773800.

[15] M. Suzuki, Y. Yamashita, and H. Morikawa, “Low-power, end-to-end reliable collection using glossy for wireless sensor networks,” *2013 IEEE 77th Vehicular Technology Conference (VTC Spring)*, Jun. 2013. doi:10.1109/vtcspring.2013.6692624

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y4T1	**Study week no.: 2**
Student Name & ID: Christopher Wong Jin Soon 2001539	
Supervisor: Ts Dr Goh Hock Guan	
Project Title: RELIABLE LOW POWER NETWORK PROTOCOL IN IOT FOR AGRICULTURE FIELD	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

Fix the broken demo leftover from FYP1

2. WORK TO BE DONE

Implement Scheduling
Implement Packet Forwarding
Integrate wireless transmission

3. PROBLEMS ENCOUNTERED

4. SELF EVALUATION OF THE PROGRESS

Behind in progress due to leftover work from FYP1

Goh

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y4T1	**Study week no.: 4**
Student Name & ID: Christopher Wong Jin Soon 2001539	
Supervisor: Ts Dr Goh Hock Guan	
Project Title: RELIABLE LOW POWER NETWORK PROTOCOL IN IOT FOR AGRICULTURE FIELD	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

Fix the broken demo leftover from FYP1

2. WORK TO BE DONE

Implement Scheduling
Implement Packet Forwarding
Integrate wireless transmission

3. PROBLEMS ENCOUNTERED

4. SELF EVALUATION OF THE PROGRESS

Still behind and stuck fixing the demo

Goh

Supervisor's signature

Cw

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y4T1	**Study week no.: 6**
Student Name & ID: Christopher Wong Jin Soon 2001539	
Supervisor: Ts Dr Goh Hock Guan	
Project Title: RELIABLE LOW POWER NETWORK PROTOCOL IN IOT FOR AGRICULTURE FIELD	

1. WORK DONE [Please write the details of the work done in the last fortnight.] **Implement Scheduling**
2. WORK TO BE DONE **Integrate wireless transmission**
3. PROBLEMS ENCOUNTERED
4. SELF EVALUATION OF THE PROGRESS **Behind schedule**

Goh

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y4T1	**Study week no.: 8**
Student Name & ID: Christopher Wong Jin Soon 2001539	
Supervisor: Ts Dr Goh Hock Guan	
Project Title: RELIABLE LOW POWER NETWORK PROTOCOL IN IOT FOR AGRICULTURE FIELD	

1. WORK DONE [Please write the details of the work done in the last fortnight.] **Implement Packet Forwarding** **Rework node structure to simpler format**
2. WORK TO BE DONE **Integrate wireless transmission**
3. PROBLEMS ENCOUNTERED **OMNeT++ node structure is very limiting**
4. SELF EVALUATION OF THE PROGRESS **Behind schedule**

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y4T1	**Study week no.: 10**
Student Name & ID: Christopher Wong Jin Soon 2001539	
Supervisor: Ts Dr Goh Hock Guan	
Project Title: RELIABLE LOW POWER NETWORK PROTOCOL IN IOT FOR AGRICULTURE FIELD	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

Integrate wireless transmission
Changed transmission to self-implemented pseudo-wireless communication

2. WORK TO BE DONE

Implement packet loss and collision
Adjust timings
Add backoff for timings

3. PROBLEMS ENCOUNTERED

INET Wireless integration is difficult and confusing

4. SELF EVALUATION OF THE PROGRESS

Behind schedule

Goh

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y4T1	**Study week no.: 12**
Student Name & ID: Christopher Wong Jin Soon 2001539	
Supervisor: Ts Dr Goh Hock Guan	
Project Title: RELIABLE LOW POWER NETWORK PROTOCOL IN IOT FOR AGRICULTURE FIELD	

1. WORK DONE

[Please write the details of the work done in the last fortnight.]

Implement packet loss and collision
Adjust timings
Add backoff for timings

2. WORK TO BE DONE

3. PROBLEMS ENCOUNTERED

Network protocol timings are very precise and hard to calibrate

4. SELF EVALUATION OF THE PROGRESS

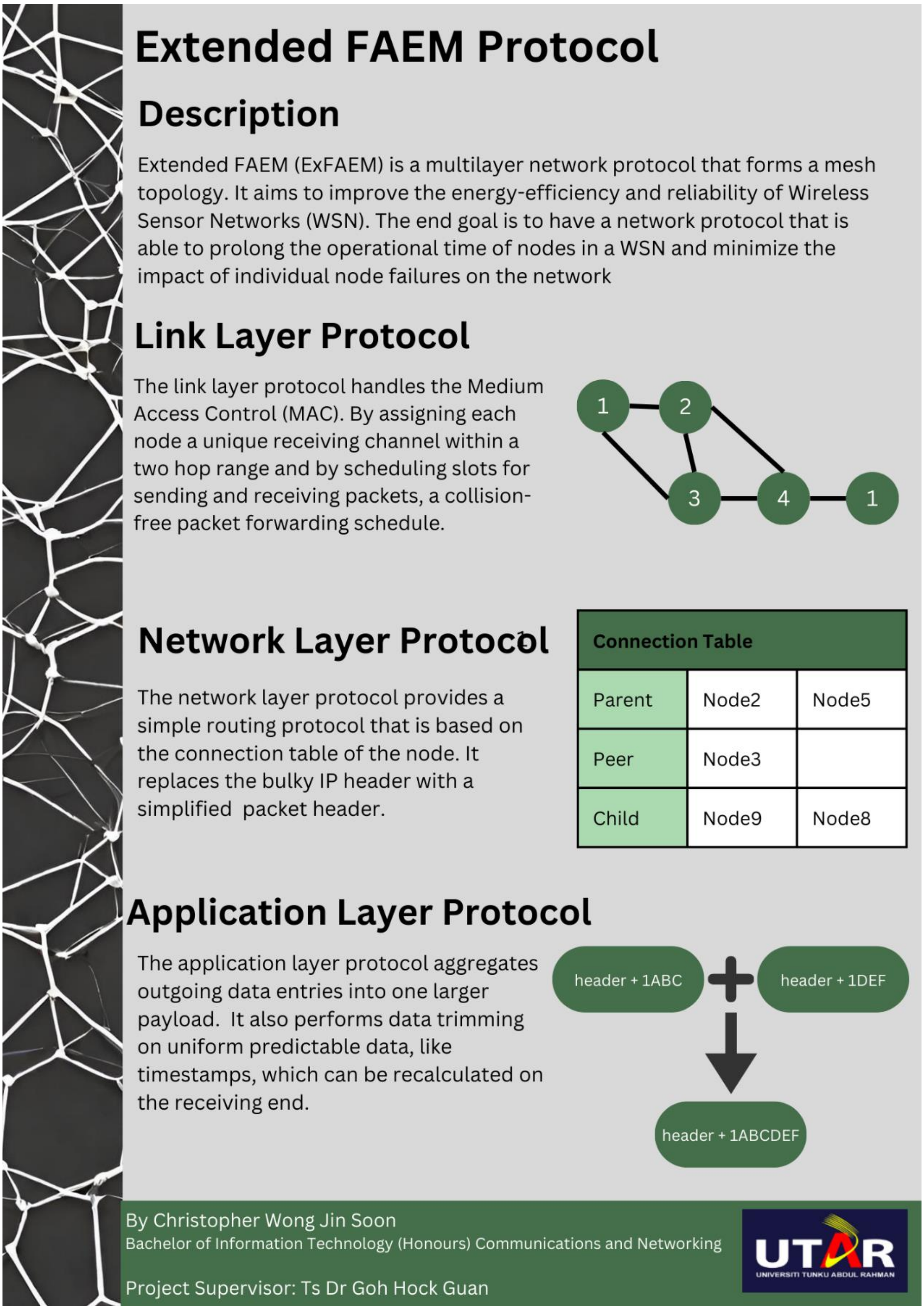
Still behind, need to rush the report

Goh

Supervisor's signature

Student's signature

Poster



Extended FAEM Protocol

Description

Extended FAEM (ExFAEM) is a multilayer network protocol that forms a mesh topology. It aims to improve the energy-efficiency and reliability of Wireless Sensor Networks (WSN). The end goal is to have a network protocol that is able to prolong the operational time of nodes in a WSN and minimize the impact of individual node failures on the network

Link Layer Protocol

The link layer protocol handles the Medium Access Control (MAC). By assigning each node a unique receiving channel within a two hop range and by scheduling slots for sending and receiving packets, a collision-free packet forwarding schedule.

Network Layer Protocol

The network layer protocol provides a simple routing protocol that is based on the connection table of the node. It replaces the bulky IP header with a simplified packet header.

Connection Table		
Parent	Node2	Node5
Peer	Node3	
Child	Node9	Node8

Application Layer Protocol

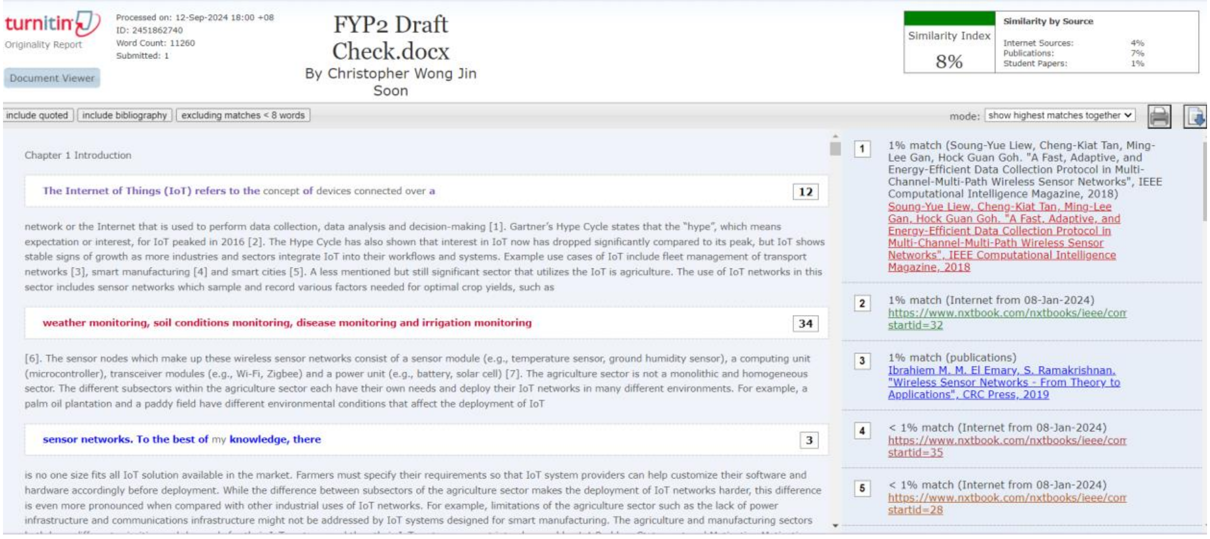
The application layer protocol aggregates outgoing data entries into one larger payload. It also performs data trimming on uniform predictable data, like timestamps, which can be recalculated on the receiving end.

By Christopher Wong Jin Soon
Bachelor of Information Technology (Honours) Communications and Networking

Project Supervisor: Ts Dr Goh Hock Guan

UTAR
UNIVERSITI TUNKU ABDUL RAHMAN

PLAGIARISM CHECK RESULT



Processed on: 12-Sep-2024 18:00 +08
ID: 2451862740
Word Count: 11260
Submitted: 1

FYP2 Draft Check.docx
By Christopher Wong Jin Soon

Similarity Index	Similarity by Source	
8%	Internet Sources:	4%
	Publications:	7%
	Student Papers:	1%

Chapter 1 Introduction

The Internet of Things (IoT) refers to the concept of devices connected over a 12

network or the Internet that is used to perform data collection, data analysis and decision-making [1]. Gartner's Hype Cycle states that the "hype", which means expectation or interest, for IoT peaked in 2016 [2]. The Hype Cycle has also shown that interest in IoT now has dropped significantly compared to its peak, but IoT shows stable signs of growth as more industries and sectors integrate IoT into their workflows and systems. Example use cases of IoT include fleet management of transport networks [3], smart manufacturing [4] and smart cities [5]. A less mentioned but still significant sector that utilizes the IoT is agriculture. The use of IoT networks in this sector includes sensor networks which sample and record various factors needed for optimal crop yields, such as

weather monitoring, soil conditions monitoring, disease monitoring and irrigation monitoring 34

[6]. The sensor nodes which make up these wireless sensor networks consist of a sensor module (e.g., temperature sensor, ground humidity sensor), a computing unit (microcontroller), transceiver modules (e.g., Wi-Fi, Zigbee) and a power unit (e.g., battery, solar cell) [7]. The agriculture sector is not a monolithic and homogeneous sector. The different subsectors within the agriculture sector each have their own needs and deploy their IoT networks in many different environments. For example, a palm oil plantation and a paddy field have different environmental conditions that affect the deployment of IoT

sensor networks. To the best of my knowledge, there 3

is no one size fits all IoT solution available in the market. Farmers must specify their requirements so that IoT system providers can help customize their software and hardware accordingly before deployment. While the difference between subsectors of the agriculture sector makes the deployment of IoT networks harder, this difference is even more pronounced when compared with other industrial uses of IoT networks. For example, limitations of the agriculture sector such as the lack of power infrastructure and communications infrastructure might not be addressed by IoT systems designed for smart manufacturing. The agriculture and manufacturing sectors

1 1% match (Soung-Yue Liew, Cheng-Kiat Tan, Ming-Lee Gan, Hock Guan Goh. "A Fast, Adaptive, and Energy-Efficient Data Collection Protocol in Multi-Channel-Multi-Path Wireless Sensor Networks", IEEE Computational Intelligence Magazine, 2018)
Soung-Yue Liew, Cheng-Kiat Tan, Ming-Lee Gan, Hock Guan Goh. "A Fast, Adaptive, and Energy-Efficient Data Collection Protocol in Multi-Channel-Multi-Path Wireless Sensor Networks", IEEE Computational Intelligence Magazine, 2018

2 1% match (Internet from 08-Jan-2024)
https://www.nxtbook.com/nxtbooks/ieee/con
startid=32

3 1% match (publications)
Ibrahiem M. M. El Emary, S. Ramakrishnan. "Wireless Sensor Networks - From Theory to Applications", CRC Press, 2019

4 < 1% match (Internet from 08-Jan-2024)
https://www.nxtbook.com/nxtbooks/ieee/con
startid=35

5 < 1% match (Internet from 08-Jan-2024)
https://www.nxtbook.com/nxtbooks/ieee/con
startid=28

Universiti Tunku Abdul Rahman			
Form Title : Supervisor's Comments on Originality Report Generated by Turnitin for Submission of Final Year Project Report (for Undergraduate Programmes)			
Form Number: FM-IAD-005	Rev No.: 0	Effective Date: 01/10/2013	Page No.: 1of 1

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY

Full Name(s) of Candidate(s)	Christopher Wong Jin Soon
ID Number(s)	20ACB01539
Programme / Course	Bachelor of Information Technology (Honours) Communications and Networking
Title of Final Year Project	RELIABLE LOW POWER NETWORK PROTOCOL IN IOT FOR AGRICULTURE FIELD

Similarity	**Supervisor's Comments** **(Compulsory if parameters of originality exceeds the limits approved by UTAR)**
Overall similarity index: 8 % **Similarity by source** Internet Sources: 4 % Publications: 7 % Student Papers: 1 %	
Number of individual sources listed of more than 3% similarity: 0	
Parameters of originality required and limits approved by UTAR are as Follows: **(i) Overall similarity index is 20% and below, and** **(ii) Matching of individual sources listed must be less than 3% each, and** **(iii) Matching texts in continuous block must not exceed 8 words** ***Note: Parameters (i) – (ii) shall exclude quotes, bibliography and text matches which are less than 8 words.***	

Note Supervisor/Candidate(s) is/are required to provide softcopy of full set of the originality report to Faculty/Institute

Based on the above results, I hereby declare that I am satisfied with the originality of the Final Year Project Report submitted by my student(s) as named above.

Signature of Supervisor

Name: Goh Hock Guan

Date: 13/9/2024

Signature of Co-Supervisor

Name: ______________________

Date: ______________________

UNIVERSITI TUNKU ABDUL RAHMAN

FACULTY OF INFORMATION & COMMUNICATION TECHNOLOGY (KAMPAR CAMPUS)

CHECKLIST FOR FYP2 THESIS SUBMISSION

Student Id	20ACB01539
Student Name	Christopher Wong Jin Soon
Supervisor Name	Ts Dr Goh Hock Guan

TICK (√)	**DOCUMENT ITEMS** Your report must include all the items below. Put a tick on the left column after you have checked your report with respect to the corresponding item.
√	Title Page
√	Signed Report Status Declaration Form
√	Signed FYP Thesis Submission Form
√	Signed form of the Declaration of Originality
√	Acknowledgement
√	Abstract
√	Table of Contents
√	List of Figures (if applicable)
√	List of Tables (if applicable)
	List of Symbols (if applicable)
√	List of Abbreviations (if applicable)
√	Chapters / Content
√	Bibliography (or References)
√	All references in bibliography are cited in the thesis, especially in the chapter of literature review
	Appendices (if applicable)
√	Weekly Log
√	Poster
√	Signed Turnitin Report (Plagiarism Check Result - Form Number: FM-IAD-005)
√	I agree 5 marks will be deducted due to incorrect format, declare wrongly the ticked of these items, and/or any dispute happening for these items in this report.

*Include this form (checklist) in the thesis (Bind together as the last page)

I, the author, have checked and confirmed all the items listed in the table are included in my report.

(Signature of Student)
Date: 12/9/2024