NETWORK PERFORMANCE EVALUATION FOR M2M WSN AND SDN BASED ON IOT APPLICATIONS

A thesis submitted for the degree of Doctor of Philosophy
Department of Electronic and Computer Engineering
College of Engineering, Design and Physical Sciences
Brunel University London

By

Wasan Adnan Twayej

Supervisor: Prof. Hamed Al-Raweshidy

2018

Abstract

This thesis introduces different mechanisms for energy efficiency in Wireless Sensor Networks (WSNs) along with maintaining high levels of Network Performance (N.P) with reduced complexity.

Firstly, a Machine-to-Machine (M2M) WSN is arranged hierarchically in a fixed infrastructure to support a routing protocol for energy-efficient data transmission among terminal nodes and sink nodes via cluster heads (CHs). A Multi-Level Clustering Multiple Sinks (MLCMS) routing protocol with the IPv6 protocol over Low Wireless Personal Area Networks (6LoWPAN) is proposed to prolong network lifetime. The simulation results show 93% and 147% enhancement in energy efficiency and system lifespan compared to M-LEACH and LEACH, respectively. By utilising 6LoWPAN in the proposed system, the number of packets delivered increases by 7%, with higher accessibility to the M2M nodes and a substantial extension of the network is enabled.

Secondly, an adaptive sleep mode with MLCMS for an efficient lifetime of M2M WSN is introduced. The time period of the active and asleep modes for the CHs has been considered according to a mathematical function. The evaluations of the proposed scheme show that the lifetime of the system is doubled and the end-to-end delay is reduced by half.

Thirdly, enhanced N.P is achieved through linear integer-based optimisation. A Self-Organising Cluster Head to Sink Algorithm (SOCHSA) is proposed, hosting Discrete Particle Swarm Optimisation (DPSO) and Genetic Algorithm (GA) as Evolutionary Algorithms (EAs) to solve the N.P optimisation problem. N.P is measured based on load fairness and average ratio residual network energy. DPSO and GA are compared with the Exhaustive Search (ES) algorithm to analyse their performances for each benchmark problem. Computational results prove that DPSO outperforms GA regarding complexity and convergence, thus it is best suited for a proactive IoT network. Both algorithms achieved optimum N.P evaluation values of 0.306287 and 0.307731 in the benchmark problems P1 and P2, respectively, for two and three sinks. The proposed mechanism satisfies different N.P requirements of M2M traffic by instant identification and dynamic rerouting to achieve optimum performance.

Finally, a Power Model (PM) is essential to investigate the power efficiency of a system. Hence, a Power Consumption (PC) profile for SDN-WISE, based on IoT is developed. The outcomes of the study offer flexibility in managing the structure of an M2M system in IoT. They enable controlling the provided Network Quality of Service (NQoS), precisely by achieving physical layer throughput. In addition, it provides a schematic framework for the Application Quality of Service (AQoS), specifically, the IoT data stream payload size (from the PC point of view). It is composed of two essential parts, i.e., control signalling and data traffic PCs and the results show a 98% PC of the data plane in the total system power, whereas the control plane PC is only 2%, with a minimum Transmission Time Interval (TTI) (5 sec) and a maximum payload size of 92 Bytes.

Table of Contents

Dedication

This thesis and all my academic achievements are dedicated to

My Father,

who inspired me to be ambitious and fully supported me throughout my student life. Despite him not being amongst us today (may his soul rest in peace), his words saying “You are strong and hardworking enough to achieve our dream” have always been a source of encouragement and motivation for me during the tough times of PhD life. I wish he was here to see that his daughter has fulfilled his dream today.

My Mum

For her prayers, continuous encouragements, unconditional support and love that helped me reach my goals.

My Husband Sahel

For your absolute support, trust, unmeasurable patience, unconditional love and understanding at times when I was stressed and at my lowest.

My Daughter and My Son

Haya and Sultan

Who tolerated a PhD student mum and remained patient kids. Love you for the constant doses of encouragement, especially, “You will be a doctor one day, mum! Don’t worry”.

Acknowledgements

First, I am infinitely thankful to God for blessing me with the strength and perseverance to pursue my dream and achieve this work.

Second, I would like to express my utmost gratitude to my supervisor Professor Hamed Al-Raweshidy. I am very grateful to Professor Al-Raweshidy for providing me an extreme research environment and his continuous guidance, wisdom, and support. I am forever indebted.

I am also thankful to my second supervisor Dr. Maysam Abbod.

I would also like to thank the Iraqi Ministry of higher education and Scientific Research and AL- Mustansiriya University for providing me a scholarship to pursue my PhD study as well as, the Iraqi Attaché in the UK for their continuous support during this time.

I am deeply grateful to my colleagues Basim, Muhammad and Etidal for their constant support.

Last, but by no means least, I would like to express my appreciation to all of my extended family; and to whom I dedicate this thesis.

Declaration

I certify that the effort in this thesis has not previously been submitted for a degree nor has it been submitted as part of requirements for a degree. I also certify that the work in this thesis has been written by me. Any help that I have received in my research work and the preparation of the thesis itself has been duly acknowledged and referenced.

Signature of Student

Wasan Twayej

List of Figures

List of Tables

List of Abbreviations

ADC	Analogue to Digital Converters
AES	Advanced Encryption Standard
BS	Base Station
CapEx	Capital Expenditure
CH	Cluster Head
CTR	CONTROL Plane Signalling
DAT	DATA Plane Traffic
Deg(n)	Number of neighbours
D(n,s)	Distance from node to sink
DLL	Data Link Layer
DPSO	Discrete Particle Swarm Optimisation
EAs	Evolutionary Algorithms
E(n)	Residual Energy of node n
EAERP	Energy-Aware Evolutionary Routing Protocol
EBUC	Energy balanced unequal clustering protocol
$\mathbf{E_{elect}}$	Energy consumption on the circuit
$\mathbf{E_{fs}}$	Free space model of transmitter amplifier
$\mathbf{E_{in}}$	Initial energy
$\mathbf{E_{mp}}$	Multi-path model of transmitter amplifier
ES	Exhaustive Search
$\mathbf{E_{TC}(n)}$	Total consumed energy
F	Number of functions
FPS	Flexible Power Scheduling
ForCES	Forwarding and Control Element Separation
FSM	Finite State Machines
FWD	Forwarding Layer
GA	Genetic Algorithm

GPS	Global Positioning System
H2H	Human-to-Human
$\mathbf{H_{CH-Node}}$	Transmission Range from CH to Sink
HEED	Hybrid Energy-Efficient Distributed
HYMN	Hybrid Multi-hop Routing
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
IGRP	Interior Gateway Routing Protocol
INPP	In-Network Packet Processing
IoTs	Internet of Things
IP	Internet Protocol
IPv6	Internet Protocol Version 6
KPI	Key Performance Indicator
LAN	Local Area Network
LEACH	Low Energy Adaptive Clustering Hierarchy
M2M	Machine-to-Machine
MLCMS	Multi-Level Clustering Multiple Sink
M-LEACH	Multi-Hop LEACH
MTU	Maximum Transmission Unit
N.P	Network Performance
NOS	Networking Operating System
NS3	Network Simulator
NQoS	Network Quality of Service
PC	Power Consumption
$\mathbf{PC_{Adapt}}$	Power Consumption in Adaptation Processing
$\mathbf{PC_{SN}}$	Power Consumption in SN
$\mathbf{PC_{SNK}}$	Power Consumption in SNK
$\mathbf{PC_{NOS}}$	Power Consumption in NOS
$\mathbf{PC_{TM}}$	Power Consumption in Topology Management
$\mathbf{PC_{(WISE-Visor)}}$	Power Consumption in WISE-VISOR Controller

PHY	Physical Layer
PM	Power Model
PSO	Particle Swarm Optimisation
PSO-C	Particle Swarm Optimisation Centralised
QoS	Quality of Service
SN	Sensor Node
SNK	Sink Node
SOCHSA	Self-Organising Cluster Head to Sink Algorithm
SDN	Software Defined Network
SDN-WISE	Software Defined Network for WIreless SEnsor networks
SDWN	Software-Defined Wireless Networking
SIG	Special Interest Group
SON	Self-Organising-Network
SPIN	Sensor Protocol for Information Via Negotiation
TC	Topology Control
TCP	Transmission Control Protocol
TD	Topology Discovery
T.R	Transmission Range
TM	Topology Management
TTI	Transmission Time Interval
V	Velocity
W(n)	Ratio weight of the node
WSN	Wireless Sensor Network
WPAN	Wireless Personal Area Network
6LoWPAN	IPv6 protocol over Low Wireless Personal Area Networks

Chapter 1 Introduction

1.1 Introduction

The Internet of Things (IoT) refers to a network of billions of objects, information and people that can send and receive data. Machine-to-Machine (M2M) communication is considered to be the core of IoT, whereby it is seen as the key to making it a reality, i.e. enhancing the communication of real things in the world, transmitting information and executing smart tasks [1] .

Many solutions have been provided in the M2M domain, and it is anticipated that billions of devices will be connected using M2M. Accordingly, M2M networks enabling networked nodes (sensors or actuators) to exchange information have been intensively studied lately, particularly for sensing and surveillance purposes where hundreds or thousands of nodes are densely located over a small or medium area [2].

Moreover, sensors and communication devices are the essential endpoints for any M2M application. However, the endpoints cannot connect directly to the network operator or interconnect, unless supported by Wireless Personal Area Network (WPAN) technology standards such as IPv6 Low Power WPAN (6LoWPAN), ZigBee, and Bluetooth. Recent technological developments, including M2M, smart grids and smart environment applications, are having a marked impact on the development of Wireless Sensor Networks (WSNs) [3]. Furthermore, most appliances are equipped with the capabilities for sensing people’s daily needs. The efficient use of the limited energy resources of WSN nodes is crucial to these technological advances, regarding which topology control methods are being employed to extend battery lifetime [4]. The vision of the future is one, where millions of small sensors, actuators, and other devices can form self-organising wireless networks. This is dependent on the IoT concept, whereby billions of machines will be able to interact and control one another with no human involvement [5] .

The main task of topology control in wireless M2M networks is to diminish energy consumption and hence prolong network lifetime. Because energy efficiency is crucial when designing M2M WSN protocols, the sensor nodes need to function in a self-governing manner with small batteries that last for several months or even years. Since the replacement of the batteries for a large number of devices is virtually impossible in distant or unfavourable environments, energy is an essential resource in WSNs [6]. Hence, keeping the PC to a minimum is a primary focus for researchers, and this topic has been investigated thoroughly. Energy-efficient transmission protocols for M2M WSN are categorised into routing and clustering types. This type of protocol substantially reduces energy consumption by aggregating multiple sensed data that are transmitted to the sink node [7]. However, if this is not associated with the position of the sink, the network's topology will contain clusters with unbalanced residual energy and a limited network lifespan [8].

In fact, the sink placement becomes an important criterion for network designers with regard to increasing the network lifetime and system performance. In a WSN, multiple sinks at correct locations can sharply decrease the energy use [9]. In a large-scale M2M scenario with a single sink WSN, a heavy traffic load may be experienced for packet transmission at the sink, which may lead to heavy packet drops at the sink as well as significant network congestion. Nodes not only gather data within their sensing range, for they also send them to those nodes remote from the sink, which leads to different PCs among them along with unequal connectivity across the network [10].

Apart from energy consumption efficiency, an important issue is the demand for more IPs as M2M is separated from the IoT and needs to communicate a vast number of things. This issue can be addressed by using 6LoWPAN, and its application is rapidly increasing. In particular, it has been playing an important role in enhancing the throughput, which has been somewhat limited, for Institute of Electrical and Electronics Engineers (IEEE) 802.15.4. The main reason for designing 6LoWPAN is to pave the way for M2M for supporting a broad range of applications [11].

However, supporting a wide variety of applications can lead to network overload situations. Overload in networks has an adverse impact on performance, which can be addressed through effective load balancing algorithms. The general aim is to share the load evenly and fairly amongst all of the resources available, with the method pursued having a direct impact on Network Performance (N.P) [12] [13]. In sum, the key objective of any load balancing algorithm is to improve the computation capability of the system by obtaining a fair and balanced use of the available capacity in the system

[14]. For instance, in an overload situation at a sink, the task causing overload may be reassigned to an under-loaded sink. When compared to traffic in a conventional Human-to-Human (H2H) arrangement, M2M has different issues and characteristics. To begin with, burst traffic generated by an enormous number of devices can lead to a server experiencing heavy processing overheads. Also, Quality of Service (QoS) can differ, according to the types of traffic from different applications. To deal with heavy network loads while keeping high levels of N.P and enhanced energy efficiency, M2M networks usually employ a load balancing server [15].

However, the future IoT will involve an increasing number of devices, which will lead to difficulties with achieving an efficient routing protocol for interval sensing, which is necessary for providing highly reliable information. Moreover, the WSN lacks the flexibility of monitoring a massive number of devices in the network. Hence, being based on the Software Defined Network (SDN) concept will support high-quality management of the network, applied by the reconfiguration rule at the controller. As a result, in order to enhance the Network Quality of Service (NQoS), the network elements have the ability to forward rules provided by the central controller by integrating the SDN model with WSNs [16].

The SDN model with WSNs has been studied in different research investigations, based on the characteristics of the latter and it makes essential differences in SDN applications compared with wired networks. It enables the monitoring and reconfiguring of the routing path and the configuration of the network elements in a stable and more efficient manner, by providing a centralised controller.

A significant number of research studies have been carried out on energy efficiency for WSN, looking at how to prolong the lifespan of the system based on clustering techniques ([17]; [8]; [18]). Moreover, prior work has been undertaken, focusing on load balancing, with most of it being related to the CHs. In addition, a number of important studies have considered the significant rise in the expected number of machines that each handles a huge number of devices, which will lead to increasing demands for traffic management with SDN in different applications. As a result, to cope well with the upcoming and current challenges, an efficient M2M WSN system should be designed, taking into account all of the important facts. This will serve specific M2M applications based on the IoT as efficiently as possible in the long term. In view of this, an efficient clustering M2M routing protocol, with high N.P for clustering M2M has been proposed. It also profiles a power model for SDN-WISE for tackling the vast network of information, considering the PC aspects in particular the control signalling and data traffic for enhancing the NQoS and AQoS [19].

1.2 Motivation

There are many challenges leading to the motivation for this study as shown in Figure 1.1, which are:

- Recent technology development is based on the deployment of self-organising WSNs and their large-scale of deployment, such as the IoT [20], which increases the demand for the use of WSN nodes that have a limited energy capacity for small batteries.

- There is a great need to use WSN in order to provide applications for controlling methods related to the updated requirements. This will have an impact on the energy efficiency and network lifetime. The most critical part in WSN is the small batteries, which provide the power and these require changing or charging after becoming exhausted. The critical issue in these batteries is the difficulty of changing them when environmental conditions are challenging.
- The IoT comprises a huge amount of things that provide services to the connected society [21]. This escalates the demand for large IP address spaces for M2M WSN, as M2M is an essential part of the IoT.

- One of the matters that needs consideration is the sensor nodes being active all the time, even in unnecessary periods, as this will be the main source of energy consumption.

- The critical requisite of considering unbalanced traffic of WSN will impact negatively on the N.P correlated to traffic management overload. As the conjunction is related to sink overload, it negatively influences the application’s reliability at the sink.

- It is difficult to manage a vast number of sensors in the network. Furthermore, there is a lack of flexibility in the reconfiguration policy for network implementation.

- Most research on SDN power models has been focused on wired networks rather than wireless. Moreover, those few on wireless networking have not considered the extra control signalling in power consumption

- The SDN in WSN has extra control signalling to cope the lack of network information, which has an effect on the NQoS.

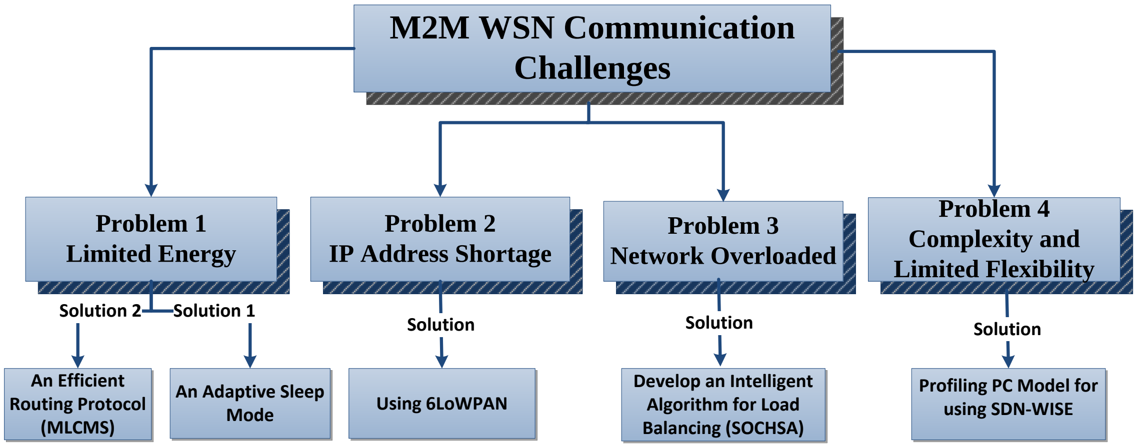


Figure 1.1 M2M WSN problems and solutions

1.3 Aim and Objectives

The overall aim of the study is to develop a mechanism of M2M WSN for prolonging the lifetime of the system and enhancing network performance with high flexibility. This can be achieved by proposing an efficient energy consumption routing protocol, a load balancing strategy for M2M communication, for fair and efficient network resource utilisation and enabling network flexibility management with high NQoS and AQoS.

The research aim is addressed through fulfilling the following objectives:

1. Review the available substantial efforts that aimed at enhancing energy efficiency in different ways, based on the taxonomy formula for cataloguing. A comprehensive

study of WSN routing protocol techniques is presented, focusing on energy efficiency and extending the system lifetime, which involves three sub-objectives:

   a- Reducing the energy consumption by proposing an M2M energy-efficient routing protocol;

   b- Selecting the CHs in a sophisticated way, depending on a mathematical function involving most of the important characteristics that should be considered when electing a CH, such that they can stay alive as long as possible. That is, this stage relates to extending the network lifespan;

   c- Based on 6LoWPAN, overcoming the shortage of IP addresses;

2. Tackle unnecessary energy consumption by proposing a model for an adaptive sleep mode approach. This maximises the lifetime of the system and minimises the delay. Furthermore, the trade-off between energy conservation and delay is taken into account.

3. Implementing a load balancing algorithm and a mathematical model for enhancing the N.P of the whole network, with the method pursued having a direct impact on N.P. In sum, the key objective of any load balancing algorithm is to improve the computation capability of the clustered system by obtaining a fair and balanced system with high residual energy. The N.P is implemented by using two Evolutionary Algorithms (EA)s:

   a- An algorithm is proposed based on Discrete Particle Swarm Optimisation (DPSO) to maximise the N.P;

   b- The same algorithm is proposed based on a Genetic Algorithm (GA);

4. Introducing SDN-WISE as a WSN based on SDN, with the former clearly escalating the overall network's PC owing to the network's elements and signaling. Hence, evaluating the PC of such architecture is essential to assess the power cost caused by additions to control signaling. Failure to consider these factors when modelling a PC model as an essential part of IoT architecture. The objectives for our investigations as follows:

a. Profiling the SDN-WISE's Power Model (PM) to evaluate the PC within WSN parts combined with SDN components as a control plane and data plane. This work allows the operators and end users to consider the PC effects on each part;

b. Determining an effective interaction between control signalling with respect to Transmission Time Interval (TTI) and data traffic related to packet size as well as the data rate to meet applications' AQoS requirements. Reducing TTI will increase the PC that will increase the PC, on the NQoS;

1.4 Contributions

- A Multi-Level Clustering Multiple Sink (MLCMS) as a M2M WSN with IPv6 protocol over Low Wireless Personal Area Networks (6LoWPAN) is promoted on a fixed infrastructure network. This has been achieved by using a advanced way of dividing the sensing regions with a fixed infrastructure model, and a mathematical function for electing cluster heads (CHs) for each level. This is considered to be the main way of extending the network lifetime. Furthermore, this contribution is published in IEEE Sensor Journal and has been published in IEEE conferences [22].
- An adaptive sleep mode scheme is presented to enhance the power consumption efficiency with respect to a reduction in delay. This approach has the ability to be change dynamically according to the structure. Moreover, this part has been published as a chapter in book series by Springer and also published in IEEE conference.
- A Self-Organising Cluster Head to Sink Algorithm (SOCHSA) is proposed, which considers two important parameters to monitor N.P, based on which, the optimum CH-Sink setting can be identified. These parameters are the load fairness index and average residual energy. The proposed SOCHSA algorithm hosts two EAs (i.e. GA and DPSO) to solve the load balancing optimisation problem in order to enable every CH to transmit flexibly to any sink based on the CH-Sink configuration. This contribution has been published in the IEEE Sensor Journal [22].
- Profiling a Power Model (PM) for SDN in WSN as SDN-WISE architecture and evaluating the PC, by considering the effectiveness of extra control signalling in an energy efficiency network. This work allows for consideration of the PC effects on NQoS with respect to the throughput and TTI as well as the impacts on the AQoS in

relation to packet size. This part of work has been submitted to the IEEE Internet of Things Journal, in the form of an article.

1.5 Thesis Outline

The study in this thesis is organised into six chapters. Each chapter will start with a brief introduction providing an overview and highlighting its main contributions, whilst at the end of each chapter there will be a brief conclusion. An outline diagram is presented in Figure 1.2, which illustrates the structure of the thesis and the relationships between its objectives.

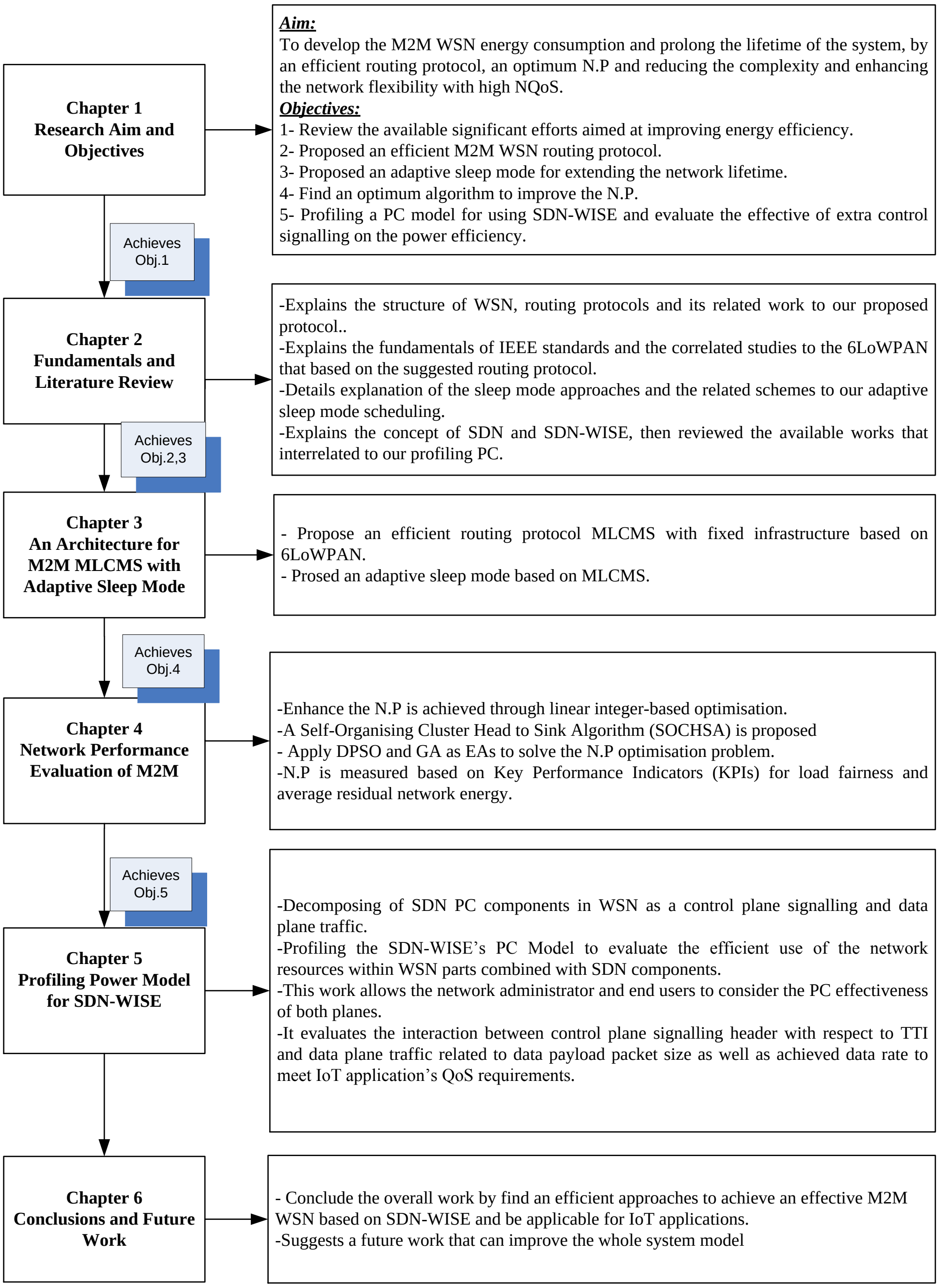
Chapter 2: This chapter presents detailed fundamentals about the WSN and its routing protocols, IEEE standards, sleep mode and the SDN in WSN over the years. Moreover, it explains the different research studies related to the hierarchal routing protocols based on improving energy efficiency. Moreover, it precedes the protocols of IEEE standards related to the 6LoWPAN scheme. Then, proposals are made for reducing energy consumption by using different sleep mode techniques. In addition, the chapter discusses the basic idea of load balancing, and the prior works based on a metaheuristic perspective of using EAs (PSO) and (GA) are reviewed. Furthermore, the latest research on implementing SDN in WSN is discussed in two parts: one correlated to programmable studies and one related to the PC part. Then, the chapter concludes with a brief summary of the research gaps and how they are filled by this work.

Chapter 3: This chapter presents the phases used to propose an efficient M2M routing protocol, MLCMS. Also, it details the steps that led to suggesting an adaptive sleep mode approach. It explains how to examine and compare the proposed model with other routing protocols, then, how the anticipated routing is enhanced by adding an adaptive sleep mode technique. The chapter concludes with a discussion of the results.

Chapter 4: This chapter starts by explaining the optimum steps for enhancing the N.P, based on the MLCMS routing protocol. A SOCHSA is proposed, hosting DPSO and GA as EAs to solve the N.P optimisation problem. Then, the results are discussed and the optimum N.P system model is illustrated.

Chapter 5: This chapter introduces the SDN-WISE Power Model (PM) paradigm by explaining the phases of the pattern. The chapter also explains how the additional control signalling will affect the total PC and the efficient use of the network resources within the WSN parts combined with SDN components is evaluated, from the control plane signalling and data plane traffic point of views. Then, the results of all of the effective control signalling on the power efficiency are discussed in detail.

Chapter 6: The study outcomes and conclusions are summarised in Chapter 6, along with suggestions for future work as well as the research impact.



Aim:
To develop the M2M WSN energy consumption and prolong the lifetime of the system, by an efficient routing protocol, an optimum N.P and reducing the complexity and enhancing the network flexibility with high NQoS.

Objectives:
1- Review the available significant efforts aimed at improving energy efficiency.
2- Proposed an efficient M2M WSN routing protocol.
3- Proposed an adaptive sleep mode for extending the network lifetime.
4- Find an optimum algorithm to improve the N.P.
5- Profiling a PC model for using SDN-WISE and evaluate the effective of extra control signalling on the power efficiency.

-Explains the structure of WSN, routing protocols and its related work to our proposed protocol..
-Explains the fundamentals of IEEE standards and the correlated studies to the 6LoWPAN that based on the suggested routing protocol.
-Details explanation of the sleep mode approaches and the related schemes to our adaptive sleep mode scheduling.
-Explains the concept of SDN and SDN-WISE, then reviewed the available works that interrelated to our profiling PC.

- Propose an efficient routing protocol MLCMS with fixed infrastructure based on 6LoWPAN.
- Prosed an adaptive sleep mode based on MLCMS.

-Enhance the N.P is achieved through linear integer-based optimisation.
-A Self-Organising Cluster Head to Sink Algorithm (SOCHSA) is proposed
- Apply DPSO and GA as EAs to solve the N.P optimisation problem.
-N.P is measured based on Key Performance Indicators (KPIs) for load fairness and average residual network energy.

-Decomposing of SDN PC components in WSN as a control plane signalling and data plane traffic.
-Profiling the SDN-WISE's PC Model to evaluate the efficient use of the network resources within WSN parts combined with SDN components.
-This work allows the network administrator and end users to consider the PC effectiveness of both planes.
-It evaluates the interaction between control plane signalling header with respect to TTI and data plane traffic related to data payload packet size as well as achieved data rate to meet IoT application's QoS requirements.

- Conclude the overall work by find an efficient approaches to achieve an effective M2M WSN based on SDN-WISE and be applicable for IoT applications.
-Suggests a future work that can improve the whole system model

Figure 1.2 Thesis outline

Chapter 2 Fundamentals and Literature Review

2.1 Introduction

The main aim of topology control in wireless M2M networks is to reduce energy consumption and, therefore, extend network lifetime. This efficiency is essential in the design of M2M WSN protocols, since the sensor nodes need to maintain autonomy over several months or even years while maintained by small size batteries with limited lifespan [4]. Therefore, many research groups have dedicated themselves to minimising energy consumption. Energy efficient transmission protocols for M2M WSN can be logically divided into two types – routing and clustering. The routing is finding an effective path between the source and destination. However, clustering in M2M WSN refers to grouping sensor nodes, according to the needs of the network. Each cluster has a main node, called the cluster head (CH), and a number of ordinary nodes [23]. The aggregation of data from multiple sensors into a single data flow that is transmitted by the CH to the sink node significantly reduces energy consumption [24]. The clusters can be configured to increase the energy efficiency of the network by varying the number of nodes included in the cluster. If clusters are not associated with the position of the sink, the network may include clusters with unbalanced residual energy and the life of the network will be short [25].

In addition to the reduction of power consumption, more IPs should be developed, since M2M is a part from the IoT and needs to communicate with a large number of sensors within the network. This issue can be addressed by using 6LoWPAN, which pertains to the use of the IPv6 protocol over Low Wireless Personal Area Networks. 6LoWPAN protocol increases network throughput, previously limited by IEEE 802.15.4, allowing IoT support of applications with multiple sensor nodes [11].

Overload in networks has a negative impact on performance, which can be addressed through effective load-balancing algorithms. These have been widely researched over the last 20 years after the innovations of parallel and distributed computing. The main target of balancing algorithms is to distribute the load evenly and fairly amongst all the available resources, which directly affects network performance [12] [26].

Recently, the integration of software defined networks (SDN) with WSN such as SDN-WISE has led to the advantage of centralised control for networked devices in M2M WSNs. SDN technology allows the implementation of reconfigurable, scalable, and energy-efficient network design [27].

This chapter discusses the fundamentals and previous state of the art related to the subject of the thesis. It is structured as follows: **Section 2.2** discusses basic information related to WSN, routing protocols and discusses different studies of hierarchical routing protocols aimed at prolonging the lifespan of the system. **Section 2.3** presents what is 6LoWPAN and discusses the different kinds of standards. **Section 2.4** gives the fundamentals for the sleep mode types, discusses the classification of the sleep mode and considers the available schemes to improve the energy efficiency. **Section 2.5** discusses the concept of M2M network performance. The fundamentals of SDN and related work to SDN-WISE are presented in **Section 2.6.** The summary of the chapter is presented in **Section 2.7**.

2.2 Fundamentals for WSN and routing protocols

Wireless Sensor Networks (WSNs) are made up of numerous sensor nodes that collect the information about their environments. The main part of each node is a sensing element designed to monitor various physical parameters including temperature, pressure, and so on. In addition to sensors, WSNs may also have gateways and clients. Arbitrarily positioned nodes located near the surveyed area (sensor field) may communicate with each other, forming a self-organised network. During the transmission, the structured data flows may be routed by multiple nodes to gain access to a sink node after multichip routing [28]. Afterwards, the data can be transmitted to the management node via the internet or other communication systems. An operator-controlled administration node is used to construct and coordinate the WSN, apply various monitoring scenarios, and archive the collected data. In line with technological development, WSN’s capital and operational expenditures have been significantly reduced, and so these networks are now widely used and have home, industrial, and military applications [29].

2.2.1 Architecture of Sensor Node

Typically, a sensor node has four major components: power supply, processing unit, sensing unit, and data transceiver.

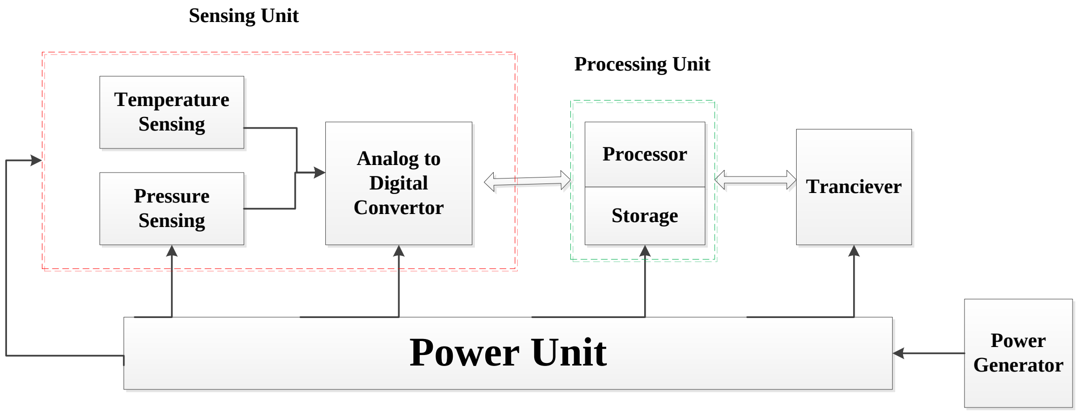


Figure 2.1 Sensor node's architecture

In general, a sensing unit is made up of sensors and Analogue to Digital Converters (ADC), which convert electrical analogue signal to digital data streams. Digitisation of the monitored signal allows for data to be processed within the node, which is connected to the network by the transceiver. The processing unit has a processor to control and manage the sensor node, and a memory device for data storage [30].

2.2.2 Overview of Routing Protocols in WSN

A routing protocol can be called adaptive if certain system variables can be changed to adapt to the present network state and existing power levels. In general, routing in WSNs can be grouped into two main types: Routing Processing and Network Structure, as shown in the block diagram of Figure 2.2 [31].

A. *Routing Protocols Based on Routing Processing*

Routing protocols can also be categorised in terms of how the source routes the signal to its destination, the categories are reactive, proactive, and hybrid. When all network routes are calculated before their use, the protocol is called proactive, in contrast to reactive protocols, where routes are calculated when they are actually needed. Hybrid protocols combine proactive and reactive strategies [32].

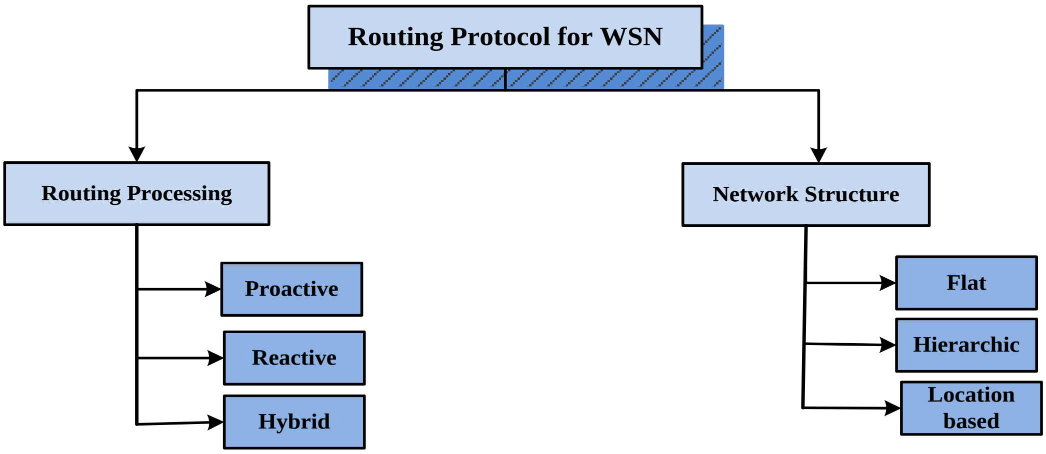


Figure 2.2 Routing protocols in WSNs

B. *Routing Protocols Based on Network Structure*

In WSNs, the core network structure can be very important in the implementation of the routing protocols. This section is dedicated to describing the routing protocols that rely on network structure [33].

- **Flat Routing**:

In this type of WSN, all nodes perform the same task in order to collect data and communicate with the sink. In other words, any node is interchangeable with any other node in the network and the data collected across the sensor field is of the same type or duplicated, since the functionality of all nodes within the network is the same. Due to the large node count in WSN, nodes typically do not have individual global identifiers. During routing, Base Station (BS) or sink transmits queries to the desired area of WSN and waits for a reply from sensors within that part of the network. Sensor Protocol for Information Via Negotiation (SPIN) is a matchable example of this type [34][35].

- **Hierarchical Routing**:

In hierarchical routing protocol, nodes can be logically divided by their access to power into high- and low-energy types. High-energy nodes perform data collection, aggregation, processing and transmission to the BS, while low-energy modes perform data acquisition and transmission of measurement results to the high-energy nodes. Since a high-energy node and a number of low-energy nodes form a cluster, hierarchical routing protocols are called cluster-based. These protocols have good scalability and allow for the reduction of energy consumption. The formation of clusters with cluster heads, and subsequent data aggregation within the cluster head extends the lifespan of the WSN and reduces overall power consumption. Low Energy Adaptive Clustering Hierarchy (LEACH), Multi-Hop LEACH (M-LEACH), Hybrid Energy-Efficient Distributed Clustering approach (HEED) [36] [37] and improved LEACH are typical hierarchical routing protocol all these protocols related to this type.

- **Location Based Routing**:

In location-based routing network, nodes are addressed by means of their positions. One way to estimate the distance between nodes is to compare their signal strengths. Neighbouring nodes exchange this information between each other to compute relative coordinates. Alternatively, the Global Positioning System (GPS) can be used to obtain nodes' coordinates by receiving signals from a number of GPS satellites, providing that the nodes have small GPS receivers with low power consumption. In order to reduce energy use, the sleep mode is activated in some location-based networks during periods of inactivity. Some location-based schemes demand that nodes should go to sleep if there is no activity to save energy. Clearly, higher energy saving can be achieved by increasing the number of sleeping nodes. Scheduling the duration of sleep periods for each node in a localised manner was described in [35] .

2.2.3 Related Work to M2M MLCMS as a Hierarchal Routing Protocol for WSN

In this section, state-of-the-art routing protocols for WSNs has been reviewed. Several existing routing protocols attempt to address the issue of providing reliable connectivity for clusters of IoT devices. This section reviews cluster based routing protocols.

In order to prolong the network lifespan, the LEACH protocol has been developed. LEACH was one of earliest protocols in which self-organisation of sensor nodes into clusters was realised. Every cluster has one node that plays the role of CH, and a number of member nodes of the cluster. CH performs data collection from member nodes, aggregation, and transmission of archived data to the BS as shown in Figure 2.3. It is important to note that member nodes do not connect to BS directly and use CHs as an intermediate router. Due to extra functionality, CHs consume energy faster than member nodes and if the CH duty is assigned permanently, CH batteries' discharge will limit the lifespan of the entire network. Therefore, a random rotation of the CH assignment was incorporated in the LEACH protocol to achieve equal battery discharge in all nodes and thus extend lifetime of the network. The LEACH protocol is based on rounds, which can be divided into setup and steady state phases. In setup phase the CHs are created, while in steady state the data transmission [7] [36].

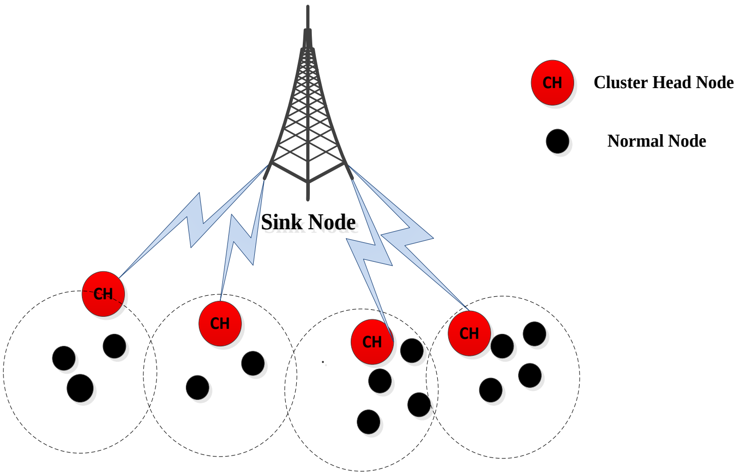


Figure 2.3 Clustering topology of LEACH

The drawbacks of LEACH originate from certain assumptions made by LEACH designers during the development of the protocol. These disadvantages are described in [38] as follows:

- LEACH supposes that every node can communicate with each other and the BS. These conditions can only be achieved in a small-sized network.

- The assignment of CH duties is performed randomly, without taking into account the conditions of nodes (e.g. remaining battery life) and their ability to communicate with other nodes and BS. Therefore, a node with an almost fully discharged battery has the same probability to become CH as every other node within the network. This will result in a reduction of operating nodes and therefore shorten the lifespan of the entire network.

Since the CH selection is random, their distribution across the network may not be uniform, and not guarantee a good elected CH. Therefore, there is a probability that all or many CHs will be selected in small network areas, and that some of the nodes will not be able to be reached or assigned to clusters if the number of nodes in the cluster is limited. An Improved LEACH protocol for application-specific WSN (I-LEACH).In this version of LEACH, the reduction of power consumption is achieved by considering the remaining energy in selecting the CHs and the position of CH as each node should has a CH close to it [39].

In LEACH protocol the energy consumption increases with the distance between CHs and BS. The Multi-hope version of LEACH (or M-LEACH) was proposed to reduce distance dependency of power consumption. In M-LEACH, distant CHs send their data to BS by forwarding information through the chain of CHs located closer to BS as shown in Figure. 2.4. For remotely located clusters, distances between their CHs and adjacent CHs are much shorter compared to the distance to the BS. This division of transmission distance into several hops can significantly reduce the transmission distance and increase the lifespan of the nodes at the edges of network.

In all other aspects, M-LEACH is similar to LEACH. In other words, the LEACH protocol is single hop, while M-LEACH is multi-hop. Moreover, it can be seen as a version of M-LEACH where the number of hops is limited to one. This is not based on any energy calculations within the network, and produces a situation where far-flung clusters may fail much faster than close ones, as nodes within the cluster must all communicate individually with the CH node. M-LEACH makes the obvious correction to this protocol, allowing for nodes to use multi-hop communication within the cluster and between clusters when reaching the base station [40].

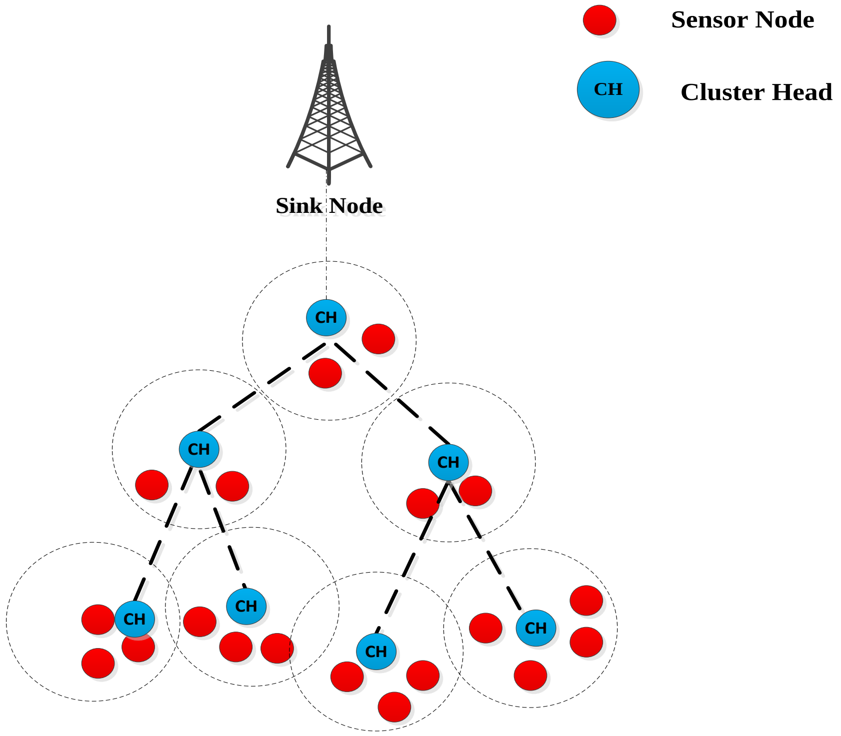


Figure 2.4 Clustering topology of M-LEACH

In contrast, in Hybrid Energy-Efficient Distributed Clustering (HEED) [37] is a distributed clustering approach, which CHs are reselected periodically depending on the scheme parameters. It is unlike LEACH, it doesn't select random cell-head. In HEED protocol, clusters are formed by considering information about remaining energy and the density of neighbouring nodes that may be included in a cluster. Unlike in the LEACH protocol, the CH assignment is decided after the assessment of residual battery life and intra-cluster traffic. The issue of this routing protocol is the nodes in final stage if they are not having a CH, then they must announce themselves as a CH. That will increase number of CHs, which decrease the lifetime of the system.

Some of the energy problems noted above, including the energy hole, may be resolved using the Hybrid Multi-hop Routing (HYMN) algorithm in [24], which is an attempt to avoid the energy hole issues discussed above. The HYMN approach represents a hybrid between flat multi-hop routing to allow for efficient transmission and hierarchical multi-hop routing algorithms to allow for a greater level of data aggregation. The model accounts for the importance of the sink node isolation issue, placing greater emphasis on connectivity time as a metric, in contrast to LEACH, which is based on overall energy usage. It has been noted that using HYMN, high node density enables sustainable connectivity over longer periods without the lag in performance associated with hierarchical multi-hop routing protocols. This study also notes the difference in the number of nodes acting as cluster heads and the increase in energy consumption when longer transmission ranges are introduced.

More importantly, HYMN further improves power savings by implementing data compression. As such, HYMN offers a great deal of benefits over the LEACH family of protocols, and represents a possible area of research. However, it is missed the accuracy in probabilistic sensing model, this probabilistic sensing model is the minimising function of sensing distance for maximising the quality of the WSNs.

All these approaches suffer somewhat from the issue of sink node isolation as owing to many-to-one data traffic that will lead to shorten the lifespan of the WSN. This is clearly a risk given the tendency of close nodes to fail and isolate nodes further from the base station. In addition, these algorithms tolerate from lacking in energy efficiency earthier in the way of forming CHs or in the parameters that are considered in CHs election.

During the development of the MLCMS protocol, it became clear that the algorithms of cluster heads could adapt according to a system’s requirements. It mitigates the isolation of sink node and provides a mathematical analysis in modelling CHs election and power consumption. Moreover, it has been demonstrated that the MLCMS model is dependent on a modification of the energy model used by others, and that its parameters are more accurate in selecting an efficient CH as CH’s election factors play an important role in energy efficiency, which improves the lifespan of the system. Moreover, the distribution of nodes across the network area is also important in improving energy efficiency.

2.3 What is 6LoWPAN

IPv6 over Low-Power Wireless Personal Area Networks (6LoWPAN) has been developed to increase the variety of devices that can be connected to the cloud. Support for IP-driven low power nodes combined into a large mesh network makes this protocol an ideal option for Internet of Things (IoT) applications. 6LoWPAN enables the efficient transmission of IPv6 packets within a small link layer frame; 6LoWPAN is another open standard that has been developed by the Internet Engineering Task Force (IETF) [11].

The main advantage of the IETF-proposed Internet Protocol Version 6 (IPv6) is its support of 128 bits addresses compared to IPv4, which only supported 32 bits addresses. The 6LoWPAN protocol has been developed to add IPv6 capability to WSNs and has become the main standard for IoT realisation. The main idea behind 6LoWPAN is that all nodes within the network, including small embedded devices, should communicate via IP. To enable the transmission of IPv6 packets over IEEE 802.15.4 based networks, IETF implemented an adaptation layer for header compression and packet encapsulation. Owing to the limitation of the payload size of the link layer in 6LoWPAN networks, the adaptation layer in the 6LoWPAN standard covers the compression of the packet header, fragmentation and reassembly of the datagram. For instance, the IEEE 802.15.4 frame size can go beyond the Maximum Transmission Unit (MTU) size of 127 bytes for big application data, whereas that for 6LoWPAN is 1,280 bytes, in which case fragmentation is needed [41]. In other words, 6LoWPAN can guarantee interoperability between WSNs and the Internet, due its support for the IPv6 protocol [11].

2.3.1 6LoWPAN Frame Format

MTU 127 bytes are defined by 802.15.4, while 1280 bytes are necessary for the link layer in IPv6. As shown in Figure 2.5, 25 bytes are the maximum overhead for the 802.15.4 frame; consequential is 102 bytes can Media Access Control (MAC) layer frame size can include.

There are just 81 bytes as a space in IP packets after striking 21 bytes link layer for security header. Moreover, for IPv6 header occupied 40 bytes from the IP packets and for User Datagram Protocol (UDP) header is 8 bytes. As a result, for upper layer data only 33 bytes are available. Furthermore, the fragmentation and header compression for the IP packet is essential for 6LoWPAN for addressing header overhead issue and save space for upper-layer data. IETF 6LoWPAN working group promoted an adaptation layer between 802.15.4 MAC layer and IPv6 layer for easily transmit data packet between IPv6 and 802.15.4 networks [42].

In addition, the compression and decompression packets, fragmentation and reassembly as well as mesh multi-hop forwarding are justified by the adaptation layer. For achieving fragmentation function, it has defined a fragmentation. Furthermore, for providing multi-hop forwarding a mesh header has been with source and destination address. The 6LoWPAN header is defined by the first byte as shown in Table 2.1. In addition, the IPv6 can be compressed or not depending on the dispatch value [43].

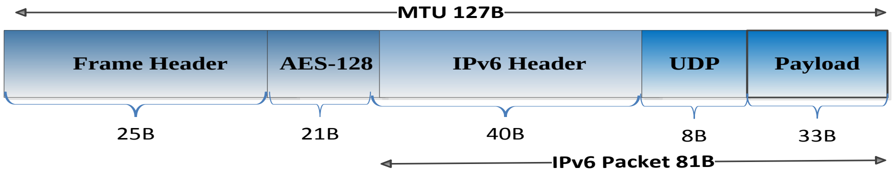


Figure 2.5 IEEE 802.15.4 frame

As a result, using an adaptation layer as intermediate layer between MAC layer (802.15.4) and IPv6 layer for managing IPv6 packet by letting link-layer fragmenting and forwarding [44]. By suppressing the unnecessary information that is inferred from other layers in the stack communication that will compress the IPv6 header as well as the next header [43]. In particular, a stack of headers prefixed all 6LoWPAN encapsulated datagrams, which must be transferred across the IEEE 802.15.4 MAC, each one identified by a type field. Especially, the types of header are logically grouped in four classes, according to the function they process in the strategy of 6LoWPAN adaptation, as shown in Table 2.1 and summarized below:

Table 2.1 6LoWPAN Header Types

First 2 bits		**Following bit combinations**	
No 6LoWPAN	00	xxxxxx	Any combination
Dispatch	01	000000	Additional Dispatch byte follows
		000001	Uncompressed IPv6 Addresses
		000010	LOWPAN HC1 compressed IPv6
		010000	LOWPAN BC0 broadcast
		1xxxxx	LOWPAN IPHC compressed IPv6
Mesh Addressing	10	xxxxxx	Any combination
Fragmentation	11	000xxx	First Fragmentation Header
		100xxx	Subsequent Fragmentation Header

- **A No 6LoWPAN Header:** Is used for identifying that the packet received is not related to 6LoWPAN conditions and it should be neglected. This provides the capability to be other routing protocol standard not only 6LoWPAN in same network.
- **A Dispatch Header:** This part is used for IPv6 header compression or for managing link layer multicast /broadcasting.
- **A Mesh Addressing Header:** The forwarding of IEEE 802.15.4 frames at link layer is provided by this header as well as turning single-hop WSNs in multi-hop ones.
- **A Fragmentation Header:** In case of not fitting the datagram within the IEEE 802.15.4 frame then this header will be used. Each header it can be presented or not depending on the requirements.

2.3.2 Related Work to 6LoWPAN In IEEE Standards

Today, there are many wireless communication standards used for WSN design. Standardisation of protocols is essential to achieving interoperability, not only between networks built by different companies, but also on domain, application, and solution levels. To create a novel cross-domain environment, it is very important for IoT and WSN to enable common access to different actors, devices, and sensors from various applications.

The IEEE 802.15.4 standard defines low-power wireless radio techniques better known as Low Power Wireless Personal Area Networks (LoWPAN). Today, IEEE 802.15.4 is a widely popular radio communication standard aimed to provide low cost and power, relaxed throughput requirements, and short-range ubiquitous communications for embedded devices. IEEE 802.15.4 has been used to specify wireless communication standards, like ZigBee or 6LoWPAN, on Physical (PHY) and Data Link layer (DLL) by adding various network and application layer protocols [45].

ZigBee is a standard proposed by the industry as an answer to the problem of designing a WSN that achieves all the objectives set by the requirements of such a network. ZigBee is a low cost and low power consumption two-way wireless communication standard. Over the last ten years, a large number of proprietary application layer protocols and sensor networks based on the IEE802.15.4 standard have been developed. In 2003, ZigBee Alliance developed the widely-used ZigBee specification to support network, security, and application layers on top of the IEEE802.15.4 standard. In order to perform authentication and ensure message privacy and integrity, ZigBee uses a 128-bit key Advanced Encryption Standard (AES) algorithm for security characteristic. Although ZigBee has been successfully implemented in the agriculture and automation industries as well as in numerous health-care projects, there are many IEEE802.15.4 based research sensor networks that do not use the ZigBee specification, especially when customisation of communication protocols is needed to meet specific requirements [46].

This limits interoperability of WSNs on network and application levels, even if they are based on the same IEEE802.15.4 chipset. To solve this problem, a 6LoWPAN standard has recently been proposed by IETF. 6LoWPAN is the standard that links together IP and IEEE802.15.4 to create wireless IoT in a standardised but easy customisable way. It should be noted that while IP technology is well-developed and familiar to the engineering community, it has not been optimised for use in low power and low bit rate personal area networks. Therefore, an adaptation layer and protocol optimisation has been implemented to allow IP communication between low-cost embedded devices. Essentially, 6LoWPAN is produced as a competitor with the ZigBee, but it has the capability of utilising 802.15.4 better than ZigBee and the comparison between them is shown in Table 2.2. As the IP routing over 6LoWPAN links not requiring any extra header information and the code size is less than the code size for ZigBee. All these features play an important role in reducing the packet overhead and increasing the packets rate.

Table 2.2 Comparison between 6LoWPAN and ZigBee

Features	**6LoWPAN**	**ZigBee**
Interoperability of Wireless Protocol	6LoWPAN device interoperate with any other IP network link	ZigBee device interoperate with ZigBee device only
Protocol Stack	Contains Extra Layer (Adaptation Layer)	No Adaptation Layer
Code Size	30KB	90KB
Security	Built-in AES128 Encryption	Built-inAES128 Encryption
Availability and Cost	Free	Free

2.4 Sleep Mode

The goal of sleep mode schemes is to extend wireless network lifespans by reducing power consumption. Energy saving measures should not reduce WSN reliability, and other network parameters such channel utilisation should not be affected. One well-known way to reduce energy consumption is to keep nodes in sleep mode as long as possible. This can be done by switching on sleep mode after operation over a certain time period.

2.4.1 Sleep Mode Approaches

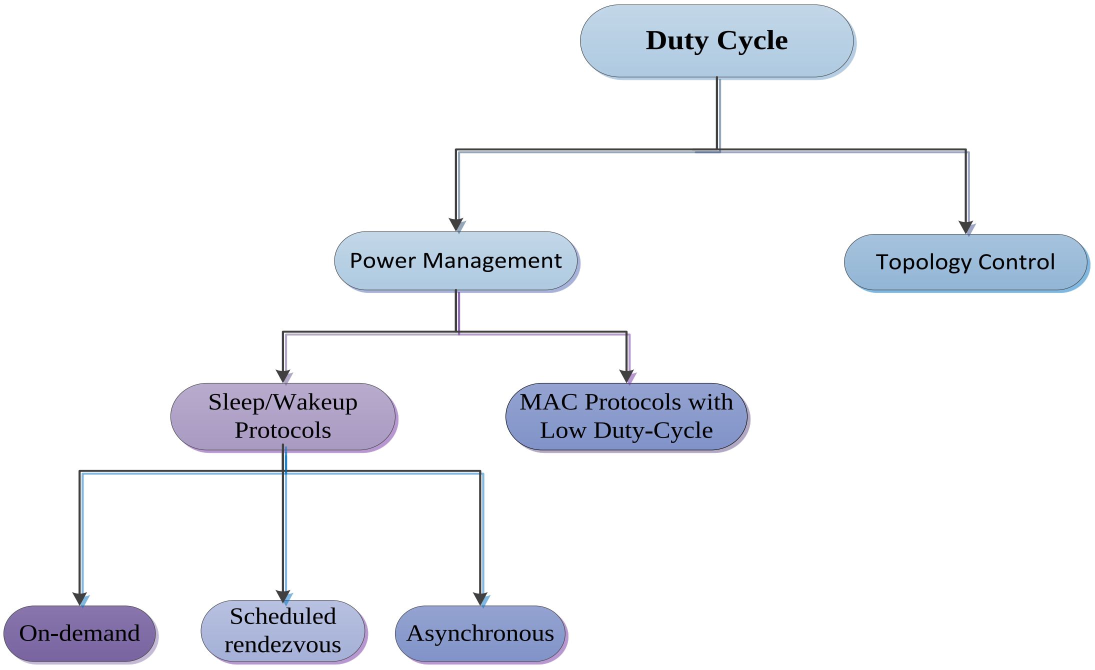


Figure 2.6 Classification of duty cycle schemes

The main concept of sleep mode has been focused on duty-cycling – a technique that enables wireless transponders only when they are needed for data transfer. This duty-cycling can be realised using a combination of Power Management (PM) and Topology Control (TC) techniques as shown in Figure 2.6 [2] [51].

TC is maintained in typical WSN, when only a fraction of nodes are required for data transmission at any given time, and therefore radio transponders in redundant nodes can be temporarily deactivated. The minimum number of active nodes required for reliable data transmission is conserved by TC protocols. The implementation of TC can expand

WSN lifespan by a factor of 2 or 3. More information about TC protocols can be found in [48].

Another way to save energy is to reduce the activity period of nodes selected by the TC protocol for data transmission. The wireless transponder can be switched on only for the short period required for aggregated data transmission, and may remain in sleep mode for the rest of time [49].

The other part of the duty cycle is Power Management (PM); the target of the PM protocol is to organise sleep time of a group of adjacent nodes, and to ensure the existence of communication channels (even in the case of extremely low duty cycle) while minimising power consumption. Since the communication period of sensor nodes is typically short, PM protocols can achieve a reduction in energy consumption, and therefore significantly expand the lifespan of the network. This thesis is dedicated to examining PM protocols, due to their major role in the reduction of power consumption [47].

PM protocols can be realised either at the MAC (device) layer or at higher hierarchical levels, including at application or network layers as an independent sleep/wakeup protocol. The first option offers the designer many options for optimising channel performance and controlling power consumption. The downside of duty-cycling MAC protocols is their device orientation, which reduces flexibility and prohibits their use in mixed-type sensor networks. In contrast, above the MAC layer, sleep/wakeup protocols may be adjusted to match the needs of the specific applications and used alongside any MAC protocols [50].

There are three main types of sleep/wakeup schemes: asynchronous, on demand, and scheduled rendezvous. In asynchronous schemes, the decision to commence communication is taken within a particular node, which gives the ability to each node to wake up independently, provided that transmission will always be supported by adjacent nodes [51] [52]. Moreover, it grantee the adjust nodes always have overlapped active times with number of other nodes. While asynchronous schemes are quite robust and their realisations are relatively straightforward, they typically have high latency and may not support network broadcast. In contrast, in on-demand schemes, the receiving nodes can be forcibly awakened before the communication event. This can be realised by using two different radios: wakeup radio and data radio [53]. It should be noted that the wakeup radio transmission range is much shorter than that of data radio. As a result, a main difference between on-demand and asynchronous schemes are that the on-demand requires that all nodes can be active any time it wants, and can be in communication with all of its neighbours constantly. A rendezvous arrangement schedules a set of nodes acting synchronously to support data transmission at a given time. In other words, the nodes are activated periodically in order to support the communication event and then sleep again until the next connection. The main benefit of this scheme is that, assuming correct functionality, the establishment of network-wide communication channels is guaranteed, making network-wide broadcasting possible by guaranteeing that messages will be sent to all neighbours that mean all neighbours are active with the main node. The scheduled rendezvous network, however, necessitates synchronising all nodes to be active at same time, which requires clock synchronisation [54].

2.4.2 Related Work to Adaptive Sleep Mode Scheme

Over the past years, many WSN sleep mode realisations have been proposed, and the overview of this research can be found in [55] [47] .

Several solutions are proposed to reduce the energy consumption as most of these schemes are beyond the scope of the sleep/wakeup protocol. The reader can refer to [56][57] for detailed surveys on time synchronisation techniques. The application of different protocols varies the sleep/wakeup time of nodes within the network during the system's lifetime.

The most straightforward rendezvous protocol is a Fully Synchronized Pattern where all network nodes wake up for a giving time (T_{active}) with a specified periodicity (T_{wakeup}). Then they will be in sleep mode until the next(T_{wakeup}). However, this scheme is very simple and is used in different application and in MAC protocols also, but it supports a low duty cycle. That because the T_{active} time is quit less than(T_{wakeup}) period. For more enhancements for the above scheme by letting the nodes with no activity detecting to be switch off in this period [53].

The main disadvantages for the Fully Synchronized Pattern are all nodes are active at same period that will lead to a huge amount of collisions, which maximise the latency. In addition, this approach is suitable for flat and structured networks only.

All types of staggered algorithms, which are under a fully synchronised scheme, have main issues. These issues are related to collisions between nodes that are active at the same time. This leads to an increased latency, and effects negatively on the network lifespan.

As a result, for reducing the latency in [47] the author proposed the Shifted Even and Odd Pattern, which is derived from the Fully Synchronised Pattern by changing the wakeup times of nodes by $(T_{wakeup})/2$, which is the wakeup period. This minimised the overall latency.

There are many approaches derived from On-demand scheme, such as Flexible Power Scheduling (FPS) [58]. This technique depends on slotted approach, in which the time is divided into slotted time(T_S). Slots are organised into periodic cycle, each cycle is m slots within duration of time$(T_C = m * T_S)$. For each node has its own radio to be on only, when it has data to send or receive. The FPS is proposed for filling the flexibility gap in slotted approach.

The FPS lacks the synchronisation organisation. As a result, the Twinkle is presented in [59] to develop the FPS, by providing broadcasting traffic and sink connection.

A multi-parent scheme that can be combined with any of the above sleep/wakeup patterns as proposed in [54]. The multi-parent scheme assigns multiple parents (with potentially different wakeup patterns) to each node in the network. These results in significant performance improvements compared to single-parent schemes.

Our adaptive sleep mode model belongs to the sleep/wakeup scheduling category. Currently, most protocols operate under the assumption that activity periods are static, which simplifies the network control but results in either an increase in latency or the maximising of power consumption. Some models, however, address both latency and power consumption but do not consider the structure and the requirements of system model, as well as considering the fact that the energy of an active node is related to its mathematical function for decision-making. In this thesis, the dynamic adaptation of activity period has been implemented for each node individually to minimise overall energy consumption and optimise network performance. As well as, it does not depend on MAC protocol as it is a platform can be in any place. The individual dynamic adaptation of awake up periods resulted in latency reduction and a significant increase in the network lifespan.

2.5 Network Performance Evaluation of M2M

M2M WSNs often suffer from the problem of cluster head reconfiguration to multiple sinks. The utilisation of sink nodes effects on the network performance, especially on the remaining energy for the entire network and the system load. Numerous studies and methods on Self-Optimisation Network have suggested addressing the problem of load balancing in M2M WSN based on IoT via Self Organising Network (SON). Improvements in network performance can be achieved by using Artificial Intelligence (AI) algorithms. Moreover, the benefits of AI techniques while designing load balancing self-organising algorithms are inevitable. Among numerous AI techniques, the Genetic Algorithm (GA) [60] is the most embraced learning algorithm inspired by the process of gene evolution. GA mimics crossover and mutation operations of chromosomes to search for optimal solutions. GAs are used to solve multi-objective optimisation problems. GAs are utilised to solve overloading and power adjustment in WSN. Swarm intelligence is an interesting topic in AI which is inspired by the self-organising behaviour of natural organisms [61]. For example, the natural actions of swarms or bees, a shoal of fish, a flock of birds etc. Many algorithms have been designed to mimic the behaviour of natural organisms, however, the Bee Colony Optimisation (BCO) [62] and Particle Swarm Optimisation (PSO) [63] remain the backbone of swarm intelligence on which all other algorithms are built. Both BCO and PSO are widely discussed in studies related to load management and routing optimisation problems. The main parameters of network performance are in sections below.

2.5.1 Load Balancing

The load balancing has an important objective for improving the power consumption with clustered system and makes the system faster. There are two main polices of performing load balancing as shown in Figure 2.7 [64]:

1- Static Load Balancing policy

2- Dynamic Load balancing policy

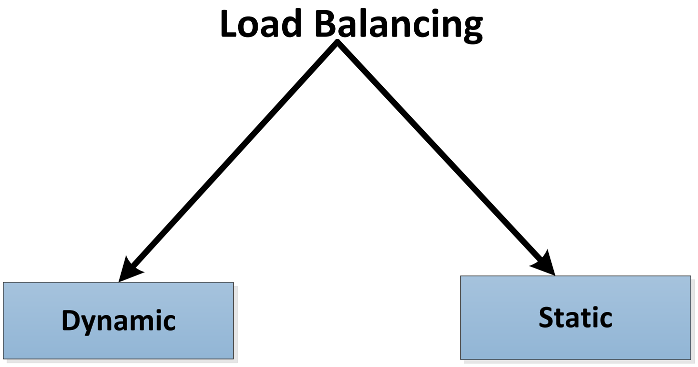


Figure 2.7 Static Load Balancing Policy

The strategies of this type of load balancing evaluate the status of the system statically and consider this information in making decision. The overhead is low in this type of load balancing strategy because the decision is made once only, and this is an essential advantage of it. However it can’t be adaptive with any changes happening in the system [65]:

A. Dynamic Load Balancing Policy

The current event is an essential part of the dynamic load balancing policy. Consequently, the workload is allocated between the processors. Depending on the information that collected from nodes the allocated of the new process to which processor will be. Whenever, one node in the system becomes overloaded, then the task should relocated to an under loaded node [65]:

2.5.2 Related Work to Self-Organising Cluster Head to Sink Algorithm (SOCHSA) Mapping

We anticipate that SON and M2M WSN have become the building blocks for future IoT. Since AI is the basis of SON technologies, it can lead to a major paradigm shift by driving the ongoing efforts in IoT application. Apart from the below studies on SON, AI and Clustering WSN, some closely related works regarding the contributions made in this thesis are discussed as follows. Most of the existing research on WSN focuses on mapping between CHs based on CH load, whereas limited work on CH-Sink mapping based on sinks' load is addressed.

Ways of enhancing the performance of WSN clustering techniques have been widely researched [17]; [8]; [18]. Prior work is reviewed based on a metaheuristic perspective. The authors considered energy consumption, task execution time and energy distribution across the network to find the best task allocation solution.

In order to maximise the system lifetime, PSO algorithm has also been applied to heterogeneous sensor network, followed by heuristic task scheduling in [66]. After computing the optimal physical location of sensors, nodes are scheduled to communicate using PSO and Artificial Bee Colony (ABC) algorithms. This heuristic algorithm performed better compared to random deployment method and allowed to achieve the network lifetime close to theoretical upper limit.

The energy balanced unequal clustering protocol (EBUC) [67] is a centralised clustering protocol, designed to resolve the hot-spot problem by creating unequal size clusters using centralised particle swarm optimisation (PSO) algorithm at the Base Station (BS). The aim of the algorithm is to reduce number of nodes in clusters located near BS. Afterwards, a greedy algorithm is deployed to select relay nodes according node's residual energy and the distance between the node and BS.

A centralised clustering protocol, which involves energy-aware clustering for WSNs implementing a PSO algorithm (PSO-C) at the BS, was put forward in [68]. Available node energy and the distances between the CHs and nodes are considered in this protocol. The algorithm aims to minimise the maximum average Euclidean distance between nodes and CH for all clusters, while seeking to reduce the ratio of the total starting energy of all nodes within the network to the summarised energy of the assigned CHs. In addition, the protocol ensures that only nodes with enough energy can be assigned the CH duty.

An energy-aware evolutionary routing protocol (EAERP) has been described in [13]. This is a single-hop clustering protocol using the BS-centred evolutionary-based optimisation of CH election and the subsequent cluster formation. The objective function of the algorithm is the minimisation of the overall dissipated energy across the network.

This is calibrated as the total energy expended by the non-CHs in dispatching data signals to their CHs, as well as the aggregate energy utilised by the latter to total the data signals and to send these signals to the BS.

In order to balance the load carried by CHs within the network, a GA-based protocol is been proposed in [12]. This protocol deliberately varies cluster sizes to minimise the peak load for all CHs. The CHs are identified a priori, with the objective of optimising the ratio of non-CH nodes to CHs to achieve cluster balance. The objective function of the algorithm is to minimise the standard deviation in the load of the CHs, and therefore provide an even load sharing for every cluster.

The proposed algorithm considers the appropriate trade-off between the fitness function and the different KPIs, to get the best overall performance. The key difference between the proposed work and the previous part is in the application of Evolutionary Algorithms (EAs) for CH re-configuration at the level of the network sink, to optimise residual energy across the all sensor nodes. Furthermore, to improve WSN task allocation a modified version of binary PSO , which has been proposed in [69]. In contrast to the previous studies on clustering and optimisation in WSNs, the proposed work considers important factors including enhanced clustering and transmission techniques, as well as load balancing among multiple sinks (rather than only in CHs), especially for M2M technologies and IoT networks.

2.6 What is Software Defined Network (SDN)

Software Defined Network (SDN) architecture consists of three layers: (i) the Infrastructure Layer (data plane), (ii) the Control Layer (control plane) and (iii) the Application Layer [70]. The SDN’s layers can logically be presented as follows:

The Infrastructure Layer: It is also known as the data plane, and consists of the forwarding elements (e.g. switches and routers) of the network. This layer also has the task of monitoring local information and traffic statistics obtained from the switches and routers [71].

The control layer: It is a software logic that provides forwarding rules for the infrastructure elements, and defines the network operation and routing. Some refer to this logically centralised control unit as the Networking Operating System (NOS) [19]. The separation of data and control planes allows a network operator to control the network behaviour using a single high-level program.

The Application Layer: It handles the monitoring applications that can introduce new features to the network to improve security, manageability, and forwarding schemes.

The communication between the SDN layers is called the southbound and northbound interfaces. The connection between the Control layer and Infrastructure Layer is the southbound interface, which most often uses the OpenFlow standard, the most common data network standard today [72]. The task of the southbound interface is to bridge control and forwarding elements through a well-defined interface. Other standard used in the southbound interface are the Forwarding and Control Element Separation (ForCES) framework [73] [74]. The northbound interface is located between the Control Layer and Application Layer which is such a centralised coordinator unit and based on simple operating system principles. It allows for fine-tuned control over the switches and permits the applications and overall management system to program the network according to the services required.

2.6.1 Benefits of SDN Implementation

SDN implementation significantly reduces network complexity and improves the flexibility of mobility support and scalability. This is mainly due to the separation of the data plane from the control plane, and the simplification of implementing new applications in the network. Another benefit achieved by using SDN is related to the capability of SDN in providing resource management. With SDN implementation, different applications can share the same network infrastructure. The control unit administrates all applications' access to the forwarding element. This is feasible because the controller has a global view of the information carried by the network, and knows the requirements of various applications [74]. Therefore, the SDN allows applications to have precedence over the conventional routing policies. This makes policy decisions across the network more consistent and effective, and also provides a more flexible way to implement new network functionalities. In summary, these benefits simplify modification of the network policies. They keep high-level policies intact by using a control program which reacts to changes in the network state. Finally, because of the centralised controller's knowledge of the network state, the development of networking applications, functions, and services are significantly simplified [75].

2.6.2 What Is SDN-WISE

To meet all IoT application necessities, most of the studies analyses for WSN are based on the SDN. Most of the proposals that considered for solving the main issues are related to the important flow table implementation. However, the devices or network issues are most things to focus on. That is why the main reason for an efficient monitoring on network elements is required and is the core objective of IoT [75]. A full description of SDN in WSN as a prototype and design was presented in [27]. The existing works discussed in this section for SDN in WSN takes different paths. Some are related to SDN in WSN as a programmable part and the others are related to the power consumption models in SDN. All are based on the concept of implanting SDN in WSN.

A. Related Work to SDN-WISE as a Programmable Part

SDN design philosophies in WSN can be logically divided into three groups. First group is sensor OpenFlow and its extension of SDN-WISE. The sensor OpenFlow is the architecture that virtually enables multi-application environments within a single physical network infrastructure. This design aims to provide compatibility between multiple suppliers and simplifies protocol evolution. This philosophy is focused on adaptation of OpenFlow architecture to WSN needs [76].

However, SDN-WISE is the extended version of Sensor OpenFlow, developed to reduce energy consumption by using software-implemented data aggregation and duty-cycling. The power consumption optimisation and multi-tasking support in the Sensor OpenFlow architecture can also be achieved by the implementation of a decoupled centralised control plane.

Task distribution between control and data planes is employed in the second group of architectures to achieve minimal energy consumption without significant performance degradation [26]. The second group of architectures includes Software-Defined Wireless Networking (SDWN) and its extension of smart sensors based on SDN , whose main goal is to optimise energy usage through task distribution between the control and data planes [77]. In the third group, data management and interaction between data and control planes are realised using hierarchical designs. All the above studies in SDN of WSN will be discussed in more detail.

Several schemes are also proposed in the context of SDN in WSNs, which address different challenges involved in the programmable sensor nodes.

A novel SDN over WSN is named Soft-WSN architecture, with sensor nodes based on programmable transceivers has recently been proposed by [77]. Depending on the environmental conditions, the transceiver switches between adequate radio communication channels and standards in order to optimise the network performance as NQoS. Therefore, this approach is capable to address different WSN communication issues in the presence of diverse transmission media.

In [78], the authors propose a software defined sensor network, it is based on a framework in which each node is embedded with multiple sensor nodes. The scheme showed that the energy consumption in sensor nodes can be reduced by embedding each node with several nodes, which are controlled by the centralised controller.

A flow-table implementation mechanism in sensor networks (Sensor OpenFlow) is propose in [76], in which the forwarding rule is defined by a centralised controller. The SDN concepts are useful to WSN to improve the network performance. The implementation of forwarding rules is based on two aspects: compact network-unique address and concatenated attribute-value pairs. Depending on the application, either the sensor node or IoT service application data, which are compared with the flow table to make the decision, such as whether to forward or drop the packet. Thus, based on the application-specific requirements, forwarding rules can be deployed in the network.

In order to reduce information exchange between controller and the sensor nodes, [27] proposed a solution concept for Soft-WSN and is called SDN-WISE. In this design, stateful processes are achieved at the sensor nodes, which are therefore able to adopt adequate procedures (earlier defined by the controller) without referring the controller every time, when needed. Therefore, the proposed scheme reduces the message overhead and energy consumption in the network. Furthermore, in this approach the sink node acts as a gateway between sensor nodes and the controller. This supports data aggregation and radio duty cycling, allowing a periodic radio turn-off to improve energy efficiency. It implements a stateful approach by encoding different features within the data structure, such as WISE state array, accepted IDs, and WISE flow table. The main limitation of SDN-WISE is the necessity to collect data and interact with the controller through the same sink node. This results in an increase of collisions of data collecting routes, as control channels are sharing the same communication channel. This is related to the work described in this thesis where the prototype and constant values are the key factors when considering the payload size and Transmission Time Interval for control signalling periods in constructing a power model.

B. Related Work to Power Consumption Model for SDN-WISE

The energy consumption of transmission equipment represents a significant fraction of capital expenditure (CapEx) tackled by service providers and data centre operators. SDN is a fresh network philosophy which aims to improve traffic on multiple layers, from Ethernet to the top of the network and above. Consequently, SDN is a strong platform that can be applied to different networks [78]. Despite this, the problem of energy used in devices, sub-systems, and systems used to paradigm SDN networks including control signalling, has not been studied in detail and how it effect on the QoS network.

In [79], the author considers the advantage of SDN by allowing the control node to collect important information from the sensors, based on hop counting local information related to traffic, meaning that decisions regarding the flow path are taken by the controller.

In [80], the authors compared the power consumption of SDN OpenFlow and Open vSwitch running on server grade hardware, using numerical modelling and experimental results. The effect of traffic and configuration management on power consumption was estimated. The authors presented a numerical model that can be used to evaluate the power consumption of various network structures and totals of traffic in wired SDN.

In this part of the chapter, which is related to SDN, serves the purpose of providing an insight into the earlier literature studies and how each related work contributes towards shaping WSN in SDN. It also proposes a power model and addresses the problem of explaining and outlining the required parts and the extra control signalling and data traffic of the WSN design based on SDN (SDN-WISE). It then illustrates the SDN approach in WSN (and also in the IoT domain) and shows how a logically centralised controller helps to manage a massive amount of data on a large scale. The existing literature is concentrated on challenges related either to sensor nodes or to flow table implementation, both the device-specific and network-specific problems should be manage in an effectual manner, to provide better observation of physical objects and improving the NQoS and AQoS to the IoT users.

2.7 Summary

Minimising energy consumption is a key focus for researchers, and this has been extensively investigated. This chapter has provided an overview for the essential fundamentals related to the all thesis work. Moreover, for supporting a clear discussion of these techniques in many studies that have worked towards reducing the energy consumption as well as the complexity of the system management.

Some studies concentrated on enhancing the energy efficiency of the system. There are many proposed ideas for clustering routing protocol, with different schemes of electing CHs. However, these approaches do not consider the most important aspects together with sophisticated sensing field. As a result, there is still a need for algorithms to reduce the energy consumption and adapt according to a system’s requirements.

Currently, most protocols operate under the assumption that activity periods are static, which simplifies the network control but results in either an increase in latency or maximising power consumption. Some models, however, address both latency and power consumption but do not consider the structure and the requirements of system model, as well as taking into consideration the fact that the energy of an active node is related to its mathematical function for decision-making.

In addition, minimising the standard deviation of the CH load is the objective function, which provides an even sharing of the load for each cluster. Prior works are reviewed based on a metaheuristic perspective approaches that concentrate on CHs overloaded or on the residual energy without considering the overloaded sinks, which has direct effect on the IoT applications.

In fact, some research scenarios are investigated for SDN in WSN to meet the QoS for IoT, but all of them focus on infrastructure networks issues to deal with it programmable more than analyse the power consumption effective by adding theses extra signalling. In the past a large number of solutions have investigated the topic of QoS support in WSNs but none of them investigates a software-defined approach. There are two major challenges in previous work. Firstly, earlier power consumption models were designed for wired networks and did not consider issues specific to WSNs, which use radio communication channels. Secondly, other models covered specific parts of SDN in WSN but did not include power consumption calculations for the whole network. This thesis is dedicated to SDN-WISE architecture as power consumption model for application-aware service provisioning.

Chapter 3 An Architecture for M2M MLCMS with an Adaptive Sleep Mode based on 6LoWPAN[1]

3.1 Introduction

M2M wireless communication is the key to implementing sophisticated connections between machines. Energy consumption and a shortfall in IP addresses are critical issues in M2M based on IoT.

Minimising energy consumption is a main focus for researchers, and this has been extensively investigated. As energy consumption is one of the main obstacles to improve the technology especially, when a long lifetime network is required. One method for doing so is energy efficient transmission protocols for wireless M2M sensor networks, which are categorised into hierarchal routing, based on clustering types [21]. Consequently, energy efficient hierarchy routing protocol for WSN is presented based on clustering, which provides a platform for controlling M2M networks and prolonging the lifetime of the system. This clustering protocol form lessens energy consumption by aggregating multiple sensed data, which are then transmitted to the sink node [81] . Furthermore, multiple sinks allow for more uniform energy consumption traffic configurations across the network [82]. In addition, it permits sharing of the communication channel of M2M network in a multilevel arrangement and is thus an efficient way of reducing energy consumption. Another method is used to reduce the energy consumption is minimising the activity of transceiver. To this end, an adaptive sleep mode approach is used. As well as, the IPv6 protocol over Low Wireless Personal Area Networks (6LoWPAN) is implemented to fill the gaps of IP addresses as the proposed model is based on IoT.

---

[1] A part of this Chapter is published in IEEE Sensor Journal [22] and part of it “Adaptive Sleep mode”, is published as a book Chapter in Springer.

This chapter promotes a M2MWSN energy efficiency routing protocol for fixed infrastructure called Multi-Level Clustering Multiple Sink (MLCMS) and an adaptive sleep mode based on 6LoWPAN. A part of this work is published in [83], IEEE conference and the second part is accepted as a book chapter in Springer. This is undertaken in two steps:

Firstly, a MLCMS with 6LoWPAN is presented in a fixed infrastructure network. The proposed (MLCMS) model is presented for enhancing the lifetime of a network through a special network structure. The sensor field is divided into quarters with different levels of CHs, depending on a formula for electing CHs for each level. Moreover, two beneficial location sinks are located in the system model.

Since, the proposed model is based on IoT concept, so the main reason for designing 6LoWPAN is to pave the way for IoT. That reasons for constructing 6LoWPAN is to fill the IPs shortage gap, improve the quality of the data transmission by reducing packet losses and to provide more flexibility. Moreover, it impacts on the performance of MLCMS by increasing the packets received. In particular, 6LoWPAN has been playing an important role in extending number of packets received.

Secondly, an adaptive sleep mode scheme is presented, for enhancing the efficiency of power consumption with respect to the delay reduction. Subsequently, inefficient energy consumption caused by nodes being active all the time is tackled by using an adaptive sleep mode solution to maintain high levels of network performance. A sleep mode is defined as where some of the nodes go to sleep voluntarily to save power. This approach is based on a CH's residual energy for managing the active time period and is implemented for fixed infrastructure network which have periodic information. The performance of the MLCMS protocol is evaluated with and without the adaptive sleep mode algorithm.

This chapter is organised as follows: **Section 3.2** presents the problem statement, whilst **Section 3.3** explains the system model architecture and cluster head electing. It also describes the 6LoWPAN standard and the energy model employed. The adaptive sleep mode approach is presented in **Section 3.4**. The results of the proposed model when compared with other protocols are presented and discussed in **Section 3.5**. Summary remarks are provided in **Section 3.6.**

3.2 Problem Statement

There are many challenges related to WSN that need to be focused on to find suitable solutions, which are summarised in Figure 3.1 and outlined below.

- **Firstly**, energy consumption is critical when designing a wireless M2M sensor network; the sensor nodes need to be self-governing with small batteries that last for several months or even years, since replacing batteries for a large number of devices is impractical in distant or unfavourable environments [84].
- **Secondly**, sink node isolation has been noted as a direct result of energy consumption imbalances in WSNs, referred to as a hotspot or energy hole problem, and energy is the most crucial resource [9].
- **Thirdly**, apart from energy efficiency, there is the demand for more IPs address.
- **Finally**, power being consumed unnecessarily by nodes being active all the time is considered as a critical issue in WSN energy efficiency and main cause of minimising the network lifetime of network.

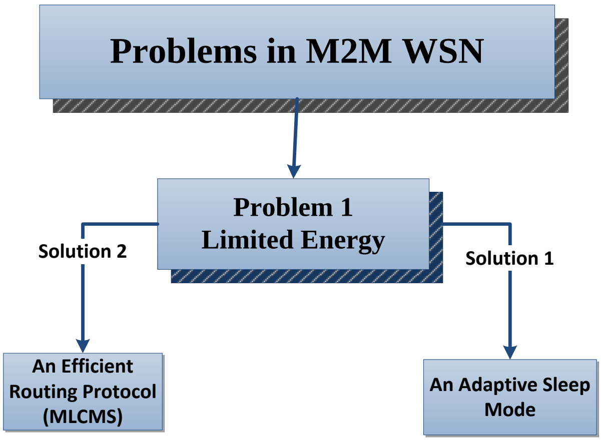


Figure 3.1 Problem statement and solution in M2M WSN

3.3 Proposed Model of MLCMS Routing Protocol Based On 6LoWPAN

The proposed M2M routing protocol for WSN is a MLCMS based on 6LoWPAN, for enhancing the lifetime of a network through a special structure. The sensor field is divided into quarters and levels of (CHs) with two sinks. The performance of the MLCMS protocol is evaluated and compared with the LEACH protocol and M-LEACH. In this section, two schemes for an energy efficient routing protocol in a fixed infrastructure are proposed as follows.

- MLCMS with 6LoWPAN is promoted according to a mathematical function for selecting CHs, which considers all the important aspects for electing CHs and this will help facilitate prolonging the lifespan of those selected. That is, the approach and the parameters selected for electing the CHs play important roles in prolonging the network lifetime.
- An adaptive sleep mode scheme as a solution for maintaining high levels of energy efficiency and delay reduction.

3.3.1 Network Architecture

A fixed infrastructure for a network is modelled by a graph $\mathbf{G = (N_n, V)}$, where $\mathbf{N_n}$ is the set of sensor nodes and $\mathbf{V} = \{(\mathbf{u}, \mathbf{n}) \subset \mathbf{n}/\mathbf{D}(\mathbf{u}, \mathbf{n}) \leq \mathbf{R}\}$ represents the wireless communication between nodes. R is the transmission range, and D(u,n) is the distance between node u and n. A hierarchical network, in which a sensor field is logically divided into quarters, with levels of CHs in each one. Each CH has a related sink, a CH and the remaining components are the ordinary sensor nodes.

In the proposed work the sensing is divided by (4) just a prototype to examine the proposed routing protocol (MLCMS) on multi-layers network structure and it can be divided by 6 or 8 according to size of the network area and number of sinks.

It is assumed that n a sensor node in an (N×N) sensing field, which is considered the size of the sensing network used, as shown in Figure 3.2. The proposed network model contains the following elements:

- **The nodes:** Distributed randomly such that coordinate (x,y) position for each will not provide uniform distribution.
- **Two stationary sinks:** One is located in the middle of the left side ($(1/4)\,\mathrm{N}$, $(1/2)\,\mathrm{N}$) and the other is in the middle of the right side (($3/4)\,\mathrm{N}$,$(1/2)\,\mathrm{N}$), as shown in Figure 3.3.
- **The transmission range of the sensing field:** Each quarter is divided into three levels horizontally. At each level, there is a maximum transmission range (TR), which is the diagonal length of the level, as shown in Figure 3.2. Depending on the probability of farthest node in each level as in Equations (3.1), (3.2) and (3.3). In other words, The CHs region, i.e. the transmission range, is based on the longest distance between the CH and farthest node in each level. Nodes that are beyond this TR are unable to connect with the CHs, whereas those within this range of the level use TR to transmit their parameters (residual energy, number of neighbours and distance to sink) so that they might themselves become the new CH, and once chosen, the other nodes will send data to it.

The maximum T.Rs are formulated below:

The transmission range of cluster head to node (T.R of CH-to-Node) is:

$$\mathbf{H_{CH-Node}}=\sqrt{(\boldsymbol{a})^2+(\boldsymbol{b})^2} \qquad (3.1)$$

The TR of CH-to-CH is:

$$\mathbf{H_{CH-CH}}=\sqrt{(\mathbf{a})^2+(\mathbf{2}*\mathbf{b})^2} \qquad (3.2)$$

The TR of CH or Nodes-to-Sink is:

$$\mathbf{H_{CH-Sink}}=\sqrt{(\mathbf{0.5}*\mathbf{a})^2+(\mathbf{3}*\mathbf{b})^2} \qquad (3.3)$$

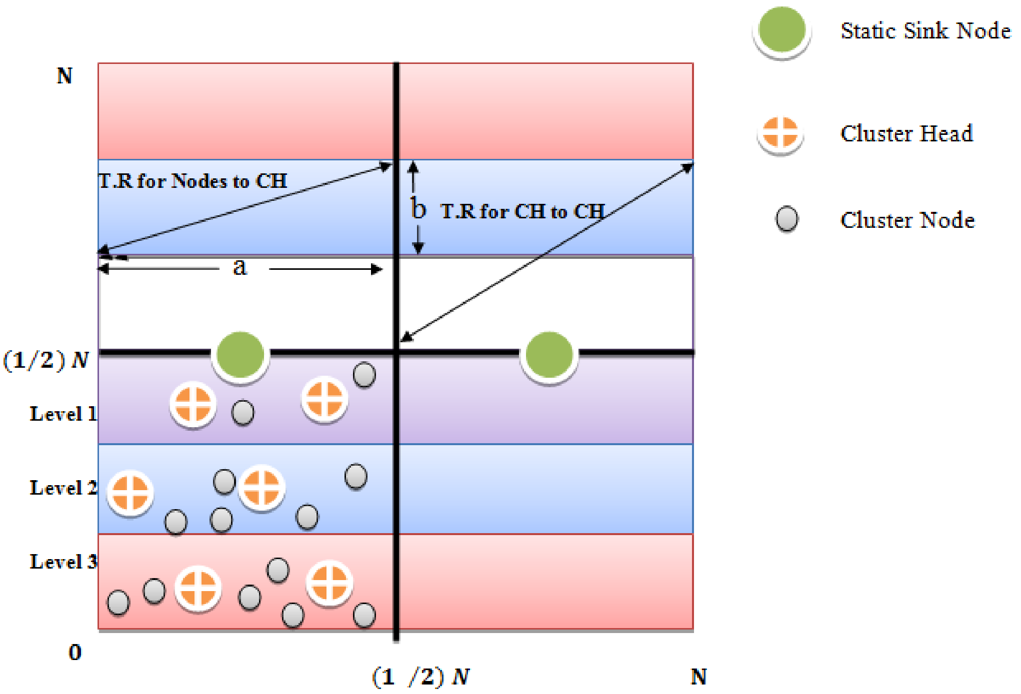


Figure 3.2 System model

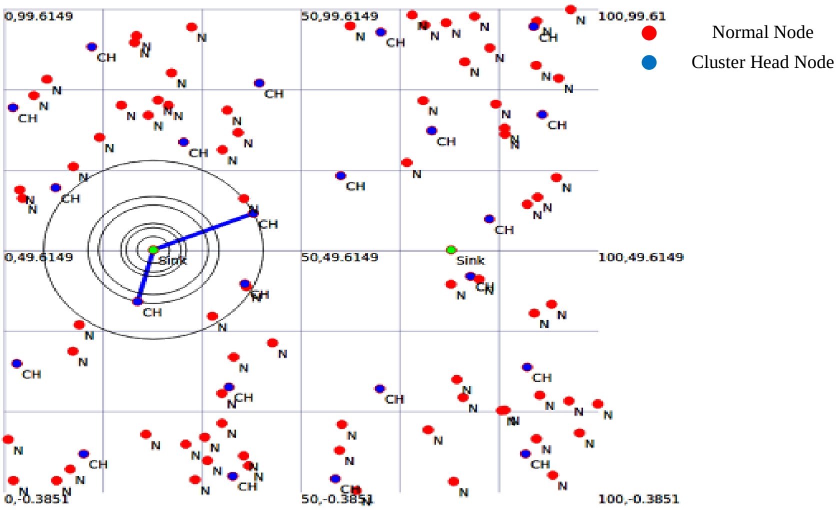


Figure 3.3 The Network model using the NetAnim animation tool in NS3

3.3.2 Cluster Head Selection and Formation

The main objective of forming a CH is enhancing the energy efficiency of the system by considering the important parameters that can improve the energy consumption, as well as rotating the CHs. The CH selection procedure is based on a mathematical function, as shown in Equation (3.4), where the node in each level with maximum weight will be selected as the CH according to three parameters: residual energy, degree and distance to sink.

Residual Energy: is an essential parameter in WSN, for when a node has low residual energy means it does not have the capability of transmitting any data to the destination, especially if the distance between the node and the receiver is quite far. Many researchers have considered the residual energy parameter during the phase of electing the CHs to decrease the probability of them dying fast and thus, impact positively on network lifespan performance. In the beginning, all sensor nodes in the system have the same amount of energy and this value is known as the initial energy, as entered in table 3.2 [85]. The residual energy is the battery level and taking into consideration the consumed energy in the process. It is measured by taking away the total energy spent by each node from its initial energy [86].

Degree: this refers to the number of node's neighbours, which are defined as one hop nodes. Many studies have chosen the degree of the node as one of the parameters in selecting CH to enhance the stability of the system, as it will be increased by highest number of neighbours to the selected CH.

The distance between nodes and the sink: This characteristic is considered an essential part of selecting CHs as the transmission power is influenced by the distance between the source and destination. Consequently, when the number of hop is small this will minimise the energy consumption [87].

The operation of MLCMS routing protocol is divided into two parts in each round, which are the set-up and steady-state phases.

A. *Set-up Phase*

The proposed approach involves building a hierarchical structure with a sophisticated CH distribution. The selection of the CHs in each quarter is considered as the set-up phase, which involves the central node or that nearest to the centre for each of the three levels in each quarter, initially sending a Hello message to all the nodes at the same level, as shown in Figure 3.3. Only those within the T.R can receive the message and hence, send their information regarding residual energy, number of neighbours and their position (x, y). In this phase, the node with the highest weight is elected as the CH. The weight of the node is calculated by Equation (3.4), which takes the residual energy, the number of neighbours and the distance to the sink into account. Thus, more balanced clusters regarding energy are generated and the position of the two nodes with the maximum W(n) are then selected as the CHs of this level. These CHs are uniformly distributed to prevent them from being too close to each other, as shown in Figure 3.3. Then, the elected CHs broadcast their election to the rest of the nodes by sending an INVITE message. The nodes within their transmission range will receive the INVITE message and will measure the signal strength to join the CH that has the strongest signal. Furthermore, the CH's energy is checked regularly, and if it is less than 25% of its residual energy, it should be changed. Where, 25% of the residual energy is considered as a reasonable minimum for a CH to successfully perform operations such as information collection from its connected nodes or other CHs, transmission to other CHs or sink, and electing a new CH. Moreover, 25% of the CH residual energy is selected to allow graceful reduction of network lifetime rather than abrupt fall in network lifespan and CH failures.

The new CHs for each level will be selected by the old CH sending a request to all related nodes for information required to calculate W(n). The one that has maximum will be awarded the role of new CH for that level, depending on Equation (3.4). The CH must have the highest residual energy to be able to manage the intra-cluster (the communication between the node with its CH) and inter-cluster communications (the communication between CH to CH). To minimise the consumed energy in such communication, the nearest node to the sink is selected as the CH.

$$\mathbf{W(n) = \varpi_1 E(n) + \varpi_2 Deg(n) - \varpi_3 D(n)} \qquad (3.4)$$

$$\mathbf{0 \leq \varpi_1, \varpi_2, \varpi_3 \leq 1}$$

$$\mathbf{\varpi_1 + \varpi_2 + \varpi_3 = 1}$$

Where W(n) is the ratio weight of each nodes, $\varpi_1, \varpi_2, \varpi_3$ are the effect factors, $\mathrm{E(n)}$ is the residual energy of node n, $\mathrm{Deg(n)}$ is the number of neighbours of node n and the $\mathrm{D(n, s)}$ is the distance between node n and the sink.

$$\mathbf{E(n) = E_{in}(n) - E_{TC}(\boldsymbol{n})} \qquad (3.5)$$

$\mathrm{E(n)}$ is the residual energy, $\mathrm{E_{TC}}(n)$ is the total consumed energy of node n and $\mathrm{E_{in}(n)}$ is the initial energy of node n.

$$\mathbf{D(n,S)} = \sqrt{(\mathbf{X_2} - \mathbf{X_1})^2 + (\mathbf{Y_2} - \mathbf{Y_1})^2} \qquad (3.6)$$

where, (X,Y) is the coordinator for the node (n) and the sink (s).

The priority levels or weights selected for Equation (3.4) are based on the Rank Order Centroid (ROC) method [88]. The ROC is a simple method of assigning weights to a number of functions, ranked according to their priority or importance. The priority of each function is taken as an input and converted into its weight. The conversion is made by Equation (3.7):

$$\boldsymbol{\varpi}_{\mathbf{i}} = \left(\frac{\mathbf{1}}{\mathbf{F}}\right) \sum_{\mathbf{n=i}}^{\mathbf{F}} \frac{\mathbf{1}}{\mathbf{n}} \qquad (3.7)$$

Where, F is the number of functions ($\mathrm{E(n), Deg(n) and\ D(n,s)}$) and ϖ_i is the weight of the ith function. E(n), ranked first, is weighted as $(1/1 + 1/2 + 1/3)/3 = 0.6$, Deg(n), ranked second, is weighted as $(1/2 + 1/3)/3 = 0.3$ and D(n), ranked third, is weighted as $(1/3)/3 = 0.1$. The CHs selection phase is described in Algorithm 3.1 based on the parameters in Table 3.1.

Table 3.1 Parameter Definitions used in Algorithm1

Parameters	Definition
$\mathrm{N_n}$	Number of nodes
D(n,s)	Distance between n and its sink(s)
Deg(n)	Number of neighbours of one hop
W(n)	Ratio wight of node n
State of n	State of n: “CH” or “NN”
CH	Cluster head
NN	Normal node
$\mathrm{L_v}$	Level
E(n)	Residual energy
$\mathrm{E_{in}(n)}$	Initial energy
$\mathrm{E_{Tc}(n)}$	Total consumed energy

Algorithm 3.1 CHs election phase

```
Step1. In each Level($L_v$) centered node send 'hello' message to all nodes within T.R.

Step 2. Find the degree of each node
Deg(n) = Number of one hop neighbors

Step 3. Calculate the residual energy $E(n)$.
$E(n) = E_{in}(n) - E_{TC}(n)$

Step 4. Calculate the distance to the sink.
$D(n) = \sqrt{(X_2 - X_1)^2 + (Y_2 - Y_1)^2}$

Step 5. Calculate the weight.
$W(n) = \varpi_1 E(n) + \varpi_2 Deg(n) - \varpi_3 D(n)$

Step 6. Find the biggest in this $L_v$.
W (n) = biggest W ($L_v$)

Step 7. state(n) = CH

send cluster head_elected
    OR
state(n) = NN
send cluster head_accepted
```

B. Steady-State Phase

The second phase is the steady-state phase when the CHs are formed and each node starts to send its sensed data to them. The CHs aggregate the received data and forward the information to the sink. One crucial factor of the model is that the CH at level three in each quarter cannot connect directly to a sink. It should be attached to the CH in level two, but if there is no CH at this level, it can connect to the nearest CH and if not, then it will send directly to a sink. Hence, this is a proposal for a multi-level clustering (CH to CH and then to a sink), which involves some CHs being far from a sink, but still being able to send their data to a faraway sink, thereby linking CHs on multiple levels without losing excessive amounts of energy. In addition, there is a particular technique for sending data from the CHs to the sinks, which depends on the priority of the farthest CH.

The clustering method proposed in the MLCMS protocol has many advantages. An important benefit is that the CHs are located in a more uniform way, which leads them to comparing favourably with the probabilistic deployment in other routing protocols, which is, in fact, more suitable for large-scale networks. Moreover, it can prolong the network lifetime. Also, there is the flexibility of scalability in extending the size of the network by increasing the number of levels in the same way.

3.3.3 Energy Consumption Model

The proposed model is based on energy model in [89] [23], but with many changes according to the MLCMS algorithm. The model is based on free space (d^2) or multi-path fading (d^4) channels being employed depending on the distance between the sender and receiver. If the distance is less than the threshold ($d_\circ$), then the model of free space is used, otherwise the adopted model will be the multipath space. In the given model, E_{TX} of energy is consumed by each sensor node to transmit an L-bits packet, which is the length of the packet over distance d, this being the distance between the transmitter and receiver.

In this model, the electronic energy E_{elec} is the energy dissipated per bit to run the transmitter or the receiver circuit, where ε_{fs} and ε_{mp} represent the transmitter amplifier’s efficiency of channel conditions. For $d \leq d_\circ$, ε_{fs} is used to represent the free space conditions, while ε_{mp} reflects longer distance affected by multi-path fading [90] [91]. However, in the MLCMS model the values of ε_{fs} and ε_{mp} are changed according to the nature of the transmission, which depends on the transmission range of the proposed model as in Equations (3.1), (3.2) and (3.3). The cluster-based network can take three forms: inter-cluster transmission (CH-to-CH); intra-cluster transmission (cluster member-to-CH); and CH-to-sink transmission, based on [89] with respect to the MLCMS infrastructure requirements.

The energy dissipated for transmission one packet over distance d between nodes is formulated by:

$$\mathbf{E_{TX}(L, d)} = \begin{cases} \mathbf{L * E_{elec} + L * \varepsilon_{fs} * d^2, d \leq\ d_\circ} \\ \mathbf{L * E_{elec} + L * \varepsilon_{mp} * d^4, d >\ d_\circ} \end{cases} \quad (3.8)$$

where:

$$\mathbf{d_0} = \sqrt{\frac{\boldsymbol{\varepsilon}_{\mathbf{fs}}}{\boldsymbol{\varepsilon}_{\mathbf{mp}}}} \quad (3.9)$$

The energy that is consumed by the receiving packet is:

$$\mathbf{E_{RX}(L) = L * E_{elec}} \quad (3.10)$$

The value of ε_{fs} and ε_{mp}in the equation is according to the transmission forms and if it is CH-to-sink these values will be the same, as shown in Table 3.2 for simulation parameters.

Whereas for transmitting cluster member-to-CH the ε_{fs1} and ε_{mp1} will be used as below:

$$\boldsymbol{\varepsilon}_{\mathbf{fs1}} = \frac{\boldsymbol{\varepsilon}_{\mathbf{fs}}}{\mathbf{K}} \quad (3.11)$$

$$\boldsymbol{\varepsilon}_{\mathbf{mp1}} = \frac{\boldsymbol{\varepsilon}_{\mathbf{mp}}}{\mathbf{K}} \quad (3.12)$$

Where:

$$\mathbf{K} = \frac{\mathbf{H_{CH-Sink}}}{\mathbf{H_{CH-Node}}} \quad (3.13)$$

While for transmitting CH-to-CH, ε_{fs2} and ε_{mp2} will be used as below:

$$\boldsymbol{\varepsilon}_{\mathbf{f}s\mathbf{2}} = \frac{\boldsymbol{\varepsilon}_{\mathbf{fs}}}{\mathbf{C}} \quad (3.14)$$

$$\boldsymbol{\varepsilon}_{\mathbf{mp2}} = \frac{\boldsymbol{\varepsilon}_{\mathbf{mp}}}{\mathbf{C}} \quad (3.15)$$

Where:

$$\mathbf{C} = \frac{\mathbf{H_{CH-Sink}}}{\mathbf{H_{CH-CH}}} \quad (3.16)$$

In the proposed work, when the transmission between nodes is initiated, then the energy equation will be used depending on the conditions above.

3.3.4 Construction of 6LoWPAN over MLCMS

6LoWPAN networks are connected to the Internet through the 6BR (6LoWPAN Border Router) [11]. Most hierarchical networks use a routing protocol based on IEEE 802.15.4, but this lacks the extension of IPs. Low power and short range wireless communication technologies have an important role in the efficiency of network communication and these technologies are commonly adopted in WSN and M2M systems. Given the lack of IP addresses and the need to overcome this, it has been demonstrated that 6LoWPAN can lessen power consumption and compress the size of the packets [11]. Moreover various IETF Working Groups have been facilitating the integration of low-power wireless networks into the Internet and defining a complete IoT stack for WSN [41]. The aim of using 6LoWPAN is to overcome the shortage of IP addresses. Moreover, it allows for dynamic short addresses capabilities, which involve the assumption that the routing protocol occurs in the adaptation layer. In addition, using 6LoWPAN provides more opportunities for M2M services, with better packet transmission quality and reduced packet loss. The main advantage of 6LoWPAN is the adaptation layer, which involves using stacking of headers, as well as adding the definition of encapsulation header stack in front of IPv6 datagram.

3.4 Proposed MLCMS with an Adaptive Sleep Mode Scheme

Based on the same structure and energy model that have been described in the previous sections, a method to enhance the lifetime of an M2M WSN by improving its energy efficiency is proposed. To prolong the network lifespan, some sleep scheduling methods are always employed in WSNs. There are many ways to meet the requirements of an effective sleep mode in terms of how and when the nodes are scheduled to be active or sleep. The main aim is to maximise the energy efficiency by controlling the supply power according to the requirements. In this section, an adaptive sleep mode based on an independent sleep/awakeup protocol for scheduling the scheme is proposed.

The scheme is aimed at scenarios where sensor nodes, which have periodic information, are monitoring the state of the environment. That is, the proposed hierarchy clustering topology architecture MLCMS with 6LoWPAN includes the addition of an adaptive sleep mode to extend the lifetime of the system. During the entire functional cycle after

electing the CHs, the MLCMS model divides the time period for time active ($\mathbf{T_{active}}$), which is the active time period of each level according to the multi-level clustering. This makes the CHs at each level active for specific periods of time $\mathbf{T_{active}}$, and the overlapping in $\mathbf{T_{active}}$ between the levels of CHs is such that on one or two levels their CHs are awake at the same, as shown in Figure 3.4. This adaptive sleep method extends the lifetime of the system as well as minimising the delay in transmission. Energy conservation is generally addressed in two ways:
(1) Alternating the nodes between sleep and active modes in an optimal manner.
(2) Efficient trade-off between energy consumption and connectivity through effective control of transmission power.

The CHs sleep along with its nodes for some duration of time, to reduce the level of contention among the nodes. Hence, the probability of collision (which is usually associated with wasted power as the node needs to retransmit the collided packet again) is reduced and is reflected into a significant reduction in the end to end delay. An effective process of adaptive scheduling to maximise energy saving is the use of sleep and active modes amongst the sensor nodes. Scheduling of these two modes is initiated immediately after CHs have been created (one or two CHs in each level).

Setting $\mathbf{T_{active}}$ for each level and the length of active period for each CH in this level depends on a mathematical function that considers the CHs' residual energy. For instance, the CH with the greatest residual energy will be active for more time than those with less energy, while if there is just one CH in the level then the active time for this CH will be same as the $\mathbf{T_{active}}$, as the active time of the level. This CH is kept active with its nodes until the period of activation is finished, it will fall into a sleep mode. Any active CH needs to identify the nodes related to it to be active at particular time. This will be communicated through a WORK message from the CH to its nodes. Also, the CH sends a SLEEP message to all the other nodes, telling them to be inactive in the next period scheduled in the adaptive sleep mode scheme.

The protocol is aimed at data collection applications (e.g. for monitoring purposes, where the sensor nodes need to report to a sink node periodically same information) and compared to others, it has two distinct advantages. First, because it is not linked to a specific MAC protocol, it is applicable to a range of sensor platforms. Second, it can quickly adjust the sleep/active patterns to suit variations in network conditions, e.g. changes in traffic demand, network congestion etc.), thus leading to improvement in energy resources deployment. Thorough simulation analysis, the outcomes show that owing to the injected flexibility, adaptive sleep mode performs better than the original scheme of MLCMS without it in relation to energy efficiency, message latency and prolonging system lifespan. Moreover, the adaptive sleep mode approach has several other strengths. Based on the CH location in the clustering routing protocol it suggests the schedules of the nodes. This helps to minimise the latency also when the nodes are sleeping for most of the time and thus, reduces energy consumption. In the promoted algorithm, the active period of each CH with its nodes is adjusted to keep the idea of multi-level clustering by considered the overlapping in time between CH levels and the operating conditions experienced by those nodes. The adaptive sleep mode is capable of adjusting the network topology and external conditions.

Consequently, the sleep adapted protocol emerges as offering a substantial improvement in monitoring, thereby making the long-term deployment of WSNs more likely. This adaptive sleep approach not only maximises the energy efficiency, but also minimises the latency with respect to the fixed structure of the algorithm scheme. In addition, providing different lengths of active period for CHs in same level based on their residual energy, impacts positively on the lifetime of the network. A CH with high residual energy will have a longer active period time as one at the same level with less energy. Moreover, these CHs will be active with their nodes such that those on same level will be active at different times, which will decrease the number of collisions.

The adaptive sleep mode scheme depends on mathematical formulas to decide the active time for each CH in each level, these formulas are following:

$$\boldsymbol{\alpha} = \frac{\mathbf{E(n)}}{\sum_{\mathbf{n=1}}^{\mathbf{CH}} \mathbf{E(n)}} * \ \%\mathbf{100} \tag{3.17}$$

Where $\boldsymbol{\alpha}$ is the percentage value from $\mathbf{T_{active}}$ to find the active time for each CH $(\mathbf{T_{CH}})$. In addition, this formula is used to find the active time for the CH $(\mathbf{T_{CH}})$ given as;

$$\mathbf{T_{CH}} = \boldsymbol{\alpha} * \mathbf{T_{active}} \tag{3.18}$$

The steps for summarising the adaptive sleep mode are in algorithm 3.2.

Algorithm 3.2 Adaptive Sleep Mode approach

Step1. Initialise the parameters, including time $\mathrm{T_{active}}$ **(**which is the active time of each level) the number of CHs (n), the number of levels ($\mathrm{L_v}$) and $\mathrm{E(n)}$ is the residual energy of node n according to Equation (3.5).

Step 2. The sink will receive information from the CHs regarding their residual energy, and in some situations *(when there are two CHs in the same level), it will perform some computations as in Equation (3.18) to get the percentage of active time for each CH in same level from the total active time period $\mathrm{T_{active}}$ regarding this level. However, if there is just one CH in this level then the active time for this CH will be equal to the $\mathrm{T_{active}}$, which is the active period for the level.

Step 3. Calculate α, which is the percentage value of active time for each CH based on the formula:

$$\alpha = \frac{\mathbf{E(n)}}{\sum_{\mathbf{n=1}}^{\mathbf{CH}} \mathbf{E(n)}} * \ \%100$$

Step 4. Calculate the activation time for each CH at each level according to the formula:

$\mathbf{T_{CH}} = \alpha * \mathbf{T_{active}}$

Step 5. Sink informs each CH with their scheduling time.

Step 6. CHs broadcast the active time command to its nodes and then enter an active mode according to the scheduling setting.

Step 7. CH informs the CHs in the level beyond and should update its scheduling time, if any change in CH has happened. Moreover, the CHs exchange their time periodically to avoid any clock drift.

Step 8. When CH is changed the new elected CH should inform the sink with its information regarding its residual energy in order to get new scheduling settings.

Obviously, the active CHs in each layer of each quarter in the sensing field must be partially overlapped to allow CHs to communicate with each other. Finally, even under the assumption of static nodes, changing of the network topology and traffic modification is still considered. The active period of the CHs and their nodes should thus adapt dynamically to such variations.

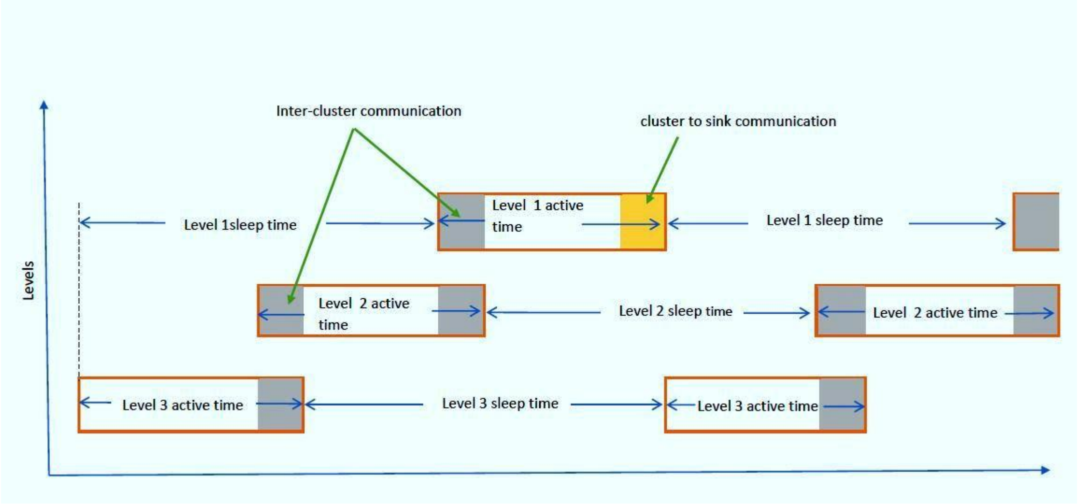


Figure 3.4 Scheduling time for an adaptive sleep mode

3.5 Results and Discussion

The NS3 is a Network Simulator for modelling different WSNs. It has been used to simulate the proposed M2M MLCMS algorithm by specifying its characteristics, and depends on the simulation parameters shown in Table 3.2. In this section, the proposed algorithm is compared with LEACH and M-LEACH, commonly deployed algorithms, in terms of their performance. Our proposal outperforms both of the latter algorithms regarding energy conservation and hence prolongs network life time. In addition, simulations are undertaken for MLCMS over 6LoWPAN with and without the adaptive sleep mode algorithm, thereby also addressing the efficiency of the method proposed. The performance indicators are network lifetime, residual energy at the end of each round and the total number of packets received by a sink. It should be noted that the sensor nodes arrangement for the model is randomly chosen, also their positioning in the square area, which implies that the results are independent of node location. All the parameters pertaining to the LEACH and M-LEACH algorithms are the same as those employed in Table 3.2, which also describes the parameters of the scenarios.

Table 3.2 Simulation Parameters and Values

Parameter Name	Value
Number of sensor nodes (N_n)	100
Base station locations (s)	(25, 50), (75, 50)
Length of the packet (L)	4,000 bits
Initial energy of the sensor nodes (E_{in})	0.5 J
Energy consumption on the circuit (E_{elect})	50nJ/bit
Free space model of transmitter amplifier (E_{fs})	10 pJ/bit/m^2
Multi-path model of transmitter amplifier (E_{mp})	0.0013 pJ/bit/m^4
Network size (N*N)	100 m ×100 m

The efficiency of the M2M MLCMS routing protocol and the performance of using 6LoWPAN as well as the proposed adaptive sleep mode are considered according to the following characteristics.

- **Life time of the System**

This is the time from the beginning of the network processing until the last node is dead in the network. It should be noted that recharging the nodes' batteries is impossible therefore, the lifetime needs to be as long as possible [15] [23].

- **Stability**

This refers to the time interval from the beginning of the network operation until the first node is dead [23].

- **Scalability**

The scalability of the system is measured according to the degree to which increasing the number of nodes on the same size of sensing field will decrease system performance.

- **End-to-End Delay**

This is the difference between the time of sending a packet from a node to the sink and the time that it is received at the sink.

$$\mathbf{EED} = \mathbf{T}_{\mathbf{Recived}} - \mathbf{T}_{\mathbf{Transmitted}} \qquad (3.19)$$

In Equation (3.19), $T_{Recived}$ is the time of receiving a packet by the sink from the sensor node, while the time for transmitting it is represented by $T_{Transmitted}$.

3.5.1 Performance Evaluation of MLCMS Based on the 6LoWPAN Model

Drawing on the definition of the stability, it is clear from Figure 3.5 that for M-LEACH and LEACH their first node is dead before that of MLCMS, hence it has superior performance in terms of stability. Specifically, it shows that the first dead node of MLCMS is in round 1,231, whereas those for M-LEACH and LEACH die in rounds 1,025 and 455, respectively. MLCMS has an overwhelmingly superior performance in terms of extending life of the first dead node, the time when half the nodes are dead and the last node dying, as shown in Table 3.3.

Table 3.3 Comparison of Routing Protocol Stability

	First Node Dead (Round)	**Half Node Dead (Round)**	**Last Node Dead (Round)**
MLCMS	1,231	2,319	3,716
M-LEACH	1,025	1,309	1,917
LEACH	455	1,009	1,447

Moreover, in Figure 3.5 the number of "alive" nodes across the rounds is illustrated. It can be seen that MLCMS lasted for 3,716 rounds, whilst LEACH and M-LEACH achieved 1,500 and 1,900 rounds, respectively. This is observed in the shape of the graphs, where a steep decrease in LEACH and M-LEACH shows where energy equalisation is attained. When last node is reached, that means all nodes are depleted of their energy resources, leading to this quick decline. On the other hand, MLCMS shows a gradual decrease in the number of alive nodes, which is due to the fact that energy equalisation is not coordinated and some nodes run out of battery energy much more quickly than others. Furthermore, an early jump for LEACH and M-LEACH in number of active nodes is observed and sharp decrease as shown in Figure 3.5. The jump is because of many sensors having depleted of their energy resources.

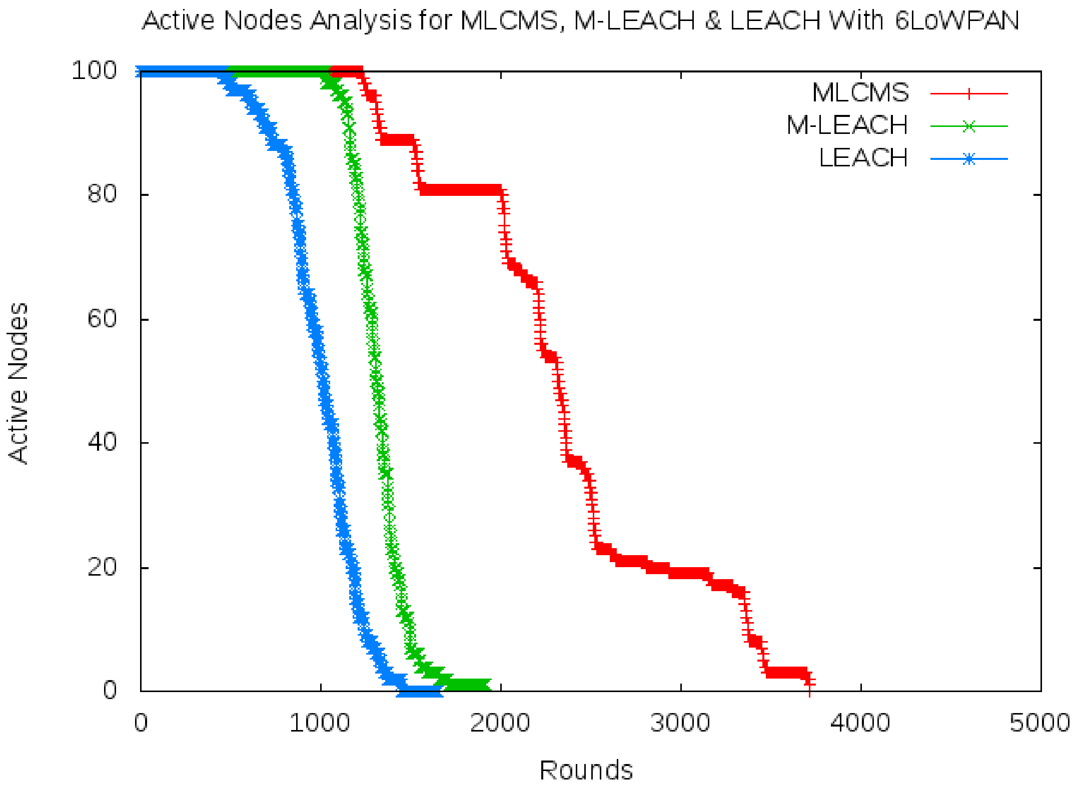


Figure 3.5 Number of active nodes

In addition, the performance of the proposed model improves the lifetime of the system over that of the LEACH and M-LEACH architecture, as given by percentage difference from the following equation:

$$W = \frac{PA - PB}{PB} \times 100\% \qquad (3.20)$$

where, PA is the performance of MLCMS, and PB is the performance of the M-LEACH and LEACH architectures. From Equation (3.20), the improvement of the MLCMS model in extending the lifespan of the system over the LEACH and M-LEACH architecture is 147% and 93%, respectively. Clearly, the number of active nodes in MLCMS can extend the lifetime of the system more than the other two algorithms. A noticeable difference between MLCMS and both LEACH and M-LEACH is that when the latter two lose all their nodes, MLCMS still has some node that are functioning. To address the question as to which protocol is best for collecting data, the total number of messages each delivers throughout the lifetime of the system is calculated.

Another essential criterion in the routing protocol for WSN is the total number of packets received by a sink and this is measured for LEACH, M-LEACH and the proposed scheme MLCMS. In Figure 3.6, it can clearly be seen that the total number of packets received for MLCMS is more than the LEACH and M-LEACH protocols, which demonstrates that it provides the best performance in this regard.

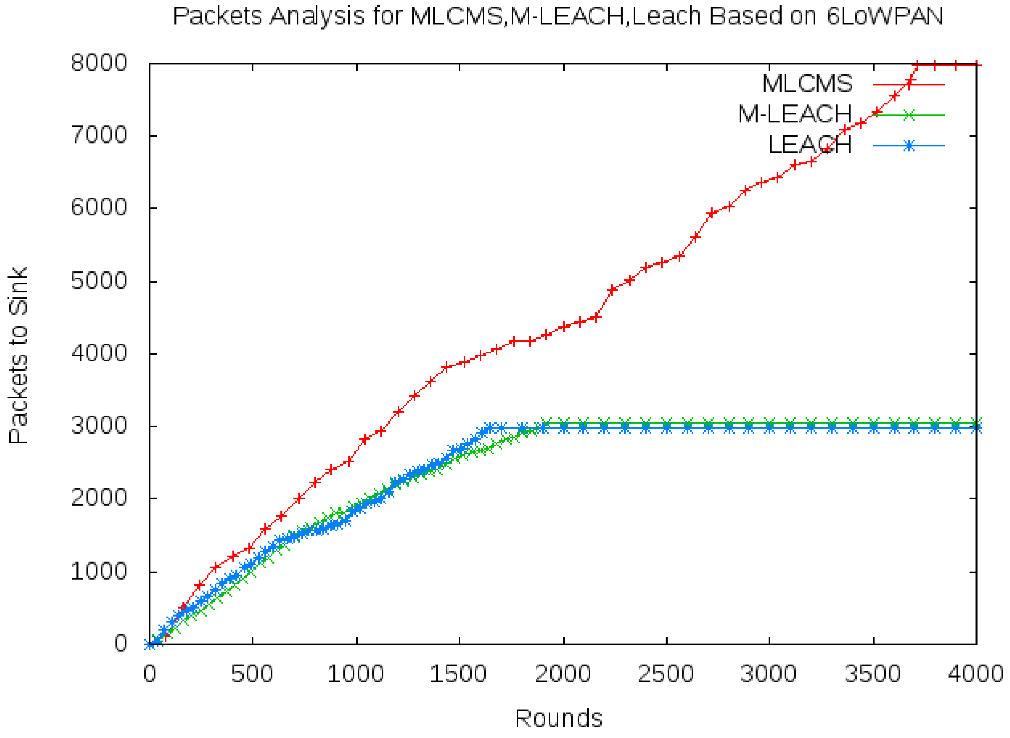


Figure 3.6 Number of total packet received by sink

In Figure 3.7 the scalability of MLCMS with respect to same network size, namely 100, 200, 300 nodes. This effect is illustrated for increasing number of nodes differently, plotting the active nodes corresponding to the rounds. It is observed that MLCMS have a consistently longer lifetime by increasing the number of nodes. As expected, lifetime decreases for algorithm since the number of nodes has been increased in the network because the need for forwarding more data due to increase in the number of nodes on each region, especially at locations closer to the sink. That means the scalability and stability of MLCMS algorithm. In addition, a sophisticated way of constructing the network model is by dividing the network into four regions that provides more scalability for the system as we can extend the region size symmetrically.

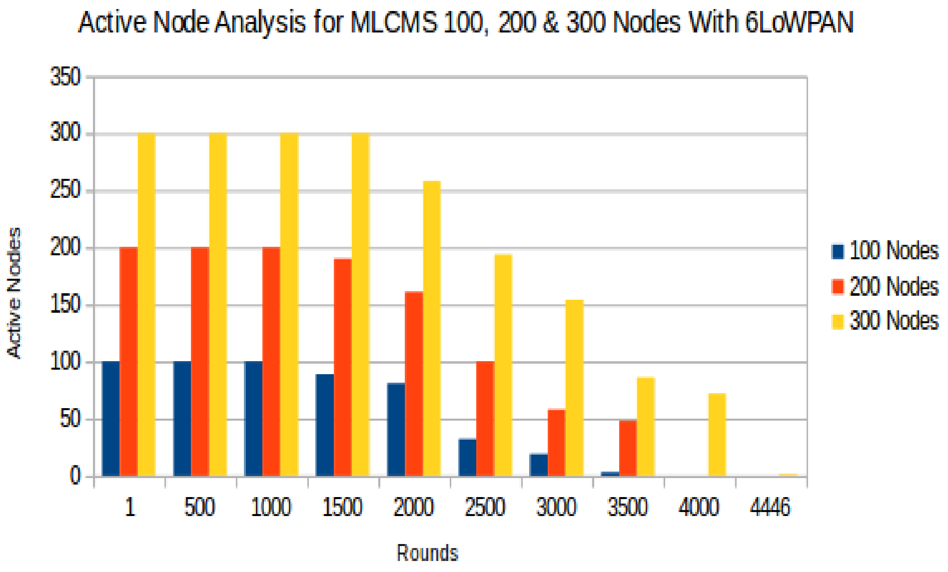


Figure 3.7 Number of active nodes

The amount of data received by the base station is an important factor for quality analysis of any routing protocol. Figure 3.8 shows that the number of packets received by the sink has improved by 7% using 6LoWPAN than without using it in the data rate of the system. MLCMS with 6LoWPAN is marginally better than without 6LoWPAN, because 6LoWPAN deploys header compression, which increases the packet rate at the destination.

This is achieved by installing the 6LoWPAN structure model in each sensor node of an M2M system and testing how this enhances performance. The aim of performance evaluation for the packets received is to test the basic operations and features in the MLCMS model when using 6LoWPAN.

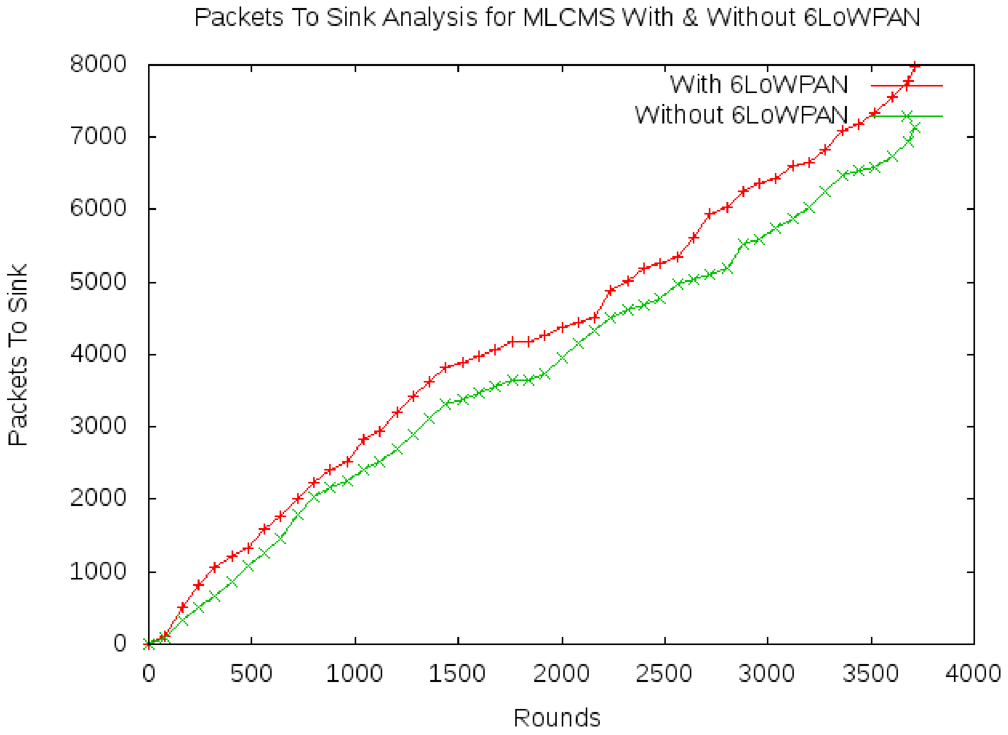


Figure 3.8 Number of packets received by the sink

3.5.2 Performance Evaluation of MLCMS Based on the 6LoWPAN Model with an Adaptive Sleep Mode

The active time period in the proposed scheme depends on the application. Comparative performance is assessed when using an adaptive sleep mode with the MLCMS with 6LoWPAN routing protocol and without using it. Clearly (see Figure 3.9 and Figure 3.10), the results obtained from using the sleep mode technique show that it enhances the lifetime of the system. Furthermore, implementing the adaptive sleep mode leads an increase in the number of alive nodes and in the residual energy. In all aspects, the proposed adaptive sleep mode increases the performance of the MLCMS when compared to without using it.

It is evident from these figures that the lifetime of the network is extended further and the network energy consumption is greatly reduced when adding the adaptive sleep mode rather than the alternative. Specifically, the extension of the lifetime for the proposed model is approximately doubled. In Figure 3.9, the coefficient of variation of active node levels is plotted with respect to the round time. It shows the active nodes in both models and reveals that the lifetime of the proposed model with adaptive sleep mode ends in round 6,100, whereas without it results in no active nodes by round 3,716.

In addition, Figure 3.10 shows the residual energy levels of the sensor nodes for the MLCMS with adaptive sleep mode and without in terms of the length of time for the network. It indicates that the sensor nodes in MLCMS with adaptive sleep are used for improving the residual energy level. As can be clearly observed, it achieves energy equalisation by using adaptive sleep mode and provides better performance regarding energy conservation. Furthermore, after 1,500 rounds, MLCMS with the adaptive sleep mode nodes has significantly higher residual energy stocks compared to the sensors in MLCMS without it. The proposed architecture can extend the lifetime of wireless sensor nodes, because the MLCMS with the adaptive sleep mode is a sophisticated algorithm that can reduce the incidence of node death during transmission and reception.

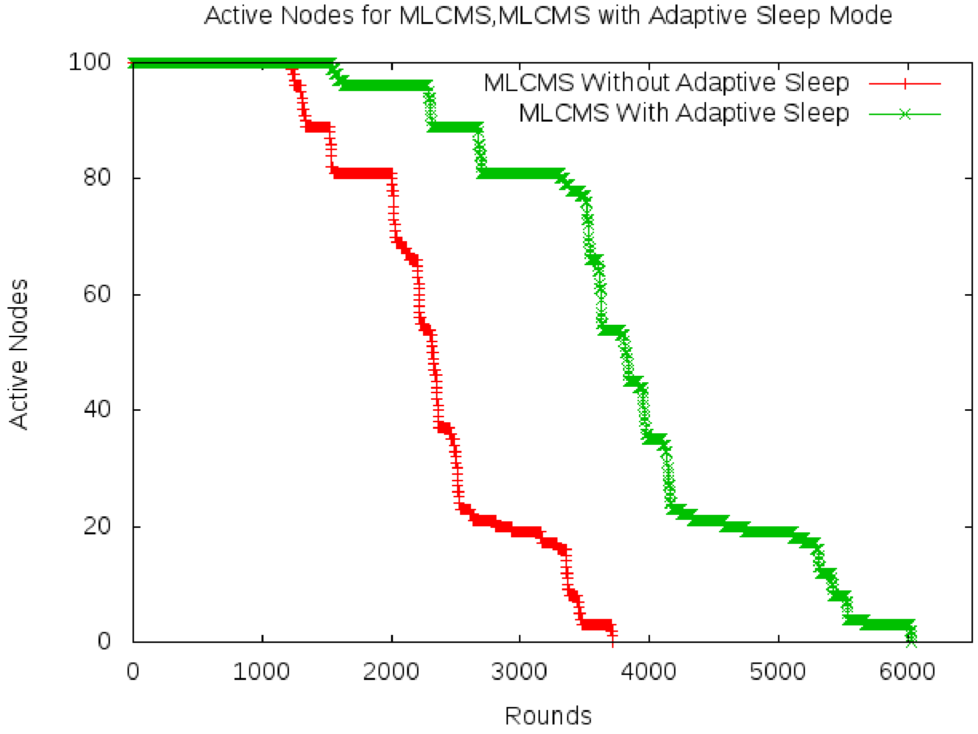


Figure 3.9 Number of active nodes

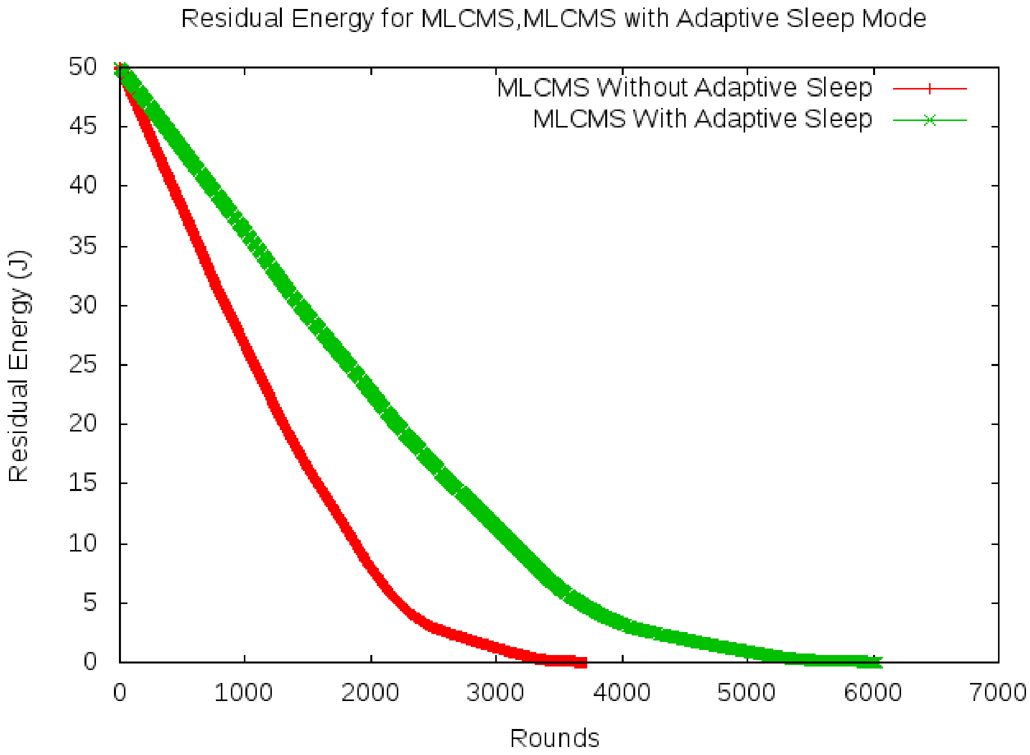


Figure 3.10 Residual energy

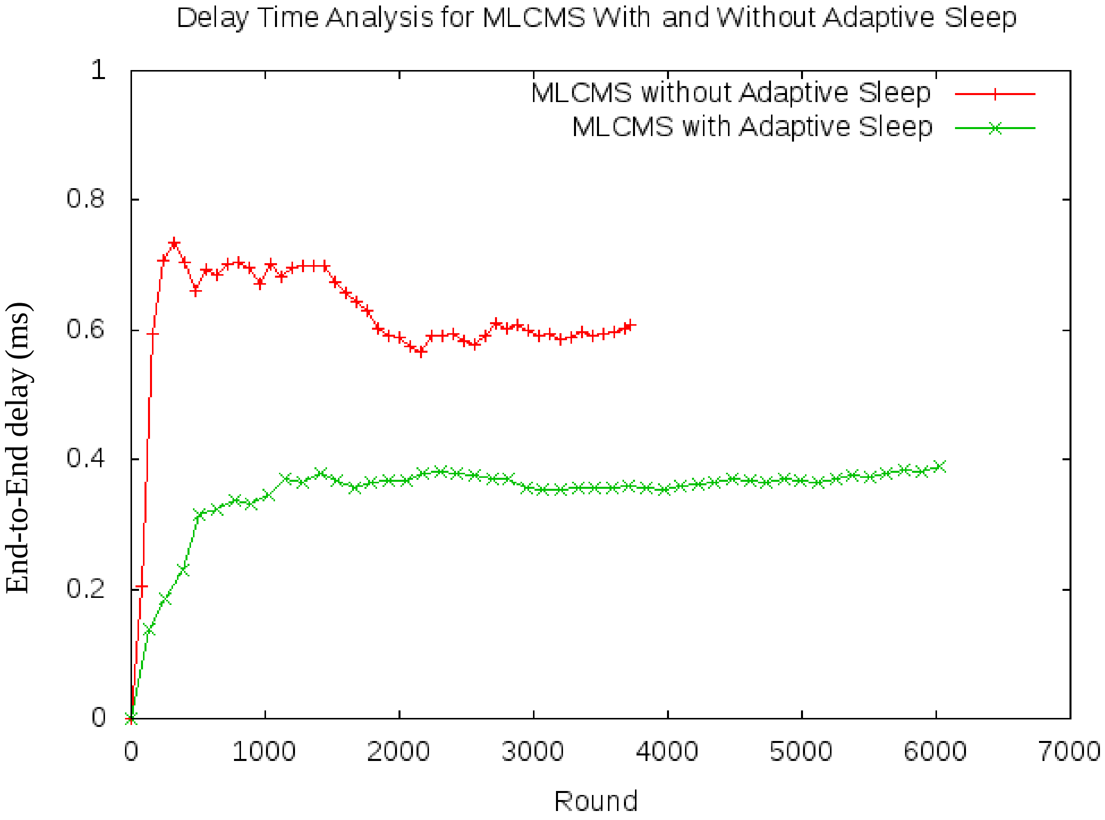


Figure 3.11 End-to-End delay

Furthermore, Figure 3.11 shows the expected end-to-end delay according to the rounds. As can be seen, the end-to-end delay decreases as the traffic load of the system is minimised, by considering the sleep mode periods for the CHs with their nodes in an adaptive way. The delay is linearly proportional to the traffic load, which means that as the traffic between the source and destination of consecutive forwarders increases, the end-to-end delay increases. This means that, the more congestion there is, the more the end-to-end delay.

3.6 Summary

In this chapter, a globally-converged M2M MLCMS model system is proposed using sensors based on the 6LoWPAN protocol. Moreover, deployment of the sink nodes with an optimal multiple number has been introduced based on a number of experiments. An energy efficient MLCMS of the M2M routing protocol has been proposed. The evaluation demonstrates that the energy consumption, when applying the MLCMS protocol is reduced compared to LEACH and M-LEACH algorithms. Simulation results show that the approach consumes less energy than M-LEACH and LEACH, increase the energy efficiency of 93% and 147%, respectively. Moreover, it not only enhances energy efficiency, but also the stability. By combining 6LoWPAN and an IP network using IPv6 addresses, the system's data rate is increased by 7%, and higher accessibility to the M2M nodes and a substantial extension of the network are realised.

Furthermore, an adaptive sleep mode for the efficient lifetime of the implemented M2M WSN with periodic data acquisition is introduced. The time period of active and asleep CHs has been considered by adding the adaptive sleep mode scheme to the MLCMS algorithm. Moreover, using MLCMS with 6LoWPAN and the proposed adaptive sleep mode in relation to the residual energy when scheduling the timing of the sleep mode, doubles the lifetime of the system and increases the reduction of end-to-end delay by half.

Chapter 4 Network Performance Evaluation of M2M[2]

4.1 Introduction

Machine-to-Machine (M2M) networks are arranged hierarchically to support an energy-efficient routing protocol for data transmission from terminal nodes to a sink node via cluster heads in a Wireless Sensor Network (WSN). Network congestion caused by heavy M2M traffic is tackled using load balancing solutions to maintain high levels of Network Performance (N.P). Enhancing the N.P based on MLCMS routing protocol (described in chapter three) is achieved through linear integer-based optimisation. A Self-Organising Cluster Head to Sink Algorithm (SOCHSA) is proposed, hosting Discrete Particle Swarm Optimisation (DPSO) [92] and Genetic Algorithm (GA) [60] as Evolutionary Algorithms (EAs) to solve the N.P optimisation problem. N.P is measured based on Key Performance Indicators (KPIs) for load fairness and average ratio residual network energy. The SOCHSA algorithm is tested by two benchmark problems with two and three sinks. DPSO and GA are compared with the Exhaustive Search (ES) algorithm to analyse their performances for each benchmark problem.

The objective of this chapter is to improve N.P by applying a load balancing technique for M2M networks with the aim to achieve a high average residual energy amongst the nodes. In order to meet the N.P. requirements of M2M, we introduce an optimal reconfiguration of the network solution that involves a traffic-aware load balancing mechanism, thereby improving energy efficiency.

The main contribution of this chapter is to present an efficient model for proper CH to sinks mapping as an approach to achieve an optimal network structure by solving a load balancing and residual energy problem, which in turn

---

[2] This Chapter is published in IEEE Sensor Journal, Aug 2017 [22].

maximise the N.P. In addition, an MLCMS protocol is presented to control CH-Sink and Node-CH transmission techniques so as to enhance network lifetime. The transmission between the CHs and sinks is dynamically adjusted by proper CH assignment to sinks via an intelligent algorithm. Enhancement of the N.P. in terms of reliability and lifetime can be achieved by the CH selecting a sink through applying an optimisation algorithm rather than the nearest sink to communicate. CH-sink allocation is achieved by formulating a linear integer-based optimisation problem with constraints. Two evolutionary algorithms, i.e. GA and DPSO, are considered for solving the optimisation problem.

A self-organised network (SON) is proposed and is formulated as a linear integer-based constrained optimisation problem, which tries to balance the load among multiple sinks. The load balancing strategy identifies optimum sink for each CH to transmit to in a multi-sink environment, while the MLCMS protocol defines the transmission strategy between nodes i.e., CH-Sink, Node-CH, and CH-CH.

The MLCMS is also responsible for CH selection satisfying a particular criterion which has a positive impact on network lifetime enhancement. The main contributions of this work are published in IEEE Journal [83] highlighted as follows:

Firstly, based on same proposed algorithm of MLCMS that explained in chapter three, which supports multiple sinks and layers, making it suitable for seamless integration in future cloud communication networks for IoT support.

Secondly, since the MLCMS protocol is responsible for CH selection and transmission strategies, the load balancer proposed in the architecture is responsible for identifying optimum CH-Sink configuration by solving an integer-based optimisation problem.

Thirdly, optimum CH-Sink re-configuration is obtained by the load balancer via an algorithm called SOCHSA. Once the load on any sink reaches an alarming threshold the load balancer triggers SOCHSA algorithm. Two important KPIs are considered to monitor N.P, based on which the optimum CH-Sink setting can be identified. These KPIs are load fairness index and average ratio residual energy.

Also**,** the proposed SOCHSA algorithm hosts two evolutionary algorithms, i.e., GA and DPSO to solve the load balancing optimisation problem. Since the standard GA and PSO cannot be applied directly to the integer-based optimisation problem in this work, a DPSO and real-valued GA are developed to provide optimum CH-Sink setting. Moreover, both GA and DPSO maximises an objective function, which is defined as N.P in this chapter and is a weighted combination of the two major KPIs discussed later in this chapter.

Finally, in order to enable every CH with the ability of flexible transmission to any sink based on the CH-Sink configuration, a sharing node with unlimited energy is employed in the architecture as shown in Figure 4.1 and Figure 4.2. The sharing nodes make the co-operative operation of MLCMS protocol and SOCHSA possible, without affecting the transmission ranges defined by the MLCMS protocol.

The remainder of the chapter is organised as follows. **Section 4.2** presents the problem statement whilst **Section 4.3** discusses the proposed work scheme. The computational results when compared different algorithms are covered in **Section 4.4**. Concluding remarks are provided in **Section 4.5**.

4.2 Problem Statement

Firstly, As M2M is an important part of IoT, this has led to much emphasis in research being placed on how to achieve fairness of the load for the whole system. The CPU workload of the sink will increase due to the processing of the data coming from the CHs sensors. Such algorithms, where the nodes are arranged as clusters in the network according to the transfer of data, are termed cluster-based methods.

The hot spot problem is the critical issue in WSNs with static sink node, because the nodes near to the sink might have a high traffic when transmitting data to the sink. Efficient sink has a positive effective on the network performance, for instance high load balancing fairness, prolong the lifetime of the system [67].

Secondly, recently it has a clear view point according experimental sensor networks and applications, which suffers from high packet loss and traffic. That leads to the critical requirement of high network performance with respect to high load balancing and energy efficiency in the network. This has a direct impact on the reliability measured at the sink node.

Thirdly, most of the congestion control approaches are effective regarding packet drop rather than the overload on the sinks, which is the main measure of fidelity at the sinks.

Finally, an essential issue in the sensor networks is the lack of an advanced design to deal with the high network traffic and congestion is maximised. This leads to a negative impact on the energy and bandwidth.

4.3 Proposed Work

4.3.1 System Architecture and the Energy Model

A. The Network Model and Cluster Generation Model

A hierarchical network in which a sensing field is logically divided into N sections, as shown in Figure 4.1 and Figure 4.2, depending on the number sinks to be used. For example, for two sinks the sensing field is divided into four sections, as shown in Figure 4.1, while when using three sinks the sensing field is divided into six sections, as shown in Figure 4.2, with three levels of CHs in each section. Each CH has a related sink, a CH and a number of normal/ordinary sensor nodes. The sensing field size is assumed to be NxN. The proposed network model contains the following elements.

- **The nodes:** Distributed randomly in the sensing field such that different nodes are selected as CHs (depending on certain requirements) or can remain as normal nodes.

- **The centralised server:** It is responsible for collecting the network information (e.g. network topology, number of CHs, Sink-CH configuration, residual energy and number of active nodes) and to identify optimum CH- Sink configuration so as to maximise N.P. The centralised server hosts an intelligent algorithm to identify optimum CH-Sink configuration, which is triggered when the load on the sinks reaches an alarming threshold. The optimum CH-Sink setting is then realised via server commands to the sink nodes.

- **The sharing sensor nodes:** The sharing unlimited sensor nodes are used to transmit data between the sections of the sensing field so as to avoid exceeding the Transmission Range (T.R) boundaries.

- **The transmission ranges of the sensing field:** Each section is divided into three levels horizontally. At each level, there is a maximum T.R, which is the diagonal length of the level, as shown in Figure 4.1 and depends on the probability of farthest node in each level. Nodes that are beyond this are unable to connect with the CHs, whereas those within the range transmit their parameters to the CH for new CH election or to the centralised node of each level initially. Using sharing sensor nodes between sections helps to provide a flexible network reconfiguring. Moreover, the T.Rs between a CH and the closest sharing node is considered equal to that of CHs and CH in the sensing field, as shown in Figure 4.2. The T.Rs are formulated same as in Equations (3.1), (3.2) and (3.3) in MLCMS in chapter three. Moreover, the T.R of (CH-to-CH) and (CH-to-Sharing sensor node) are same.

The steps and the parameters of selection cluster heads are same as in chapter three in section 3.3.2 and is shown in Figure 4.3.

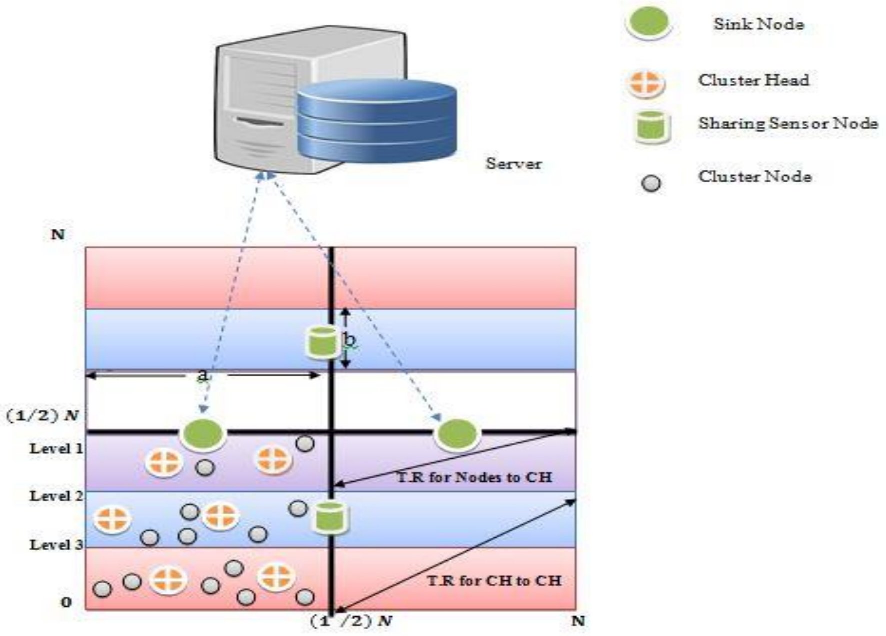


Figure 4.1 System model of two sinks

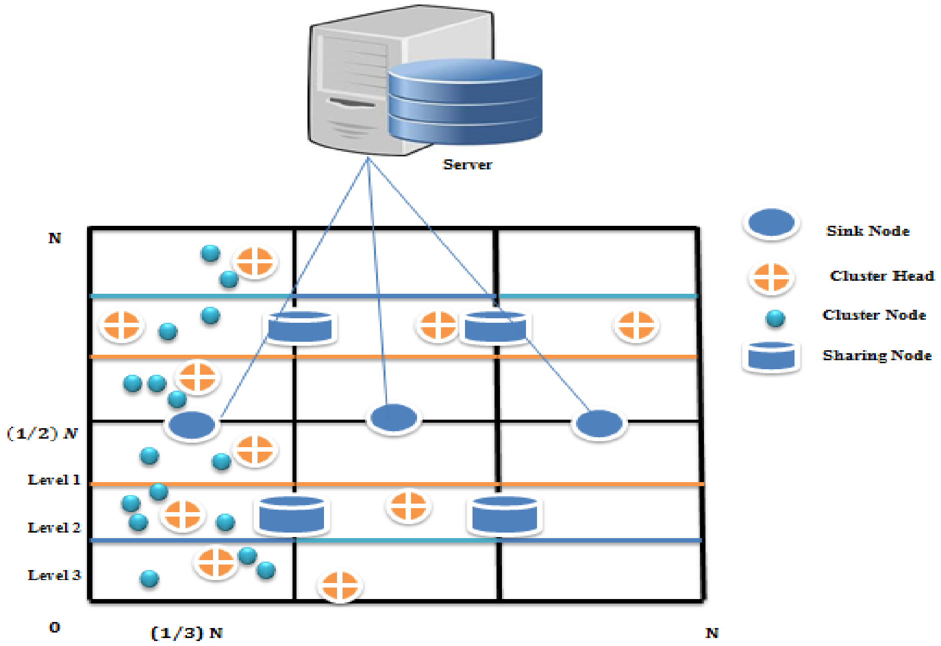


Figure 4.2 System model of two sinks

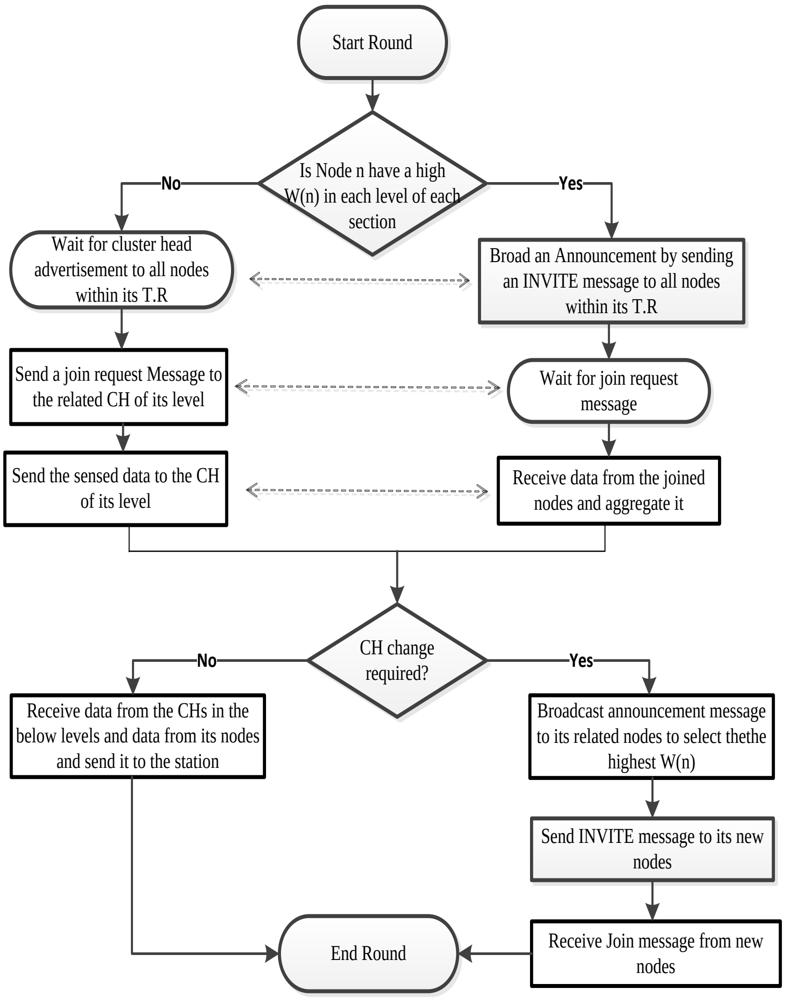


Figure 4.3 CH selection procedure flow diagram

B. Energy Consumption Model

The energy model used is based on the same structure and formula that described in section 3.3.3 of power model in chapter three. The energy consumption model described in [15] was designed to calculate the energy consumed during transmission and reception of data.

Table 4.1 Simulation Parameters and Values

Parameter Name	Value
Number of sensor nodes (n)	1000
Base station locations for two sinks	(25,50),(75,50)
Base station locations for three sinks	(50,100),(100,100),(150,100)
Length of the packet (L)	4,000 bits
Initial energy of the sensor nodes ($\mathrm{E_{in}}$)	0.5 J
Energy consumption in the circuit ($\mathrm{E_{elect}}$)	50 nJ/bit
Free space model of transmitter amplifier ($\mathrm{E_{fs}}$)	10 pJ/bit/m^2
Multi-path model of transmitter amplifier ($\mathrm{E_{mp}}$)	0.0013 pJ/bit/m^4
Network Size (N*N) for two sinks	100 m ×100 m
Network Size (N*N) for three sinks	200 m× 200 m

4.3.2 Dynamic Network Reconfiguration

Network performance can be measured by utilising multiple KPIs. These KPIs help the network to react proactively against events that threaten network performance and services availability. A different set of objectives can be employed to evaluate N.P, which can then be mapped to predefined performance metrics. A (SON) algorithm is presented in section 4.3.3, which is performed at the load balancing medium (also referred to as server Unit). In the SON algorithms, the expected N.P, which consists of network objectives in the form of KPIs, is used to perform optimum CH-Sink configuration. Note that, a weighted normalised function is required when considering different sets of objectives. This chapter presents KPIs that define N.P. The medium load balancer (centralised server) is responsible for manipulating the route of the request nodes and allowing the system to scale so as to service more requests by just adding nodes. When the CPU workload attains a certain threshold level, an alarm is sent by the aggregator to the server, which triggers the load balance algorithm.

The CHs in the network at a particular time are re-configured optimally for sink transmission, based on the KPIs' information to achieve a balanced load and high residual energy in the network. If the CH-Sink configuration is known at time t, then finding the optimum CH-Sink configuration at time t+1 is the main challenge. Let H be the number of CHs in the network and M be the number of sinks. Then, the CH-Sink association at time t can be represented by a vector$\mathrm{C}^{\boldsymbol{t}} = \{\mathrm{C}_1^t, \mathrm{C}_2^t, \dots, \mathrm{C}_h^t\}$, where

$\mathrm{C}_h^t \in \{1,2,\dots,H\}$. $\mathrm{C}_h^t = m$ shows that CH_h is transmitting to Sink_m. Then, finding the optimum CH-Sink configuration vector at time t+1 i.e., $\mathbf{C^{t+1}} = \{\mathrm{C}_1^{t+1}, \mathrm{C}_2^{t+1}, \dots, \mathrm{C}_h^{t+1}\}$ is the main objective. The following KPIs are considered for the CH-Sink association problem.

A. *KPI for Average Ratio Residual Energy (KPI_{ARE})*

The KPI for the average ratio residual energy of the network is given as:

$$\mathrm{KPI}_{ARE} = \frac{1}{\mathrm{N}_n}\sum_{i=1}^{N_n} \mathrm{E}_i\,(\mathrm{t}) \tag{4.1}$$

where, $\mathrm{E}_i(\mathrm{t})$ is the ratio residual energy of Node_i at time t and N_n represents the total number of active sensor nodes in the network. Furthermore, the ratio residual energy $\mathrm{E}_i(\mathrm{t})$ is defined as the ratio of remaining energy $\mathrm{R}_i(\mathrm{t})$ of Node_i (at time t) to the initial energy $\hat{\mathrm{E}}$ of each node, i.e. $\mathrm{E}_i(\mathrm{t}) = \mathrm{R}_i(\mathrm{t})/\hat{\mathrm{E}}$ [85]. The initial energy $\hat{\mathrm{E}}$ is same for all nodes in the network.

B. *KPI for the load fairness index (KPI_{LFI})*

In order to monitor the level of load balancing in the network, this chapter considers Jain’s fairness index it means be sure that all data received by sink in fairness manner, which at a particular time, t, can be defined as:

$$\mathrm{KPI}_{LFI} = \frac{1}{M} * \frac{(\sum_{i=1}^{M} \eta_i(\mathrm{t}))^2}{\sum_{i=1}^{M}(\eta_i(\mathrm{t}))^2} \tag{4.2}$$

where, $\eta_i(\mathrm{t})$ defines the load on Sink_i at time t. The load $\eta_i(\mathrm{t})$ on Sink_i is measured by taking the ratio of the number of packets received at Sink_i (ϕ) from all its associated CHs at time *t*, to the maximum packet handling capability (ϕ_{max}) of Sink_i, i.e.

$$\eta_i(\mathrm{t}) = \frac{\sum_{h=1}^{H} \mathrm{I}_{h,i}\,\phi_h(\mathrm{t})}{\phi_{max}} \tag{4.3}$$

where, $\mathrm{I}_{h,i}$ is a binary indicator, i.e. $\mathrm{I}_{h,i} = 1$, if CH_h transmits packets (ϕ_h) to Sink_i at time *t* otherwise $\mathrm{I}_{h,i} = 0$

To maximise the performance of the network, the weighted normalised sum of the above mentioned KPIs is taken in order to define a new objective function, which is given as:

$$\text{Max N. P (t)} = \alpha \text{KPI}_{ARE} + \beta \text{KPI}_{LFI} \tag{4.4}$$

where, N.P is the network performance fitness function and α and β are the weights assigned to each KPI, which represent the priority level of each KPI in the objective function. In this chapter, the values for α and β are 0.8 and 0.2, respectively, based on the condition that satisfies $\alpha+\beta=1$ (explained in the later section). A higher priority is given to the average ratio residual energy since enhancing the lifetime of the network remains the first priority during the solving of the CH-Sink configuration problem. Maximising the N.P function is the main objective.

In order to find the optimal CH-Sink configuration an exhaustive search for all possible CH-Sink combinations is required. This makes the search space size equal to M^{H}, where M is the number of sinks and H is the number of active CHs in the network at a particular time t. Since number of possible CH-Sink configurations exponentially increases with the number of CHs and sinks in the network, the algorithm execution time also increases exponentially. Therefore, evolutionary algorithms are proposed in the following section to resolve the CH-Sink configuration as an optimisation problem.

4.3.3 Self-Optimised Cluster Head to Sink Association (SOCHSA) Algorithm

In this section, a centralised SOCHSA algorithm is proposed, which depends on the above intuitive analysis to execute proper CH-Sink configuration. Figure 4.4 presents a block diagram of the SOCHSA algorithm, where network information is utilised to analyse the KPIs and measure the overall network performance of the current CH-Sink configuration in the initial step. Network information involves the load on each sink, number of cluster heads, packets transmitted by each cluster head at a particular time and the residual energy of the nodes in the network. For optimisation, the same network information is utilised for KPI and network performance analysis of other candidate CH-Sink configurations. The SOCHSA algorithm adjusts the CH-Sink configuration after the optimisation step by comparing the network performance of current and optimised CH-Sink configurations. The KPIs defined are maximised by

SOCHSA algorithm to maximise the N.P by utilising evolutionary algorithms. Two evolutionary algorithms, i.e., Genetic Algorithm (GA) and Discrete Particle Swarm Optimisation (DPSO) [92] [60] are examined in this chapter to solve the CH-Sink configuration as an optimisation problem. The GA and DPSO search algorithms are based on population, where population means a group of possible solutions (i.e. CH-Sink configuration solutions) for an optimisation problem. In GA, the population is referred to as a group of chromosomes, since it is a bio-inspired search algorithm, where the genes of the fittest available chromosomes are utilised to make new ones (new CH-Sink configuration solutions) via processes called crossover and mutation.

However, in standard PSO the population of possible solutions is referred to as a swarm of particles. PSO is inspired from the concept of birds flocking, whereby a swarm of birds in search of food probe the search space with different velocities [61]. The velocity of each particle is then changed stochastically (to generate new particles). Thus, regarding the algorithms, each individual particle in PSO and each individual chromosome in GA is considered as a candidate solution for the CH-Sink configuration problem.

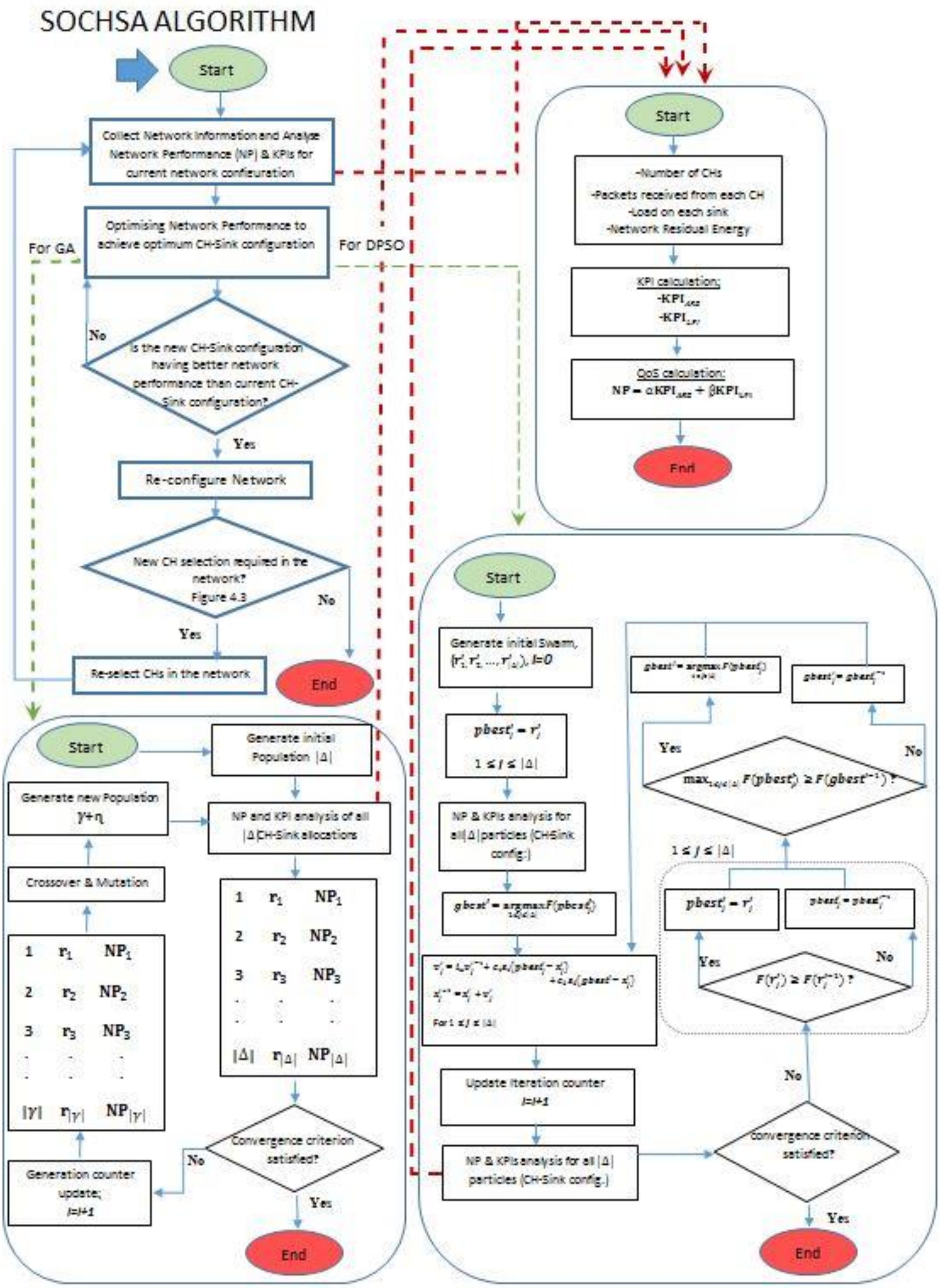


Figure 4.4 SOCHSA algorithm block diagram

In PSO the velocity is updated for each particle depending on the historic best position experienced (pbest) by the particle itself and the best position experienced by the neighbouring particles, i.e. the global best position (gbest). Standard PSO cannot be applied to the discrete CH-Sink configuration optimisation problem. Therefore, for a real valued CH-Sink configuration, a Discrete PSO is proposed in this chapter to solve the network performance maximisation problem defined in Equation (4.4). Both GA and DPSO, as presented in Figure 4.4, are explained by the following steps:

Step 1: Initiate random population R^0 with a population size of |Δ|. Each candidate solution in the population has B components, where the number of components is taken according to the number of CHs in the network at a particular time t. For DPSO, initiate the best position for each particle, i.e. $pbest_j^0 = r_j^0$, where $1 \leq j \leq |\Delta|$ with a random velocity of v_j^I for each particle.

Step 2: Determine the fitness value of each chromosome in GA and each particle in DPSO in the initial population using the objective function F (i.e. the N.P function defined in Equation (4.4)). For DPSO, identify the global best position in the swarm/population, i.e. $gbest^0 == \underset{1\leq j\leq|\Delta|}{\text{argmax}} F(pbest_j^I)$.

Step 3: In the case of GA, if the convergence criterion is satisfied, i.e. if the best candidate solution is achieved or the maximum number of generations has passed, then end, or else go to Step 4. However, in the case of DPSO, update the velocity of all the particles in the current population. The velocity update equation is given as:

$$v_j^I = i_w v_j^{I-1} + c_1\varepsilon_1\left(pbest_j^I - x_j^I\right) + c_2\varepsilon_2\left(gbest_j^I - x_j^I\right) \qquad (4.5)$$

$$1 \leq j \leq |\Delta|$$

where, x_j^I shows particle j's current position at iteration number I. ε_1 and ε_2 are random numbers chosen within the 0-1 range, whereas c_1 and c_2 are considered to be the acceleration constants that are required to pull the particles towards the best position. The values chosen for c_1 and c_2 lie within the range 0-5. i_w shows the effect of inertia of the preceding particle velocity over the updated particle velocity. The value of i_w is altered to achieve global exploration or to expedite local search. Optimum value selection for i_w can assist in both global and local exploration of the search space. Typical values for i_w are selected in the range 0-1 [61][63] and in this chapter the value selected is 0.9 [93] . Moreover, the updated positions of all particles for the next iteration (I+1) are given as:

$$x_j^{I+1} = \left(x_j^I + v_j^I\right) \qquad 1 \le j \le |\Delta| \tag{4.6}$$

Step 4: In the case of GA, extract a set of $|\gamma|$ best chromosomes from current population, i.e. R^I with a chromosome selection probability of P_s. In the case of DPSO, update the iteration number, i.e. I=I+1.

Step 5: For GA, perform crossover and mutation operations on set γ. The unfeasible solutions ($R^I - \gamma$) are replaced with newly generated chromosomes μ (i.e. $\gamma + \mu$). For DPSO, if the convergence criterion is satisfied, then end, or else go to Step 6.

Step 6: In the case of GA, repeat all steps from Step 2, whereas, for DPSO, update each particle's personal best position, i.e.

$$\mathrm{pbest}_j^I = \begin{cases} \mathrm{pbest}_j^{I-1} & \text{if } F(r_j^I) \le F(\mathrm{pbest}_j^{I-1}) \\ r_j^I & \text{if } F(r_j^I) > F(\mathrm{pbest}_j^{I-1}) \end{cases} \tag{4.7}$$

Step 7*:* For DPSO, Update the global best position achieved.

$$\mathrm{gbest}_j^I = \begin{cases} \underset{1 \le j \le |\Delta|}{\mathrm{argmax}} F(\mathrm{pbest}_j^I) & \text{if } F(\mathrm{pbest}_j^I) > F(\mathrm{pbest}_j^{I-1}) \\ \mathrm{gbest}_j^{I-1} & \text{otherise} \end{cases} \tag{4.8}$$

Step 8: For DPSO, repeat all steps from Step 1.

4.4 Computational Results and Complexity

Two benchmark problems P1 and P2 are considered in this chapter, in order to analyse, verify and demonstrate the performance of the SOCHSA algorithm. Both benchmark problems consist of 1,000 sensor nodes randomly distributed in the sensing field. In this work we considered DPSO and GA, which have been popular in academic and industry for solving complex optimization problems. These facts are based on many research’s comparisons in [63] [94].

Table 4.2 GA and DPSO Computational Results

	Network Performance value	Convergence Rate		Number of iterations		CPU time (sec)	
	GA/DPSO	GA	DPSO	GA	DPSO	GA	DPSO
2 sinks	0.306287	0.895	0.935	21	13	0.10	0.09
3 Sinks	0.307731	0.86	0.9	28	20	0.48	0.32

Both P1 and P2 are tested at a particular time instance, where 270 CHs are selected in the network. The only difference between the two benchmark problems is that P1 has two sinks, whereas P2 has three. The selection of a different number of sinks in the two problems helps to differentiate between the lifetimes of the network in terms of average ratio residual energy. An exhaustive search for the optimum CH-Sink configuration is made in order to compare with the results of SOCHSA algorithm. The Exhaustive Search (ES) algorithm is used to find the optimum CH-Sink configuration. The α and β parameters for Equation (4.4) are chosen according to a constraint such that $\alpha + \beta \leq 1$. Selecting proper weights for the optimisation function itself is challenging and necessary for proper tuning of evolutionary algorithms. Therefore, an ES for optimum CH-Sink configuration is performed on P1 with different α and β settings and the overall packet drop is analysed, as shown in Figure 4.5. The α and β parameters are changed orderly i.e., 0, 0.1,…, 1 subject to the constraint $\alpha + \beta \leq 1$. The much lower packet drop is observed when the α=0.8 and β=0.2, giving a higher priority to average ratio residual energy over load fairness index maximization as shown in Figure 4.5.

ES performs M^{H} possible CH-Sink configuration solutions for both P1 and P2, where M is the number of sinks and H is the number of CHs in the network, i.e. 2^{270} and 3^{270} for P1 and P2, respectively. Note that the initial CH-Sink configuration for both P1 and P2 allows the CHs to transmit to the sinks just above or below their current position, i.e. each sink handles the area above and below it, as the coverage area is divided into four quarters and six sections for P1 and P2, respectively. Figures 4.6-4.8 show the network performance (fitness function), the average ratio residual energy, and the network load fairness index over 200 generations or iterations for both P1 and P2. The optimum values shown in the figure are achieved by the ES algorithm. Note that, ES does not depend on generations or iterations. The optimum values are used to demonstrate the improvement achieved by both GA and DPSO at each generation or iteration.

The optimum results achieved from ES are presented in Table 4.2, where the convergence rate for both GA and DPSO is also given. This is defined as the number of times an optimum solution (or the optimum CH-Sink configuration solution) is achieved over the entire number of generations or iterations.

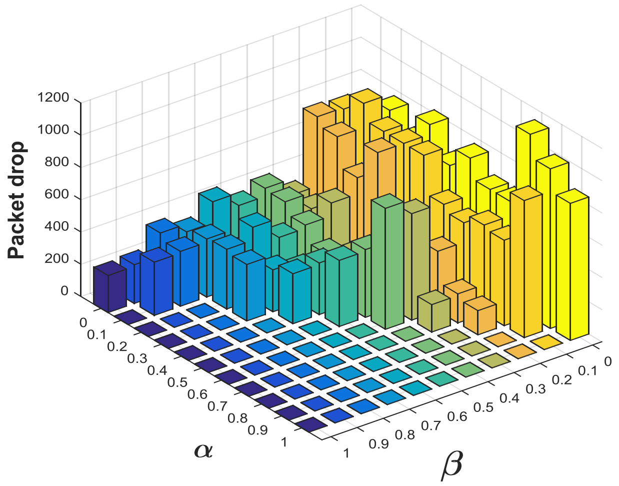


Figure 4.5 Packets drop with different α and β

In P1 for two sinks, GA converges to the optimum CH-Sink configuration solution after the 21st generation, with a network performance evaluation value of 0.30628 (which is the optimum), whereas DPSO achieves the optimum network performance value after the 13th iteration. 179 optimum CH-Sink configuration solutions are achieved by GA over 200 generations with a convergence rate of 0.895, whilst the optimum CH-Sink configuration solutions by DPSO are 187 over 200 iterations with a convergence rate of 0.935. The DPSO converges faster to the optimum CH-Sink configuration solution compared to GA, as shown in Figure 4.6 and Table 4.2. The optimum network performance value is achieved after 21x500 (i.e., 21x|Δ|) and 13x500 (i.e. 13x|Δ|) fitness evaluations by GA and DPSO, respectively. However, the ES has to evaluate 2^{270} possible CH-Sink configuration solutions to find the optimum network performance value, which is too enormous.

For P2 for three sinks, the optimum network performance value achieved by GA and DPSO is after 28th and 20th generations or iterations, respectively. The convergence rates for GA and DPSO are 0.86 and 0.9, respectively, with 172 optimum CH-Sink configurations solutions achieved over 200 generations for GA and 180 for DPSO. Figure 4.7 and Figure 4.8 show that the average ratio residual energy and the network load fairness index (for both benchmark problems) increase as the GA and DPSO converge to the optimum CH-Sink configuration solution. The average ratio residual energy is improved by 2% when 3 sinks are used compared to 2 sinks. This shows that the use of multiple sinks with the proposed MLCMS protocol for large networks can enhance network lifetime. Since the DPSO outperforms GA. in both the benchmark problems, it is best suitable for the proposed SOCHSA.

This is because both algorithms operate differently. In DPSO, each particle acts as an agent, which updates its velocity in the search space based on social information (i.e. best particle position in the current population or swarm) and global information (i.e. best particle position for all iterations). However, the chromosomes in GA do not act like agents and have no information of other chromosomes in a population. In fact, GA relies on crossover and mutation operations, which could disturb the convergence of the algorithm. Hence, DPSO is suitable for global exploration of the search space.

This study further demonstrates the performance of MLCMS with and without SOCHSA over 100 nodes as shown in Figure 4.9. It is evident from the figures that the lifetime of the network is extended further and the network energy consumption is greatly reduced when the proposed algorithm (SOCHSA) works in tandem with MLCMS protocol.

In Figure 4.9 coefficient of residual energy for MLCMS with and without SOCHSA is plotted with respect to round time. MLCMS with SOCHSA shows minimal variation in energy levels, while without SOCHSA, large fluctuations are observed. This indicates that the sensor nodes residual energy level in MLCMS with SOCHSA is significantly improved.

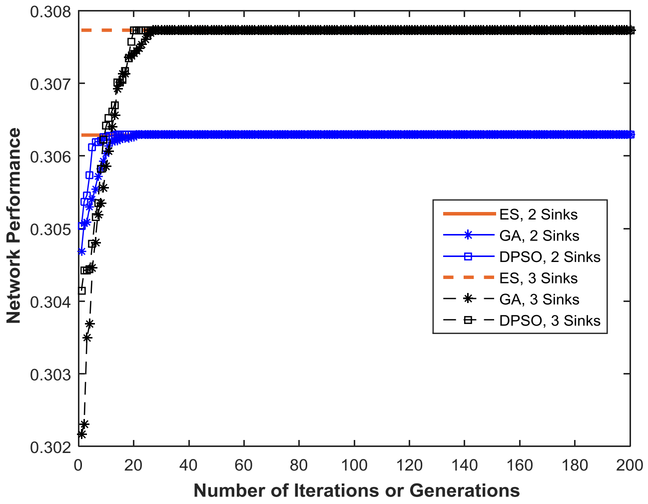


Figure 4.6 Network performance measurements for GA and DPSO in benchmark problems P1 and P2

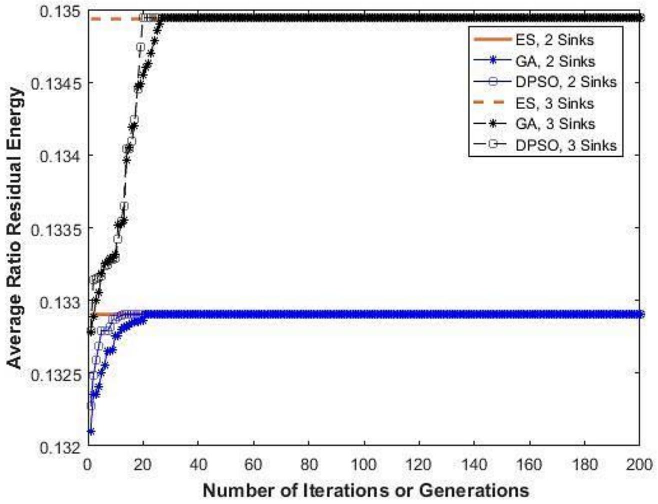


Figure 4.7 Average ratio residual energy for GA and DPSO in benchmark problems P1 and P2

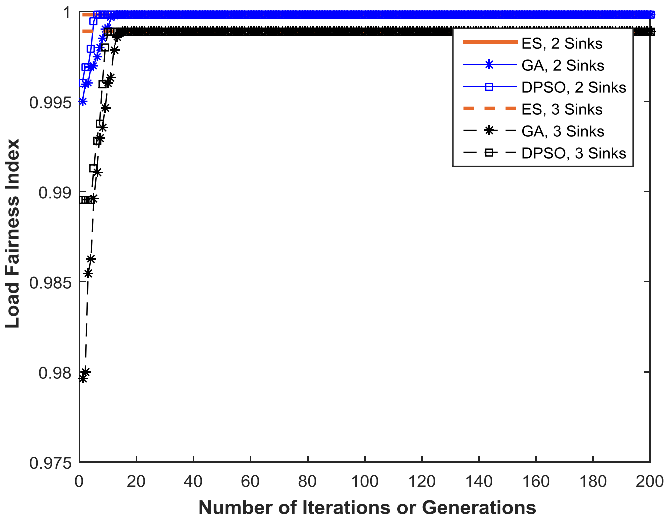


Figure 4.8 Load fairness index for GA and DPSO in benchmark problems P_1 &P_2

This reveals that the lifetime of MLCMS with SOCHSA ends in 4,500 rounds, and after 3,716 rounds without SOCHSA. The proposed architecture can extend the lifetime of wireless sensor nodes, because the MLCMS with SOCHSA provides a sophisticated algorithm aimed to improve the N.P by maximising the load balancing fairness as well as the overall residual energy.

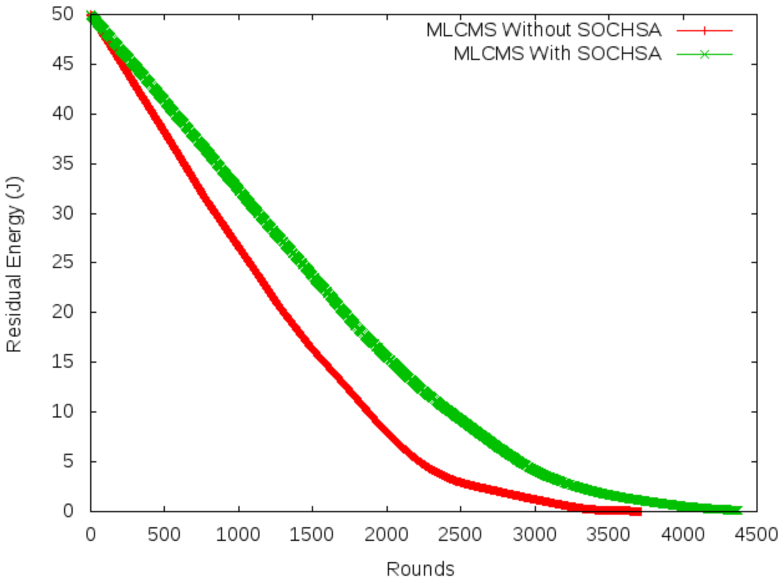


Figure 4.9 Residual energy for MLCMS with SOCHSA and without SOCHSA

4.5 Summary

In this chapter, an MLCMS protocol to enhance network lifetime along with load balancing solutions is presented with the aim of improving the N.P of M2M wireless sensor networks. A dynamic CH to sink allocation technique has been investigated. Practical CH-Sink mapping solutions that deliver balanced traffic in the network with high ratio residual energy have been probed. To this end, a self-optimising M2M-WSN algorithm that efficiently deploys network resources has been proposed. The CH-Sink allocation arrangement maximises N.P. by balancing the load across multiple sinks to avoid over-utilisation of the network. Maximising the load fairness index and the average ratio residual energy for the whole M2M network is solved as an optimisation problem. The CH distribution problem is tackled by obtaining an optimal solution with two evolutionary algorithms, namely, GA and DPSO, hosted by the SOCHSA algorithm. The performances of both GA and DPSO have been tested and compared using two benchmark scenarios. The MLCMS with SOCHSA algorithm provides significant network benefits regarding network lifetime when compared to MLCMS without SOCHSA. DPSO converged notably quicker than GA and also outperformed it in large network scenarios. It is observed that both GA and DPSO achieve optimum N.P evaluation values 0.306287 and 0.307731 were achieved in the benchmark problems P1 and P2, respectively by both two and three sinks for GA and DPSO. It is found the average ratio residual energy is improved by 2% when using three sinks under the same simulation settings, compared to a two sinks setting. Based on computational results, the DPSO is observed to outperform GA in terms of complexity and convergence, thus being considered best suited for a proactive IoT network. The proposed mechanism can satisfy different N.P requirements of M2M traffic by instant traffic identification and dynamic traffic rerouting.

Chapter 5

Profiling Power Model for SDN-WISE[3]

5.1 Introduction

Over recent years, real elements collaborating with each other define the Internet of Things (IoT). As the development of the Internet continues in large strides, it is forecasted that ultimately billions of appliances will be linked using this technology. Countless applications, such as the employment of IoT for environmental condition monitoring, will rise in popularity. Due to the quick high-tech expansion of sensors, WSNs will become a crucial technology in the IoT industry [29].

Recently, SDN solutions for WIreless SEnsor networks such as SDN-WISE have merged to allow the complexity of massive networks to be reduced. This makes SDN-WISE a suitable approach for a huge number of applications and a solution for the previous issue [19]. However, little is known about the Power Consumption (PC) of SDN-WISE, especially when enabling it to control signalling.

Modelling the usage and integration of the SDN with WSNs has been studied in different research studies; they have considered various categories of WSNs and their essential differences in comparison with SDN in wired network. This integration of WSN and SDN enables ease usage of real time network resources monitoring, with reconfigurable routing path as an adaptive routing. All these fashion the network elements communication in a more stable and efficient manner, which is achieved by having a centralised controller [76] [26]. Thus, it provides a dynamic support to maintain Network Quality of Service (NQoS), which positively affects the power efficiency.

The requirements for utilising the SDN architecture in WSNs have already been analysed in [16]. Since, the WSN is characterised to have low capability in terms of

---

[3] This chapter is submitted to IEEE Internet of Things Journal, in the form of an article.

energy, processing, and memory, compared to wired network. However, the SDN-WISE can support an efficient use of network resources, specifically in sensors, but this efficiency can impact negatively on the data rate.

The main efforts of this work is investigating and modelling the impact of the integration between SDN and WSN, among control plane signalling streams and data plane traffic streams in wireless SDN system. From PC point of view, the effects of the offered Network Quality of Service (NQoS) such as achieved physical layer throughput, as well as Application Quality of Service (AQoS) such as IoT application data stream payload packet size need to be investigated. This could be done by profiling a PC model for wireless SDN platform. This is essential and required to evaluate the overall cost and figure out any extra demand caused by extra control signalling. Having a clear view of these contributed factors when modelling a PC model is an essential part of the IoT system architecture. The proposed Power Model (PM) provides a clear perspective about additional control signalling which can support SDN in WSN as SDN-WISE scheme, and the effects of this on the system model. This shapes the objectives of this study as follows:

Firstly, decomposing SDN-WISE PC components as a data plane traffic PC and control plane signalling PC with their triggered interaction effort in order to tackle with its complexity in a simplified manner.

Secondly, profiling of the SDN PC in order to assess the efficient use of network resources, when WSN is integrated with SDN, i.e., firstly from data plane traffic point of view and secondly from control plane signalling point of view. This work will allow the network administrators and end users to consider the PC effectiveness in both planes. Furthermore, the proposed model can provide one of the important building blocks to introduce an optimised energy-saving approach in IoT application based SDN.

Thirdly, it estimates the interaction among the control plane signalling header with respect to Transmission Time Interval (TTI) and data plane traffic related to data payload packet size as well as achieving a data rate to meet the IoT application’s QoS requirements.

Moreover, in this chapter the discussion is supported through a case study of power demands, based on an infrastructure that can be used for evaluating the SDN-WISE and its control signalling for creating a generic power model. This investigation takes into consideration PC, performance and the effects of extra control signalling on the total energy consumption for the whole system.

The rest of the chapter is organized as follows. **Section 5.2** presents the problem statement whilst **Section 5.3** provides an overview of SDN-WISE, SDN-WISE Supportive and the SDN-WISE layers. The power model of the proposed system with the SDN-WISE is presented in **Section 5.4**. The results and discussion are provided in **Section 5.5**. The chapter's conclusion is presented in **Section 5.6.**

5.2 Problem Statement

A huge amount of data for a distributed setting is created with large number of entities in dynamic IoT environment. This vast amount of information leads to following main challenges in the administration and utilisation of data [21],

- The requirements for modern WSN usage need to change the approach, while keeping in mind the end goal to meet these prerequisites. Furthermore, it is necessary to take optimal decisions in order to control the vast amount of data from the sensor nodes in WSN [16].
- Hence, the level of intelligence at the terminal nodes plays an important role in the control paradigm, as well as in implementing policy on sensor nodes in a sophisticated manner. However, node doesn't have intelligent level for making decision, is considered as stateless node.
- Most of the solutions for SDN in WSN focus on the architecture programmable field of WSN. As well as, lacking the mathematical investigation regarding the extra signalling in SDN based on WSN.
- The complexity of managing the WSN is the main issue.
- The importance of the power efficient system in WSN based on IoT and all issues in WSN are related to how to minimise the energy consumption.

5.3 SDN-WISE Overview

In this section we provide an overview of the SDN-WISE solution. More specifically, we will first briefly outline the supportive solution provided by the SDN-WISE design. Then will provide an overview of the SDN-WISE technical approach and the layers to justify the benefit of using SDN-WISE.

5.3.1 SDN-WISE Supportive

The essential solutions provided by SDN in WSN will be characterised according to different applications. Therefore, there is not a signal solution for WSN problems. So in an attempt to find a solution for WSN problems, many research studies have been conducted. One of these solutions is the SDN-WISE, which is considered an efficient way of using the sensors' resources. The benefits of using SDN-WISE can be summarised as follows:

- The main objective of using SDN architecture in WSN is minimising the node sate information exchange among the Sensor Nodes (SNs), the proxy controller and the operation sensor node programmable mechanism. Finite State Machines (FSM) enable them to run operations that cannot be supported by stateless solutions [27]. The main concept behind the SDN-WISE is that it gives permission for each node to make decisions regarding incoming packets by taking into account the value of the recent state (stateful) [27].
- It provides flexibility for managing packet forwarding rules that fit WSN current topology. For instance, the report messages are used for updating the controller regarding the link state between nodes as well as the battery level information. The essential purpose of SDN-WISE is introducing simple network management by enhancing a novel application and investigating solutions for WSNs as a new type of networking [95].
- Duty cycles – switching off the radio is the most important technique for energy consumption reduction [26].
- Data aggregation technique is an initial part in SDN-WISE network. Removing the redundancy from the data in the network can reduce the energy that is used by adding a data aggregation technique. Consequently, that is made more possible by

minimising the number of transmission packets in the network, as this has a positive effect on the power efficiency [96].

- Data centric service is provided to make credentials for sensor nodes, particularly concerning their data as an alternative for their address. SDN-WISE has invented this solution to specify the rules from the controller for packets with changed features [26][27].

5.3.2 SDN-WISE Protocol Architecture

This section describes the details of the major features of the SDN-WISE protocols stack. It explains all the architecture of the layers and the responsibility of each one as shown in Figure 5.1. Additionally, it will describe the SDN-WISE layers and their packets in details.

A. Topology Discovery (TD)

Topology Discovery (TD) describes as a protocol layer, which is based on processing and exchanging of packets named TD packets, which are Beacon and Report packets as well as Configuration packet, received by controller. The beacon packet contains information about node's battery level, the Received Signal Strength Indication (RSSI) and the number of hops that the node has developed to the nearest sink (gateway).

The main responsibility of this protocol is generating information about the topology from the node and sending it to the WISE-VISOR, these information are transmitted by Report packet to the WISE-VISOR [27]. This information is collected by broadcasting the Beacon periodically according to the TTI. The reason behind the broadcasting is to keep a clear view and update map of the network elements by sending Report packet to the controller. At the same time, the Beacon is received by the sensor node to clear its list of neighbours; due to the fact the SN will receive a fully updated list later from the controller by Configuration packet based TTI. Whenever the sensor node receives a packet, it will compare it with the previous information that has acquired then updates the old information, particularly with regards to the number of hops to the sink (SNK). This also shows how the sensors (SNs) populate their neighbours' list, which contains the neighbouring nodes in the network. Moreover, the operation of each node undergoes is as follows: [26] [97]

1. The SN will send the Beacon to its neighbours that are included in its list in order to ask about their battery level and RSSI. The list of neighbours in the SN is built by inserting the SN that sent the Beacon signalling. If the neighbour already exists then it will only update the information about the battery level and RSSI.

2. The SN will check the distance as a number of hops between the SN and the SNK that recently received a Beacon from it. If it has a number of hops different than the current, then it will update to the new value and consider it.

3. The Beacon packet is retransmitted in order to update the SN state. The TD layer is the only layer that has the capability to access all the other layers. This provides the TD layer with the possibility of collecting the information and assessing the SN at each layer and how it behaves.

4. Transmitting the information by using a Report packet to the WISE-VISOR.

5. The WISE-VISOR will receive the Report packets and update the network map by sending a Configuration packet the network map.

B. Topology Management (TM)

The TM is a protocol layer and an essential part of the WISE-VISOR controller in the control plane. It supports the ability of different controllers with different policy using same physical network elements in the same system[27]. Moreover, it produces an up to date map of all network's elements after receiving the Report packets and then sending the updated information to the network elements by the Configuration packets. This layer is paired with TD protocol to collect and update the information of the network elements and configure network status by building the flow table.

C. Forwarding (FWD)

This has the responsibility of executing all of the arrival packets that are specified by the WISE-VISOR. It is located on the top of the sensor nodes's MAC layer. This layer is responsible for treating the arriving packets from MAC layer and updates the flow table to meet the aim of dealing all of the arrived packets. Keeping up with the SDN scheme, the flow table entries are defined in three parts, which are rule, action and statistic information. A rule figures out the features of the packet that belong to the flow, which must be treated by the node simultaneously. The other part of the flow table entry is the action, which is executed by all packets satisfying the above rule. Lastly, the statistic information states the number of packets received in the table flow entry that have satisfied the rule. Moreover, the packets arriving at the forwarding layer are provided by the MAC layer. Depending on this, the packet type will be identified. However, if it is a control packet then it will be sent to the In-Network Packet Processing (INPP) layer. Otherwise, if the packet is a data packet then it will be sent to the forwarding layer where is checked with the rules in the flow table, to see whether it is a match or mismatch. If the packet is a match then it is dealt with the action that is included in the flow table and is related to this packet. Otherwise it is forwarded to the INPP layer [26] [27].

This layer has the ability to process all of the packets arriving at the sensor node based on the information provided by the WISE Flow Table. The information of the table is updated periodically by FWD layer according to the comments received from the controller [27].

D. In-Network Packet Processing (INPP)

This layer is applied over the Forwarding layer. The responsibility of this layer is to implement the important actions needed for aggregating the information by connecting small packets that must be sent through similar paths, so as to reduce the overhead on the whole network [26]. In addition, it is considered as Network Operating System (NOS) for sensor nodes and responsible of processing data, providing the node the capability of making decisions.

E. Adaptation

This layer is included in the WISE-VISOR controller and the sink nodes; it has the ability to interface between WSN platform, which is based on 802.15.4 and the WISE-VISOR controller, which is based on TCP/IP and Ethernet 802.3. Consequently, it modifies the TCP/IP address to the network ID and vice versa, to make an adaptation, which is considered in SDN for WSN [26] [98]. As the main responsibility is formatting the received message from sink in such a way, which can be treated by the WISE- VISOR and vice versa.

F. Network Operating System (NOS) layer

This layer on the top of the whole SDN network and managing the SDN configuration policies [19].

5.3.3 SDN-WISE Packet Description

The packet of SDN-WISE has 10 Bytes in the header; the packet is divided into eight different parts [26]:

The packet size (payload) is given by **Packet Length 7 bits.** The controllers included in the packet, which are identified by **Scope 7 bits**, **14 bits** for the **source address** and another **14 bits** for destination. The packet is marked by using a flag **U 0 bits** that is essential for sending this packet to the nearest sink. In addition, the recognition of the message type of the packet is delivered from others is known in **Type of Packet 7 bits** is **Time To Live (TTL)** takes **7 bits**.

5.4 Proposed Model for Profiling PC model for SDN-WISE

To address the previous issues, the PM is considered in WSN based on SDN architecture of IoT applications and services as the main part for solving the complexity problem in management of the WSNs. In addition, measuring the efficiency of PC in SDN-WISE, is considered a crucial factor for fulfilling the requirements of the IoT regarding resource usage with improved NQoS. The problem is tackled by profiling a model for the SDN-WISE PC.

5.4.1 System Model and Network Components

This section describes the components of the proposed system model regarding sensors, sinks, and WISE-VISOR, with all of the data stream that are contributing to the system data plane traffic and control plane signalling. In addition, this work proposes an observation of PC modelling in the SDN-WISE based on the prototype parameters in [27].

A sensing field is logically divided into N sections, depending on the number of sinks, as shown in Figure 5.2. In addition, the SDN-WISE protocol stacks as shown in Figure 5.1 is based on the SDN-WISE architecture proposed in [27], with complete layers related to the proposed architecture. The introduced network model is based on the power model parameters definitions and the simulation parameters, which are listed in table 5.1and table 5.2 respectively. The sensing field size is assumed as NxN. In consequence, the proposed network model would contain the following elements:

- **Sensor Nodes (SNs):** The sensors contain the following layers:

  1. **IoT Application layer**
  2. **SDN-WISE layers**
     - **Topology Discovery (TD) layer**
     - **In-Network Packet Processing (INPP) layer**
     - **Forwarding layer (FWD**)
  3. **MAC layer**
  4. **Physical layer**

- **Sinks Nodes (SNKs):** Are an aggregation point for sensors with short-range interfaces and the gateway between sensor nodes, which are running the data plane and the elements of control plane of the proposed network. It is a bidirectional protocol convertor. It is the same as a sensor node but with unlimited energy. This sink roots the same layers in sensor nodes in addition to extra layers, which are **the adaptation layer**, as well as **TCP/IP** protocol with **external network interface** as Ethernet 802.3. This sink has two ports one deals with 802.15.4 for connecting with WSN (network elements) and the other corresponds to the Ethernet 802.3 for connecting with external network as well as SDN-WISE controller.
- **WISE-VISOR:** Is a controller device, which is a proxy between the SDN-WISE network components and other networks. Moreover, the WISE-VISOR is responsible for creating and managing the SDN-WISE flow table. Furthermore, it has the **Adaptation Layer** and the **Topology Management (TM)** layer.
- **NOS** : This layer is runs beyond the application layer.

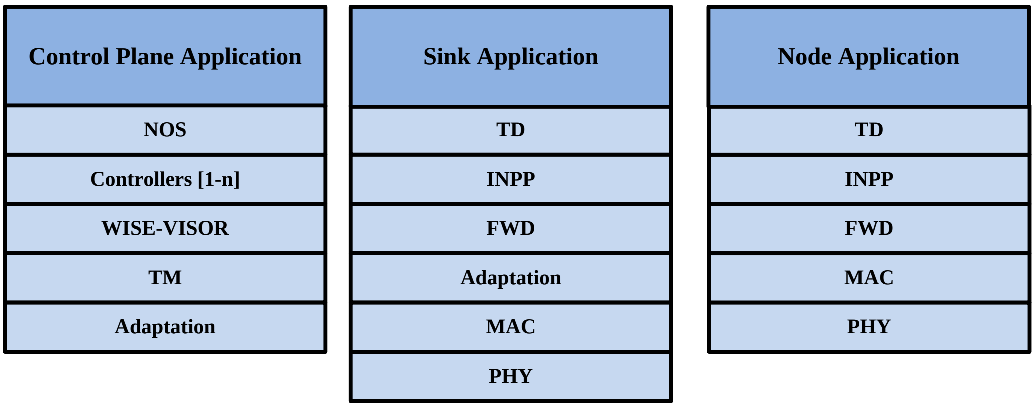


Figure 5.1 SDN-WISE protocol stack

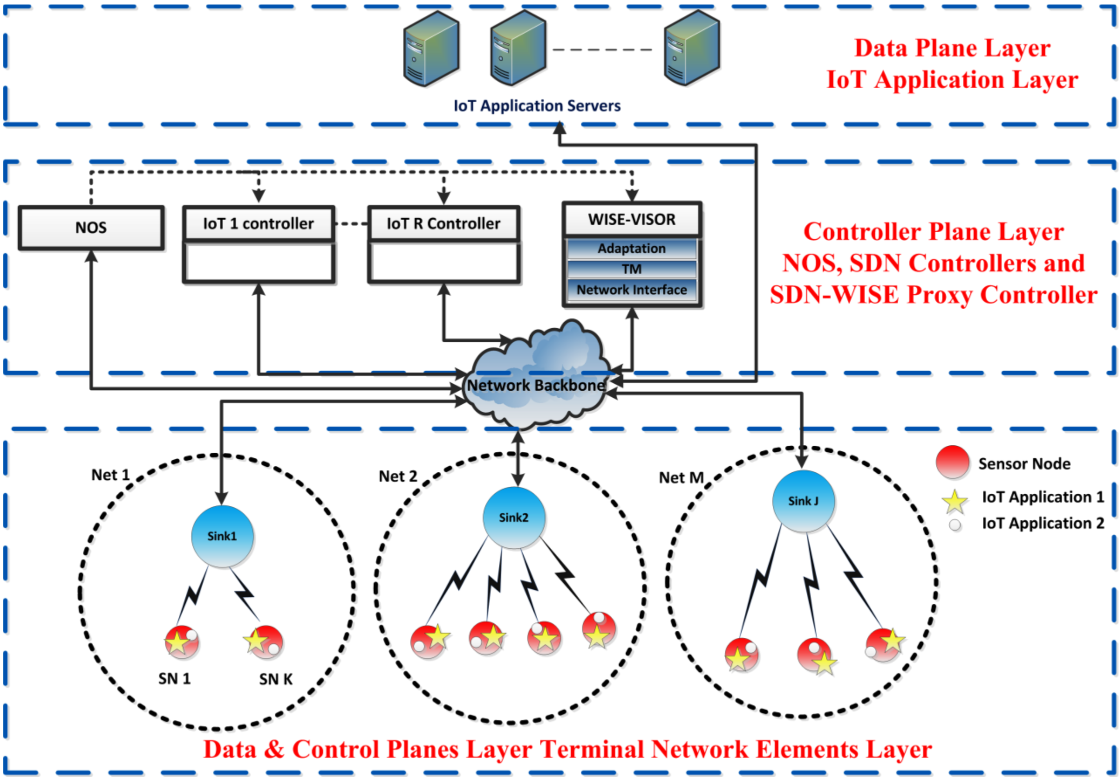


Figure 5.2 Architecture SDN-WISE system model

5.4.2 POWER MODEL COMPONENTS

The proposed PM model introduces all SDN-WISE architecture network components such as terminal SDN-WISE data plane network elements represented by SNs and SNKs, as well as external SDN-WISE elements such as WISE-VISOR and NOS as shown in the following equation.

$$\mathbf{PM} = \mathbf{PC_{SDN-WISE}} \tag{5.1}$$

$$\mathbf{PC_{SDN-WISE}} = \sum_{\mathbf{Net=1}}^{\mathbf{M}} \Big(\sum_{\mathbf{SN=1}}^{\mathbf{K}_{Net}} \mathbf{PC_{SN\,(CTR+DAT)}} + \sum_{\mathbf{SNK=1}}^{\mathbf{J}_{Net}} \mathbf{PC_{SNK(CTR+DAT)}} \Big)_{\mathbf{Net}} + \mathbf{PC_{(WISE-Visor)}} + \mathbf{PC_{NOS}} \tag{5.2}$$

Where, M is the number of terminal Networks (WSNs) (Nets), K_Net and J_Net are the numbers of SNs & SNKs in each Net respectively.

The SDN-WISE's PC Model components are defined in Table 5.1.

The first part of the Equation (5.2) is composed of the power consumed by each SN ($\mathrm{PC_{SN}}$) and SNK ($\mathrm{PC_{SNK}}$) related to DATA plane traffic (DAT) PC and other for CONTROL plane signalling (CTR) PC. The second part is the power consumed by the WISE-VISOR controller, which is considered as part of the control plane signalling PC. Moreover, the last part is according to SDN architecture, which represents the NOS layer PC that has included as part of the total SDN-WISE related functions and applications PCs.

Table 5.1 Parameter Definitions

Parameters	**Definition**
SN	Sensor Node
SNK	Sink Node
CTR	CONTROL Plane Signalling
DAT	DATA Plane Traffic
$\mathrm{PC_{SN}}$	Power Consumption in SN
$\mathrm{PC_{SNK}}$	Power Consumption in SNK
$\mathrm{PC_{(WISE-Visor)}}$	Power Consumption in WISE-VISOR Controller
$\mathrm{PC_{NOS}}$	Power Consumption in NOS
$\mathrm{PC_{TM}}$	Power Consumption in Topology Management
$\mathrm{PC_{Adapt}}$	Power Consumption in Adaptation Processing

5.4.2.1 Power Consumption for Sensor Nodes (PC_{SN})

$$\mathbf{PC_{SN}} = \mathbf{PC_{SN\,(DAT)}} + \mathbf{PC_{SN\,(CTR)}} + \mathbf{PC_{SN\,(idle)}} \tag{5.3}$$

In this section, the $\mathbf{PC_{SN}}$ is related to DATA and CONTROL planes PC, which are $\mathbf{PC_{SN\ (DAT)}}$ and $\mathbf{PC_{SN\ (CTR)}}$ as well as, $\mathbf{PC_{SN\ (idle)}}$, which is related to the power dissipated in idle mode. The measured PC of the idle part is considered according to wireless module device developed by embit for LR-WPAN applications. The module associates high performance to small dimensions and low cost to support easy and simple integrator for the system enabling IEEE 802.15.4 / ZigBee low range wireless connectivity and multi-hop networking into existing products. This device is organised as an embedded micro system or simple data modem for low power applications in the 2.4 GHz. It is based on a Texas Instruments **EMB-Z2530PA [**102**]** based on [27] as shown in table 5.2.

Table 5.2 Simulation Parameters

Parameter Name	**Value**
PC (Idle)	24mW [102]
Energy consumption of circuit (E_{elect})	50 nJ/bit
Channel Parameter of free space model(E_{fs})	10 pJ/bit/m^2
Energy consumption for INPP	5 nJ/bit/signal/process.packet [99]
Energy consumption for FWD	5 nJ/bit/signal/process.packet [99]
Energy consumption for TD	5 nJ/bit/signal/process.packet [99]
PC (SDN controller)	0.3909 W [100]
Energy consumption for Adaptation process	197.208 nJ [101]
Network Size (N*N)	350 m ×350 m
Transmission Range	155 m
Number of SNs	2000
Number of SNKs	16
Number of Terminal WSNs	16 WSNs
Number of SNs per Terminal WSN	125 SNs

The first two parts of PC in SN in Equation (5.3) are related to the data plane and control plane in SN respectively. Each part in the Equation (5.3) is described in detail in the following sections:

A. Power Consumption for Data Plane Traffic in SNs ($PC_{SN\,(DAT)}$)

This section of the PC model is based on single-hop, the PC in any SN is related to the power required to transmit and receive data from source to destination. This PC model is based on the energy consumption model used in chapter three and also in [99]. Moreover, in the SDN-WISE layer, this is related to three processes sub-layers (INPP, TD and FWD).

The PC of data plane traffic in SN is a composite of transmission and reception processes PC.

$$\mathbf{PC_{SN(DAT)}} = \mathbf{PC_{SN(DAT)(Tx)}} + \mathbf{PC_{SN(DAT)(Rx)}} \tag{5.4}$$

where $\mathbf{PC_{SN(DAT)(Tx)}}$ and $\mathbf{PC_{SN(DAT)(Rx)}}$ represents the PC for transmission and reception processes respectively.

The transmission PC of the transmitted data plane traffic for single SN is described as

$$\mathbf{PC_{SN(DAT)(Tx)}} = \mathbf{E_{Tx}} * \mathbf{R_{dataTx}} + \mathbf{E_{Tx(SDN-WISE\ Layer)}} * \mathbf{R_{Tx(SDN-WISE\ Layer)}} \tag{5.4a}$$

where $\mathbf{R_{dataTx}}$ and $\mathbf{E_{Tx}}$ are the bit rate of data traffic stream and consumed energy per bit in 802.15.4 layers, respectively. In addition, $\mathbf{R_{Tx(SDN-WISE\ Layer)}}$ is the bit rate of transmitted data streams in SDN-WISE layers.

The PC for the reception process in a single SN as

$$\mathbf{PC_{SN(DAT)(Rx)}} = \mathbf{E_{Rx}} * \mathbf{R_{dataRx}} + \mathbf{E_{Rx\,(SDN-WISE\ Layer)}} * \mathbf{R_{Rx(sdn-wise\ Layer)}} \tag{5.4b}$$

Where $\mathbf{R_{dataRx}}$ and $\mathbf{R_{Rx(sdn\text{-}wise\ Layer)}}$ are the bit rates of received data streams in 802.15.4 and SDN-WISE layers, respectively, as well as $\mathbf{E_{Rx}}$ and $\mathbf{E_{Rx(SDN\text{-}WISE\ Layer)}}$ are the energy consumption rates of 802.15.4 reception process and SDN-WISE layers streaming process , respectively. Therefore, the total PC of transmitted data plane traffic in a single terminal WSN contains k SNs (the sum of each individual SN that sends data plane traffic).

$$\mathbf{PC_{SN(Tx)}} = \sum_{\mathbf{n=1}}^{\mathbf{K}} [\mathbf{E_{Tx(n)}} * \mathbf{R_{dataTx(n)}} + \mathbf{E_{Tx(SDN-WISE\ Layer)(n)}} * \mathbf{R_{Tx(SDN-WISE\ Layer)(n)}}] \tag{5.5}$$

Where n represents the SN and K is the total number of SNs.

Moreover, the total PC of received data plane traffic in a single terminal WSN contains k SNs will be the sum of each individual SN that receives data plane traffic.

$$\mathbf{PC_{SN\,(RX)}} = \sum_{n=1}^{K}[\mathbf{E_{Rx(n)}} * \mathbf{R_{dataRx(n)}} + \mathbf{E_{Rx\,(SDN-WISE\,Layer)(n)}} * \mathbf{R_{Rx(SDN-WISE\,Layer)(n)}}]$$

where (5.6)

$$\mathbf{E_{Tx(L,d)}} = \mathbf{L} * \mathbf{E_{elec}} + \mathbf{L} * \boldsymbol{\varepsilon}_{\mathbf{fs}} * \mathbf{d^2} \qquad , \mathbf{d} \leq \mathbf{d}_{\circ} \tag{5.7}$$

$$\mathbf{E_{Rx}(L)} = \mathbf{L} * \mathbf{E_{elec}} \tag{5.8}$$

where,

$$\mathbf{d}_{\circ} = \sqrt{\frac{\boldsymbol{\varepsilon}_{\mathbf{fs}}}{\boldsymbol{\varepsilon}_{\mathbf{mp}}}} \tag{5.9}$$

$$\mathbf{E_{Tx(SDN-WISE\,Layer)}} = \mathbf{E_{Rx(SDN-WISE\,Layer)}} = \mathbf{E_{INPP}} + \mathbf{E_{TD}} + \mathbf{E_{FWD}} \tag{5.10}$$

In the given model, $\mathbf{E_{Tx}}$ of energy is consumed by each sensor node to transmit each packet length (L in bits to obtain the energy density as Joule/bit over a distance d. However, the energy that is consumed by the receiving packet is $\mathbf{E_{Rx}(L)}$. In this model, the energy consumed per bit is $\mathbf{E_{elec}}$ and it is used to run the transmitter or receiver circuit, where ε_{fs} represents the transmitter amplifier's efficiency of channel conditions [99] [23].The model is based on free space (power loss) (d^2). A free space model is calculated based on the distance between a transmitter and receiver.

where, $\mathbf{PC_{SN\,(TX)}}$ is the power consumption for transmitting information related to data plane for SN, $\mathbf{PC_{SN\,(RX)}}$ is the power deplaced when the sensor node receives data related to the data plane part and $\mathbf{R_{data}}$ is the achieved data rate in PHY layer presented by the wireless link between transmitter node and receiver node as proposed in the model of single hop transmission. Moreover, $\mathbf{R_{Tx(SDN-WISE\,Layer)}}$ and $\mathbf{R_{Rx(SDN-WISE\,Layer)}}$ are the data rate in SDN-WISE layer for transmitting and receiving data streams respectively. Furthermore, $\mathbf{E_{Tx(SDN\text{-}WISE\,Layer)}}$ and $\mathbf{E_{Rx(SDN\text{-}WISE}}$

$_{\mathbf{Layer)}}$ are the consumption energy densities per bit of data streams in SDN-WISE layer, they are a $\mathbf{E_{INPP}}$, $\mathbf{E_{TD}}$ and $\mathbf{E_{FWD}}$ as aggregation layers [99][27][70].

B. Power Consumption for Control Plane Signalling in SNs ($PC_{SN\,(CTR)}$)

The PC of control plane signalling in SN ($\mathbf{PC_{SN(CTR)}}$) is the same as of PC of data plane traffic, it is also a composite of transmission and reception processes PCs as in equation (5.11):

$$\mathbf{PC_{SN(CTR)} = PC_{SN(CTR)(Tx)} + PC_{SN(CTR)(RX)}} \tag{5.11}$$

where $\mathbf{PC_{SN(CTR)(Tx)}}$ and $\mathbf{PC_{SN(CTR)(Rx)}}$ representing the PC of transmission and reception processes in the SN of control signalling respectively. The PC of transmission process of control plane signalling in a single SN can be described as:

$$\mathbf{PC_{SN(CTR)(Tx)} = PC_{(REP)(Tx)} + PC_{(REQ)(Tx)} + PC_{(BEC)(TX)}} \tag{5.11a}$$

$$\mathbf{PC_{SN(CTR)(Rx)} = PC_{(OPEN)(Rx)} + PC_{(Config)(Rx)} + PC_{(BEC)(Rx)} + PC_{(RES)(Rx)}} \tag{5.11b}$$

and the total PC of transmission and receiving processes of control plane signalling of k SNs as:

$$\mathbf{PC_{SN(CTR)(Tx)} = \textstyle\sum_{n=1}^{k}[\,PC_{(REP)(Tx)} + PC_{(REQ)(Tx)} + PC_{(BEC)(TX)}]_{(n)}} \tag{5.12}$$

$$\mathbf{PC_{SN(CTR)(Rx)} = \textstyle\sum_{n=1}^{k}[\,PC_{(OPEN)(Rx)} + PC_{(Config)(Rx)} + PC_{(BEC)(RX)} + PC_{(RES)(Rx)}]_{(n)}} \tag{5.13}$$

$$\mathbf{PC_{REP} = E_{Tx} * R_{REP} + E_{SDN-WISE\ Layer} * R_{REP-SDN-WISE\ Layer}} \tag{5.14}$$

$$\mathbf{PC_{REQ} = E_{Tx} * R_{REQ} + E_{SDN-WISE\ Layer} * R_{REQ-SDN-WISE\ Layer}} \tag{5.15}$$

$$\mathbf{PC_{RES} = E_{Rx} * R_{RES} + E_{SDN-WISE\ Layer} * R_{RES-SDN-WISE\ Layer}} \tag{5.16}$$

$$\mathbf{PC_{OPEN} = E_{Rx} * R_{OPEN} + E_{SDN-WISE\ Layer} * R_{OPEN-SDN-WISE\ Layer}} \tag{5.17}$$

$$\mathbf{PC_{Config} = E_{Rx} * R_{Config} + E_{SDN-WISE\ Layer} * R_{Config-SDN-WISE\ Layer}} \tag{5.18}$$

$$\mathbf{PC_{BEC-Rx} = E_{Rx} * R_{BEC(SN-SNs-SNK)} + E_{SDN-WISE} * R_{BEC-SDN-WISE\ Layer}} \tag{5.19}$$

$$\mathbf{PC_{BEC-Tx} = E_{Tx} * R_{BEC(SN-SN)} + E_{SDN-WISE\ Layer} * R_{BEC-SDN-WISE\ Layer}} \quad (5.20)$$

$\mathrm{R_{Report}}$, $\mathrm{R_{Requist}}$ and $\mathrm{R_{Beacon}}$ are the associated packet rates related to the information of the control plane signalling, processed as transmission data from the sensor nodes to the controller, and manipulated by sensor nodes. Depending on these data, the controller will build the flow table. However, $\mathrm{R_{Response}}$, $\mathrm{R_{Openpath\ and}}$ $\mathrm{R_{Config}}$ are bits rate received from control plane. In addition, $\mathrm{R_{SDN-WISE\ Layer}}$ is the bit rate in SDN-WISE layer. All of them are described in Table 5.3 based on [103]. $\mathrm{E_{TX}}$ and $\mathrm{E_{RX}}$ are same as the Equations (5.7) and (5.8).

Furthermore, $\mathbf{PC_{REP}}$, $\mathbf{PC_{REQ}}$, $\mathbf{PC_{BEC-TX}}$, $\mathbf{PC_{RES}}$, $\mathbf{PC_{OPEN}}$, $\mathbf{PC_{Config}}$ and $\mathbf{PC_{BEC-RX}}$ respectively are power consumed by report, request, beacon for transmitting, response, open path, configuration and beacon for receiving, respectively.

Table 5.3 Control Signalling Packet Size

Control Packet Types	**Size/Bytes without SDN-WISE Header**	**Source**	**Destination**
***Report**	**18**	**Sensor, Sink**	**Controller**
Config	**3**	**Controller**	**Sensor, Sink**
****Request**	**92**	**Sensor, Sink**	**Controller**
*****Open Path**	**11**	**Controller**	**Sensor & Sink**
RegProxy	**28**	**Sink**	**Controller**
Beacon	**2**	**Sensor, Sink**	**Sensor, Sink**
Response	**6**	**Controller**	**Sensor, Sink**

* This is for 5 neighbour nodes [27]

** Max value

*** This is for window size=2 and nodes=3 [27]

5.4.2.2 Power Consumption for Sink Node (PC_{SNK})

$$\mathbf{PC_{SNK}} = \mathbf{PC_{SNK\,(DAT)}} + \mathbf{PC_{SNK\,(CTR)}} + \mathbf{PC_{SNK\,(idle)}} + \mathbf{PC_{SNK\,(Eth-Port-Config)}} + \mathbf{PC_{SNK\,(Adapt)}} \quad (5.21)$$

where $\mathrm{PC_{SNK\,(DAT)}}$, is the power depleted in sink as the part related to the data traffic and $\mathrm{PC_{SNK\,(CTR)}}$ is the PC in the sink node related to the control signalling, these two parts presents sink's PC in 802.15.4 protocol. Whereas, the TCP/IP data plane traffic and control plane signalling PCs are represented by $\mathbf{PC_{SNK\,(Eth-Port-Config)}}$ as an Ethernet port configuration PC of connection interface of the SNK, which depends on the achieved 802.3 PHY layer link speed. This configuration is based on the (following equations), Table 5.2 and those values depend on the [104].

$$\mathbf{PC_{SNK(Eth-Port-Config)}} = \begin{cases} (\mathbf{0.0183} * \mathbf{L} + \mathbf{277.913}) & \mathbf{10\ Based - T} \\ (\mathbf{0.0026} * \mathbf{L} + \mathbf{291.326}) & \mathbf{100\ Base - T} \end{cases} \quad (5.22)$$

where L is the size in Bytes of transmitted TCP/IP packet.

$$\mathrm{L = SDN - WISE\ Packet\ Size + 66} \quad (5.23)$$

The proposed connectivity with the WISE-VISOR and data plane destination end devices, such as services and/or end users, is provided via an Ethernet link, in order to generalise the PC model, which adapts to the IEEE 802.3. This study investigates the demand of PC in two operation modes, 10 Base-T and 100 Base-T, as shown in Figure 5.3.

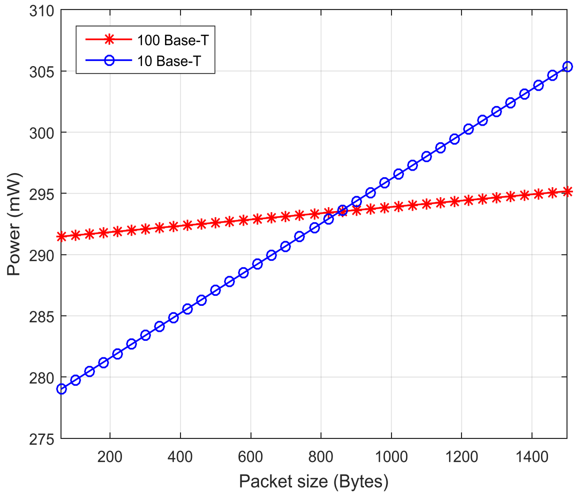


Figure 5.3 Port configuration

This figure clearly illustrates the effect that the operating conditions, which are demonstrated by the achieved Ethernet port traffic. This Port-Config demonstrates one of the important parts between PC and QoS, since it contributes to the offered NQoS that defines the maximum physical layer link throughput, as well as packet size (L) of the transmitted TCP/IP packet as an AQoS metric, and its contribution with total PC. Furthermore, the value of L for control signalling will consider the type of the specified control signalling and the size of the its packet as listed in Table 5.3 .

Another component of PC in SNK is the power consumed in the adaptation process $\mathbf{PC_{SNK\ (Adapt)}}$, which is dissipated for in the formatting process of the up and down data streams of the messages received and/or transmitted from by the sink node. This value of adaptation process PC is assumed to be per packet process for up and down data streams. $\mathbf{PC_{SNK\ (Adapt)}}$ value is assumed from [101] as it was considered in the measurement of PC.

A. Power Consumption for Data Plane Traffic in SNKs ($PC_{SNK(DAT)}$)

This section is related to the SNK PC of data plane traffic, which is based on the upstream traffic from IoT end devices toward destination points. The SNK node will act as a gateway and this $\mathrm{PC_{SNK(DAT)}}$ will be a reception process PC of 802.15.4 interface as $\mathrm{PC_{SNK(DAT)(Rx)}}$ and a transmission process PC of TCP/IP with 802.3 interfaces or Ethernet interface as$\mathbf{PC_{SNK(Eth-Port-Config)}}$. Then, $\mathrm{PC_{SNK(DAT)}}$ is written as:

$$\mathbf{PC_{SNK(DAT)}} = \mathbf{PC_{SNK\,(Rx)}} \tag{5.24}$$

Where

$$\mathbf{PC_{SNK\,(DAT)(Rx)}} = \mathbf{E_{Rx}} * \mathbf{R_{dataRx}} + \mathbf{E_{Rx(SDN-WISE\,Layer)}} * \mathbf{R_{Rx(SDN-WISE\,Layer)}} \tag{5.25}$$

This equation is based on the up-link direction; therefore, the flow of traffic will only deal with the received up-stream data flow, while the transmission PC of TCP/IP with Ethernet port will be processed as $\mathbf{PC_{SNK(Eth-Port-Config)}}$.

B. Power Consumption for Control Plane Signalling in SNKs ($PC_{SNK(CTR)}$)

The control plane signalling is directed into two directions (up-link and down-link) and it specifies both groups of signalling.

$$\mathbf{PC_{SNK\,(CTR)}} = \mathbf{PC_{SNK\,(CTR)(Tx)}} + \mathbf{PC_{SNK(CTR)(Rx)}} \tag{5.26}$$

for single SNK node the PC for transmission process of 802.15.4 radio and SDN-WISE processing $\mathbf{PC_{SNK\,(CTR)(Tx)}}$ is written as:

$$\mathbf{PC_{SNK(CTR)(TX)}} = (\mathbf{E_{Tx}} * \mathbf{R_{Config}} + \mathbf{E_{SDN-WISE\,Layer}} * \mathbf{R_{Config-SDN-WISE\,Layer}}) + (\mathbf{E_{Tx}} * \mathbf{R_{OPEN}} + \mathbf{E_{SDN-WISE\,Layer}} * \mathbf{R_{OPEN-SDN-WISE\,Layer}}) + (\mathbf{E_{Tx}} * \mathbf{R_{BEC}} + \mathbf{E_{SDN-WISE\,Layer}} * \mathbf{R_{BEC-SDN-WISE\,Layer}}) + (\mathbf{E_{Tx}} * \mathbf{R_{RES}} + \mathbf{E_{SDN-WISE\,Layer}} * \mathbf{R_{RES-SDN-WISE\,Layer}}) \tag{5.27}$$

and $\mathbf{PC_{SNK\,(CTR)(Rx)}}$ represent the PC for reception process of 802.15.4 radio and SDN-WISE processing for single SNK node is written as:

$$\mathbf{PC_{SNK(CTR)(Rx)}} = (\mathbf{E_{Rx}} * \mathbf{R_{REP}} + \mathbf{E_{SDN-WISE\,Layer}} * \mathbf{R_{REP-SDN-WISE\,Layer}}) + (\mathbf{E_{RX}} * \mathbf{R_{REQ}} + \mathbf{E_{SDN-WISE\,Layer}} * \mathbf{R_{REQ-SDN-WISE\,Layer}}) + (\mathbf{E_{RX}} * \mathbf{R_{BEC}} + \mathbf{E_{SDN-WISE\,Layer}} * \mathbf{R_{BEC-SDN-WISE\,Layer}}) \tag{5.28}$$

5.4.2.3 Power Consumption for SDN-WISE Controller

The power consumption for SDN-WISE controller, which includes the SDN-WISE ($\mathrm{PC_{CTR}}$) as Controller Application PC, TM PC ($\mathrm{PC_{TM}}$) and Adaptation process PC as $\mathrm{PC_{Adapt}}$, .The controller application is assumed to have a constant PC value based on the controller as an application as described in [100] as shown in Table 5.2.

5.4.2.4 Assumptions

There are some specific assumptions considered in the calculations of total PC of the introduced network model. Firstly, the TTI of the control signalling assumed to be the same signalling rate for all of them. Secondly, the transmitted data plane traffic and the interaction between achieved 802.15.4 radio throughput and IoT application data rate as a required AQoS. Thirdly, the single hop transmission between SNs and SNK in the 802.15.4 radio is considered, in order to get a fair comparison in the data analysis, and to get a fair contrast between data plane traffic and control plane signalling streams in end nodes PC with respect to TTI and IoT applications data rates. Furthermore, all constant values of PC based on in our profiling PC model are listed in Table 5.2 [100].

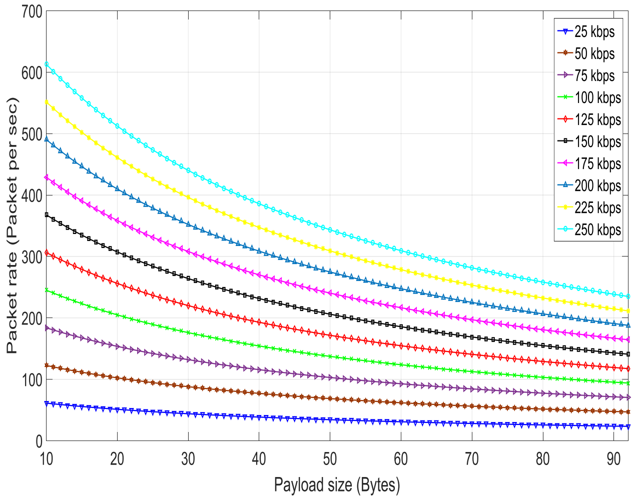


Figure 5.4 Packet data unit for packet rate with respect to payload size of different link rates

In addition, the packet per sec is considered according to the Equation below. As well as, Figure 5.4 shows the analysis of packet rate as a function of IoT application payload size for different 802.15.4 physical layer link rates between SN and SNK.

The Uplink packet rate

$$\textbf{Packet Per sec (PPs)} = [\textbf{Link Rate (bps)}] \,/\, [[\textbf{Payload Size(Byte/Packet)} + \textbf{41}] * \textbf{8}] \qquad (5.29)$$

The maximum pps = 235 (PPs) with 92 (Bytes) payload packet size when the link rate is 250 (kbps) and 612.7 with 10 (Bytes) payload when the link rate is 250 (kbps). In the proposed model we considered the packet per sec as 66.667.

5.5 Results and Discussion

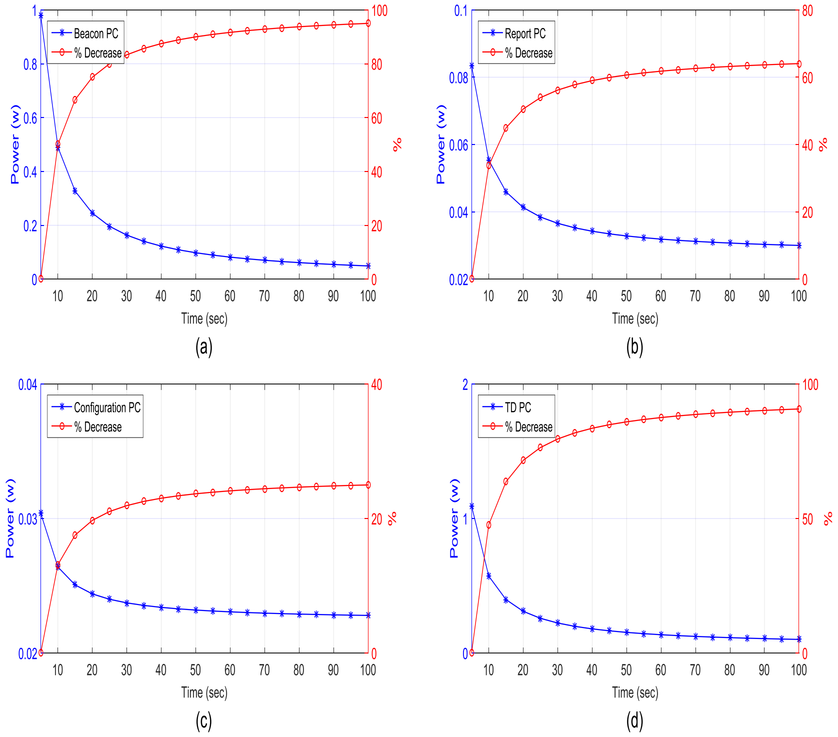
The proposed network model is used to effectively analyse and identify the PC in terms of the proposed network nodes setting as shown in table 5.2, which were chosen based on [27] and the scenario of 2000 sensor nodes and 16 sinks., thereby each sink node will attached with 125 SNs. In this section the results discussion will be divided into two parts, which are control and data planes.

4.5.1 Control Plane PC

In control plane, the controllers and WISE-VISOR are determined by the network logics. The power consumed in the control plane is related to the aspects below:

4.5.1.1 Topology Discovery PC

The first set of the PC model elements to consider here is the topology discovery signalling power consumption with respect to the TTI. This is because the construction of the network topology initiates at the network elements, which is sink nodes and SNs as described in TD (5.3.2 (A)) section [27].



(a) (b) (c) (d)

Figure 5.5 PC Topology Discovery control signaling (a) Beacon, (b) Report, (c) Configuration and (d) Total

Figure 5.5 shows an overview of all the elements of the TD signalling PC in the network elements of the proposed network scenario. Figure 5.5 (a) presents the PC of the Beacon signalling. The most interesting aspect of this graph is the interaction between the PC of the Beacon traffic in the network elements and TTI. Furthermore, Figure 5.5 (a) provides the percentage decrease in the Beacon process's PC with respect to TTI. It shows a significant reduction of 50% in the consumed power when the TTI is increased to 10 seconds, and 75% at 20 seconds. Surprisingly, only a 20% extra reduction in power consumption is achieved when the TTI reaches 90 seconds in comparison with 20 seconds (TTI beacon PC).

Figures 5.5 (b) and (c) demonstrate the PC and percentage decrease of the Report signalling and Configuration signalling with respect to TTI. The Report and Configuration signalling represent the complementary processes of the network TD framework, since they are accompanying with the Beacon signalling process. Report signalling is used to send the required information from the terminal devices SNs and sink nodes to the WISE-VISOR as well as to the TM application. The Configuration signalling is employed to send the related modification/adjustment information from

Wise-Visor to the terminal devices. As shown in Figure 5.5 (b), the PC of Report signalling decreases with an increase in the TTI. The percentage decrease of Report PC gives 33.71% decrease in PC at 10 seconds TTI and achieves 50.56% decrease of PC at 20 seconds TTI, whereas the percentage decrease gives 63.67% at 90 seconds TTI.

Figure 5.5 (c) shows a decrease in Configuration PC with the increase in TTI, as the percentage decrease of configuration signalling PC achieves 13.15% at 10 seconds TTI and 19.73% at 20 seconds, while the percentage decrease only reaches 24.85% at 90 seconds. Figure 5.5 (d) provides the total PC and percentage decrease of all TD signalling PC; the Beacon, Report and Configuration PCs were added together, and the PC of the total TD signalling decreased as the TTI increased. The percentage decrease of TD PC gives 47.73% and 71.6% at 10 and 20 seconds TTI respectively, whereas it achieves 90.16% at 90 seconds of TTI. Table 5.4 lists the PC percentage share of Beacon, Report and Configuration of total TD signalling PC. Closer inspection of the table shows the Beacon share decreased as the TTI increased. Furthermore, the Report and Configuration PCs increased with an increased TTI.

Table 5.4 Percentage Share of TD Signaling PC

TTI (sec)	Beacon %	Report %	Configuration %
5	89.59	7.842	2.778
10	85.71	9.677	4.616
20	78.87	13.28	7.852
40	68.01	19	12.99
80	53.33	26.74	19.93
90	50.6	28.18	21.22

4.5.1.2 Request and Response PC

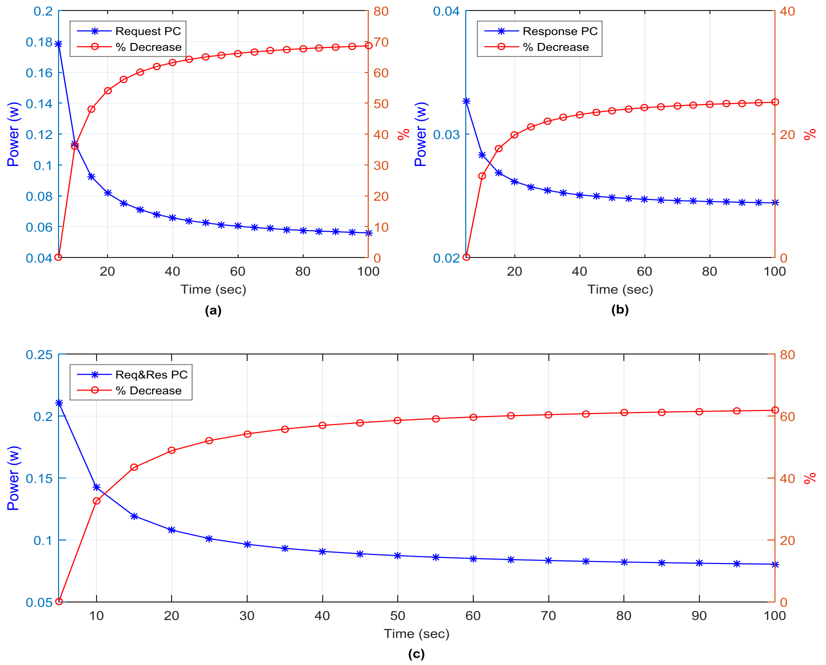


Figure 5.6 (a) Request, (b) Response and (c) Total Request and Response with respect to the TTI

When the received packet passed the accepted ID process, at network element, as soon as there is a no match occurrance in the Flow Table or a specific amount of delay is experienced in FWD process and the INPP as well cannot state any decision about the processed packet, then Request packet will be sent to the WISE-VISOR. Consequently, the Response packet containing updated flow rules will be activated according to the received Request. Figures 5.6 (a) and (b) provides an overview of the PC for Request and Response control signalling in the network elements in the proposed model. Assuming that all network elements, excluding the sink nodes, was sending Request and receiving Response. Essentially, the transmission of the Request packet and the complementary Responses are not carried out on a periodic basis; their presence is a part of the control signalling rules framework, which is set by NOS.

Figure 5.6 (a) presents the sum of the Request signalling PC and the percentage decrease in PC with respect to TTI. The PC decreases as the TTI increases; the percentage decrease provides an indication that the decrease will be 36.1% at 10 seconds and 68.18% at 90 seconds.

Figure 5.6 (b) introduces the Response signalling PC and the percentage decrease in PC with respect to TTI. It shows a decrease in the PC as the percentage decreases by 13.25% at 10 seconds TTI and 25.01% at 90 seconds.

Figure 5.6 (c) gives both the Request and Response processes PC and their percentage decrease with TTI. The PC decreases as the TTI increased, which is clear from the percentage decrease by 32.56% at 10 seconds, 48.84% at 20 seconds and 61.5% at 90 seconds. The most interesting aspect of this graph is that the percentage decrease in variation rate of PC reduces as the TTI increases. The reduction in PC performance is 12.66% over 20 seconds.

4.5.1.3 Registry Proxy and Open Path PC

RegProxy signalling is considered to be a part of the TD signalling of SDN-WISE control plane signalling, as its packet is used to send some information about the existence of the sink node to the WISE-VISOR in order to prevent the registration of the sink node as a switch on the other controllers.

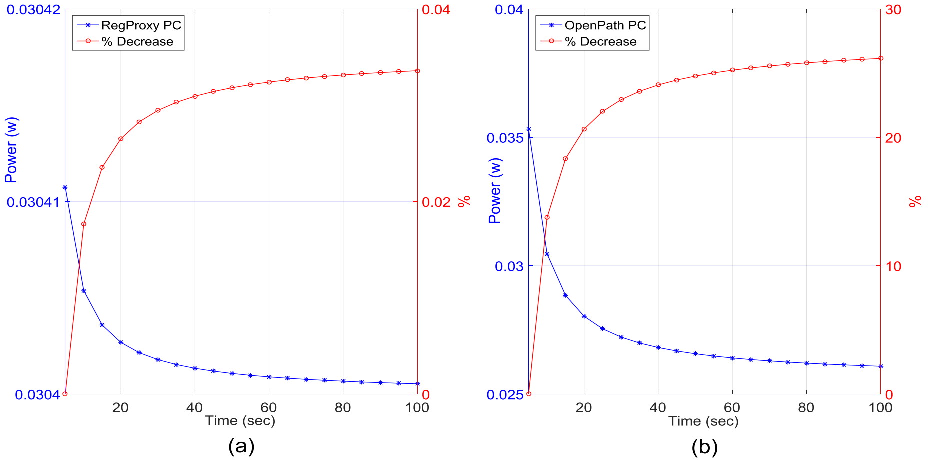


Figure 5.7 (a) Registry Proxy and (b) Openpath

The openpath signalling is initiated at the sink nodes towards the WISE-VISOR. It can be classified as a QoS proactive support signalling. As well as, it is considered one of the important features of SDN-WISE, because it is a reactive support signalling to the data plane traffic, which is used to reduce the control messages transmitted from the control plane towards nodes.

Figure 5.7 (a) shows the PC and the percentage decrease in PC of RegProxy control signalling with respect to the TTI. The PC in the 16 sinks starts with 30.41 mW at 5 seconds TTI and reduces approximately to 30.4 mW at 90 seconds TTI. The percentage decrease in RegProxy PC is 0.0177% at 10 seconds and it increases to 0.0334% at 90 seconds TTI.

Figure 5.7 (b) presents the PC in the network elements of Openpath signalling with respect to TTI. The PC starts with 35.32 mW at 5 seconds of TTI and reduces 26.14 mW at 90 seconds TTI. The percentage decrease in PC with respect to the increase in TTI shows a decrease of 13.76% at 10 seconds TTI and it decreases further by 26% at 90 seconds of TTI.

4.5.1.4 Total Power Consumption for all control Plane

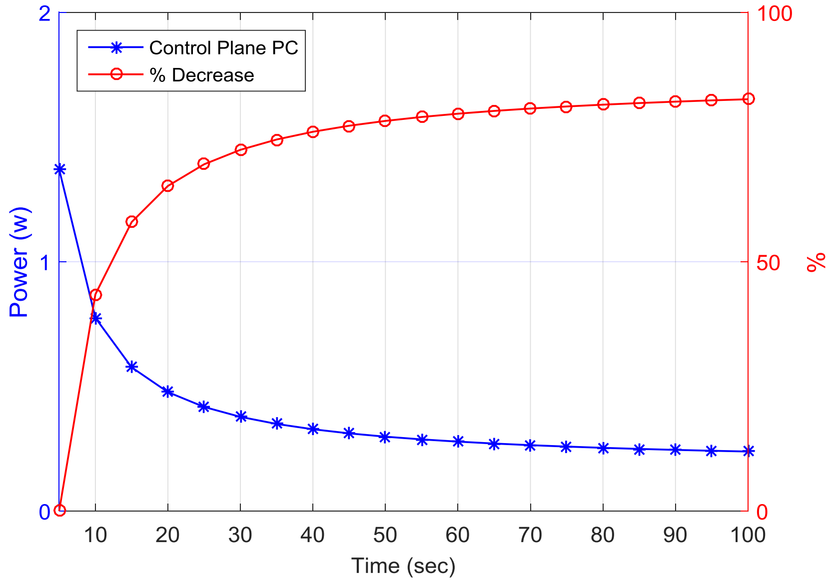


Figure 5.8 The total Control Plane signaling

The aforementioned group of control signalling characterises the control plane signalling of the SDN-WISE in the network elements, and the total PC of the control plane in the network elements. Figure 5.8 represents the total PC of the control plane and the percentage decrease of the PC with respect to TTI in the network elements such as sensor nodes and sinks in the proposed network scenario configuration. This figure shows that the PC decreases as the TTI increases. The percentage decrease is 43.46% when the TTI is 10 seconds, 65.2% at 20 seconds and 82.1% at 90 seconds

TTI. There is a dramatic reduction in the control plane PC in the network elements with respect to TTI, specifically in the range between 0 to 40 seconds, where the percentage decrease reaches 76.06%. In addition, high QoS requires high control signalling for it updating network topology configuration, which needs additional power to be consumed. These percentages can be presented in a pie-chart as shown in Figure 5.9. The original setting of the proposed scenario of SDN-WISE network by [27], where the possibility of having sensor nodes that respond to the incoming messages depending on state variables that reduces the number of messages exchanged between each sensor node and the controller, thus reducing the number of packets sent over the network and the total energy consumption.

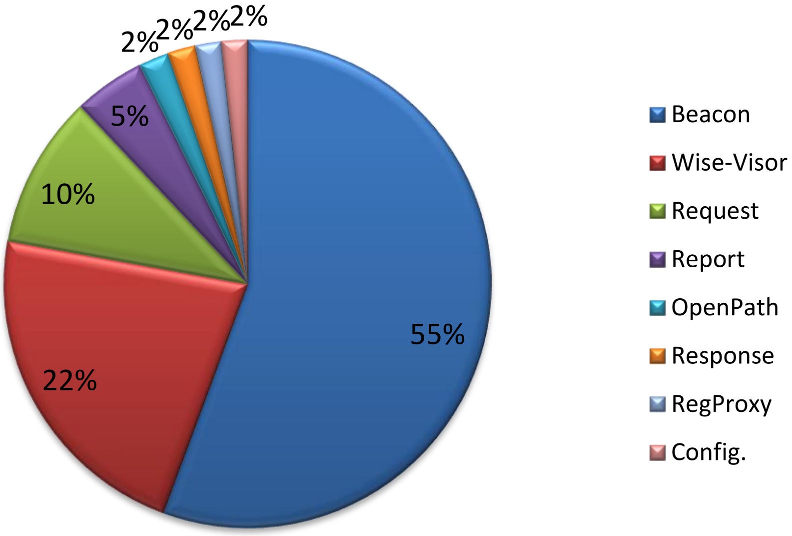


Figure 5.9 Pie-chart for the total PC of Control Plane traffic

4.5.2 Data Plane PC

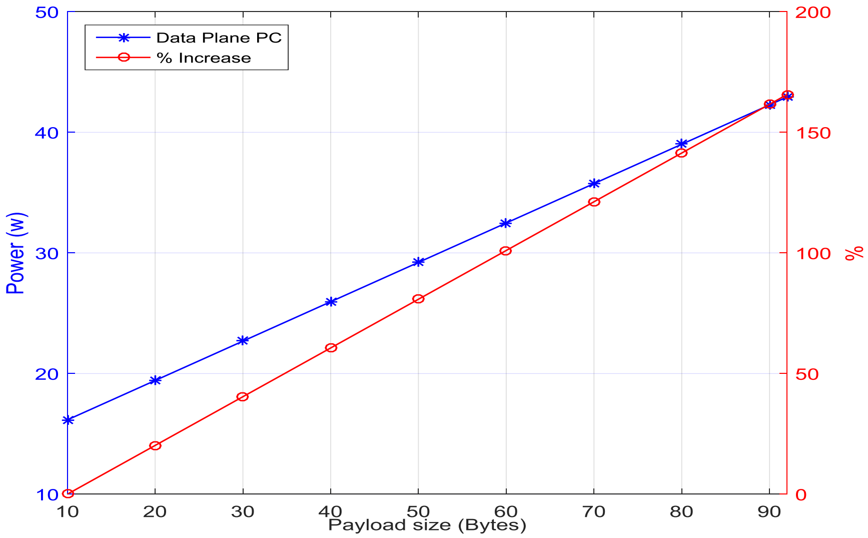


Figure 5.10 The total PC of Data Plane signaling

Figure 5.10 shows the PC of data plane traffic and the percentage increase with respect to the payload size (Bytes). The payload size starts with 10 Bytes, the same as the prototype testbed of the SDN-WISE in [27], and the maximum payload size is set to 92 Bytes in order to match the maximum allowed packet payload size of 802.15.4 standard [46][27]. In this figure there is a clear trend of increasing PC with increasing payload size. The percentage increase of PC is not only directly related to the payload size but also includes the power consumed by the packet headers of SDN-WISE and 802.15.4. The PC of Idle SNs and SNKs as well as the PC of constant port configuration of the Ethernet port in the sinks are not included here, because they are just constant values and enables us to emphasise on the dynamic change of PC with payload size.

4.5.3 Total Power Consumption

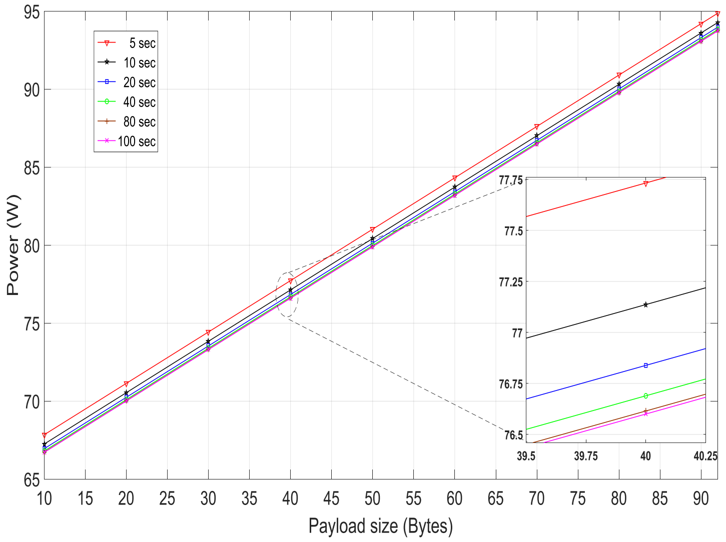


Figure 5.11 The Total PC in Control Plane and Data Plane

Figure 5.11 provides the total PC (Data plane and Control plane) of the network scenario in the SNs and SNKs without idle and constant port configuration. As, it can be seen from Figure 5.11, the total PC reduces when the TTI increases and it increases with the payload size. Closer inspection of the above figure shows that there is a significant decrease of the total PC when the control plane TTI is increased.

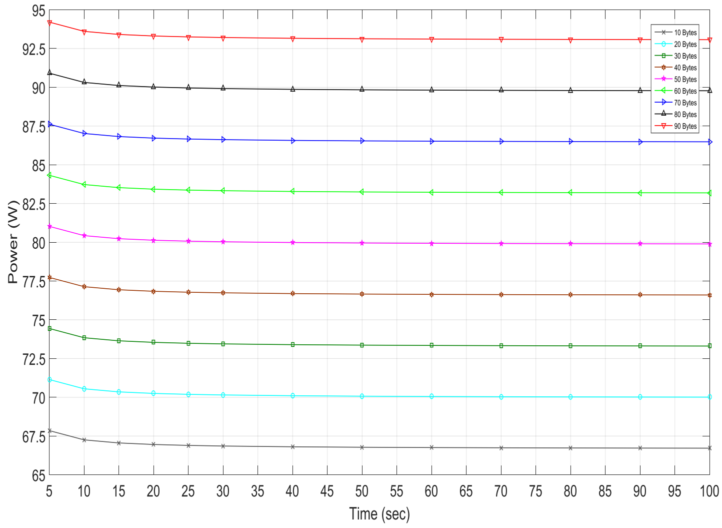


Figure 5.12 Total PC in Control Plane and Data Plane

Figure 5.12 presents the total PC in the network elements, represented by the PC of the SNs and the SNKs including data plane and control plane traffic consumption. The PC is represented by control signalling of the network scenario with the PC of the controller application with respect to variation in control plane signalling TTI (for different payload sizes starting with 10 Bytes and increasing to 90 Bytes).

Figure 5.12 shows that there is significant change (decrease) in the total PC when the TTI of control signalling is between 5 and 40 seconds. The lower TTI means that the highly supported QoS network needs to provide more network resources to control plane signalling, which in turn is reflected as power consumed in the network element. The overall efficiency of the network can be divided into three parts. The first concerns is the PC, the second concerns is the ratio of the data plane traffic to the control plane signalling, and the third is related to NQoS provided to IoT application.

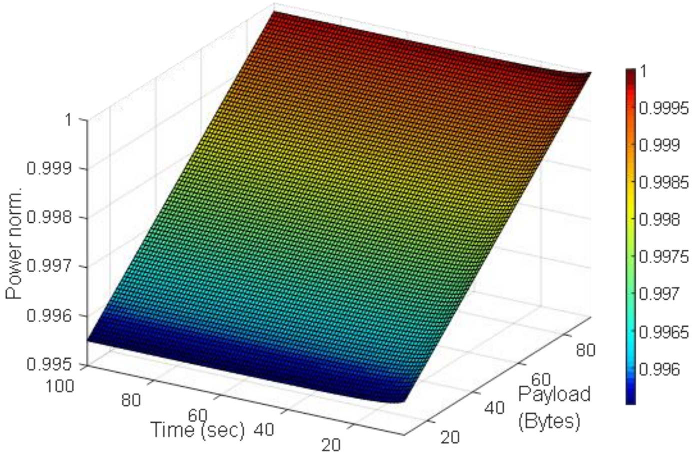


Figure 5.13 The total normalized PC of with respect to TTI and payload size

Furthermore, Figure 5.13 represents a normalized 3D plot of PC with respect to the TTI of control signalling and the payload size of data plane traffic. The idle and constant port configuration power consumptions are not considered and they are only in the SNs and SNKs. This shows that the PC increases with increasing payload size, as well as with decreasing TTI.

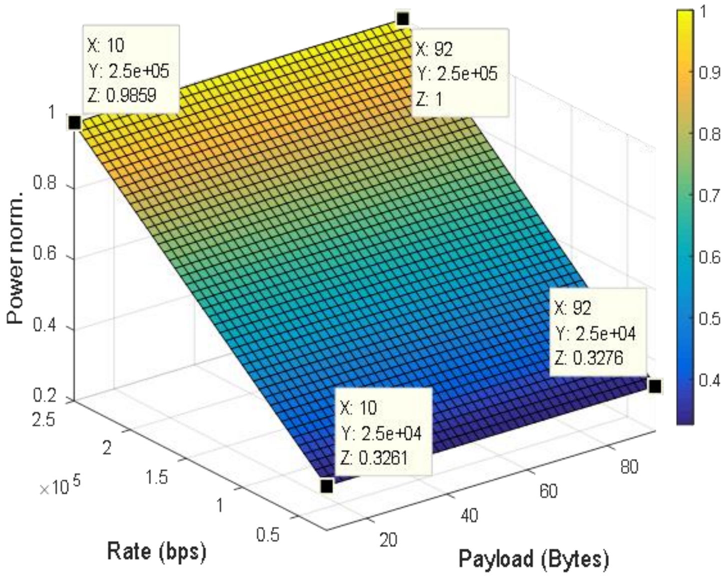


Figure 5.14 The total normalized PC with respect to payload size and data rate

Figure 5.14 shows the normalized total PC for 2000 nodes and 16 sinks with 5 seconds TTI, with respect to payload size (Bytes) and the physical layer link data rate (bps). A closer look at this 3D plot shows that the PC increases linearly with the payload size and with the achieved physical layer link rate between SNs and SNKs. This figure is normalised for the maximum achieved power consumption related to maximum payload size and maximum link rate. Here the dominant factor in PC is the achieved link rate. The minimum link rate of 25 kbps gives 18.25% and 18.43% associated with 10 Bytes and 90 Bytes payload sizes respectively, while the maximum link rate of 250 kbps achieves 98.29% and 100% related to 10 Bytes and 90 Bytes payload sizes, respectively. Moreover, the total PC of the Data Plane (DP) from the whole system power is 98%, whereas the Control Plane (CP) spends just 2%, with minimum TTI (5 sec) and maximum payload size of 92 Bytes as shown in Figure 5.15.

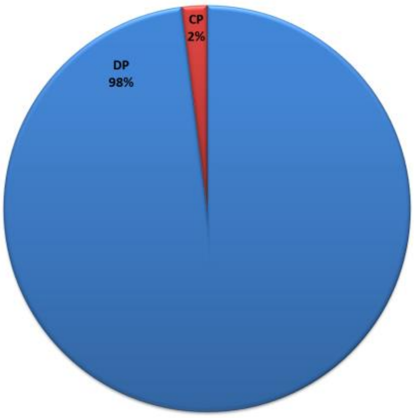


Figure 5.15 The total PC for DP and CP

4.6 Summary

This study performs a deep analysis and tries to model the PC of a SDN based IoT system. It decomposes all of the element functions of the SDN-WISE structure, which supports the IoT application and services of WSN operated by the SDN. They work together but according to assorted margins to sustain the high NQoS. The architecture model of the proposed work is based on [27] for profiling a PM for SDN in WSN. The PM is an essential part for both the devices and the network to be managed in flexible manner. Consequently, the proposed PM for measuring the PC strongly plays an essential role in the offered SDN NQoS for meeting the required AQoS of IoT applications. This study casts a mathematical framework in the investigation of the PC model. It is subdivided into two important parts: control signalling and data traffic PC. This methodology and the associated framework is based on the power-efficient 802.15.4 in the wireless part and TCP/IP in the wired access technologies for the IoT over a range of traffic levels. A range of parameters have been used to construct the PC of the IoT applications, this done by introducing the SDN-WISE PC model. The propose evaluation processes of the WSN architecture in the SDN-WISE paradigm could be modified depending on the control plane signalling parameters and achieved data traffic metrics. The results indicate that the effectiveness of the additional control signalling results in a power consumption of 98% from the total PC in the Data Plane traffic and 2% in the Control Plane signalling. This figure can be steered in order to improve the NQoS, such as reducing the TTI of the control signalling in the control plane.

Chapter 6 Conclusions and Future work

6.1 Conclusions

The overall aim of this thesis is to address the issues associated with energy efficiency for WSN based on IoT, which is summarised in Chapter One. The work started with the explanation of fundamentals of research based on the energy efficiency of WSN, with the purpose of prolonging the lifespan of the system. The literature review raised four main issues, as which are follows.

Firstly, the fundamentals of WSN routing protocols are examined and a number of studies that have addressed energy consumption matters based on clustering routing protocols are outlined. Secondly, the basic concepts of sleep mode approaches and classification techniques are discussed along with a review of some studies that have considered modes for enhancing energy efficiency. Thirdly, the essential ideas behind improving network performance are described, focussing on a load balancing strategy to introduce new transmission techniques, while attaining a more balanced system. Then, prior work is based on evolutionary algorithms for solving optimisation problems are considered. The last part of the literature review describes the background and related work using SDN, which is deployed to reduce the complexities and difficulties in managing the current and future IoT networks.

An M2M MLCMS routing protocol is proposed for the WSN model system, which is implemented based on the 6LoWPAN protocol. Moreover, the deployment of sink nodes with an optimal number was introduced based on a number of experiments. The selection procedure of the CHs plays an essential role in improving the lifetime of the system. In the proposed MLCMS model, the CHs election is based on a mathematical function, which considers the most effective parameters. The evaluation has demonstrated that when applying the MLCMS protocol, energy consumption is reduced when compared with the LEACH and M-LEACH algorithms. According to the simulation results, the approach consumes much less energy than M-LEACH and LEACH, with 93% and 147% enhancement of energy efficiency, respectively. This positively affects the system lifetime and stability. By combining 6LoWPAN, the number of packets received by the system increases by 7%, and higher accessibility to the M2M nodes as well as substantial extension of the network are realised. By using 6LoWPAN the IoT paradigm is enabled automatically, as it provides a wise IP. In addition, an adaptive sleep mode scheme has been produced for a more efficient lifetime of M2M. The time period of active and sleep CHs has been considered by adding such a scheme to the MLCMS algorithm. The decision on the length for the active period time for a CH is made according to a mathematical function. This technique doubles the lifetime of the system and increases the reduction of the end to end delay by half, when compared to MLCMS model without the adaptive sleep mode.

Furthermore, the Network Performance (N.P) was improved based on the MLCMS protocol to prolong the lifetime along with load balancing solutions. Allocating the CHs to sinks dynamically is evaluated. In order to provide balanced traffic to the network with high residual energy a practical CH-Sink mapping solution is delivered. That is, the Self-Optimising Cluster Head to Sink Algorithm (SOCHSA) for M2M-WSN is proposed to deploy network resources efficiently. This algorithm maximises N.P by balancing the load across multiple sinks to avoid over-utilisation of the network resources. Increasing the load fairness index and the average residual energy for the whole M2M network is solved as an optimisation problem. The optimal solution with two evolutionary algorithms (GA) and (DPSO) is used to tackle the CH distribution issue. The performances of both GA and DPSO are tested and compared using two benchmark scenarios. MLCMS with the SOCHSA algorithm provides significant network benefit in terms of network lifetime when compared to MLCMS without SOCHSA. Moreover, DPSO converges notably quicker than GA and also outperforms it in large network scenarios. Optimum N.P evolution values of 0.306287 and 0.307731 are achieved in the benchmark problems P1 and P2, respectively, by both two and three sinks for GA and DPSO. In addition, the average residual energy improved by 2% when using three sinks compared to two. In terms of the complexity and convergence, the computational results show that DPSO is better than GA and it is thus, considered best suited to a proactive IoT network. The proposed scheme can fulfil different N.P requirements of M2M traffic by instant traffic identification and rerouting dynamic traffic.

In addition, the SDN solution for WIreless SEnsor networks (SDN-WISE) has been used to evaluate the power consumption model. This is considered to be key for solving the issue of the huge number of devices in the IoT. The services on a massive number of physical devices used wirelessly usually have high power consumption. A power consumption (PC) model for the SDN-WISE system is analysed deeply. All of the element functions for the SDN-WISE system and the effects on the power consumption for the whole system are evaluated. The model empowered the services of WSN and SDN to work together, according to different constraints, while maintaining the stability of the network performance. Furthermore, a mathematical framework is employed to examine the PC for the IoT model based on SDN-WISE. Two important measures are considered in this study, namely control signalling and data traffic with respect to the TTI, throughput, packet size and their effects on NQoS and AQoS from a PC point of view. The Power Model (PM) is proposed to profile PC in SDN-WISE. The range of parameters is based on constructing the PC model in the IoT system to solve important issues when introducing the SDN-WISE PM. The assessment processes of the WSN architecture in the SDN-WISE paradigm is proposed, which can be modified depending on the control plane signalling and data traffic requirements. The PC measurement results indicate the effectiveness of the additional control signalling it. That is, it led to a power consumption of 98% from the total PC as Data Plane traffic and 2% as a Control Plane signalling, which is a proof of its feasibility.

6.2 Future Work

To recap, the proposed model for the M2M WSN system performance is evaluated as the first contribution and this is optimised to arrive at an improved system as the second. Moreover, the PC model was profiled for using an up-to-date paradigm, which is SDN-WISE. From all these achievements, it can be concluded that the results are encouraging. However, they need to be followed up with a concerted development effort to move this mechanism into reality. In this thesis, the work opens up research on different directions.

For short-term future work, the following issues need to be considered. Firstly, regarding the first contribution the network parameters could be extended and implemented as a hardware scheme as well as, the proposed routing protocol algorithm can be investigated by using mobile sinks. . Secondly, the evolutionary algorithms, GA and DPSO, are used in the second contribution for solving the optimisation problem, but other AI optimisation techniques can also be employed as a future work. For instance, ant colony and honeybee protocols can be deployed and compared with the algorithms that have been used.

Furthermore, in the network performance formula we can consider more than two KPIs depending on the requirements of the proposed work for instance , consider the delay as one of the KPI as a third one.

In the third part of the work, the profiling of the PC model is based on SDN-WISE and this model can be implemented in the future for evaluating the power saving approaches in SDN-WISE. This can involve implementing different techniques aimed at avoiding extra power consumption. Moreover, we can improve the SDN-WISE network by involving the optimisation techniques for enhancing the whole network performance.

Furthermore, as the system model is based on the IoT, then the network could be extended by including heterogeneous networks, such as Cloud Radio Access Network (CRAN). Moreover, for long-term future work, experimenting with a real test bed together with a workload from different applications is important for continuously improving M2M WSN based on SDN-WISE in IoT applications.

6.3 Research Impact

Energy waste is a major problem in today’s communication systems. The reasons for this are the lack of efficient protocols and techniques to utilise all available network resources, mainly in relation to the physical hardware. An idle or non-fully utilised device is a reason for energy waste. Another cause of energy waste is the inflexibility to apply techniques that arte aimed to make the device utilisation efficient. The contributions made by this research work can prevent adverse impacts of inefficient energy and uneven load distributions in M2M WSN based on the IoT and improve the overall system performance by maintaining efficient power management of network resources. The schemes proposed in this thesis provide the impact of load balancing and power saving in M2M applications for the IoT. Moreover, the centralised architecture of M2M WSN has been explored to achieve a globally balanced load across all network elements. In sum, the proposed work in this thesis effectively addresses inefficient energy consumption, the load balancing optimisation problem based on a clustering approach and profiling the power consumption when using an SDN-WISE technique with the IoT. This research work contributes to the ongoing efforts towards the IoT with SDN-WISE by offering solutions based on important key network performance indicators.

References

[1] M. A. Razzaque, M. Milojevic-Jevric, A. Palade, and S. Clarke, “Middleware for Internet of Things: A Survey,” *IEEE Internet Things J.*, vol. 3, no. 1, pp. 70–95, Feb. 2016.

[2] P. K. Verma *et al.*, “Machine-to-Machine (M2M) communications: A survey,” *J. Netw. Comput. Appl.*, vol. 66, pp. 83–105, 2016.

[3] I. Park, D. Kim, and D. Har, “MAC Achieving Low Latency and Energy Efficiency in Hierarchical M2M Networks With Clustered Nodes,” *Sensors Journal, IEEE*, vol. 15, no. 3, pp. 1657–1661, Mar. 2015.

[4] A. A. Aziz, Y. A. Sekercioglu, P. Fitzpatrick, and M. Ivanovich, “A Survey on Distributed Topology Control Techniques for Extending the Lifetime of Battery Powered Wireless Sensor Networks,” *Commun. Surv. Tutorials, IEEE*, vol. 15, no. 1, pp. 121–144, 2013.

[5] Y. Cao, T. Jiang, and Z. Han, “A Survey of Emerging M2M Systems: Context, Task, and Objective,” *IEEE Internet Things J.*, vol. 3, no. 6, pp. 1246–1258, Dec. 2016.

[6] F. Ahmed, J. Deng, and O. Tirkkonen, “Self-organizing networks for 5G: Directional cell search in mmW networks,” in *Personal, Indoor, and Mobile*

Radio Communications (PIMRC), 2016 IEEE 27th Annual International Symposium on, 2016, pp. 1–5.

[7] S. Tyagi and N. Kumar, “A systematic review on clustering and routing techniques based upon LEACH protocol for wireless sensor networks,” *J. Netw. Comput. Appl.*, vol. 36, no. 2, pp. 623–645, 2013.

[8] O. Younis, M. Krunz, and S. Ramasubramanian, “Node clustering in wireless sensor networks: Recent developments and deployment challenges,” *IEEE Netw.*, vol. 20, no. 3, pp. 20–25, 2006.

[9] D. Kim *et al.*, “Minimum Data-Latency-Bound k-Sink Placement Problem in Wireless Sensor Networks,” *IEEE/ACM Trans. Netw.*, vol. 19, no. 5, pp. 1344–1353, Oct. 2011.

[10] L. Mottola and G. P. Picco, “MUSTER: Adaptive Energy-Aware Multisink Routing in Wireless Sensor Networks,” *IEEE Trans. Mob. Comput.*, vol. 10, no. 12, pp. 1694–1709, Dec. 2011.

[11] P. Pongle and G. Chavan, “A survey: Attacks on RPL and 6LoWPAN in IoT,” in *Pervasive Computing (ICPC), 2015 International Conference on*, 2015, pp. 1–6.

[12] P. Kuila, S. K. Gupta, and P. K. Jana, “A novel evolutionary approach for load balanced clustering problem for wireless sensor networks,” *Swarm Evol. Comput.*, vol. 12, pp. 48–56, 2013.

[13] E. A. Khalil and A. A. Bara’a, “Energy-aware evolutionary routing protocol for dynamic clustering of wireless sensor networks,” *Swarm Evol. Comput.*, vol. 1, no. 4, pp. 195–203, 2011.

[14] S. Jain and A. K. Saxena, “A survey of load balancing challenges in cloud environment,” in *2016 International Conference System Modeling Advancement in Research Trends (SMART)*, 2016, pp. 291–293.

[15] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, “An application-specific protocol architecture for wireless microsensor networks,” *IEEE Trans. Wirel. Commun.*, vol. 1, no. 4, pp. 660–670, 2002.

[16] S. Bera, S. Misra, S. K. Roy, and M. S. Obaidat, “Soft-WSN: Software-Defined WSN Management System for IoT Applications,” *IEEE Syst. J.*, vol. PP, no. 99, pp. 1–8, 2016.

[17] S. Tyagi and N. Kumar, “Journal of Network and Computer Applications A

systematic review on clustering and routing techniques based upon LEACH protocol for wireless sensor networks," *J. Netw. Comput. Appl.*, vol. 36, no. 2, pp. 623–645, 2013.

[18] A. A. Abbasi, "A survey on clustering algorithms for wireless sensor networks," *Comput. Commun.*, vol. 30, no. 14, pp. 2826–2841, 2007.

[19] I. T. Haque and N. Abu-ghazaleh, "Wireless Software Defined Networking: A Survey and Taxonomy," *IEEE Commun. Surv. Tutorials*, pp. 1–25, 2016.

[20] C. Gray, R. Ayre, K. Hinton, and R. S. Tucker, "Power consumption of IoT access network technologies," in *2015 IEEE International Conference on Communication Workshop (ICCW)*, 2015, pp. 2818–2823.

[21] M. Alam, R. H. Nielsen, and N. R. Prasad, "The evolution of M2M into IoT," in *Communications and Networking (BlackSeaCom), 2013 First International Black Sea Conference on*, 2013, pp. 112–115.

[22] w. twayej, M. Khan, and h. al-raweshidy, "Network Performance Evaluation of M2M With Self Organising Cluster Head to Sink Mapping," *IEEE Sens. J.*, vol. PP, no. 99, p. 1, 2017.

[23] F. Farouk, R. Rizk, and F. W. Zaki, "Multi-level stable and energy-efficient clustering protocol in heterogeneous wireless sensor networks," *Wirel. Sens. Syst. IET*, vol. 4, no. 4, pp. 159–169, 2014.

[24] A. E. A. A. Abdulla, H. Nishiyama, J. Yang, N. Ansari, and N. Kato, "HYMN: A Novel Hybrid Multi-Hop Routing Algorithm to Improve the Longevity of WSNs," *Wirel. Commun. IEEE Trans.*, vol. 11, no. 7, pp. 2531–2541, Jul. 2012.

[25] R. Dutta, S. Gupta, and M. K. Das, "Low-Energy Adaptive Unequal Clustering Protocol Using Fuzzy c-Means in Wireless Sensor Networks," *Wirel. Pers. Commun.*, vol. 79, no. 2, pp. 1187–1209, Jul. 2014.

[26] S. Costanzo, L. Galluccio, G. Morabito, and S. Palazzo, "Software Defined Wireless Networks: Unbridling SDNs," in *2012 European Workshop on Software Defined Networking*, 2012, pp. 1–6.

[27] L. Galluccio, S. Milardo, G. Morabito, and S. Palazzo, "SDN-WISE: Design, prototyping and experimentation of a stateful SDN solution for WIreless SEnsor networks," in *2015 IEEE Conference on Computer Communications (INFOCOM)*, 2015, pp. 513–521.

[28] M. Li, Z. Li, and A. V Vasilakos, "A Survey on Topology Control in Wireless

Sensor Networks: Taxonomy, Comparative Study, and Open Issues," *Proc. IEEE,* vol. 101, no. 12, pp. 2538–2557, Dec. 2013.

[29] L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," *Comput. networks,* vol. 54, no. 15, pp. 2787–2805, 2010.

[30] F. Losilla, C. Vicente-Chicote, B. Álvarez, A. Iborra, and P. Sánchez, "Wireless sensor network application development: An architecture-centric mde approach," in *European Conference on Software Architecture,* 2007, pp. 179–194.

[31] J. N. Al-Karaki and A. E. Kamal, "Routing techniques in wireless sensor networks: a survey," *IEEE Wirel. Commun.*, vol. 11, no. 6, pp. 6–28, 2004.

[32] K. Akkaya and M. Younis, "A survey on routing protocols for wireless sensor networks," *Ad hoc networks,* vol. 3, no. 3, pp. 325–349, 2005.

[33] N. A. Pantazis, S. A. Nikolidakis, and D. D. Vergados, "Energy-Efficient Routing Protocols in Wireless Sensor Networks: A Survey," *IEEE Commun. Surv. Tutorials*, vol. 15, no. 2, pp. 551–591, 2013.

[34] W. R. Heinzelman, J. Kulik, and H. Balakrishnan, "Adaptive protocols for information dissemination in wireless sensor networks," in *Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking*, 1999, pp. 174–185.

[35] D. Bhattacharyya, T. Kim, and S. Pal, "A comparative study of wireless sensor networks and their routing protocols," *Sensors*, vol. 10, no. 12, pp. 10506–10523, 2010.

[36] S. K. Singh, P. Kumar, and J. P. Singh, "A Survey on Successors of LEACH Protocol," *IEEE Access*, vol. 5, pp. 4298–4328, 2017.

[37] O. Younis and S. Fahmy, "HEED: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks," *IEEE Trans. Mob. Comput.*, vol. 3, no. 4, pp. 366–379, Oct. 2004.

[38] A. Ihsan, K. Saghar, and T. Fatima, "Analysis of LEACH protocol(s) using formal verification," in *Applied Sciences and Technology (IBCAST), 2015 12th International Bhurban Conference on*, 2015, pp. 254–262.

[39] N. Kumar and J. Kaur, "Improved LEACH Protocol for Wireless Sensor Networks," in *2011 7th International Conference on Wireless Communications, Networking and Mobile Computing,* 2011, pp. 1–5.

[40] S. Tanessakulwattana, C. Pornavalai, and G. Chakraborty, “Adaptive multi-hop routing for wireless sensor networks,” in *Computer Science and Software Engineering (JCSSE), 2013 10th International Joint Conference on*, 2013, pp. 105–110.

[41] X. Wang, “Multicast for 6LoWPAN Wireless Sensor Networks,” *Sensors Journal, IEEE*, vol. 15, no. 5, pp. 3076–3083, May 2015.

[42] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, “Internet of things: A survey on enabling technologies, protocols, and applications,” *IEEE Commun. Surv. Tutorials*, vol. 17, no. 4, pp. 2347–2376, 2015.

[43] M. R. Palattella *et al.*, “Standardized Protocol Stack for the Internet of (Important) Things,” *Commun. Surv. Tutorials, IEEE*, vol. 15, no. 3, pp. 1389–1406, 2013.

[44] S.-J. Jung and W.-Y. Chung, “Non-Intrusive Healthcare System in Global Machine-to-Machine Networks,” *Sensors Journal, IEEE*, vol. 13, no. 12, pp. 4824–4830, Dec. 2013.

[45] Mh. P. Layer, “Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs),” 2013.

[46] E. Callaway, “Low Power Consumption Features of the IEEE 802.15. 4/ZigBee LR-WPAN Standard.”

[47] G. Anastasi, M. Conti, M. Di Francesco, and A. Passarella, “Energy conservation in wireless sensor networks: A survey,” *Ad hoc networks*, vol. 7, no. 3, pp. 537–568, 2009.

[48] D. Ganesan, A. Cerpa, W. Ye, Y. Yu, J. Zhao, and D. Estrin, “Networking issues in wireless sensor networks,” *J. Parallel Distrib. Comput.*, vol. 64, no. 7, pp. 799–814, 2004.

[49] P. Godfrey and D. Ratajczak, “Naps: scalable, robust topology management in wireless ad hoc networks,” in *Proceedings of the 3rd international symposium on Information processing in sensor networks*, 2004, pp. 443–451.

[50] L. Gu and J. A. Stankovic, “Radio-triggered wake-up for wireless sensor networks,” *Real-Time Syst.*, vol. 29, no. 2, pp. 157–182, 2005.

[51] A. Deshpande and S. Madden, “MauveDB: supporting model-based user views in database systems,” in *Proceedings of the 2006 ACM SIGMOD international*

conference on Management of data, 2006, pp. 73–84.

[52] A. Deshpande, C. Guestrin, S. R. Madden, J. M. Hellerstein, and W. Hong, "Model-driven data acquisition in sensor networks," in *Proceedings of the Thirtieth international conference on Very large data bases-Volume 30*, 2004, pp. 588–599.

[53] T. Van Dam and K. Langendoen, "An adaptive energy-efficient MAC protocol for wireless sensor networks," in *Proceedings of the 1st international conference on Embedded networked sensor systems*, 2003, pp. 171–180.

[54] A. Keshavarzian, H. Lee, and L. Venkatraman, "Wakeup scheduling in wireless sensor networks," in *Proceedings of the 7th ACM international symposium on Mobile ad hoc networking and computing*, 2006, pp. 322–333.

[55] A. A. Babayo, M. H. Anisi, and I. Ali, "A review on energy management schemes in energy harvesting wireless sensor networks," *Renew. Sustain. Energy Rev.*, vol. 76, pp. 1176–1184, 2017.

[56] Y. R. Faizulkhakov, "Time synchronization methods for wireless sensor networks: A survey," *Program. Comput. Softw.*, vol. 33, no. 4, pp. 214–226, 2007.

[57] F. Sivrikaya and B. Yener, "Time synchronization in sensor networks: a survey," *IEEE Netw.*, vol. 18, no. 4, pp. 45–50, 2004.

[58] B. Hohlt, L. Doherty, and E. Brewer, "Flexible power scheduling for sensor networks," in *Proceedings of the 3rd international symposium on Information processing in sensor networks*, 2004, pp. 205–214.

[59] B. Hohlt and E. Brewer, "Network power scheduling for TinyOS applications," in *International Conference on Distributed Computing in Sensor Systems*, 2006, pp. 443–462.

[60] M. D. Vose, *The simple genetic algorithm: foundations and theory*, vol. 12. MIT press, 1999.

[61] J. Kennedy, "Particle swarm optimization," in *Encyclopedia of machine learning*, Springer, 2011, pp. 760–766.

[62] D. Teodorovic, P. Lucic, G. Markovic, and M. D. Orco, "Bee Colony Optimization: Principles and Applications," in *2006 8th Seminar on Neural Network Applications in Electrical Engineering*, 2006, pp. 151–156.

[63] J. Vesterstrom and R. Thomsen, "A comparative study of differential evolution,

particle swarm optimization, and evolutionary algorithms on numerical benchmark problems," in *Evolutionary Computation, 2004. CEC2004. Congress on*, 2004, vol. 2, pp. 1980–1987.

[64] Q. Zhou and R. Liu, "Strategy optimization of resource scheduling based on cluster rendering," *Cluster Comput.*, vol. 19, no. 4, pp. 2109–2117, 2016.

[65] S. Patil, "Cluster performance evaluation using load balancing algorithm," in *Information Communication and Embedded Systems (ICICES), 2013 International Conference on*, 2013, pp. 104–108.

[66] S. Mini, S. K. Udgata, and S. L. Sabat, "Sensor deployment and scheduling for target coverage problem in wireless sensor networks," *IEEE Sens. J.*, vol. 14, no. 3, pp. 636–644, 2014.

[67] C. J. Jiang, W. R. Shi, M. Xiang, and X. L. Tang, "Energy-balanced unequal clustering protocol for wireless sensor networks," *J. China Univ. Posts Telecommun.*, vol. 17, no. 4, pp. 94–99, 2010.

[68] N. M. A. Latiff, C. C. Tsimenidis, B. S. Sharif, and U. Kingdom, "Energy-Aware Clustering for Wireless Sensor Networks Using Particle Swarm Optimization," in *The 18th Annual IEEE International Sysmposium on Personal, Indoor and Mobile Radio Communications (PIMRC'07)*, 2007, pp. 5–9.

[69] J. Yang, H. Zhang, Y. Ling, C. Pan, and W. Sun, "Task allocation for wireless sensor network using modified binary particle swarm optimization," *IEEE Sens. J.*, vol. 14, no. 3, pp. 882–892, 2014.

[70] S. Costanzo, L. Galluccio, G. Morabito, and S. Palazzo, "Software defined wireless networks: Unbridling SDNs," *Proc. - Eur. Work. Softw. Defin. Networks, EWSDN 2012*, pp. 1–6, 2012.

[71] S. Gao, Y. Zeng, H. Luo, and H. Zhang, "Scalable control plane for intra-domain communication in software defined information centric networking," *Futur. Gener. Comput. Syst.*, vol. 56, pp. 110–120, Mar. 2016.

[72] D. B. Rawat and S. R. Reddy, "Software Defined Networking Architecture, Security and Energy Efficiency: A Survey," *IEEE Commun. Surv. Tutorials*, vol. 19, no. 1, pp. 325–346, 2017.

[73] G. Bang and I. S. Kweon, "Multi-target tracking using social force model in discrete-continuous optimisation framework," *Electron. Lett.*, vol. 49, no. 21,

pp. 1331–1333, Oct. 2013.

[74] S. Khan *et al.*, “Software-Defined Network Forensics: Motivation, Potential Locations, Requirements, and Challenges,” *IEEE Netw.*, vol. 30, no. 6, pp. 6–13, Nov. 2016.

[75] T. Xu, D. Gao, P. Dong, H. Zhang, C. H. Foh, and H. C. Chao, “Defending Against New-Flow Attack in SDN-Based Internet of Things,” *IEEE Access*, vol. 5, pp. 3431–3443, 2017.

[76] T. Luo, H. P. Tan, and T. Q. S. Quek, “Sensor OpenFlow: Enabling Software-Defined Wireless Sensor Networks,” *IEEE Commun. Lett.*, vol. 16, no. 11, pp. 1896–1899, Nov. 2012.

[77] P. Ferrari, A. Flammini, and E. Sisinni, “New Architecture for a Wireless Smart Sensor Based on a Software-Defined Radio,” *IEEE Trans. Instrum. Meas.*, vol. 60, no. 6, pp. 2133–2141, Jun. 2011.

[78] D. Zeng, P. Li, S. Guo, T. Miyazaki, J. Hu, and Y. Xiang, “Energy minimization in multi-task software-defined sensor networks,” *IEEE Trans. Comput.*, vol. 64, no. 11, pp. 3128–3139, 2015.

[79] H. Fotouhi, M. Vahabi, A. Ray, and M. Bj�rkman, “SDN-TAP: An SDN-based traffic aware protocol for wireless sensor networks,” in *2016 IEEE 18th International Conference on e-Health Networking, Applications and Services (Healthcom)*, 2016, pp. 1–6.

[80] F. Kaup, S. Melnikowitsch, and D. Hausheer, “Measuring and Modeling the Power Consumption of OpenFlow Switches,” in *10th International Conference on Network and Service Management (CNSM) and Workshop*, 2014, pp. 181–186.

[81] A. K. Tripathy and S. Chinara, “Distributed dynamic clustering protocol for wireless sensor network,” *Int. J. Comput. Appl. Technol.*, vol. 51, no. 2, p. 112, 2015.

[82] P. Chatterjee and N. Das, “Multiple sink deployment in multi-hop wireless sensor networks to enhance lifetime,” in *Applications and Innovations in Mobile Computing (AIMoC), 2015*, 2015, pp. 48–54.

[83] W. Twayej, M. Khan, and H. S. Al-Raweshidy, “Network Performance Evaluation of M2M With Self Organizing Cluster Head to Sink Mapping,” *IEEE Sens. J.*, vol. 17, no. 15, pp. 4962–4974, Aug. 2017.

[84] X. . Asha Shiny and R. Jagadeesh Kannan, “Energy efficient clustering protocol using self organizing map in MANET,” *Indian J. Sci. Technol.*, vol. 8, no. 28, 2015.

[85] C.-Y. Wan, S. B. Eisenman, A. T. Campbell, and J. Crowcroft, “Overload traffic management for sensor networks,” *ACM Trans. Sens. Networks*, vol. 3, no. 4, p. 18–es, 2007.

[86] S. R. Gandham, M. Dawande, R. Prakash, and S. Venkatesan, “Energy efficient schemes for wireless sensor networks with multiple mobile base stations,” in *Global telecommunications conference, 2003. GLOBECOM’03. IEEE*, 2003, vol. 1, pp. 377–381.

[87] M. Bsoul, A. Al-Khasawneh, A. E. Abdallah, E. E. Abdallah, and I. Obeidat, “An energy-efficient threshold-based clustering protocol for wireless sensor networks,” *Wirel. Pers. Commun.*, vol. 70, no. 1, pp. 99–112, 2013.

[88] R. Roberts and P. Goodwin, “Weight approximations in multiattribute decision models.,” *J. Multi-Criteria Decis. Anal.*, vol. 303, no. June, pp. 291–303, 2002.

[89] D. Mahmood, N. Javaid, S. Mahmood, S. Qureshi, A. M. Memon, and T. Zaman, “MODLEACH: A Variant of LEACH for WSNs,” in *Broadband and Wireless Computing, Communication and Applications (BWCCA), 2013 Eighth International Conference on*, 2013, pp. 158–163.

[90] D. Wei, Y. Jin, S. Vural, K. Moessner, and R. Tafazolli, “An energy-efficient clustering solution for wireless sensor networks,” *IEEE Trans. Wirel. Commun.*, vol. 10, no. 11, pp. 3973–3983, 2011.

[91] V. K. Subhashree, C. Tharini, and M. Swarna Lakshmi, “Modified LEACH: A QoS-aware clustering algorithm for Wireless Sensor Networks,” in *Communication and Network Technologies (ICCNT), 2014 International Conference on*, 2014, pp. 119–123.

[92] C.-J. Liao, C.-T. Tseng, and P. Luarn, “A discrete version of particle swarm optimization for flowshop scheduling problems,” *Comput. Oper. Res.*, vol. 34, no. 10, pp. 3099–3111, 2007.

[93] J. C. Bansal, P. K. Singh, M. Saraswat, A. Verma, S. S. Jadon, and A. Abraham, “Inertia weight strategies in particle swarm optimization,” in *Nature and Biologically Inspired Computing (NaBIC), 2011 Third World Congress on*, 2011, pp. 633–640.

[94] E. Elbeltagi, T. Hegazy, and D. Grierson, “Comparison among five evolutionary-based optimization algorithms,” *Adv. Eng. informatics*, vol. 19, no. 1, pp. 43–53, 2005.

[95] A. C. Anadiotis, G. Morabito, and S. Palazzo, “An SDN-assisted Framework for Optimal Deployment of MapReduce Functions in WSNs,” *IEEE Trans. Mob. Comput.*, vol. PP, no. 99, p. 1, 2015.

[96] C. Intanagonwiwat, D. Estrin, R. Govindan, and J. Heidemann, “Impact of network density on data aggregation in wireless sensor networks,” in *Distributed Computing Systems, 2002. Proceedings. 22nd International Conference on*, 2002, pp. 457–458.

[97] P. Di Dio *et al.*, “Exploiting state information to support QoS in Software-Defined WSNs,” in *2016 Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net)*, 2016, pp. 1–7.

[98] R. A. Karam, “Enabling Cisco legacy power to support IEEE 802.3 AF standard power.” Google Patents, 2005.

[99] W. Xiang, N. Wang, and Y. Zhou, “An Energy-Efficient Routing Algorithm for Software-Defined Wireless Sensor Networks,” *IEEE Sens. J.*, vol. 16, no. 20, pp. 7393–7400, 2016.

[100] A. Carroll, G. Heiser, and others, “An Analysis of Power Consumption in a Smartphone.,” in *USENIX annual technical conference*, 2010, vol. 14, p. 21.

[101] V. Sivaraman, A. Vishwanath, Z. Zhao, and C. Russell, “Profiling per-packet and per-byte energy consumption in the NetFPGA Gigabit router,” in *2011 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 2011, pp. 331–336.

[102] Embit, “EMB-Z2530PA ,” Datasheet, Nov. 2006 [Online], [Revised April. 2013], availqble on : http://www.embit.eu/?s=Z2530PA .

[103] Core Part1, http://sdn-wise.dieei.unict.it/docs/guides/CorePart1.html, 2017.

[104] Texas Instruments, “Power Measurement of Etherenet Physical Layer Products ,” AN-1540 Application Report, Nov. 2006 [Revised April. 2013].