الجمهورية الجزائرية الديمقراطية الشعبية
PEOPLE'S DEMOCRATIC REPUBLIC OF ALGERIA
وزارة التعليم العالي و البحث العلمي
MINISTRY OF HIGHER EDUCATION AND SCIENTIFIC RESEARCH
جامعة عمار ثليجي بالأغواط
UNIVERSITY OF AMMAR TELIDJI LAGHOUAT

كلية العلوم
FACULTY OF SCIENCES
قسم الإعلام الآلي
DEPARTMENT OF COMPUTER SCIENCE

MASTER THESIS

Field : Mathematics and Computer Science

Option : Computer Science

Specialization : Distributed Networks, Systems, and Applications

Submitted by : DJAAFARI Sarra

Theme

Proposition and Simulation of LEACH-MQTT Routing Protocol for Publish/Subscribe Systems in IoT Networks

Jury members :

Mrs.	Atika SAHLAOUI	MC(B)	(University of Laghouat)	President
Ms.	Zohra ABDELHAFIDI	MC(A)	(University of Laghouat)	Examiner
Mr.	Lakhdar OULAD DJEDID	MC(B)	(University of Laghouat)	Examiner
Ms.	Narjes HAMINI	MC(B)	(University of Laghouat)	Advisor

Academic Year 2022/2023

الجـمهورية الجـزائرية الديمقراطـية الشـعبية
RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE
وزارة التعليم العـالي و البحث العـلمي
MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE SCIENTIFIQUE
جـامعة عمار ثـليجي بالأغـواط
UNIVERSITÉ AMMAR TELIDJI LAGHOUAT

كـليـة العلـوم
FACULTÉ DES SCIENCES
قـسـم الإعـلام الآلـي
DÉPARTEMENT D'INFORMATIQUE

THÈSE DE MASTER

Domaine : Mathématique et Informatique

Filière : Informatique

Option : Réseaux, Systèmes et Applications Répartis

Présenté par : DJAAFARI Sarra

Thème

Proposition et Simulation d'un Protocole de Routage LEACH-MQTT pour les systèmes de publish/subscribe dans les réseaux IoT

Jury :

Mme.	Atika SAHLAOUI	MC(B)	(Université de Laghouat)	President
Mlle.	Zohra ABDELHAFIDI	MC(A)	(Université de Laghouat)	Examinateur
Mr.	Lakhdar OULAD DJEDID	MC(B)	(Université de Laghouat)	Examinateur
Mlle.	Narjes HAMINI	MC(B)	(Université de Laghouat)	Encadreur

Année Universitaire 2022/2023

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Dedications

With heartfelt gratitude and profound appreciation,
I would like to dedicate this thesis,
To my loving mother and father, who were
always my true inspiration and guiding stars,
To my wonderful family for their patience,
unconditional love, enormous support,
and continuous encouragement.
To my extraordinary friends.
To all those who love me.
To all those I love.

For you all, this is forever yours!

Sarra Djaafari

Acknowledgments

First and foremost, no volume of words is enough to express praise to *Allah Almighty*, who gave me the life, the strength, the ability and the will to accomplish this thesis. All praise is due to Allah, the Lord of the Worlds.

I extend my deep gratitude towards my thesis supervisor Dr. Narjes HAMINI, for her encouragement, her guidance, and her advice which allowed me to carry out this thesis.

My sincere thanks go to the committee members : Dr. Amel BELABBACI, Dr. Atika SAHLAOUI, and Prof. Nasreddine LAGRAA, for accepting to review my humble work, spending their precious time to read it and giving me valuable comments and insightful feedback.

My deepest appreciation goes to my beloved parents, to my beautiful sisters, and to my cherished brothers, whose love, encouragement, and sacrifices have been always my driving force. Your unwavering support and belief in me have given me the strength to overcome challenges and pursue my goals and be the person I am today.

I address my warmest thanks to my dearest friends, for their constant support, for our deep conversations, and for our laughs that have kept me going through the highs and lows of this academic pursuit.

Lastly, I do not forget my gratitude to all the professors of the Computer Science Department at Laghouat University and every professor pass in my academic career, and thanks to everyone who contributed to the development of this work from near or far.

I will be forever grateful for all those who have made this journey possible.

Sarra Djaafari, June 2023

Proposition and Simulation of LEACH-MQTT Routing Protocol for Publish/Subscribe Systems in IoT Networks

DJAAFARI Sarra

Abstract

Nowadays, the increasing prevalence of Internet Of Things (IOT) networks has propelled the need for efficient communication protocols to handle the exchange of data in a scalable and reliable manner. Wireless Sensor Networks (WSNs) play a crucial role in collecting and disseminating data from sensor nodes to central entities in IOT networks. In this context, the reliable routing with reducing energy consumption of publish and subscribe messages in a publish/subscribe system, such as the Message Queuing Telemetry Transport (MQTT) system, poses a significant challenge. This thesis aims to propose a novel routing protocol specifically designed for IOT networks with a focus on Pub/-Sub systems. The primary objective of the protocol is to ensure the efficient routing of publish and subscribe messages among publishers, subscribers, and a central broker, with low energy consumption. To achieve this, the protocol incorporates the concept of utilizing cluster-heads in the Low Energy Adaptive Clustering Hierarchy Routing Protocol (LEACH) as broker bridges to facilitate communication with the central broker and also to balance the load on it. The proposed protocol addresses the challenges of finding the best routes to transmit publish and subscribe messages while considering factors such as energy consumption and scalability. Through simulations using the OMNet++ simulator and the Castalia framework, the protocol demonstrates superior performance in terms of reduced energy, network lifetime, and network stability period.

Keywords : IOT, WSN, publish/subscribe system, MQTT, LEACH, routing protocol, cluster-heads, simulation.

اقتراح ومحاكاة بروتوكول توجيه LEACH-MQTT لأنظمة النشر\الاشتراك في شبكات إنترنت الأشياء

سـارة جـعـفـاري

مـلـخـص

في الوقت الحاضر، أدى الانتشار المتزايد لشبكات إنترنت الأشياء (IOT) إلى زيادة الحاجة إلى بروتوكولات اتصال فعالة للتعامل مع تبادل البيانات بطريقة قابلة للتطوير وموثوق بها. تلعب شبكات الاستشعار اللاسلكية (WSN) دورا مهما في جمع ونشر البيانات من وحدات الاستشعار إلى الوحدات المركزية في شبكات إنترنت الأشياء. في هذا السياق، يمثل التوجيه الفعال مع تقليل استهلاك الطاقة لرسائل النشر والاشتراك في نظام النشر \ الاشتراك، مثل نظام نقل القياس عن بُعد في قائمة انتظار الرسائل (MQTT)، تحديا كبيرا. تهدف هذه الأطروحة إلى اقتراح بروتوكول توجيه جديد مصمم خصيصا لشبكات IOT مع التركيز على أنظمة Pub / Sub. الهدف الأساسي للبروتوكول هو ضمان التوجيه الفعال لنشر الرسائل والاشتراك فيها بين الناشرين والمشتركين والوسيط المركزي، مع استهلاك منخفض للطاقة. لتحقيق ذلك، يشتمل البروتوكول على مفهوم استخدام رؤوس المجموعات في بروتوكول توجيه التسلسل الهرمي التكيفي منخفض الطاقة (LEACH) كجسور وسيطة لتسهيل الاتصال مع الوسيط المركزي وأيضا لموازنة الحمل عليه. يعالج البروتوكول المقترح تحديات العثور على أفضل الطرق لنقل الرسائل المنشورة والاشتراك مع مراعاة عوامل مثل استهلاك الطاقة وقابلية التوسع. من خلال عمليات المحاكاة باستخدام محاكي ++OMNET وإطار عمل Castalia، يوضح البروتوكول أداءً فائقا من حيث انخفاض الطاقة وعمر الشبكة وفترة استقرار الشبكة.

الكلمات المفتاحية : IOT، WSN، نظام النشر \ الاشتراك، MQTT، LEACH، بروتوكول التوجيه، رؤوس المجموعات، المحاكاة.

Proposition et Simulation d'un Protocole de Routage LEACH-MQTT pour les systémes de publish/subscribe dans les réseaux IoT

DJAAFARI Sarra

Résumé

De nos jours, la prévalence croissante des réseaux de l'Internet des objets (IOT) a propulsé le besoin de protocoles de communication efficaces pour gérer l'échange de données de manière évolutive et fiable. Les réseaux de capteurs sans fil (WSN) jouent un rôle crucial dans la collecte et la diffusion de données depuis les nœuds de capteurs vers les entités centrales des réseaux IOT. Dans ce contexte, le routage fiable des messages de publication et d'abonnement avec une réduction de la consommation d'énergie dans un système de publish/subscribe, tel que le système Message Queuing Telemetry Transport (MQTT), pose un défi important. Cette thèse vise à proposer un nouveau protocole de routage spécialement conçu pour les réseaux IOT avec un accent sur les systèmes Pub/-Sub. L'objectif principal du protocole est d'assurer le routage efficace des messages de publication et d'abonnement entre les éditeurs, les abonnés et un broker central, avec une faible consommation d'énergie. Pour y parvenir, le protocole intègre le concept d'utilisation des chefs de cluster dans le protocole LEACH en tant que brokers ponts pour faciliter la communication avec le broker central et également pour équilibrer la charge sur celui-ci. Le protocole proposé relève les défis consistant à trouver les meilleurs itinéraires pour transmettre les messages de publication et d'abonnement tout en tenant compte de facteurs tels que la consommation d'énergie et l'évolutivité. Grâce à des simulations utilisant le simulateur OMNet++ et le framework Castalia, le protocole démontre des performances supérieures en termes de réduction d'énergie, de durée de vie du réseau et de période de stabilité du réseau.

Mots Clés : IOT, WSN, système publish/subscribe, MQTT, LEACH, protocole de routage, chefs de cluster, simulation.

Contents

List of Figures

List of Tables

List of Algorithms

List of Acronyms

ADC Analog to Digital Converters. 8

ADV Advertisement message. 35

AODV Ad hoc On-Demand Distance Vector. 14

BB Broker Bridge. 31, 36

BS Base Station. 5, 10, 13, 19, 25, 27

BSs Base Stations. 12

CB Central Broker. 30, 31, 44

CH Cluster-Head. 10, 16, 19, 25–27, 30, 33, 36, 44

CHs Clusters heads. 10, 18–20, 25, 27, 36, 44

CSMA/CA Carrier-sense multiple access with collision avoidance. 31, 35

DD Directed Diffusion. 15

DSDV Destination Sequenced Distance Vector. 14

DSR Dynamic Source Routing. 14

DT-MANET Delay-Tolerant Mobile Ad-hoc NETworks. 25

EAR Energy Aware Routing. 15, 16

FDN First Dead Node. 48

GPS Global Positioning System. 9, 16

GUI Graphical User Interface. D

HDN Half Dead Nodes. 48

IDE Integrated Development Environment. C, 27, 28

IOT Internet Of Things. iii, v, 1, 2, 4, 5, 21, 24, 25, 28, 30–32, 40, 42, 50, 51

ISO International Organization for Standardization. 12

LEACH Low Energy Adaptive Clustering Hierarchy Routing Protocol. iii, v, ix, 2, 13, 18, 20, 30, 32, 33, 44–46, 48, 49, 51, 54

MATLAB MATrix LABoratory. 28

MCFA Minimum Cost Forwarding Algorithm. 16

MQTT Message Queuing Telemetry Transport. iii, v, ix, 1, 2, 21, 25, 27, 30, 32, 44, 46, 50–52

NED Network Description Language. I, 41

OLSR Optimized Link State Routing. 14

OMNet++ Objective Modular Network Testbed in C++. iii, v, 30, 41, 43, 50, 51

OSI Open Systems Interconnection. 12

PEGASIS Power-Efficient Gathering in Sensor Information Systems. 54

Pub/Sub Publish/Subscribe. iii, v, 1, 46

QOS Quality Of Service. 15, 21

RSSI Received Signal Strength Indicator. 35

SAR Sequential Assignment Routing. 15

SEP Sequential Exchange Protocol. 13

SPIN Sensor Protocols for Information via Negotiation. 15, 16

SSL Secure Socket Layer. 52

TDMA Time Division Multiple Access. 20, 31, 35

TEEN Threshold Sensitive Energy Efficient Sensor Network Protocol. 13

TLS Transport Layer Security. 52

WSN Wireless Sensor Network. v, ix, 1, 2, 5, 7, 10, 12, 24, 25, 28, 30, 31, 40, 42, 51

WSNs Wireless Sensor Networks. iii, ix, 4, 6, 7, 12, 13, 20, 24, 30, 40, 51

Chapter 1

Introduction

Contents

1.1 Context

During the last years, the advent of Internet Of Things (IOT) has revolutionized various industries, enabling seamless connectivity and data exchange between a myriad of devices [1]. In IOT networks, Wireless Sensor Network (WSN) play a crucial role in collecting and disseminating data from sensor nodes to central entities [2]. Within this context, the publish/subscribe system stands out as a fundamental communication paradigm for efficient message routing in IOT networks. In particular, the Message Queuing Telemetry Transport (MQTT) protocol has emerged as a widely adopted standard for publish/subscribe messaging, offering scalability and reliability in transmitting data between publishers and subscribers [3]. One of the fundamental aspects of IOT communication is the efficient routing of publish and subscribe messages in a scalable and reliable manner.

1.2 Problem Statement and Objective

In this paper, we study the message routing problem in the context of a WSN-based IOT Pub/Sub service network. This research focuses on proposing a new routing protocol for a publish/subscribe system, specifically leveraging the MQTT (Message Queuing Telemetry Transport) protocol, to address the challenges associated with message routing between network nodes (subscribers and publishers).

Routing Publish/Subscribe messages between publishers, brokers, and subscribers present a significant challenge that necessitates thorough examination and resolution [4].

The objective of this study is to address this problem by identifying the most suitable and efficient route to exchange these messages while minimizing energy consumption.

Our aim is to developing a routing protocol based on the MQTT and LEACH protocols, for optimizing message routing process between publishers, subscribers, and a central broker.

1.3 Organization of the Thesis

The organization of this thesis is structured as follows :

- **Chapter two** (2) provides an overview of IOT, WSN, and routing in those networks, discussing existing routing protocols and their limitations.
- **Chapter three** (3) illustrates some relevant studies with a brief show to their advantages, drawbacks, and a concise comparison among them.
- **Chapter four** (4) presents the proposed routing protocol, detailing its architecture and message routing mechanisms, focusing on the performance evaluation of the protocol through extensive simulations. It also analyzes and discusses the results.
- Finally, **Chapter five** (5) concludes the thesis with a summary of findings and suggestions for future research.

Chapter 2

Generalities

Contents

2.1 Introduction

In the past few years, wireless sensor networks (WSNs) have been extensively increasing and widely applied in diverse IOT applications including real-time monitoring, remote sensing, security and surveillance, and precision agriculture a little more.

Along this chapter, we will introduce the generalities of wireless sensor networks (WSNs) and discuss their applications.

We will also delve into the architecture and protocol stack of WSNs. The chapter will describe the architecture of the sensor and the different network architectures, along with the protocol stack used in those networks. Furthermore, we will discuss routing in WSNs, including the classifications and characteristics of hierarchical routing protocols and the low energy adaptive clustering hierarchy routing protocol.

The chapter will also explore the publish/subscribe protocol and the constraints that affect WSNs. Finally, we will conclude the chapter by summarizing the key points discussed.

2.2 Wireless Sensor-IOT Networks

Internet Of Things (IOT) is a concept that has recently gained popularity. IOT refers to a network of physical objects or things equipped with sensors, electronics, circuits, software and connectivity capabilities, enabling them to collect and exchange data with other devices over the internet, so that each device have an ID (IP address) [5]. These objects can range from everyday consumer devices like house equipment and vehicles to industrial infrastructure (Figure. 2.1).

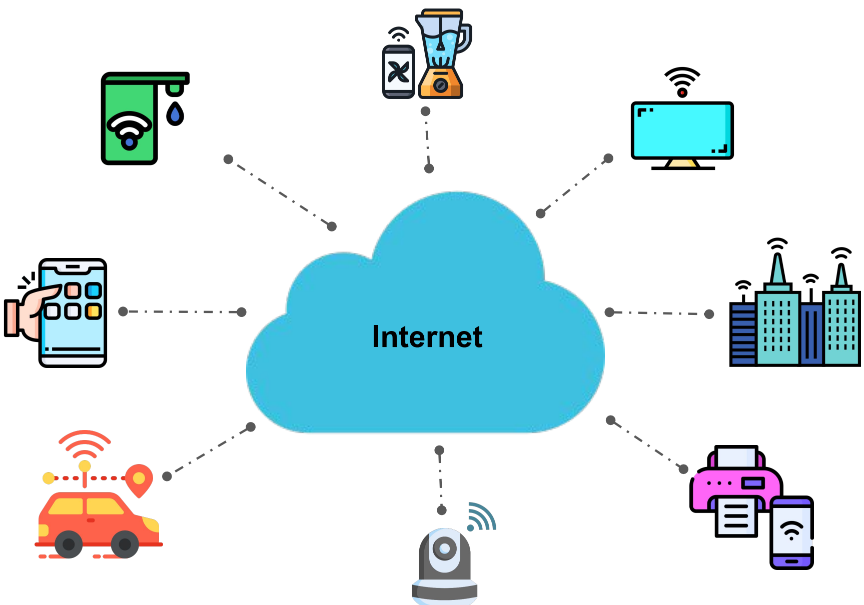

Figure 2.1: Internet of Things.

Wireless Sensor Network (WSN) is a collection with significant number of small, low-cost, low-power wireless sensor nodes that are distributed randomly or manually over a geographical area to measure and monitor physical or environmental conditions. Each sensor node collect local physical information, process them, and transmit data wirelessly to a central node called BS or sink which is consulted by the final user [6]. For the public notice of the phenomenon, the central node is connected to the internet.
The data collected by the sensor nodes can be used for various applications, such as environmental monitoring, healthcare and industrial. The design of the WSN is described by Figure. 2.2.

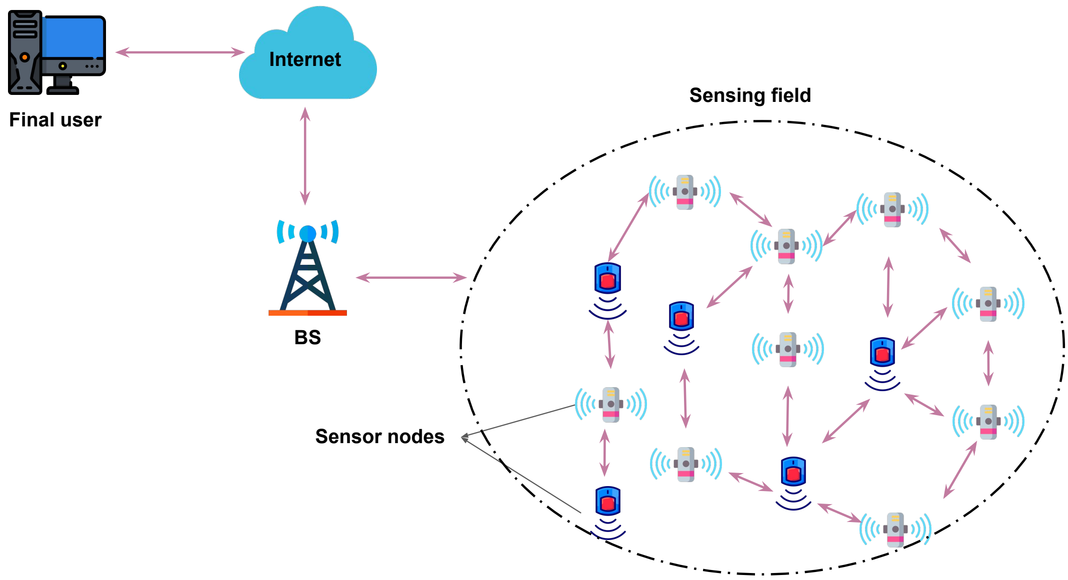

Figure 2.2: Wireless sensor network.

WSN is a crucial component of Internet Of Things, which is rapidly growing and expanding and plays a significant role in IOT by collecting and transmitting data from various sensors. The data can then be analyzed and used to make decisions. IOT-based WSNs' sensor nodes continuously scan their surroundings and send data to other nodes for notify the BS of any event [2].

2.3 Network Applications

A wide range of physical factors or situations, including light, sound, pressure, temperature, humidity, the composition of the soil, the quality of the air or water, and the characteristics of an item, such as its size, weight, location, speed, and direction, can be detected or monitored by sensor nodes. They can be used in any environment and reduce both the implementation cost and delay because of the availability of low-cost sensors and wireless communication, therefore the WSNs have a variety of real-life applications in both civilian and military fields.
And these are some popular applications of sensor networks (Figure. 2.3) :

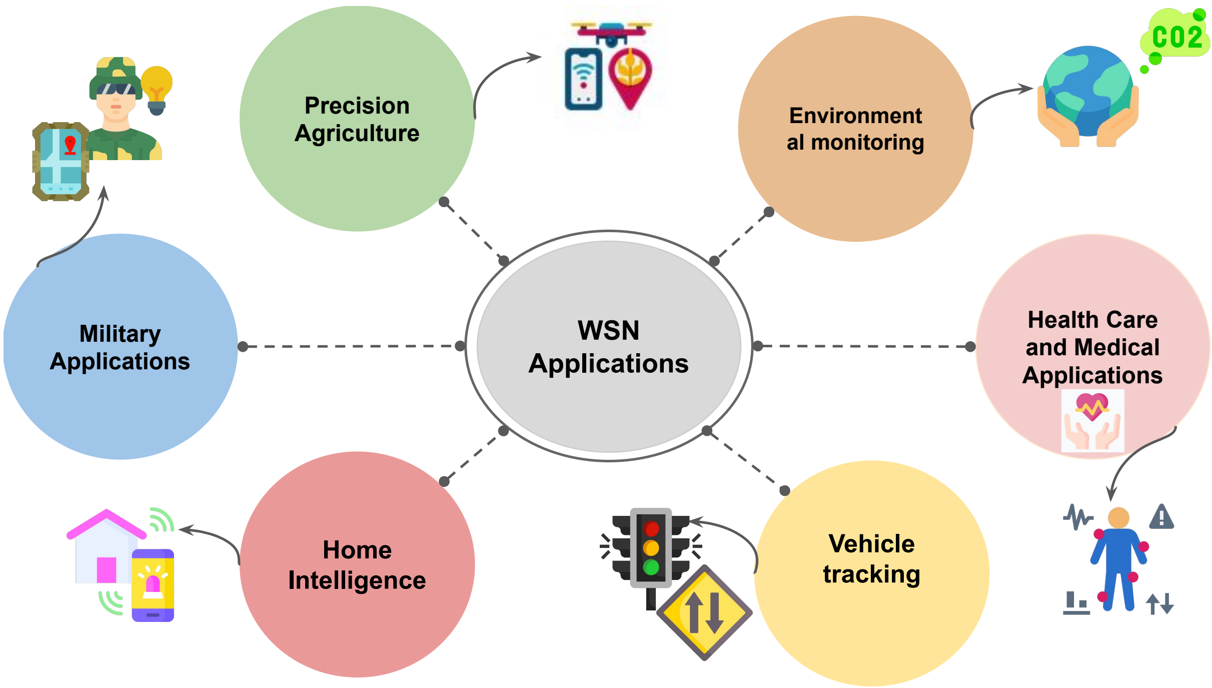

Figure 2.3: Wireless sensor network applications.

2.3.1 Precision Agriculture

In agricultural fields, the sensor nodes can be sown with the seeds.
The data collected by the sensors (temperature, humidity, carbon dioxide gas level...) is used to assess the most favorable seed density, fertilizers estimate, more accurately forecast crop yields and other input needs [7], or it is used by other devices that act based on this data received from the sensors. As well as, dry areas will be easily identified, therefore irrigation will be more efficient.

2.3.2 Environmental Monitoring

Environmental monitoring is one of the earliest applications of sensor networks. WSNs dispersed across hostile areas can detect fires, coal mining, earthquakes, tsunamis, floods, gas leaks and chemical and biological detection (detect pollution -CO2 levels) [7]. Thus it helps in taking precautions and also helps in raising awareness of the disaster that is about to occur.

2.3.3 Health Care and Medical Applications

The use of WSN in the medical field could ensure a permanent monitoring of the vital organs of the human being such as glucose monitoring, cancer detection at an early stage, thanks to micro-sensors that can be swallowed or implanted inside the human body (under the skin) [8]. Hence, they facilitate the diagnosis of some diseases by collecting physiological data like blood pressure, heart rate, pH levels, etc., using sensors each with a very particular task. The collected data is wirelessly transmitted to a central unit for processing and analysis. Algorithms identify e.g. the cancer-related patterns or anomalies, generating alerts for timely intervention.

2.3.4 Military Applications

WSNs are increasingly important in military operations for detecting nuclear weapons, monitoring enemy movements, and other tasks. They can be quickly deployed in a battlefield or enemy surveillance area without the need for infrastructure.
Furthermore it offers several benefits such as self-configuration, untended operation, and fault tolerance, making them critical components of future military systems. This will lead to more intelligent warfare with less human involvement [9].

2.3.5 Home Intelligence and Vehicle Tracking

WSN have the potential to create more convenient and intelligent living environments for people. For instance, wireless sensor nodes can be employed to remotely read utility meters such as water, gas, or electricity, and send the readings through wireless communication to a remote center [10].
To reduce congestion, Wireless Sensor Network (WSN) are employed to observe traffic by closely monitoring vehicles in a city, this is accomplished by keeping tabs on vehicles and detecting traffic violations [7].

2.4 Network Architectures and Protocol Stack

Wireless sensor network architectures and protocol stack form the foundation of designing and implementing those networks. Network architectures define the structure of the network, while the protocol stack comprises the set of rules that regulate communication between different layers of a network. Comprehending these concepts is crucial for developing efficient and effective wireless sensor networks that can collect and transmit data in real-time.

2.4.1 Architecture of Sensor

A sensor is a device having processing capability and memory, that measures physical or environmental changes and converts them into electrical signals and data that can be processed and analyzed by other devices or systems (Figure. 2.4).

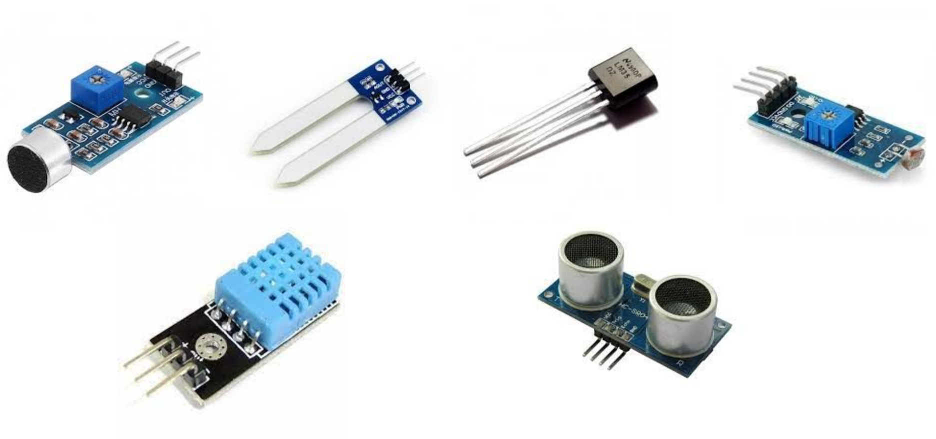

Figure 2.4: Sensors.

The architecture of a sensor generally consists of four main components : a Sensing unit, a Processing unit, a Power Unit and a Transceiver unit [11], as shown in Figure. 2.5.

- The sensing unit comprises two sub-units, a physical capture device that is responsible for detecting physical or environmental changes, and an analog/digital converter (electrical signals) known as ADC.
- The processing unit is analyzing the signals and making sense of the data collected by the sensing unit, where it is stored in memory.

- The transceiver unit (Communication interface) is composed of a transmitter/receiver (radio module), which enables communication between the various nodes in the network.

- The power unit is typically a non-replaceable battery. The primary constraint when developing protocols for sensor networks is the limited energy capacity at the sensor level.

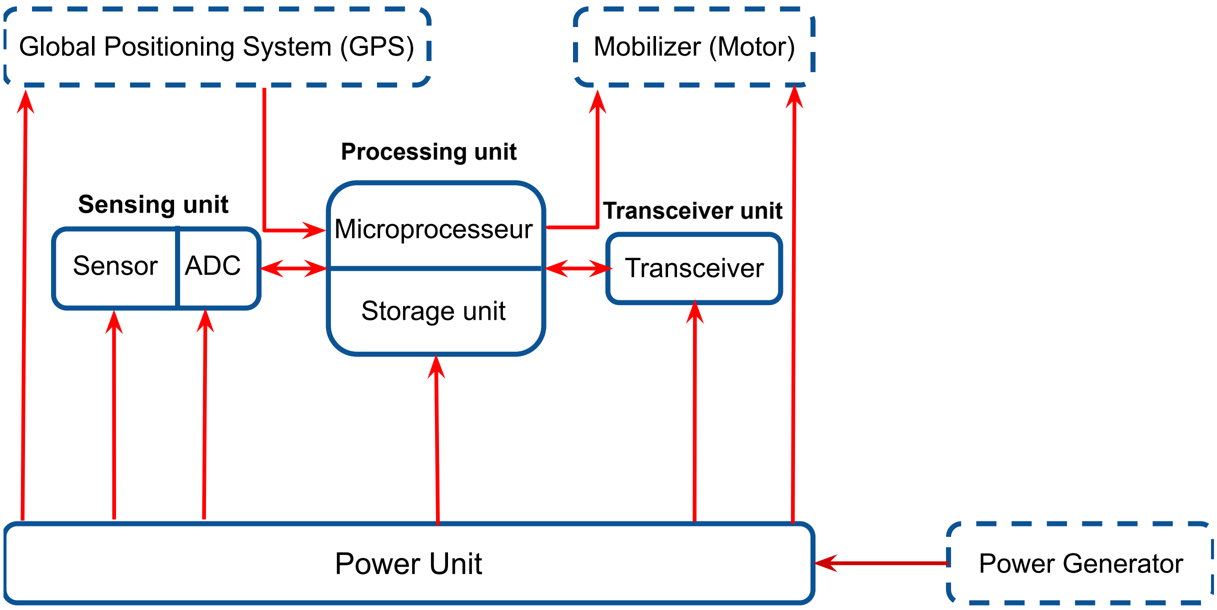

Figure 2.5: Sensor units.

Moreover, depending on the application and the type of sensor used, sensor nodes may require additional units such as GPS for location information or a motor for movement. However, it is important that these units are designed to be compact, energy-efficient, and cost-effective.

Sensor nodes often consist of a combination of hardware and software components, the software include the operating system designed for low-power such as TinyOS, Contiki, and RIOT, application software, and protocols that manage the sensor node's resources and enable communication.

2.4.2 Network Architectures

The architecture of a WSN includes a large number of sensor nodes deployed in a specific area, as well as one or more base stations. These nodes work together to collect and transmit data.
To send data to the sink, each sensor node can use Single-Hop or Multi-Hop communication (transmits its data toward the sink via one or more intermediate nodes). The architecture of network can be organized into two types : flat and hierarchical, which will be described below [12].

- **Flat Architecture** : All sensor nodes are considered homogeneous (the same capacities in communication and sensing task). Each node communicates with the BS (sink) via a multi-hop path and uses its peer nodes as relays. This results in a simple and low-cost network design, but can lead to issues such as network congestion and higher energy consumption due to long-range transmissions. However, flat architectures are suitable for small-scale networks or applications where the nodes do not have any specific roles. Figure. 2.6 illustrates a typical flat network architecture.

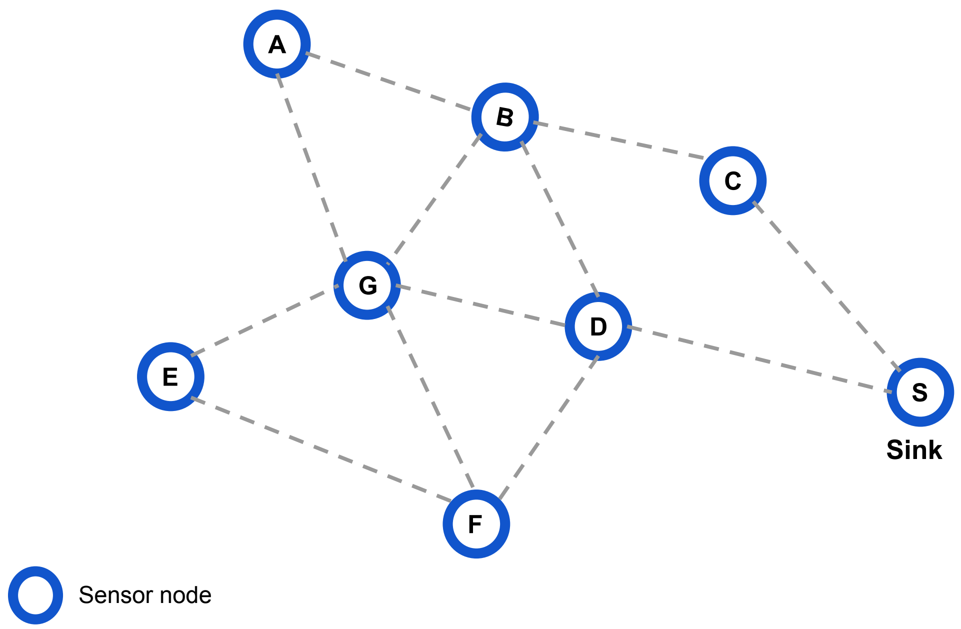

Figure 2.6: Flat Architecture.

- **Hierarchical Architecture** : Sensor nodes are organized into groups, called clusters. The node representing the cluster, called Cluster-Head (CH), where the cluster members send their data at short distance to the CHs which serve as relays for directly transmitting the data to the sink (Figure. 2.7).

  The main components of a cluster architecture are :

1. Sensor nodes, which can perform multiple roles such as data acquisition, storage, routing, and processing in a surveillance area.
2. Clusters, which are groups of sensor nodes that form the organizational unit of the network. The dense nature of WSNs requires them to be divided into groups to simplify communication tasks and respond to different constraints.
3. Cluster-heads, which are representatives of the clusters and are often required for organizing activities within the clusters. They are responsible for data routing, processing, aggregation, and fusion to simplify network management and optimize resource usage.
4. The base station, also known as the sink, is a collection point with superior computing power and potentially infinite energy. It can be connected to the internet or have a radio link to send data or alerts to a control center for the end user.
5. End users, who can use data from the sensor network via the internet or desktop computer for various applications.

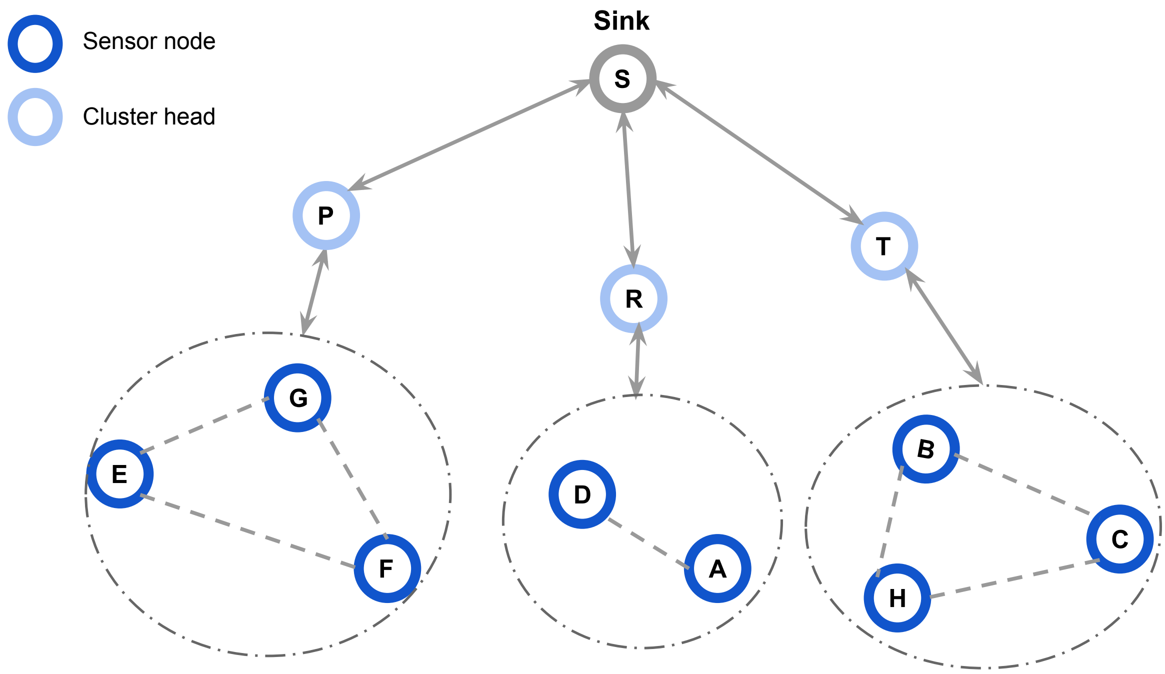

Figure 2.7: Hierarchical Architecture.

This architecture can reduce energy consumption, balance traffic load and enhance network scalability by minimizing the number of long-range transmissions when the network size grows. However hierarchical architecture is suitable for large-scale networks or applications that require specific roles for the nodes.
Figure. 2.7 illustrates a typical hierarchical network architecture.

2.4.3 Protocol Stack for WSNs

A protocol stack consists of multiple layers of communication protocols that provide the necessary functionality for enabling communication among nodes and with external systems.
The OSI seven-layer model, proposed by the International Organization for Standardization (ISO), forms the basis for the design of the WSN protocol stack. However, unlike the OSI model, the WSN protocol stack does not adopt all the seven layers. In reality, the seven-layer OSI model has too many layers making it overly complex and difficult to implement. The protocol stack for WSNs consists of five protocol layers as shown in Figure. 2.8.

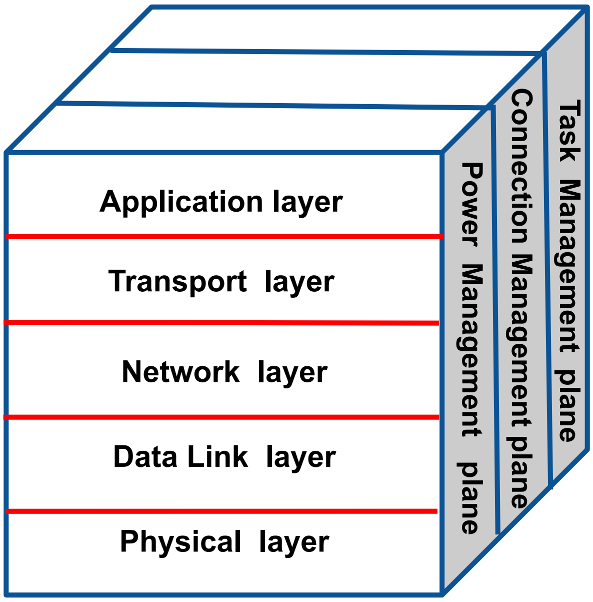

Figure 2.8: Protocol stack for WSNs.

On the other hand, the protocol stack can be divided into a group of management planes across each layer, including power, connection, and task management planes [13].

- **Application Layer** : Provide the interface for high-level applications, enabling data to be accessed and processed by external entities (final users).
- **Transport Layer** : Offers transportation of data and flow control.
- **Network Layer** : Manage neighbor discovery and data routing among nodes and can be either flat or hierarchical, as discussed earlier.
- **Data Link Layer** : Provide links between the nodes and the BSs, data reliability and error checking and correction.

- **Physical Layer** : Represents the hardware that handle the transmission and reception of data over the wireless medium.

The power management plan controls battery usage. For example, after receiving or sending a message, the sensor switches off its radio (entering sleep mode) in order to prevent redundant messages and conserve energy. While the mobility management plan tracks sensor node movement in a way that allows them to enable efficient monitoring of neighbors, optimizing energy use and task performance.
Finally the task management plan distributes and orders the different data collection tasks within a region. It is not mandatory for all nodes to perform data collection simultaneously. Instead, some nodes may undertake this task more frequently than others based on their current battery level.

2.5 Routing in Wireless Sensor Networks

Routing protocols play a vital role in WSNs, it refers to the process of finding the most efficient path for data transmission between nodes to reach the BS. It involves the selection of intermediate nodes that relay data from one node until it reaches the intended destination [14].
The objective of routing in those networks is to optimize network efficiency, minimize energy consumption and prolong network lifetime while ensuring reliable data delivery. Various routing protocols have been developed for this networks, such as LEACH, TEEN, and SEP, to name a few.

2.5.1 Classifications

Routing protocols in WSNs can be classified based on network structure, route discovery process, and protocol routing policy. However, some routing protocols can fit into multiple categories and sub-classes, Figure. 2.9 shows the classification.

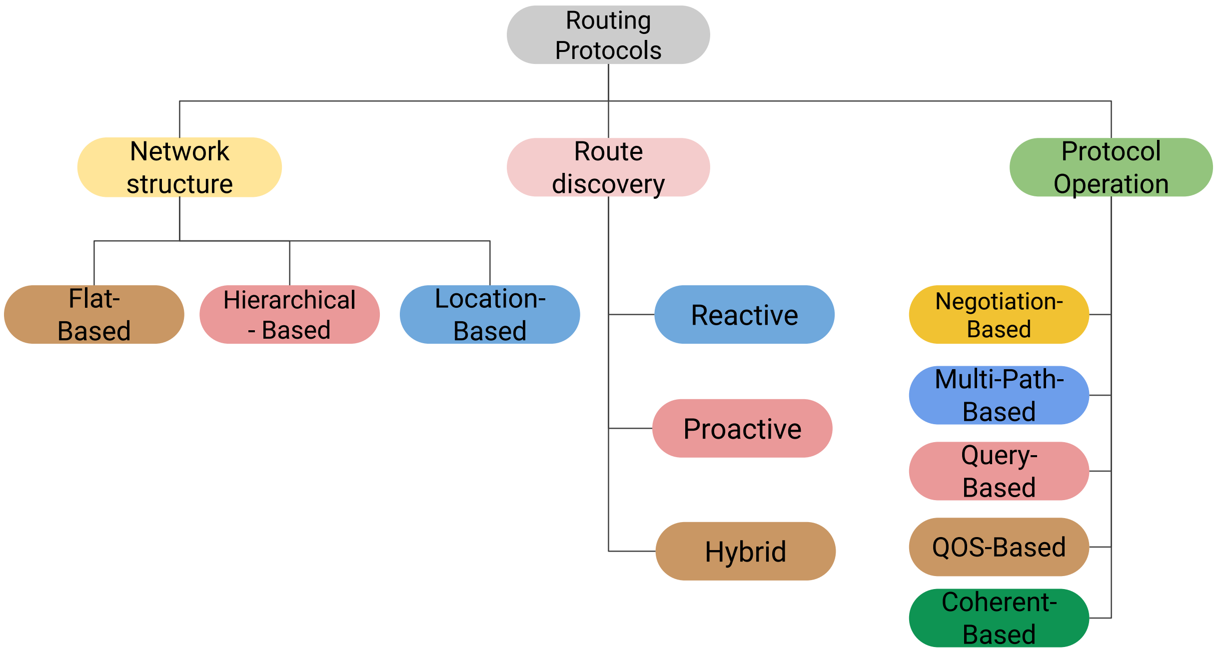

Figure 2.9: Classification of routing protocols in WSN [15].

1. **Route discovery routing protocols** : Routing paths based on route establishment, can be established either in advance, "proactively" or instantly, "reactively" as per the requirement [16].

   (a) **Proactive routing protocols** : Known as table-driven protocols, they establish routing information at all times, even if it is not needed. They continuously update routing tables and use these tables to determine the best path to a destination node. Examples of proactive protocols include DSDV and OLSR.

   (b) **Reactive routing protocols** : Known as on-demand protocols, they establish routes only when required. These protocols do not maintain information about the network topology unless it is requested. When a node needs to communicate with another, it broadcasts a route discovery message, and the protocol tries to find the route to the destination node. Examples of reactive protocols include AODV and DSR.

   (c) **Hybrid protocols** : The aim is to combine the advantages of proactive and reactive routing protocols, while avoiding their disadvantages. However, hybrid protocols come with the added cost of increased complexity due to more functionality and source code.

2. **Protocol operation routing protocols** : The operation is the way in which data is propagated in the network [16].

   (a) **Negotiation-Based routing protocols** : When sensor nodes detect the same phenomenon, they often flood the network with identical data packets. This redundancy issue can be addressed by utilizing negotiation-based routing protocols.
   By engaging in negotiation through messages, sensor nodes can avoid redundant data transmission. This process ensures that only valuable information is transmitted by making adequate decisions. However, this mechanism produces non-tolerable delays in delivering data to the sink. Moreover, control messages exchanged between nodes can lead to network congestion, increased load, and additional energy consumption.
   Examples of those protocols include SPIN.

   (b) **Multi-Path-Based routing protocols** : This routing technique ensures the presence of multiple reliable paths for data transmission between sensors and the sink. It enhances fault tolerance by providing alternative paths, allowing for seamless data transfer even if the primary path fails. So there are multiple routes exist between the source and destination, but an energy loss can occur due to the periodic transmission of messages required to maintain the alternative paths.
   Some well-known multi-path routing protocols are DD and EAR.

   (c) **Query-Based routing protocols** : The sink node initiates a data request that propagates through the network to query the sensor nodes. The node holding the requested data responds to the requesting node by sending the data back along the reverse path of the request. This reduces network load by using the same path for data as the request. However, congestion or busyness on the reverse path can pose a potential issue during transmission.

   (d) **QOS-Based routing protocols** : The aim is to fulfill specific quality of service requirements, such as transmission delay, latency, bandwidth or reliability level, during the process of data routing. These protocols are well-suited for surveillance applications, as they consider transmission delays. However, energy constraints must be considered alongside QOS criteria. One limitation is the lack of safety consideration, which is an important aspect of QOS.
   One example of such a protocol is SAR.

(e) **Coherent-Based routing protocols** : Coherent-Based protocols focus on minimal processing, eliminating duplicates before data transmission to recipients and data aggregators. In contrast, non-coherent protocols involve significant local processing of raw data by nodes before further processing. To achieve energy-efficient routing, it is recommended to use Coherent-Based processing methods.

3. **Network structure routing protocols** : This category is based on the structure of the network which determines the organization of the nodes in the network [17].

   (a) **Flat protocols** : All nodes in the network are identical and communicate directly with each other, except for the sink node which collects information from sensor nodes and forwards it to the end user over the internet. Nonetheless, the performance degrades when the network size increases.Examples of such protocols include SPIN, MCFA, and EAR.

   (b) **Hierarchical protocols** : The nodes have different roles and the whole network is divided into clusters. Indeed, certain nodes are selected to perform particular functions as group leader (CH). Its major drawback is the overload of the cluster-heads which induces an imbalance in the energy consumption in the network.

   (c) **Location-based protocols** : Sensor nodes are identified based on their locations. The proximity of neighboring nodes is estimated by analyzing the signal strengths received from source nodes. The relative coordinates of neighboring nodes can be acquired through information exchange among neighbors, or by utilizing GPS communication with a satellite.

The following comparative table (Table. 2.1) provides an analysis of various routing protocols used in wireless sensor networks montioned in this section. It aims to highlight the type, QOS service, and data aggregation availability of each protocol. This comparison aids in understanding the diverse approaches and methods.

Table 2.1: Summary of some routing protocols for WSNs.

Protocol name	Type	QOS	Data aggregation
AODV	Reactive	-	-
DSDV	Proactive	-	-
DSR	Reactive	-	-
DD	Multi-Path-Based/Flat	-	✓
EAR	Multi-Path-Based/Flat	-	-
LEACH	Hierarchical	-	✓
MCFA	Flat	-	✓
OLSR	Proactive	-	-
PEGASIS	Hierarchical	-	✓
SAR	QOS-Based	✓	-
SPIN	Negotiation-Based/Flat	-	✓
TEEN	Hierarchical	-	✓

2.5.2 Hierarchical Routing Protocols

Hierarchical routing originally developed for routing data in wired networks, hierarchical routing can also be applied to wireless networks with appropriate enhancements for improved scalability and communication efficiency. This routing approach involves dividing wireless sensor nodes into multiple levels, with most protocols featuring two routing layers. The first layer selects cluster heads while the second layer makes routing decisions [12]. One of the earliest hierarchical routing protocols is LEACH. Further details on LEACH are provided in the following section.

The collaboration of sensor nodes plays a crucial role. These nodes continuously monitor and gather data about the environment, process it, and collaborate to transmit reliable information to a base station. By pooling data from multiple nodes, valuable insights can be gained about the environment. Collaborating nodes can also balance communication costs and computation energy.

Hierarchical routing, which divides the network into clusters and designates some nodes as cluster heads (CHs), offers improved energy efficiency and scalability. CHs collect, aggregate, and compress information from their cluster member nodes, before transmitting it to the central node. This protocol helps to streamline data transmission and optimize network efficiency.

Clustering is the most used method in this topology thanks to these advantages such as data aggregation in order to reduce the number of messages transmitted to the destination as well as great scalability and better energy efficiency. The major challenge of clustering is the selection of CH and the organization of the clusters.

The different schemes for hierarchical routing protocols mainly differ in how the cluster head is selected and how the nodes behave in the inter and intra-cluster domain [18].

In our work, we have opted to develop a new protocol within the hierarchical protocols category. This choice is driven by the inherent advantages that hierarchical protocols offer, such as energy conservation and effective network organization, especially in scenarios with a large number of nodes. By adopting a hierarchical approach, our proposed protocol aims to leverage these benefits to enhance energy efficiency and optimize network management.

2.5.3 Characteristics of Hierarchical Routing Protocol

A hierarchical routing protocol entails the definition of multiple tasks, which serve as criteria for their classification :

- **The clustering algorithm** : A classification into three types of these algorithms is proposed in [18] based on the type of control algorithm executed :
  -Centralized algorithm : The algorithm is executed on a single node that has a global view of the network, generally, at the BS level.
  -Distributed algorithm : The algorithm is executed by each node in the network.
  -Local based algorithm : The algorithm is based on the nodes geographical position.

- **The re-election of the CHs** : CH consumes more energy compared to other nodes in the network due to their function of coordinating member nodes and aggregating data. This can lead to an imbalance in energy distribution. To overcome this problem, a rotation of the CH role is organized periodically within the cluster or the entire network. Thus allows for more even distribution and prevents early depletion of energy consumption across the network [18].

- **The nature of the generated clusters** : The Clustering algorithms used can generate two types of clusters [19] :
  -Disjoint clusters : Node can only belong to one and only one cluster at a time.
  -Interconnected clusters : This type of clustering allows nodes to belong to one or more clusters at a time.

- **Inter-cluster communication** : CHs can communicate directly with the base station (one hop), or in several hops [19].

- **Intra-cluster communication** : Communication between the CH node and the other cluster members can be done either in a single hop or in several hops [20].
  -Direct communication (in one hop) : The data packets are sent directly to the CH.
  -Indirect communication (in several hops) : In this case, each member of the cluster sends its data to the nearest member of its cluster until the end at the CH.

- **The level of data aggregation** : Depending on the type of sensors used, data aggregation can be performed at each network node or exclusively at the CHs [20]. Data aggregation reduces the amount of data exchanged between nodes, and consequently reduces the energy spent.

2.5.4 Low Energy Adaptive Clustering Hierarchy Routing Protocol (LEACH)

LEACH is a clustering routing protocol for WSNs that aims to increase energy efficiency by rotating cluster head (CH) selection through a random priority value. It operates in rounds with two phases : a setup phase and a steady state phase. During setup phase, CH selection, cluster formation, and TDMA schedule assignment occur. In the steady state phase, member nodes send data to the CH during their allocated time slot, reducing collisions and energy consumption. CHs aggregate data and transmit it to the base station using the TDMA schedule while sensing the channel states to avoid interference. LEACH improves battery life for member nodes [21].

Figure. 2.10 shows two rounds of the leach protocol which is based on the space location for the selection of the CH (the nearest CH) which is shown by the Voronoi diagram. All nodes are marked with a given symbol belong to the same cluster, and the CH nodes are marked with a •.

In this research, we have selected the LEACH protocol as the base for building our idea and proposed work, due to its above-mentioned advantage, with a focus on addressing its disadvantages and shortcomings.

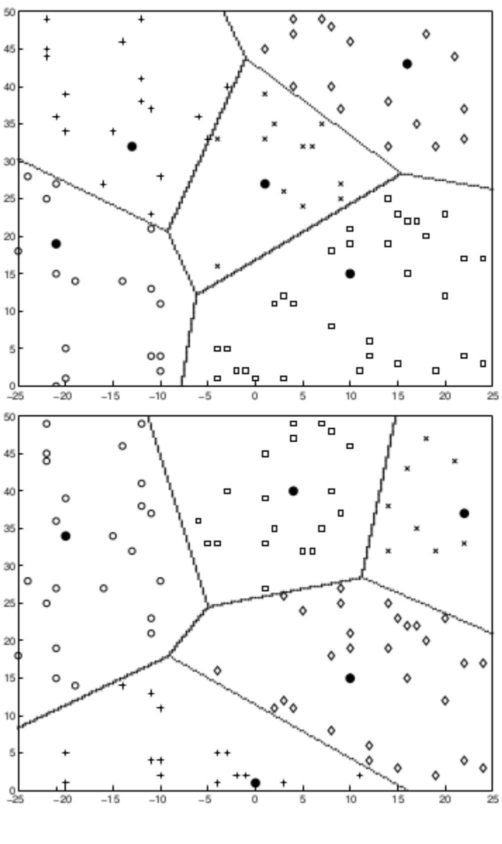

Figure 2.10: LEACH clustering architecture [21].

2.5.5 Publish/Subscribe Protocol

One of the best known Publish/Subscribe protocols is Message Queuing Telemetry Transport (MQTT) Protocol.

MQTT is a lightweight, simple and efficient messaging protocol designed for the Internet of Things (IOT) applications where bandwidth, power and resources are limited such as resource-constrained sensor nodes. It is a publish/subscribe-based protocol where clients connect to a broker (MQTT server) and publish messages to topics or subscribe to receive messages from topics [3], (Figure. 2.11). MQTT uses a small header size, making it lightweight and suitable for low-bandwidth networks. It also supports quality of service (QOS) levels, allowing the sender to specify the level of assurance of message delivery. In addition, MQTT is designed to work over unreliable and intermittent networks and supports multiple clients and servers. MQTT has become a widely used protocol in IOT applications due to its simplicity, low overhead, and support for low-power devices [22].

The use of a routing protocol based on MQTT in a sensor network can greatly improve the network's efficiency. MQTT's lightweight and efficient messaging protocol can save energy by reducing the amount of data sent between nodes. This leads to a more robust and fault-tolerant network, as nodes can dynamically adjust their communication patterns in response to changes in the network.

Ultimately, a routing protocol based on MQTT can optimize the performance of a sensor network, enabling it to efficiently gather and transmit data over extended periods of time.

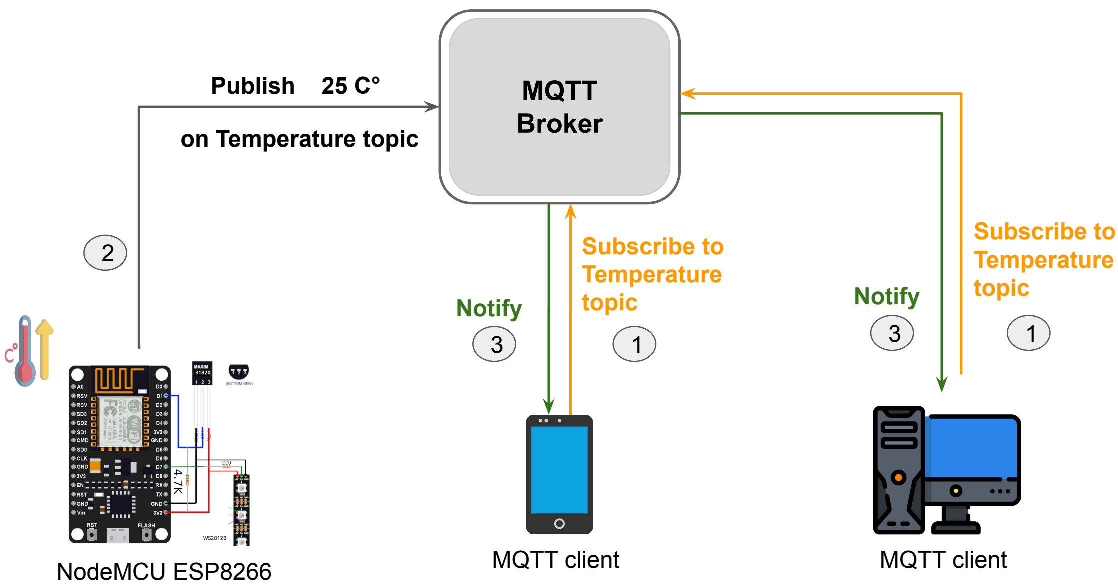

Figure 2.11: MQTT architecture [23].

Conveying and routing Publish/Subscribe messages between publishers, broker and subscribers pose a significant challenge that should be looked at and solved.
This is what we would like to do in this study, to find a solution to this problem by finding the most appropriate and effective route to exchange these messages with less usage of energy.

2.6 Constraints

Research in the field of sensor networks has revealed several challenges and constraints, among these issues, we cite :

1. WSN networks constraints :

   (a) **Routing** : Design an efficient routing protocol in terms of minimizing energy consumption, choosing optimal routes for routing information is a big challenge in IOT and WSN, when the network must scale up without its performance degrading [24].

   (b) **Cross-layer** : Traditional network design models focus on optimizing performance at each layer independently without considering the impact on other layers, which is not practical for sensor networks. To optimize the overall network capabilities, a collaborative approach that balances performance, dependence, and flexibility must be developed, allowing for compromise and cooperation among different layers of the network [24].

   (c) **Limited security** : Compared to traditional wired networks, WSN networks have limited security. Since data is transmitted wirelessly, attackers can easily intercept the exchanged messages, applying noise to the channel, Therefore prevent sensor nodes from transmitting messages. However, securing such networks is challenging due to constraints such as limited resources. Using security algorithms requires memory resources for storing code and data, as well as energy and computational resources [24].

   (d) **Production cost** : Production cost is a crucial factor in IOT and sensor networks as they typically consist of a large number of nodes. The price of a node is critical in order to be able to compete with a traditional surveillance network [11].

2. WSN routing constraints :

   (a) **Energy consumption** : Energy is necessary for a sensor node to perform various tasks such as data collection, processing, and communication to reorganize the network. However, due to the difficulties in recharging batteries, particularly in hostile catchment areas, it is crucial to conserve energy [25].

(b) **Fault tolerance** : In hostile environments, some nodes may fail due to lack of power, physical damage, or obstacles. Fault tolerance refers to the ability of the sensor network to continue functioning without interruption despite the presence of faulty nodes [25].

(c) **Scalability** : Scalability is an important factor to consider, as the number of deployed nodes can range from hundreds to thousands, depending on the application. Efficient routing protocols are required to handle the large number of nodes and ensure that performance is not significantly impacted even if some nodes are damaged. Additionally, the base station must have enough memory to store the information received from a high number of sensors [25].

(d) **Data aggregation** : Data aggregation is a technique used in sensor networks to reduce the number of transmissions by combining similar packets from different nodes and applying certain aggregation functions such as suppression, minimum, maximum, and average [19]. However, it can pose constraints in routing within WSNs.

(e) **Addressing** : Due to the sheer number of nodes involved, establishing a global addressing scheme for all the nodes is not feasible. Hence, routing protocols based on new addressing mechanisms or new routing techniques that do not require unique identifiers for each node are needed (data-Centric routing) [16].

(f) **Computing power** : Network protocols do not have to be complex or necessarily require complex computing power [11].

2.7 Conclusion

In conclusion, this chapter provided an overview of wireless sensor networks and their applications.

The chapter also discussed network architectures and protocol stacks, as well as routing protocols such as hierarchical routing and the Low Energy Adaptive Clustering Hierarchy Routing Protocol.

The publish/subscribe protocol was also highlighted, along with some constraints that can affect the performance of wireless sensor networks. Overall, this chapter serves as a foundation for further exploration of WSNs and their potential in various fields.

In the next chapter, we will delve into the existing literature related to our work and identify gaps and limitations that need to be addressed.

Chapter 3

Related Works

Contents

3.1 Introduction

In the recent past years, many strategies of routing have been proposed for WSN-based IOT to solve series of challenges that face WSNs to accomplish their tasks with full success and less cost.

Along with this chapter, we start with the presentation of some existing approach of routing and research in the field. Then, we will discuss their gaps and limitations.
We also illustrate a summary of the key findings and themes identified before concluding the chapter by highlighting the chapter's main points.

3.2 Literature Review

This section provides a thorough review of the related literature to the development of routing protocols in IOT and WSN. In this review we have chosen five papers that dealt with this topic.

Azim and Islam [26] proposed the Hybrid-Low Energy Adaptive Clustering Hierarchy protocol. H-LEACH calculates the threshold value in each round using the residual energy and maximum energy of nodes. Nodes with residual energy above the average network energy are selected as CHs. Comparative analysis with the LEACH protocol showed H-LEACH's superior efficiency in simulations. However, it is important to note that relying only on node residual energy is inadequate for ensuring stable power consumption in WSN. Since the density of nodes also influences power dissipation, indicating the necessity for additional considerations with residual energy.

Sarobin and Thomas [27] proposed an energy-robust clustering algorithm called Improved LEACH. It incorporates node heterogeneity by considering varying capabilities within the network. Nodes transmit their IDs and residual energy to the BS, which sorts them based on residual energy levels. High-energy nodes are selected as CHs, and their IDs are broadcasted for cluster formation. Simulation results demonstrated improved network lifetime. However, the algorithm does not address network stability concerns, especially when the first node exhausts its energy prematurely, also nodes lose a lot of energy when sending energy and ID each time.

The paper [28] by Paolo Costa and others, explores socially-aware routing in delay-tolerant mobile ad hoc networks (DT-MANETs) for publish-subscribe communication. The authors propose an algorithm that considers social relationships between nodes to improve message delivery. While the paper addresses the challenges of communication in DT-MANET and introduces an interesting concept, it lacks detailed evaluation, scalability analysis and consideration of recent advancements in the field. Further research and evaluation are needed to assess the algorithm's performance in practical scenarios.

In the paper [29], the authors discuss the application of the MQTT protocol for routing in IOT systems. The authors use MQTT protocol to demonstrate effective communication between client and server. While the paper highlights the potential benefits of using MQTT for efficient communication in IOT, it lacks in-depth details regarding network topology and comparison with existing routing approaches. Furthermore, evaluation metrics like the energy efficency or the network lifetime are not discussed, making it difficult to assess the effectiveness of the proposed routing mechanism.

In a study by Mohd Adib Omar and others [30], a routing algorithm was proposed that employed a three-level hierarchy to reduce the CH's load. The algorithm utilized random selection of cluster heads. Additionally, multi-hop transmissions were employed for inter-cluster communication. This approach resulted in a longer network lifetime compared to LEACH. However, the random selection of CH raised fairness concerns, as there was a possibility of selecting a node with low energy or lower density as a CH.

3.3 Gaps and Limitations

The main problems identified in above survey are as follows :

- The reliance on insufficient factors during the CH selection and the cluster formation in the papers [26] and [27]. In addition, failure to take into account the maintenance of the energy level in the network.
- Performance Evaluation : Some papers lack performance evaluation and comparison with existing approaches. It is important to assess the proposed algorithms or protocols against relevant performance metrics, such as latency or energy efficiency, to prove its effectiveness.
- Network Congestion : Above papers need to address how their solutions handle network congestion and ensure efficient utilization of network resources. Because in dense networks or scenarios with high traffic loads, congestion can occur, leading to performance degradation and increased energy consumption.
- Energy Efficiency : Energy consumption is a critical concern in WSN due to the limited battery life of sensor nodes. While approaches [28] and [29] lack to improve energy efficiency, there are gaps regarding the evaluating of long-term effects on network lifetime in [26] and [27].
- Scalability : Many algorithms and protocols proposed for wireless sensor networks or IoT may work well in small-scale scenarios but face challenges when scaled up to larger networks. The above-mentioned studies lack scalability evaluation.
- Fault Tolerance : WSN and IoT systems are vulnerable to node failures, network partitions, or communication disruptions. The relevant papers are lacking to address how their proposed solutions handle such scenarios and ensure fault tolerance.
- The shortage of addressing the explanation of the technologies proposed details, which opens the way for many questions that make the research somewhat unclear at some points.

- Lack of using a specialized and certified simulation tool, and relying instead on simple programming with an IDE (in [29]).

3.4 Summary

The subsequent tables (Table. 3.1 & Table. 3.2) will analyze the aforementioned related studies. Table. 3.1 offers a summary and evaluation of the above-mentioned related studies, while Table. 3.2 presents a comparison of the papers in term of approach-based on, network lifetime, energy efficiency, and the simulator used.

Table 3.1: Summary of related studies.

Reference	Description	Remarks
Azim and Islam [26]	Determines the threshold value for each round based on the residual energy and maximum energy of nodes. It selects CHs having more energy.	The remaining energy is insufficient to guarantee stable power usage.
Sarobin and Thomas [27]	Nodes transmit their IDs and remaining energy to the BS to sort them based on their residual energy. High-energy nodes are selected as CHs, their IDs are broadcasted to facilitate cluster formation.	Failure to consider network stability concerns. Furthermore, there is energy wastage during the forwarding of IDs and energy.
Paolo Costa Et al [28]	The proposed algorithm takes into account the social relationships between nodes in order to enhance the delivery of messages.	Lack of thorough evaluation, network life time and scalability analysis.
Rahul Johari Et al [29]	Examines the MQTT protocol by analyzing the communication between multiple clients and a server. The goal is to assess the protocol's strengths and uncover any weaknesses.	Absence of evaluation metrics, the specialized simulator shortage and details regarding approach and network topology.
Mohd Adib Omar Et al [30]	Three levels to minimize cluster head load, with multi-hop communications for intra-cluster.	Random selection of cluster heads.

Table 3.2: Comparative analysis.

Ref	Approach-based on	Network lifetime	Stability period	Energy	Simulator
[26]	LEACH	Mediocre	Medium	Good	J-sim
[27]	LEACH	Very good	Good	-	MATLAB
[28]	Pub/Sub	Not mentioned	Not mentioned	-	OMNeT++
[29]	MQTT	Not mentioned	Not mentioned	-	Eclipse IDE
[30]	LEACH	Good	Acceptable	Good	MATLAB

3.5 Conclusion

In conclusion, this chapter has provided a comprehensive overview of the related works in the field of routing protocols and energy efficiency in WSN and IOT.

The literature review has highlighted the current state of research and identified the gaps and limitations in the existing literature. These limitations provide an opportunity for further research to contribute to the advancement of knowledge in this area.

Finally the summary has provided a succinct overview of the key findings of the chapter.

In the next chapter, we will present our contribution.

Chapter 4

Contribution : Proposition of our Routing Protocol for IOT Networks

Contents

4.1 Introduction

One of the essential parameter in WSN-based IOT is the energy parameter.

In this chapter, we focus on our contribution. We present our new routing protocol which is a combination between two protocols, MQTT and LEACH. Our main aim is to help subscriber and publisher to select the best route to send messages in term of energy and efficiency.

The results of our simulation conducted using OMNet++ simulator [31] depict the good performance of our solution.

The rest of this chapter is organized as follow : We will present our contribution with development of our different phases. Then, we will evaluate and discuss the performance of the proposed approach.

4.2 Proposed Protocol

In order to ensure the efficient event delivery (pub/sub messages delivery) in publish/subscribe systems between publishers, central broker and subscribers, we propose our routing protocol for an MQTT system to provide a method to allow nodes to find the best route or way for the pub/sub messages transmission in the network. Node clustering is a critical element in the practical deployment and operation of WSNs, for this reason our work is based on LEACH protocol (2.5.4) where we suggest to divide the network into clusters, each cluster have a responsible (Cluster Head) who is, in the same time, a broker bridge that forwards the data from his own cluster to the central broker.

The nodes with both, higher energy and density, will act as a broker bridge, as a CH. While the lower energy and density nodes will act as simple sensors or actuators (publishers and subscribers), joining to the cluster having the nearest CH, and turning their radio off during their inactive periods to serve the energy. Each node have the possibility to be a bridge node (broker bridge) to the central broker after calculating a $T(n)$ threshold detailed in the next subsection 4.2.1. This architecture can reduce energy consumption, balance traffic load and enhance network scalability by minimizing the number of long-range transmissions when the network size grows.

Moreover, data acquired by one sensor node is often correlated with its neighbors, so in our protocol, data aggregation will produce by the broker bridge in each cluster, by reducing the data redundancy in the network. Specially when the central broker (CB) is far away, local data aggregation is preferable to direct communication.

(i) Network congestion solution : In dense networks with high traffic loads, congestion can occur, leading to performance degradation and increased energy consumption. In our protocol, to handle network congestion we use TDMA mac protocol for inter-cluster communications and the CSMA/CA for intra-cluster communications, to ensure efficient utilization of network resources. The TDMA table is provided by the broker bridges (BB) to their clusters to avoid congestion and collision.

(ii) Fault tolerance solution : WSN and IOT systems are vulnerable to node failures. In this study, to ensure fault tolerance, we divide the network lifetime into rounds, in each of which new clusters are formed (new clustering phase generated) to achieve fault tolerance and balance energy between nodes.

(iii) Assumptions : The network model used in this research is based on certain assumptions listed below :

- ✓ The central broker (CB) have a fixed location.
- ✓ Nodes of the network are heterogeneous (actuators and sensors) and limited in energy.
- ✓ N nodes are randomly distributed in a square area of M x M.
- ✓ The same initial energy is equipped for all nodes.
- ✓ A unique ID is assigned for each node.
- ✓ The CB has an unlimited energy reserve and a substantial transmission power. As a result, all sensor nodes are within range of the CB.
- ✓ The nodes can use if necessary, the power control in order to regulate the power of transmission according to the distance of transmission. Thus, a CH can directly transmit.
- ✓ Nodes location is fixed (immobile).
- ✓ The nodes have sufficient memory to avoid congestion.
- ✓ All sensors are sensing the environment at a fixed rate and thus always have data to send.

(iv) Network nodes : Our protocol proposed in this study is designed to be applied in an agricultural environment, where there are sensor nodes collecting data, and actuator or operator nodes that act according to the information they receive from the sensors. These are examples of nodes that could be used in our protocol for agricultural applications :

- Sensor nodes : These are nodes that are equipped with sensors to collect data such as temperature, humidity, light, sound, or motion, then transmit it by publish in a topic to the broker bridge for processing and analysis. Sensor node could be a soil moisture, weather station node or crop monitoring node.

- Actuator nodes : These are nodes that can actuate physical devices such as motors, valves, or switches based on commands received from other nodes (usually from the broker bridge) in the network by subscribing to a topic. An actuator node could be an irrigation or a light actuator node used to turn on a light or adjust a thermostat.

- Central Broker : This node is used to aggregate data from multiple bridge brokers and forward it to subscribed client in the network when it is published in their topics.

- Cluster Head nodes : These are nodes that act as a bridge (Broker Bridges) between the central broker and nodes (actuators and sensors), they are responsible for managing a cluster of nodes in the network. It aggregate data published from its cluster members, before transmit it to the central broker.

Overall, any node that needs to communicate in a low-power, low-bandwidth IOT network can potentially use this protocol. The LEACH and MQTT protocols are designed to be flexible and scalable, allowing them to be adapted to a wide variety of IoT applications and use cases.

4.2.1 Protocol Description

The operation of our routing protocol is divided into rounds, where each round starts with a broker bridge discovery phase for cluster organization, followed by a publish/subscribe routing phase for data transmission. To minimize overhead, the publish/subscribe routing phase is relatively longer compared to the broker bridge discovery phase.

The Broker bridge discovery phase : This phase represents the first phase in our protocol, during which the broker bridges will be selected and the clusters will be formed. This occur in three main ordered sub-phases described bellow : Initialization, Broker bridge discovery and Cluster formation.

(1) Initialization : In this step, each node in the network is initialized with a unique identifier, energy level, specific location. The network nodes exchange among them small HELLO messages described in Figure. 4.1 to discover neighbors, and then each one calculate his own density (Formula 4.3).

(2) Broker bridge selection : Each node in the network use the discovery algorithm to decide whether or not to become a broker bridge (become a CH) for the current round. The broker bridge has the responsibility of managing the cluster and gathering data from cluster member nodes and transmitting aggregated data to the central broker. This task consumes a significant amount of energy, necessitating high-power amplification for the transmission process.

The decision to be a broker takes into account four parameters : the history of the node (if he has already been a CH), the optimal percentage of a node, the average energy of the node and its own density (neighbor discovery). A random number is generated for each node (between 0 and $T(n)_{max}$), and if the generated number falls below a threshold $T(n)$, for that round the node is designated as a broker bridge.

Remark. $T(n)_{max}$ represents the maximum value that $T(n)$ can reach, considering the maximum values of both energy average and density, as well as the maximum value of the number of rounds. This ensures that no node will generate a random number greater than $T(n)_{max}$.

The threshold $T(n)$ is calculated using the following formula (Formula 4.1) which is an improving of the LEACH formula :

$$T(n) = \begin{cases} \frac{P}{1-P*(r \mod \frac{1}{P})} * E_{avg} * D, & \text{if } n \in G. \\ 0, & \text{otherwise.} \end{cases} \tag{4.1}$$

Table 4.1: Describing function of $T(n)$.

Giving Letter	Meaning
T	The threshold representing the possibility of a node to be a CH.
E_{avg}	The energy average of node i.
D	The density of node i.
G	The set of nodes that were not selected as CH in the previous $\frac{1}{P}$ rounds.
r	The number of round.
p	The percentage of CH.
n	The number of node.

While :

$$E_{avg} = \frac{E_{residual}}{E_{total}}. \tag{4.2}$$

Table 4.2: Describing expression of E_{avg}.

Giving Letter	Meaning
E	The energy average of node i.
$E_{residual}$	The residual energy of node i.
$E_{initial}$	The initial energy of node i.

And :

$$D = \frac{D_{actual}}{D_{potential}}. \tag{4.3}$$

Table 4.3: Describing expression of D.

Giving Letter	Meaning
D	The density of node i.
D_{actual}	The actual edges (connections) of node i.
$D_{potential}$	The potential edges of node i.

Remark. Let D_{actual} be the number of real connections that a node i have it, then $D_{potential}$ is the total connections that node i should have it.

In each round, nodes that have elected themselves as broker bridges, will broadcast an advertisement messages (ADV) to other nodes utilizing CSMA/CA protocol. Non-broker-bridge nodes are required to keep their receivers active during this phase in order to receive the advertisements from all the brokers bridge.

(3) Cluster formation : Nodes establish communication between themselves to form clusters, with one node designated as the CH resulting in disjoint clusters.

After receiving the ADV, each node determines the cluster to join for the current round based on the received signal strength (RSSI) of the advertisement.
Once the decision is made, the node sends a JOIN request to the corresponding cluster head (CH) to become a member. The node uses a CSMA/CA protocol to transmit this information to the CH. The CH receives all the JOIN requests from the nodes willing to join the cluster and creates a TDMA schedule, specifying when each node can transmit. This schedule is then broadcasted to all nodes within the cluster. Throughout this process, all cluster head nodes must keep their receivers active.

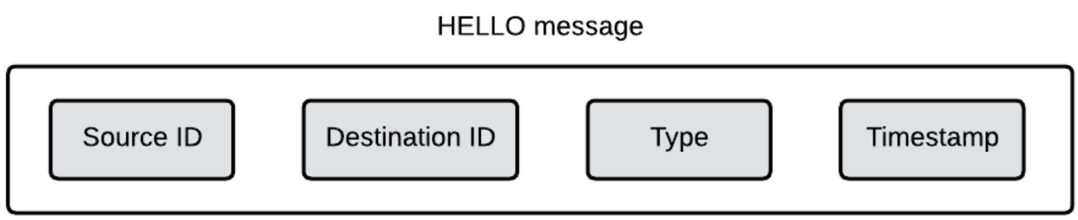

Figure 4.1: HELLO message structure.

The Publish/Subscribe routing phase : In this phase, which is the second phase of our protocol, the broker bridges play a crucial role in handling data messages between the cluster publishers and subscribers (forwarding data to subscribed clients when it is published in their respective topics). Additionally, the broker bridges aggregate the published data and publish it to the central broker, acting as a client to the central broker. They also subscribe to a topic on the central broker in order to receive data from other clusters.

Once the clusters are created and the TDMA schedule is fixed, data transmission can begin. During their allocated transmission time (time slot), publisher nodes collect data from its sensors before publish it in a specific topic on the chosen broker, while the subscribers node subscribe to a specific topic on the chosen broker in order to receive data from it. This transmission uses a minimal amount of energy chosen based on the strength of the advertisement received from cluster-head. The radio of each publisher and subscriber node must be turned off until the node's allocated time slot, thus minimizing energy dissipation in these nodes. The broker node must keep its receiver on to receive all the publish message from the publisher nodes in the cluster and then resend it to corresponding cluster subscribers.

When all the data has been received, the CHs which are in the same time broker bridge (BB) nodes perform signal processing functions to compress and aggregate the data into a single signal, in order to minimize the amount of data transmission and conserve energy. This composite signal is published in a topic on the central broker which forward it to the corresponding subscribers.

Our protocol includes mechanisms for energy-efficient routing and cluster formation to minimize energy usage.

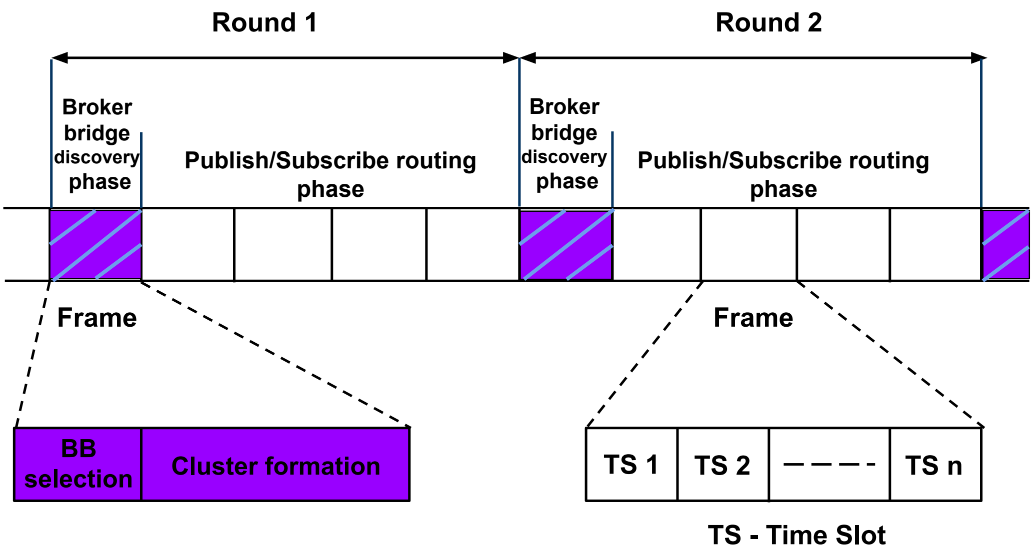

Figure 4.2: Explanatory schema of the protocol phases.

Figure. 4.2 presents a comprehensive overview and explanation of the aforementioned phases, showing two consecutive rounds. Each round is divided into broker bridge discovery and publish/subscribe routing phases, providing a clear visualization of the process.

Remark. Nodes will assume the role of a CH at most once within a span of $\frac{1}{P}$ rounds. CHs in round r cannot serve as cluster-heads for the next $\frac{1}{P} - 1$ rounds, this increases the probability for the remaining nodes to become CHs. After $\frac{1}{P}$ rounds, all nodes regain their eligibility to function as CH once again.

4.2.2 Algorithm of the Protocol

The entire proposed scheme is presented in detail through the following algorithms : Algorithm. 4.1 and Algorithm. 4.2.

Algorithm 4.1: Broker bridge discovery phase.

```
/* Clustering and Scheduling phase */
initialization of IOT ; /* HELLO messages */
calculate energy average E of each node n_i..n_n(E_residual, E_total);
calculate density D of each node n_i..n_n(Edge_actual, Edge_potential);
calculate T_i(t) of each node (n_i..n_n ) ; /* Based on E and D */
n ← RundomNumber(0, 1);
if n < P_i(t) then
    isCH ← True;
    sendADVmsg();
    receiveJOINreq();
    sendTDMA();
else
    receiveADVmsg();
    chooseCH();
    sendJOINreq();
    receiveTDMA();
end
```

Algorithm 4.2: Publish/Subscribe routing phase.

```
/* Data transmission phase */
if Publisher then
    publish sensing data() ; /* To the CH (Broker-Bridge) at thier slot */
else
    if Subscriber then
        subscribe to topic() ; /* To the CH (Broker-Bridge) at thier slot */
        receive data();
        Make data-based decision();
    end
end
if Broker-Bridge then
    subscribe to subscribers topic();
    aggregate published data();
    send aggregation();
    receive data() ; /* From the central broker */
    forward data to cluster subscribers();
else
    if Central-Broker then
        receive aggregated data();
        forward data to sub() ; /* To corresponding subscribers */
    end
end
```

4.2.3 Protocol Workflow Chart

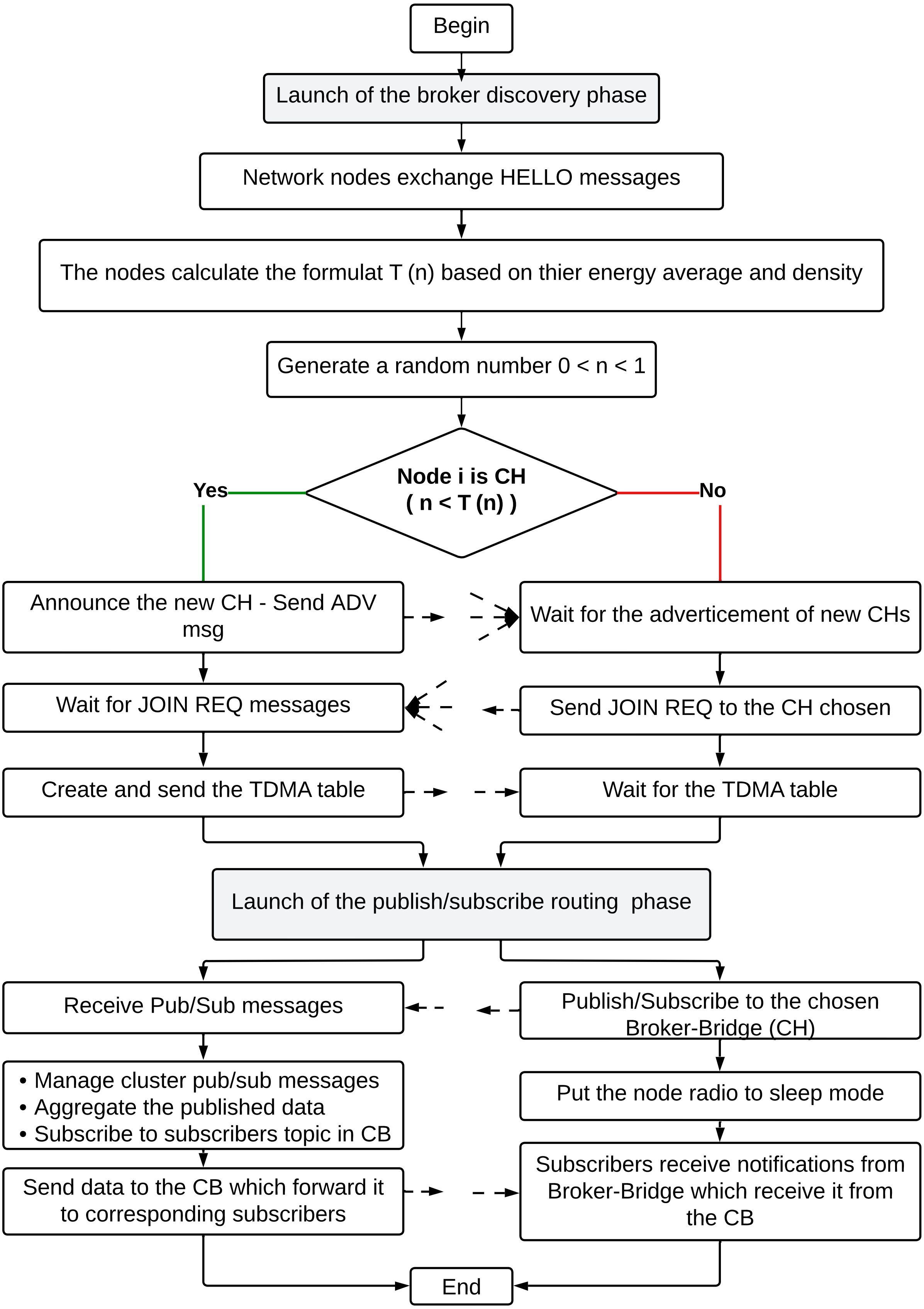

Figure 4.3: Workflow chart of the protocol (depicting one round).

Figure. 4.3 shows the diagram illustrating the sequence of steps in the proposed protocol.

4.3 Energy Model used

The communication between the sensors of the network is the most sensitive part in the WSN and IOT, see that in the anatomy of a sensor this unit is composed of a transmitter/receiver which consumes energy compared to the other units of the system (Figure. 4.4). Therefore, in order to reduce the level of consumed energy, it is better to rely on nodes that have more energy (compared to other nodes) to carry out the process of transmitting and receiving data.

The CC2420 [32] radio model was used in our simulation. It is a widely used radio transceiver in WSNs [33]. It has a specific energy model that considers factors such as transmit power levels, data rate, modulation schemes, and receive modes to estimate energy consumption during communication operations.

This makes it possible to get as close as possible to reality with the vagaries of a communication medium, with realistic node behavior in particular in terms of access to the radio.

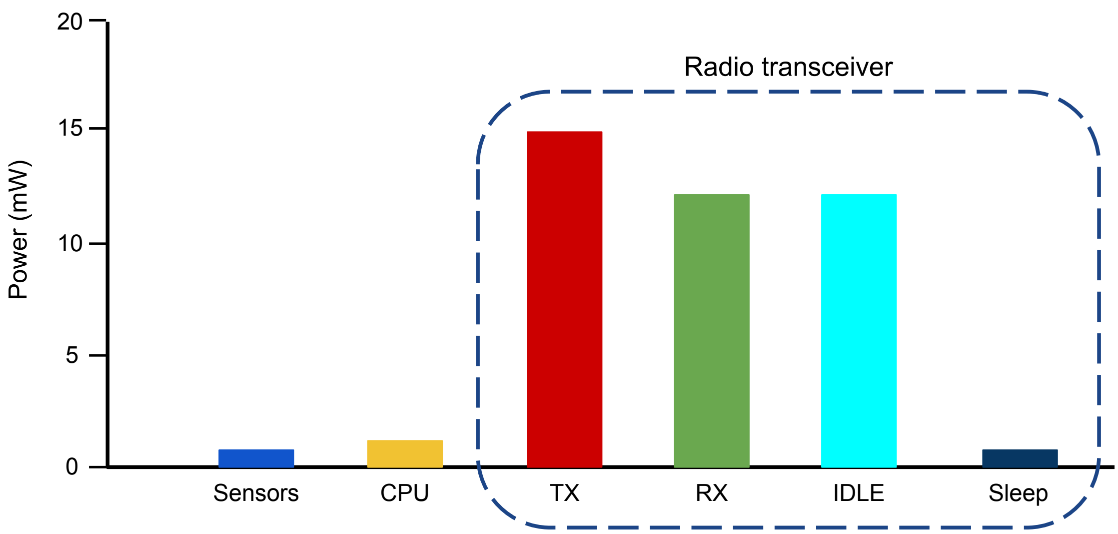

Figure 4.4: Energy consumption in acquisition, processing, and communication [34].

4.4 Simulation

This section provides an overview of the simulation aspect in the study. It covers the tools used and the parameters considered during the simulation process.

4.4.1 Simulation Tools

Two prominent software tools, OMNet++ and Castalia, are employed to model and simulate the behavior of the network environment.

(a) OMNET++ : Is an open-source, modular (allowing new model integration), and extensible simulator used for modeling and simulating various types of networks.

C++, NED, and ini files languages are utilized for simulation modeling. C++ is used for implementing module behavior, while NED is used for defining the structure and connectivity of the simulation model, and ini files are used for configuration. These languages work together to provide a comprehensive framework for simulation modeling.

You can find the steps to install OMNET++ in the Appendix A.1.

OMNeT uses simple modules to represent individual components with input/output gates, specific functionality and internal behavior. Compound modules, on the other hand, serve as containers to multiple simple modules or other compound modules. This hierarchical structure allows for the organization and connectivity of components into larger system structures, enhancing modularity and reusability in simulation models (Figure. 4.5).

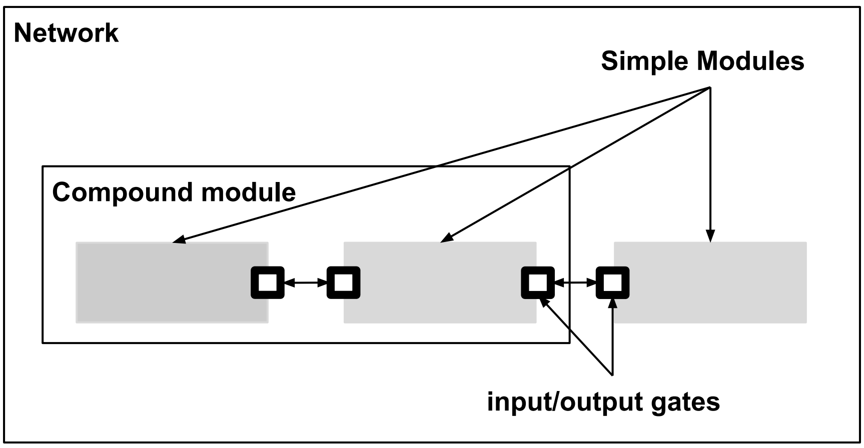

Figure 4.5: Simple and Compound modules in OMNET [35].

While the OMNeT++ simulator itself is not specifically designed for WSNs, several platforms and simulators try to address this lack like "Mixim" and "Castalia".

(b) Castalia : Castalia [36] is a network simulator specifically designed for WSN and IOT applications. It is built on top of the OMNeT++ simulator and provides a comprehensive platform for modeling and simulating WSN protocols, algorithms and applications. Its main modules are the node, physical processes and the wireless channel (Figure. 4.6), the nodes connect to each other via the wireless channel module, and they are also linked through the physical processes they monitor.

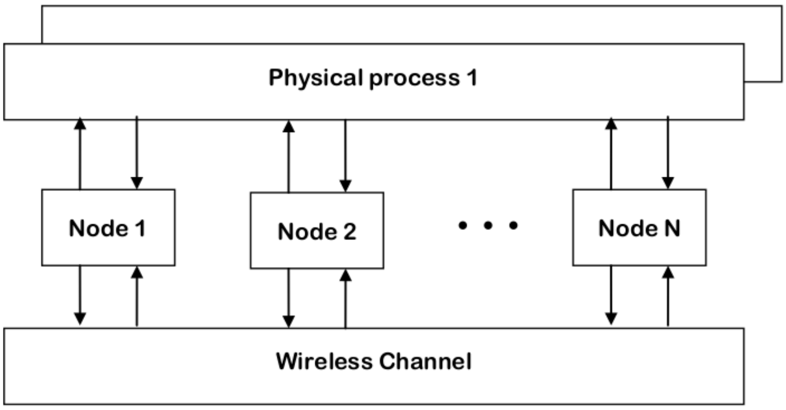

Figure 4.6: The connections of the modules under Castalia [37].

In Castalia, the node module is a composite module with an internal structure shown in Figure. 4.7. Solid arrows represent message passing, while dotted arrows indicate simple function calls. For example, most modules call a resource manager module function to report that power has been consumed.

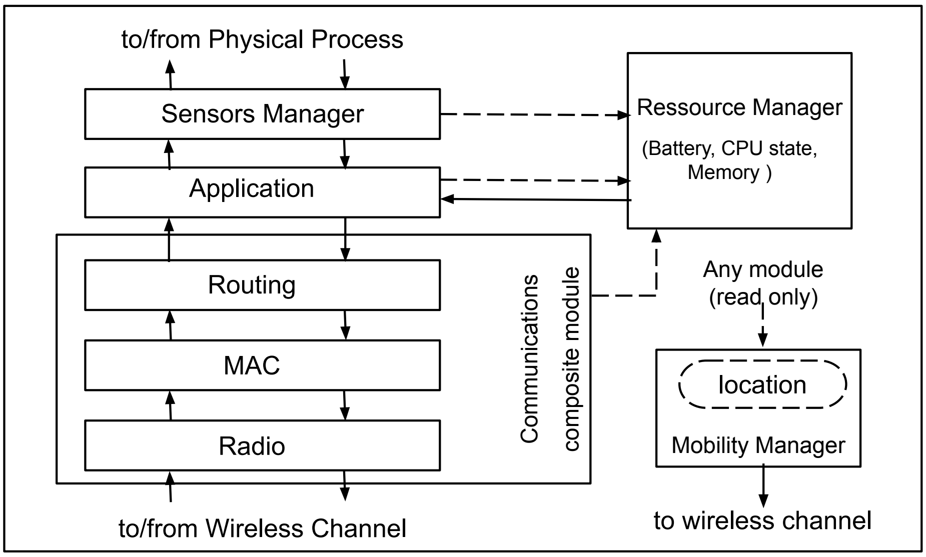

Figure 4.7: The node composite module [38].

4.4.2 Simulation Parameters

To evaluate our proposal routing protocol, we used the OMNet++ simulator and Castalia framework. For an environment in a 100m x 100m area, where the simulation time is about 1000 seconds.

Simulation parameters are summarized in Table. 4.4.

Table 4.4: Simulation parameters.

Parameter	Value
Simulation area	100 x 100
Simulation time (s)	200
Number of nodes	[50, 100, 150, 200, 250, 300]
Position of nodes	Between (0,0) and (100,100)
Number of base stations	1
The location of the base station	(1 , 1)
Base station ID	0
The initial energy of nodes	17000 (mJ)
Radio model	CC2420
Node deployment	Random
Duration of a round (s)	20
The percentage of CH in the topology	5%
Physical Medium	Wireless

The simulation was carried out on a computer with the following configuration and software environment (Table. 4.5) :

Table 4.5: Hardware and software requirements used.

Parameter	Value
Processor	Intel(R) Celeron(R) CPU N3060 @ 1.60GHz 64 bits
RAM	4 GB
Hard disk	HDD 500 GB
Operating system	UBUNTU 20.04
Simulator	OMNET++ (version 4.6)
Framework	Castalia (version 3.3)

4.5 Analytical Results and Discussion

A clustering routing protocol based on LEACH and MQTT protocols with CHs acting as broker bridges have a significant impact on the energy efficiency of message routing between publishers and subscribers. By having the CH acting as broker bridge to the central broker, it reduces the amount of energy needed by the individual sensor nodes to transmit their publish/subscribe messages and optimizes the routing process. Additionally, the CHs perform data aggregation to reduce energy consumption. The CB at the base station efficiently manage the flow of messages between publishers and subscribers.

To validate the effectiveness of our newly proposed protocol in this study, we conducted a performance evaluation of this routing protocol across various scenarios, comparing it to the LEACH protocol. Our evaluation focused on crucial metrics such as energy efficiency, network lifetime, and stability period, considering the impact of time and the network size.

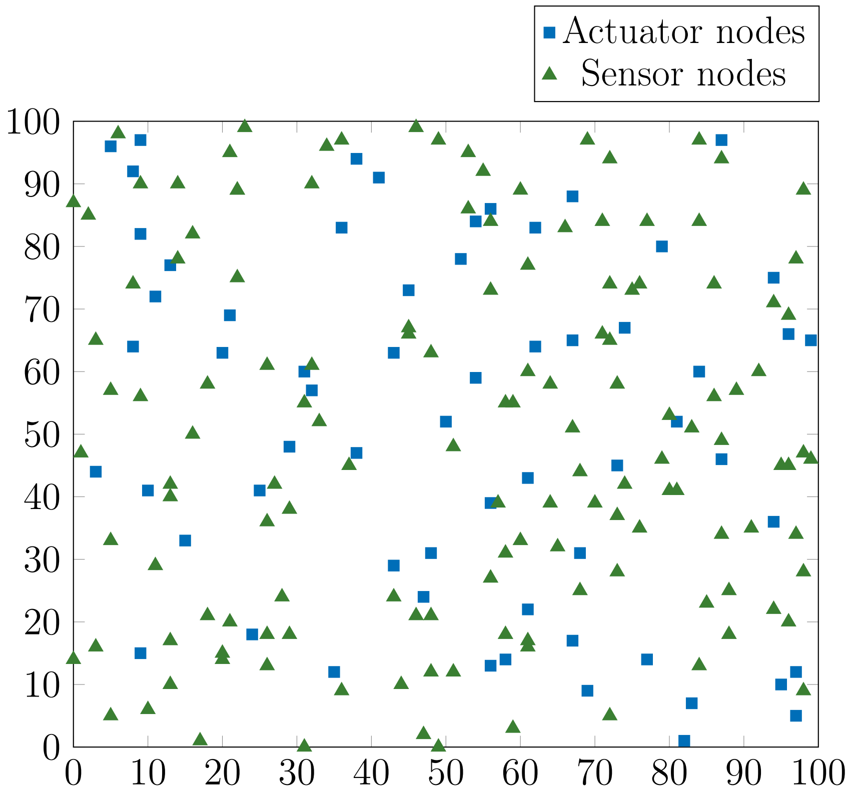

Figure 4.8: Heterogeneous nodes deployment in 100 x 100 area.

4.5.1 Energy Evaluation

Given the energy limitations of the sensors, it is crucial to minimize energy consumption at all levels to extend the network's lifespan. Thus we conducted a number of tests :

(i) Energy vs. time : In order to test the adaptation of our approach on a large scale, we have simulated the routing protocol in different periods.

The conducted simulations using the random 200-node network shown in Figure. 4.8 demonstrate that the proposed protocol achieves a lower average consumed energy (Figure. 4.9) and a higher average remaining energy (Figure. 4.10), compared to LEACH protocol. This outcome can be attributed to the reduced number of messages transmitted throughout the network, the selection of cluster heads based on node density and residual energy to be as relay nodes to the central broker, and also due to the radio sleep mode during inactive periods.

Where nodes with higher energy and density are more likely to be chosen as cluster heads to play the role of broker bridges, while non-cluster head nodes join the nearest cluster head with better energy and density. This leads to a reducing energy consumption and improved performance of the protocol.

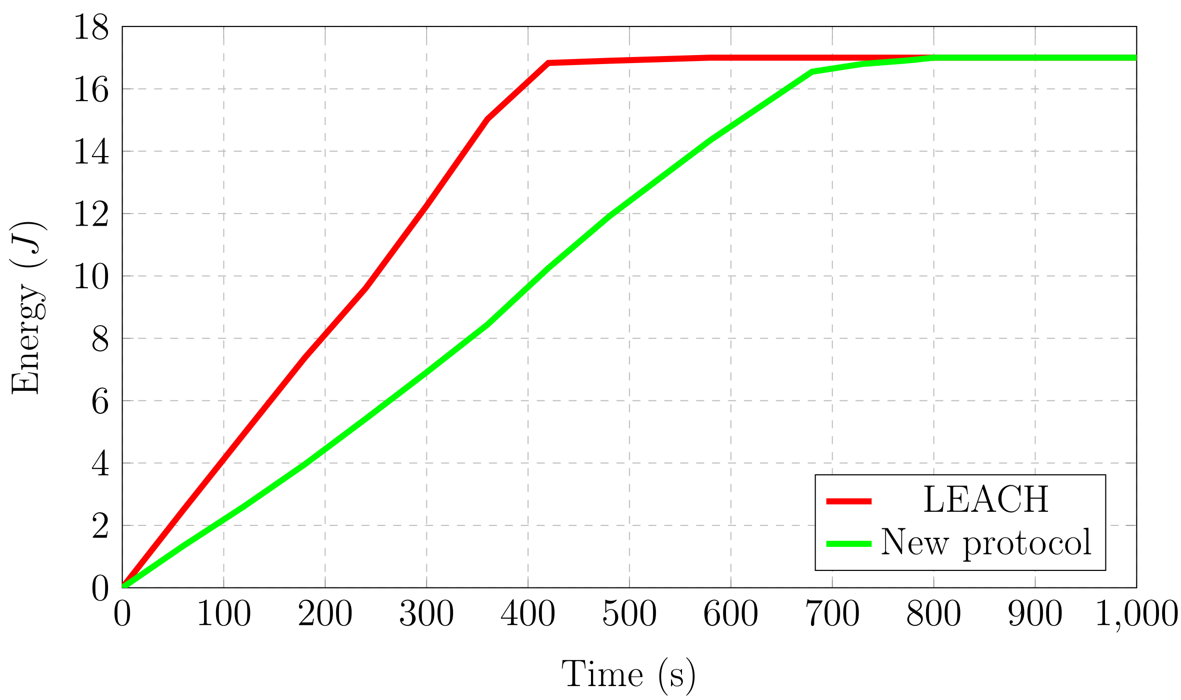

Figure 4.9: Consumed energy vs. time.

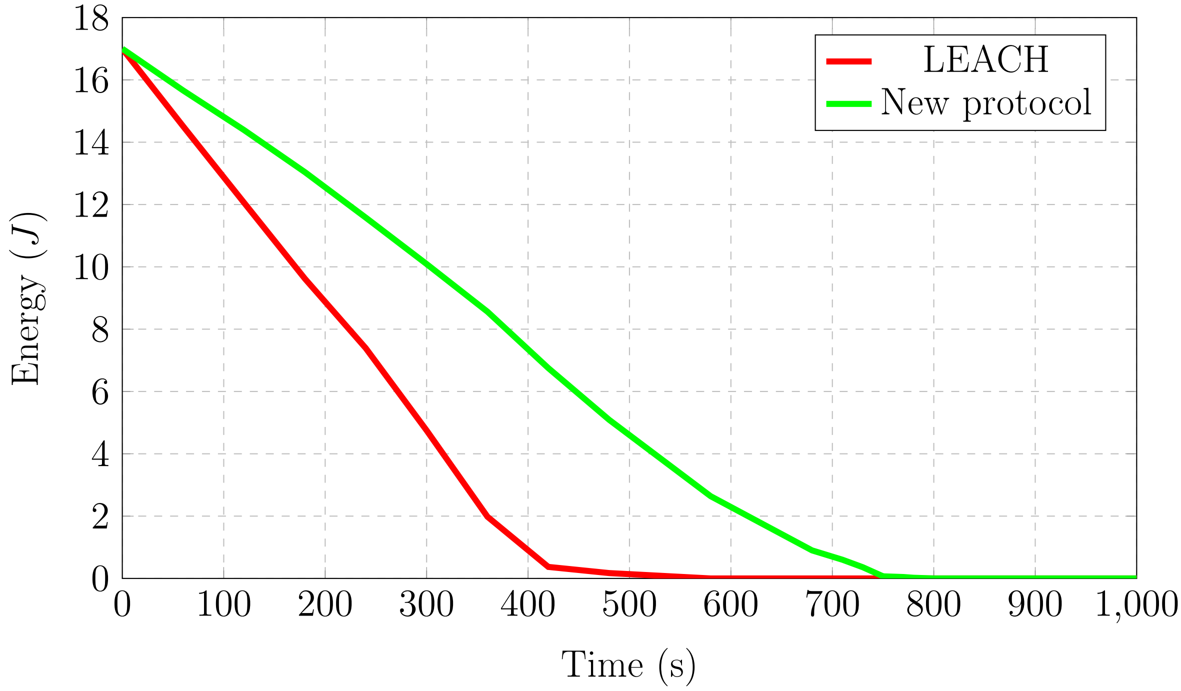

Figure 4.10: Residual energy vs. time.

(ii) Energy vs. number of nodes : In order to test the adaptation of our approach on a large scale, we have simulated the routing protocol in networks of different sizes.

Figure. 4.11 illustrates the energy consumption in relation to the number of nodes in the network over a 200-second period, while Figure. 4.12 depicts the remaining energy in relation to the number of nodes in the network for the same duration. Our proposed approach exhibits significant energy savings compared to the LEACH protocol. This outcome can be attributed to the strategic distribution of cluster-heads throughout the entire network, achieved through meticulous selection based on proposed mathematical formulas, as well as the lightweight nature of Pub/Sub MQTT messages and the usage of radio sleep mode during inactive periods.

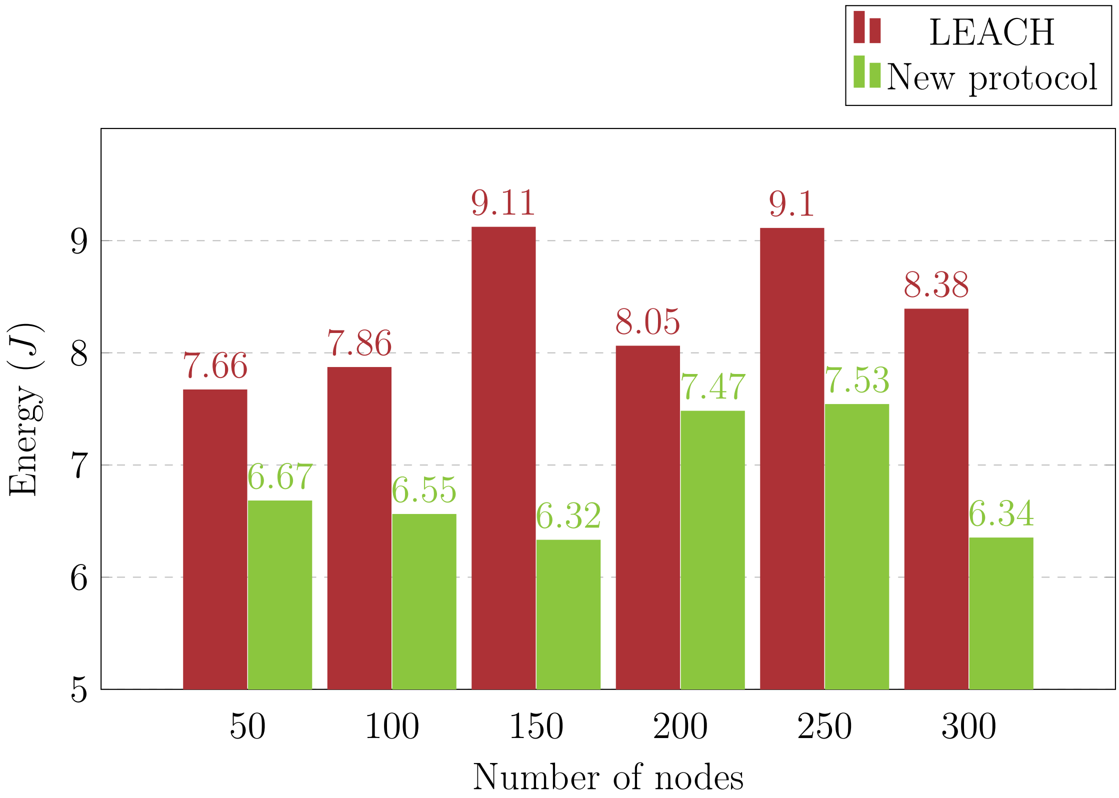

Figure 4.11: Energy consumption vs. number of nodes.

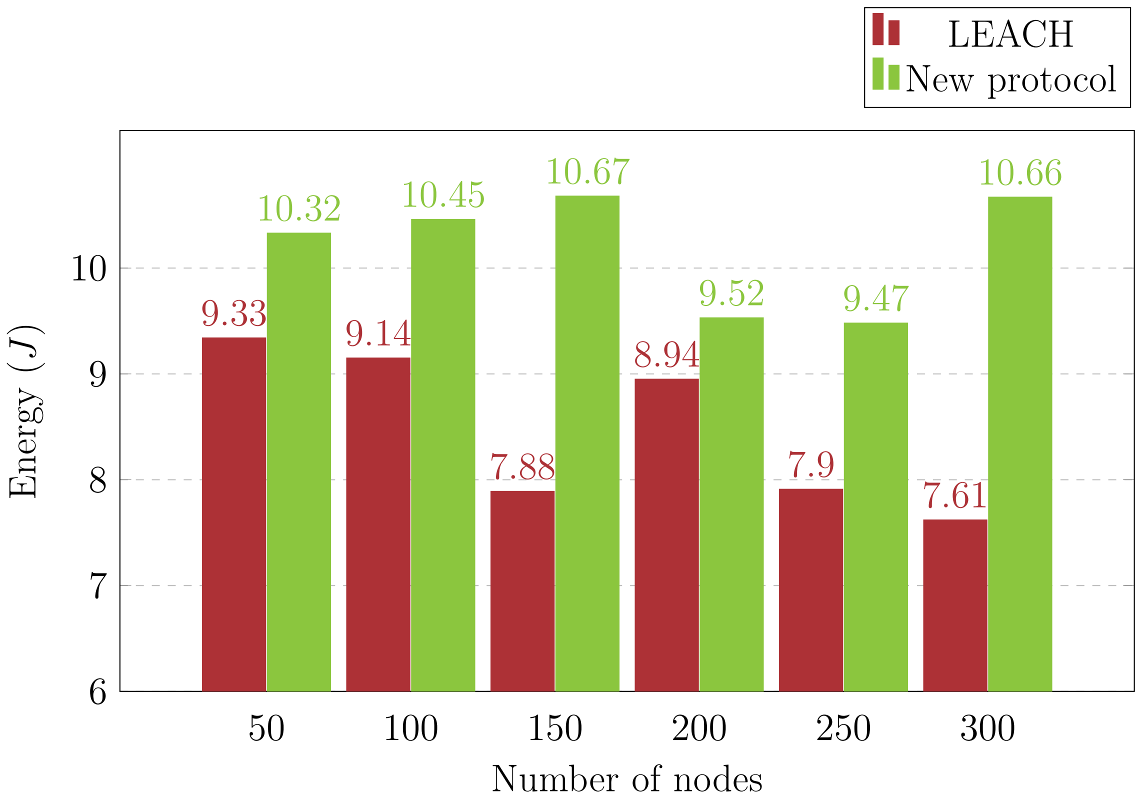

Figure 4.12: Residual energy vs. number of nodes.

4.5.2 Network Stability Period

The network stability period refers to the duration it takes for the first node to cease functioning [39]. The term of the first dead node (FDN) refers to the point at which the first sensor node becomes inactive, while the half of the nodes dead (HDN) denotes the moment when half of the sensor nodes in the network die. Through our simulations, we observed that the proposed routing protocol exhibits superior stability when compared to the LEACH protocol, as demonstrated in Figure. 4.13.

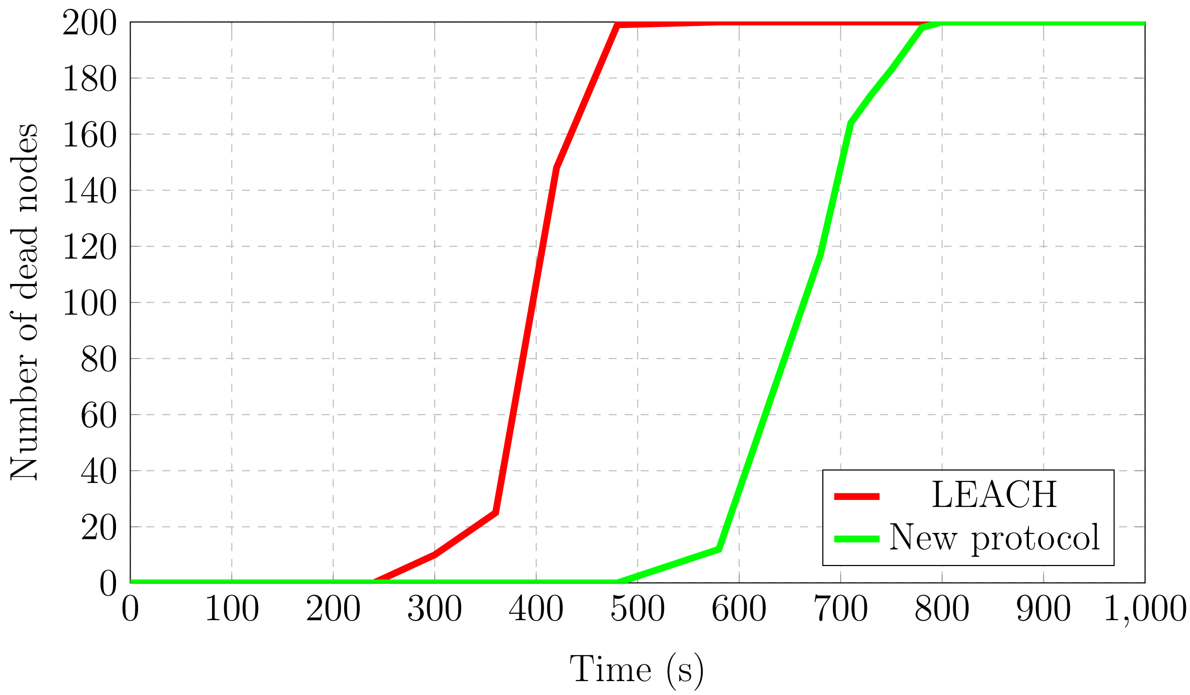

Figure 4.13: Network stability period

The simulation results presented in Figure. 4.13 demonstrate the positive impact of the proposed algorithm on the network stability period. The FDN and HDN occur much later with the new protocol compared to the LEACH protocol, indicating improved network stability.

With the new protocol, the first dead node occurred at time 490, while the half of the nodes dead occurred at time 680. In contrast, with the LEACH protocol, the first dead node occurred at time 250 and the HDN occurred at time 400. This improvement in stability period can be attributed to the efficient energy consumption in the proposed routing protocol.

4.5.3 Network Lifetime

The network lifetime is defined as the duration from the initiation of communication until the last node becomes inactive (from the start of network activity, to the point when the network becomes non-functional) [40]. Our simulation results indicate that our protocol surpasses LEACH in terms of network lifetime, as demonstrated in Figure. 4.14.

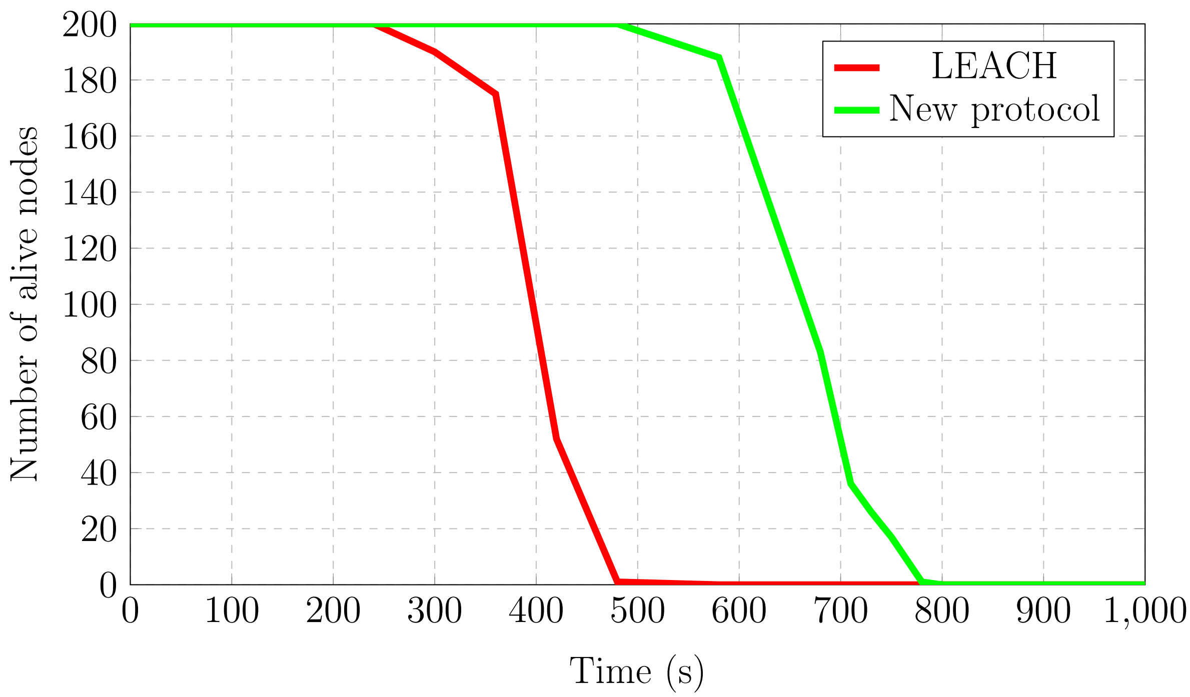

Figure 4.14: Network lifetime

As illustrated in the Figure. 4.14, we notice that the lifetime of our approach increases compared to standard LEACH. The efficient energy consumption in the proposed routing protocol contributes to this enhancement.

Therefore, our clustering routing protocol is effective where energy efficiency, scalability, and lifetime are critical factors.

4.6 Conclusion

In conclusion, this chapter has presented our proposed routing protocol for a publish/subscribe system (MQTT system) in IOT based on wireless sensor networks in details.

The results of the simulation conducted using OMNet++ simulator and Castalia framework argued in favor of the efficiency performance of our routing protocol.

These results have shown that our contribution is effective in achieving significant energy savings and improved network lifetime.

Overall, the findings of this chapter set the stage for further investigation and refinement.

Chapter 5

General Conclusion

Contents

5.1 Summary of our Work

This work focused on the study of routing in IOT and WSNs, comprising several key steps. Firstly, we provided an introduction to IOT and WSN networks, highlighting their significance in enabling seamless connectivity and data exchange. Additionally, we delved into hierarchical protocols, with LEACH serving as a model for our analysis. Furthermore, we explored the concept of publish/subscribe systems, with MQTT serving as a prominent example. In the subsequent step, we reviewed relevant literature, offering a concise overview and comparison of related papers. Finally, we presented our approach and contribution to the field.

In particular, in this thesis we have proposed our new routing protocol for publish/-subscribe systems in WSN-based IOT networks for agricultural applications. Our protocol aims to improve the delivery of publish and subscribe messages by intelligently routing them between publishers, subscribers, and a central broker. By incorporating the cluster-heads from the LEACH protocol as broker bridges, we leverage their existing roles to enhance message routing between publishers, subscribers, and the central broker.

Throughout our research, we have conducted extensive simulations using OMNet++ simulator and Castalia framework to evaluate the performance of our protocol. The results have demonstrated the efficiency and effectiveness of our routing protocol in terms of energy, network lifetime, and stability period.

Our work contributes to the advancement of IOT and WSN communication protocols and inspires future research in this field.

5.2 Future Perspectives

To expand upon the achievements of this work, it is crucial to consider several future perspectives for designing novel approaches. These points outline the directions for future work aimed at further advancing the research :

- Implementation of a second scenario involving mobile nodes, such as the utilization of mobile actuator nodes.
- Integration of security parameters into our routing protocol (enabling MQTT secure communication through SSL/TLS).
- Conducting performance analysis of our protocol after a real-world implementation.

Bibliography

[1] Sabir Awan, Sheeraz Ahmed, Fasee Ullah, Asif Nawaz, Atif Khan, and other. Iot with blockchain : A futuristic approach in agriculture and food supply chain. *Wireless Communications and Mobile Computing*, 2021 :01–14, 2021.

[2] B. Kiruthika and Shyamala Bharathi P. Intelligent dynamic trust secure attacker detection routing for wsn-iot networks. *Mathematical Biosciences and Engineering*, 20(2) :4243–4257, 2022.

[3] Anwar D. Alhejaili and Omar H. Alhazmi. Lightweight algorithm for mqtt protocol to enhance power consumption in healthcare environment. *Journal on Internet Of Things (JOIT*, 4(1) :21–33, February 2022.

[4] Roberto Baldoni, Leonardo Querzoni, Antonino Virgillito, and Tarkoma Sasu. Distributed event routing in publish/subscribe communication systems : a survey. *Department of Computer Science and Systems University of Rome "La Sapienza"*, June 2015.

[5] Internet of Things (IOT) ,[online] available. https://www.oracle.com/internet-of-things/. accessed : 25 February 2023.

[6] Ian F Akyildiz, Weilian Su, Yogesh Sankarasubramaniam, and Erdal Cayirci. A survey on sensor networks. *IEEE Communications magazine*, pages 102–114, August 2002.

[7] S.R.Jino Ramson and D.Jackuline Moni. Applications of wireless sensor networks – a survey. *Proceedings of IEEE International Conference on Innovations in Electrical, Electronics, Instrumentation and Media Technology ICIEEIMT 17*, page 325–329, November 2017.

[8] Dirk Trossen, Dana Pavel, Glenn Platt, Joshua Wall, Philip Valencia, et al. Sensor networks, wearable computing, and healthcare applications. *IEEE Pervasive Computing*, 6(2) :58–61, April 2007.

[9] Ishfaq Ahmad, Khalil Shah, Saif Ullah, et al. Military applications using wireless sensor networks : A survey. *International Journal of Engineering Science and Computing*, 6(6) :7039–7043, June 2016.

[10] Andrea J Goldsmith and Stephen B Wicker. Design challenges for energy-constrained ad hoc wireless networks. *IEEE Wireless Communications*, 9(4) :08–27, August 2002.

[11] Ala Eddine Boulifa and Mohammed Habes. Étude comparative entre deux protocoles de routage géographiques sans balises blr et boss dans les réseaux wsns. *Memoir of Master, University Kasdi Merbah Ouargla Spécialité Administration and Security of Netwok*, page 01–60, 2020.

[12] Riad Aouabed. Cross-layer routing protocols in wireless sensor networks. *Doctoral Thesis, Ferhat Abbas University Setif*, page 01–109, 2023.

[13] YACINE CHALLAL. Réseaux de capteurs sans fils. *http ://creativecommons.org/licenses/by-nc/2.0/fr/*, pages 01–103, November 2008.

[14] Gagandeep Kaur, Prasenjit Chanak, and Mahua Bhattacharya. Energy-efficient intelligent routing scheme for iot-enabled wsns. *IEEE Internet of things Journal*, 8(14) :11440–11449, July 2021.

[15] Chaker Abdelaziz Kerrache. Le routage. *Slides of courses, Mobile networks module, Computer Science department, Ammar Thelidji University Laghouat*, pages 01–60, 2021.

[16] Ali Benzerbadj. Approche inter-couches pour l'économie d'énergie et la fiabilité dans les réseaux de capteurs sans fil dédiés aux applications critiques de surveillance. *Université de Bretagne occidentale - Brest*, pages 01–135, 2018.

[17] Alakesh Braman and Umapathi G. R. A comparative study on advances in leach routing protocol for wireless sensor networks : A survey. *International Journal of Advanced Research in Computer and Communication Engineering (IJARCCE)*, 3(2) :5683–5690, February 2014.

[18] Villalba LJ, Orozco AL, Cabrera AT, and Abbas CJ. Routing protocols in wireless sensor networks. *Sensors (Basel, Switzerland)*, 9(11) :8399–8421, 2009.

[19] Mehiaoui Amina. Etude comparative entre les deux protocoles de routage LEACH et PEGASIS dans les réseaux de capteurs sans fil. *Memoir of Master, Department of telecommunication, University Abou Bekr Belkaid Tlemcen*, pages 01–61, March 2015.

[20] Guemmadi Sabrina. Adaptation de clustering phase de protocole de routage « leach » pour les réseaux de capteurs sans fil. *Memoir of Master, University Larbi Tebessi - TEBESSA*, pages 01–63, 2020.

[21] Wendi Rabiner Heinzelman, Anantha Chandrakasan, and Hari Balakrishnan. Energy-efficient communication protocol for wireless microsensor networks. *Proceedings of the 33rd Hawaii International Conference on System Sciences ,IEEE*, pages 01–10, 2000.

[22] Ivan Vaccari, Giovanni Chiola, Maurizio Aiello, Maurizio Mongelli, and Enrico Cambiaso. Mqttset, a new dataset for machine learning techniques on mqtt. *Sensors (Basel, Switzerland)*, 20(22) :6578, 2020.

[23] MQTT protocol ,[online] available. https://www.javatpoint.com/mqtt-protocol. accessed : 15 April 2023.

[24] Laouari Rahma and Besline Nassima. Routage hiérarchique dans les rcsf. *Memoir of Master, Computer Science department, Mouloud Mammeri University Tizi-ouzou*, pages 01–73, 2013.

[25] Amziane Razika and Mahmoudi Djedjiga. Simulation d'un protocole leach. *Memoir of Master, Computer Science department, Mouloud Mammeri University Tizi-ouzou*, pages 01–70, 2020.

[26] Akramul Azim and Mohammad Mahfuzul Islam. Hybrid leach : A relay node based low energy adaptive clustering hierarchy for wireless sensor networks. *Proceedings of the IEEE 9th Malaysia International Conference on Communications (MICC) 15 -17 December 2009 Kuala Lumpur Malaysia*, pages 911–916, 2009.

[27] Vergin Raja Sarobin M and Linda Ann Thomas. Improved leach algorithm for energy efficient clustering of wireless sensor network (wsn). *International Journal of Technology (IJOT)*, 1 :50–60, 2016.

[28] Paolo Costa, Cecilia Mascolo, Mirco Musolesi, and Gian Pietro Picco. Socially-aware routing for publish-subscribe in delay-tolerant mobile ad hoc networks. *IEEE Journal on selected areas in communications*, 26(5) :748–760, June 2008.

[29] Rahul Johari, Shubhi Bansal, and Kanika Gupta. Routing in iot using mqtt protocol. *12 th International Conference on Computational Intelligence and Communication Networks*, pages 205–290, 2020.

[30] Eyman Fathelrhman Ahmed Elsmany, Mohd Adib Omar, Tat-Chee Wan, et al. Eesra : Energy efficient scalable routing algorithm for wireless sensor networks. *IEEE Access*, 7 :96974–96983, 2019.

[31] OMNET++ ,[online] available. https://omnetpp.org/. accessed : 02 January 2023.

[32] CC2420 ,[online] available. https://www.ti.com/product/CC2420. accessed : 18 March 2023.

[33] Kannan Srinivasan, Prabal Dutta, Arsalan Tavakoli, and Philip Levis. An empirical study of low-power wireless. *ACM Transactions on Sensor Networks*, 6(2) :01–49, February 2010.

[34] Chaker Abdelaziz Kerrache. Les protocoles mac. *Slides of courses, Mobile networks module, Computer Science department, Ammar Thelidji University Laghouat*, pages 01–47, 2021.

[35] Espen Nilsen. Energy consumption investigations in wsn using omnet++. *Master thesis, University of Oslo*, pages 01–81, May 2013.

[36] An OMNeT-based simulator for low-power wireless networks such as Wireless Sensor Networks and Body Area Networks ,[online] available. https://github.com/boulis/Castalia. accessed : 08 January 2023.

[37] Dimosthenis Pediaditakis, Yuri Tselishchev, and Athanassios Boulis. Performance and scalability evaluation of the castalia wireless sensor network simulator. pages 01–06, 2010.

[38] Athanassios Boulis. Castalia, a simulator for wireless sensor networks and body area networks version 3.2 user's manual. pages 01–120, March 2011.

[39] Ammar S. Al-Zubaidi, Ahmad Alauddin Ariffin, and Abdulrahman K. Al-Qadhi. Enhancing the stability of the improved-leach routing protocol for wsns. *Journal of Information and Communication Technology (ICT) Research and Applications*, 12(1) :01–13, 2018.

[40] Alim Al Islam, Chowdhury Sayeed Hyder, Humayun Kabir, and Mahmuda Naznin. Stable sensor network (ssn) : A dynamic clustering technique for maximizing stability in wireless sensor networks. *Wireless Sensor Network*, 2 :538–554, 2010.

[41] Lesly Maygua-Marcillo, Luis Urquiza-Aguiar, Martha Paredes-Paredes, and Pablo Barbecho Bautista. Deployment of omnet++. *Preprints.org*, pages 01–10, 2018.

Appendices

Appendix A

Simulation Script and Tools

Contents

A.1 OMNET Installation

To install OMNET++ in Ubuntu, follow these steps [41] :

1. Install all packages needed to compile and run OMNet++, you can run the following command to install all packages for multiple platforms :
`$ sudo apt-get install build-essential gcc g++ bison flex perl\tcl-dev tk-dev libxml2-dev zlib1g-dev default-jre\doxygen graphviz libwebkitgtk-1.0-0`

2. Download `omnetpp-4.6-src.tgz` from : `https://omnetpp.org/`

3. Unzip the file : `$ tar xvfz omnetpp-4.6-src.tgz`

4. Enter the omnetpp-4.6 folder : `$ cd omnetpp-4.6`

5. Set the environment variables in a non-permanent way through the following command : `$. Setenv`
Or permanently through the modification of `.bashrc` file (`$ sudo nano ~/.bashrc`) in the home directory by adding the following line :
`export PATH=$PATH:$HOME/omnetpp-4.6/bin` , then save it.

6. In the omnetpp-4.6 directory type the following command : `$ ./configure`

7. When ./configure ends, you can compile OMNeT++ through the command : `$ make`

8. You can launch the OMNeT++ simulation IDE by typing the following command in the terminal : `$ omnetpp`

A.2 Castalia

The Castalia 3.3 version works with OMNeT versions 4.3 to 4.6. The Linux operating system is recommended for using Castalia.

To install Castalia in Ubuntu, follow these steps :

1. Download Castalia from the site : `https://github.com/boulis/castalia`
2. Extract the Castalia folder : `$ tar -xvzf Castalia-master.tar.gz`
3. Compile Castalia :
```
$ cd Castalia/Castalia
$ ./makemake
$ make
```
   If the access to the script is refused, make sure you have the right permissions to the file. If not, type `$ chmod u+x makemake` and then try again.

Castalia uses OMNET features to define the architecture of a sensor node. All definitions are described and implemented in the `Castalia/src` directory.

Castalia is a command line tool, but it can work in the graphical mode after some modifications. To use Castalia with OMNeT's GUI, follow these steps :

1. Create new Omnet++ Project : File -> New -> Omnet++ Project.

2. Choose a project name. For example "castaliaTest" and click next.

3. Select Empty project and click Finish.

4. Right click on the newly created Omnet++ project (castaliaTest) and choose Import.

5. Select General -> File System and click Next.

6. Click Browse and search for your Castalia-master directory (or whatever name you have for castalia's root) and click OK.

7. Click on "Select All" button and then Finish.

8. You should now see the directory hierarchy under the project castaliaTest.

9. Right click above the project name (castaliaTest) -> Properties.

10. In the emerging window, expand OMNeT++ on the left side menu, and then select "NED Source Folders".

11. Expand the project name and mark "Simulations" and "src" directories as NED Source Folders. Click OK.

12. Again in the IDE main view, navigate into the project (castaliaTest) -> src .

13. Double click on `SensorNetwork.ned` file. Select the "source" tab and add ".*" to the three import lines. You should end up with something like this :

    ```
    import wirelessChannel.*;
    import physicalProcess.*;
    import node.*;
    ```

14. Save the file.

15. Now open src -> node -> Node.ned file and go to source tab.

16. Add ".*" at the end of the import mobility manager line. You'll end up with this :

    ```
    import node.mobilityManager.*;
    ```

17. Save the file. Clean and build the project.

A.3 Algorithmic Scripts

Our proposed routing protocol is represented by the class named `NewprotocolRouting`, which is defined as follows :

```
//***************************************************************************
//*  Header file (.h) of a .cc file describe the behaviour of the module.
//***************************************************************************/
#ifndef _NEWPROTOCOLROUTING_H_
#define _NEWPROTOCOLROUTING_H_

#include <queue>
#include <vector>
#include <omnetpp.h>
#include <algorithm>
#include <string>
#include <math.h>
#include <stdlib.h>
#include <stdio.h>
#include <iostream>
#include <sstream>
#include "VirtualRouting.h"
#include "VirtualApplication.h"
#include "NewprotocolRoutingPacket_m.h"
#include "ApplicationPacket_m.h"
#include "NoMobilityManager.h"
#include "ResourceManager.h"

using namespace std;

enum LeachRoutingTimers {
    START_ROUND = 1,
    SEND_HELLO = 2,
    CALCULATE_PROBA = 3,
    SEND_ADV = 4,
    JOIN_CH = 5,
    MAKE_TDMA_Crtl = 6,
    START_SLOT = 7,
    END_SLOT = 8,
};
struct CHInfo
{
    int src;
    double rssi;
};
struct CMember
{
```

```
    string src;
    string topic;
};
class NewprotocolRouting : public VirtualRouting {
private:
    string applicationID;
    int advPacketSize;
    int tdmaCrtlPacketSize;
    int joinPacketSize;
    double maxPower;
    double sensibility;
    double aggrConsumption;
    double slotLength;
    int clusterLength;
    double percentage;
    double probability;
    double roundLength;
    int roundNumber;
    int dataSN;

    double energyConsumed;
    double initial;
    double remainEnergy;

    int helloPacketSize;
    int cbdataPacketSize;
    int chdataPacketSize;
    int subdataPacketSize;
    int pubdataPacketSize;

    double density;
    double actual = 0;
    double total;
    int totalSNnodes;
    string topic;
    string type;  /*    publisher or subscriber     */
    vector <CMember> Subscribers;

    bool isCH;
    bool isSink;
    bool isCt;
    bool endFormClus;

    vector<RoutingPacket> bufferAggregate;
    vector<int> powers;
    queue <cPacket *> tempTXBuffer;
    vector <int> clusterMembers;
```

```
    list <CHInfo> CHcandidates;
    ResourceManager *resMgr;
    ResourceManager *hello;

protected:
    void startup();
    void fromApplicationLayer(cPacket *, const char *);
    void fromMacLayer(cPacket *, int, double, double);
    void timerFiredCallback(int);
    void processBufferedPacket();
    void sendAggregate();
    void setPowerLevel(double);
    void setStateSleep();
    void setStateRx();
    void levelTxPower(int);
    void readXMLparams();
};
bool cmpRssi(CHInfo a, CHInfo b);
#endif
```

SourceCode/NewprotocolRouting.h

The configuration file (omnetpp.ini file) of the module is illustrated as follows :

```
[General]
#########################
## Network #####
########################
include ../Parameters/Castalia.ini
include ../Parameters/MAC/CSMA.ini
sim-time-limit = 200s
SN.field_x = 100  #meters
SN.field_y = 100  #meters
SN.numNodes = 201
SN.node[0].xCoor = 1
SN.node[0].yCoor = 1
SN.deployment = "[1..200]->uniform"
#########################
## Traces #####
########################
SN.wirelessChannel.collectTraceInfo = false
SN.node[*].Communication.Radio.collectTraceInfo = false
SN.node[*].Communication.MAC.collectTraceInfo = false
SN.node[*].Communication.Routing.collectTraceInfo = false
SN.node[*].Application.collectTraceInfo = false
```

```
SN.node[*].SensorManager.collectTraceInfo = false
SN.node[*].ResourceManager.collectTraceInfo = false
#########################
## Ressource Manager ####
#########################
SN.node[*].ResourceManager.initialEnergy = 17  #initial enery of nodes = 17J
#########################
## MAC #####
########################

#-----CSMA-CA-----#

#########################
## Routing #####
########################
SN.node[*].Communication.RoutingProtocolName = "NewprotocolRouting"
SN.node[0].Communication.Routing.isSink = true
SN.node[*].Communication.Routing.slotLength = 0.2
SN.node[*].Communication.Routing.roundLength = 20s
SN.node[*].Communication.Routing.percentage = 0.05
SN.node[*].Communication.Routing.powersConfig = xmldoc("powersConfig.xml")
#########################
## Application #####
########################
SN.node[*].ApplicationName = "ThroughputTest"
SN.node[*].Application.packet_rate = 1
SN.node[*].Application.constantDataPayload = 2000
#########################
## Wireless Channel #####
########################
SN.wirelessChannel.onlyStaticNodes = true #no mobility of nodes
SN.wirelessChannel.sigma = 0
SN.wirelessChannel.bidirectionalSigma = 0 #two directions of link are the same
SN.wirelessChannel.pathLossExponent = 1.0
#########################
## Radio #####
########################
SN.node[*].Communication.Radio.RadioParametersFile = "../Parameters/Radio/CC2420.txt"
```

SourceCode/omnetpp.ini

The module's network description file (NED file) used is depicted as follows :

```
//*************************************************/
//* NED file describe the base stucture of the module.
//*************************************************/
package node.communication.routing.newprotocolRouting;

simple NewprotocolRouting like node.communication.routing.iRouting
{
parameters:
    string applicationID = default ("throughputTest");
    bool collectTraceInfo;
    int maxNetFrameSize = default (0);
    int netDataFrameOverhead = default (14);
    int netBufferSize = default (32);
    bool isSink = default (false);
    double percentage;
    double roundLength @unit(s);
    double slotLength;
    //routing layer packet sizes
    int helloPacketSize = default (9); //hello msg for calculate density
    int advPacketSize = default (9);  // Type + Source + Destination = 9 bytes
    int joinPacketSize = default (9);
    int tdmaCrtlPacketSize = default (150);// Type + Source + Destination + tdma = 150 bytes
    int pubdataPacketSize = default (9);
    int subdataPacketSize = default (4);
    int cbdataPacketSize = default (9);
    int chdataPacketSize = default (9);
    //Parameters used to power control.
    xml powersConfig;
gates:
    output toCommunicationModule;
    output toMacModule;
    input fromCommunicationModule;
    input fromMacModule;
    input fromCommModuleResourceMgr;
}
```

SourceCode/NewprotocolRouting.ned