Optimised Cloud-Based 6LoWPAN Network Using SDN/NFV Concepts for Energy-Aware IoT Applications

A Thesis Submitted
in Fulfilment of the Requirements for the Degree of
Doctor of Philosophy

By
Bilal R. Al-Kaseem

Supervisor
Professor Hamed Al-Raweshidy

Department of Electronic and Computer Engineering
College of Engineering, Design and Physical Sciences
Brunel University London
London, United Kingdom

July 2017

To my beloved mother,
in memory of my beloved father,
to my wife and my son,
to my sisters and brothers.

Abstract

The Internet of Things (IoT) concept has been realised with the advent of Machine-to-Machine (M2M) communication through which the vision of future Internet has been revolutionised. IPv6 over Low power Wireless Personal Area Networks (6LoWPAN) provides feasible IPv6 connectivity to previously isolated environments, e.g. wireless M2M sensors and actuator networks. This thesis's contributions include a novel mathematical model, energy-efficient algorithms, and a centralised software controller for dynamic consolidation of programmability features in cloud-based M2M networks.

A new generalised joint mathematical model has been proposed for performance analysis of the 6LoWPAN MAC and PHY layers. The proposed model differs from existing analytical models as it precisely adopts the 6LoWPAN specifications introduced by the Internet Engineering Task Force (IETF) working group. The proposed approach is based on Markov chain modelling and validated through Monte-Carlo simulation. In addition, an intelligent mechanism has been proposed for optimal 6LoWPAN MAC layer parameters set selection. The proposed mechanism depends on Artificial Neural Network (ANN), Genetic Algorithm (GA), and Particles Swarm Optimisation (PSO). Simulation results show that utilising the optimal MAC parameters improve the 6LoWPAN network throughput by 52-63% and reduce end-to-end delay by 54-65%.

This thesis focuses on energy-efficient data extraction and dissemination in a wireless M2M sensor network based on 6LoWPAN. A new scalable and self-organised clustering technique with a smart sleep scheduler has been proposed for prolonging M2M network's lifetime and enhancing network connectivity. These solutions succeed in overcoming performance degradation and unbalanced energy consumption problems in homogeneous and heterogeneous sensor networks. Simulation results show that by adopting the proposed schemes in multiple mobile sink sensory field will improve the total aggregated packets by 38-167% and extend network lifetime by 30-78%.

Proof-of-concept real-time hardware testbed experiments are used to verify the effectiveness of Software-Defined Networking (SDN), Network Function Virtualisation (NFV) and cloud computing on a 6LoWPAN network. The implemented testbed is based on open standards development boards (i.e. Arduino), with one sink, which is the M2M 6LoWPAN gateway, where the network coordinator and the customised SDN controller operated. Experimental results indicate that the proposed approach reduces network discovery time by 60% and extends the node lifetime by 65% in comparison with the traditional 6LoWPAN network. Finally, the thesis is concluded with an overall picture of the research conducted and some suggestions for future work.

Acknowledgements

First and foremost, I thank Allah Almighty for giving me the strength and the means to complete my doctoral studies.

PhD journey is a once-in-a-lifetime opportunity and experience, which would not be possible without the guidance, support and motivation from amazing people whom I wish to acknowledge as my journey is near to its end.

I would also like to acknowledge the funding support of the Iraqi Ministry of Higher Education and Scientific Research.

I owe my deepest gratitude to my supervisor, Prof Hamed Al-Raweshidy for giving me the opportunity to pursue my studies in his eminent WNCC group. His continuous support, relentless, insightful guidance, and inspirational mentorship helped me in all aspects of my research. Thank you Sir, for everything you did for me and for always believing in me whilst allowing me the room to work in my own way.

This wonderful and challenging journey, which started in Oct. 2013, would not have been possible without the prayers of my mother. I am heartily thankful to her for the unconditional love, support, and encouragement at all times. It is to her that I owe all the success in life, your prayer for me was what sustained me thus far.

I would like to thank my beloved wife, who has been making my life so incredible and prosperous each and every day. I would like to express my deepest appreciation for her endless support during these past years.

Last, but not the least, I wish to thank my sisters and brothers, who has always stood by me and inspired me during the ups and downs of this pleasant experience.

My appreciation also goes to my dearest friends for sharing their research experiences and insights with me. It is a privilege to have them as colleagues, and a blessing to know them as friends.

Table of Contents

List of Publications

This list contains the published papers that reflect the results achieved during the development of the research presented in this thesis.

- Published Journal Papers
  - **B. R. Al-Kaseem** and H. S. Al-Raweshidy, "SD-NFV as an Energy Efficient Approach for M2M Networks Using Cloud-Based 6LoWPAN Testbed," in *IEEE Internet of Things Journal*, vol. 4, no. 5, pp. 1787-1797, 2017.
  - **B. R. Al-Kaseem**, H. S. Al-Raweshidy, Y. Al-Dunainawi and K. Banitsas, "A New Intelligent Approach for Optimising 6LoWPAN MAC Layer Parameters," in *IEEE Access Journal*, vol. 5, pp. 16229-16240, 2017.
- Peer Reviewed Conference Publications
  - **B. R. Al-Kaseem** and H. S. Al-Raweshidy, "Scalable M2M Routing Protocol for Energy Efficient IoT Wireless Applications," *2016 8th Computer Science and Electronic Engineering (CEEC)*, Colchester, 2016, pp. 30-35.
  - **B. R. Al-Kaseem** and H. S. Al-Raweshidy, "Enabling Wireless Software Defined Networking in Cloud Based Machine-to-Machine Gateway," *2016 8th Computer Science and Electronic Engineering (CEEC)*, Colchester, 2016, pp. 24-29.
  - **B. R. Al-Kaseem**, A. O. Nyanteh and H. S. Al-Raweshidy, "Self-Organized Clustering Technique Based on Sink Mobility in Heterogeneous M2M Sensor Networks," *2016 International Conference for Students on Applied Engineering (ICSAE)*, Newcastle upon Tyne, 2016, pp. 431-436.
  - **B. R. Al-Kaseem** and H. S. Al-Raweshidy, "Energy Efficient MAC Protocol with Smart Sleep Scheduling for Cluster-Based M2M Networks," *2016 6th International Conference on Information Communication and Management (ICICM)*, Hatfield, 2016, pp. 227-232.

List of Figures

List of Tables

List of Abbreviations

Abbreviation	Definition
6LoWPAN	IPv6 over Low power Wireless Personal Area Networks
ABC	Artificial Bee Colony
ADC	Analog-Digital Converter
AES	Advanced Encryption Standard
AND	Application-Defined Networking
ANN	Artificial Neural Network
API	Application Programming Interface
CAP	Contention Access Period
CCA	Clear Channel Assessment
CFP	Contention-Free Period
CH	Cluster Head
CSMA/CA	Carrier Sense Multiple Access / Collision Avoidance
DAC	Duplicate Address Confirmation
DAR	Duplicate Address Request
DEEC	Distributed Energy Efficient Clustering
DSSS	Direct Sequence Spread Spectrum
EA	Evolutionary Algorithm
ETSI	European Telecommunications Standards Institute
FF	Feed-Forward
FFD	Full-Function Device
GA	Genetic Algorithm
GPRS	General Packet Radio Service
GPS	Global Positioning System
GTS	Guarantee Time Slot
GUI	Graphical User Interface
H2H	Human-to-Human
IaaS	Infrastructure as a Service
IDE	Integrated Development Environment
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
IoE	Internet-of-Everything
IoNT	Internet-of-Nano-Things
IoT	Internet-of-Things
IoV	Internet-of-Vehicle
IP	Internet Protocol
IPv6	Internet Protocol version 6
ISO	International Organisation for Standardisation
LEACH	Low Energy Adaptive Cluster Hierarchy
LED	Light Emitting Diode

LLN	Low-Power and Lossy Networks
LM	Levenberg Marquardt Algorithm
LoWPAN	Low power Wireless Personal Area Network
LQI	Link Quality Indication
LR-WPAN	Low Rate - Wireless Personal Area Network
LTE	Long Term Evolution
M2M	Machine-to-Machine
MAC	Medium Access Control
MSE	Mean Squared Error
MTU	Maximum Transmission Unit
NA	Neighbour Advertisement
ND	Node Discovery
NF	Network Function
NFV	Network Function Virtualisation
NUD	Neighbour Unreachability Detection
OSI	Open System Interconnection
OvS	Open vSwitch
PaaS	Platform as a Service
PAN	Personal Area Network
PHY	Physical Layer
pIPv6	pico Internet Protocol version 6
PNF	Physical Network Function
PSO	Particles Swarm Optimisation
Q-OFSK	Quadrature-Orthogonal Phase Shift Keying
QoS	Quality-of-Service
RA	Router Advertisement
RFC	Request for Comments
RFD	Reduced-Function Device
RFID	Radio Frequency Identification
RPL	Routing Protocol for Low-power and Lossy Networks
RSS	Received Signal Strength
SaaS	Software as a Service
SC	Soft-Computing
SDN	Software-Defined Networking
SD-NFV	Software Defined - Network Functioning Virtualisation
SEP	Stable Election Protocol
SFV	Sensor Function Virtualisation
SNR	Signal-to-Noise Ratio
TCP/IP	Transmission Control Protocol/Internet Protocol
TDMA	Time Division Multiple Access
TEEN	Threshold sensitive Energy Efficient sensor Network
UDP	User Datagram Protocol
VNF	Virtual Network Function
Wi-Fi	Wireless Fidelity
WiMAX	WorldWide Interoperability for Microwave Access
WLAN	Wireless Local Area Network
WSN	Wireless Sensor Network

Chapter 1

Introduction

1.1 Machine-to-Machine Communications

Only a few years from now, by 2020, the number of devices connected to the Internet will increase exponentially [1]. The connected devices will be quite diverse in functionality and processing capability, having the ability to sense, actuate, process, and store data. These devices can communicate with each other and exchange information in a Machine-to-Machine (M2M) paradigm [2]. M2M communication refers to the communication between two connected devices in homogeneous or heterogeneous networks without or with limited human intervention. M2M communication pertains the algorithms, mechanisms and technologies that can handle the large diversity of limited energy embedded devices that forward the data to collection base stations. In M2M sensor networks, there are a large number of sensor nodes ranging from hundreds to thousands, which run data sensing tasks and send the information to the M2M gateway. Due to the variety of sensor network heterogeneity, M2M networks provide different levels of reliability and critical data have higher reliability demands [3]. Nowadays, wireless sensors are the main building blocks of M2M communication and are being deployed in various systems, including surveillance systems, smart grids, and healthcare monitoring systems [4]. Traditionally, such connected devices have worked locally and provided services for human users in an independent way. Advances in radio communication technologies have enabled more connectivity, such as Radio Frequency Identification (RFID) and Low power Wireless Personal Area Networks (LoWPAN) to connect these devices to the Internet. In particular, M2M communications act as an enabling technology for the Internet of Things (IoT) [5].

The IoT can be defined as a global network infrastructure that includes the existing and future Internet networks. IoT networks have self-configuring capabilities that make network extension possible with different networks technologies [5]. M2M communication constitutes the principle communication paradigm in realising the IoT revolution. IoT enables physical objects to have virtual identity and will be integrated into a wide range of applications to enhance daily life activities, such as home and industrial automation, energy management, etc. Moreover, cloud computing architectures are the most promising technology in leveraging some of the applications, services, and networks of IoT [6]. The merging of M2M networks in IoT produces new applications for remote measurement and control of the connected devices. Remote measurements refer to the sensing process of physical phenomena then storing and sending the data to extract the useful information. In addition, the remote control of devices includes processing the received data and sending or receiving control commands among the connected devices.

The Institute of Electrical and Electronics Engineers standard (IEEE 802.15.4) is aimed at providing cheap, low-power, short-range communications for embedded devices. The low-power and low-data rate devices provide ubiquitous connectivity at low cost and they can access the Internet via a border gateway. The emerging of Internet Protocol (IP)-connected devices, like sensors, actuators, and smart objects are having a great impact in terms of providing a new interconnected service. M2M sensor networks comprise mainly a large number of small devices that run on batteries. The limited battery power of an M2M node is consumed during its lifetime performing the sensing and transmitting of data. Given the limited energy source, there is a need for energy-efficient algorithms to balance the energy consumption and the quality of information, because the node's lifetime depends on the availability of the residual energy [7].

The fundamental differences between M2M and IoT regarding the networking and computing perspective were investigated in [8]. An innovative IoT architecture was proposed in [9] for real-time interaction between the mobile clients and smart/legacy things (sensors and actuators) via a wireless gateway. Also, an IoT overview and enabling technologies, protocols, and application issues were summarised in [5]. IoT is enabled by merging the development of different technologies like intelligent sensors and ubiquitous connectivity. The availability of smart sensors that can interact with the human to deliver new services will open a new plethora of potential applications. In [10], a comprehensive state-of-the-art for M2M networks was provided, which also covered the standardisation efforts for the development of new protocols.

1.1.1 M2M Architecture

In recent years, the European Telecommunications Standards Institute (ETSI), an independent, not-for-profit, standardisation organisation, has become involved in developing a generic standard for M2M systems. It aims to provide interoperability between different M2M components and the available existing technologies, in addition to providing a framework for developing services independently of the underlying network [11]. The ETSI has proposed a detailed functional architecture with a generic set of service capabilities for M2M communication, by dividing the system into three domains: (i) the device and gateway domain; (ii) the network domain; and (iii) the application domain.

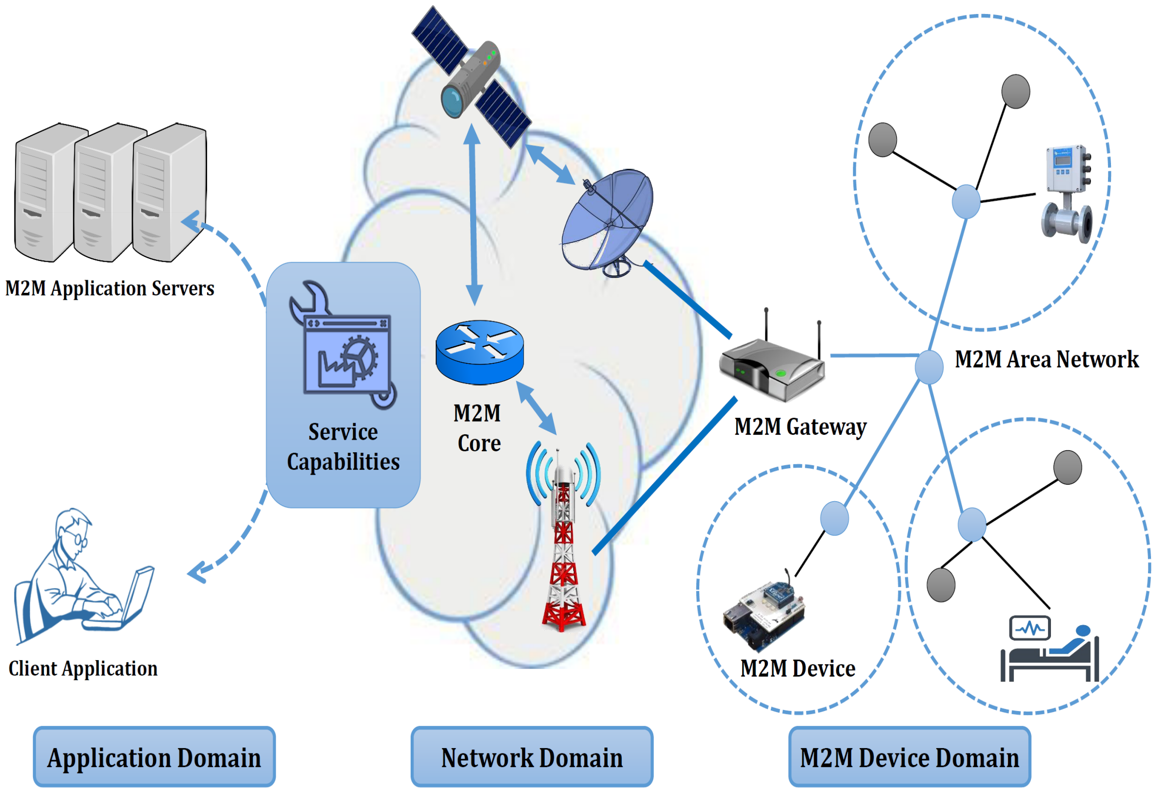


Figure 1.1: M2M communication architecture [11]

Figure 1.1 shows the architecture of a wireless M2M system and the ETSI architecture consists of five distinct components, as described below:

1. The M2M devices: characterised by low-power and low-data rate small embedded devices, which are capable of transmitting data autonomously. Each device typically consists of four principle parts: sensing unit, processing unit, power unit and communication unit. The communication unit enables the M2M device to interact with the other devices through short-range wireless communication.

2. The M2M area network: also known as capillary network, is a short-range network that connects M2M devices and provides a link to the M2M gateway. The M2M area network carries the sensed data by M2M devices to a M2M gateway for further processing and extracts useful information. The IEEE 802.15.4 is an emerging communication standard developed specifically for LoWPAN and the other compliant technologies for the capillary network include: IPv6 over Low power Wireless Personal Area Networks (6LoWPAN), ZigBee, and Bluetooth.
3. The M2M gateway: acts as a proxy between M2M devices and the M2M network domain. The gateway must be an infinite power device with advanced processing capabilities and multiple radio interfaces. These multiple radios enable the M2M gateway to operate in technologies employed by both the M2M area network and the communication network used in the M2M network domain.
4. The M2M access communication network: connects the M2M gateway to the Internet, and also provides access to the M2M application server. The communication network provides long-range communication technologies to gain access the core network, which include Long-Term Evolution (LTE), Worldwide Interoperability for Microwave Access (WiMAX), and Wireless Local Area network (WLAN), while the core network technologies provide IP connectivity.
5. M2M applications: are located on remote servers containing the service middleware layer, where the sensed data travel through various application services to extract useful information for specific business applications.

1.1.2 M2M Communication Features

M2M communication has unique features that distinguished it from conventional Human-to-Human (H2H) communication and creates new service management [12]. M2M communication captures a number of unique features that are highlighted below:

- Massive Number: the number of connected devices is expected to be more than that of the population. The embedded devices will exceed the amount of human interactive devices (e.g. tablets, mobile phones, etc.). These massive devices will generate large volumes of data, which may cause a scalability problem for the existing network infrastructures.
- Energy Constrained: M2M devices must efficiently use their energy storage in order to prolong the network lifetime, because some devices are battery powered and need to run for a long time without charging or changing the batteries. Accordingly,

this goal can be achieved by putting their communication antennae in sleep mode to conserve energy, when their operation is not necessary.

- Lossy Transmission: the M2M devices are deployed in an environment where the wireless links are subject to frequent failures and the reliability of transmitted data is as important as energy efficiency in M2M communication.
- Device Heterogeneity: a large variety of connected M2M devices with diverse functionality and services needs many M2M applications to handle the system requirements. The device heterogeneity creates interoperability issues, which affect the generalisation of M2M systems.
- Limited Mobility: most M2M devices are stationary and deployed in the monitoring area without any mobility. Some M2M devices have a constrained mobility function, having to move in a predefined path in the monitoring area.
- Small Packet Size: most M2M devices generate periodic traffic, with few applications generating event-driven traffic using the IEEE 802.15.4 standard. Hence, M2M devices must use small data transmissions for efficient energy resource utilisation.
- Quality-of-Service (QoS) Requirements: most M2M area network consists of many M2M devices transmitting data using low-power and low-data rate communication. However, applications using high-data rate also exist and hence, the end-to-end delay requirements may range from milliseconds to minutes depending on the application.

1.2 Research Motivations

IoT networks are built-up of a large number of M2M nodes, with the main objectives being to have energy-efficient and scalable routing protocols to prolong the lifetime of the connected hardware by converting the consumed energy into useful data transmissions. 6LoWPAN, introduced by the Internet Engineering Task Force (IETF) working group, is defined the implementation of Internet protocols over low-power and low-data rate devices using the IEEE 802.15.4 as the Medium Access Control (MAC) and Physical (PHY) layers standard. 6LoWPAN enables constrained low-power and existing IP devices to communicate with each other over the IEEE 802.15.4 standard. However, 6LoWPAN networks still have constrained properties, including limited residual energy, fixed packet size, high packet loss, limited mobility, and limited throughput. In addition, 6LoWPAN has an extra layer in its protocol stack, which is called the adaptation layer. This layer is responsible for header compression, fragmentation and reassembly of an IPv6 packet when it is sent or received over the IEEE 802.15.4 standard.

M2M sensor networks may adopt unexpected topology changes, where M2M nodes are being deployed randomly in the target area. In order to improve the availability of shared resources, M2M sensor nodes should have the ability of node reconfiguration even after deployment. Software-Defined Networking (SDN) was proposed to separate the control plane from the data plane. This feature provides strong capability to work within the resource constraints of low-power, low-memory, and low-bandwidth devices. SDN can enable sensor node re-tasking in M2M networks and it also provides seamless resource management for implementing different algorithms through the centralised controller. SDN and Network Function Virtualisation (NFV) represent the most promising advances in terms of a programmable network and dynamic resource allocation for M2M networks in IoT architecture. NFV can be applied to any packet processing plane (data plane) and route decision plane (control plane) of M2M network infrastructure, while cloud computing provides ubiquitous connectivity to enhance network functionality.

1.3 Research Aim and Objectives

This research was driven by the motivation to model, optimise, develop, design, and implement a framework of integrated M2M sensor nodes to the Internet. The main aim of this research is to implement self-organised, scalable, and energy-efficient routing protocol for M2M sensor networks that can overcome the resource constrained challenges. Hence, minimising packet losses and energy consumption will enhance network throughput, and reduce end-to-end delay in sensor networks. 6LoWPAN switches the IEEE 802.15.4 standard into the next IP-enabled link by enabling low-power devices and existing IP devices to communicate directly. The IEEE 802.15.4 MAC can be run in two modes: beacon and non-beacon enabled modes. This work is focused on evaluation and development of new MAC layer solutions entirely based on IEEE 802.15.4 non-beacon enabled operation. To this end, this research focuses on implementing a customised SDN controller for cloud-based 6LoWPAN network with integrated NFV technology. The expected outcome of this effort is to enhance network programmability and re-tasking via the proposed customised Software Defined-Network Functioning Virtualisation (SD-NFV) approach for cloud-based 6LoWPAN network. The proposed approach can provide dynamic and scalable deployment for both M2M sensor nodes and application in heterogeneous IoT networks. The SD-NFV approach targets to balance the trade-off between quality of information and energy efficiency while ensuring its applicability for a wide variety of applications in IoT environment.

To reach the aforementioned primary research aim, a set of scientific and technical objectives have been identified. The objectives of the conducted research are:

1. Investigate the relevant state-of-the-art and related works to obtain a solid background about the research topic. The research gaps are identified regarding the modelling of IEEE 802.15.4 protocol stack layers, developing a low-cost M2M sensor node, proposing an energy-efficient routing protocol and deploying multiple mobile sinks, instead of static ones in 6LoWPAN based sensor network;
2. Explore the existing research efforts for integrating SDN and NFV in 6LoWPAN networks. Summarise the expected achievements in terms of sensor nodes' energy consumption, packet delivery ratio and end-to-end delay;
3. Integrate the cloud computing platform with the M2M sensor network through the M2M gateway in 6LoWPAN networks. This integration provides global connectivity and increases network monitoring and management via an SDN-based gateway;
4. Design a new self-organised clustering technique and smart sleep scheduling mechanism for an energy-efficient routing protocol. The proposed approach involves adopting multiple mobile sinks deployment to balance the traffic load and to conserve nodes' energy in IoT homogeneous and heterogeneous M2M sensor network;
5. Develop a new framework capable of providing optimised, self-organised, scalable, and energy-efficient features for IoT networks. The proposed approach must be:
   (a) Independent of the network topology, heterogeneity and routing algorithms;
   (b) Independent of traffic patterns (unidirectional, bidirectional or both);
   (c) Independent of the underlying node software and hardware platform;
6. Develop a proof-of-concept testbed (demonstration system) to validate and test the desired research objectives.

1.4 Thesis Contributions

The contributions of this thesis cover different aspects of M2M communications and the possible solution for integrating SDN, NFV and cloud computing. The contributions to knowledge comprise five parts, while the outcomes in the form novel solutions, algorithms and protocol enhancements are summarised below:

1. Developing a New Joint Mathematical Model for 6LoWPAN MAC and PHY Layers
   The IEEE 802.15.4 standard has been recognised as the most commonly applied MAC and PHY layers standard for Low Rate-Wireless Personal Area Network (LR-WPAN).

6LoWPAN has attracted lots of interest from the research community, because it seems to be one of the efficient ways to build new applications for IoT. This contribution introduces a new analytical model for performance analysis of 6LoWPAN MAC and PHY layers based on Markov chain modelling and queue theory. This approach models the stochastic behaviour of the network throughput and packet delay in terms of MAC parameters. The mathematical model is validated through MATLAB simulation and is proven to be accurate with low computational complexity.

2. Developing an Intelligent Approach for Optimising 6LoWPAN MAC Layer
Fairness, low latency and high throughput with low energy consumption, are the desirable attributes of MAC protocols. When inappropriate parameter settings are used, the default MAC parameters generate excessive collisions, packet losses and high latency under high traffic with a larger number of nodes. Attention is paid for optimising these parameters to achieve high throughput with minimum latency. This contribution involves proposing an optimisation mechanism to select the optimal 6LoWPAN MAC layer parameters set. This mechanism depends on Artificial Neural Networks (ANN), Genetic Algorithm (GA), and Particles Swarm Optimisation (PSO) to select and validate the optimised MAC parameters. The obtained results show that utilising the optimal MAC parameters improve 6LoWPAN network throughput by 52-63% and reduces end-to-end delay by 54-65%, with this percentage depending upon the number of M2M nodes.

3. Developing a Scalable M2M Routing Protocol for Energy-Efficient IoT Applications
The aim of this contribution is to design a self-organised and scalable routing protocol for M2M sensor network. The proposed approach involves adopting multiple mobile sinks for IoT applications based on 6LoWPAN. A self-organised energy-efficient clustering mechanism is proposed implicitly to solve network scalability, performance degradation, and unbalance energy consumption problems for both homogeneous and heterogeneous sensor networks. The proposed clustering technique will use a heuristic approach for cluster head selection and cluster head rotation, whilst also decreasing the control messages and saving energy. The sink nodes dynamically collect the data from cluster heads and eliminate the problems of energy holes, network coverage and multi-hop packets sending. In addition, deploying multiple mobile sinks collaborate in solving network scalability problem and creating a dynamic topology to adapt variable extensions of the system in IoT architecture. The simulation results showed that using multiple mobile sink nodes and the proposed clustering technique improves the energy distribution and pro-

longs network lifetime. More specifically, the percentage difference between the proposed energy-efficient routing protocol and existing routing protocols showed remarkable enhancement of about 74% and 167% in terms of network lifetime and packets delivered to the sink nodes, respectively, in homogeneous sensor networks. On the other hand, the network lifetime expansion for heterogeneous sensor networks ranged from 41% to 78%, while the packets delivered to the sinks ranged from 38% to 165%, the enhancement range depending on network heterogeneity.

4. Developing an Energy-Efficient MAC Protocol with Smart Sleep Scheduling
   This contribution is based on the clustering mechanism proposed in (3) with smart sleep mode to extend M2M network lifetime and enhance network connectivity. In order to improve the energy utilisation in each M2M node effectively, sink mobility has been adopted as well as implicitly providing load balancing and reducing end-to-end delay. The smart sleep scheduling is based on Time Division Multiple Access (TDMA) and has been simulated using MATLAB. The simulation results show that the smart sleep scheduling mechanism succeeds in extending the network lifetime by 67.8% in homogeneous networks. While in heterogeneous networks, the network lifetime is extended by 68.9% and 66.5% for two and three heterogeneity levels, respectively.

5. Implementing a Proof-of-Concept Testbed based on SD-NFV Approach
   To the best of my knowledge, there has been limited research implementing SDN and NFV using existing 6LoWPAN hardware in the literature. To this end, this contribution focuses on implementing a customised SDN controller for a 6LoWPAN network with integrated NFV and cloud technologies. The implemented testbed is based on open source software and hardware platforms. It shows a remarkable reduction in the consumed energy of M2M sensor nodes compared to a traditional 6LoWPAN network. The proposed SD-NFV infrastructure provides a dynamic and scalable deployment for both M2M sensor nodes and applications in heterogeneous M2M networks. The aim of the SD-NFV approach is to simplify network management through a programmability feature and to provide global connectivity for IoT architecture.

1.5 Research Methodology

The connotation development stage was the first stage in this research and it consisted of an inclusive literature review. Research gap identification was carried out by compartmentalisation of the existing energy-efficient routing protocols. In addition, a detailed analysis was performed to determine how the clustering techniques and smart sleep scheduling can be combined together to enhance the performance of the routing protocol without affecting the latency and network performance. Moreover, the existing enhancement schemes are presented in the form of a timeline to ease the analysis and to determine the research gaps during the last years. During the second stage, the evaluation of existing clustering techniques, sleep mechanisms and routing protocols, as well as the design of the proposed clustering and sleep scheduling were performed. As part of the second stage, different simulation software was studied to validate the availability of libraries and to check the compatibility with the designed approach. Subsequently, the simulation environment was selected carefully for future evaluation of the proposed approach. This research is mainly focused on the IPv6 M2M sensor network (6LoWPAN), which plays an important role in many applications. The applications diversity will have different parameters, such as various traffic pattern, different addressing space and variable packet size. These parameters needed to be taken in account in the design considerations as well as when considering the advantages and reasons behind choosing this protocol stack and compliant technologies.

Most researchers in the literature tried to model the MAC and PHY layers of the IEEE 802.15.4 standard in both its operation modes: beacon and non-beacon. However, none has attempted to model these layers according to the 6LoWPAN standard provided by the IETF working group [13]. Accordingly, the third stage was started, with the proposed analytical modelling being carried out using Markov chain analysis, taking into account the operation frequency, data rate and modulation type in the PHY layer as well as the buffer in the M2M node. After analysing and validating the proposed mathematical model, the optimisation phase started as part of the third stage. ANN, GA and PSO were used to optimise the MAC layer parameter to enhance the network performance in terms of network throughput and end-to-end delay. The fourth stage of this research was focused on the integration of SDN, NFV and cloud computing with a 6LoWPAN based network, where the SDN was used to decouple the control and data planes from network devices to increase network programmability. Whilst the NFV was used to virtualise some network functions and to integrate it with the centralised SDN controller, NFV enhances the network functions and eases the development of new service with enhanced management. The main advantage of using cloud computing is that it adds ubiquitous connectivity to engage LoWPANs with the IoT environment.

The proof-of-concept testbed implementation, based on open source hardware platform (e.g. Arduino board). The availability of open source software libraries led to a significant delay in the implementation phase of the 6LoWPAN protocol stack and customised SDN controller. As a final stage, various scenarios were evaluated and validated through both simulation and testbed experiments under various traffic parameters. This provided valuable information about the M2M network behaviour in the IoT environment. As part of this stage a comparison of the traditional and proposed approaches was conducted.

1.6 Thesis Outline

This thesis has begun with an introduction to M2M communication and the enabling technologies required as active parts of IoT. The rest of the thesis is organised as follows:

Chapter 2 gives an overview of the backgrounds and related works regarding the research topics to identify the research gaps in the existing research. Furthermore, existing energy-efficient routing protocols are characterised in terms of clustering mechanism and network heterogeneity. The literature regarding 6LoWPAN networks, sink mobility and most demanding network technologies (e.g. SDN and NFV) with cloud computing is discussed. The technologies critically assessed in this chapter guide the direction of this research.

Chapter 3 presents a detailed overview of the proposed joint mathematical model. At the beginning, an introduction to the IEEE 802.15.4 standard is given to illustrate the operational mode of the 6LoWPAN standard. Furthermore, this chapter provides a detailed analysis of the MAC and PHY layer parameters, in addition to the optimisation techniques that are used to get the best MAC layer parameters set among the default ones.

Chapter 4 presents a detailed analysis and results of how the various clustering techniques are affected by the unique characteristics of 6LoWPAN networks, and how the proposed self-organised clustering technique with smart sleep scheduling succeeds in prolonging the network lifetime without affecting network performance. In addition, network heterogeneity is also considered in the simulation to generalise the proposed clustering technique.

Chapter 5 presents a new SD-NFV approach for 6LoWPAN networks with cloud computing connectivity. SD-NFV is designed as an energy-efficient approach for IoT networks. All of the specific characteristics of 6LoWPAN, SDN, NFV and cloud computing are described in order to study the effects of these characteristics in the proposed scheme. In addition, this chapter presents the hardware and software tools used in implementing the experimental testbed. Moreover, the customised SDN controller implementation is also explained in detail. Finally, the testbed results and discussions are presented at the end of the chapter.

Chapter 6 concludes the thesis, providing the overall picture of the research conducted pertaining to different aspects of M2M communications. It provides a summary of how the objectives have been achieved and what conclusions can be drawn. Moreover, it gives some suggestions of directions for future work.

Chapter 2

Preliminaries and Related Works

2.1 IPv6 over IEEE 802.15.4 Standard (6LoWPAN)

In 2003, the first version was released of the IEEE 802.15.4 standard [14] and was then revised in 2006 [15]. This standard defines the radio communication at 868 MHz, 915 MHz and 2.4 GHz for low-power and low-data rate wireless embedded devices. Practically, IEEE 802.15.4 at 2.4 GHz is used in almost all low-power devices, as it provides a feasible data rate and can be used globally. Depending on operational frequency of the IEEE 802.15.4 standard, different data rates are provided ranging from 20-250 kbps, while the channel access mechanism operates using Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA). In CSMA/CA, the node with a packet to transmit tries to listen to the communication channel to check whether there is another node transmitting a packet within its communication range. If this is true, the node waits until the current transmission is complete. If the medium becomes free and no traffic is generated, the node will start transmitting a packet, otherwise, the node has to wait until the medium becomes clear, as it is being used by another node. In addition, CSMA/CA with acknowledgement is provided to increase the reliability at the MAC layer and the128-bit Advanced Encryption Standard (AES) is provided at the link layer. Unicast and broadcast addressing capability are provided with two addressing modes: short (16-bit) and long (64-bit). The MAC layer runs in one of two modes: beacon-enabled and beaconless modes. The beacon-enabled mode uses Time Division Multiple Access (TDMA) for allocating time slots to each node, while the beaconless mode uses the pure CSMA mechanism. Finally, the PHY layer payload is up to 127 bytes, and it uses different types of modulation depending on the operational radio frequency.

The Internet can be divided into three categories [16]: the core Internet, the edge Internet and the wireless embedded Internet. The core internet pertains to the servers, router and the connected network devices, while the edge Internet is related to the user's units such as PCs, cell phones, etc. The last category is not very large-scaled and is the wireless embedded Internet. It refers to the smart objects and small embedded devices with limited computation capability and that are power constrained. The wireless embedded Internet is built-up by many small stub networks. 6LoWPAN is one of these networks, where the border router or gateway shares the same IPv6 address prefix with all the connected nodes. The border router can be connected to the Internet and be responsible of routing the traffic from and to the 6LoWPAN nodes. 6LoWPAN networks can also exist without a border router connected to the Internet, as so-called Ad-hoc LoWPAN, which is outside the scope of this thesis.

As stated earlier, 6LoWPAN was developed by the IETF working group in 2007 and IPv6 is the newest version of the Internet protocol, with 6LoWPAN providing direct integration of IPv6 over the IEEE 802.15.4 standard. Accordingly, each node in the 6LoWPAN network becomes accessible from the Internet. 6LoWPAN has been designed with the IEEE 802.15.4 in mind and consequently, some 6LoWPAN specifications are closely tied to features of the IEEE 802.15.4 link layer, such as using the Personal Area Network Identification (PAN ID) for address management. 6LoWPAN networks run in the beaconless mode of the CSMA/CA via IEEE 802.15.4 channel, which is called un-slotted by the IEEE 802.15.4 standard. According to the 6LoWPAN specifications provided in [13], the acknowledgements are recommended in order to enhance the network reliability and recover the lost frames during transmissions at the link layer. The IEEE 802.15.4 standard determines the maximum number of retries allowed to resend an unacknowledged frame, which ranges between 0 and 7 that defaults to 3.

Due to M2M nodes constraints, supporting IPv6 to these heavily constrained devices poses several challenges: IPv6 datagrams do not directly fit to LoWPAN, limited buffer size in M2M devices, and energy efficiency requirements. The minimum Maximum Transmission Unit (MTU) required for IPv6 is 1,280 bytes, whereas the IEEE 802.15.4 link layer frame is only 127 bytes long which is one-tenth of IPv6 frame. Accordingly, data fragmentation and compression are needed. 6LoWPAN defines an intermediary adaptation layer between the IEEE 802.15.4 MAC layer and the IPv6 layer for compressing the IPv6 header, performing fragmentation and assembly of an IPv6 packet when it is sent or received over IEEE 802.15.4, respectively as well as providing seamless integration with the existing Internet network [17].

2.2 6LoWPAN Architecture

The 6LoWPAN network consists of many embedded wireless devices that are characterised by power constrained, low-cost, low-data rate with limited memory. The 6LoWPAN architecture is depicted in Figure 2.1 in which the end-to-end communication for interconnecting LoWPANs to the Internet is illustrated. Each LoWPAN is an IPv6 stub network on the Internet, because the IP packets can be received from or sent to it, but they cannot be a packet transit to other Internet networks.

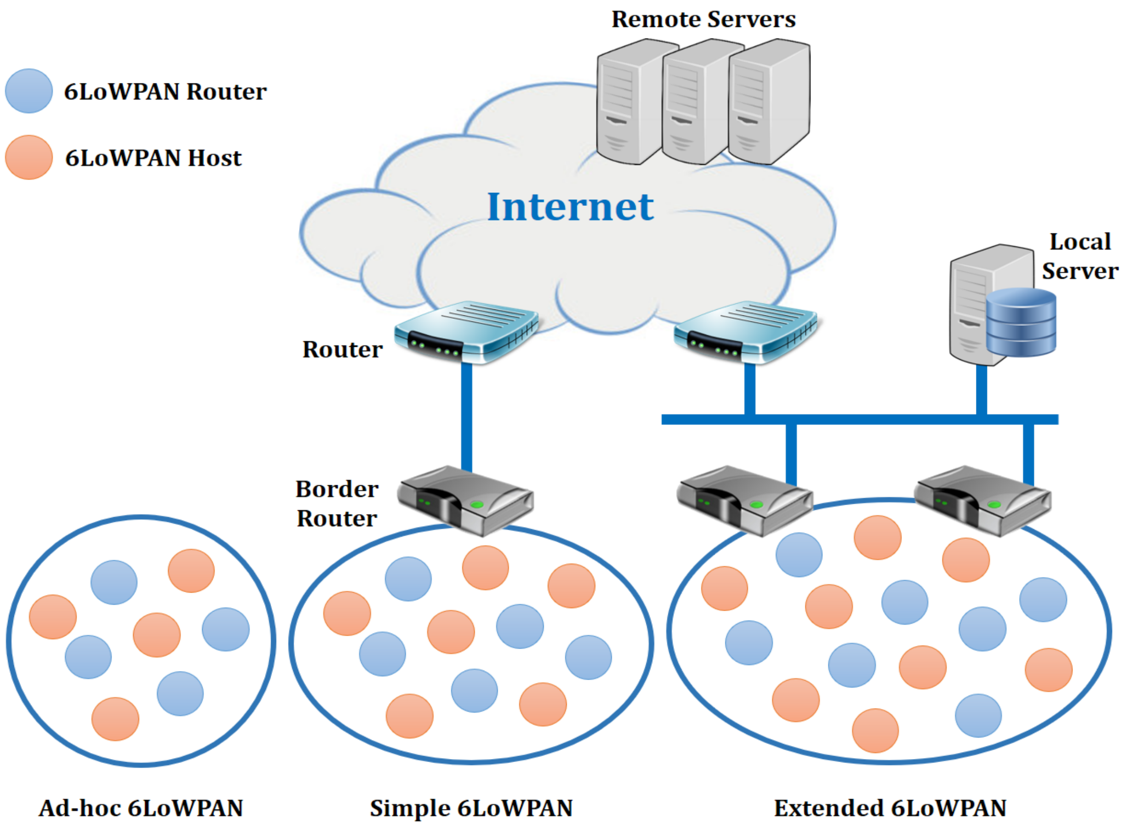


Figure 2.1: The 6LoWPAN architecture [16]

As shown in Figure 2.1, the architecture defines three types of LoWPANs, where each consists of multiple 6LoWPAN nodes with role and function: Ad-Hoc LoWPAN, simple LoWPAN and extended LoWPAN. The Ad-Hoc LoWPAN is recognised as being infrastructureless and functions autonomously without being connected to the Internet. The simple and the extended LoWPANs are also infrastructureless. However, the simple LoWPAN is connected to another IP network through one LoWPAN edge router, while the extended LoWPAN has multiple edge routers to connect LoWPAN to the external IP networks. These multiple edge routers of extended LoWPAN are connected together through a backbone link. Typical 6LoWPAN consists of many nodes with one or more edge router, with the nodes using the same active link to communicate with each other and send data to the IP network via the edge router. In the scenario of one or multiple edge routers, the LoWPAN may be connected to the external IP networks through one or more dedicated links, such as Ethernet, Wireless-Fidelity (Wi-Fi) or General Packet Radio Service (GPRS) communications [16] [18].

2.2.1 Characteristics of 6LoWPAN Devices

According to 6LoWPAN Request For Comments (RFC) 4944 [13], there are two types of devices participating in these networks based on the IEEE 802.15.4 standard: Full-Function Device (FFD) and Reduced-Function Device (RFD). The FFD is capable of implementing full 6LoWPAN protocol stack functions, which represent the ability to become a Personal Area Network (PAN) coordinator. The PAN coordinator is responsible for initialising and establishing the 6LoWPAN network, and responding to the associated devices requests to join the network. Alternatively, it is possible to be a coordinator only, which has the same PAN functions except for the network initiation function. The FFD can communicate with other FFDs and RFDs in the 6LoWPAN network, and makes the implementation of multi-hop routing protocol possible by the upper layer of the protocol stack. The FFDs are able to run complicated applications to perform network maintenance and management. On the other hand, the RFD is only capable of implementing the basic functions of the 6LoWPAN protocol, being unable to initiate and establish the 6LoWPAN network. Moreover, the RFD can only communicate with the FFDs due to the limited functionality of network management. Accordingly, the RFDs are inappropriate for collaborating in complex network services, such as network synchronisation. RFDs are only capable of running simple applications compared to FFDs, which can be used to perform straightforward tasks.

According to these two types of 6LoWPAN devices, the 6LoWPAN network needs to take into account the most important design features, which are unique to the low-power and low-data rate embedded devices, these features being [19] [20]:

i. Power Consumption: some FFDs in a 6LoWPAN network are mains-powered, such as the PAN coordinator, while the majority are either battery-powered FFDs or RFDs and need to run for a long time without battery replacement;

ii. Performance Regression: the limited memory size and moderate processor of these embedded devices need to be utilised efficiently in order to enhance the packet losses and low-bandwidth. As a result, the node lifetime will be extended;

Any FFD or RFD running the 6LoWPAN protocol stack can be setup to one of the following two roles during its operation in a 6LoWPAN network [21]:

1. A 6LoWPAN Border Router (6LBR) is an FFD, which represents the PAN coordinator of the 6LoWPAN network and implements the full protocol stack. The 6LBR is responsible for authenticating the joined nodes and keeps track of the whole network topology. The 6LBR is located between the 6LoWPAN network and the IPv6 network performing data routing to and from the 6LoWPAN network;
2. A 6LoWPAN Node (6LN) is any router or host participating in 6LoWPAN network. The term is generally used to refer to any node that plays the assigned role:
   a. A 6LoWPAN Router (6LR) is an FFD used to route the data inside the 6LoWPAN network by participating in multi-hop packet routing. This role is necessary in 6LoWPAN topology construction;
   b. A 6LoWPAN Host (6LH) is an RFD without data routing capability, which depends on 6LR for packet routing. It is used for the sensing and actuating roles and it utilises sleep periods to conserve energy.

2.2.2 6LoWPAN Topologies

According to the different types of 6LoWPAN devices and the role played by each of them, different topologies will exist, ranging from simple to complex. The LoWPAN may be built-up of just RFDs with the presence of at least one FFD to work as a PAN coordinator. Two types of topologies are supported by 6LoWPAN specifications: star and peer-to-peer topologies [22]:

1. Star Topology
   In star topology, a unique centralised FFD is operated as a PAN coordinator, where the whole star topology is being controlled and managed as depicted in Figure 2.2. Other 6LoWPAN network nodes should associate themselves with the PAN coordinator in order to join the network. Each 6LoWPAN node (FFD or RFD) joining the centralised PAN coordinator and prepared to communicate with other 6LoWPAN nodes should transmit its packets to the 6LoWPAN coordinator prior to forwarding them to the appropriate destination.
2. Peer-to-Peer Topology
   The peer-to-peer topology is also initiated and controlled by the PAN coordinator. Communication between any 6LoWPAN node is possible as long as they are within the effective communication range of each other.

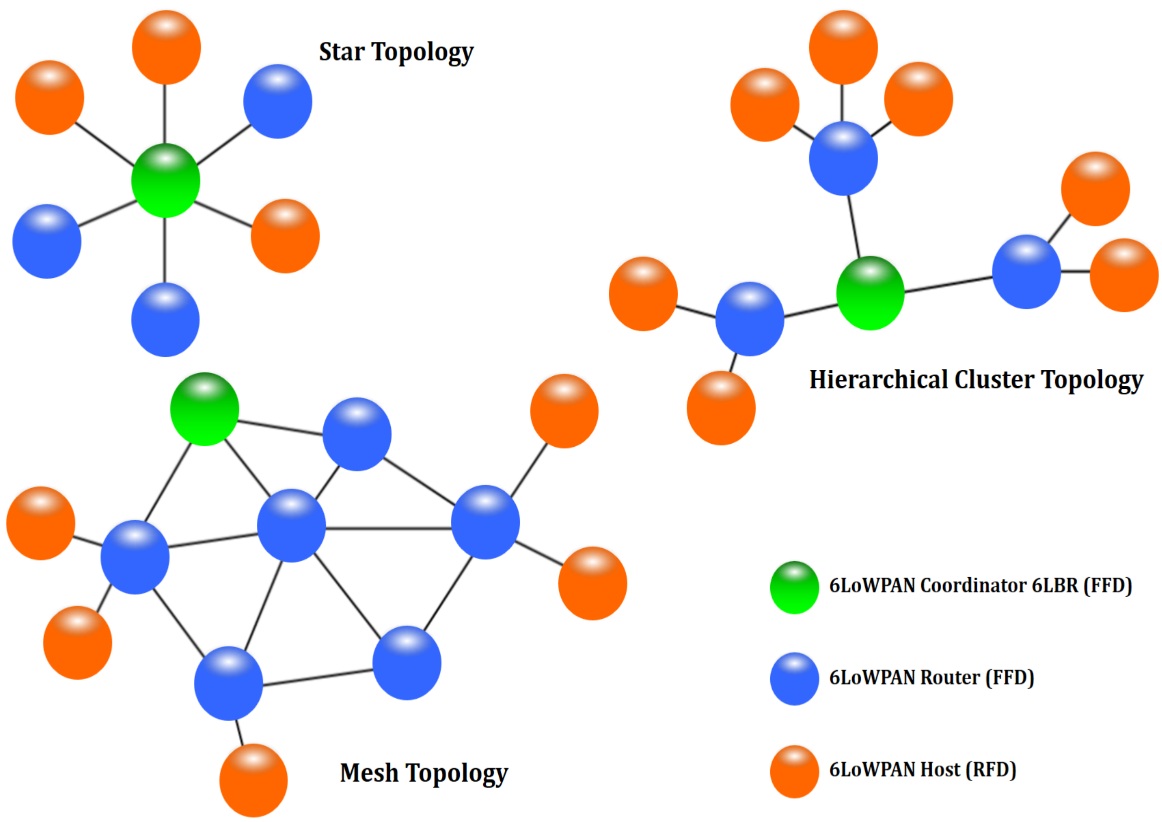


Figure 2.2: 6LoWPAN network topologies

The communication in peer-to-peer topology is not restricted through the coordinator and therefore, any FFD can communicate with other FFDs and RDFs. Also, RFDs can communicate with other FFDs, but they cannot directly communicate with other RFDs. The peer-to-peer topology enhances network scalability, but at the cost of increased node energy consumption. It can be further sub-divided into two types: cluster and mesh topologies:

A. Hierarchical Cluster Topology
   Cluster topology can be formed by single or multiple clusters, with a single cluster topology consisting of one Cluster Head (CH) and all the cluster members are connected to the CH with one hop, whilst the network topology becomes similar to star topology. This is unlike multiple cluster topology, where the network consists of more than one cluster head. Each node within the cluster can only communicate with its CH. That is, other 6LoWPAN nodes from different clusters cannot communicate with each other directly, but they can communicate via their CHs, as depicted in Figure 2.2. All the CHs in a 6LoWPAN network form an upper level hierarchy structure in which the CHs can communicate with each other or to the sink node (border router), which provides a connection to the external IP networks.

B. Mesh Topology

This topology also includes a 6LoWPAN coordinator that initiates and starts up the entire network. However, the communication in mesh topology is decentralised, i.e. each 6LoWPAN node can directly communicate with any other nodes within its communication range, as depicted in Figure 2.2. The mesh topology runs in an Ad-hoc manner and allows multi-hops data transmission across the network. It enhances network scalability, but it also increases network complexity for end-to-end data routing. In comparison with star topology, the mesh topology is power efficient, as the communication does not depend on one specific node.

In summary, each network topology is customised for particular application, with the major differences between 6LoWPAN topologies being summarised in Table 2.1.

Table 2.1: The major features of 6LoWPAN topologies

Features	Star	Mesh	Cluster Hierarchy
Scalability	No	Yes	Yes
Synchronisation	Yes	No	Yes
Inactive Period	6LN	6LH	6LN
Guaranteed Bandwidth	Yes	No	Yes
Redundant Path	N/A	Yes	No
Routing Protocol Overhead	N/A	Yes	No

2.2.3 6LoWPAN Protocol Stack

The International Organisation for Standardisation (ISO) proposed the Open System Interconnection (OSI) model, which is composed of seven layers and the development of the 6LoWPAN protocol stack is based on this seven-layer model. However, the 6LoWPAN protocol stack does not consider all the OSI layers and instead, consists of only five layers [23] [24]. Figure 2.3 shows the 6LoWPAN protocol stack alongside the OSI and the Transmission Control Protocol/Internet Protocol (TCP/IP) models.

The 6LoWPAN protocol stack is similar to the TCP/IP stack. However, there are a few differences between them, such as the 6LoWPAN stack adopting only IPv6 with an adaptation layer to optimise IPv6 packet transmission over the IEEE 802.15.4 standard. Also, the 6LoWPAN protocol stack is implemented to target the wireless embedded devices that are characterised by limited memory size, being power constrained, and having relaxed throughput.

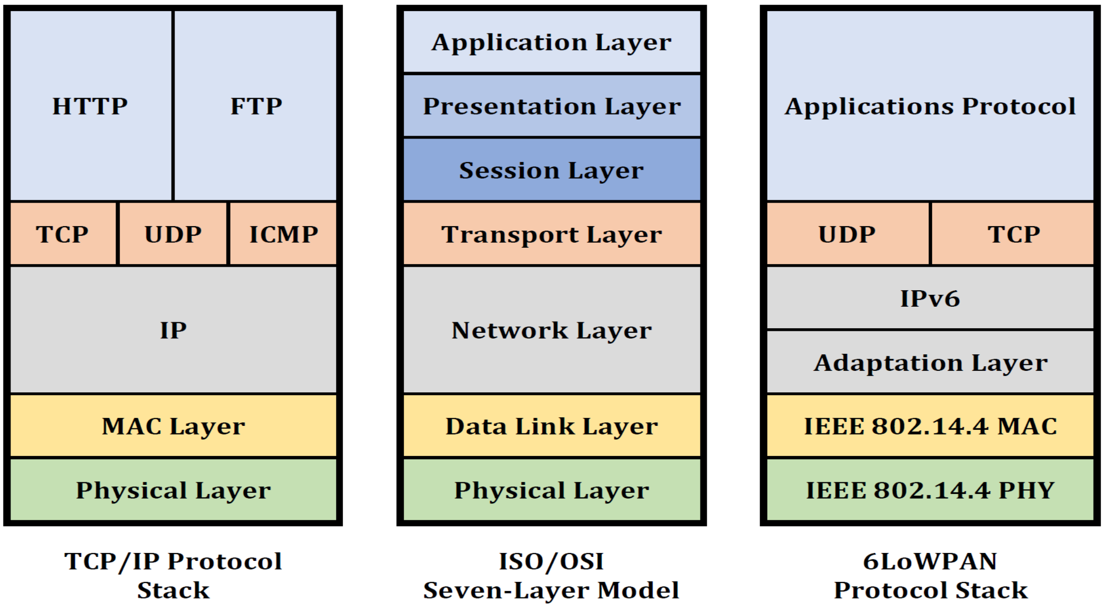


Figure 2.3: Protocol stack of TCP/IP, OSI model and 6LoWPAN [16]

The 6LoWPAN application layer builds adjacent to the end user of the system. This layer uses an Application Programming Interface (API) socket to send or retrieve packets between the external network and the 6LoWPAN network via the 6LBR or gateway. The API socket is correlated with a protocol (i.e. TCP or User Datagram Protocol (UDP)) in addition to the source and/or destination ports.

The transport layer in a 6LoWPAN protocol stack is responsible for process-to-process delivery, as in the OSI and TCP/IP models. The most well-known transport layer protocols are: TCP, a connection oriented protocol and UDP, a connectionless one. UDP is lightweight protocol, but it provides unreliable service with minimum error handling whereas, TCP provides reliable service and extensive error handling. However, UDP preferred to TCP regarding the aspects of performance, efficiency and implementation complexity.

The 6LoWPAN network layer is responsible for providing interconnecting capability between 6LN in the network and routing the packets along these connections. The network layer consists of an IPv6 layer and adaptation layer (i.e. 6LoWPAN layer). The adaptation layer is proposed to make IPv6 packets manageable and ready to transmit over the IEEE 802.15.4 standard, performing the following functions:

A. Header Compression: the MTU of the IEEE 802.15.4 standard is 127 bytes, while the IPv6 header is 40 bytes, this relinquishes 30% of the packet for the header only and less than 70% for the payload. For this reason, the header compression is necessary to leave more space for the payload;

B. Fragmentation and Reassembly: the IPv6 packet size is 1,280 bytes, while that for an IEEE 802.15.4 one is 127 bytes. It is impossible to encapsulated IPv6 directly in one IEEE 802.15.4 frame. Consequently, one IPv6 packet needs to be divided into more than 16 fragments. Hence, the adaptation layer performs fragmentation at the sender and reassembly at the destination;

C. Routing: the adaptation layer supports layer-two routing and the requirements of the routing protocol may vary depending on the selected communication path. Some routing protocols prefer energy saving paths, while other algorithms prefer path selection that provides best Quality-of-Service (QoS) and the remainder use both energy saving and QoS to achieve the best route.

The data link layer of the 6LoWPAN protocol stack represents the MAC sub-layer that delivers two services: the MAC management service and MAC data service. These services allow for multiple 6LNs to access and share the common communication medium as well as providing error detection and error correction services.

The physical layer is responsible for transmitting and receiving 6LoWPAN packets using a certain radio frequency and spreading technique. According to the 6LoWPAN specification, IEEE 802.15.4 uses 2.4 GHz as operational frequency with 16 channels at 250 kbps and a Direct Sequence Spread Spectrum (DSSS) as a spreading technique. The physical layer is in-charge of radio transceiver activation and deactivation, Link Quality Indication (LQI), Energy Detection (ED) (i.e. received signal power estimation) and finally, Clear Channel Assessment (CCA).

2.3 Modelling IEEE 802.15.4 MAC/PHY Layers

Developing a mathematical model of the IEEE 802.15.4 standard for slotted and un-slotted CSMA/CA received increasing attention in recent years. Great effort has been made in modelling the IEEE 802.15.4-based networks. In [25- 37], the authors have taken different approaches to model the MAC/PHY layers, trying to evaluate the performance of the analytical approach either by simulation or experiments. The available literature on analytical models can be classified into two groups. According to the parameters being investigated, traffic patterns (saturated and unsaturated traffic) and the CSMA/CA mechanism (slotted and un-slotted). Moreover, they have aimed to construct the system transition matrix from which the performance of the system can be analysed using probability methods. It is of interest to propose a mathematical model for predicting network performance accurately without resorting to complex experiments or lengthy simulations.

2.3.1 Modelling Beacon Enabled Networks

Two scenarios were introduced in [25] to model the behaviour of the IEEE 802.15.4 standard. The first involved proposing direct communication between the sensor node and the PAN coordinator. While in the second scenario, the sensor nodes communicated with the PAN coordinator through relay nodes. The sensor node adopted packet transmission without acknowledgement. The authors focused on the network throughput and the effects of the finite buffer size on data aggregation.

The authors in [26] presented a generalised analysis of the IEEE 802.15.4 MAC protocol. The system was modelled through a Markov chain, taking into account retry limits, acknowledgements, unsaturated traffic, and also the impact of the MAC parameters. The authors reported that Monte-Carlo simulations confirmed that the proposed approximations offered a satisfactory accuracy.

The authors in [27] developed a flexible mathematical model for a beacon-enabled mode of the IEEE 802.15.4 MAC layer standard for different network topologies. Both the Contention Access Period (CAP) and the Contention-Free Period (CFP), as defined by the standard, were considered. The proposed model dealt with star and tree based topologies to describe the probability of successful transmission for tuning the MAC parameters of the selected topology and the results were validated through simulation.

The effects of deferment packets on the performance of CSMA/CA of IEEE 802.15.4 were analysed in [28]. A Markov chain model was used to develop the analytical model for the deferred transmission of IEEE 802.15.4. The simulation results showed that the throughput was affected by the deferment of packets for short super-frame duration.

The authors in [29] proposed an Adaptive Priority Based Service differentiation CSMA/CA algorithm (APBCSMA/CA) for different services of the IEEE 802.15.4 standard. For the proposed algorithm, two mechanisms were adopted: adaptive backoff mechanism and priority-based service-differentiation mechanism. Markov chain model was developed and compared to the simulation results. Based on the observed results, the proposed algorithm performed better in terms of reliability and channel utilisation.

Both ZigBee and Wireless Local Area Networks (WLAN), based on IEEE 802.15.4 and IEEE 802.11 standards, respectively, operate in overlapping unlicensed frequency bands. The authors in [30] developed a mathematical model based on a Markov chain to evaluate the throughput performance of a coexisting IEEE 802.15.4 ZigBee network and 802.11 WLAN operating on the same channel. The numerical results showed high accuracy for the proposed model and its benefits in designing coexisting network.

A station-level stochastic time-domain method was proposed in [31], which provided non-stationary model for the IEEE 802.15.4 with slotted CSMA/CA MAC protocol and without the optional acknowledgements. The authors stated that the proposed model proved to be of lower memory and computational complexity order compared to existing ones.

Markov chain modelling of randomly deployed IEEE 802.15.4 nodes was proposed in [32], with the proposed model considering the nodes with finite buffer, beacon enabled data transfer with no acknowledgements and no retransmissions. In addition, the wireless sensor nodes communicated to the coordinator in a single hop. The network was simulated using NS-2 and the obtained results illustrated the influence of fixed buffer and transmitted packet size on the network throughput. The authors concluded that the obtained simulated results were close enough to the theoretical values.

2.3.2 Modelling Non-Beacon Enabled Networks

An analytical model for a non-beacon enabled mode of the IEEE 802.15.4 MAC protocol was provided in [33]. The authors presented a mathematical model that enabled the evaluation of the statistical distribution of the traffic generated by the sensor nodes. The obtained simulation results showed that when different loads were generated across the network, the traffic distribution changed accordingly. Also, the optimum size of the packet was evaluated in order to obtain the maximum successful transmission probability.

In [34], a Hidden-node Aware Grouping (HAG) algorithm was proposed in order to enhance the performance of IEEE 802.15.4 networks. For accurate evaluation of HAG algorithm, throughput under both saturated and unsaturated environments was investigated. A conventional Markov chain model of IEEE 802.15.4 was developed taking into account the deferment technique and a traffic source to enhance the simulation results.

Cross-layer models are becoming popular in various wireless networking domains. A combined PHY/MAC layer energy consumption model was developed in [35] to consider short range IEEE 802.15.4 networks for the non-beacon-enable mode. The proposed model was focused on the differences between single layer models (PHY or MAC) and a cross layer model from an energy consumption point of view.

The authors in [36] presented an analytical model for IEEE 802.15.4 networks with joint MAC and PHY layers. The log-normal shadowing, interference, hidden terminal and modulation type were evaluated in the proposed model. MATLAB was used to simulate the proposed model and its performance compared to the OPNET simulation tool. The simulation results showed that consideration of the realistic channel conditions had a significant impact on the network performance compared to the ideal one.

Markov chain analysis for low-data rate unsaturated traffic was proposed in [37]. The proposed model was focused on the channel access probability, packet collision, acknowledged transmissions, and exponential backoff with retry limits. The nodes generated periodic traffic and were equipped with a D/M/1 queue. This analysis could be used as a system design tool, where various network system parameters could be configured to meet a desired end-to-end delay, energy efficiency, and throughput.

Chapter 3 is motivated from many research that tried to analytically described the behaviour of the IEEE 802.15.4 MAC layer standard. Most of the previous analytical models are related to beacon enabled networks [25-32], while a few of the works have devoted towards non-beacon enabled one [33-37]. However, none has focused on the modelling 6LoWPAN MAC and PHY layers standard specification, including operational frequency and modulation type. In summary, the main contribution of Chapter 3 is the development of comprehensive mathematical model for a 6LoWPAN MAC and PHY layer, taking into account the different operation frequency, modulation type and unslotted CSMA/CA mechanism of the IEEE 802.15.4 standard. In addition, the proposed model is fully analysed regarding the system throughput, system probabilities (idle, collision, transmission), and the delay of transmitted packets in both the node queue and channel. In contrast with the previous cited works, the proposed mathematical model also includes the effect of an M/M/1/K queue model, where different parameters, such as packet arrival rate, buffer size, and packet size are considered.

2.3.3 Optimising MAC Layer Parameters

Nowadays, the IEEE 802.15.4 standard is a key technology for the development of M2M and IoT. Consequently, much research has been verified by simulation tools, such as MATLAB, NS-2, NS-3, OPNET, or OMNET. Many works have involved studying the performance of the CSMA/CA mechanism of the IEEE 802.15.4 standard and different algorithms, either to enhance the end-to-end delay or improve the energy consumption in IEEE 802.15.4-based networks. Some of the research by these authors is summarised in the following paragraphs.

The authors in [38] presented a novel approach for minimising the energy consumption of un-slotted IEEE 802.15.4 MAC protocols using optimisation techniques.

The objective function was related to the total energy consumption in the transmit, receive, listen, and sleep states, in addition to the delay and reliability of the packet delivery. While the decision variables were the sleep and wake time of the receivers. Storing light look-up tables in the receiver nodes represented the optimal solution and made it easy to implement on existing IEEE 802.15.4 hardware platforms.

A fuzzy CSMA/CA MAC protocol was proposed in [39], with the proposed protocol utilising two separate fuzzy logic controllers. The first, was used to optimise the MAC parameters and a sleeping schedule duty-cycle, whilst the second was aimed at optimising the size of the contention window using three performance metrics as inputs. These two fuzzy logic controllers were deployed with purpose of ensuring maximum power efficiency of the network.

The authors in [40] proposed a Collision-Aware Backoff algorithm (CABEB) to improve the performance of a slotted CSMA/CA for the IEEE 802.15.4 standard. The CABEB algorithm provided dynamic selection of a backoff period depending on the current collision probability of the network. The proposed approach was able to configure the MAC layer parameters autonomously based on the available channel state information. The analytical results were based on Markov chain modelling, while the simulation results were based on OMNET++ simulation software. The obtained results showed that the CABEB algorithm performed better that the default IEEE 802.15.4 standard and the knowledge-based exponential backoff algorithm.

The analysis of un-slotted IEEE 802.15.4 MAC was given in [41], the expressions of which were represented as a function of sleep time, listening time, traffic rate and MAC parameters. The analytical results were then used to optimise the duty cycle of the nodes and MAC protocol parameters. The authors reported that significant reduction of energy consumption compared to existing solutions was achieved.

The authors in [42] applied models that led to the idle sensing access method of IEEE 802.11 to the slotted CSMA/CA of IEEE 802.15.4 standard. They were taking into account the central role of the coordinator as well as the burst nature of the traffic. The contention window was adjusted depending on optimal values to achieve high throughput along with low duty cycles, which led to low energy consumption.

The authors in [43] provided an analysis of the fundamental MAC and routing protocols for Low-power and Lossy Networks (LLNs): IEEE 802.15.4 MAC and IETF IPv6 Routing Protocol for Low-power and lossy networks (RPL). The characterisation of their cross layer interactions was presented in the form of a mathematical description, with a protocol selection mechanism being implemented to select the appropriate routing metric and MAC parameters for given specific performance constraints. Both the analytical and experimental results showed that the behaviour of the MAC protocol affected the performance of the routing protocol and vice versa, unless these two were carefully optimised together.

The ZigBee limitation could become a real problem, if the user wishes to transmit a large amount of data in a very short time. The authors in [44] proposed a solution by applying particle swarm optimisation to scalable rate control in order to increase the available bandwidth. This approach led to improvement in the quality of picture and reduction in data loss when transmitting Motion Picture Experts Group layer-4 (MPEG-4) videos over the ZigBee network.

A Genetic Machine Learning Algorithm (GMLA) was proposed in [45] for Wireless Sensor Network (WSN) data fusion applications, with the aim of improving communication efficiency. Random topologies were used in the simulation and GMLA presented almost 13% of gain over IEEE 802.15.4 in 1,000 simulation rounds.

Off-line computation, model-based adaptation, and measurement-based adaptation were compared by simulation in [46] to select the optimal MAC parameter setting to provide reliability with minimum energy consumption with the IEEE 802.15.4 standard. The adaptive algorithms performed well compared to other models, which were unsuitable in practical scenarios, where the transmission errors could not be neglected.

In order to address the problem of selecting parameters that minimise the average packet delay, the authors in [47] developed a queueing model to evaluate the delay of a class of discrete-time, throughput-optimal MAC protocols. Then, the queuing model was used to derive the optimal parameter settings for the MAC protocol. The parameters selection and the delay model were validated using simulation tools.

The energy consumption in WSN is affected by a variety of MAC parameters and the challenge of optimising WSN networks in terms of a low-energy consumption has been a difficult problem facing researchers. The authors in [48] focused on optimising WSN protocols using the Ichi Taguchi (Taguchi) optimisation method. That is, the energy consumed by sensor nodes were optimised using the Taguchi method to predict network topology design parameters. The simulation results were obtained using an OMNET++ simulator, with the results showing the impact of the network protocols on energy consumption.

The authors in [49] proposed the Adaptive Access Parameters Tuning (ADAPT) algorithm for dynamically adjusting the MAC parameters, based on the desired level of reliability and actual operating conditions experienced by the sensor nodes. The simulation results showed that the ADAPT algorithm was able to provide the desired reliability with a very low energy expenditure, even under operating conditions that dynamically change with time during network operation.

In the previous literature given in Section 2.3.3, many studies have shown that the IEEE 802.15.4 standard may suffer from severe limitations in terms of network reliability and energy efficiency, if non-appropriate parameter settings are used. Much effort has made regarding MAC parameters selection in terms of achieving better power consumption and overcoming delay constraints. However, less attention has been paid to optimising these parameters and selecting an optimal set that provide high reliability with minimum energy consumption. This issue is solved in Chapter 3, where the evaluation of an optimal MAC parameters set is proposed. The evaluation technique is based on Artificial Neural Network (ANN) and optimisation techniques to achieve high throughput with minimum delay. Also, a comparison between a Genetic Algorithm (GA) and Particles Swarm Optimisation (PSO) was conducted to choose the best intelligent optimiser that provides the optimal set for 6LoWPAN MAC layer parameters.

2.4 Routing Protocols in PAN Networks

The M2M sensor network does not have a fixed infrastructure, and its topology may vary from star to multi-hop mesh network. It differs from other networks in the aspects of densely deployment, small node size, frequent node failure, and bidirectional communication being available. Moreover, the topology is dynamic due to node-to-node link failure, node battery exhaustion or node mobility. Topology control techniques are important in such networks for managing the complexity of randomly deployed nodes and provide full connected topology via self-organised capability [50].

Topology control techniques can be defined as promising ways for controlling network parameters to generate an energy-efficient topology for the entire network. These techniques can be assigned to four different categories, depending on the implemented approach for reducing M2M sensor node energy consumption [51].

Figure 2.4 depicts the interaction between the topology control technique categories, these categories are: (i) the power adjustment approach, which enables M2M sensor nodes to collaborate with each other to find the adequate transmission power for full topology construction by changing their transmission range to reduce communication energy. (ii) The power mode approach enables the M2M sensor nodes to change their operational states (idle, sleep, transmit and receive), which optimises node energy and prolongs the network lifetime by placing most of the sensor nodes in sleep mode without sacrificing the whole topology connectivity. (iii) The clustering approach constructs M2M sensor network topology in hierarchical fashion, with the idea being to select a set of nodes called cluster heads to collect, process and forward packets from cluster members. The advantages of the clustering approach, include scalability, data aggregation and load balancing to prolong the network lifetime. (iv) The hybrid approach is the integration of the clustering approach with other approaches, such as power mode or power adjustment, to enhance the network connectivity and to prolong the network lifetime by conserving M2M nodes' energy.

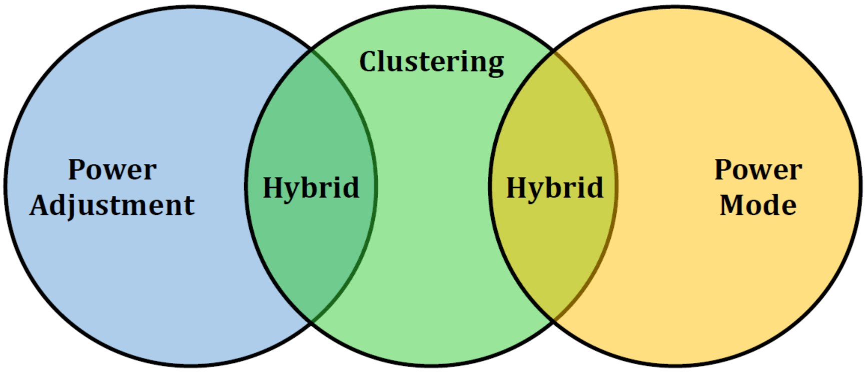


Figure 2.4: The four categories of topology control techniques in sensor networks

Routing can be defined as the mechanism that determines the path between two nodes in the network, with the first being the sender (source) and the second, the receiver (destination). In M2M sensor networks, most routing mechanisms are implemented in the network layer. In large scale M2M sensor networks, where the sensor nodes are deployed randomly, the source node cannot reach the destination directly. Consequently, relay nodes are used to forward the data to the destination if the incoming messages are not designated to itself.

Energy efficiency has been received a considerable attention by the research community in research related to M2M sensor networks, because most M2M sensor nodes are battery powered and battery replacement is unpractical. In addition, hardware complexity is another issue that needs to be considered when designing new routing algorithms. Designing a routing protocol for M2M sensor network is a real challenge due to limited bandwidth, unreliable wireless channel and dynamic topology changes. These limitations and challenges can be addressed via intelligent management of network resources. Since a massive number of M2M sensor nodes are deployed randomly in the sensing field, each node can cover a limited area, multi-hop communication is frequently deployed in order to reach the sink node. However, because M2M sensor nodes are deployed densely in the same geographical area, node overlapping may occur, which causes redundant data and leads to an energy waste problem. Accordingly, routing protocols must have the ability to adopt topology changing frequency and resource utilisation in order to enhance the energy consumption, load balancing and scalability.

The routing protocols in M2M sensor networks can be divided into three main sub-divisions depending on the topology structure and the role played by the sensor nodes in data routing. These three sub-divisions are as follows [52] [53]:

1. Location-Based Protocols
   Location-based routing protocols are also known as geographic routing protocols; the routing decisions are based on the location information exchanged between M2M sensor nodes instead of the connected topology information. The sender node is aware of its location as well as the location of the destination node in order to transmit a unicast packet to a single destination. On the other hand, for multicast or broadcast transmission, the sender packets are propagated to multiple destinations. The redundant links are reduced in these protocols by knowing the location of the destination nodes and hence, this results in reduced energy consumption.
2. Data Centric Protocols
   Data centric routing protocols are also known as flat-based routing protocols and all the M2M sensor nodes are identical in functionality and role. It is not possible to provide a global unique identifier of each node in large scale M2M sensor networks. Consequently, a data centric routing technique is deployed that focuses on selecting certain nodes with special attributes for information retrieval and dissemination. The selected nodes aggregate multiple node data prior to be forwarded to the sink.
3. Hierarchical Protocols
   With hierarchical routing protocols, also known as clustering protocols, the M2M sensor nodes are grouped into clusters in order to provide a solution for the scalability and energy efficiency issues addressed in the other types of routing protocols. In this routing technique, the M2M sensor nodes within a particular cluster communicate directly with the leader node called the cluster head. The cluster heads are selected firm among the cluster members to reduce the energy consumption in cluster nodes (members) by performing data aggregation and forwarding them on behalf of the cluster members. The cluster heads will deplete their energies more quickly than the cluster members as they experience more traffic than the sensor nodes. Their selection and rotation are important matters that need to be taken into account when designing a hierarchical routing protocol due to dynamic topology

changes and cluster head failure. Compared to other routing techniques, cluster-based routing protocols may eliminate the collisions in data transmission in PAN networks and increase the M2M sensor nodes' duty cycle for energy efficiency data routing. In sum, a hierarchical structure improves network scalability and lifetime.

After the M2M sensor nodes are identified, the routing protocol is responsible for building up the routes for data delivery between different nodes and maintaining these routes. The ways in which different routing protocols behave make them suitable for particular applications. There are plenty of routing protocols available in the related literature regarding clustering, energy efficiency, and scalability in sensor networks. This section aims at providing the most relevant protocols to expedite the perception of various routing techniques that can be adopted in M2M sensor networks.

The energy consumed in transmission is significantly more than the energy dissipated in sensing and processing. Therefore, evolving energy efficient algorithms is necessary for energy reduction in M2M sensor networks. The clustering routing protocols provide longer network lifetime and enhance the network scalability as compared to the other routing protocols (i.e. location-based and data centric routing protocols). Accordingly, this thesis will focus only on the energy-efficient clustering routing protocols for M2M sensor networks that extend the network lifetime, enhance the scalability, and provide more management features. Figure 2.5 illustrates the taxonomy of energy-efficient routing protocols and the green rectangles represent what this thesis will deal with. In the literature, plenty of routing protocols have been proposed to improve the performance of sensor networks in terms of network lifetime and scalability. The next subsections will provide a survey of hybrid hierarchical routing protocols that adopt clustering and power mode topology control techniques in sensor networks.

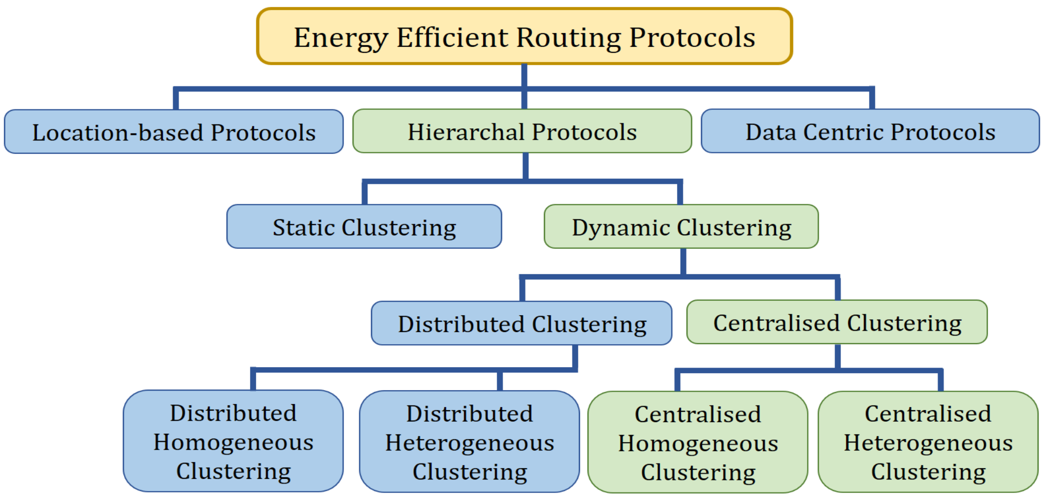


Figure 2.5: Classification of energy efficient routing protocols

2.4.1 Cluster-Based Routing Protocols

Clustering mechanisms have been developed to cope with the energetic constraints of M2M nodes that are deployed in remote monitoring areas. M2M nodes are usually battery powered devices that cannot change or charge their batteries and hence, energy constraints are the most critical problem that need to be addressed and solved efficiently. M2M networks consist of a large number of nodes and in order to overcome M2M nodes' energy depletion, these nodes are grouped to form clusters. In the clusters, as aforementioned, some nodes are nominated as Cluster Heads (CHs) for each created cluster. The cluster-based M2M networks are composed of M2M nodes, clusters and CHs. Cluster member nodes transmit their data to the CH, which is the organiser of the cluster that performs data-aggregation, data fusion and forwarding the aggregated data to the sink node. Accordingly, the CHs deplete more energy than cluster members, because of the frequent packet relaying between cluster members and the sink node [54].

The clustering approach can be used to balance the power consumption across the M2M network by periodic reselection of the CH within each cluster based on the residual energy, or periodic rotation of the CH within the cluster [55]. Also, the clustering approach can be used to conserve energy by switching the nodes to low-power sleep mode most of the time and reducing the energy consumption in M2M node [56].

There are several aspects regarding CH reselection or CH rotation, which concern the residual energy in M2M node to be selected as a CH, the number of neighbour nodes, the distance between node, and the execution delay of the algorithm to select a CH to form a cluster.

Low Energy Adaptive Cluster Hierarchy (LEACH) is a pioneering cluster-based routing protocol proposed by Heinzelman *et al.* in [57] to maximise the network lifetime in microsensor networks. The network is divided into clusters, with the cluster heads being randomly rotated in order to distribute the energy load among sensor nodes equally. Simulation results showed that the proposed LEACH outperformed static clustering algorithms. The authors in [58] proposed LEACH-Centralised (LEACH-C) with a centralised clustering scheme, the algorithm for which required knowledge of node location by the sink node. Each sensor node was equipped with a Global Positioning System (GPS) for node position determination, then the sink node transmitted a message of optimum cluster head locations to all sensor nodes in the network. The simulation results showed that LEACH-C is better than LEACH in terms of network lifetime and energy conservation. Two Level-LEACH (TL-LEACH) was proposed in [59], where two level hierarchy was presented to achieve better energy dissemination among cluster member nodes. Energy-LEACH and Multihop-LEACH were proposed in [60] with different improvements over the LEACH protocol. Energy-LEACH was the first protocol that considered the node's residual energy in cluster head selection, while the Multihop-LEACH considered multihop communication in sensor networks.

There have been different versions of the LEACH routing protocol proposed in the literature [61] [62]. Energy Efficient Extended LEACH (EEE-LEACH) is another version of LEACH proposed in [63], where multilevel clustering was formed and a nominated master cluster head along with cluster heads to shorten the communication distance between the sensor nodes. The authors in [64] proposed another enhanced LEACH algorithm called Modified-LEACH (MOD-LEACH), where dual transmission power levels were introduced between intra clusters as well as between the cluster head and sink node, which aimed to minimise network energy consumption by efficient cluster head replacement. In MOD-LEACH, a cluster head was replaced when its energy fell below a certain threshold to minimise routing load of the protocol. The authors in [65] proposed a Hierarchical Cost Effective LEACH (HCEL) protocol for three levels heterogeneous networks, which aimed to enhance the energy efficiency of the nodes and maximise network performance. The simulation results showed that HCEL achieved a gradual decrease in the network deployment cost ratio in terms of powerful nodes and energy factor. Advanced-Multi-hop LEACH, proposed in [66] for heterogeneous sensor networks, involved using the optimal path for data transmission between cluster heads and the sink node. The network lifetime was prolonged by evaluating the optimum number of clusters and also, considering the optimum number of hops in the network with heterogeneous nodes. The simulation results showed that the advanced multi-hop LEACH protocol outperformed the advanced single-hop LEACH protocol in terms of energy efficiency, stability and network lifetime.

Threshold sensitive Energy Efficient Sensor Network (TEEN) is a cluster-based routing protocol proposed by Manjeshwar *et al.* in [67] for event-aware sensor network applications. TEEN defined two types of thresholds: soft and hard. The hard threshold was used as sensed attributes and the node compared the sensed value with this threshold. If the hard threshold was exceeded, the sensor node would turn on its communication transmitter and transmit the observed data to the cluster head. On the other hand, the soft threshold was used to prevent data transmission redundancy. The sensor node did not transmit the data when the difference between two consecutive observations did not exceed the soft threshold. TEEN protocol was extended in [68], with the new version being called the Adaptive Threshold sensitive Energy Efficient Sensor Network (APTEEN) protocol. The enhanced version of TEEN used TDMA scheduling for observed data transmission to the cluster head. A Multipath Adaptive Periodic Threshold sensitive Energy Efficient Network Protocol (MAPTEEN) was proposed in [69] to allow the sensor node to find multiple routes for the observed data to reach a certain destination. The MAPTEEN approach succeeded in increasing network reliability and decreasing the consumed power in the sensor nodes, thus increasing the network lifetime. Another extension based on TEEN and APTEEN was proposed in [70] in order to eliminate the redundancy of the observed data and hence, to improve the energy efficiency of the sensor node, which was called the Distance Adaptive Threshold sensitive Energy Efficient Sensor Network (DAPTEEN).

Stable Election Protocol (SEP) is a cluster-based protocol proposed by Smaragdakis *et al.* in [71] as the first protocol that considered the concept of network heterogeneity in routing the sensed data. There are two types of nodes, advanced and normal, with the difference between them being the initial energy level of each node. The energy level of the advanced node was higher than the normal node and their number was less. The cluster head selection was based on weighted probability, which used a node's residual energy as a function for uniform energy consumption among the sensor nodes in the network. The authors in [72] proposed two extensions to the SEP routing protocol, namely, Zonal-Stable Election Protocol (Z-SEP) and Threshold-sensitive Stable Election Protocol (TSEP). Z-SEP was designed for two levels of network heterogeneity, where some nodes transmitted their data directly to the sink node. While TSEP was designed for three levels network heterogeneity and similar to TEEN in the way that the cluster members sent their data to the cluster head. A modified routing protocol based on SEP was proposed in [73], which was called the Efficient Modified-Stable Election Protocol (EM-SEP). EM-SEP was aimed at prolonging the stable period of the sensor network by maintaining balanced energy consumption. Advanced nodes were chosen more frequently than the normal nodes to become cluster heads, as was the case with the SEP protocol. Furthermore, EM-SEP took into account the number of nodes that were associated with each cluster head and the sensor with the highest energy was nominated to be the cluster head at a certain round among other sensors. A Modified-Stable Election Protocol (M-SEP) with two levels of heterogeneity was proposed in [74], with simulation results showing that it prolonged network lifetime by 55% and 40% compared to LEACH and SEP, respectively. Accordingly, M-SEP increases the stability period and packet transmission rate as compared with LEACH and SEP.

Distributed Energy Efficient Clustering (DEEC) is a cluster based heterogeneous routing protocol proposed by Qing in [75] for three level heterogeneous sensor networks. Three types of sensor nodes are present, normal, advanced and super, which are different to each other according to the initial energy level. Cluster head selection is based on the ratio between the remaining node energy to their initial energy and that with highest ratio becomes the cluster head. A Balanced and Centralised Distributed Energy Efficient Clustering (BCDEEC) was proposed in [76] as an extension to DEEC, where the sink node (base station) selected the nodes with higher energy to be a gateway and cluster heads in order to enhance the network lifetime. A clustering algorithm based on DEEC was proposed in [77] to solve the isolated nodes problem. The nodes calculated the distance to all cluster heads and to the sink node, with those nodes closer to the sink node than any cluster head being able to communicate directly with the sink with minimum transmission power and hence, they became isolated nodes. The isolated nodes enhanced the network lifetime by reducing the energy consumption during communication. Another improvement to DEEC proposed in [78] was called the Improve Threshold Distributed Energy Efficient Clustering (ITDEEC) protocol, for which the nodes nearer to the sink node were excluded from the clustering process. The ITDEEC protocol prolonged the network lifetime by 46% compared to traditional TDEEC. The authors in [79] proposed a Bio inspired Distributed Energy Efficient Clustering (B-DEEC) protocol based on the Artificial Bee Colony (ABC) algorithm in order to increase both network throughput and lifetime. The authors in [80] proposed a new cluster head selection mechanism based on DEEC depending on nodes' residual energy, called the Modified-Distributed Energy Efficient Clustering Algorithm (M-DEEC). The proposed clustering technique successfully prolonged network lifetime by 12.5% in comparison to the DEEC protocol. Regarding the Enhanced Developed-Distributed Energy Efficient Clustering (ED-DEEC) routing protocol proposed in [81] which is an energy aware three level heterogeneous clustering protocol, the comparative analysis showed that ED-DEEC outperformed the exciting DEEC protocols.

M2M sensor networks are deployed randomly in the monitoring area and make decisions independently. It is clear from the previously surveyed protocols that each node in the sensing field makes autonomous decisions without any centralised control. Energy-efficient routing protocols have received much attention by researchers aiming to enhance network lifetime and improve energy consumption. However, limited research has been targeted at network centralised control and management to enhance its programmability by enabling IoT technologies (SDN, NFV and cloud computing).

2.4.2 Sleep Scheduling Routing Protocols

Energy efficiency in M2M networks has always been a hot research topic and has received a lot of interest from the research community. A sleep scheduling mechanism is an efficient method for managing the residual energy of each node and being capable of prolonging the lifetime of the entire network. Sleep mode can be interpreted as turning off "sleep" subsets of nodes, while other nodes remain "active" without affecting network connectivity. Figure 2.6 shows different sleep scheduling mechanisms that will be studied in this subsection, including energy-efficient TDMA sleep scheduling, balanced-energy sleep scheduling, optimal sleep scheduling and dynamic sleep scheduling. These techniques have been used to improve M2M node's energy consumption and prolong network lifetime.

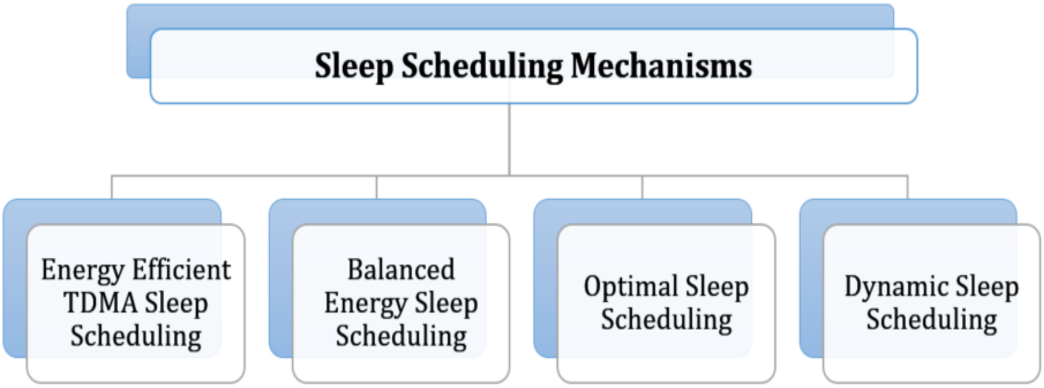


Figure 2.6: Classification of sleep scheduling mechanisms

A. Energy-Efficient TDMA Sleep Scheduling

In energy-efficient TDMA-based protocols, the time is divided into multiple time slots, which are allocated among many M2M nodes. These time slots are assigned for switching on a node's radio during the assigned time slot and switching the radio off at the end of the slot, when no-transmission or receiving of packets takes place. The sleep mode scheduling mechanism can be performed by the sink node or the CH among the cluster members within the node's transmission range [82]. The advantages of TDMA sleep scheduling is the maximising of network lifetime and reduction of packet losses during the sleep period.

B. Balanced-Energy Sleep Scheduling

The sleep mode is used to conserve the energy of the node's battery by rotating its states between active and sleep within the clusters in M2M networks. The main objective of balanced energy sleep scheduling is to balance the energy dissipation of the nodes during active/sleep periods with respect to the node's residual energy [83]. In this scheduler, the sleep probability is calculated according to the node's distance from the cluster head to ensure the average energy consumption of all the nodes is the same. The main advantages of this scheduling are an increase in the sleep period by using redundant nodes and enhancing network performance by balancing its load.

C. Optimal Sleep Scheduling

M2M nodes perform active/sleep period periodically within fixed time intervals, rather than evaluating the system with fixed intervals. Many researchers have proposed a cost structure for an optimal value for active and sleep periods, respectively, such as in [84]. The optimal sleep scheduler focuses on the tradeoffs between energy consumption and packet delay. It attempts to identify the manner in which the optimal sleep schedule varies with the length of the sleep period, the statistics of arriving packets, packet delay and energy consumption. The advantage of this technique is to minimise the delay during the sleep period.

D. Dynamic Sleep Scheduling

The dynamic sleep scheduler allows each sensor node to adjust its duty-cycle (the time interval for active/sleep period) according to the node's residual energy, as in [85]. The residual energy of the M2M nodes can be decreased over time due to sensing, processing, and communication processes. In dynamic sleep scheduling, each M2M node can reduce its duty-cycle depending on its residual energy or traffic pattern. The advantages of this scheduler can be summarised as avoiding packet losses while the communication antenna is off and providing high throughput.

In order to conserve the battery power in very dense M2M network, some nodes may be put into the sleep mode while other nodes remain active for the sensing and communication tasks. This subsection explores and presents the latest techniques of consumed energy reduction in M2M networks. The related works [84-87] have reviewed the energy-efficient techniques from different points of view.

Three different sleep scheduling schemes for cluster-based sensor networks were proposed in [86] for prolonging the network lifetime: Randomised Scheduling (RS), Distance-based Scheduling (DS) and Balanced energy Scheduling (BS). In RS, the sensor nodes were selected randomly among cluster members to be put in sleep mode. While in DS, the sleeping probability of the sensor node related to the distance between the node and the cluster head; the farther node had higher probability in order to conserve its energy. On the other hand, the BS was a special case of a DS scheme in which the sleeping probability was chosen in a way that all the sensor nodes depleted the same amount of energy. Sleep control laws were proposed in [87] to minimise the cost function value for both energy consumption costs and backlogged packet holding cost. The sleep control laws were optimised under two conditions: finite and infinite average cost function. The main difference from infinite cost function was the appearance of a "shutdown" period at the end of the time slot when the queue was filled and stopping serving the arrived packets. The authors in [88] proposed another sleep scheduling mechanism called Dynamic Sleep Scheduling (DSS) to minimise the delay in sensor networks. The sleep/awake periods were identified according to the traffic load and events that occurred at the sensor node. This approach performed different active/awake intervals and as a result, end-to-end delay and energy consumption were minimised by reducing the congestion occurrence under the heavy traffic condition. An Energy Saving Sleep Scheduling (EPSS) algorithm was proposed in [89] to adopt unequal clusters and develop a new clustering mechanism, which considered the node's residual energy and the distance between sensor nodes. Once the clusters were formed, the cluster head performed sleep scheduling among its members depending on the node's distance. The authors in [90] proposed two energy efficient sleep algorithms called the Energy Saving via Opportunistic Routing (ENS-OR) algorithm and Geographic Random Forwarding (GeRaF) algorithm, to minimise energy consumption and extend network lifetime.

2.5 Sink Mobility in M2M Sensor Networks

Information acquisition is the essential task of sensor networks. Using a single sink in a sensor network can cause an uneven energy depletion problem, whereby the sensor nodes closer to the sink exhaust their energy quicker than those far away due to heavy message relay on behalf on the other nodes, known as the hot-spot problem [91]. Non-uniform energy depletion shortens network lifetime and causes degradation in network performance. The sink becomes isolated from the sensor network when the sensor nodes about the sink run out of energy, thus causing network failure. There are considerable benefits of applying multiple sinks in a sensor network with respect to energy consumption and latency of the gathered data. Deploying multiple mobile sinks in a sensor network mitigates the hot-spots by sharing the data transmission overhead between all the sinks in the sensor network. In addition, the average distance is reduced between the sensor nodes and sink nodes, ultimately resulting in conserving node energy and minimising the latency. The single bottleneck problem is eliminated by deploying multiple sinks, for in the case that a particular sink becomes disconnected from the sensor network, the sensor nodes can still have a redundant path to transmit the data to other sinks in the sensor field and hence, the network continues to function. Finally, load balancing can be achieved by multiple sinks through sensor field partitioning. [92]. The careful selection of the number and accurate locations of sink nodes directly influence the network performance in an M2M sensor network. There are many approaches available in the literature for finding the exact number [93] and best location for sink nodes [94]. Recent research has shown that adopting mobility in sensor or sink nodes can improve network connectivity [95], reduce delay [96] and extend network lifetime [97]. Mobility in sensor networks can be classified into three main categories [98] as follows:

1. Node Mobility: the wireless sensor nodes themselves can be mobile, with the mobility depending on the sensor network application. In many applications, like environmental monitoring, sensor node mobility is not possible, while in livestock surveillance (i.e. sensor nodes attached to cattle), it frequently happens. This self-organised feature needs to adopt frequent movement of the sensor nodes, which involve trade-offs between the quality of information and the consumed energy, whilst the network functionality depends on the speed and frequency of the node movement.
2. Sink Mobility: data requesters or information sinks can be mobile, which can be seen as a special type of node mobility. The main differences are that the mobile sink is used to gather the sensed information and it is not part of the sensing task. A simple application is a human requesting information while walking inside an intelligent building via a personal digital assistance device. The data requester can communicate with the sensor nodes using unrelated and separated data requests. The mobile data requester (mobile sink) can request data from sensor nodes in its proximity, whilst the network routing protocol must take into account the movement of the mobile sink and the requested data must reach the sink regardless of its movement.
3. Event Mobility: the stimulation of the events or the trackable objects are mobile, such as tracking and event detection scenarios. In such scenarios, a sufficient number of sensor nodes is required in order to spot the event and to be involved in the consequent activities in relation to observing the trackable object. After these activities, the sensor node goes back to low-power mode (sleep mode). As the event location changes over time, the sensor nodes that sensed the event will change over time as well (i.e. source node). The routing protocol for such scenarios must have the ability to support appropriate functions for unexpected event mobility.

Event mobility is quite uncommon compared to node mobility and sink mobility. Moreover, sensor node mobility and event mobility are not under consideration of this thesis, with the focus thus being on sink mobility. Without losing the generality, sink node mobility can be categorised into three categories depending on the mobile sink node path determination technique:

A. Random sink mobility: a sink node moves in random trajectories that are built up of linear segments or different lengths and directions. The mobility of the sink is considered as uncontrolled. In some scenarios, the mobile sink speed and the pause time along different segments is also assumed to be arbitrary. The research works for this mobility type are included in [99]. In all these works, the authors concluded that using random mobility for the mobile sink results in better load balancing and thus, longer lifetime for the sensor network. The authors focused on how source sensors can locate the constantly moving mobile sinks and send data packets through a smaller number of forwarding hop counts. The authors in [100] used uncontrolled sink mobility to optimise network lifetime and meet certain QoS.

B. Controlled sink mobility: the mobile sink path can be predicted depending on the network's interest variables (i.e. energy or distance) and the path gets adjusted to guarantee best network performance in terms of network lifetime and latency. The controlled sink mobility is adopted to avoid hot-spots around the sink node and to distribute the energy depletion across the whole sensing field. In addition, the controlled sink mobility is used to avoid long distance data transmission by enabling single-hop communication between the sensor node and the sink. An example clarifying the concept of controlled trajectory in multi-hop sensor networks is given in [101] in which the mobile sink receives periodic updated information about a node's energy level. Based on these updates, the mobile sink regulates its trajectory to minimise to the routing load on the nodes with low energy level. Other works that have dealt with optimising mobile sink trajectory are [102] [103].

C. Constrained sink mobility: in contrast to controlled and random sink mobility, the constrained mobility is fully deterministic. In this type of sink mobility, the mobile sink moves in limited and predefined paths inside the sensing field. The authors in [104] provided a mechanism for sending the sensed data to the closest future position of the mobile sink in order to guarantee the shortest data path is achieved.

Indeed, the presence of sink mobility in the sensor network will enhance the network lifetime and avoid hot-spot regions around the sink. This thesis is motivated by the features of constrained (fixed) mobility for improving M2M network performance.

2.6 Software-Defined Networking

Traditional network devices, such as switches and routers, have control and data planes. Network intelligence is centred at the control plane in which the decision what to do with incoming traffic made. On the other hand, the data plane performs action on the arrived packets based on the decision taken by the control plane. The standardised protocols provided by the IETF and IEEE are usually implemented by different vendors. As a result, the same protocol may be implemented in several ways by various vendors. In addition, these vendors may add proprietary features to the standardised protocols, which results in complex configuration and their being prone to error due to vendors' diversity in manufacturing network devices. These issues make the control plane of legacy networks to be distributed. Accordingly, every node in the network is independent and does not have the complete information about the network connectivity [105].

The Open Networking Foundation (ONF) [106] is a user-driven non-profit organisation, which focuses on standards development of Software Defined Networking (SDN). SDN is an umbrella term covering different types of network architectures, where the aim has been to make the network as agile and manageable through programming. SDN can be characterised by two main features, decoupling of the data and control planes as well as adding centralised programmability in the network control plane. The decoupling feature of SDN provides greater control and management to network resources by programming the control plane. The centralised programmability brings a new innovation to optimise network configurations and improve its performance by instantaneously monitoring its resources and apply user defined policies [107]. A comparison between SDN and traditional network architecture is summarised in Table 2.2.

Table 2.2: Comparison between SDN and traditional networking

Features	Software Defined Networking	Traditional Networking
Methodology	Centralised protocol by separation of data and control plane	Dedicated protocol for each problem
Configuration	Automated and centralised configuration	Manual configuration
Control	Cross layers and dynamic global control	Single layer and static control
Implementation	Software based environment and new ideas implemented in software	Hardware based environment and limited implementation of new ideas due to hardware difficulty

2.6.1 SDN Architecture

The idea of programmable networks has been developed as new way to facilitate network management and control. SDN was developed to enable simple management and control of the network data-path. It is a new networking paradigm in which, as aforementioned, the control plane is decoupled from data plane. Since network control is no longer included in each network element, SDN introduces a new component: the centralised SDN controller. The separation of the forwarding hardware from the control logic allows the network administrator to deploy more new protocols and applications straightforwardly in the network controller [108]. Figure 2.7 shows SDN architecture, where the network management is logically centralised in software-based controllers (the control plane), and network devices become a hardware forwarding device (the data plane) that can be programmed via an open interface (e.g., OpenFlow) [109].

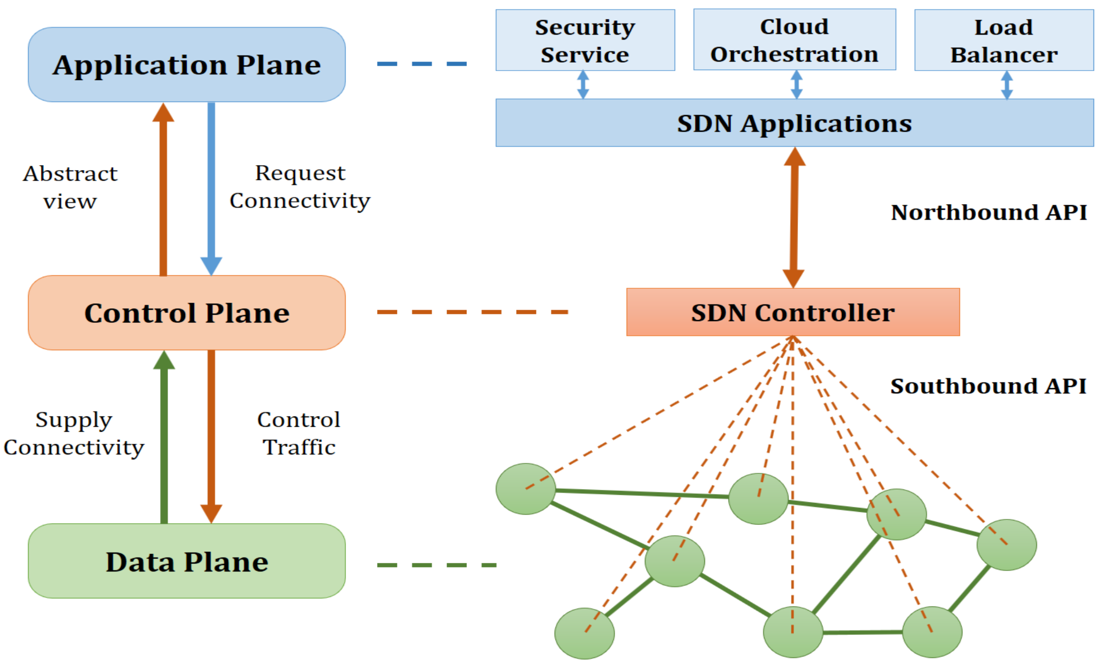


Figure 2.7: High-level SDN architecture

When the control plane is decoupled from the data plane of networking devices, the whole network architecture is changed. Accordingly, the SDN architecture consists of three main components [107].

1. The data plane pertains to the physical network forwarding devices (network elements), such as switches and routers, which provide network connectivity. Network elements are used to route the data in the network based on the flow tables supported by the SDN controller through the southbound Application Programming

Interface (API). These southbound APIs are used by the controller to optimise and configure the network.

2. The control plane or the centralised controller is the brain of the SDN networking infrastructure. It is responsible for managing the flow control (forwarding rules) in SDN networking devices, based on the desired performance requested from the SDN applications. The SDN controller configures the network elements via the general southbound API, and provides an abstract view about the network to the SDN applications via the northbound API. The SDN controller brings intelligence to the network architecture by enabling network programmability and management.
3. The application plane are programs that exchange information with the SDN controller via northbound APIs. These applications construct an abstracted network infrastructure based on the information collected by the SDN controller. The advantages over the current state networks, is that this enables new protocols to be deployed, tested and the network being customised by only replacing the applications. In sum, network behaviour can be altered via software with SDN, because it is fast, inexpensive, and easy to replace or update.

OpenFlow defines the southbound communication protocol that enables the SDN controller to communicate directly with the SDN networking devices. In an OpenFlow environment, the routers and switches should support OpenFlow protocol to exchange information with the SDN controller and it is considered as one of the first SDN communication protocol standards. The proprietary southbound APIs can be also defined by the users to customise the SDN controllers for a particular application [110] [111].

2.6.2 SDN in Wireless Networks

The topics M2M, SDN, NFV and cloud computing are the hot research areas of recent years and much research has been focused on how to integrate them together to achieve seamless connectivity and to balance the traffic load across the network. Extended SDN to M2M sensor networks were considered to be unpractical due to the resource constrained features of the embedded IoT devices in terms of small memory size, limited computation and low energy consumption. Most M2M devices in the IoT environment format the transmitted packets close to 127 bytes, while the current OpenFlow packets are close to 1,500 bytes. In addition, the addressing scheme of an M2M sensor network is unlike that used by IP-based networks. Consequently, the current OpenFlow protocol is not suitable for M2M sensor networks, because of the addressing scheme and the limited resources. In order to make the SDN concept valuable and applicable to M2M sensor nodes in the IoT environment, there is a need to propose an unprecedented customised SDN controller with an OpenFlow approach to overcome the limitations of M2M sensor networks outlined earlier.

The approach proposed in [112] represented the first effort that synergised SDN and WSN, being aimed at tackling the inherent problems of WSN, such as network management and policy changes. The authors developed a new architecture called Software Defined-WSN (SD-WSN) with Sensor OpenFlow (SOF) to address the key technical challenges in WSN. The authors in [113] presented the deployment of OpenFlow technology in WSN, the proposed approach being called the Flow-Sensor, which led to a considerable achievement in IoT and cloud computing through network virtualisation. In an ideal scenario, the Flow-Sensor had reachability points more than the typical sensor. The authors concluded that better results might be achieved in a large scale network. The authors in [114] introduced a Software Defined Wireless Network (SDWN) that benefited from a wireless infrastructureless networking environment with special emphasis on WPAN. They analysed SDN in IEEE 802.15.4-based WPAN and discussed the SDWN requirements to adopt flexibility in flow table rules and a node's duty cycle.

The authors in [115] proposed an improved load balancing mechanism for M2M networks using SDN. The proposed mechanism was capable of adopting different QoS requirements by dynamic traffic rerouting and instant traffic identification. The authors in [116] proposed a smart management technique based on SDN for WSN. The proposed architecture illustrated the implementation of the SDN controller in the base station (sink node). Also, the authors argued that the developed approach offered a new solution for some inherited problems, such as energy saving and network management.

It was one of the thesis objectives to investigate different challenges related to SDN and M2M sensor networks in order to develop a real-time testbed. Accordingly, there is a focus on the most recent research that covers SDN testbed implementation. The authors in [117] suggested a cost effective implementation of an SDN testbed using Raspberry Pi and Open vSwitch (OVS). This testbed was validated using OpenFlow specification 1.0 and proved to maximise the network throughput compared to NetFPGA-1G. The authors in [118] introduced Software Defined Networking - WIreless SEnsor networks (SDN-WISE) to reduce packets exchange between the nodes and SDN controller, as well as to make the nodes programmable for running different applications. The APIs of the SDN-WISE allowed the developers to build SDN controllers using the preferred programming language. The SDN-WISE prototype was implemented using a real

SDN controller and OMNet++ simulator. An OpenFlow testbed was proposed in [119] using a low-priced computer board and open source base virtual switch, the testbed being called the Pi Stack Switch. The programmable network was implemented using an ONOS SDN controller and OpenVirteX as the network hypervisor. The implemented network infrastructure consisted of the SDN control layer and the virtualisation layer. A structured and hierarchical management mechanism was proposed in [120], based on SDN for WSN. The proposed approach was called Software-Defined Clustered Sensor Networks (SDCSN), which the authors argued solved the inherent problems in WSN and they highlighted some suggestions for future research on ad-hoc networks. A real time Software Defined Wireless Network (SDWN) testbed was implemented in [121] using Raspberry Pi as OpenFlow Switches. An OpenDayLight controller was used to analyse the network events. In addition, a traffic aware routing algorithm was implemented to manage and monitor the network flow and Quality-of-Service (QoS) requirements. The hybrid approach proposed in [122] enabled the traditional IP network to work together with SDN based network within the same service provider. This system was called OSHI, being implemented using pseudo wire and virtual switches with a Mininet emulator. Finally, the authors in [123] provided a comprehensive survey of the research challenges in large-scale SDN testbeds.

2.7 Network Function Virtualisation

In the near future, the vast number of embedded devices connected to the Internet will need a new architecture in order to adopt the massive quantities of generated traffic. Network Function Virtualisation (NFV) technology endeavours to virtualise network applications or services in order to be executed on a single programmable component. NFV has drawn considerable interest from both industry and academia as important technology towards virtualisation of network applications. It reduces operating and capital expenses, whilst also enabling the deployment of different services across the network by decoupling network functions from the physical network devices on which the functions run and new services can be deployed faster over the same physical platform. NFV enables the behaviour of networking devices to be modified during operation, and hence, replacing and upgrading software is much easier than doing so for all network devices [105] [124].

The NFV increases network infrastructure flexibility and reduces hardware cost, because NFV it sets out to accomplish network functions in software installed on a shared server instead of running on dedicated hardware devices. Accordingly, NFV will simplify, organise and expand network services more quickly with less installation cost.

Figure 2.8 shows the basic concept of NFV, where Network Function (NF) has been implemented apart from the network devices hardware. The Virtual Network Function (VNF) in the NFV technique is similar to Physical Network Function (PNF) in traditional networks. Multiple PNFs can be assembled into a single VNF or a single PNF can be divided into multiple VNF. The relationship between VNF and PNF could be one to one mapping, or one to many. These mapping relationships can be optimised to enhance network resource management [125] and consequently, NFV may be an adequate technology for future network infrastructure in terms of the following [126]:

- Network Performance: NFV architecture might be able to obtain the same network performance compared to that achieved from network functions running on dedicated hardware by evaluating network deadlocks and mitigating them;
- Heterogeneity Support: the big challenge to NFV is to support network heterogeneity from proprietary hardware based service perspectives and fragmentise the barriers to synchronising different standards;
- Dynamic Resource Allocation: NFV should perform different network functions at various times on the same physical hardware by reallocating the shared infrastructure resources among the hardware and software components of the network.

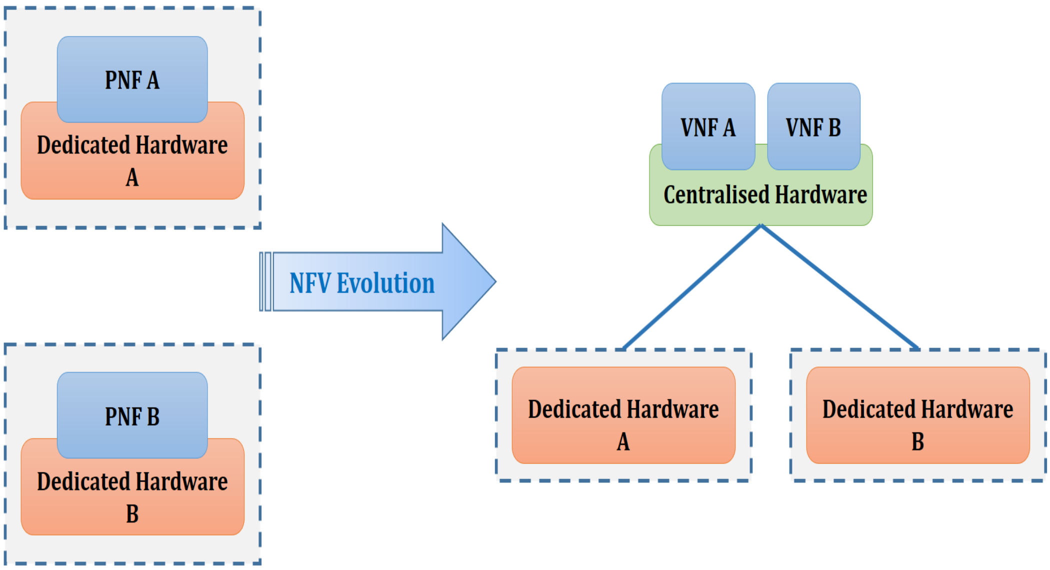


Figure 2.8: Basic NFV architecture

NFV and its relationship with the complementary fields of SDN and cloud computing discussed in [125] and [127], with a comprehension of the state-of-the-art being provided in both works. A case study on NFV based gateways for Virtualised Wireless Sensor Networks (VWSN) was proposed in [128]. The authors in [129] reviewed some general SDN/NFV-enabled IoT architectures and identify promising research directions for future research.

The NFV has changed the network vision. It moved away from customised hardware with pre-packaged software to any generic hardware that offer the basic data processing resources for network functions implementation. The abstraction layer that provided by the NFV enabled the commercial off the shelf hardware to be integrated together in the NFV infrastructure while emulation or simulating the hardware platform in software. The key benefits of utilising NFV can be summarised as follows [124]:

1. Faster deployment of network services: the VNF can be added, modified and removed on the fly. NFV provides dynamic lifecycle to the VNF in contrast to the traditional physical devices, since these virtualised functions may be added when necessary, analysed easily and then removed without any on-site activities which can be time consuming and costly;
2. Extensibility and flexibility: the virtualised infrastructure manager is able to extend the VNF resources such as bandwidth, storage and CPU depending on the virtualised function demands. In contrast to traditional network, it may require to be upgraded the existing hardware or fully replaced in order to change any of the aforementioned parameters. On the other hand, the NFV enhances network flexibility by dividing the load among the available VNFs which is unpractical in traditional network devices;
3. Vendor independent and agile deployment: the NFV provides seamless integration of different vendors hardware without relying on costly solutions associated with changing the current deployed hardware. Such agile deployment of various services should base on open source software tools to facilitate its support and popularity;
4. Amorphous service delivery: the NFV services are not limited to a one-time design and deployment. It can alter during network operation to track the demand changes in order to allocate the proper VNF. In NFV-based deployment, it is possible to change the capacity and location of the VNFs and accordingly, load mobility will be attained;
5. Effectiveness of unprecedented solution: traditionally, service providers should replicate their production in order to build a test environment for the new solutions. While in NFV-based deployment, the service providers can test their new solutions and validating them in test environment before introducing them as a final production. The NFV test environment provides dynamic tool to verify different scenarios.

2.8 Cloud Computing Platform

The term cloud computing refers to the delivery of computing resources over the Internet. The data are stored in a remote location instead of locally and can be accessed via the Internet. The cloud has no borders and has global communication paths. Accordingly, the data can be accessed from everywhere to deliver a service to other people from anywhere else [130]. In recent years, cloud computing and IoT are emerging technologies that have led the evolution of programmable networks. Cloud computing means accessing stored data and programs over the Internet, while IoT simplifies the way in which large amounts of data are being collected over interconnected M2M nodes. Cloud computing is characterised by having on-demand service, global network and shared pool of resources. On-demand service refers to the user requesting to manage his own computing resources. The global network provides ubiquitous connectivity over the Internet to deliver different services. The shared pool of resources allows the user to fetch data from the shared resources located in remote data centres [105].

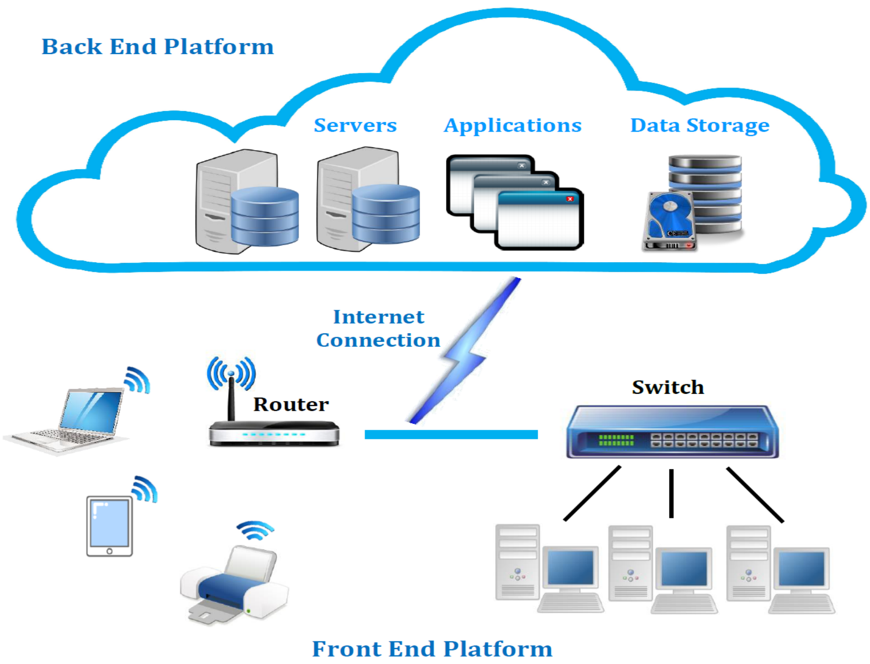


Figure 2.9: Cloud computing architecture

The cloud computing architecture is illustrated in Figure 2.9 in which the shared pool of resources can provide different services for the connected users. The cloud computing architecture consists of three main parts [131]:

1. The front end platform represents the visible part of the network for the client, including the infrastructure of the network that is used for accessing the cloud system;

2. The connected network pertains to the Internet connection that provides global access between the front end and back end platforms;
3. The back end platform refers to the service provider side, where various servers and data storage systems are installed with different dedicated applications to manage each server.

The main advantages of cloud computing are providing scalability to the network shared resources in terms of processing and storage, delivering reliability by allowing access to the cloud resources via the Internet, and it is considered to be efficient, because it enables the deployment of new algorithms and applications for delivering new services for remote M2M sensor networks [105]. There are three service models for cloud computing, commonly known as: Software as a Service (SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS), which differ in terms of the control privilege that the users have over the stored information [132]:

1. Software as a Service (SaaS) is a software distribution model, where the software is owned by the cloud provider and it enables remote users to access and use applications that are hosted in the cloud;
2. Platform as a Service (PaaS) is a shared hardware and software platform on which the users are enabled to deploy their own applications with certain constraints;
3. Infrastructure as a Service (IaaS) is virtualised computing resources over the Internet, where the users can manage and control the cloud applications, storage and network connectivity without the ability to control the cloud infrastructure.

Different deployment models of cloud computing are existed depending on the operation of cloud infrastructure. These models are classified into [133] [134]:

1. Public Cloud is based on the standard model of cloud computing platform in which the computing and storage resources are available to the public user over the Internet. The service provider offers the cloud services for free or pay-as-you-go charging model. The main advantages of using a public cloud deployment are: ultimate scalability, cost effective, reliability and location independence. The most salient examples of public cloud are: cloud storage services and online software applications;
2. Private cloud is a particular model of cloud computing in which an organisation is responsible for managing the entirely cloud infrastructure. The main advantages of using a private cloud deployment are: ultimate security, high privacy and more control as a private cloud is only accessible by a specified client. While the downside on the private cloud be expensive due to platform scaling;

3. Hybrid cloud is a combination of one or more public cloud provider with a private cloud platform. There is a need for orchestration between the two platforms while being worked individually. The main advantages of a hybrid cloud deployment are: dynamic workload, flexibility, cost effectiveness and security.

This thesis addresses a triple transition related to the evolution of SDN, NFV and cloud computing for 6LoWPAN based M2M sensor networks. This work focuses on IaaS in a cloud computing environment, i.e. where multiple independent nodes access the cloud via border gateway and various types of application are being deployed to achieve certain level of QoS by the remote user. The cloud computing provides real-time data processing of the sensors' data and taking suitable actions on regular basis. Hence, the cloud computing offers reliable and extendible environment for the WSN via the Internet. The authors in [135] developed a cloud based system to simulate WSN routing protocols, this cloud platform allowing for different simulation platforms to be available among multiple users for testing different WSN hardware platforms. A mobile cloud computing platform was proposed in [136] for Collaborative Location based Sleep Scheduling (CLSS) of the sensor nodes. The CLSS reduced node energy consumption through the determination of the awake and asleep status of each node based on the location of the mobile user.

2.9 Summary

This chapter commenced with 6LoWPAN architecture and issues related to its integration with the Internet. In addition, the chapter has comprised the detail of technologies, which enable the IoT. Then a comprehensive literature review has been included to pave the way to the research area. The related works have been studied with respect to energy consumption and scalability, as they were the most significant challenges that affected the performance of the M2M sensor networks. The discussion has been extended to include the modelling and optimisation of MAC and PHY layers' parameters of the IEEE 802.15.4 standard. A brief description of the concepts of M2M sensor networks and some routing protocols have been analysed in order to design a new self-organised, scalable and energy-efficient routing protocol for autonomous IoT applications. Programmable network concepts have been revised and explained in detail alongside with the cloud computing. The SDN and NFV were utilised to tackle the issues of energy consumption from different perspectives and enhance network control and management. The integration of cloud computing with sensor networks provided an easy way to access the sensors' data and enhanced networks' reliability.

Chapter 3

Optimised 6LoWPAN MAC Layer

3.1 Relation of 6LoWPAN to other IEEE 802.15.4 Trends

There are many different trends that need to be considered when considering the development of the IoT, which include the IEEE 802.15.4 compliant protocols, future Internet and M2M communication. Nowadays, the IEEE 802.15.4 is a standard used by the LoWPAN devices for lower protocol layers. However, problems emerge when presenting the upper layers of the protocol stack. To address this, ZigBee Alliance [137], an industrial group, developed the ZigBee protocol in 2003 as an IEEE 802.15.4 compliant protocol and specified the vertical upper layers of the protocol stack. The ZigBee protocol has suffered from many limitations including the dependency on a single wireless link and application profile, along with scalability and Internet integration. The term future Internet was introduced in [138] [139] to depict the Internet architecture and protocols research in the next 20 coming years. There are several European projects targeting future Internet research (i.e., EU 4WARD [140]), but are not focusing on embedded Internet devices and LoWPANs. Internet integration was not considered in traditional LoWPAN, because it was thought to be completely isolated. However, the EU SENSEI project [141] has focused on the integration of embedded devices with 6LoWPAN functionality in the current and future global Internet. Traditional M2M devices include cellular modems along with an Internet based back end system for IP communications. Recently, the M2M gateway has been used to bridge local embedded networked devices with IP based networks and 6LoWPAN can be connected to the Internet via this gateway. 6LoWPAN standardisation is encouraging both the research community and industry to become involved with the IoT revolution.

Wireless M2M sensor networks are usually composed of hundreds of energy constrained and short range communication devices. These limitations affect the selection of one protocol stack over the others. In fact, the increasing interest in M2M sensor networks has led to the development of a range of different communication protocols, but their diversity has limited the integration of different networks. Regarding the MAC and PHY layers, a widely used solution has been offered by the IEEE 802.15.4 standard and the IPv6 because the IP layer will cope the isolated network integration problems. In this chapter, the focus is just on the MAC and PHY layer of the 6LoWPAN protocol stack based on the specifications released by the Internet Engineering Task Force (IETF) working group [13].

3.1.1 IEEE 802.15.4 MAC Layer

The main function of the MAC layer is to assure reliable data transmission over the shared channel specified by the PHY layer. Two services are provided by the MAC layer of the IEEE 802.15.4 standard: management service and data service. The management services include: associate and disassociate sensor nodes with the PAN coordinator, channel access mechanism, beacon management, Guaranteed Time Slot (GTS) frame verification, acknowledgement frames generation (if used), and security. Moreover, the data service allows the MAC layer data unit to be transmitted and received through PHY layer data services interaction. The IEEE 802.15.4 standard defines two different channel access mechanisms, namely: beacon enabled (slotted) and non-beacon enabled (unslotted). The nodes in the IEEE 802.15.4 standard access the wireless channel using a Carrier Sense Multiple Access / Collision Avoidance (CSMA/CA) mechanism [142].

The IETF working group has introduced the IEEE 802.15.4 standard as the MAC and PHY layers standard with a fully asynchronous enabled mode for 6LoWPAN [13]. Whilst the beacon enabled mode of the CSMA/CA is used by IEEE 802.15.4/ZigBee, which is beyond the scope of this thesis.

The unslotted CSMA/CA mechanism of 6LoWPAN is illustrated in Figure 3.1; multiple devices are allowed in the IEEE 802.15.4 standard to access the communication medium using the same operation frequency. According to the CSMA/CA mechanism, at any particular time when a node has data to send, it initiates a set of variables, namely, the backoff counter (it is initialised to 0 before each new transmission attempt) and backoff exponent (*BE*), with the BE pertaining to the number of backoff periods a node will wait before checking the channel state. Two parameters are related to *BE*: minimum backoff exponent (*macMinBE*) and maximum backoff exponent (*macMaxBE*). The

MAC layer activities are delayed by a random backoff timer in each backoff stage and has a value in the range $[0, 2^{BE} - 1]$. Once the backoff timer expires, the sensor node will check the transmission channel to see if it is busy by performing a Clear Channel Assessment (CCA). When the node is assured that the channel state is idle, it initiates the transmission. Alternatively, the node increments the backoff counter and updates the BE (*BE=min (BE+1,macMaxBE)*). Then, the node backoffs for a random duration to check the channel state again. The channel checking process and backoff will be continued until the sensed channel becomes clear and transmission is started, or the maximum allowed value for the backoff stages (*macMaxCSMABackoffs*) is reached [143].

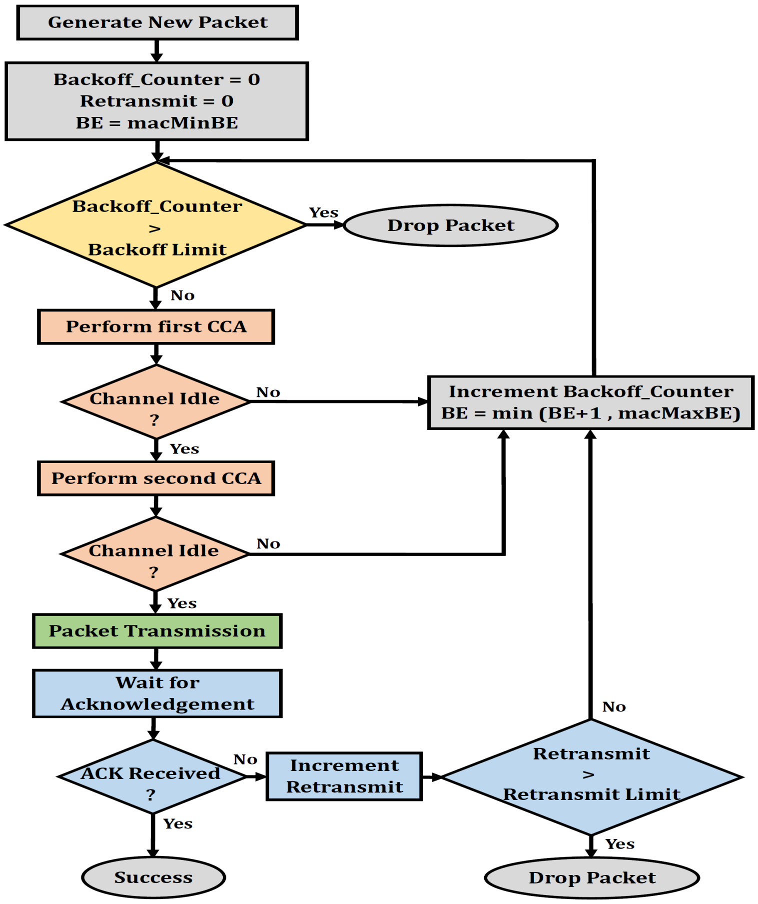


Figure 3.1: Block diagram of an unslotted CSMA/CA for 6LoWPAN [143]

The optional retransmission scheme, based on acknowledgement and timeout, is adopted by the CSMA/CA algorithm. In the case of the enabled retransmissions, the destination node must send an acknowledgement whenever it receives an error free packet. While on the sender side, the node will retransmit the previously sent packet, if the acknowledgement is not received within a predefined timeout. Accordingly, the retransmissions will be performed until the maximum number of allowed retransmissions (*macMaxFrameRetries*) has been reached and the packet will be dropped [144].

3.1.2 IEEE 802.15.4 PHY Layer

The physical layer is the closest layer to the node's hardware and provides the communication interface with the physical media. The PHY layer of the IEEE 802.15.4 standard provides two services: data service and management service. The data service allows the physical layer to transmit packets received from the MAC data service unit, while forwarding the received packet across the wireless channel to the MAC data services unit. On the other hand, the management services include [142] [143]:

1. Activate or deactivate of the radio transceiver: three states exist depending on the request of the MAC management service unit, namely: transmit, receive and sleep;
2. Energy Detection (ED): the physical layer estimates the energy level depending on the Received Signal Strength (RSS). The ED service can be used by the upper layers (MAC and network layers) to determine whether the communication channel is idle or busy;
3. Link Quality Indication (LQI): determines the strength/quality of the of the received packets over the physical layer. LQI may be implemented using transceiver ED, a signal to noise estimation or the integration of both techniques;
4. Clear Channel Assessment (CCA): used to evaluate the current medium activities, i.e. busy or idle. Channel assessment can be performed in three operational modes: (i) energy detection mode, in which the CCA identifies a busy medium if the detected energy is above the threshold ED value; (ii) carrier sensing mode, where the CCA identifies a busy medium if it detects a signal with the same spreading characteristics and modulation, which may be lower or higher than the threshold ED value; (iii) carrier sensing with energy detection mode, which is a hybrid mode formed by integrating the aforementioned techniques used in (i) and (ii);
5. Channel frequency selection: the physical layer provides dynamic channel selection in which the node transceiver may be tuned to a specific channel depending on the requests received from the upper layers.

Based on the Direct Sequence Spread Spectrum (DSSS), the IEEE 802.15.4 standard specifies three operational frequency bands: 868 MHz and 915 MHz, with data transfer rates of 20 and 40 kbps, respectively, whilst the 2.4 GHz operates with a data transfer rate of 250 kbps. Referring back to the 6LoWPAN specifications, the IETF defined the PHY layer frequency to be 2.4 GHz with a data transfer rate of 250 kbps, and a modulation method of Offset Quadrature Phase Shift Keying (OQPSK). The maximum PHY layer data unit is 127 bytes, including MAC and PHY overheads [145].

3.2 A Joint Model of IEEE 802.15.4 MAC and PHY Layers

In this section, a generalised mathematical model is proposed for 6LoWPAN MAC and PHY layers, which is based on the combination of the theory of Discrete Time Markov Chains (DTMCs) [146], and the theory of M/M/1/K queues [147]. The new model takes into account the finite buffer size of the node (sources and relays) in addition to the 6LoWPAN specifications. However, it is of interest to develop stochastic analytical tools, which can be used to predict the network performance accurately and at lower computation complexity. The performance indicators of the developed model are aggregated throughput and packet delivery delay.

The proposed Markov chain model notations closely follow those introduced earlier in the literature review (Section 2.3), with modification to improve the accuracy of the proposed model. According to the IEEE 802.15.4 standard, the proposed model provides significant flexibility in tuning the MAC layer parameters, which includes the number of backoffs, with or without optional retransmission, based on the received acknowledgement. This flexibility enables the developed model to adopt the implementation of the Transfer Control Protocol (TCP) and User Datagram Protocol (UDP) using the 6LoWPAN standard. The sensor node activity per single transmission attempt is modelled using a two dimensional Markov chain. The introduced Markov chain model is shown in Figure 3.2, with states given by {*s(t), c(t)*}, where *s(t)* and *c(t)* are the stochastic process and process state, respectively, at a given communication time slot *t*.

Consider a sensor node trying to transmit a packet using unslotted CSMA/CA of the IEEE 802.15.4 standard. The MAC layer initialises four parameters: backoff exponent, backoff window ($W_i = 2^{macMinBE}$), backoff counter (*M=macMaxCSMABackoffs*) and a maximum number of allowed retransmission limits (*N=macMaxFrameRetries*). When the backoff counter decrements to zero, two CCAs will be performed, followed by the packet transmission. Accordingly, any node in the network has four states and it will be in one of them at a particular instant:

1. Backoff state: the node waits in this state for random duration called backoff time or duration. When the backoff counter reaches zero, the node transfers to channel sensing state where a transmission may takes place;
2. Channel sensing state: the node is trying to access the channel and competing with its neighbour nodes;
3. Transmission states: the node is transmitting the generated packet and it includes the waiting time to receive the acknowledgement;

4. Acknowledgement state: the node performs retransmission based on the received acknowledgement or it generates new packet when the retransmission limit reached.

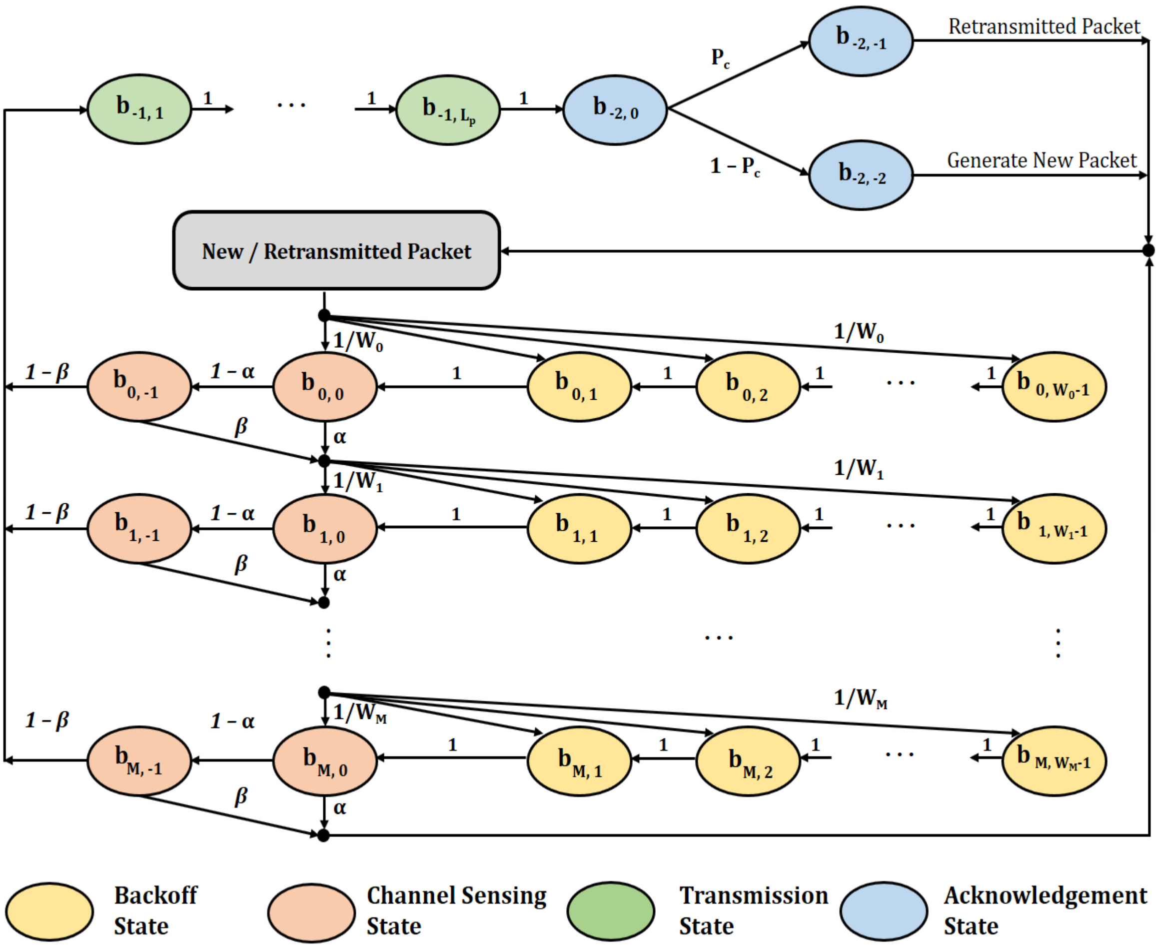


Figure 3.2: Markov chain model for CSMA/CA of IEEE 802.15.4 standard per single transmission attempt [148]

The values of *s(t)* and *c(t)* are summarised in Table 3.1 for different states of the proposed Markov chain model and L_P is the transmitted packet length. As stated earlier, this joint model is per-node transmission attempt and the other nodes influence on the activities of a given node can be recognised through the values of the busy channel probabilities.

Table 3.1: States combination of the IEEE 802.15.4 standard [148]

States		**c(t)**				
		-2	-1	0	$1 - W_{M-1}$	$1 - L_P$
s(t)	0 – M	–	CCA[2]	CCA[1]	Backoff	–
	-1	–	–	–	–	Transmission
	-2	ACK Received	ACK Lost	Wait for ACK	–	–

Let $b_{i,j}$ be the steady-state probability of the proposed discrete Markov chain model being in certain state $\{i, j\}$, while the $b_{i,j}$ = $\lim_{t\to\infty} \{s(t) = i, c(t) = j\}$. The steady-state probabilities given in Figure 3.2 are related to each other through the following set of equations which are based on the work presented by Faridi *et al.* in [148]:

$$b_{i,j} = \frac{W_i - j}{W_i} \times b_{i,0} \quad 0 \leq i \leq M; 0 < j < W_{i-1} \tag{3.1}$$

$$b_{i,0} = (\alpha - \alpha\beta + \beta)^i \, b_{0,0} \qquad 1 \leq i \leq M \tag{3.2}$$

$$b_{i,-1} = (1 - \alpha) \, b_{i,0} \tag{3.3}$$

$$b_{-1,j} = (1 - \alpha)(1 - \beta) \sum_{i=1}^{M} b_{i,0} \qquad 1 \leq j \leq L_P \tag{3.4}$$

$$b_{-2,0} = (1 - \alpha)(1 - \beta) \sum_{i=1}^{M} b_{i,0} \tag{3.5}$$

$$b_{-2,-1} = P_c \times (1 - \alpha)(1 - \beta) \sum_{i=1}^{M} b_{i,0} \tag{3.6}$$

$$b_{-2,-2} = (1 - P_c)(1 - \alpha)(1 - \beta) \sum_{i=1}^{M} b_{i,0} \tag{3.7}$$

where, α is the probability of sensing the busy channel during the first CCA (CCA^1), and β is the probability of sensing the busy channel during the second (CCA^2), given that it was idle in CCA^1. Accordingly, the probability of sensing the channel free (ϵ) in two successive time slots is given by:

$$\epsilon = (1 - \alpha)(1 - \beta) \tag{3.8}$$

The P_c is the collision probability per-node transmission attempt, for which a given node is transmitting a packet and one or more nodes within its transmission range are also transmitting their packets at the same time. The P_c can be evaluated based on the free channel probability, as depicted in the equation below:

$$P_c = \frac{1 - (1 - \epsilon)^{M+1}}{1 - \epsilon} \times b_{0,0} \tag{3.9}$$

The packet discard probability ($P_{discard}$) can be defined as the fraction of generated packets that are not transmitted successfully. Regarding the per-node transmission attempts, each may terminate with one of the following: failure due to channel access, failure due to collision or there is successful transmission. The $P_{failure}$, $P_{collision}$ and $P_{success}$ are the channel access failure probability, collision probability and successful probability, respectively, which are related to each other by the following relationships:

$$P_{failure} = (\alpha + (1 - \alpha) \times \beta)) \sum_{i=1}^{M} b_{i,0} \tag{3.10}$$

$$P_{collision} = P_c \times (1 - P_{failure}) \tag{3.11}$$

$$P_{success} = (1 - P_c)(1 - P_{failure}) \tag{3.12}$$

The node may adopt i retransmission attempt with $1 \leq i \leq N$, to achieve successful transmission. Consequently, a packet may discard if all the N^{th} retransmission limits end in collision or if one attempt ends in channel access failure. The packet discard probability ($P_{discard}$) is given by:

$$P_{discard} = P_{dis-limit} + P_{dis-failure} \tag{3.13}$$

where, the $P_{dis\text{-}limit}$ is the discard probability due to reaching the maximum retransmission limits, while the $P_{dis\text{-}failure}$ is the discard probability due to channel access failure.

$$P_{dis-limit} = P_{collision}^{N+1} \tag{3.14}$$

$$\begin{aligned} P_{dis-failure} =& \sum_{i=0}^{N} P_{failure} P_{collision}^{N+1} \\ =& P_{failure} \times \frac{1 - P_{collision}^{N+1}}{1 - P_{collision}} \end{aligned} \tag{3.15}$$

Finally, the packet discard probability can be evaluated using the formula:

$$P_{discard} = P_{collision}^{N+1} + P_{failure} \times \frac{1 - P_{collision}^{N+1}}{1 - P_{collision}} \tag{3.16}$$

On the other hand, the reliability probability $\mathbb{R}$ is the probability of the packets being delivered successfully, which can be viewed as a function of the traffic conditions (generated packets). $\mathbb{R}$ can be derived from Eq. 3.16.

$$\mathbb{R} = 1 - P_{discard} \tag{3.17}$$

In order to investigate the performance of the MAC and PHY layers in IEEE 802.15.4 standard, a comprehensive and accurate model is needed, with minimal complexity. The proposed model enables us to consider real packet losses due to finite buffer size and generated traffic.

The proposed PHY model was inspired by the approach adopted by Zuniga [149] and Zamalloa [150]. The developed PHY model involves introducing the distance as a function of neighbour discovery and node coverage. Node heterogeneity is a common feature of IoT applications. Consequently, the radio frequency and the channel parameters are introduced in order to take into account the network heterogeneity and to adopt the differences in node types with different transmission range.

A contribution of this paper is to include the real conditions of packet loss/discard in both MAC and PHY layers. Whilst collision is the main cause of packet loss, However, packets discarded due to a full buffer at the node also affect overall network performance depending on the generated traffic. Each node in the network is modelled to have a finite size of M/M/1/K queue as a buffer. This buffer has a Poisson arrival process (λ frame/s), and steady state probability when there are i frames in the queue is:

$$P_i = \frac{\rho^i}{\sum_{j=0}^{K} \rho^j} \tag{3.18}$$

where, the ρ is the queue utilisation and the value of P_0 is given by:

$$P_0 = \left[\sum_{j=0}^{K} \rho^j \right]^{-1} \tag{3.19}$$

The process continues until the value of P_0 converges to a stable value. The average queue waiting time is a function of the average frame delay and the full buffer probability P_k.

$$T_{wait} = \frac{L_P}{\lambda \times (1 - P_k)} \tag{3.20}$$

On the other hand, the average service time ($T_{service}$) can be interpreted as the time interval spent by the packet to reach the head of the MAC layer queue until the packet is transmitted and as acknowledgement for that packet received.

$$T_{service} = T_{backoff} + T_{collision} + T_{transmission} \tag{3.21}$$

The $T_{backoff}$ is the time interval when the packet is in the backoff state, while the T_{stage} is the time delay in a single backoff step.

$$T_{backoff} = \sum_{i=1}^{M} \sum_{j=1}^{W_i-1} T_{stage}(i, j) \tag{3.22}$$

The $T_{collision}$ is the time interval experienced by the transmitted packet, but without having received an acknowledgement due to collision, while T_{tx} is the delay needed to transmit a packet of length L_P and the $T_{ack\text{-}out}$ is the acknowledgement time-out to receive an acknowledgement.

$$T_{collision} = \sum_{i=1}^{N} T_{tx}^{i} + T_{ack-out} \tag{3.23}$$

$T_{transmission}$ is the time interval experienced by the transmitted packet and an acknowledgement being received, whilst T_{ack} is the delay needed to receive an acknowledgement packet of length L_{ack} and T_{IFS} is the inter-frame spacing interval between the transmitted packets.

$$T_{transmission} = \sum_{i=1}^{N} T_{tx}^{i} + T_{ack-out} + T_{ack} + T_{IFS} \tag{3.24}$$

The following assumptions are made to reduce the computational complexity with acceptable tolerance during the network metrics calculations:

- The busy channel probabilities of CCA^1 and CCA^2 will not depend on the backoff stage;
- The number of packet retransmissions will not depend on the busy channel probabilities α and β.

The network reliability is not only affected by the packet discard probability given in Eq. 3.16, but also, by the probability of having a full buffer. The final reliability expression is given by:

$$\mathbb{R}' = (1 - P_k) \times (1 - P_{discard}) \tag{3.25}$$

The network throughput γ_{NET} can be evaluated by the total number of successful transmissions per node in the network, where ϑ is the total number of nodes:

$$\gamma_{NET} = \vartheta \times L_P \times (\alpha + (1 - \alpha)\beta)^{N-1} \tag{3.26}$$

The instantaneous throughput per-node $\gamma_{instant}$ depends on the generated packets arrival rate and the successful transmissions:

$$\gamma_{instant} = \lambda \times \mathbb{R}' \times L_p \tag{3.27}$$

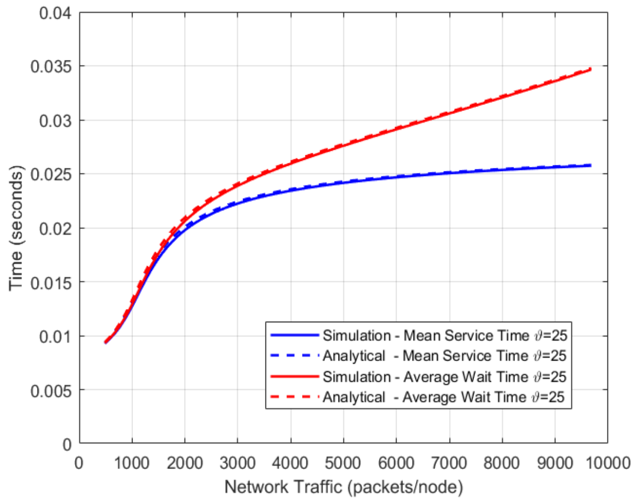
3.3 6LoWPAN Mathematical Model Validation

In order to validate the proposed mathematical model, a Monte-Carlo simulation [151] of the IEEE 802.15.4 standard is developed in MATLAB. The MAC and PHY layer parameters are chosen to meet the 6LoWPAN specification. Since the IEEE 802.15.4 CSMA/CA scheme can be initiated with a fixed backoff duration, it is also possible to consider a new slot in each simulation step with several values for network traffic. The simulation parameters for both the MAC and PHY layers of 6LoWPAN are given in Table 3.2. The network model is considered to be of a fixed number of sensor nodes, each with a finite buffer and always having packets to transmit in unslotted CSMA/CA. In addition, the proposed model enables retransmission based on acknowledgement. In this section, the obtained results from the proposed mathematical model are denoted by analytical whilst the results from Monte-Carlo simulation are denoted by simulation. The simulation results are used as references in comparisons.

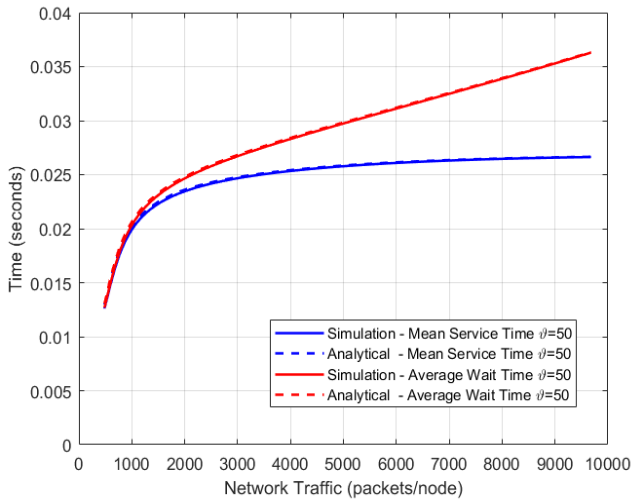
Table 3.2: 6LoWPAN MAC and PHY layers parameters

MAC Layer Parameters	**Value**	**PHY Layer Parameters**	**Value**
No. of nodes	25, 50, 100	Noise figure	23 dB
macMinBE	3	Bandwidth	30 kHz
macMaxBE	5	Pathloss exponent	4
macMaxCSMABackoffs	4	Noise	15 dB
macMaxFrameRetreis	3	Min transmission range	1 meter
Payload size	121 byte	Max transmission range	30 meter
Queue size	100 Frame	Standard transmission power	5 dBm
Data rate	250 kbps	PHY overhead	6 byte
Acknowledgement size	11 byte		

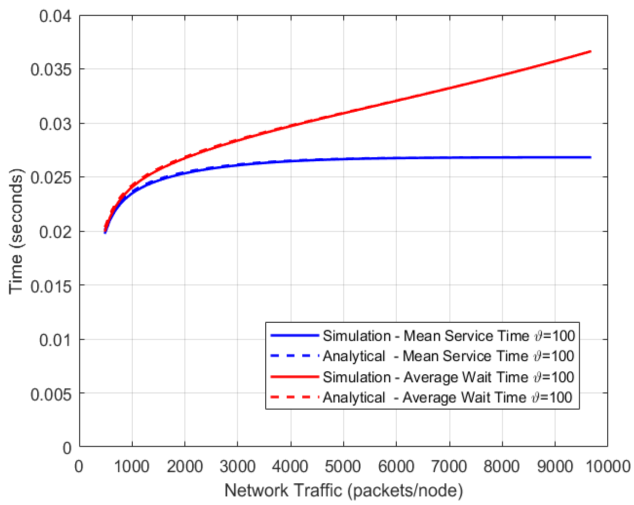
The mean service time is the time interval between two successive packets until the first packet transmitted. The overall packet service time is composed of the time slots necessary to transmit a packet and the time slots spent during the channel access mechanism. The average wait time is, in fact, the time spent in the device's queue during the elapsed packet service time. Figure 3.3 shows the deviation between the mean service time and average waiting time for both the analytical and simulation approaches. This deviation becomes larger as the network traffic is increased. When the network has light traffic the number of collisions is likely to be low. Frequent collisions are more likely to occur when the traffic become higher as the probability of collision rises dramatically, especially when a node has many neighbours within its maximum transmission range.



(a) Number of nodes is ϑ=25



(b) Number of nodes is ϑ=50



(c) Number of nodes is ϑ=100

Figure 3.3: The mean service time and average wait time at MAC queue

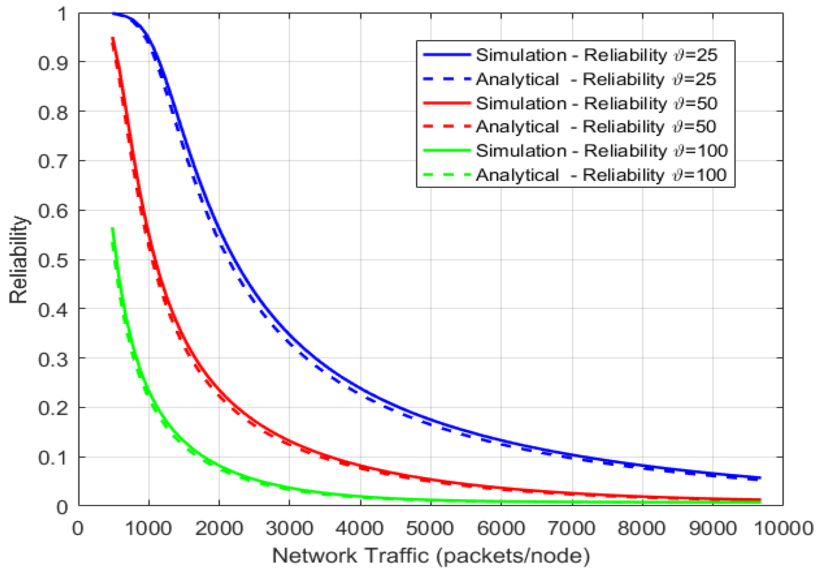


Figure 3.4: Reliability versus load offered

Figure 3.4 confirms that the network reliability achieved by the proposed joint mathematical model matches the simulation results very well. The same figure also shows that as the number of nodes increases the reliability decreases, which is due to more nodes trying to access the channel at the same time and hence, there are a growing number of collisions. Accordingly, the number of discarded packet will increase due to retransmission failures and collisions in the dense 6LoWPAN sensor network.

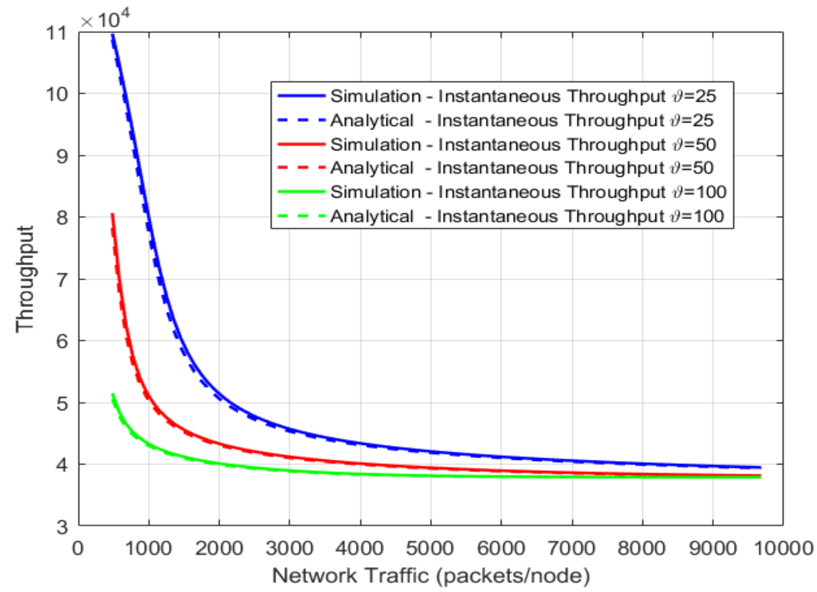


Figure 3.5: Instantaneous throughput versus network traffic

Figure 3.5 shows the instantaneous throughput versus the network traffic, with the former being affected by the frame discards after reaching the maximum frame retransmissions or maximum CSMA backoffs. This degradation in the instantaneous throughput can be interpreted as being due to the full buffer of the node as the generated traffic increases. The most interesting finding in Figure 3.5 is that with successive increases in network traffic, the analytical model tracked the simulation results, whilst no significant differences were found between them.

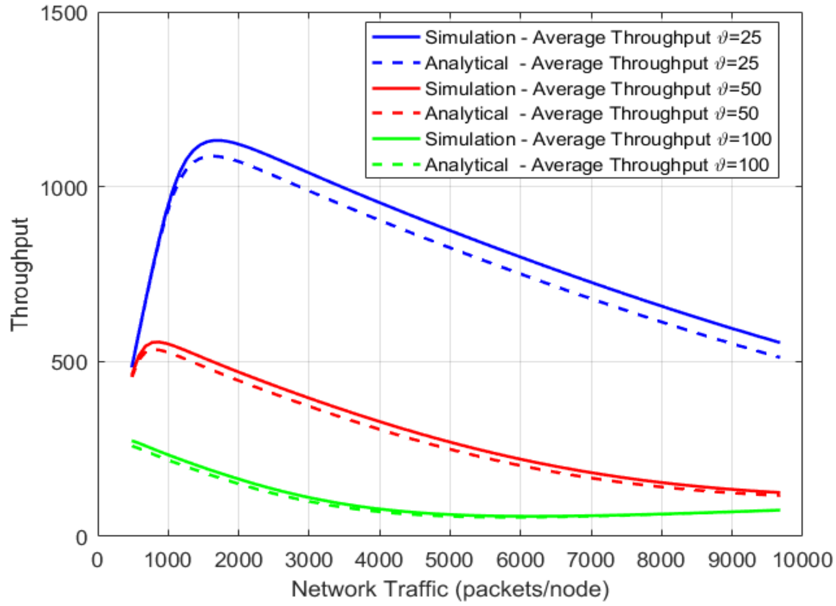


Figure 3.6: Average throughput versus network traffic

Figure 3.6 shows the average throughput in relation to the generated traffic, the curves for which become flattened as the generated traffic increases for particular nodes. The number of nodes is directly proportional to the throughput. The throughput curve starts from the peak and then quickly decays at a rate depending on the collision probability. When collisions become frequent in dense networks, packets retransmissions will affect the overall performance. In fact, collision and retransmission are inversely proportional to the network throughput.

To evaluate access channel and transmission failure probabilities, different network sizes were used, as shown in Figure 3.7 and Figure 3.8. The access channel failure probability has a greater impact compared to the transmission failure probability. As the number of nodes increase, the generated traffic will increase too and as a consequence, the higher number of nodes trying to access the channel will cause more frequent collisions. These collisions will affect the reception of acknowledgement packets and the nodes will be forced to perform retransmissions. The retransmission feature based on acknowledgement is necessary, but not suitable, for high reliability demand under heavy traffic due to high energy consumption.

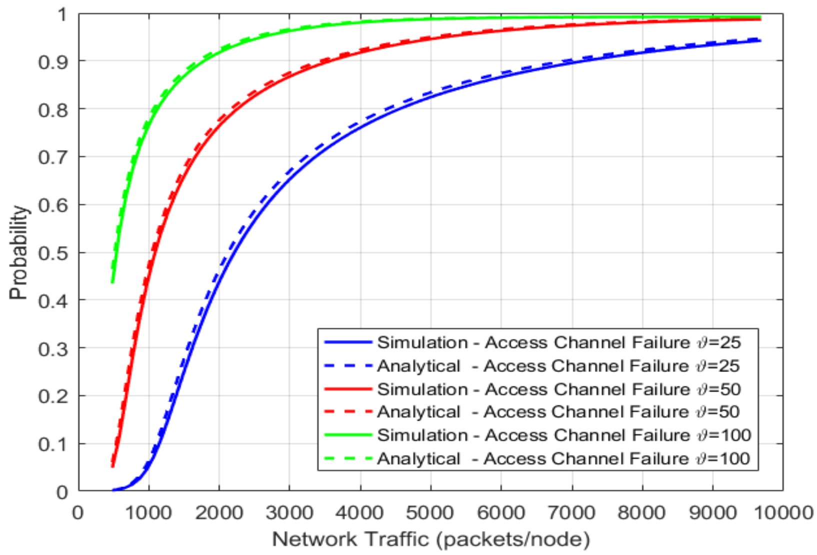


Figure 3.7: Access channel failure probability versus network traffic

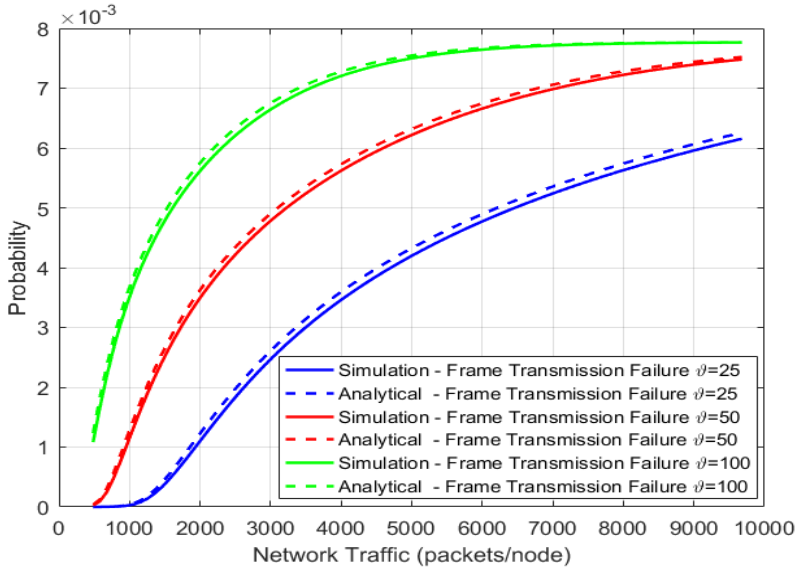


Figure 3.8: Frame transmission failure probability versus network traffic

In order to generalise the proposed mathematical model for 6LoWPAN MAC/PHY layers, the retransmission capability with acknowledgement has been studied separately. Figure 3.9 shows the instantaneous throughput vs network traffic for different transmission retries. The throughput without an acknowledgement/retransmission represents the implementation of UDP protocols, while the curves of different transmission retries represent the implementation of TCP protocols. The maximum throughput is achieved when the network operates without acknowledgement and transmission retries. A gentle degradation in overall performance occurs when the discarded packets increase due to heavy network traffic.

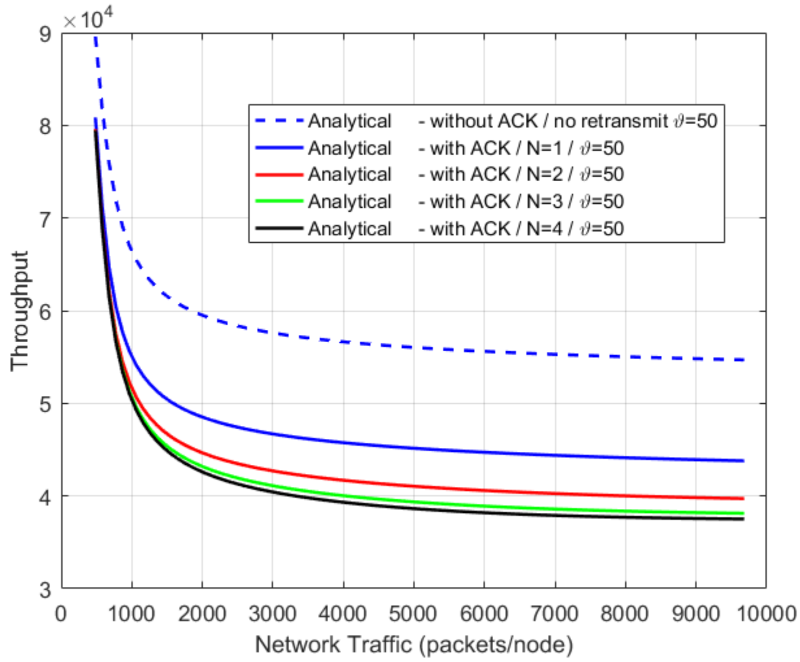


Figure 3.9: Instantaneous throughput with retransmission versus network traffic

3.4 Optimising 6LoWPAN MAC Layer Parameters

Any MAC protocol for an M2M sensor network should ensure optimal energy conservation in all M2M nodes and hence, prolong the lifetime of the nodes. This section is motivated from the previous work in Section 3.2, where the joint analytical model for the main characteristics of the IEEE 802.15.4 standard was studied and verified using a Markov chain model and Monte-Carlo simulation. The level of contention at the MAC and PHY layers influences the network throughput and end-to-end delay. In addition, the performance indicators at the MAC and PHY layers showed that the selection of appropriate MAC parameters will lead to minimise the energy consumption whilst maintaining high reliability with minimum delay. The core contribution of the proposed approach is to select the optimal MAC layer parameters, which is carried out using a) Artificial Neural Networks (ANN) and b) the effectiveness of optimiser selection mechanism. Moreover, the results are validated by using a Generic Algorithm (GA) and Particle Swarm Optimisation (PSO) to verify the selected parameters values.

Soft-Computing (SC) is one of the possible ways for building intelligent and wiser machines. It aims to model and provide solutions for existing problems that are not modelled or not easy to modelled mathematically. Accordingly, SC will achieve a robust, tractable and low-cost solution from uncertainty and approximate reasoning [152]. The techniques of SC are nowadays being used successfully in many applications and three are used in the proposed approach to determine the optimal MAC layer parameters for 6LoWPAN networks, these being:

1. Artificial Neural Network;
2. Genetic Algorithm;
3. Particle Swarm Optimisation.

3.4.1 Artificial Neural Network

Artificial Neural Networks (ANN) are a family of models inspired by biological neural networks, which can be viewed as a network of simple processing elements called *neurons*. These neurons work in harmony to provide the solution for scientific problems, such as pattern recognition or data classification, through a learning process. In general, they are composed of three layers, which are an input layer, some hidden layers and an output layer. The pool of neurons or simple processing elements communicate by sending signals to each other over a large number of weighted connections. These connections have numeric weights that can be tuned based on experience, making the ANN adaptive to inputs and capable of learning [153].

A mathematical representation of the neuron j is illustrated in Figure 3.10, which depicts the dendrite weights wn, the bias bn, the summation of weighted incoming signals, and the linear or nonlinear function $f(\cdot)$ (also called activation function). The cell inputs u are the n signals at a time instant k and the output is the scalar y, which can be expressed as:

$$y_j(k) = f\left(\sum_{i=1}^{n} wn_{i,j}u_i(k) + bn_{1,j}\right) \tag{3.28}$$

Where positive weights wn correspond to excitatory synapses and negative weights to inhibitory synapses [154].

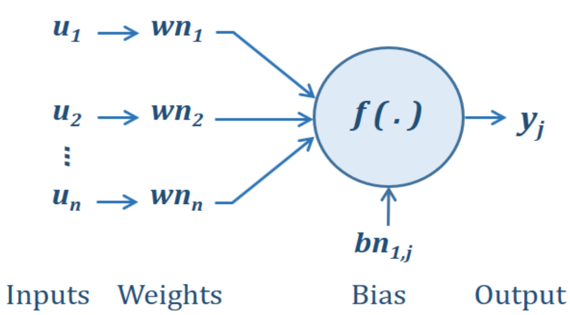
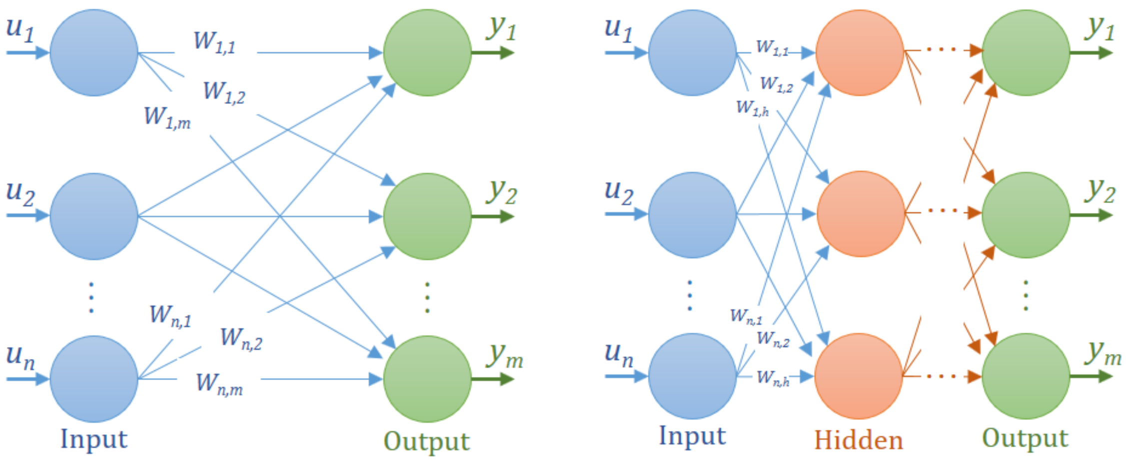


Figure 3.10: Mathematical model of neuron [154]

ANN are typically organised in layers, these being composed of a number of interconnected neurons, which contain an activation function. The input data are presented to the ANN via the input layer, which is linked to one or more hidden layers for actual data processing through a system of weighted connections. The hidden layers are then linked to an output layer where the predicted output is found [155]. The predicted output can be found by minimising the error between the ANN output(s) and the actual output(s). Figure 3.11 shows the neural network architecture, sometimes referred to it as neural network topology, for single layer and multi-layer neural networks.



(a) Single layer (b) Multi-Layer

Figure 3.11: Neural network architecture [155]

The most efficient and accurate learning process in ANN is the Feed-Forward (FF) and the selection of proper ANN topology depends on the number of neurons in the input, hidden and output layers. Moreover, there are two main approaches to make the topology selection: a) evolutionary algorithms (EAs), such as a GA or PSO; and b) exhaustive search, which is based on the neurons prediction number in each layer. The developed approach in this section is based on exhaustive search method in order to build the optimal ANN topology.

3.4.2 Genetic Algorithm

A Genetic Algorithm (GA) is a method for solving both constrained and unconstrained optimisation problems based on a natural selection process. It evolves a set of individuals, also called chromosomes, which constitutes the generational population and produces a new population. These individuals are developed according to selection rules and other genetic operators, such as mutation and crossover, with each individual receiving a measure of fitness. The selection rules focus on the individuals that have high fitness. Mutation and crossover provide an attempt to simulate the natural breeding process that simulates the reproduction process [156].

GA is implemented through the procedure described in Algorithm 1, where *ps*, *ef* and *gn* are the population size, the expected fitness of the returned solution and the maximum number of generations allowed, respectively. The procedures are repeated until the particular fitness is accepted (termination criterion is reached), or the predetermined number of iterations (generations) have been run.

Algorithm 1 Genetic algorithm

```
Require: population size ps,
         expected fitness ef,
         generation number gn,
Ensure:  the problem solution
  generation = 0
  population = initialPopulation()
  fitness = evaluate(population)
  repeat
      parents = select(population)
      population = mutate(crossover(parents))
      fitness = evaluate(population)
      generation = generation + 1
  until (fitness[i] = ef , 1 ≤ i ≤ ps) or generation ≥ gn
```

3.4.3 Particle Swarm Optimisation

Particle Swarm Optimisation (PSO) is a computational method that tries to solve complicated problems using an iterative approach to optimise a candidate's solution with regard to a given performance. The main steps of the PSO algorithm are described in Algorithm 2, where each particle has a velocity and an adaptive direction that determines its next movement within the search space. The particle is also endowed with a memory that makes it able to remember the best previous position that it passed by [157].

The PSO is formed by a set of particles, each one of which represents a potential solution to the given problem. The particle has a velocity value to indicate how much the data can be changed across position coordinates in n-dimensional search space. The PSO algorithm keeps track of three global variables to reach the target:

1. Target value or condition;
2. Global best value indicates which particle's data is currently closest to the target;
3. Stopping value indicates when the algorithm should stop if the target is not found.

Algorithm 2 Particle swarm optimisation algorithm

```
for i = 1 to n-particles do
    Initialise the information of particle i
    Random initialise position and velocity of particle i
end for
repeat
  for i = 1 to n-particles do
    Compute the Fitness_i of particle i
    if Fitness_i ≤ Pbest then
      Update Pbest using the position of particle i
    end if
    if Fitness_i ≤ Gbest then
      Update Gbest using the position of particle i
    end if
    Update the velocity of particle i
    Update the position of particle i
  end for
until Stopping condition is true
return Gbest and corresponding position
```

To update the position of each particle i, there is a set of velocities, each of which is the element that promotes the capacity of particle location and can be computed as described in Eq. 3.29, where w is called the inertia weight, r_1 and r_2 are random numbers in the interval [0,1], c_1 and c_2 are positive constants, y_{ij} is the best position (*Pbest*) found by the particle i with respect to dimension j, and finally y_j is the best position (*Gbest*) with respect to dimension j. The position of each particle is updated according to the formula in Eq. 3.30.

$$v_{ij}(t+1) = wv_{ij}(t) + c_1r_1(y_{ij} - x_{ij}(t)) + c_2r_2(y_{ij} - x_{ij}(t)) \tag{3.29}$$

$$x_{ij}(t+1) = v_{ij}(t+1) + x_{ij}(t) \tag{3.30}$$

while $x_{ij}(t+1)$ is the current position and $x_{ij}(t)$ is the previous position.

3.4.4 The Proposed Intelligent Optimisation Approach

Low energy consumption is vital in M2M sensor networks and nodes can achieve high throughput by extending the network lifetime or reducing packet drops. Packets are dropped either because the channel is busy or the maximum number of retries limit has been reached. Extension of network lifetime with reduced delay can be achieved by selecting the optimal MAC layer parameters and the detailed steps for optimising the MAC parameters are as follows:

1. Data Collection: complete data sets were collected from the proposed mathematical model in Section 3.2 for different network size;
2. Data Analysis: collected data were analysed and pre-processed prior to the training stage. the datasets are separated into inputs and outputs, and divided randomly into three subsets: training set (70%), testing set (15%), and validation set (15%);
3. ANN Training: the analysed data were fed as inputs to the ANN for complete output prediction prior to the optimisation stage;
4. Data Post-Processing and Testing: the predicted ANN output was verified with unseen raw data to validate ANN training;
5. Data Optimisation: once the ANN output was verified, two optimisation techniques (PSO and GA) were run individually to choose the optimal 6LoWPAN MAC parameters set with different network size. These optimisation techniques compared among each others to give more certainty to the optimal selected parameters set, and determine the technique's efficiency when being deployed in the developed approach.

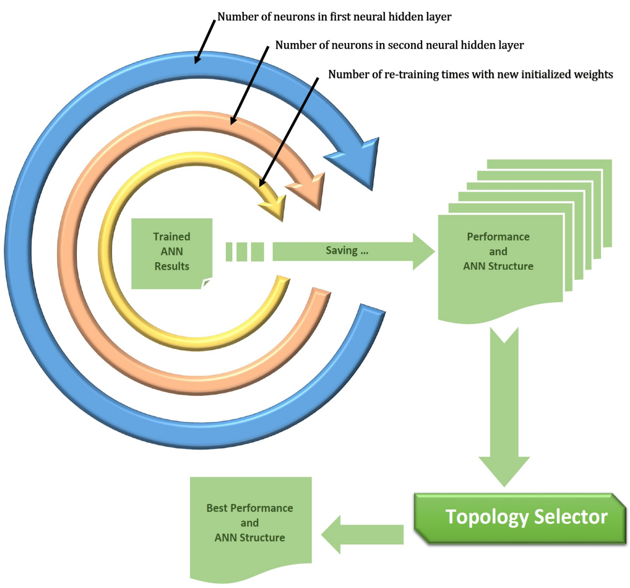


Figure 3.12: Exhaustive search method for optimal selection of ANN topology

The Levenberg Marquardt algorithm (LM) was used to search the ANN. During the training phase, the data set was first tested using a single hidden layer but, unfortunately, the training failed to give a good performance in terms of Mean Squared Error (MSE). Hence, the optimal topology for ANN was selected by conducting an exhaustive search. Multiple layer ANN were studied to find the best number of neurons in both the first and second hidden layers in a nested loop fashion, as depicted in Figure 3.12. Owing to the random initialisation of the ANN parameters (weights and biases), every selected topology was trained ten times to ensure that the network was not trapped in the local minima. The performance of the network as MSE versus the network architecture for single and double layers are shown in Figure 3.13 and Figure 3.14, respectively.

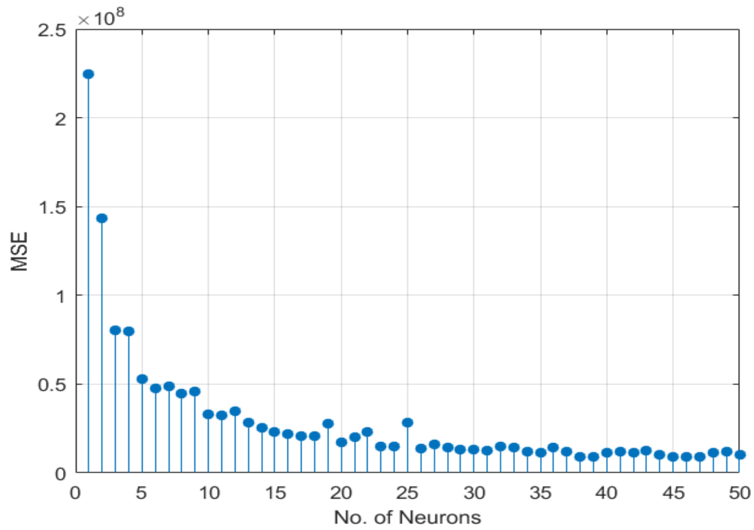


Figure 3.13: Performance of a single hidden layer ANN

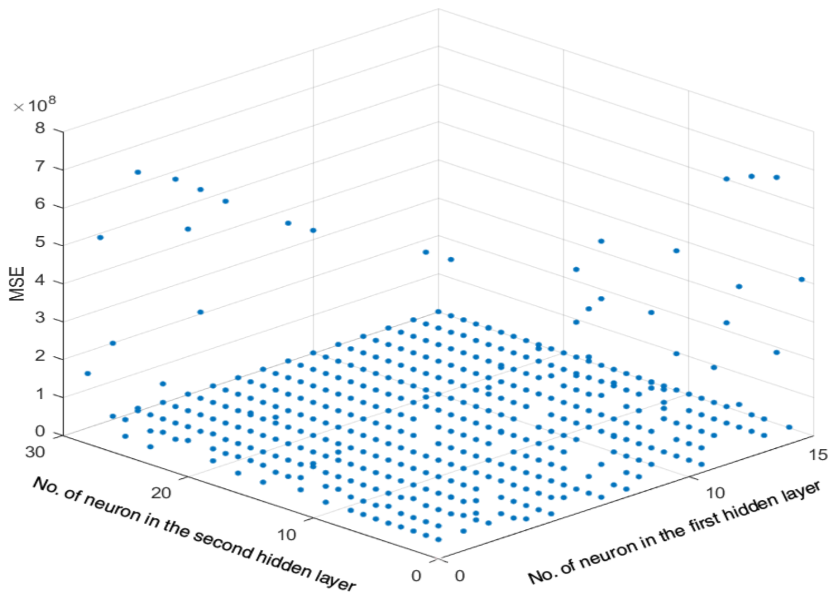


Figure 3.14: Performance of a double hidden layer ANN

The MAC parameters set was fed into the ANN as inputs in addition to the desired network size (number of nodes), while the outputs were throughput and delay. As stated earlier in this subsection, the ANN was trained using 2304 input/output pairs in order to predict the actual output and produce an accurate ANN model for the optimisation stage. The sigmoid activation function is used for the input and hidden layers while linear activation function is used for the output layer of the ANN. Whilst the objective function was to obtain the optimised MAC layer parameters that gives maximum throughput with minimum delay. Figure 3.15 shows the proposed optimising approaches for selecting the optimal MAC layer parameters set of a 6LoWPAN network. The optimiser tuned the following input parameters in order to achieve maximum throughput with minimum end-to-end delay:

- *BE*: Backoff exponent is a random number that determines a random backoff interval;
- *macMaxCSMABackoffs*: is the number of times that the node stays in the backoff stage after unsuccessful channel sensing before the packet being dropped;
- *macMaxFrameRetries*: is the number of the retransmissions limit when there is no acknowledgement received and the packet will be dropped.

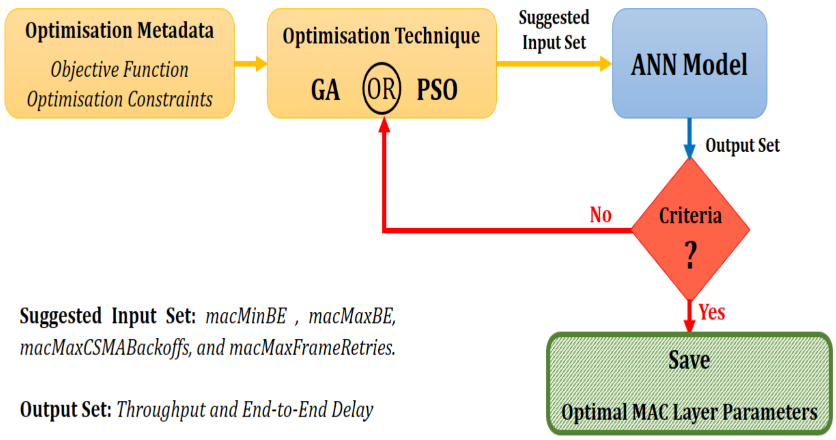


Figure 3.15: ANN-based optimiser for MAC layer parameter selection

3.5 Optimised Parameters: Verification and Discussion

After investigating the performance of different ANN architectures using an exhaustive search method, the best trained ANN with two hidden layers was reached by 15 neurons in the first and 12 in the second as shown in Figure 3.16. This ANN topology demonstrated that the MSE is less than 1.29×10^{-22}. Figure 3.17 and Figure 3.18 show the performance of the network in terms of MSE versus the number of samples in the training and testing phases, respectively.

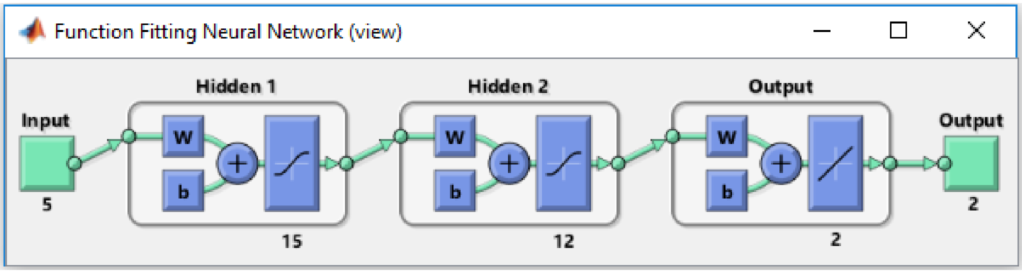


Figure 3.16: The best ANN topology obtained from the input/output pairs

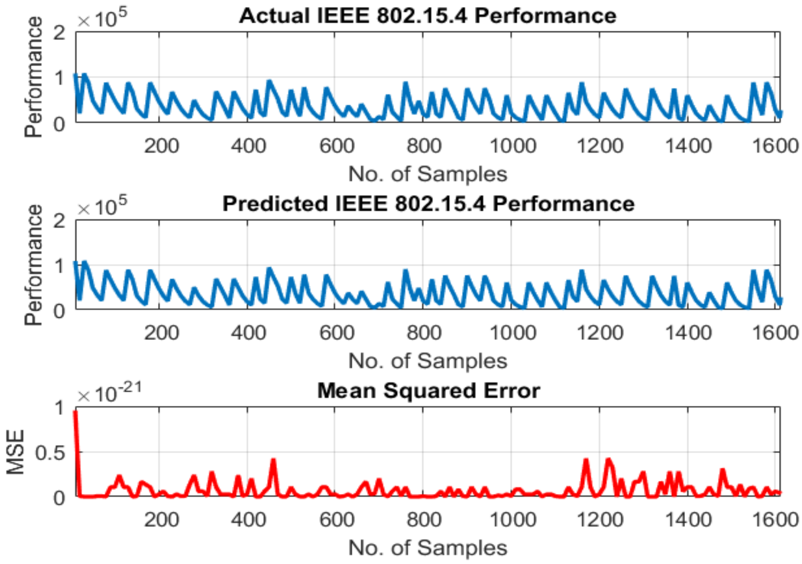


Figure 3.17: Actual and predicted output for training sets

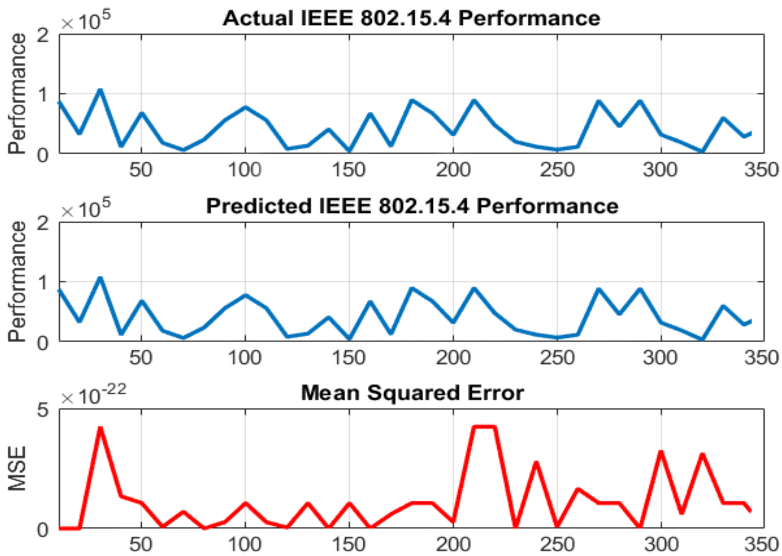
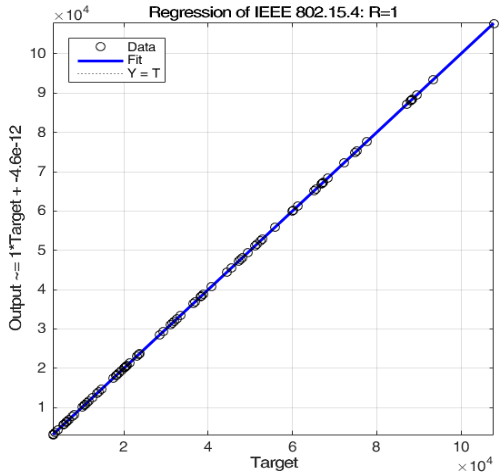
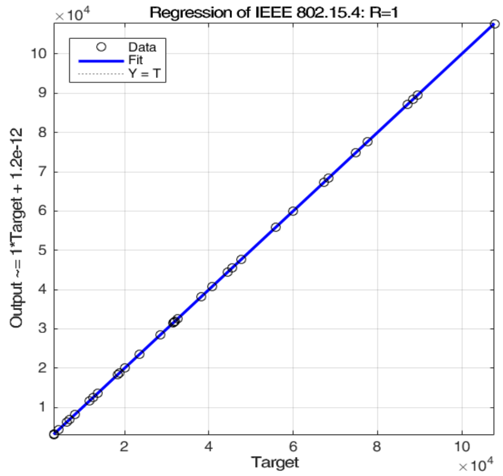


Figure 3.18: Actual and predicted output for testing sets

The results of the linear regression of the trained and tested samples are shown in Figure 3.19, with their verifying the validity of the trained ANN and its ability to predict the output of the MAC layer, which can be used efficiently to improve the performance of 6LoWPAN network by selecting the best MAC layer parameters.



(a) Targets vs outputs for training sets



(b) Targets vs outputs for testing sets

Figure 3.19: The linear regression of the ANN output

MATLAB has been used as a simulator for medium and large scale M2M sensor networks to implement the 6LoWPAN MAC layer represented by the IEEE 802.15.4 standard. A 6LoWPAN network with 50 and 100 M2M sensor nodes are considered, with the impact of each single MAC parameter being evaluated in terms of node throughput. In the conducted simulation scenario, it is assumed that the message generation process is periodic to evaluate saturated and unsaturated traffic. Figure 3.20 and Figure 3.21 are for 50 and 100 nodes, respectively and the MAC parameters observations are:

1. Impact of *BE*

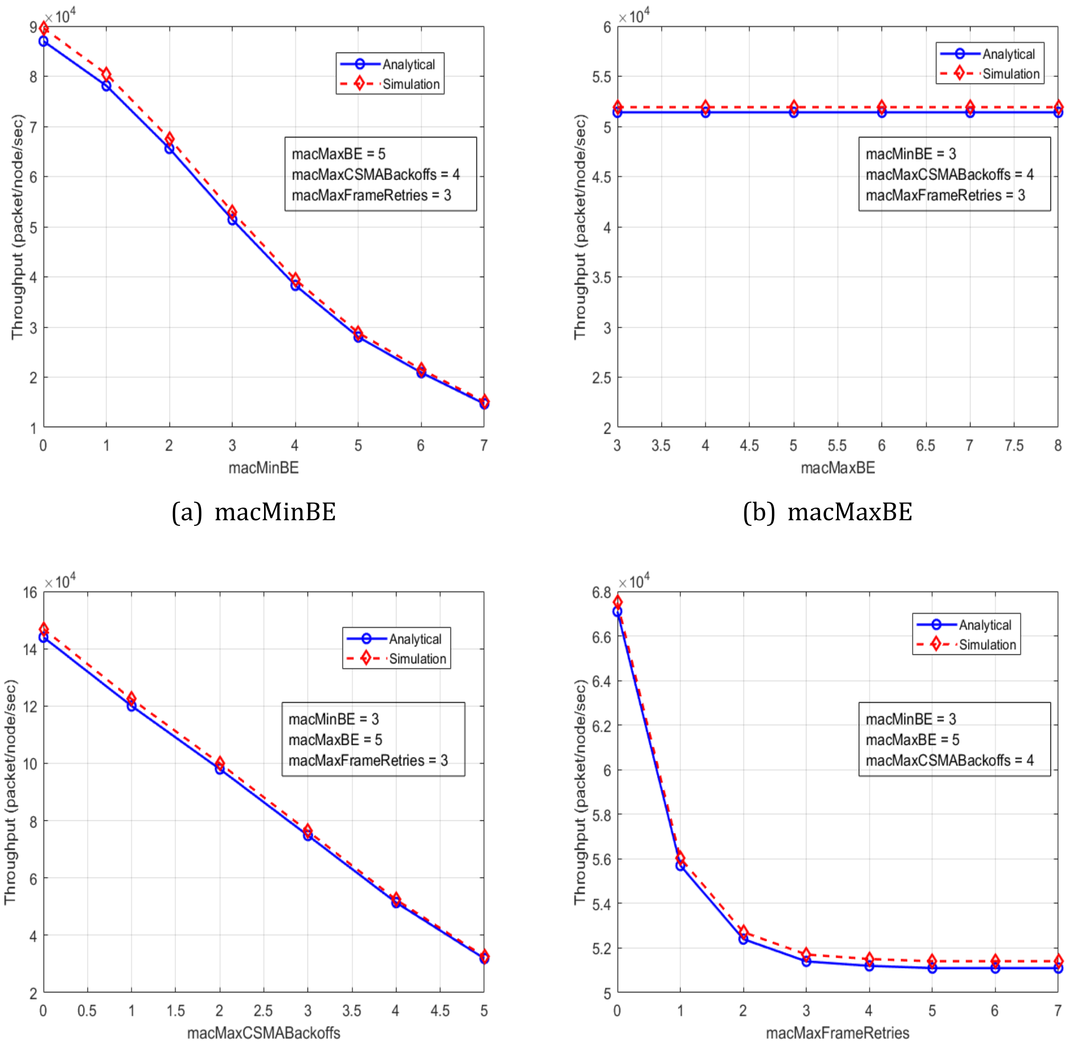
   Figure 3.17(a)(b) and Figure 3.18(a)(b) show the impact of *minMacBE* and *maxMacBE* on throughput, respectively. The *minMacBE* is in the range between 0 and 7, *maxMacBE* is in the range between 3 and 8, while the other parameters with their default values are shown in Table 3.3. For a fixed value of *maxMacBE*, the throughput tends to be improved when increasing *minMacBE*, because a larger initial backoff window reduces the collision probability in the first backoff stages;

2. Impact of *macMaxCSMABackoffs*

   Figure 3.17(c) and Figure 3.18(c) show the impact of *macMaxCSMABackoffs* on network throughput. This parameter is in the range between 0 and 5, whilst the others, are with their default values, as shown in Table 3.3. When *maxMacCSMABackoffs* value increases, the node's throughput will increase to some extends in medium size network as shown in 3.20(c), after that the throughput decreased when the traffic increases as multiple nodes try to access the channel many times and collisions occur frequently. Figure 3.21(c) shows the impact of *maxMacCSMABackoffs* in large networks, whereby the throughput decreases as its value increases, because nodes have a high probability of sensing the channel and it is busy in dense networks;

3. Impact of *macMaxFrameRetries*

   Figure 3.17(d) and Figure 3.18(d) show the impact of *macMaxFrameRetries* on network throughput. This parameter is in the interval between 0 and 7, while the others have the default values shown in Table 3.3. The throughput remains constant for the values equal to or greater than 2 in medium size networks, as shown in Figure 3.20(d) and to or greater than 3 in larger networks Figure 3.21(d).



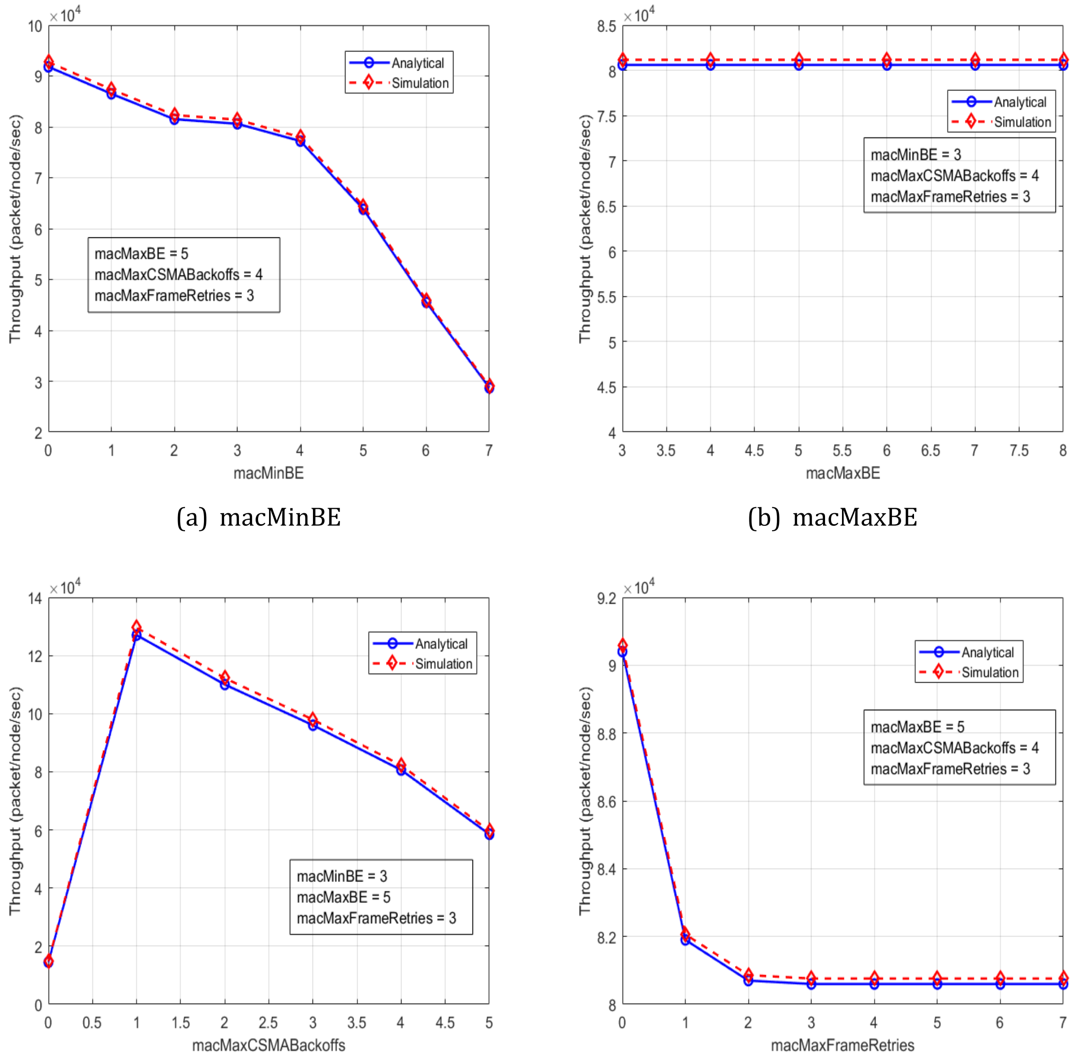
(a) macMinBE

(b) macMaxBE

(c) macMaxCSMABackoffs

(d) macMaxFrameRetries

Figure 3.20: The effect of the MAC layer parameters for 50 node network size and offered load of 1,000 packet/node



(a) macMinBE (b) macMaxBE

(c) macMaxCSMABackoffs (d) macMaxFrameRetries

Figure 3.21: The effect of the MAC layer parameters for 100 node network size and offered load of 1,000 packet/node

Table 3.3: Default and optimised 6LoWPAN MAC layer parameters range

Parameters	Value Range	Default Value	Optimised Value
macMinBE	0 – 7	3	2
macMaxBE	3 – 8	5	6
macMaxCSMABackoffs	0 – 5	4	3
macMaxFrameRetries	0 – 7	3	0 – 5

Table 3.3 shows the optimal MAC layer parameter values obtained from the two optimisation techniques (GA and PSO). The input and output sets of the ANN fed back to an optimiser running GA and PSO to predict the input set that provides maximum throughput and minimum delay. The optimiser outputs are the optimal 6LoWPAN MAC

parameters given in the last column of Table 3.3. To summarise, from the previous analysis it is concluded that *macMaxCSMABackoffs* and *macMaxFrameRetries* should set to the optimal values (and not the default ones) as the sensor nodes need to adapt optimal BE to increase the throughput.

Rather than setting the default values of the 6LoWPAN MAC layer, the optimised parameters achieve highest throughput and less service delay for a given node number, as shown in Figure 3.22 and Figure 3.23, respectively. The optimised MAC parameters enhance network throughput by 52-63% and reduced the end-to-end delay by 54-65%, depending on the 6LoWPAN network size.

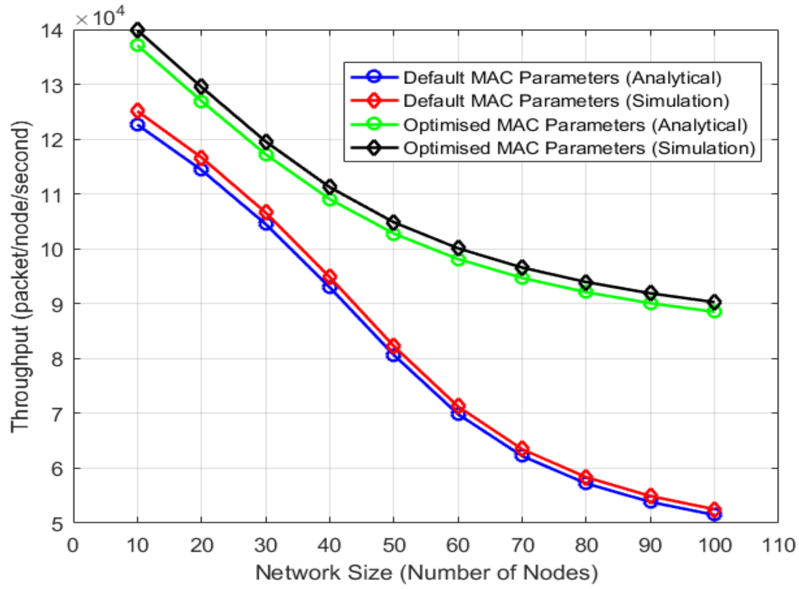


Figure 3.22: Throughput

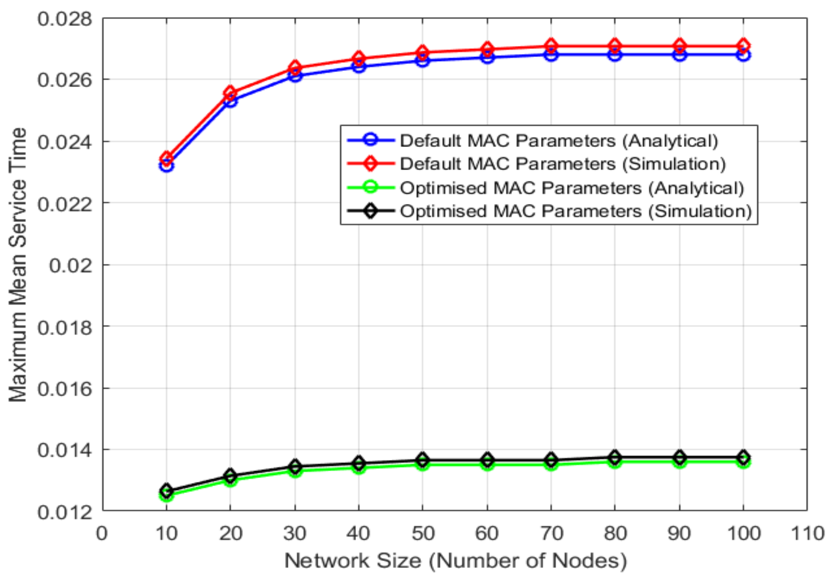


Figure 3.23: Mean service time

Figure 3.24 shows the access channel probability versus different node numbers in 6LoWPAN network. The most obvious finding to emerge from the analysis is that the reduction in access channel probability and mean service time led to enhancement of the network throughput as more packets were successfully delivered to the destination.

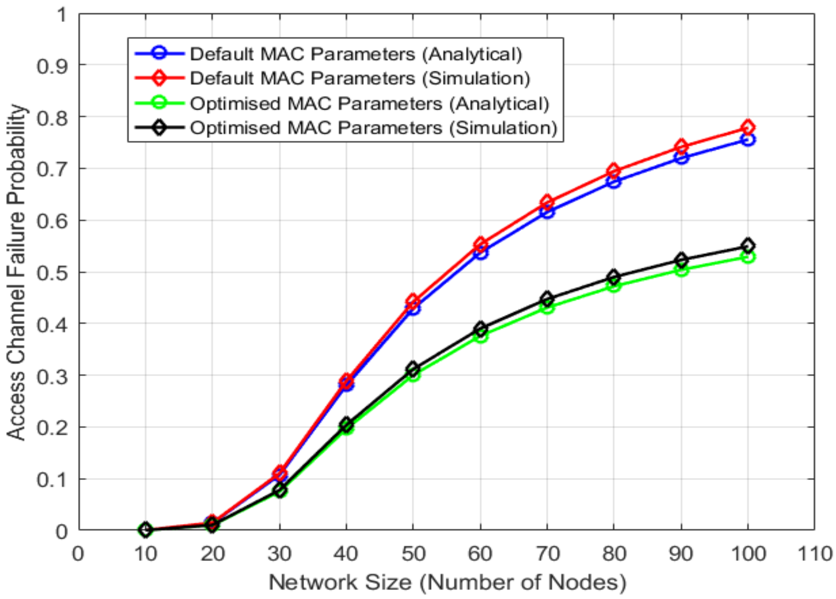


Figure 3.24: Access channel failure probability

Extensive simulations were carried out to find the optimal initial parameters for GA and PSO, like population size, initial condition, weight, etc. Due to the randomness of the initialisation stage, 10 simulation runs were performed independently of each algorithm. The metadata of the GA and PSO algorithms are given in Tables 3.4 and 3.5 respectively. The performance for both GA and PSO are shown in Table 3.6 and Figure 3.25, its clear that the PSO-based optimisation indicates better achievement regarding the convergence speed as well as the computation time than with GA.

Table 3.4: Metadata of GA-based optimiser

Parameter	Value
Population size	100
Maximum number of iterations	1000
Function tolerance	0.0001
Crossover fraction	0.8
Migration fraction	0.2

Table 3.5: Metadata of PSO-based optimiser

Parameter	Value
Number of particles (size of the swarm)	100
Maximum number of iterations	1000
Cognitive acceleration (c_1)	1.2
Social acceleration (c_2)	0.12
Momentum or inertia (w)	0.90

Table 3.6: Comparison metrics of GA and PSO

Algorithm	Total Iteration	Convergance	Computation Time (sec)
Genetic Algorithm	1000	327	72
Particle Swarm Optimisation	1000	85	39

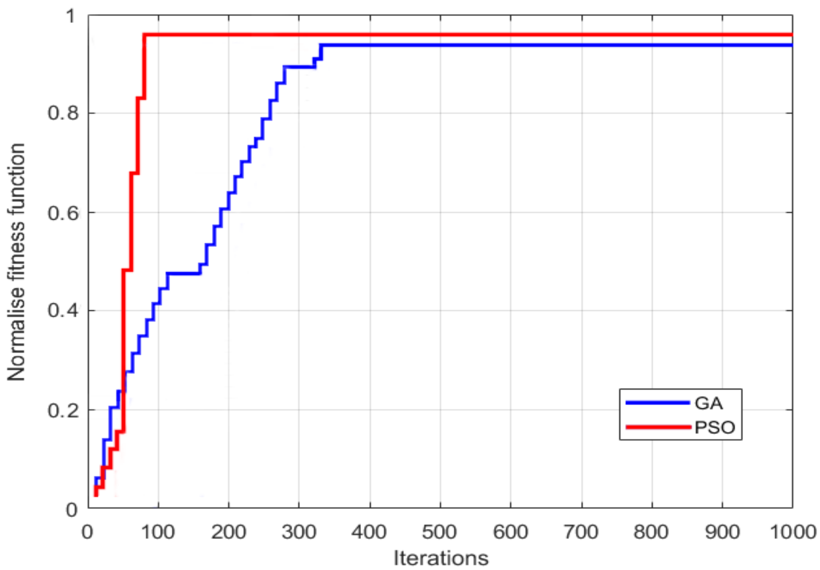


Figure 3.25: Fitness function convergence of GA and PSO optimisers

In contrast to the conventional optimisation methods, the EAs have been chosen because: (i) the EAs start searching from the population of points rather than single point of search in conventional optimisation methods; (ii) the EAs is able to find the optimal solution for discrete and non-differential functions that are existed in the real-life optimisation problems. On the other hand, the ANN has been adopted to approximate the objective function for the evolutionary optimiser. This technique will eliminate the calls to the mathematical model that are computationally expensive because of the very large number of fitness function evaluations needed for a given input parameters vector. The evolutionary optimisation with approximate fitness function based on neural networks will avoid the convergence to false optima and computationally cheaper.

As the Internet traffic increases, the energy utilisation becomes one of the most promising approaches that need to be considered in order to improve energy efficiency and reduce energy waste. The 6LoWPAN nodes are generally battery powered, and hence, energy efficiency is one of the key issues of 6LoWPAN network. Figure 3.26 shows the total remaining energy versus simulation rounds for a 6LoWPAN network consists of 100 nodes with 0.5 J per node. Compared with the default MAC layer parameters' set, it is obvious that the optimised MAC layer parameters effectively prolong the node's working time and therefore, the overall lifetime of the network extends. In relation to the previously obtained results, the proposed MAC layer optimisation scheme succeeded in prolonging the 6LoWPAN network lifetime by 40%, in addition to throughput enhancement and end-to-end delay reduction.

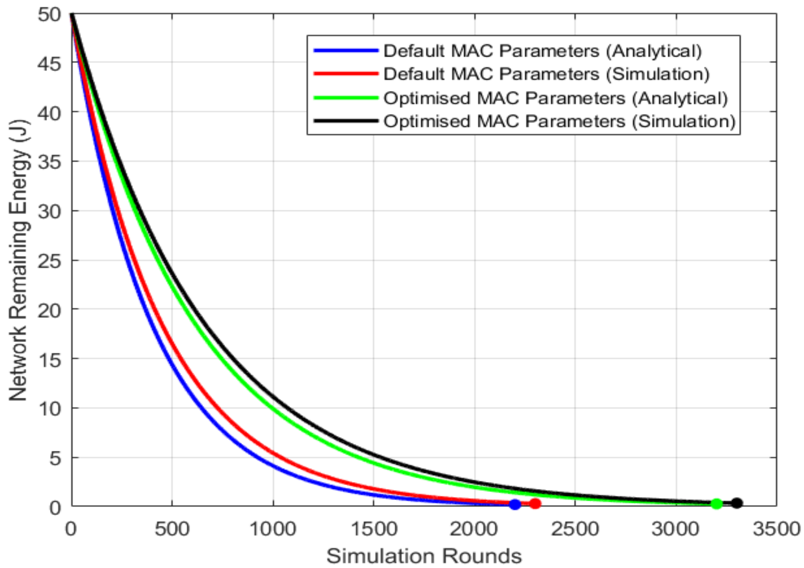


Figure 3.26: Residual network energy

3.6 Summary

In this chapter, a joint mathematical model for MAC and PHY layers of the 6LoWPAN has been proposed using Markov chain modelling. The proposed model has four states: backoff, channel listening, transmission and acknowledgement states. Numerous approaches are attempting to model the IEEE 802.15.4 standard using different techniques and assumption but none of these works tried to adopt the specification introduced by the IETF working group for 6LoWPAN MAC and PHY layers. The developed model is proven to be accurate with low computational complexity as it is being compared to a Monte-Carlo simulation.

This chapter has discussed the effects of using the default parameters set for the 6LoWPAN MAC layer. These parameters are: minimum backoff exponent maximum backoff exponent, maximum backoff of the CSMA and maximum frame retransmission limits. This chapter has shown that utilising the default parameters set generates extensive collisions for the resource-constrained 6LoWPAN devices. Accordingly, a new intelligent approach is used for optimising these parameters in order to enhance the energy depletion and reduce the end-to-end delay in M2M sensor nodes. The developed approach is based on ANN to approximate the objective function and EA-based optimiser to find the optimal values without performing a very large number of fitness function evaluations. Finally, the obtained results have shown that the new intelligent approach improves the performance of the 6LoWPAN based sensor networks.

Chapter 4

Load Balancing Using Multiple Mobile Sinks in M2M Network

4.1 Energy Hole and Scalability Problems

The essential objective of wireless M2M sensor nodes is to observe valuable information by monitoring the target area and reporting the sensed information to the data collector, namely, a sink node. Each node in an M2M sensor network senses the local environment individually, but sophisticated information gathering and data propagation tasks can be achieved, if the sensor nodes work in collaborative manner. Consequently, the fundamental aim of wireless M2M sensor nodes is twofold: (1) acquire the information about the physical besetment via sensors; and (2) to collaborate wirelessly with other nodes to deliver a complete depiction about the monitoring area. Gathering the sensed data is a main task in an M2M sensor network in which the sensor readings are collected at the sink node for further processing and analysis [158]. In the static sink approach, the sensor nodes closer to the sink node run out of their battery storage quicker than those further away. The reason is intuitively obvious: the sensor nodes that are nearby the sink node, are being shared among multiple sensor to sink paths and as a result, deplete more energy. The uneven energy consumption phenomenon around a sink node is called *energy holes* and these energy holes cause network performance degradation. If sensor nodes around the sink node exhaust their energy, the sink node will be secluded and if all the sink nodes in the sensory field are isolated then the entire network becomes disconnected [159]. Accordingly, minimising and balancing energy consumption among the sensor nodes is needed. Adding mobility to the sink node can effectively enhance network performance without the aforementioned network degradation impacts [160].

As the mobile sink travels across the sensing field, the energy holes will be rotated among the sensor nodes and hence, balance the nodes energy consumption. There are two data transmission approaches: push or pull. In the push approach, the sensor nodes transmit the data to the sink, while in the pull approach, the data transmission will start upon the sink's request. Considerable enhancements in terms of energy consumption and latency can be achieved by deploying multiple mobile sinks. These significant advantages are: mitigating unfair energy consumption by sharing the data transmission load among all the sensor nodes; reducing the hop count between sensors and sink nodes by shortening the mean distance between them and; finally, adopting multiple mobile sinks in the sensory field will eliminate the single bottleneck problem in case of one sink being disconnected, thereby avoiding energy holes and the entire network ceasing to function.

Scalability is an important factor affecting the performance of energy-efficient routing protocol design. That in an M2M sensor network refers to the ability to adopt network expansion to accommodate more nodes during the network lifespan. Thus, the scalable protocol should perform well even if the network topology and node number are changed from time to time during sensor network operation. In large scale M2M sensor networks, to achieve network scalability without performance degradation, one of the energy-efficient techniques is hierarchical clustering in which the sensor nodes are grouped together to form clusters [161]. Each cluster has one cluster head and multiple member nodes. The cluster head is responsible for collecting the data from its cluster members and performing aggregation or fusion for further relaying towards the sink. Grouping the sensor nodes in clusters reduces the routing algorithm complexity, optimises the network management, optimises energy consumption in cluster members, and enhances network scalability. The cluster head will carry a heavy processing and retransmission load, which leads to depletion its energy faster than the cluster member nodes. This heavy burden issue can be addressed by rotation of the cluster head role among cluster members. From a data routing perspective, the clustering approach involves dividing the data transmission into intra-cluster and inter-cluster communications. This division leads to significant energy saving per sensor node. A cluster based routing protocol reduces intensive message exchanges to find the sensor to sink data paths [162]. In sum, the clustering technique enhances network scalability, conserves sensor nodes energy and balances the load across the network. In this chapter, the energy hole problem is solved by deploying multiple mobile sinks, while the scalability issues are addressed by proposing self-organised clustering techniques.

4.2 M2M Sensor Node Architecture

The sensor node, also known as a mote, is the main building block of the M2M sensor network. Each has the ability to monitor its local environment and disseminate the sensed data via a wireless communication link. Figure 4.1 shows the architecture of an M2M sensor node, which consists of a: sensing unit, processing unit, communication unit, and a power unit. Additional components may be added to the M2M nodes depending on the application and the purpose of their use, such as being a position finding unit or mobility unit [163] [164].

1. The sensing unit is responsible for measuring different phenomena in the surrounding environment by means of sensors. A sensor node may have one or more sensors, such as temperature, humidity or light sensors. The analogue signal generated by the attached sensors during a sensing task is converted to a digital signal by the Analog-Digital Converter (ADC) and then fed into the processing unit;
2. The processing unit is responsible for interconnecting all the other subsystems to work in harmony. It is capable of executing arbitrary codes to perform preprocessing sensor readings and data routing tasks. It includes a small size memory, which temporarily stores sensor readings and a node program code;
3. The communication unit is responsible for interconnecting the individual sensor nodes and forms a sensor network by enabling the sensor nodes to receive and transmit information over the wireless channel. It is characterised by low-power, low-data rate, and short range wireless communication;
4. The power unit is responsible for powering up the sensor nodes. It consists of a battery where the energy for node operation is stored, which determines the lifetime of the sensor node;

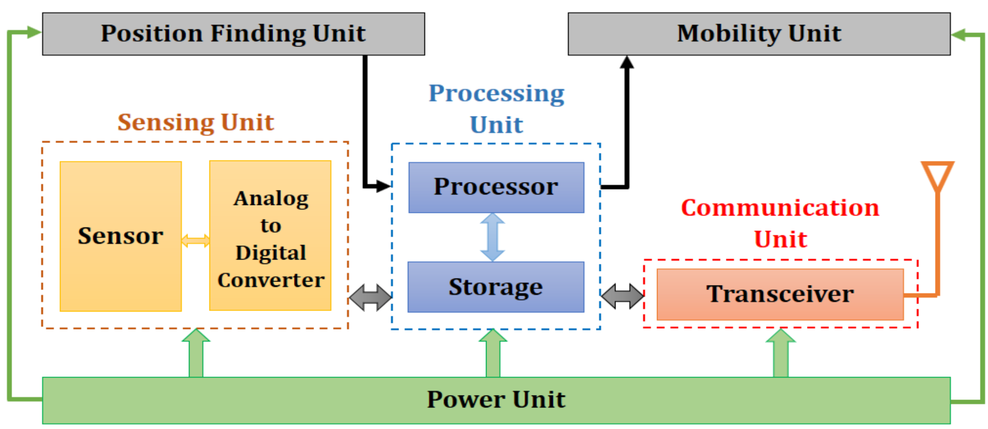


Figure 4.1: M2M sensor node architecture

5. The position finding unit is responsible for determining the physical coordinates information of a sensor node. The physical location information can be determined either by a Global Positioning System (GPS) module or a software module running a GPS-free localisation algorithm;
6. The mobility unit is responsible for sensor node movement from one location to another. It requires additional energy resources and it operates in a collaborative manner with the sensor unit, position finding unit, and processing unit to control the movement of the sensor nodes when they travel across the sensory field.

Most energy efficient routing protocols adopt a first order radio model, such as that given in [165]. Figure 4.2 shows the first order radio model adopted in this chapter to analyse the proposed scalable energy efficient M2M routing protocol with the proposed self-organised clustering technique.

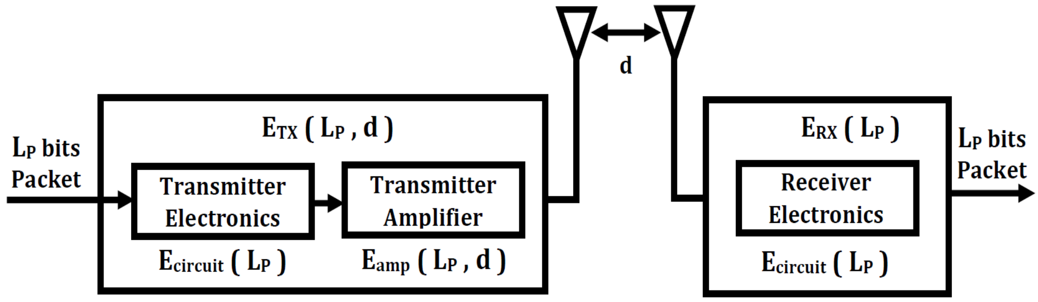


Figure 4.2: First order radio model [165]

The mathematical representation of the radio communication model given in Figure 4.2 is illustrated in Eq. 4.1 and Eq. 4.2 for both the transmitter and receiver, respectively. The communication happens only between cluster members and the cluster head, known as intra-cluster communication. There is no direct communication between cluster members and the sink nodes. The cluster head also communicates with the mobile sink node using intra-cluster communication and the mobile sink is designed to address each cluster head within a single hop.

$$\begin{aligned} E_{TX}(L_p, d) =& E_{circuit}(L_p) + E_{amp}(L_p, d) \\ =& E_{circuit} * L_p + E_{amp} * L_p * d^2 \end{aligned} \tag{4.1}$$

$$\begin{aligned} E_{RX}(L_p) =& E_{circuit}(L_p) \\ =& E_{circuit} * L_p \end{aligned} \tag{4.2}$$

where, E_{TX} and E_{RX} are the transmission and receiving energies, respectively, for L_P packet size, $E_{circuit}$ is the energy dissipated in the electronic circuit of the transmitter and receiver, while d is the distance between the cluster members and cluster head, or the distance between cluster heads and mobile sinks. The d is assumed to be fixed in the simulation, which means that the node does not change its transmission power during the simulation. E_{amp} is the is the energy dissipated in the amplifier circuit to achieve certain level of Signal-to-Noise Ratio (SNR).

For ϑ number of nodes in the M2M sensing field, there are CN cluster heads with CM cluster members in each cluster. The energy dissipation in the cluster heads can be evaluated using the formula given in Eq. 4.3 and Eq. 4.4 for transmitting $E_{CH\text{-}TX}$ and receiving $E_{CH\text{-}RX}$ cluster member packets, respectively:

$$E_{CH-TX}(L_p, d) = \sum_{i=1}^{CM} E_{circuit} * L_p + E_{amp} * L_p * \sum_{i=1}^{CM} d^2 \tag{4.3}$$

$$E_{CH-RX}(L_p) = \sum_{i=1}^{CM} E_{circuit} * L_p \tag{4.4}$$

The radio channel is assumed to be symmetric, such that the energy required to transmit a message from node A to node B is the same as that required to do so from node B to node A for a given SNR.

4.3 M2M Sensor Network Architecture

The M2M sensor network consists of multiple energy-hungry pervasive wireless nodes, a sink node (or multiple sink nodes), an external network, and an end-user. The sensor nodes can be distributed across the area of interest in a well planned manner and this approach is called regular deployment. However, this is not possible for many applications (i.e., sensor nodes may be dropped from an airplane) and such a distribution is called random deployment [166].

The deployed sensor nodes communicate with each other over the wireless channel to form the M2M sensor network to collect, propagate, and process the incoming data from the physical surrounding environment. Accordingly, the sensor nodes can act as a data source and one or more nodes act as sink nodes, where the sensed data are passed on. The sink node is the main coordinator of the network, which bridges the deployed sensor nodes and the end-user. Consequently, the sink node can be treated as a gateway, having more processing and memory resources for data processing before the sensed data is forwarded to the final destination. It may have multiple radio frequency technologies (i.e. IEEE 802.15.4 and IEEE 802.11) and use short range technology when it communicates with the sensor network, while high distance transmission is used for external network communications, such as the Internet [167].

Figure 4.3 shows the M2M sensing field that is adopted in this thesis with M2M sensor nodes deployed randomly in the monitoring area. The sensing field is divided into equal sized sub-regions to provide independent cluster head selection, solve the scalability problem and ultimately, reduce the energy consumption. The proposed network model is composed of static M2M sensor nodes with four mobile sink nodes moving in a predefined path with a constant speed. The four mobile sink nodes will collect the sensed data from cluster heads when they reach the collection points at the end of each segment in their movement paths. This can reduce the energy consumption near the sink nodes (energy holes) and enable the sensor network to last longer.

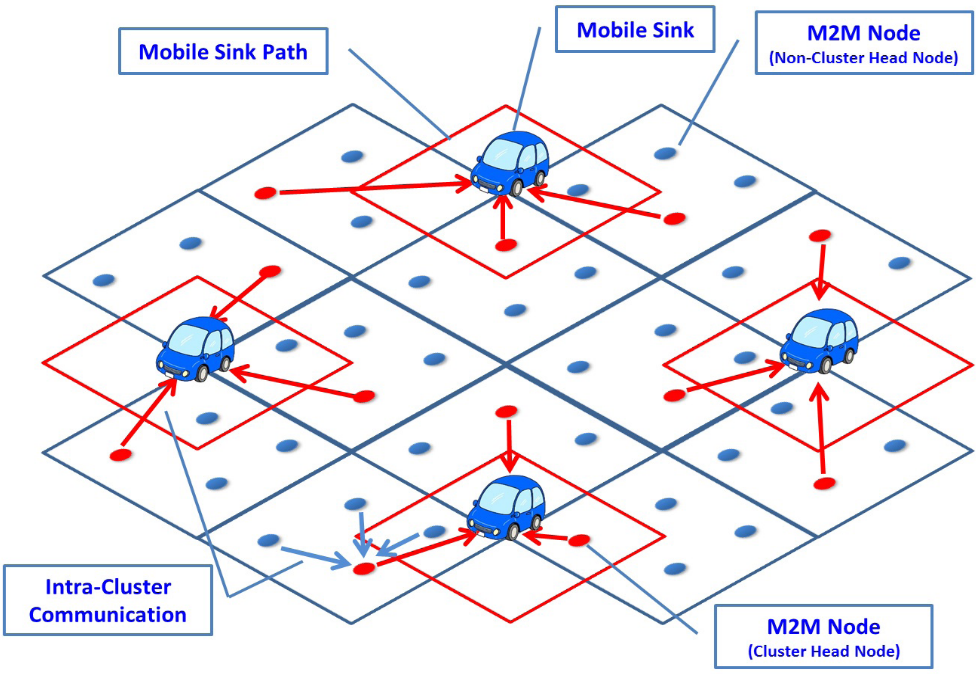


Figure 4.3: M2M sensor network architecture

4.4 Performance Evaluation Metrics

In order to evaluate the performance of the proposed self-organised clustering technique with multiple mobile sink nodes, a set of evaluation metrics needs to be defined. These metrics include [51] [62] [168] [169]:

- Network lifetime is the most important metric in evaluating the performance of the energy-efficient routing protocol and it is typically evaluated based on its definitions. Various definitions for network lifetime are available in the literature, and has a different impact on network performance evaluation. The most commonly used definitions are as follows:

- The first node depleting its residual energy and failing to transmit more packets is used to define the network lifetime. In some research, the time until the first node fails is called the stability period;
- The number of surviving nodes as a function of time is considered in network lifetime measurement, with a longer surviving node meaning longer network lifetime;
- The fraction of remaining alive nodes as a function of time is used to define the network lifetime, with the network considered to be alive if the remaining alive nodes are above certain threshold value;
- The time until the network is unable to construct a backbone. This set of nodes is changed in order to prolong the network lifetime;
- The number of nodes that are still connected to the sink node, which can be predetermined to evaluate the network lifetime;

• The number of cluster heads in each simulation round in used to evaluate the clustering technique steadiness in terms of cluster head selection and cluster head rotation. In addition, the cluster head selection and rotation criteria have a great impact on the nodes' energy, whereby fewer message exchanges during cluster formation will prolong the network lifetime;

• The network's remaining energy is used to determine the energy-efficiency of the proposed clustering technique and routing algorithm. The remaining energy indicates how the sensor nodes deplete their energy during the network lifespan and a high remaining energy level means longer network lifetime;

• Aggregated packets at the sink node are used to evaluate how many packets are received there during the network lifespan and the more packets received, means the fewer the number of packets that have been dropped.

In this chapter, the simulations are performed using MATLAB as the programming tool. The self-organised clustering and sleep scheduling techniques are implemented and compared in MATLAB simulations. First, the performance of different routing metrics (network lifetime, cluster heads number, remaining energy, and aggregated packets at the sink) are analysed and discussed for both homogeneous and heterogeneous networks. For both networks, a random node deployment is adopted in a fixed dimension sensing field with multiple mobile sinks. Finally, a sleep mode scheduling is proposed to conserve node energy, while maintaining network performance.

4.5 Self-Organised Clustering Technique

Consider the M2M sensor network given in Figure 4.3 consists of four unlimited-energy mobile sinks and ϑ energy-constraint static sensor nodes. The sinks move in predefined paths and at constant speed, while all the M2M sensor nodes are stationary and randomly deployed over the sensing field. Due to random node deployment, the M2M network topology is unexpected. The objective of the proposed protocol is to minimise the total energy consumption. In order to conserve energy in M2M sensor nodes, a self-organised clustering technique with multiple mobile sinks is used.

As previously explained, networks become disconnected when the nodes near to sinks deplete their energy more quickly than other nodes. To overcome this problem and extend the network lifetime, mobile sinks have been used to collect the data from the sensor nodes. Before the sink changes its position and continues travelling in the sensing field, it stops for a fixed amount of time to collect the data from the cluster heads within its transmission range, these stopping places being called sojourn points. The network is divided into a group of sensor nodes with a data aggregator node. The network subdivisions are called cluster and the data aggregator is called the cluster head, which is responsible for data aggregation and transmitting the aggregated data to the sink node for further processing. The proposed self-organised clustering techniques are simulated in two different network scenarios: homogeneous and heterogeneous sensor networks.

4.5.1 Homogeneous Network Clustering

In a homogeneous M2M sensor network, all the sensor nodes are identical, having the same level of initial energy and communication range. The self-organised cluster based routing protocol works in three main phases, the: cluster initialisation phase, cluster head rotation phase and finally, the data aggregation and transmission phase.

1. Cluster Initialisation Phase: when clusters created in a region, each node has equal probability to become a cluster head for the current round. The first clusters will be formed depending on the number of neighbour nodes. The node that has the maximum number of neighbour nodes will be chosen as a cluster head among the other nodes within its transmission range.
2. Cluster Head Rotation Phase: when the current cluster head residual energy reaches the predefined threshold value, an advertisement message will be generated to in-

form other nodes to nominate another cluster head for that cluster. The decision for winning the competition to become the next cluster head depends on the information matrix generated by each node at each simulation round. The information matrix includes node residual energy as the driving parameter for cluster head rotation. Then, the node with highest rotation index (R_i) given in Eq. 4.5 will become the new cluster head and form a new cluster;

$$R_i = \left[\sqrt{\left| \frac{\sigma}{1 - \sigma(r \bmod (1/\sigma))} \right|} \times E_{residual} \times CH_{counter} \right] \times M_{index} \tag{4.5}$$

where, σ is an initialisation random variable between 0 and 1, r is the simulation round, $E_{residual}$ is the node residual energy, $CH_{counter}$ is the number of times that this node has became the cluster head so far (initially set to 1), and M_{index} is the multiplication index (M_{index}=1 when $E_{residual} > E_{Th}$ and M_{index}=0 when $E_{residual} < E_{Th}$), where E_{Th} is the threshold energy for the node to continue contributing to the communication. This technique will not generate lots of control messages that deplete much energy without carrying useful information. After many rounds, a few control messages will pass across the network to generate new clusters with new cluster heads.

3. Data Aggregation and Transmission Phase: upon the completion of cluster head selection and cluster formation, the transmission of data will start. The cluster head is ready to receive all the data from the cluster member nodes. When all the data from the nodes has been received, the cluster head performs data aggregation and transmits them to the mobile sink. As the multiple mobile sink crosses every sub-region, it requires low transmission energy, and the same mechanism will be performed in every sub-region in the M2M sensor network.

4.5.2 Heterogeneous Network Clustering

The proposed heterogeneous network model assumes that there are ϑ M2M sensor nodes, which are randomly deployed in a remote sensory field. The nodes always have data to transmit towards the mobile sink nodes via the cluster heads. The M2M sensor nodes are grouped together to form a clustering hierarchy, and the cluster heads perform data aggregation and data fusion to reduce the data correlation produced by the M2M nodes within the clusters. In addition, the following assumptions are made: M2M sensor nodes are stationary; the M2M nodes are unaware of their location; and they are identical in terms of their sensing and communication range.

The M2M sensor nodes have different initial energy values, according to the network heterogeneity level. These levels of network heterogeneity are used to estimate the network lifetime, and can be characterised into the following categories.

1. Two Level Heterogeneous Network

   In two level heterogeneous M2M networks, there are two types of M2M nodes: advanced and normal. Let n be the fraction of the advanced nodes in the M2M networks and E_0 be the initial energy of the normal nodes. The advanced nodes have a times more energy than the normal ones.

   Accordingly, there are $n\vartheta$ advanced nodes with initial energy of $(1{+}a)E_0$ and $(1{-}n)\vartheta$ normal nodes with initial energy of E_0. The total initial energy (E_{total}) of the two level heterogeneous M2M sensor network can be calculated by:

$$\begin{aligned} E_{total} &= \vartheta \times (1-n) \times E_0 + n \times \vartheta \times E_0 \times (1+a) \\ &= \vartheta \times E_0 \times (1+an) \end{aligned} \tag{4.6}$$

   In sum, the two level heterogeneous M2M sensor network has *an* times more energy than the previously described homogeneous M2M sensor network, and virtually has *an* more M2M nodes.

2. Three Level Heterogeneous Network

   In the three level heterogeneous M2M network, there are three types of M2M sensor nodes: super, advanced and normal. Let n be the fraction of the advanced nodes in the M2M networks, while n_0 is the super nodes fraction of the total number of nodes n. The $(1{-}n_0)\vartheta$ is the number of advanced nodes and the remaining $(1{-}n)\vartheta$ is the number of normal nodes in the M2M sensor network.

   Accordingly, there are $\vartheta n n_0$ super nodes with initial energy of $E_0(1{+}b)$, $\vartheta n(1{-}n_0)$ advanced nodes with initial energy of $E_0(1{+}a)$, and $(1{-}n)\vartheta$ normal nodes with initial energy of E_0. The total initial energy (E_{total}) of the three level heterogeneous M2M sensor network can be calculated by:

$$\begin{aligned} E_{total} &= \vartheta \times n \times n_0 \times E_0 \times (1+b) \\ &\quad + \vartheta \times n \times (1-n_0) \times E_0 \times (1+a) \\ &\quad + \vartheta \times (1-n) \times E_0 \\ &= \vartheta \times E_0 \times (1 + n \times (a + n_0 \times (b-a))) \end{aligned} \tag{4.7}$$

   In sum, the three level heterogeneous M2M sensor network has $n\ (a + n_0\ (b{-}a))$ times more energy than the previously introduced homogeneous M2M sensor network, and virtually has $n\ (a + n_0\ (b{-}a))$ more M2M nodes.

In this section, a self-organised clustering technique has been proposed to take into account the real-time parameters that affect the formation and selection of cluster heads in heterogeneous sensor networks. The proposed approach has three stages:

1. Initialisation Stage: the number of nodes required for initialising the network is evaluated based on the total number of nodes ϑ in the network, with the nodes then being divided into two or three types depending on the heterogeneity model in which they are characterised as normal, advanced and super nodes with different energy levels. The mobile sinks' paths are determined during this stage and the sojourn points are chosen to achieve a single-hop transmission from the cluster heads to the sinks.

2. Self-Organising Stage: the first clusters will be generated and cluster head selection will take place as well. Each node starts to share its residual energy ($E_{residual}$) with the neighbour nodes *k* within its transmission range and calculates the average residual energy of the cluster ($\bar{E}_{cluster}$) prior to selecting the cluster head.

$$\bar{E}_{cluster} = \frac{1}{k} \times \sum_{i=1}^{k} E_{residual,i} \tag{4.8}$$

The average residual energy of the cluster and the residual energy of the node in the current round are considered for cluster head selection among the different types of M2M sensor nodes. The cluster head selection probability ($\mathbb{P}$) depends entirely on the available energy in the cluster and can be calculated as:

$$\mathbb{P} = \left[\frac{r}{\psi} \times \frac{E_{residual}(r)}{\bar{E}_{cluster}} \times CH_{counter} \right] \times M_{index} \tag{4.9}$$

where *r* is the simulation round, $E_{residual}$ is the node's residual energy for the current round, $CH_{counter}$ is the number of times that this node has became cluster head so far (initially set to 1), and M_{index} is the multiplication index (M_{index}=1 when $E_{residual} > E_{TH}$ and M_{index}=0 when $E_{residual} \leqslant E_{TH}$), where E_{TH} is the threshold energy for the node to continue contributing to the communication. The term ψ represents the extra energy added to the network, Eq. 4.10 and Eq. 4.11 are the extra initial energy added to the network for two and three levels of heterogeneity, respectively:

$$\psi = \vartheta \times E_0 \times a \times n \tag{4.10}$$

$$\psi = \vartheta \times E_0 \times n \times (a + n_0 \times (b - a)) \tag{4.11}$$

In each simulation round, *r*, the node *i* is eligible to be a cluster head, if and only if, it has a cluster head probability ($\mathbb{P}$) greater than the other nodes within the cluster.

3. Maintenance Stage: repeated unnecessary clustering in every round depletes the energy of the network more quickly. Accordingly, the proposed self-organised technique re-clusters the network only when the residual energy of the current cluster heads fall below a threshold level, and a broadcast message will be generated to inform the cluster members to nominate the eligible node with the highest probability $\mathbb{P}$ given in Eq. 4.9 to be the cluster head for the coming rounds.

4.5.3 Performance Evaluation Results

This section introduces a Self-Organised Clustering (SOC) technique with multiple mobile sink nodes for M2M sensor networks. A detailed comparison between the proposed scalable energy-efficient M2M routing protocol and four existing cluster based routing protocols is given with respect to network lifetime, cluster head numbers and aggregated packets at the sink nodes. The four cluster based protocols are: Low-Energy Adaptive Clustering Hierarchy (LEACH), Stable Election Protocol (SEP), Threshold sensitive Energy Efficient Sensor Network (TEEN), and Distributed Energy Efficient Clustering (DEEC). The proposed energy-efficient routing protocol and the four existing protocols were initiated using the same network and node parameters listed in Table 4.1, with the simulation scenarios being carried out using MATLAB.

Table 4.1: Parameters initialisation for simulation scenario

Parameters	Value
Sensor network dimensions	200×200 m^2
Number of nodes	100
Deployment type	Random
Initial energy of sensor nodes	0.5 J
Packet size	127 bytes
Transceiver idle state energy consumption	50 nJ/bit
Data aggregation energy consumption	5 nJ/bit
Amplifier energy	100 pJ/bit
Simulation rounds	7000

The simulation scenarios involve using the first order radio model given in Section 4.2. Both homogeneous and heterogeneous networks are used to evaluate the energy efficiency of the SOC routing protocol as well as its scalability to adopt the increment of sensor nodes in predetermined sensing field. LEACH is a homogeneous network routing protocol that is modified to support two and three levels of network heterogeneity. SEP and TEEN are used for two level network heterogeneity comparisons, and finally, DEEC is used for three levels of network heterogeneity comparisons.

The predetermined sensing field is given in Figure 4.4 in which the sensor nodes are randomly deployed to monitor certain phenomena. These nodes have fixed transmission range (30 m), with 127 bytes as a Maximum Transmission Unit (MTU), according to the IEEE 802.15.4 standard.

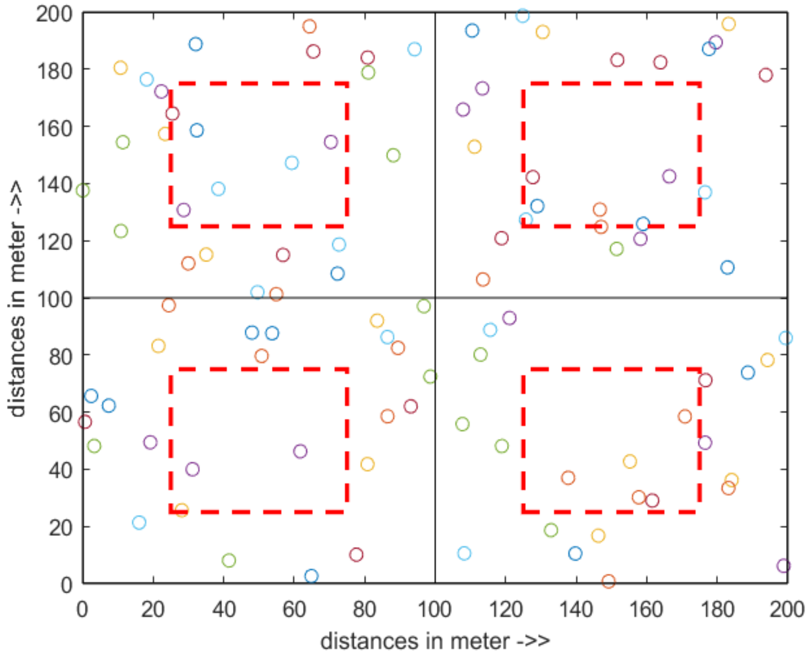
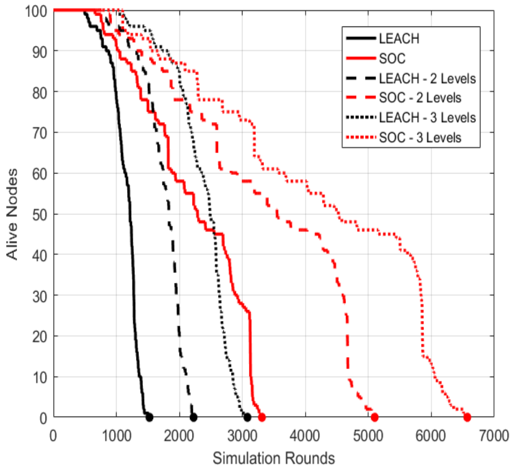
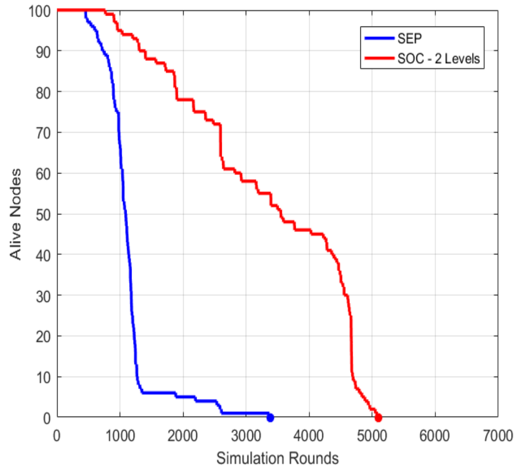


Figure 4.4: The predetermined M2M sensor field

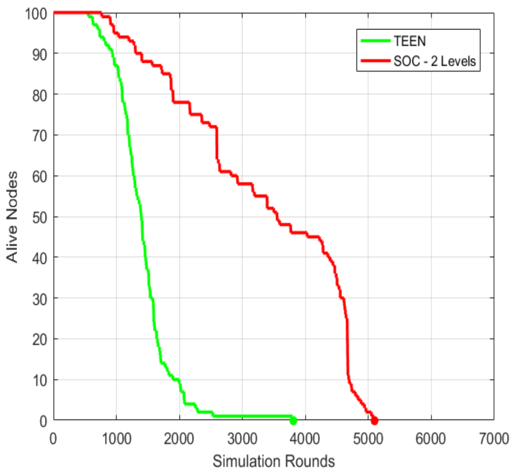
The number of alive nodes versus the simulation rounds is illustrated in Figure 4.5. In general, the proposed energy-efficient routing protocol with its self-organised clustering technique succeeds in prolonging the network lifetime by conserving the sensor node energy for both homogeneous and heterogeneous sensor networks. The energy efficient routing algorithm takes into account the advantages of four mobile sink nodes travelling in the sensing field with predefined path and constant speed to reduce multiple hop data transmission to a single hop. Fair distribution of energy depletion increases the network lifetime and makes most of the sensing field still reachable, because the dead nodes are distributed across the whole field, rather than being in a certain region. Due to the random node deployment in M2M sensing field, different schemes of initial energy will appear. The initial energy (E_0) is set to be 0.5 J, n=0.4, n_0=0.2, a=1, and b=2. Accordingly, the total initial energy of the sensing field is 50 J, 70 J, and 74 J for the homogeneous, two-level and three-level heterogeneous M2M sensor networks, respectively. Each single node in the proposed energy-efficient M2M protocol has more residual energy than the nodes of the LEACH, SEP, TEEN, and DEEC protocols during the whole simulation rounds due to the fairness in energy consumption.



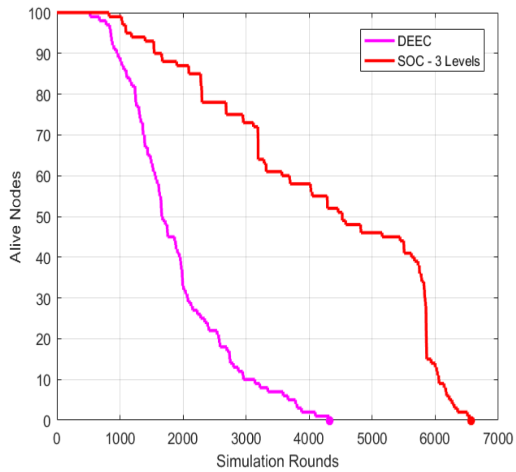
(a) LEACH



(b) SEP



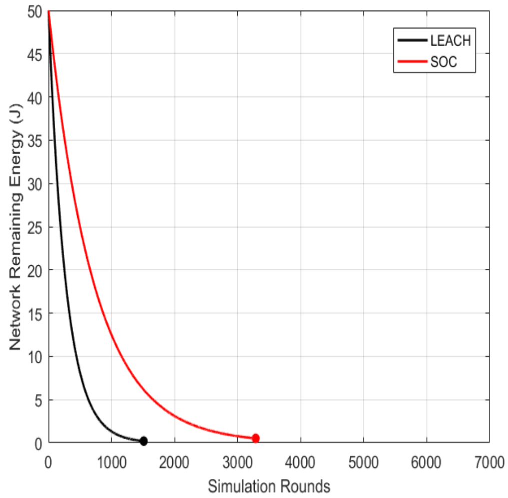
(c) TEEN



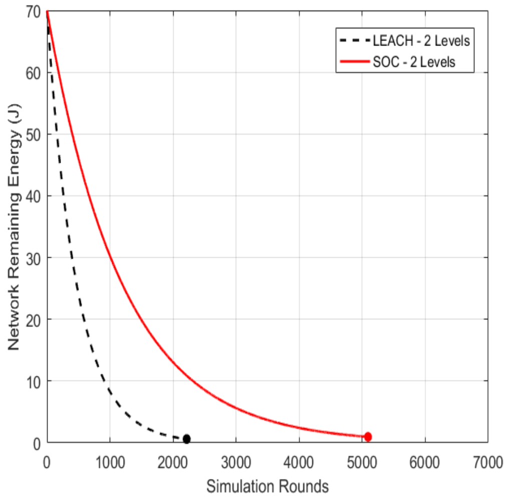
(d) DEEC

Figure 4.5: Network lifetime as a function of the simulation rounds

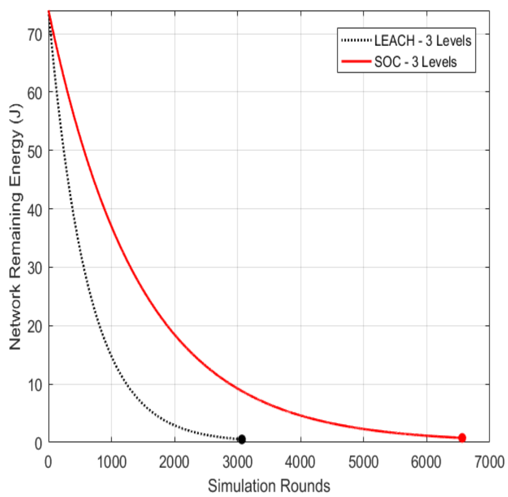
M2M sensor nodes are powered by the energy accumulated in their batteries. Accordingly, the consumed energy by the M2M sensor node has a great impact on the network lifetime and the overall sensing coverage. That is, when a sensor node depletes all its residual energy, it dies and may form a sensing coverage hole. Figure 4.6 shows the network remaining energy as a function of simulation rounds for LEACH, SEP, TEEN, DEEC, and the proposed SOC protocol. It is clear that energy depletion is much quicker with LEACH, SEP, TEEN, and DEEC than with the proposed energy-efficient approach. As shown in the abovementioned figure, the proposed energy-efficient routing protocol does not have a sharp slope compared to the other protocols, because of its self-organised clustering technique and the deployment of mobile sinks. The proposed SOC routing protocol has the same behaviour in both homogeneous and heterogeneous sensor networks due to heuristic clustering and fair load balancing.



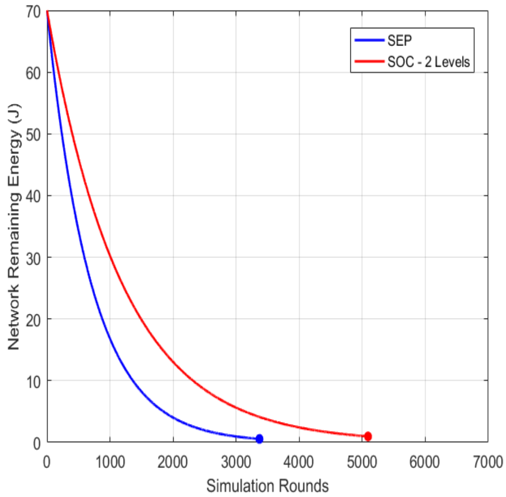
(a) Homogeneous LEACH



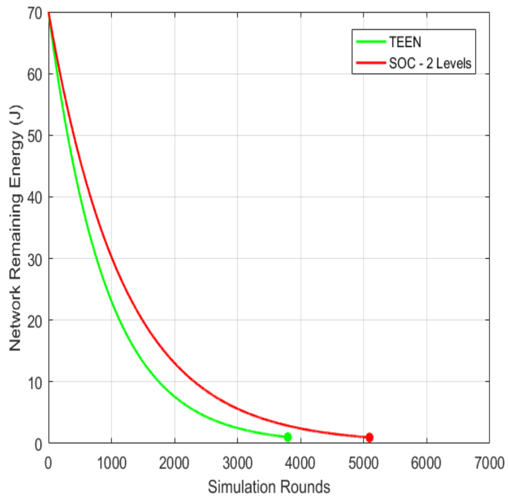
(b) 2-Levels LEACH



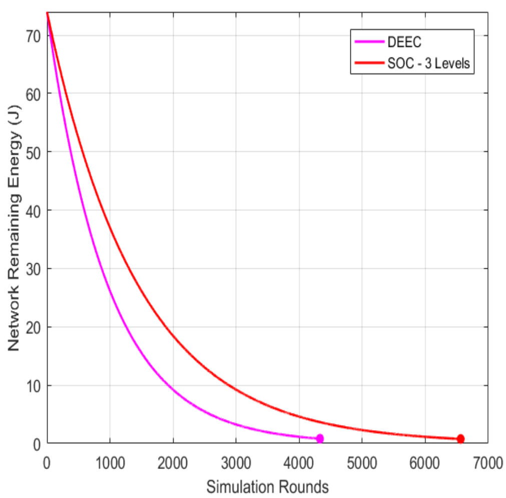
(c) 3-Levels LEACH



(d) SEP



(e) TEEN



(f) DEEC

Figure 4.6: Network remaining energy for LEACH, SEP, TEEN and DEEC protocols

Table 4.2 summarises the performance of the proposed energy-efficient routing protocol and the other existing protocols (LEACH, SEP, TEEN, and DEEC) in terms of different network lifetime definitions and cluster head number fluctuation.

Table 4.2: Routing protocols performance summary

Routing Protocol	Stable Period Rounds	Network Lifetime Rounds	Cluster Head Fluctuation
LEACH	1 – 498	1 – 1511	1 – 20
LEACH - 2 Levels	1 – 763	1 – 2216	1 – 22
LEACH - 3 Levels	1 – 1033	1 – 3068	1 – 19
SEP	1 – 455	1 – 3370	1 – 21
TEEN	1 – 556	1 – 3796	1 – 42
DEEC	1 – 526	1 – 4330	1 – 35
SOC	1 – 661	1 – 3287	Null
SOC - 2 Levels	1 – 768	1 – 5092	Null
SOC - 3 Levels	1 – 1100	1 – 6561	Null

Clearly, The SOC-based routing protocol has a longer stable period and network lifetime than LEACH, SEP, TEEN and DEEC. The prolonging of network lifetime is an important metric, because it increases network reliability, which means that the SOC-based routing protocol is an energy-aware adaptive clustering protocol. To summarise, the proposed energy-efficient cluster-based routing protocol prolongs the network lifetime by the percentages given in Table 4.3 when compared to the abovementioned cluster-based routing protocols.

Table 4.3: Network lifetime enhancement percentage

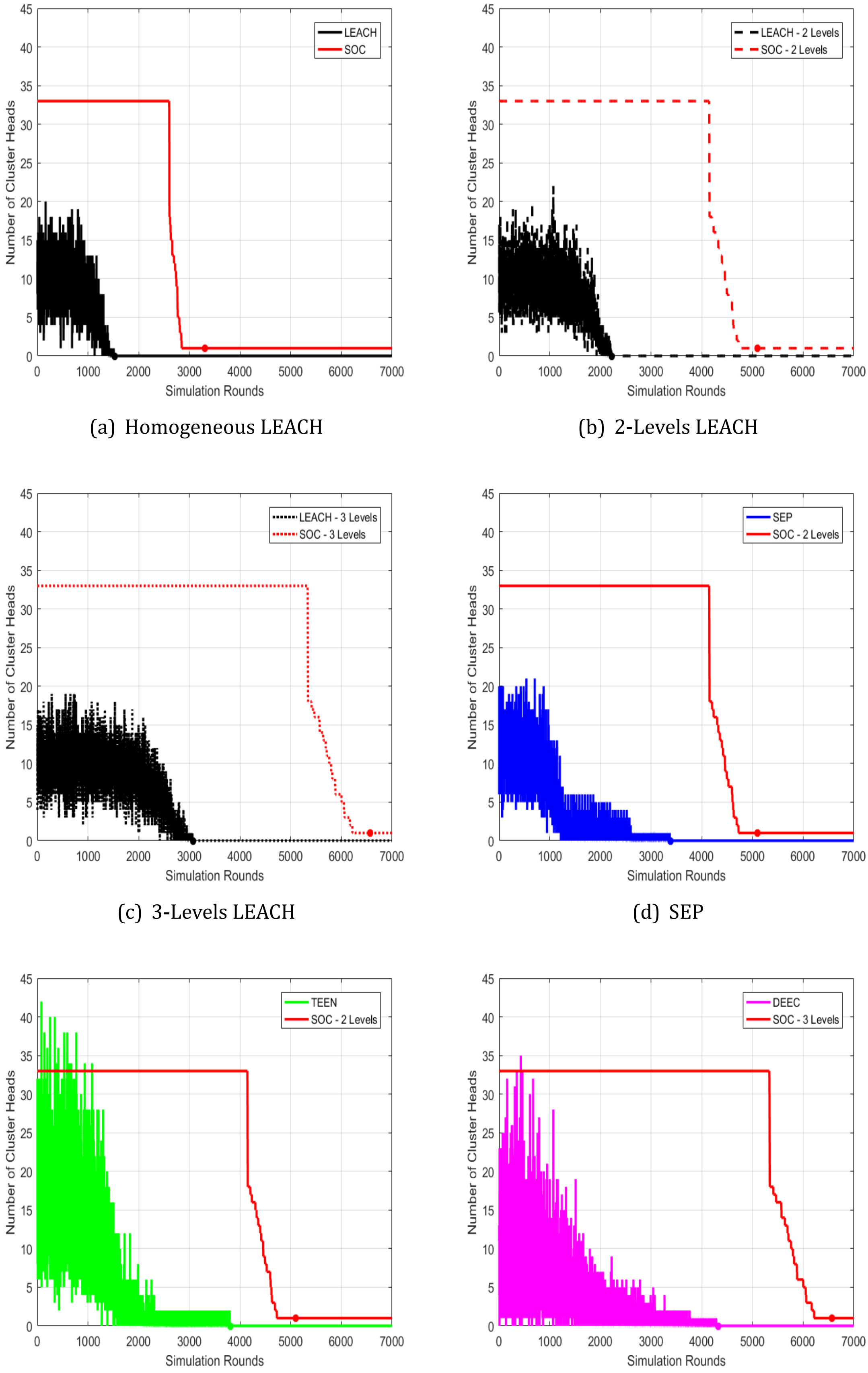
Protocol	LEACH	LEACH 2 - Levels	LEACH 3 - Levels	SEP	TEEN	DEEC
SOC	74%	–	–	–	–	–
SOC - 2 Levels	–	78%	–	40%	30%	–
SOC - 3 Levels	–	–	73%	–	–	41%

The hierarchical topology can provide satisfactory scalability for M2M sensor networks in which the whole network is divided into distributed groups. Each group has a group leader called a cluster head that performs data aggregation tasks to save sensor node energies and to prolong the network lifetime. The cluster head numbers versus the simulation rounds is shown in Figure 4.7 for both homogeneous and heterogeneous sensor networks. In comparison with the other cluster-based routing protocols, SOC has stable cluster head numbers across all the simulation rounds. This stability in cluster heads number occurs because cluster head selection and rotation are controlled by the probability formula given in Eq. 4.5 and Eq. 4.9 for homogeneous and heterogeneous sensor networks, respectively, that prevents the fluctuation in cluster head numbers, which the other routing protocols cannot deliver. The cluster head rotation does not happen for each simulation round, as in LEACH, through weighted probability, as in SEP, based on the sensing value threshold as in TEEN or needing knowledge of the whole network's energy, as in DEEC, but rather, when certain criteria are met, as explained previously. Accordingly, the SOC technique reduces the energy consumption and the number of transmitted control packets for both the selection and rotation, which prolongs the network lifetime. In addition, the SOC technique considers the issue of unequal sized clusters existing during the whole network lifetime and performs fair re-clustering to give every node in the network a chance to become a cluster head when it meets the selection condition. In the heterogeneous network scenario, the advanced and super nodes have a higher probability to become cluster heads due to the amount of residual energy that they have to form the top layer of the hierarchy scheme.

The total aggregated packets at the four mobile sinks versus the simulation rounds is depicted in Figure 4.8. The fairness in energy depletion across the sensing field ensures that the SOC-based routing protocol delivers many more packets to the sinks than the other routing protocols. It also consumes less energy, which helps to extend the network lifespan and increase the packet delivery to the sinks. The total aggregated packets at the sinks indicates that the energy holes problem is eliminated by deploying four mobile sinks with predefined paths and constant speed. In sum, the proposed energy-efficient cluster-based routing protocol enhances the total of aggregated packets at the sinks by the percentages given in Table 4.4.

Table 4.4: Aggregated packets enhancement percentage

Protocol	LEACH	LEACH 2 - Levels	LEACH 3 - Levels	SEP	TEEN	DEEC
SOC	167%	–	–	–	–	–
SOC - 2 Levels	–	165%	–	76%	40%	–
SOC - 3 Levels	–	–	162%	–	–	38%



(a) Homogeneous LEACH

(b) 2-Levels LEACH

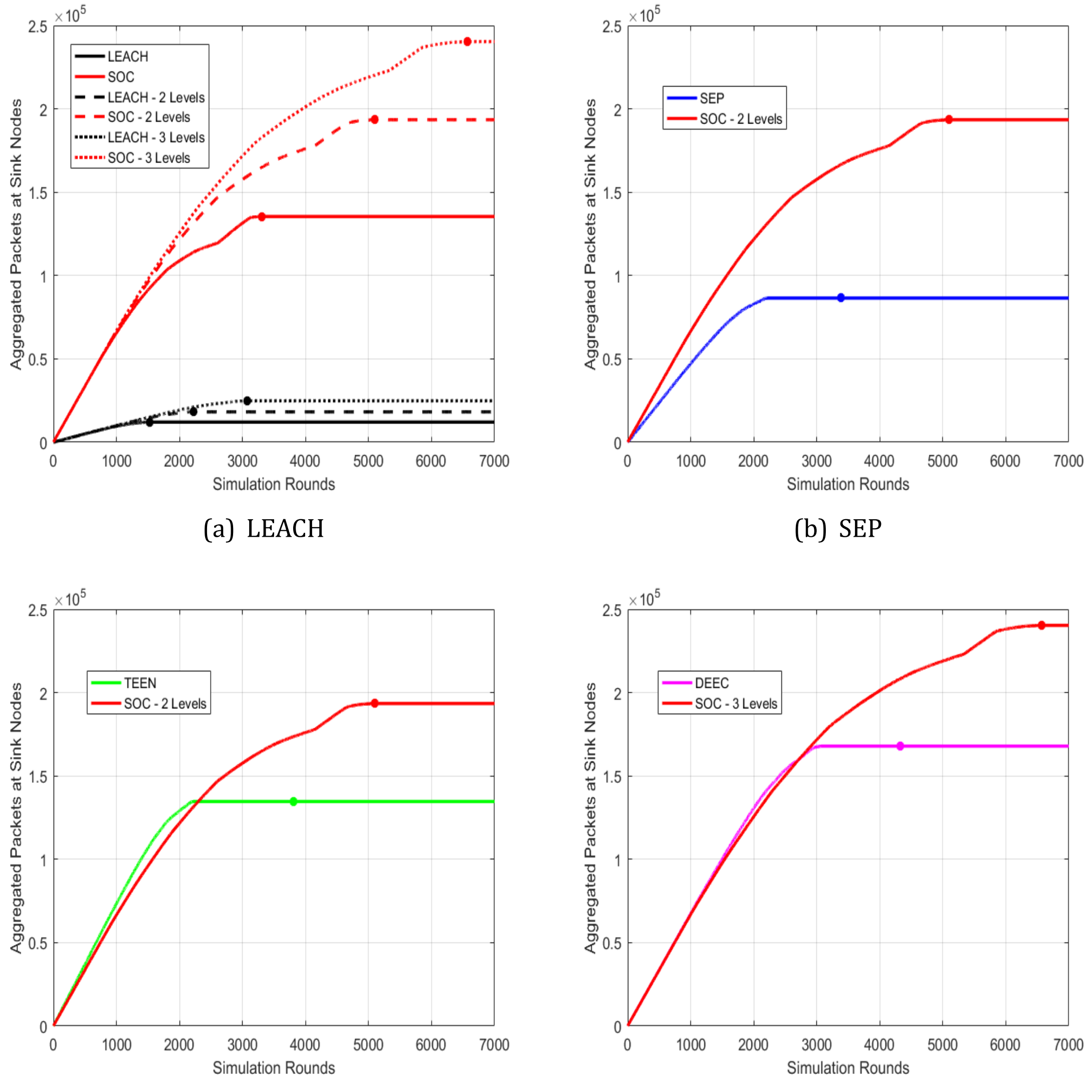
(c) 3-Levels LEACH

(d) SEP

(e) TEEN

(f) DEEC

Figure 4.7: Cluster head numbers for LEACH, SEP, TEEN and DEEC protocols



(a) LEACH (b) SEP

(c) TEEN (d) DEEC

Figure 4.8: The total aggregated packets at the mobile sink nodes

The scalability is a very significant and critical issue in the design of routing protocols for M2M sensor networks, which can be defined as the ability of extending the network without performance degradation. That of the proposed energy-efficient routing protocol are studied under different network sizes for both homogeneous and heterogeneous sensor networks. The First Dead Node Index (FDNI), 50% Dead Nodes Index (50DNI), and 100% Dead Nodes Index (100DNI) are taken into account as comparing metrics to investigate the scalability of the energy-efficient routing protocol. The obtained results from the conducted simulation scenarios are shown in Figure 4.9 in which the SOC technique has fixed behaviour as the number of sensor nodes increased. For a fixed dimensions sensing field, as the number of sensor nodes is increased, the traffic across the network will increase as well. Accordingly, the cluster heads will die more quickly due to receiving huge numbers of packets from cluster members and transmitting these towards the mobile sinks.

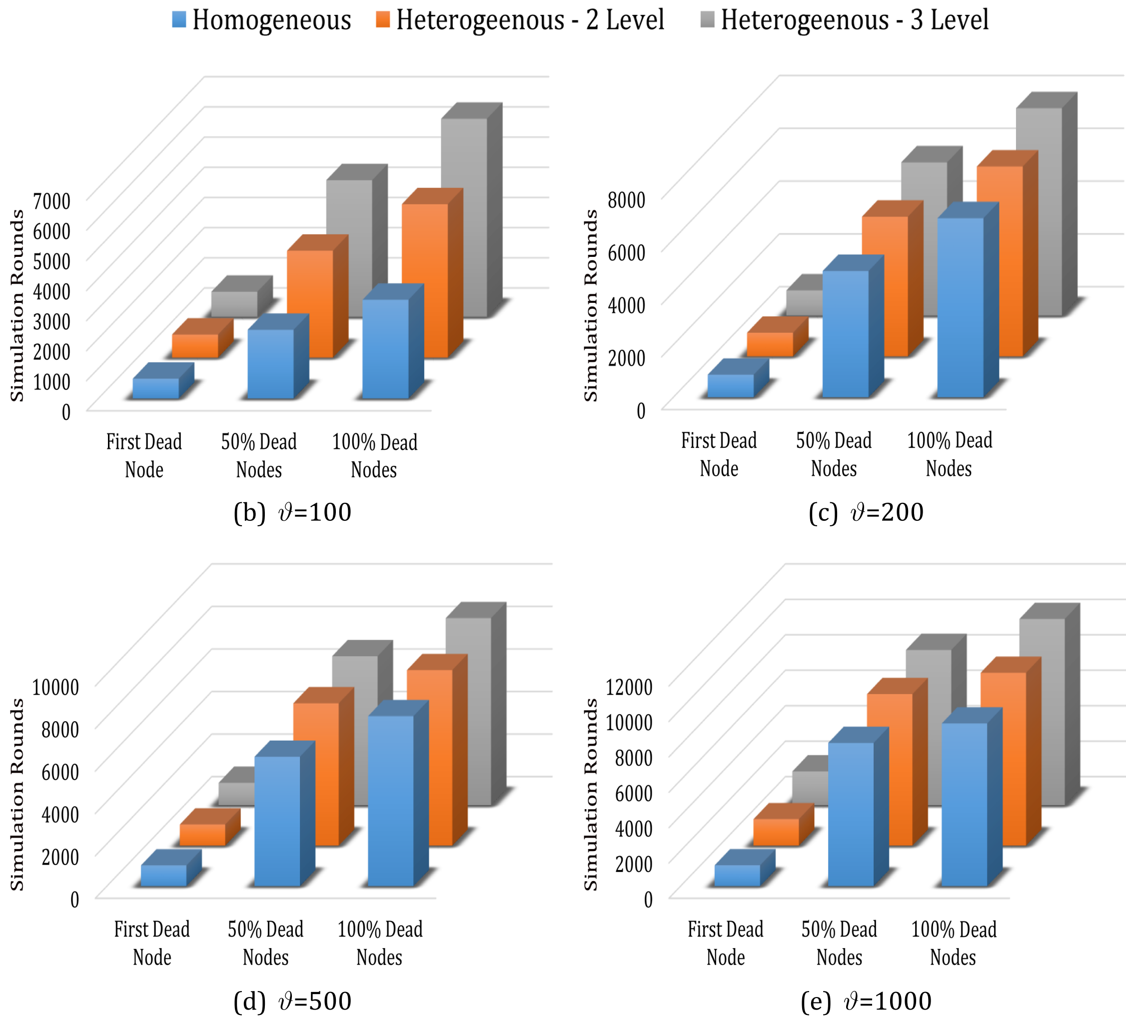


Figure 4.9: The scalability of the proposed energy-efficient SOC protocol

In sum, both homogeneous and heterogeneous sensor networks have been investigated in terms of energy hole problems and scalability. The energy hole problems have been solved by deploying four mobile sinks in the sensory field with predefined paths and constant speed. The outcomes of this approach pertain to balancing the traffic load among all the sensor nodes, rather than being concentrated on the nodes near to sinks. Moreover, the SOC technique has been used to achieve two main goals: (i) providing local fair energy consumption among all the nodes within the clusters, and (ii) enhancing network scalability through a hierarchical structure. In addition, the obtained results have shown that the scalability of the proposed energy-efficient routing protocol is not linear. Finally, the proposed SOC protocol is able to handle and perform well with topology changes that may occur from time to time during network lifetime.

4.6 Smart Sleep Scheduling Technique

In this section, an energy-efficient MAC protocol with smart sleep scheduling for cluster-based M2M networks is proposed. It extends network lifetime by using a new clustering technique with a sleep mode in M2M nodes. The developed MAC protocol is based on the IEEE 802.15.4 standard specification in terms of the energy model, coverage distance, packet size and radio frequency. According to the M2M network type and application, the M2M nodes are assumed to be randomly deployed in the monitoring area. Based on their position and their transmission range, the nodes are grouped together to form clusters. Figure 4.10 shows the proposed Smart Sleep – M2M (SS-M2M) protocol states from node deployment until the protocol objective is achieved. This objective refers to maximising network lifetime using the SOC technique and sleep scheduling mechanisms.

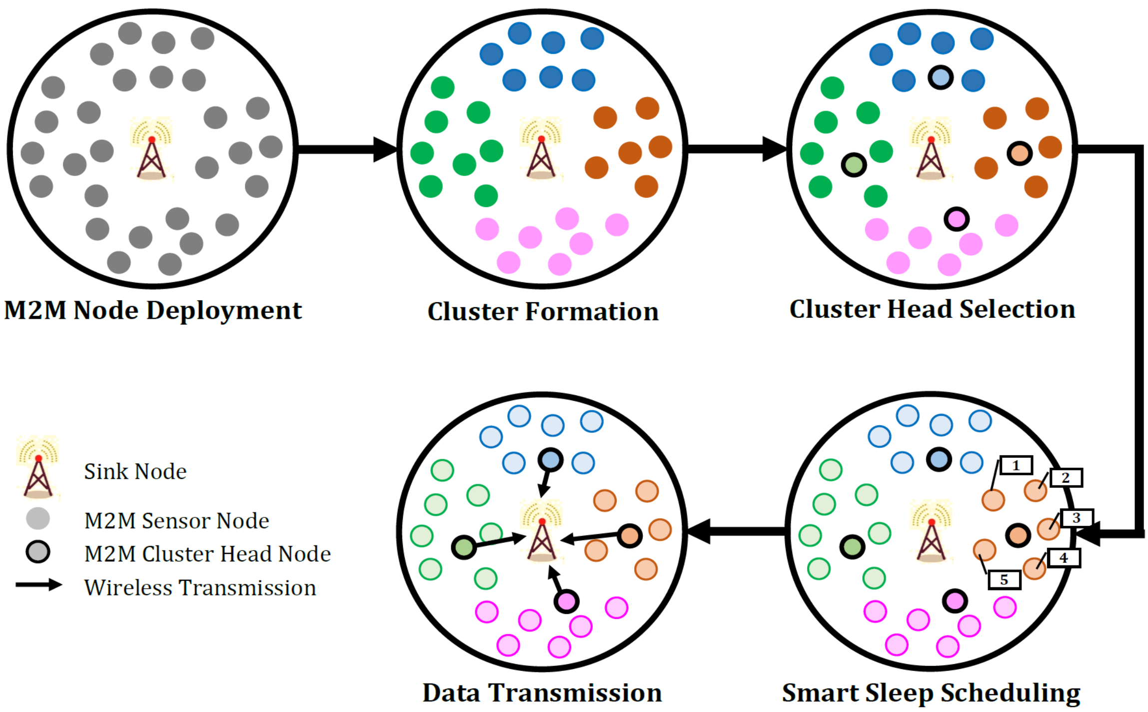


Figure 4.10: Proposed SS-M2M protocol states

After M2M sensor nodes have been deployed in the monitoring area, all have to broadcast the "Hello" message in order to join the network and those in their neighbourhood send an acknowledgement. From this message exchange, every M2M node builds a table containing the neighbouring nodes' identity, then the initial cluster structure is built. The sink node represents the Personal Area Network (PAN) coordinator of the network. The clustering algorithm can be explained as follows:

i. The M2M nodes are deployed randomly in a predefined field with fixed dimensions, all being battery powered with the same initial energy;
ii. Each node builds a neighbour table for all the nodes within its transmission range depending on the transmitted "Hello" messages and the corresponding received acknowledgement;
iii. After building the neighbour table, the initial CH formation, CH selection, and CH rotation are executed following the steps of the self-organised clustering technique presented in Section 4.5;
iv. Each M2M node continues to monitor its residual energy and updates the value of the rotation index. When the node residual energy drops below a certain threshold value, the multiplication index M_{index} in Eq. 4.5 (Eq. 4.9 for heterogeneous sensor networks) changes to 0, thus preventing it from being a CH in future rounds, but allowing it to contribute to the communication until its residual energy is completely depleted;
v. When the index value of the current CH becomes less than the σ value, the CH broadcasts a rotation message to inform the cluster members to select a new CH for the next rounds;
vi. After the removal of old the CHs or dead nodes, the cluster construction process takes place using the selection and rotation procedures given in the SOC technique and it continues until all nodes become dead.

A sink node represents an important component an M2M sensor network as it acts as border gateway between the sensor nodes and the Internet. As previously explained, it can be static or mobile [170]. However, a mobile sink improves network lifetime and packets drop rate in comparison to a static one, especially when the latter is far away from the sensor field or the sensor field is too large with a massive number of M2M sensor nodes, which would mean most of the nodes would need multiple hops to reach the sink. On the other hand, the mobile sink minimises the number of hops or even delivers a single hop. As a result, the number of dropped packets is reduced, network connectivity is improved and the energy holes problem is eliminated due to balancing the load of data routing among the M2M nodes [171].

The proposed MAC protocol was motivated by recognition of the advantages of mobile sinks and the sleep scheduling mechanism for prolonging network lifetime. After selecting the cluster heads and the sojourn locations τ_s of the mobile sink in a region τ, then the M2M nodes wake-up prior to the movement of the sink to its sojourn location, while the rest of the nodes in the other $(\tau - \tau_s)$ regions are in sleep mode. The

CH is responsible for executing the smart sleep scheduling mechanism. The proposed approach was aimed at conserving M2M node energy by leaving only a necessary set of nodes active and putting the rest into sleep mode. Under this arrangement, only the active nodes will deplete their energy, while the remaining nodes will sleep and preserve their energies for future use. The proposed smart sleep scheduling algorithm can be explained as follows:

1. The CH builds up the cluster member table $CH_{member}(j)$ of a node *j* as follows:

$$CH_{member}(j) = \{k \in N_C | \delta(j,k) \leqslant \upsilon, j \neq k\} \tag{4.12}$$

   where, N_C is the node set deployed in the region, $\delta(j,k)$ denotes the distance between node *j* and node *k*, and υ is the sensing range of a node. This sensing range is assumed to be a circle with radius υ centred at the node, as illustrated in Figure 4.11;

2. The CH builds up the nearby table $CH_{near}(j)$ for each cluster member *j* as follows:

$$CH_{near}(j) = \{k \in N_C | \delta(j,k) \leqslant 2\upsilon, j \neq k\} \tag{4.13}$$

   where the nearby nodes of a node *j* are the set that may not in its range *j* but have some common sensing areas with it;

3. The CH checks whether the sensing area of cluster members can be covered by neighbours and nearby nodes. In addition, the removal of the node does not cause network dis-connectivity;

4. If the condition in step (3) is true, then the node is considered to be eligible for going into sleep mode, and the cluster head adds it to the redundant table $CH_{red}(j)$. The cluster head uses the TDMA mechanism for the rest nodes that are not it the $CH_{red}(j)$ table;

5. The CH also goes into sleep mode after collecting the data from the cluster member and wakes up in the next period or when the sink node arrives at the sojourn location in the sensing field;

6. This mechanism is repeated for the remaining cluster member nodes until all have been checked as to whether they can be put into sleep mode or not.

It is clear from Figure 4.11 that the sensing area of node C could be covered by the neighbour (node A) and nearby (nodes B and D). Hence, node C goes into sleep mode, with the area sensed by node C being still so by some other nodes and this segment will not disconnect. The sleep mode of the other nodes (B, D and E) is scheduled using TDMA scheduling. After the data collection phase, the cluster head puts itself into sleep mode and back into active mode either in the next period or on the arrival of the sink.

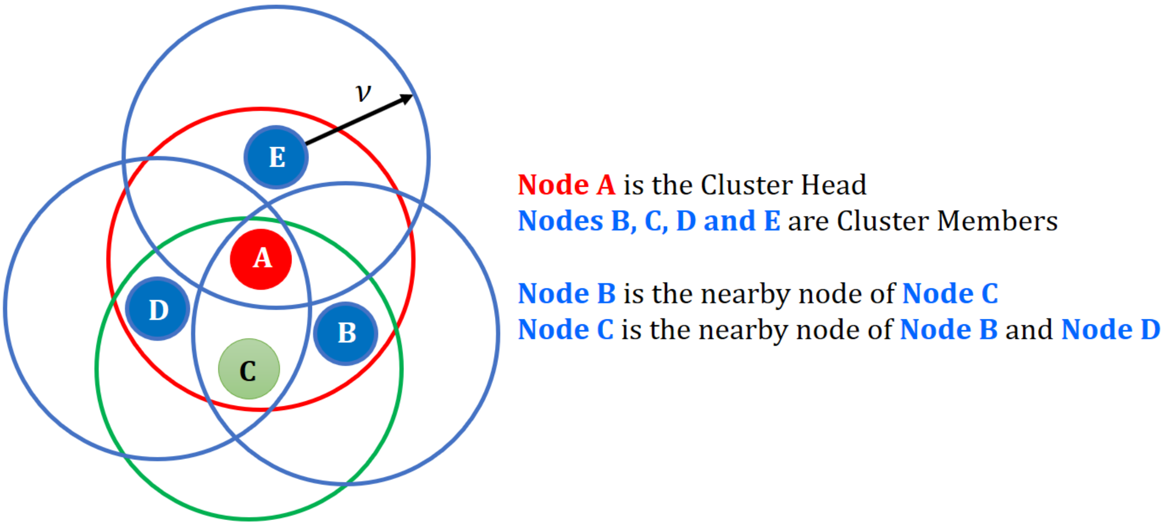


Figure 4.11: Neighbour and nearby nodes in the M2M sensing field

After validating the proposed SOC techniques in the previous section, it was shown how it outperforms the other cluster-based routing protocols. This section will conduct a comparison between the SOC technique with and without the smart sleep scheduling technique for both homogeneous and heterogeneous sensor networks. MATLAB has been used to evaluate the performance of the proposed smart sleep scheduling, with the relevant simulation parameters being listed in Table 4.1 and the same radio model given in Section 4.2.

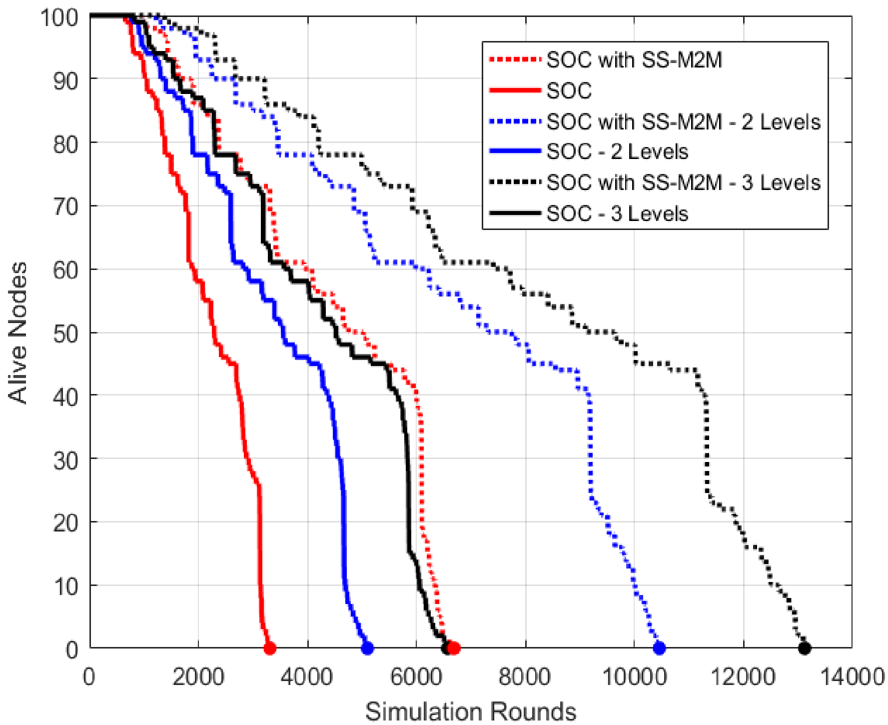


Figure 4.12: M2M sensor network lifetime with smart sleep scheduling

Figure 4.12 shows the network lifetime of the proposed smart sleep scheduling algorithm versus the simulation rounds. By using the developed SOC techniques with the smart sleep scheduler, the network lifetime is extended by 67.8% in homogeneous sensor networks, while it is prolonged by 68.9% and 66.5% in 2-level and 3-level sensor network heterogeneity, respectively. The smart sleep mechanism enhances network lifetime through a dynamic duty-cycle in M2M sensor nodes.

4.7 Summary

This chapter presented a scalable energy-efficient routing protocol for M2M sensor networks. The chapter started with an overview of the key requirements, constraints and assumptions that related to the protocol design challenges in order to propose a self-organised routing algorithm. In general, energy efficiency achieved in two ways: hierarchical structure and sleep scheduling. All the sensor nodes are stationary, then the entire network topology might alter much slower. Accordingly, more energy can be conserved. The M2M sensor network was organised in hierarchical topology that guaranteed to eliminate energy holes and network convergence problems. Finally, the proposed self-organised clustering technique alongside with the smart sleep scheduling succeeded in prolonging the network lifetime and enhanced its convergence through multiple mobile sinks.

Chapter 5

SD-NFV for Cloud-Based IoT Networks

5.1 Introduction to Programmable Networks

The data communication in traditional sensor networks occurs between one end of the system to the other. The network scalability and availability are limited to the routing algorithm, congestion and moderate Quality-of-Service (QoS). In recent years, increasing interest has been raised in deploying more functions inside current network elements in order to achieve better performance in terms of network services and cost. As a result, programmable networks have been developed to cope the limitations present in traditional networks. These are able to build adaptive networks that have the ability to be reprogrammed even after deployment. The major difference between the traditional and programmable networks is that for the latter network elements are directly programmed using a minimal set of Application Programming Interfaces (APIs) by the user. The programmable networks target reconfiguration, simplifying and accelerating network programmability in a secure and centralised manner. The most recent approaches to programmable networks are Software – Defined Networking (SDN) and Network Function Virtualisation (NFV). The SDN architecture provides centralised control and management by increasing the network programmability through a centralised software-based controller (the control plane), while the network devices become simple packet forwarding devices (the data plane). On the other hand, the NFV architecture provides a programmability feature by running different network applications simultaneously on a single physical network node. SDN and NFV with the cloud-based gateway enable the remote user to deploy an executable code dynamically to perform a new functionality at runtime [172] [173].

5.2 Enabling SDN, NFV and Cloud in 6LoWPAN Network

The Software Defined – Network Functioning Virtualisation (SD-NFV) approach has been proposed as an energy-efficient way of prolonging the 6LoWPAN network lifetime through the network programmability feature. The SD-NFV approach enables the 6LoWPAN node to be re-tasked after deployment. The developed 6LoWPAN testbed has been integrated with a cloud computing platform to provide global access to the M2M sensor network.

5.2.1 6LoWPAN Hardware Platform

The M2M sensor node consists of various sub-systems, such as sensing, computation, communication and power sub-systems. 6LoWPAN is a pioneer protocol aimed at providing small devices that have constrained processing and limited energy with the ability to have IPv6 global connectivity. Over the last few years, there have been several free and commercial solutions developed for 6LoWPAN. Most of the developed approaches were implemented based on an operating system, where the 6LoWPAN protocol stack was used along with the node's operating system. However, as the M2M sensor nodes are characterised by a small memory and moderate processing unit with limited energy source, it is not practical to include an operating system with dedicated software applications in those devices at the same time.

In order to develop the 6LoWPAN protocol stack with the low memory M2M sensor node and make it workable with existing IP networks, it is necessary to use the available open source resources to the maximum possible extent. Consequently, the open source hardware platform has been chosen and integrated to fit into the designed M2M sensor node scheme as well as the 6LoWPAN gateway.

One of the most important features of the M2M sensor node is the selection of the processing platform. The M2M nodes need to be cost-effective and energy-efficient to meet the IoT network promises. There are several types of nodes available in both commercial and open-source domains. The proposed approach is based on an open source hardware platform represented by the Arduino [174], which is a microcontroller board based on the ATmega328 chip, as a processing platform. The XBee module is deployed as a radio communication for MAC and PHY layers of the IEEE 802.15.4 standard, while a temperature and humidity sensor is used as a sensing unit for the M2M node. The Arduino board has been chosen due to its low energy consumption, small size, cost-effectiveness, and programmability feature. Choosing the Arduino board will open new horizons to increase network programmability and management through open source hardware platforms.

To realise the concept of the IoT paradigm, it is essential to make the things (connected objects) addressable, controllable, and accessible via the Internet. The proposed approach has been tested using a simple temperature and humidity sensing application in which the M2M sensor nodes transmit the sensed data to the M2M gateway. The Figs. 5.1 – 5.3 show a 6LoWPAN-based M2M nodes prototypes, which are classified into:

- *Simple node*: the simple node performs sensing and communication only, without any processing capabilities and it cannot be selected as a cluster head. The simple node is composed of a temperature sensor (TMP36) attached directly to the XBee module, and an LED used as an indicator for receiving the control signal from the cluster head in a hierarchical topology or from the sink node in a star topology. The simple node is battery powered (3.7 V/1000 mAh) and is shown in Figure 5.1;
- *Advanced node*: the advanced node performs sensing, communication, and processing of the sensed data. It can be selected as the cluster head among cluster members. It comprises a temperature and humidity sensor (DHT11), XBee module, and an LED. All these components are attached to the Arduino Uno board. The advanced node is battery powered (9 V/1600 mAh) and is shown in Figure 5.2;
- *Sink node*: the sink node is the final destination for all the data being sensed by the M2M sensor nodes. It can be static or mobile depending on the application. It is built-up of an Arduino Uno board equipped with two communication modules (XBee and ESP8266). The XBee module is used for the M2M sensor network communication, while the ESP8266 is used for Internet communication. The ESP8266 connects the 6LoWPAN network with the IP networks as well as the cloud platform where the data is being stored. The sink node is permanently powered with extra storage capability and is shown in Figure 5.3.

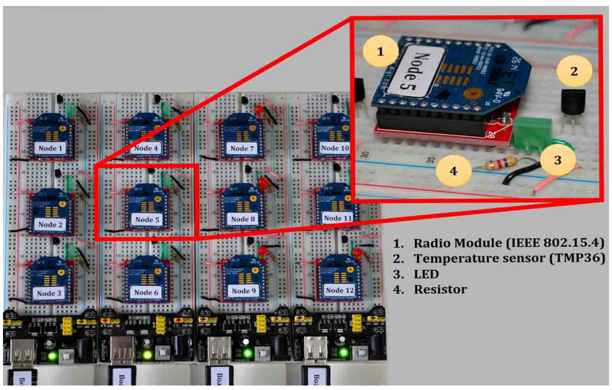


Figure 5.1: Simple node prototype

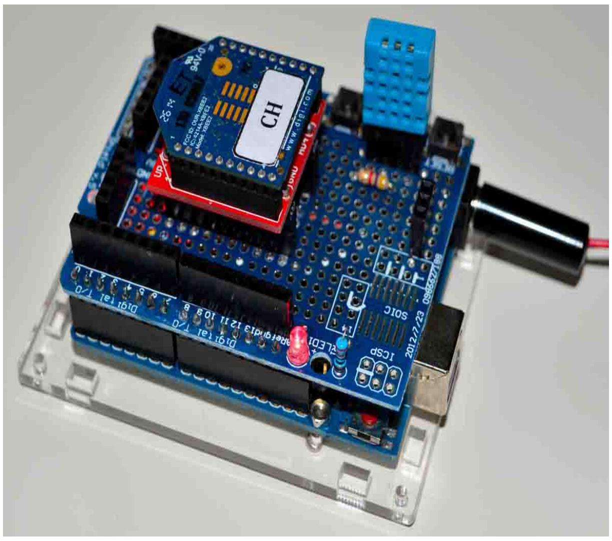


Figure 5.2: Advanced node prototype

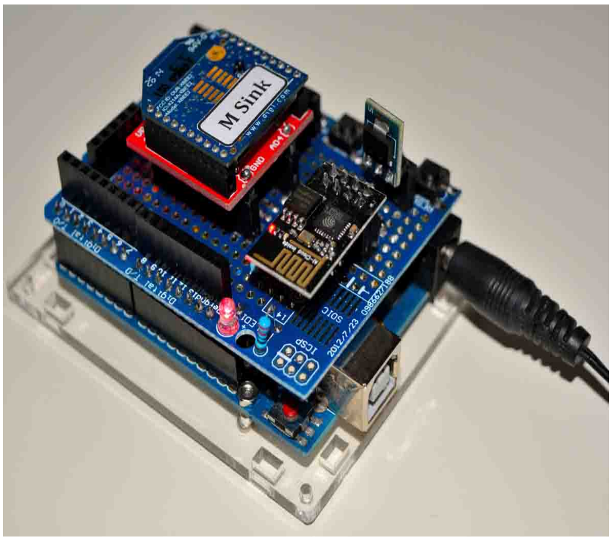


Figure 5.3: Sink node prototype

5.2.2 Customised SDN Controller Design Considerations

SDN and NFV are complementary technologies that have a lot in common, because they are both aimed at developing open software for standardised network hardware. NFV technology is geared towards creating on-demand programmable network functions and locating them at the most suitable location in the network infrastructure using adequate network resources [175]. SDN technology can decouple the control plane and data plane in order to increase network programmability and reconfiguration. Whilst SDN and NFV have value when implemented separately, combining them in one network will achieve greater value.

Current approaches to integrating M2M sensor nodes in the Internet have several drawbacks, and hence, alternative architectures need to be proposed and evaluated. The proposed integration of SDN and NFV is aimed at deploying different routing algorithms for a 6LoWPAN network by deploying the Virtual Network Function (VNF). VNF is deployed on top of the integrated cloud-based 6LoWPAN gateway for both homogeneous and heterogeneous M2M sensor networks. Figure 5.4 shows the typical architecture of the M2M sensor node with a 6LoWPAN protocol stack alongside the TCP/IP stack. The SDN controller is a software-based network entity that is used to manage and control network devices using programmable elements via different APIs. In order to adopt the SDN concept in the M2M sensor network, a novel customised SDN controller is proposed to bridge the research gap outlined earlier. The proposal regarding the customised SDN controller needs to take into account the limited memory and processing unit of the M2M sensor nodes to achieve a low software footprint.

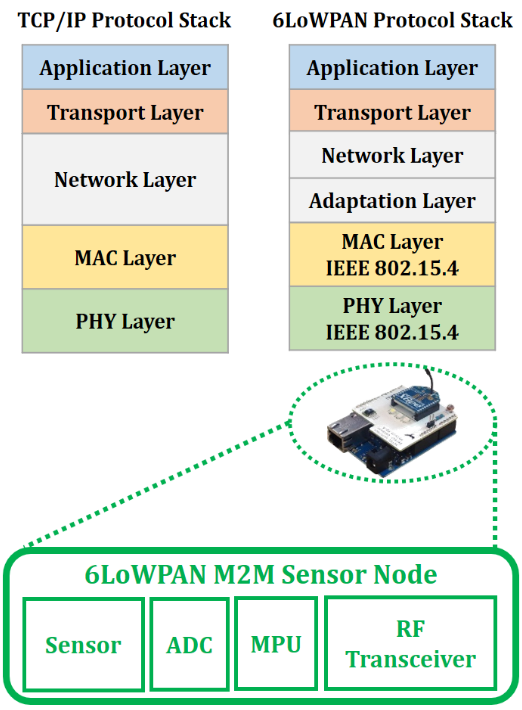


Figure 5.4: M2M sensor node architecture

The SDN controller is a software artifact being customised to fit a 6LoWPAN protocol stack, which enables end-to-end services on resource constrained devices. The controller is responsible for the following: (i) discovery of network topology; (ii) service management; (iii) virtualisation service; and (iv) data routing and load balancing. Additionally, the SD-NFV introduces a new flow table entry to cope with the high memory usage of the programmable interface.

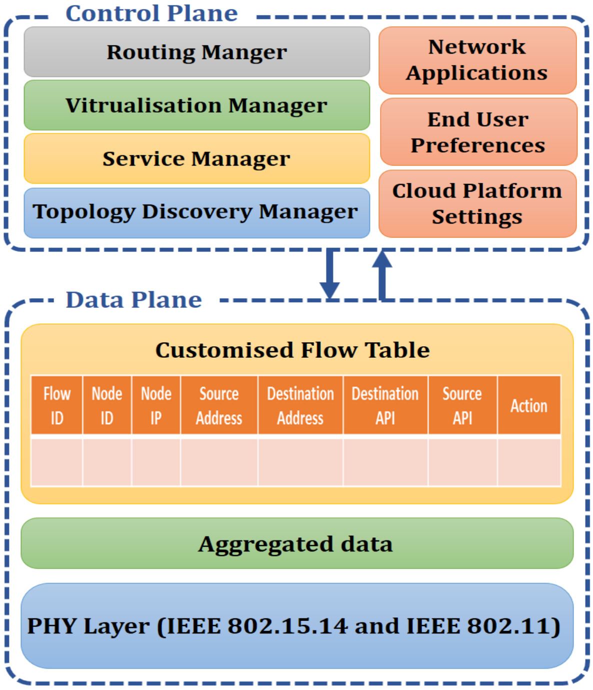


Figure 5.5: Customised SDN controller architecture

Figure 5.5 illustrates the architecture of the customised SDN controller used in the proposed SD-NFV approach. The controller is part of the 6LoWPAN coordinator, which is the 6LoWPAN gateway that starts up and initiates the network using unique PAN ID. The network discovery manager uses a discovery function to check the available alive nodes or newly joined nodes; this function is performed periodically to keep the global topology of the network up-to-date and to modify the alive node table entries. The service manager is important for allocating each node with a different level of services depending on the node's priority in the customised flow table entities. Also, the service manager is responsible for providing cloud service connectivity to the 6LoWPAN network. While the virtualisation manager allows different 6LoWPAN nodes to share the same network functions in the gateway as well as providing virtual individual connectivity between the 6LoWPAN nodes and cloud computing platform. Finally, the routing and load balancing manager is capable of executing different routing algorithms and performs load balancing optimisation techniques to achieve high throughput and to reduce the end-to-end delay in the sensor network based-on a 6LoWPAN protocol stack.

Figure 5.6 shows the detailed implementation of the flow table at the 6LoWPAN gateway and it is continued in Figure 5.7. The flow table contains the list of rules to perform certain actions depending on the ingress and egress flow entities. In the proposed SD-NFV testbed, the control traffic from the controller to the data plane (i.e., downstream and upstream traffics) contains forward, modify-state (configuration), and drop.

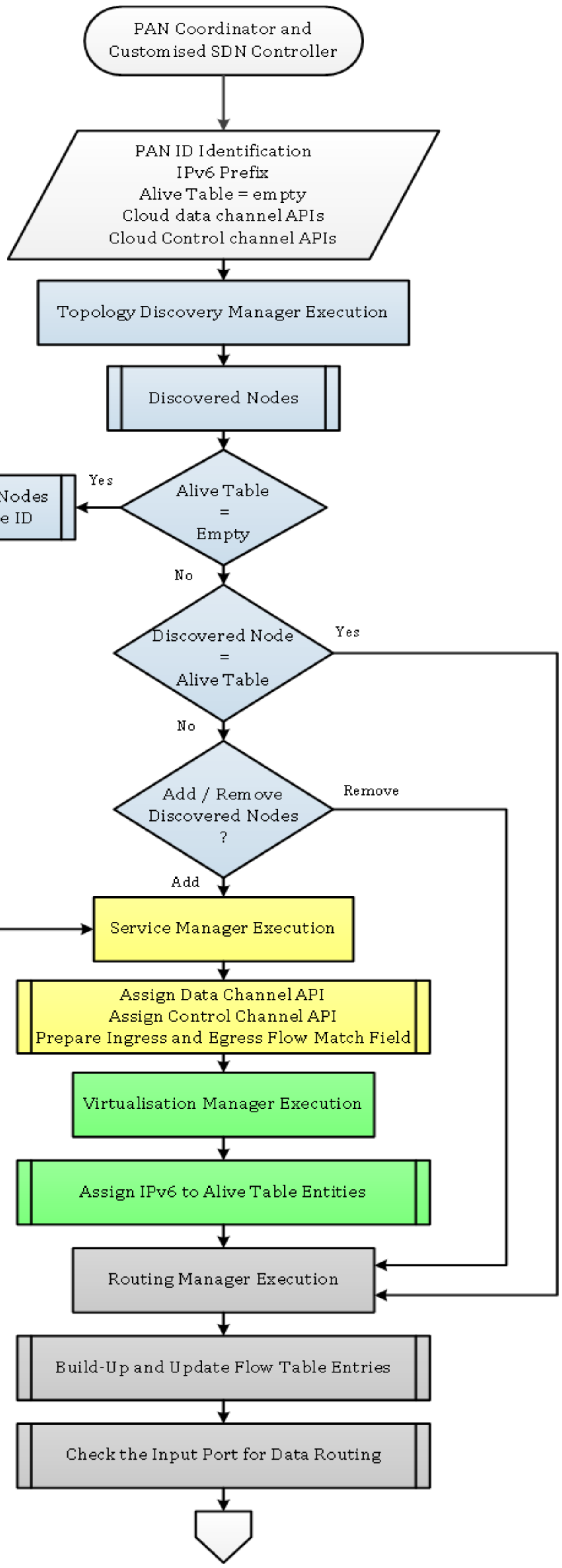


Figure 5.6: Implementation process of the customised SDN flow table

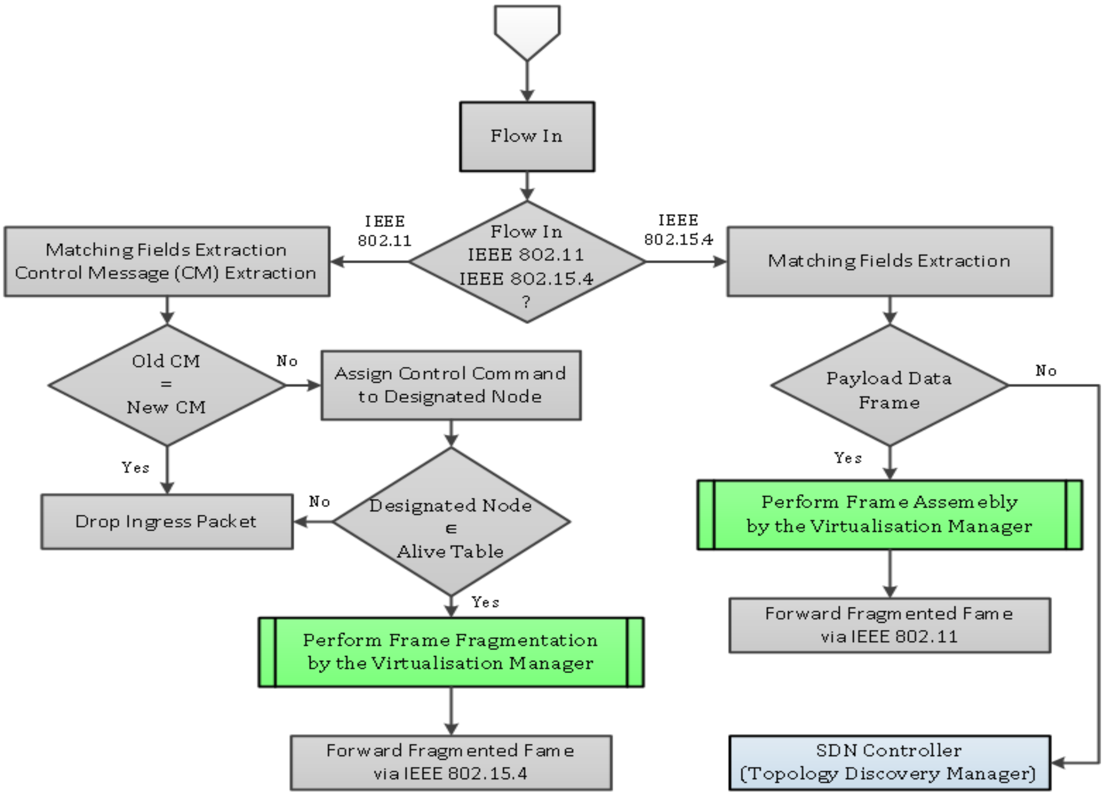


Figure 5.7: Implementation process of the customised SDN flow table (continued)

The OpenFlow protocol [109] has entirely been developed for wired networks and cannot be applied directly to wireless M2M sensor networks due to its complexity. The flow table entries are shown in Figure 5.8, which are customised from the conventional SDN concepts and the OpenFlow standard. Its entries take into account the constrained nature of the M2M sensor nodes and each table entry is divided in three entities. The matching fields contain the conditions a packet needs to comply in order to be processed, the action field specifies the executed action, and the statistics field is used for processed packet counting. The most common actions are to discard the packet, to forward it or to modify the flow table entries.

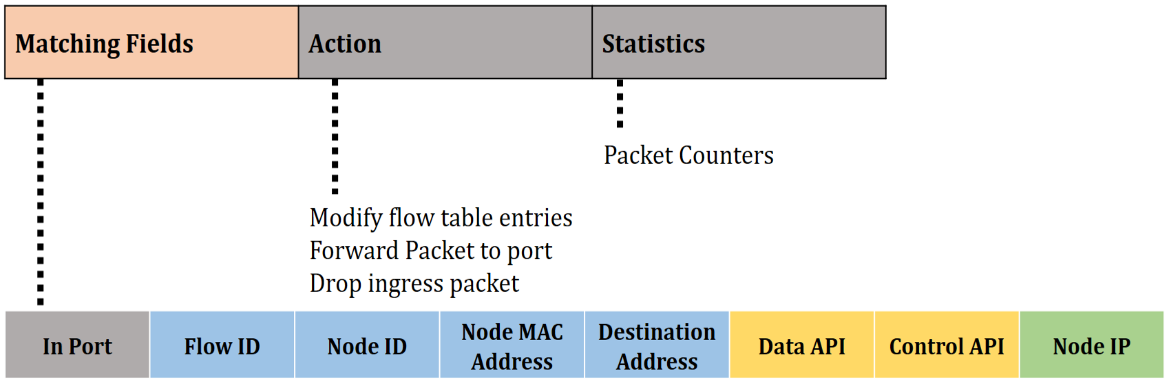


Figure 5.8: The proposed flow table entries of the customised SDN controller

To summarise the proposed approach, Figure 5.9 illustrates the relationship between SDN, NFV, and cloud computing with the 6LoWPAN M2M sensor network. It is clear that each technology abstracts certain functions from different network resources; the benefits obtained from each of them are similar in terms of traffic agility, cost-effectiveness, reduction in nodes' energy consumption, and dynamic network scalability. In addition, Figure 5.9 depicts the layer's abstraction and the customised SDN flow table entries.

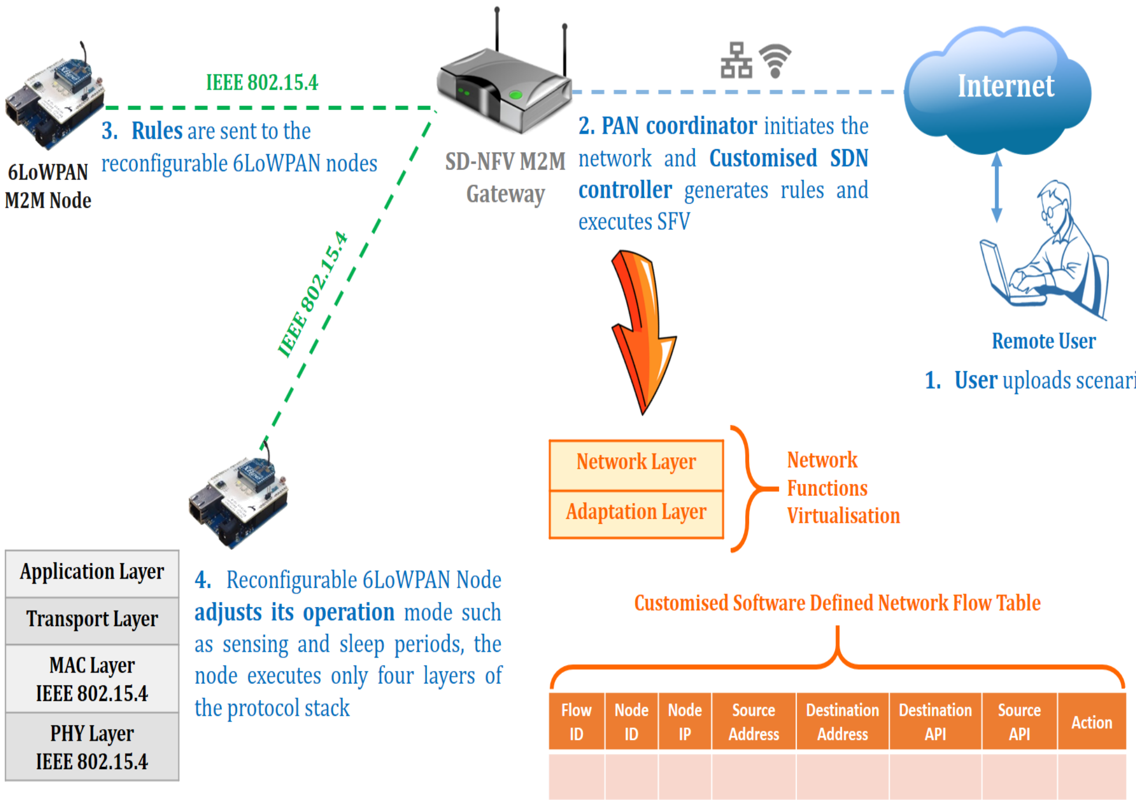


Figure 5.9: The proposed SD-NFV architecture based-on a 6LoWPAN testbed

The 6LoWPAN protocol stack has been implemented using the Arduino pico Internet Protocol version 6 (pIPv6) stack, the library being available at [176]. It was the first IP stack for embedded systems to implement in full the internet protocols using M2M constrained sensor nodes. The customised SDN controller was built using C++ language and deployed in the 6LoWPAN gateway, while the NFV is used to migrate the network layer and adaptation layer from a node's protocol stack to the gateway protocol stack and merge them with the SDN controller. This virtualisation function is called Sensor Function Virtualisation (SFV), which transforms multiple node tasks into software packages inside the 6LoWPAN gateway. This gateway now handles the 6LoWPAN coordinator for network initialisation, the customised SDN controller, and the two layers (network and adaptation layers) from the 6LoWPAN protocol stack.

The Requests for Comments (RFC) 4861 [177] was the first IPv6 node discovery specification, which was extended in 2012 by RFC 6775 [178] so as to be able to adopt 6LoWPAN node discovery requirements. The 6LoWPAN edge router or PAN coordinator is responsible for connecting the 6LoWPAN network to the external IP networks and to propagate the IPv6 prefixes among the 6LoWPAN nodes. In the traditional 6LoWPAN network, each node keeps checking its reachability to the edge router by performing a heavy control message exchange, such as Node Discovery (ND), Router Advertisement (RA), Neighbour Advertisement (NA), Neighbour Unreachability Detection (NUD), Duplicate Address Request (DAR) or Duplicate Address Confirmation (DAC). The 6LoWPAN node periodically sends NUDs until it receives a confirmation, even if it does not have data to send. The major issues in traditional node discovery are heavy packet transmission over the IEEE 802.15.4 medium, significant energy consumption to maintain network connectivity, and reduction in link reliability. The main challenge in 6LoWPAN node discovery is to develop a mechanism that provides less packet exchange for network connectivity with minimum discovery latency and power consumption to form the global network topology.

On the other hand, the customised SDN controller has a topology discovery manager that is responsible for maintaining network connectivity for efficient data routing. The proposed topology discovery mechanism takes advantage of the virtualised layers and the 6LoWPAN network does not need any IP connectivity at the node level. Hence, this will reduce the generated packets for node discovery and minimise the node's energy consumption by preventing the periodic NUDs and relevant message exchange. The proposed approach is based on the SDN flow table entries, whereby after the network initiation phase is completed, the 6LoWPAN coordinator reports to the SDN controller the address of the alive nodes. Subsequently, the customised SDN controller assigns each node an IP and saves this entry in the alive node table. As the SDN controller knows the global topology of the network, it can build-up the flow table to each node including its IP assignment, as shown in Figure 5.9. The topology discovery manager performs network discovery on a regular basis, but the table will only be updated when a node joins or leaves the 6LoWPAN network. The proposed SD-NFV approach reduces the network discovery latency as well as reducing energy consumption in the 6LoWPAN nodes during the network topology discovery phase.

The leveraging of cloud, SDN, and NFV technologies in 6LoWPAN node discovery and data routing have not been considered in current literature. The proposed SD-NFV approach compromises energy consumption with end-to-end delay to prolong network lifetime. Each technology abstracts certain functions to provide a wireless programmable network that supports heterogeneous M2M networks. The customised SDN controller with the NFV and cloud computing is aimed at providing multi-vendor compatibility for smooth protocol evaluation and implementation. In sum, the proposed approach will bridge the research gap with the SD-NFV approach that offers hardware-independent and on-demand function installation.

5.2.3 Integration of Cloud Computing Services

ThingSpeak [179] is the cloud computing platform used in the implementation of the SD-NFV approach based on a 6LoWPAN testbed, which provides free storage with a data visualisation feature. The cloud platform connects the 6LoWPAN network with the global Internet through the SD-NFV gateway. There are two types of channels, namely: data channels and control channels. The data channels are used for storing the sensed data, while the control channels are used for sending control commands to a specific node over the IP network. These ThingSpeak channels are chosen to demonstrate that the SD-NFV gateway provides bidirectional communication for the 6LoWPAN nodes. Cloud computing provides ubiquitous connectivity between the M2M sensor nodes and the external IP networks via the SD-NFV 6LoWPAN gateway. Figure 5.10 illustrates service chaining operations and network connectivity for the proposed SD-NFV approach.

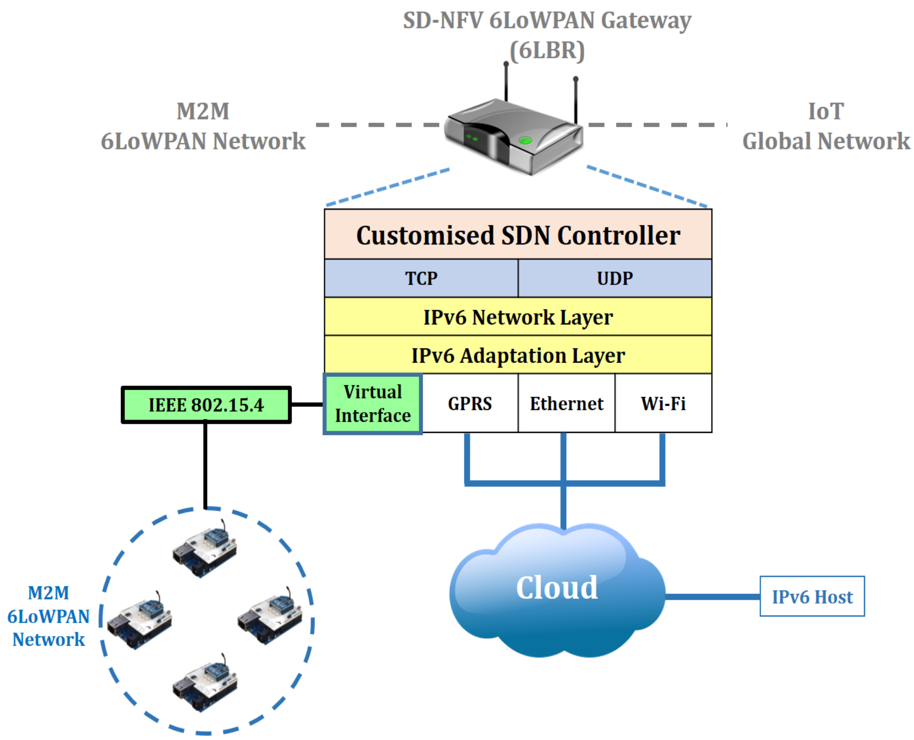


Figure 5.10: 6LoWPAN cloud connectivity through the SD-NFV gateway

The cloud front-end API is visible to the user in order to deploy different algorithms and change the network preferences, while the back-end of the network resides in the 6LoWPAN network to provide a global connection via the edge router (M2M gateway).

5.2.4 Remote End-User Application

A simple end-user application is built using MATLAB software to emulate external IP access to the 6LoWPAN network through the SD-NFV gateway. The application is used to retrieve the data from the cloud and to analyse it on a remote PC. In addition, the remote application is used to send control messages to the M2M sensor nodes by turning the attached LED on/off to verify IP connectivity and network heterogeneity. Figure 5.11 shows the Graphical User Interface (GUI) of the end-user or remote application. The remote application reads data from data channels in the cloud (ThingSpeak) and sends the network preferences to the 6LoWPAN gateway using the control channels of the cloud computing platform.

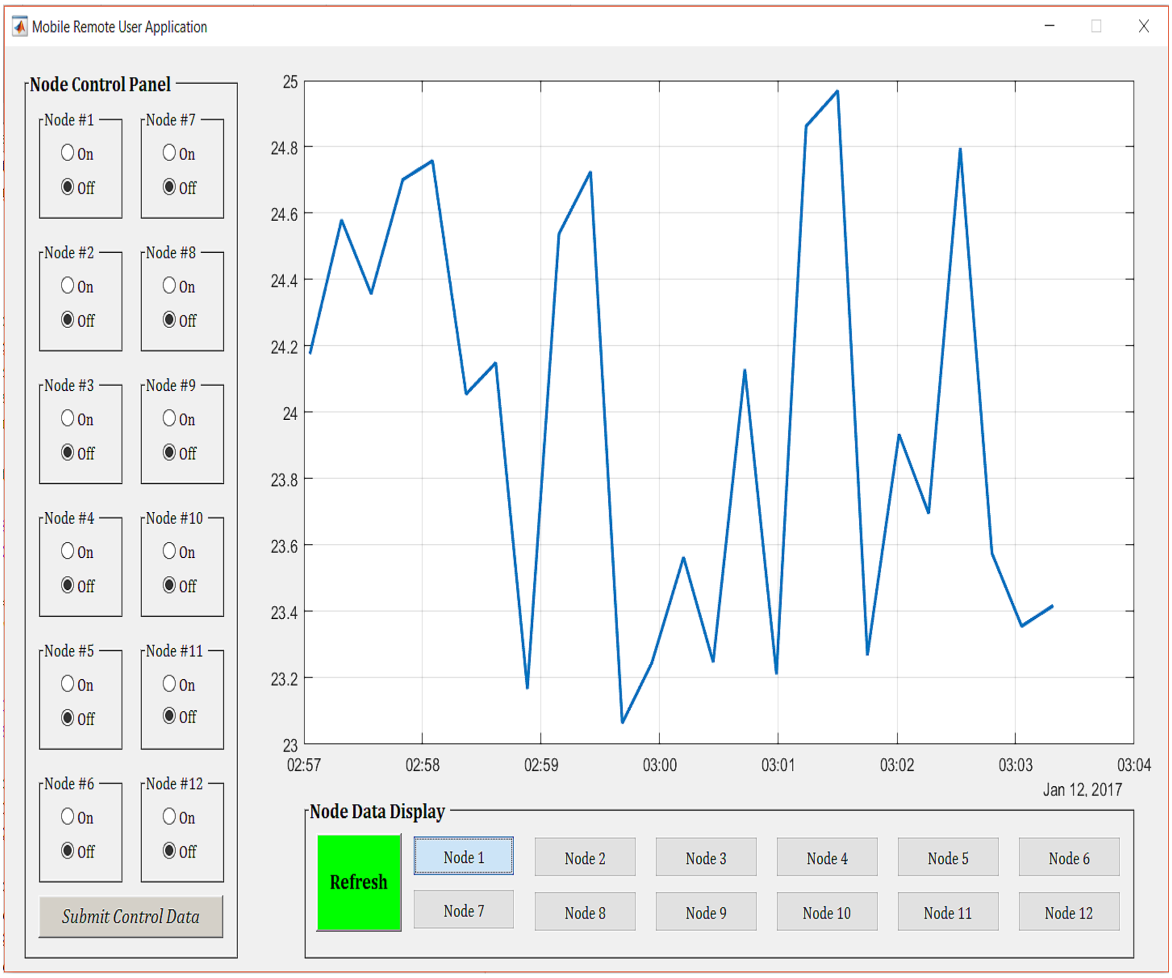


Figure 5.11: GUI of the end-user application

5.3 Testbed Experimental Results and Discussions

The IoT is the most recent technological trend, the main idea behind which is to equip every object with Internet access. This will enable the connected objects to send sensed data to a cloud platform so as to be accessible from anywhere on a continual basis. In addition, it will enable the connected objects to communicate directly with each other. This type of communication is called M2M communication. This section provides detailed proof-of-concept testbed results for SDN and NFV integration in a cloud based 6LoWPAN M2M network. The testbed experiments were carried out in indoor environments. The 6LoWPAN network initiates two successive steps. First, network discovery is performed before carrying out any sensing tasks to make the network topology visible to the customised SDN controller. After the nodes are discovered, the second step is started by executing the sensing application in each M2M node and reporting the sensed data to the gateway prior to their storage in the cloud platform.

Nowadays, there are many software and hardware platforms for implementing M2M sensor networks based on 6LoWPAN. In the proposed SD-NFV approach, the proof-of-concept testbed has three main components:

1. The advanced M2M sensor node and the 6LoWPAN gateway (sink node) are built using Arduino Uno board, which is a microcontroller board based on the ATmega328, with 32 kbytes of flash memory and 2 kbytes of SRAM. On the other hand, the sink node is equipped with Wi-Fi module and extra memory to communicate with IP networks and store sensors' data respectively. These boards can be programmed in an Integrated Development Environment (IDE) using C++ language. The Arduino programmes are compiled using a standard GNU tool chain;
2. IoT middleware is a software based SDN controller responsible for interfacing, managing, and controlling the application layer and network infrastructure (sensing, processing, and communication). It enables the 6LoWPAN nodes to be re-tasked after deployment via high level APIs. The customised SDN controller can also perform complex tasks with the sensed data obtained from the 6LoWPAN sensor nodes, and orchestrate the interaction of high level web-based applications and the M2M sensor nodes;
3. The cloud computing platform is used to provide ubiquitous connectivity via web-based applications. It prompts interoperability between heterogeneous M2M devices and the external IP network. Cloud storage advantages are reducing data retrial latency and providing redundant data paths.

5.3.1 Node Discovery Phase

It is hard and resource intensive to discover each alive node in a 6LoWPAN network manually. Consequently, automated node discovery has been designed to monitor the states of the nodes in the network, which is delegated to the 6LoWPAN coordinator. The network coordinator is integrated with the SDN controller to work in harmony for communication cost reduction between M2M sensor nodes. In addition, integrating the PAN coordinator with the SDN controller can achieve high bandwidth utilisation compared to the traditional 6LoWPAN node discovery, where the cost associated with communication is usually more than that of sensing and processing.

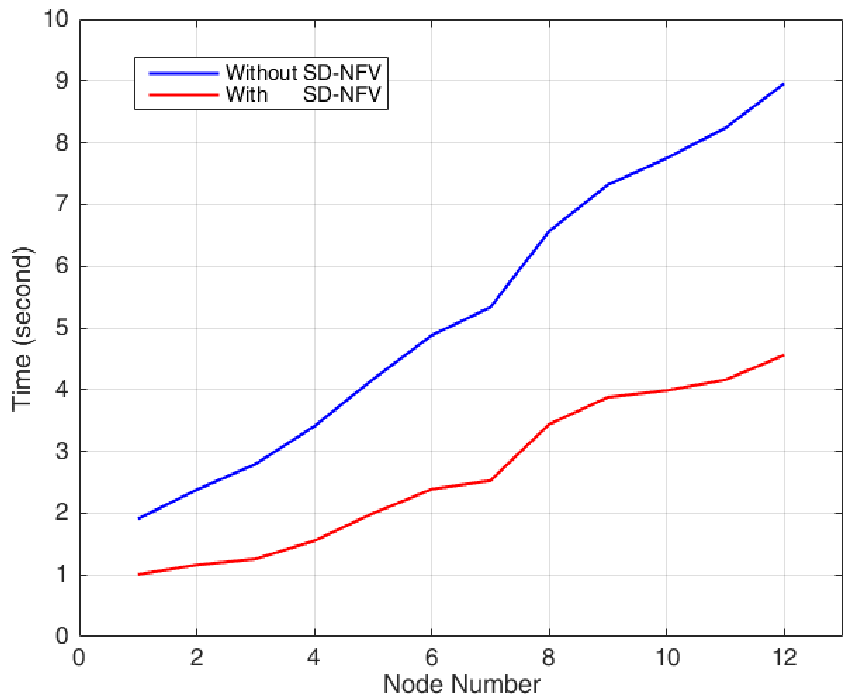


Figure 5.12: Node discovery delay

Figure 5.12 illustrates the node discovery time versus the number of 6LoWPAN nodes. The main objective of the node discovery function is to achieve the lowest number of transmitted packets for a node to be still connected with the customised SDN controller. The proposed topology discovery manager is aimed at making the PAN coordinator with the customised SDN controller responsible for maintaining the status information of all M2M sensor nodes down the hierarchy and reporting the status updates to the SDN controller. According to this information, the SDN controller builds-up two tables: one that contains the connected or alive nodes, and a second that is the flow table where each node is mapped to node IP, cloud APIs, and actions. From Figure 5.12, it is clear that the SD-NFV approach enhances the network discovery process by reducing the topology discovery time by 60% compared to the traditional 6LoWPAN network discovery time. This reduction in network discovery time can be justified due to the decoupling of the control and data planes, whereby the SDN controller will not update the tables frequently. The flow table update takes place when there is no reply from the node, and the corresponding node will be removed from the table or when a new node joins the network and it will be added to both tables.

5.3.2 Execution of Sensing Application

When the execution of the sensing application starts, the SD-NFV gateway becomes responsible for filtering the ingress traffic from both the IEEE 802.15.4 and IEEE 802.11 transceivers. It will perform packet fragmentation and packet assembly depending on the destination address of the packet.

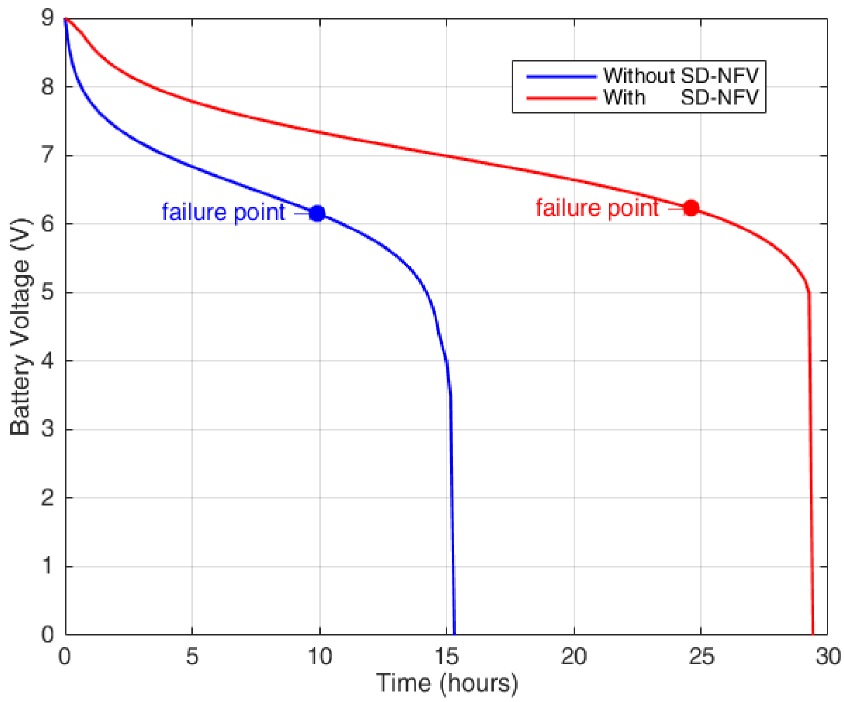


Figure 5.13: 6LoWPAN node lifetime

The traditional 6LoWPAN network is only able to execute a single application. Due to the limited energy source attached to the 6LoWPAN node, the nodes need to use their energies efficiently. The analysis focuses on the advanced nodes because these plays the role of being cluster heads in hierarchical topology. As shown in Figure 5.13, the proposed SD-NFV approach succeeds in enhancing the advanced nodes' lifetime by approximately 65% compared to traditional 6LoWPAN networks without this approach. As stated previously, the 6LoWPAN nodes are characterised by low-data rates, low-energy consumption, low-cost, and generation of flexible topologies. The node lifetime is the time span from deployment to the instant when the node is considered non-functional or failed. A 6LoWPAN node joining the SD-NFV gateway will not deplete its energy more quickly, because unnecessary IPv6 packets transmission is eliminated (i.e. IPv6 headers and fragmentation) and hence, the network lifetime will be enhanced. In addition, unnecessary periodic packet transmissions for maintaining node connectivity are also reduced by the network discovery manager and accordingly, a significant amount of energy is saved. Finally, the Arduino board draws significantly high current compared to other microcontroller boards and hence, it works only for a day not for months or even a year. However, Arduino boards have been chosen in order to investigate the effects of the SD-NFV approach over a very short running time.

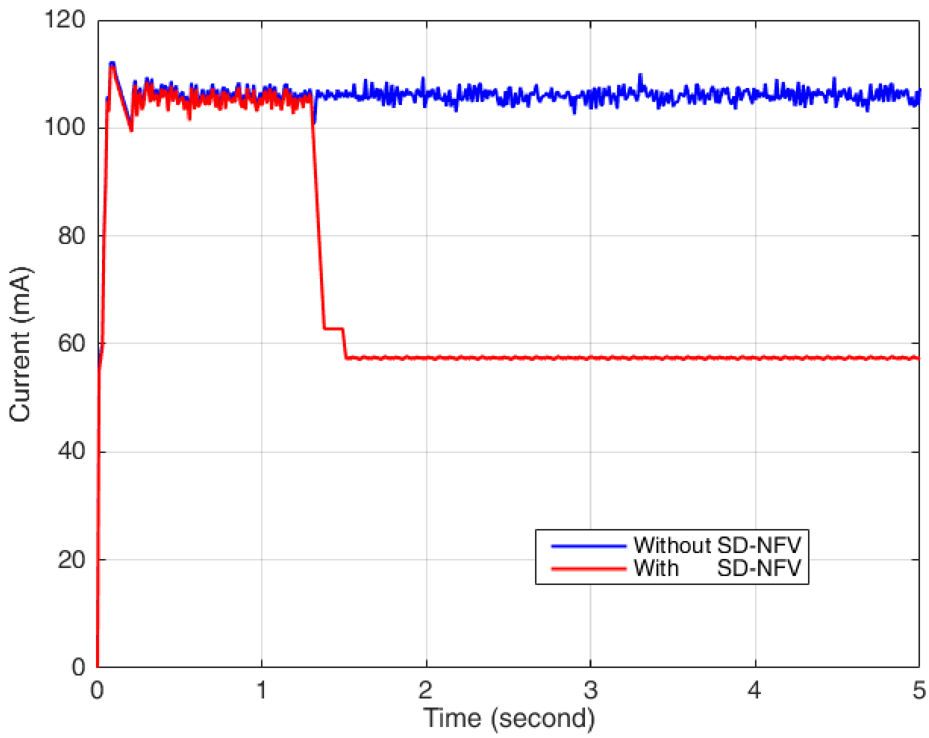


Figure 5.14: First six seconds of the network initialisation phase

Virtualising the network and adaptation layers of the 6LoWPAN protocol stack in the SD-NFV gateway enables the M2M node to perform low-energy sleep mode in order to conserve energy for a long period. Figure 5.14 shows the current drawn from the Arduino Uno board of the advanced node for the first six seconds of network initialisation. The node joining the SD-NFV gateway will not need to have the communication antenna to be on all the time; it will turn on its communication antenna when sending or receiving 6LoWPAN packets. Figure 5.15 shows the current drawn from a node's battery under a periodic traffic scenario. The spikes in the red curve represent the time instance when the sensed data have been transmitted to the SD-NFV gateway.

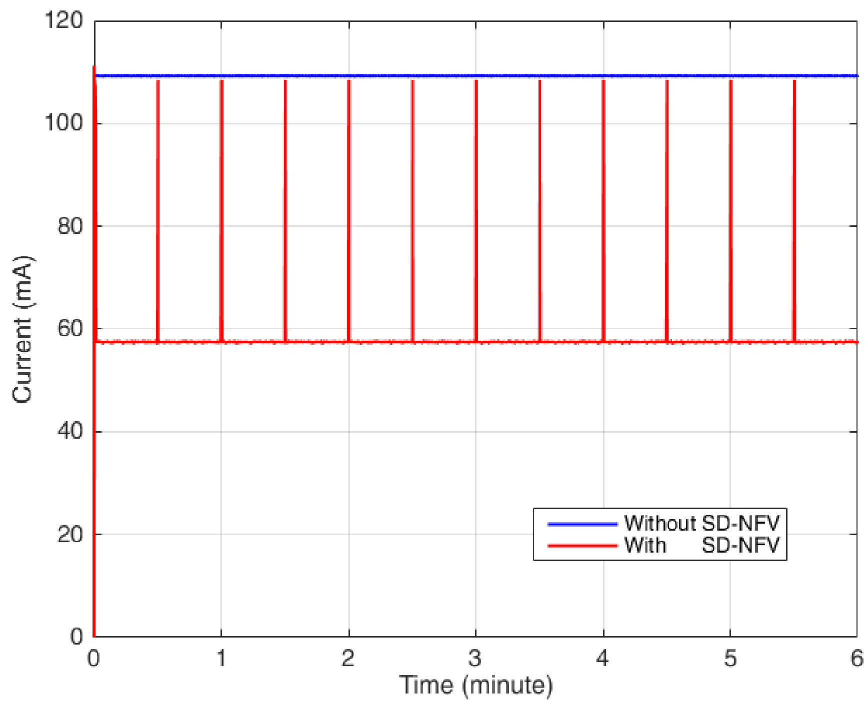
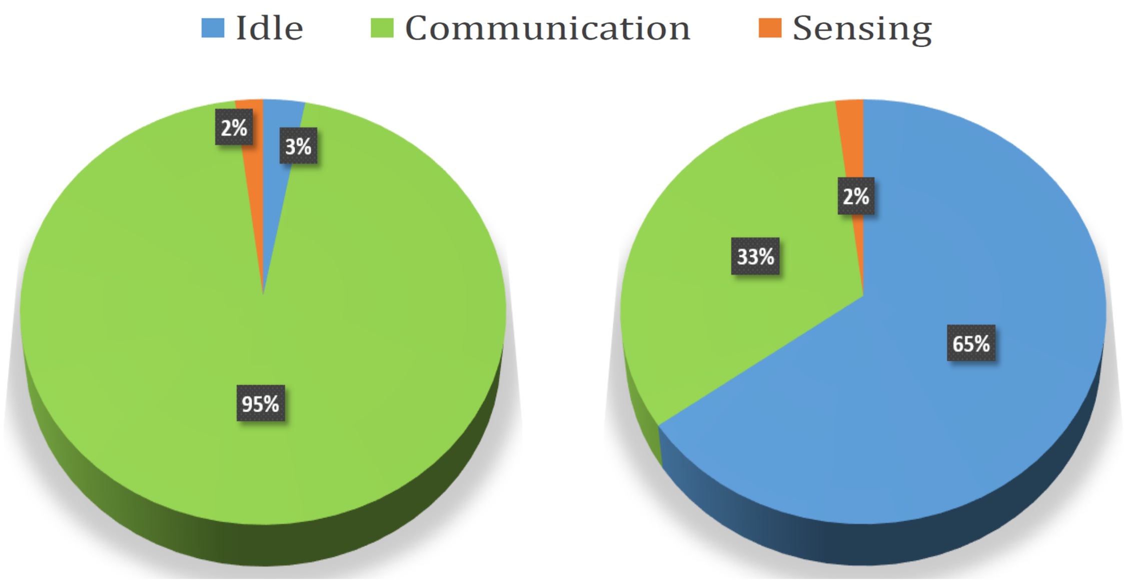


Figure 5.15: Current drawn from 6LoWPAN node under a periodic traffic condition

Figure 5.16 illustrates the relationship between advanced 6LoWPAN node activity as a percentage of the node's residual energy. The experimental testbed results indicate that in a traditional 6LoWPAN network, the node spends all its energy listening to the channel and transmitting a fragmented large packet size of IPv6 (1280 byte). As a result, it depletes its energy more quickly, whilst transmitting fragmented IPv6 datagrams over the LoWPAN links efficiently. While in the proposed SD-NFV approach, IP connectivity at the node level is not necessary, because the customised SDN controller has a virtualisation manager with Sensor Function Virtualisation (SFV), which abstracts IP connectivity from node's protocol stack to the SD-NFV gateway protocol stack. Accordingly, the advanced 6LoWPAN node only sends IEEE 802.15.4 packets (127 byte) and performs sleep mode by turning off its communication antenna. The SD-NFV gateway performs the fragmentation and assembly of IPv6 packets on behalf of the 6LoWPAN nodes. Accordingly, the node can conserve its residual energy, which is powered by batteries only and hence, prolong its lifetime. Finally, the customised SDN controller conserves nodes' energies by an indirect load balancing mechanism.



(b) Traditional 6LoWPAN node (c) SD-NFV 6LoWPAN node

Figure 5.16: 6LoWPAN node activity in relation to the node energy

5.4 Summary

This chapter introduced a new architecture for 6LoWPAN-based sensor networks using programmable network concepts. The proposed approach adopted SDN and NFV technologies with cloud computing platform to enhance network management and extend its lifespan. The developed SD-NFV approach presented a customised SDN controller to alleviate the need for massive packet exchange for network discovery and topology management. The NFV was used to virtualise the most energy harvesting layers to conserve the 6LoWPAN sensor node energy and reduce the time needed by the remote user when provisioning and managing sensor networks. Both SDN and NFV enabled sensor node reconfiguration even after deployment and this alleviated the need for declaring queries and flow patterns to the PAN coordinator during network operation. In addition, NFV techniques automatically enabled the 6LoWPAN gateway to utilise virtualised packet processing functions efficiently on behalf of the M2M sensor nodes . The cloud computing, represented by the ThingSpeak platform, connected the isolated sensor devices and re-tasked them according to both their initial and subsequent configuration settings that deployed by the remote user in the 6LoWPAN gateway. The SD-NFV approach implemented using open source software and hardware platforms to ease their integration in the real-time daily activities such as smart home, smart city and Healthcare. The SD-NFV approach is vendor dependent and provides seamless integration with the IP networks at low cost.

Chapter 6

Conclusions and Future Work

6.1 Conclusions and Discussions

Machine-to-Machine (M2M) communication refers to the execution of automated applications on smart embedded devices that can communicate through wired or wireless networks with or without human intervention. It has dramatically changed the real world around us by providing smart embedded devices with the ability to communicate with each other directly. Many applications can benefit from the advantages of M2M communication, such as healthcare, smart cities, security and transportation. A wireless M2M sensor network consists of spatially distributed embedded sensors that have the ability to sense the surrounding environment. The sensed data are reported towards the sink node in single hop or multi-hop transmissions. Nowadays, M2M sensor networks rely on bi-directional communication links, which in turn enable the M2M sensor nodes to be monitored and accessed individually. The access mechanism can be achieved through the integration of Internet Protocol version 6 (IPv6), such as IPv6 over Low power Wireless Personal Area Network (6LoWPAN) as each node can be accessed via a dedicated IP address. The Internet of Things (IoT) is currently receiving increasing interest from both industry and academia. The IoT can be considered as the extension of the Internet in which it enriches the Internet resources with real world information through data sensed by smart embedded M2M sensor nodes. M2M communication based on 6LoWPAN act as promising technologies for practical realisation, not only for the IoT, but also for the Internet of Nano Things (IoNT) [180] and the Internet of Vehicles (IoV) [181], ultimately leading to the development of what is known as the Internet of Everything (IoE) [182].

It was the aim of this thesis to address and solve different challenges related to IoT applications based on M2M communication. The thesis has made original contributions to the research community as well as opened up a new horizon for future research. Since each chapter of the thesis involved investigating individual and independent research problems, the main thesis's contributions are concluded below to provide an overall picture of the conducted research.

In the literature, there are a great number of mathematical models aimed at modelling the Medium Access Control (MAC) layer and the Physical (PHY) layer, based on the IEEE 802.15.4 standard. However, careful and deep study has revealed that layers, based on 6LoWPAN specifications, are often neglected during the modelling, evaluation and validation of the existing analytical models. Additionally, most of the existing mathematical models are tailored to certain assumptions and thus, are not a fit for the 6LoWPAN specifications provided by the Internet Engineering Task Force (IETF) working group. In order to demonstrate the importance of the proposed mathematical model, Chapter Three provided the evaluation and validation results pertaining to the findings in relation to throughput and end-to-end enhancements. The optimised MAC layer parameters were also presented in that chapter.

An improved joint mathematical model for the 6LoWPAN MAC and PHY layers took the form of a non-beacon enabled CSMA/CA of the IEEE 802.15.4 standard. The proposed mathematical model analysed the behaviour of the IEEE 802.15.4 standard using stochastic representation based on Markov chain modelling. Additionally, a Monte-Carlo based simulation was used to validate the proposed mathematical model. The obtained simulation results showed how the number of nodes affected the generated traffic across the network and the collision probability had a greater impact on the delay and throughput compared to the retransmission failure probability. Moreover, the retransmission failure probability was relative to the maximum backoff and retries allowance. Also, the finite buffer size had implicit impact on this probability. The proposed Markov chain model does not suffer from the limitations uncovered in the literature review, but rather, adopts retransmission based on the received optional acknowledgement to cope with the limitation when implementing the Transfer Communication Protocol (TCP) and User Datagram Protocol (UDP) using 6LoWPAN. The obtained results from the comparison of the simulation and mathematical approaches showed a great similarity in the behaviour of the MAC and PHY layers in terms of delay and throughput. In sum, this model could ease the estimation of network behaviour under different scenarios of a general IoT application. The performance predicted by the proposed mathematical model was very close to that obtained by simulation, which went some way towards enhancing understanding of the MAC and PHY layer interaction. The proposed approach was proved as being accurate, with minimum computational complexity.

A new intelligent approach for optimising the proposed joint analytical model for 6LoWPAN under unsaturated and saturated conditions was developed. In addition, the effects of MAC parameters were investigated in medium to large size networks. An Artificial Neural Network (ANN) was proposed for finding the correlation between the most effective MAC layer parameters as inputs and throughput as output. The various topologies of the ANN were tested by applying one and two hidden layers with different numbers of neurons. Moreover, Levenberg-Marquardt (LM) was used as learning algorithm in the feed-forward ANN structure. Two optimisation techniques were used to optimise the 6LoWPAN MAC layer parameters for a given channel throughput and the number of nodes in the network. A Genetic Algorithm (GA) and Particle Swarm Optimisation (PSO) algorithm were used for deriving the optimal settings of the IEEE 802.15.4 MAC layer parameters in 6LoWPAN networks. The optimised parameters guaranteed the reliability requirements of the application with minimum computational complexity. Both the GA and PSO algorithms performed well the latter had faster convergence to the optimal set than the former.

The obtained results of the optimised model showed that the optimal MAC parameters were feasible for both the unsaturated and saturated conditions with or without a retransmission option. The obtained results were validated by simulation and showed that the channel throughput can be increased by setting the MAC layer with the optimised parameters for a given number of nodes in the network. Moreover, the optimised MAC parameters showed that the throughput was considerably higher than the network set by the default MAC parameters of the IEEE 802.15.4 standard. By utilising the optimal MAC parameters, the 6LoWPAN network throughput was enhanced by 52–63% and end-to-end delay reduced by 54–65%, with the enhancement percentage depending on the number of nodes in the 6LoWPAN network.

All the following research contributions of this thesis are based on 6LoWPAN, which is currently the most widely used technology in M2M sensor networks. Controlled sink mobility was adopted to prolong network lifetime and to enhance packet delivery to sink nodes. The proposed approach was based on multiple mobile sinks to cope the energy hole problem around the sink and provide fair energy depletion by changing sink position. Also, it was focused on developing an energy efficient routing algorithm using the well-known IEEE 802.15.4 communication standard for low-power and low-data rate Personal Area Networks (PANs). Some improvements were made in terms of achieving better network performance under different network models (homogeneous and heterogeneous) and traffic parameters.

The proposed scalable Self-Organised Clustering (SOC) protocol was simulated using MATLAB software and compared among four well-known cluster based routing algorithms: Low-Energy Adaptive Clustering Hierarchy (LEACH), Stable Election Protocol (SEP), Threshold sensitive Energy Efficient Sensor Network (TEEN), and Distributed Energy Efficient Clustering (DEEC). The simulation was conducted using homogeneous and heterogeneous networks. The homogeneous network was divided into two types depending on the heterogeneity level, either two and three levels. The SOC technique involved adopting four mobile sink nodes with controlled mobility travelling across the sensory field along a predefined path with constant speed to collect the data from the cluster heads. The outcomes from simulating the proposed energy-efficient SOC-M2M routing protocol are:

- Multiple mobile sink nodes in large-scale M2M sensor networks are advocated as one possible way to shorten the communication path to a single hop from a cluster head to a mobile sink and thus, enhance network coverage;
- The multiple mobile sinks enhance the performance of an M2M network by increasing the network connectivity and reducing the number of dropped packets by eliminating the energy hole problem;
- The SOC techniques solve the scalability problem by efficiently rotating the cluster head among cluster members based on a heuristic formula;
- The selection and rotation of cluster heads among cluster members lead to fair energy distribution among all sensor nodes and fair traffic balancing across the network. As a result, the network lifetime is increased. Consequently, the number of packets delivered to the sink is also increased;
- The developed approach can potentially reduce the energy consumption compared to the same clustering techniques without sleep mode by adopting a smart sleep scheduling mechanism.

Software-Defined Networking (SDN) is a new network paradigm that decouples the control plane from the data plane and delivers centralised software to support network programmability. The centralised software is called the SDN controller, which is itself controlled by an application layer. It has a global view of the network and instructs the hardware data plane on how to process and forward data. The OpenFlow protocol is one of the southbound interfaces between the controller and the SDN switches for exchanging network information. However, this protocol is not suitable for M2M sensor networks due to its large packet size and communication overhead. Network Function Virtualisation (NFV) provides a reconfigurable feature of sharing the same network resources among all the other nodes in the network. SDN and NFV can work in harmony to achieve network programmability and agile traffic flow with a load balancing mechanism. On the other hand, the cloud commuting platform has expanded the vision of the IoT by providing ubiquitous connectivity to small embedded devices through the border gateway of the M2M sensor network. To distinguish the proposed solution of this thesis, a proof-of-concept testbed was implemented using the SDN and NFV approaches in a cloud-based 6LoWPAN gateway. A customised SDN controller was built using C++ language to adopt the nature of the energy constrained M2M sensor nodes. Also, the implemented testbed can be viewed as a first attempt to analyse the challenges of integrating SDN and NFV together in a IEEE 802.15.4 network, which is characterised by low-power and low-data rate M2M sensor nodes. Currently, the implemented 6LoWPAN testbed has been tested for the specific solution called Software Defined-Network Functioning Virtualisation (SD-NFV). The testbed was built based on open source hardware and software platforms to achieve low-cost M2M sensor nodes and increase its popularity.

The implemented architecture achieves a good performance in terms of the node discovery function in the gateway for global topology construction. The node discovery time has been reduced by 60% compared to the traditional 6LoWPAN approach. Also, the proposed SD-NFV approach is aimed at abstracting the most energy harvesting layers from the 6LoWPAN node and making them virtualised among all the other nodes in the network through the SDN controller. The virtualisation approach enhances the network lifetime; the node joined by the SD-NFV gateway is able to enhance its lifetime by 65% in comparison to the existing node joined by the traditional 6LoWPAN gateway.

Finally, SDN offers a new way to design, deploy, and manage IoT devices by improving the interaction between the customised SDN controller and the network infrastructure. While the NFV reshapes the current network services and makes the IoT service function chaining more agile. The proposed SD-NFV approach is quite suitable for constrained networks, where energy and processing efficiency are the major concern. Furthermore, the SD-NFV gateway can handle bidirectional communication between 6LoWPAN nodes and a remote user.

6.2 Future Research Directions

Despite the substantial contributions of this thesis that pertain to energy-efficient IoT applications, there are a number of open research challenges that are expected to be continuously evolving along with the developments in M2M communications. These new challenges need to be addressed with fresh contributions in order to advance the research area further. This section delineates open research fields and future directions in this regard.

The proposed 6LoWPAN joint mathematical model of the IEEE 802.15.4 standard can be extended to include a sensory field with mobile sensor nodes, rather than stationary nodes or both of them. Also, it will be a challenge to develop an analytical model that deals with sensor node mobility in order to investigate the impact of node mobility on network performance. On the other hand, the optimised MAC layer parameters can be evaluated through testbed implementation and the obtained results can be compared to the simulation results in order to investigate their impacts on 6LoWPAN network behaviour. In addition, different types of M2M sensor nodes based on 6LoWPAN can be used in the evolutionary testbed, which will open a new horizon depicting the scenarios envisioned in industrial IoT applications better.

The presence of mobility in M2M sensor networks requires special attention due to its importance in enhancing the network operation and bringing new challenges to the IoT environment. In this thesis, a self-organised scalable M2M routing protocol has been proposed for a multiple mobile sink sensory field in order to improve network performance. A mobile sink moving trajectory based on the data generation rate is a new challenge that stems from the fact that an M2M sensor network is operating mostly to its areas of interests. This approach will reduce the data gathering latency especially in critical time based IoT applications. On the other hand, the proposed self-organised clustering technique prolongs node lifetime and hence, expand the network life-span. In addition to lifetime expansion, the proposed clustering technique succeeds in solving the scalability issue inherited owing the sensor network nature. At present, most M2M sensor nodes are battery powered, and their lifetime depends on the stored energy. There are a few factors affecting node lifetime that were not considered in this work, such as humidity and battery recovery. The humidity will shorten the node battery lifetime if it is high by filling up the internal space that was allotted for cell discharge. The intermittent discharges, while performing sensing and communication, can be recovered during idle intervals, which leads to prolonging of the battery lifespan. Another challenge is that of the M2M sensor nodes being powered by a stochastic renewable energy source. It is very challenging to analyse and optimise M2M network lifetime under a continuous and unstable energy supply (i.e. solar panel).

A 6LoWPAN based M2M sensor network may face network security problems, as the M2M nodes are exposed to the Internet directly via the border router (gateway). This issue can be addressed by developing a lightweight firewall application or through strong authentication before gaining access to an individual sensor node. The limited resources of the 6LoWPAN sensor nodes make the security solutions more challenging in terms of providing data confidentiality, secure routing, and secure data aggregation.

Research efforts concerning SDN, NFV, and cloud computing are growing rapidly. A proof-of-concept implementation of the proposed SD-NFV architecture on an open source physical network testbed was demonstrated along with the integration of a cloud computing platform. When application development costs are sufficiently lowered, then more ideas can be afforded by the developers. Some new research trends were left out of the scope of this thesis, because of the software and time constraints. These could be taken into account in future research. SDN provides a flexible foundation for future network trends, especially in case of providing centralised control, robust and elastic network scaling, and network programmability. One of the possible improvements to the customised SDN controller is to build global loop-free topology of the sensor network by adding centralised intelligence to the network core. Application-Defined Networking (ADN) will be a new SDN based future research trend in which the applications specify their requirements and let the network satisfy them. These requirements can be load balancing, firewall, and QoS, for example. ADN is a direct consequence of cloud computing platform needs. NFV provides elastic scalability and configuration to network functions. NFV introduces a new future work direction of supporting fully programmable network configuration technologies that make the network simple and flexible. This flexibility in network configuration and scaling is very important, since it could bring the same kinds of functionality to wireless M2M sensor networks as cloud systems brought to computing.

To summarise, this thesis has addressed some of the most recent fundamental issues in M2M sensor networks using 6LoWPAN in an IoT environment. The aim has been to reorient the thinking towards energy-efficiency, self-organised clustering, scalability, and programmable networks in energy constrained M2M sensor networks. The research questions and the proposed solutions in this thesis will act as a starting point for future research on the IoT along with the new upcoming directions.

6.3 The Impact of this Research on Industrial Practice

Previously, the transformation of research results for the communications and networking industry was fragmented and limited, with this lack of coherency meaning such research had a low impact in practice. However, nowadays, most contributions to this industry come from academic research. The most notable innovations incremental improvements to existing knowledge including the enhancements in technology, a global view of customer needs, design refinements, and integration of previously separated technologies (i.e. LoWPAN).

The findings presented in this thesis are the results of comparisons between the developed approaches in this work and the existing schemes. The impact of the thesis contributions on the academic and industry communities relates to three aspects: embedding, convergence, and network applications.

- Embedding: The IoT environment enables the integration of every connected device to have a virtual identity in order to be addressed and accessed over the Internet. This tremendous growth in IoT devices will bring new investment into developing low-cost devices that have the ability to exchange information over a wired or wireless link and reducing the labour cost for some applications, such as smart grids.
- Convergence: 6LoWPAN represents the most promising technology for connected IoT devices, because it provides energy-efficient communication at low cost and integrates directly with the Internet via border router. The energy-efficiency feature will reduce the environmental pollution by reducing the frequency of battery replacement, given that most small IoT devices are battery-powered. Moreover, IPv6 connectivity has led to a breakthrough in network coverage through M2M communication by enabling a wide range of devices from different vendors to work in harmony.
- Network Applications: None of the industrial research has addressed the issues regarding service delivery in a sustained manner. However, this thesis has shown the impact of SDN and NFV in providing a sustainable network by migrating the hardware functions to software packages executed inside the SDN controller. This approach becomes the key to building networks that can: (i) enable the software developers to create new types of services and business models; (ii) reduce capital expenses by allowing network functions to run on, off-the-shelf, vendor-independent hardware; (iii) reduce operational expenses by supporting an autonomous algorithm via the programmability feature in network elements; and (iv) deliver agility and flexibility by rapidly deploying new policies to meet networks' QoS requirements quickly.

References

[1] Cisco, “Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2016-2021,” 2017, White Paper, [Accessed on April 2017].

[2] B. C. Villaverde, R. D. P. Alberola, A. J. Jara, S. Fedor, S. K. Das, and D. Pesch, “Service Discovery Protocols for Constrained Machine-to-Machine Communications,” IEEE Communications Surveys Tutorials, vol. 16, no. 1, pp. 41–60, First 2014.

[3] K. Yoi, H. Yamaguchi, A. Hiromori, A. Uchiyama, T. Higashino, N. Yanagiya, T. Nakatani, A. Tachibana, and T. Hasegawa, “Multi-Dimensional Sensor Data Aggregator for Adaptive Network Management in M2M Communications,” in 2015 IFIP/IEEE International Symposium on Integrated Network Management (IM), May 2015, pp. 1047–1052.

[4] A. Rajandekar and B. Sikdar, “A Survey of MAC Layer Issues and Protocols for Machine-to-Machine Communications,” IEEE Internet of Things Journal, vol. 2, no. 2, pp. 175–186, April 2015.

[5] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, “Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications,” Communications Surveys Tutorials, IEEE, vol. 17, no. 4, pp. 2347–2376, Fourthquarter 2015.

[6] M. Aazam and E. N. Huh, “Fog Computing: The Cloud-IoT/IoE Middleware Paradigm,” IEEE Potentials, vol. 35, no. 3, pp. 40–44, May 2016.

[7] Y. H. Zhu, K. Chi, X. Tian, and V. C. M. Leung, “Network Coding-Based Reliable IPv6 Packet Delivery Over IEEE 802.15.4 Wireless Personal Area Networks,” IEEE Transactions on Vehicular Technology, vol. 65, no. 4, pp. 2219–2230, April 2016.

[8] M. Alam, R. Nielsen, and N. Prasad, “The Evolution of M2M into IoT,” in Communications and Networking (BlackSeaCom), 2013 First International Black Sea Conference on, July 2013, pp. 112–115.

[9] S. Datta, C. Bonnet, and N. Nikaein, “An IoT Gateway Centric Architecture to Provide Novel M2M Services,” in Internet of Things (WF-IoT), 2014 IEEE World Forum on, March 2014, pp. 514–519.

[10] A. Aijaz and A. Aghvami, “Cognitive Machine-to-Machine Communications for Internet-of-Things: A Protocol Stack Perspective,” Internet of Things Journal, IEEE, vol. 2, no. 2, pp. 103–112, April 2015.

[11] ETSI, "Machine-to-Machine Communications (M2M) Interworking between the M2M Architecture and M2M Area Network Technologies," http://www.etsi.org, 2014, [Accessed on April 2017].

[12] D. Boswarthick, O. Elloumi, and O. Hersent, M2M Communications: A Systems Approach, 1st ed. Wiley Publishing, 2012.

[13] IETF, "Transmission of IPv6 Packets over IEEE 802.15.4 Networks, RFC 4944, Network Working Group," https://www.ietf.org, 2007, [Accessed on April 2017].

[14] IEEE 802.15.4-2003, "IEEE Standard for Telecommunications and Information Exchange Between Systems - LAN/MAN Specific Requirements - Part 15: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low Rate Wireless Personal Area Networks (WPAN)," https://standards.ieee.org, 2003, [Accessed on April 2017].

[15] IEEE 802.15.4-2006 (Revision of IEEE Std 802.15.4-2003), "IEEE Standard for Information Technology - Local and Metropolitan Area Networks - Specific Requirements - Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low Rate Wireless Personal Area Networks (WPANs)," https://standards.ieee.org, 2006, [Accessed on April 2017].

[16] Z. Shelby and C. Bormann, 6LoWPAN: The Wireless Embedded Internet. Wiley Publishing, 2010.

[17] Y. Qiu and M. Ma, "A Mutual Authentication and Key Establishment Scheme for M2M Communication in 6LoWPAN Networks," IEEE Transactions on Industrial Informatics, vol. 12, no. 6, pp. 2074–2085, Dec 2016.

[18] S. H. Yang, Wireless Sensor Networks: Principles, Design and Applications. Springer Publishing Company, Incorporated, 2013.

[19] Z. Sheng, H. Wang, C. Yin, X. Hu, S. Yang, and V. C. M. Leung, "Lightweight Management of Resource-Constrained Sensor Devices in Internet of Things," IEEE Internet of Things Journal, vol. 2, no. 5, pp. 402–411, Oct 2015.

[20] G. Moritz, F. Golatowski, C. Lerche, and D. Timmermann, "Beyond 6LoWPAN: Web Services in Wireless Sensor Networks," IEEE Transactions on Industrial Informatics, vol. 9, no. 4, pp. 1795–1805, Nov 2013.

[21] Y. Qiu and M. Ma, "A Mutual Authentication and Key Establishment Scheme for M2M Communication in 6LoWPAN Networks," IEEE Transactions on Industrial Informatics, vol. 12, no. 6, pp. 2074–2085, Dec 2016.

[22] C. Buratti, M. Martalo, R. Verdone, and G. Ferrari, Sensor Networks with IEEE

802.15.4 Systems: Distributed Processing, MAC, and Connectivity. Springer Publishing Company, Incorporated, 2013.

[23] J. Holler, V. Tsiatsis, C. Mulligan, S. Avesand, S. Karnouskos, and D. Boyle, From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence. Elsevier Science, 2014.

[24] J.-P. Vasseur and A. Dunkels, Interconnecting Smart Objects with IP: The Next Internet. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2010.

[25] M. Martalo, S. Busanelli, and G. Ferrari, "Multihop IEEE 802.15.4 Wireless Networks With Finite Node Buffers: Markov Chain-Based Analysis," in Spread Spectrum Techniques and Applications, 2008 IEEE 10th International Symposium on, Aug 2008, pp. 644–648.

[26] P. Park, P. Di Marco, P. Soldati, C. Fischione, and K. Johansson, "A Generalized Markov Chain Model for Effective Analysis of Slotted IEEE 802.15.4," in Mobile Adhoc and Sensor Systems, 2009. MASS '09. IEEE 6th International Conference on, Oct 2009, pp. 130–139.

[27] C. Buratti, "Performance Analysis of IEEE 802.15.4 Beacon-Enabled Mode," Vehicular Technology, IEEE Transactions on, vol. 59, no. 4, pp. 2031–2045, May 2010.

[28] S. Rehman, F. Bhatti, F. Rasool, M. Iqbal, and Z. Sabir, "Modeling the Impact of Deferred Transmission in CSMA/CA Algorithm for IEEE 802.15.4 Using Markov Chain Model," in Multitopic Conference (INMIC), 2011 IEEE 14th International, Dec 2011, pp. 334–339.

[29] Q. Liu and A. Czylwik, "Study on Adaptive Priority-Based Service-Differentiation Scheme for IEEE 802.15.4 Wireless Sensor Networks," in Computers and Communication (ISCC), 2014 IEEE Symposium on, June 2014, pp. 1–6.

[30] P. Luong, T. M. Nguyen, and L. B. Le, "Throughput Analysis and Design for Coexisting WLAN and ZigBee Network," in Vehicular Technology Conference (VTC Fall), 2014 IEEE 80th, Sept 2014, pp. 1–5.

[31] M. Haghighi, Y. Xiang, V. Varadharajan, and B. Quinn, "A Stochastic Time-Domain Model for Burst Data Aggregation in IEEE 802.15.4 Wireless Sensor Networks," Computers, IEEE Transactions on, vol. 64, no. 3, pp. 627–639, March 2015.

[32] R. Vatti and A. Gaikwad, "Modeling and Throughput Analysis of Randomly Deployed IEEE 802.15.4 Based Wireless Personal Area Networks," in Pervasive Computing (ICPC), 2015 International Conference on, Jan 2015, pp. 1–5.

[33] C. Buratti and R. Verdone, "Performance Analysis of IEEE 802.15.4 Non Beacon-

Enabled Mode," Vehicular Technology, IEEE Transactions on, vol. 58, no. 7, pp. 3480–3493, Sept 2009.

[34] J. Um, J. Ahn, and K. Lee, "Evaluation of the Effects of a Grouping Algorithm on IEEE 802.15.4 Networks with Hidden Nodes," Communications and Networks, Journal of, vol. 16, no. 1, pp. 81–91, Feb 2014.

[35] S. Biswas, A. Chandra, and S. Roy, "Cross-Layer Energy Model for Relay Assisted 802.15.4 Networks in a Non-Beacon Enabled Mode," in Advances in Computing, Communications and Informatics (ICACCI, 2014 International Conference on, Sept 2014, pp. 55–61.

[36] M. Rizk, O. Hussein, N. Sadek, and S. Elnoubi, "Analytical Model of Multihop IEEE 802.15.4 for Lossy Medium," in Networking, Sensing and Control (ICNSC), 2015 IEEE 12th International Conference on, April 2015, pp. 202–207.

[37] K. Govindan, A. Azad, K. Bynam, S. Patil, and T. Kim, "Modeling and Analysis of Non Beacon Mode for Low-Rate WPAN," in Consumer Communications and Networking Conference (CCNC), 2015 12th Annual IEEE, Jan 2015, pp. 549–555.

[38] S. C. Ergen, C. Fischione, D. Marandin, and A. Sangiovanni-Vincentelli, "Duty-Cycle Optimization in Unslotted 802.15.4 Wireless Sensor Networks," in Global Telecommunications Conference, 2008. IEEE GLOBECOM 2008. IEEE, Nov 2008, pp. 1–6.

[39] J. Wallace, D. Pesch, S. Rea, and J. Irvine, "Fuzzy Logic Optimisation of MAC Parameters and Sleeping Duty-Cycles in Wireless Sensor Networks," in Vehicular Technology Conference, 2005. VTC-2005-Fall. 2005 IEEE 62nd, vol. 3, Sept 2005, pp. 1824–1828.

[40] Q. Liu and P. Li, "Backoff algorithm optimization and analysis for IEEE 802.15.4 wireless sensor networks," in 2014 9th International Symposium on Communication Systems, Networks Digital Sign (CSNDSP), July 2014, pp. 411–416.

[41] C. Fischione, S. C. Ergen, P. Park, K. H. Johansson, and A. Sangiovanni-Vincentelli, "Medium Access Control Analytical Modeling and Optimization in Unslotted IEEE 802.15.4 Wireless Sensor Networks," in Sensor, Mesh and Ad Hoc Communications and Networks, 2009. SECON '09. 6th Annual IEEE Communications Society Conference on, June 2009, pp. 1–9.

[42] N. Abdeddaim, F. Theoleyre, M. Heusse, and A. Duda, "Adaptive IEEE 802.15.4 MAC for Throughput and Energy Optimization," in Distributed Computing in Sensor Systems (DCOSS), 2013 IEEE International Conference on, May 2013, pp. 223–230.

[43] P. D. Marco, C. Fischione, G. Athanasiou, and P. V. Mekikis, “Harmonizing MAC and Routing in Low Power and Lossy Networks,” in Global Communications Conference (GLOBECOM), 2013 IEEE, Dec 2013, pp. 231–236.

[44] I. Samizadeh, H. B. Kazemian, K. Fisher, and K. Ouazzane, “Adaptive Scalable Rate Control over IEEE 802.15.4 Using Particle Swarm Optimization,” in Information, Communication and Automation Technologies (ICAT), 2013 XXIV International Symposium on, Oct 2013, pp. 1–5.

[45] A. R. Pinto, M. Camada, M. A. R. Dantas, C. Montez, P. Portugal, and F. Vasques, “Genetic Machine Learning Algorithms in the Optimization of Communication Efficiency in Wireless Sensor Networks,” in Industrial Electronics, 2009. IECON ’09. 35th Annual Conference of IEEE, Nov 2009, pp. 2448–2453.

[46] S. Brienza, D. D. Guglielmo, G. Anastasi, M. Conti, and V. Neri, “Strategies for Optimal MAC Parameter Setting In IEEE 802.15.4 Wireless Sensor Networks: A Performance Comparison,” in Computers and Communications (ISCC), 2013 IEEE Symposium on, July 2013, pp. 000 898–000 903.

[47] H. Li and B. Sikdar, “Optimal Parameter Selection for Discrete-Time Throughput-Optimal MAC Protocols,” in Wireless Communications and Networking Conference (WCNC), 2014 IEEE, April 2014, pp. 1526–1531.

[48] M. Elshaikh, M. N. B. M. Warip, O. B. Lynn, R. B. Ahmad, P. Ehkan, F. F. Zakaria, and F. A. A. Fuad, “Energy Consumption Optimization with Ichi Taguchi Method for Wireless Sensor Networks,” in Electronic Design (ICED), 2014 2nd International Conference on, Aug 2014, pp. 493–498.

[49] M. D. Francesco, G. Anastasi, M. Conti, S. K. Das, and V. Neri, “An Adaptive Algorithm for Dynamic Tuning of MAC Parameters in IEEE 802.15.4/ZigBee Sensor Networks,” in Pervasive Computing and Communications Workshops (PERCOM Workshops), 2010 8th IEEE International Conference on, March 2010, pp. 400–405.

[50] G. Yi, J. H. Park, and S. Choi, “Energy-Efficient Distributed Topology Control Algorithm for Low-Power IoT Communication Networks,” IEEE Access, vol. PP, no. 99, pp. 1–1, 2016.

[51] 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,” IEEE Communications Surveys Tutorials, vol. 15, no. 1, pp. 121–144, First 2013.

[52] N. A. Pantazis, S. A. Nikolidakis, and D. D. Vergados, “Energy-Efficient Routing

Protocols in Wireless Sensor Networks: A Survey," IEEE Communications Surveys Tutorials, vol. 15, no. 2, pp. 551–591, Second 2013.

[53] A. A. K. S., K. Ovsthus, and L. M. Kristensen., "An Industrial Perspective on Wireless Sensor Networks - A Survey of Requirements, Protocols, and Challenges," IEEE Communications Surveys Tutorials, vol. 16, no. 3, pp. 1391–1412, Third 2014.

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

[55] H. Lin, L. Wang, and R. Kong, "Energy Efficient Clustering Protocol for Large-Scale Sensor Networks," IEEE Sensors Journal, vol. 15, no. 12, pp. 7150–7160, Dec 2015.

[56] M. M. Shurman, M. F. Al-Mistarihi, and S. Harb, "An Energy-Efficient Coverage Aware Clustering Mechanism for Wireless Sensor Networks," in Communications, Computers and Applications (MIC-CCA), 2012 Mosharaka International Conference on, Oct 2012, pp. 154–158.

[57] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-Efficient Communication Protocol for Wireless Microsensor Networks," in Proceedings of the 33rd Annual Hawaii International Conference on System Sciences, Jan 2000, pp. 10 pp. vol.2–.

[58] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An Application-Specific Protocol Architecture for Wireless Microsensor Networks," IEEE Transactions on Wireless Communications, vol. 1, no. 4, pp. 660–670, Oct 2002.

[59] V. Loscri, G. Morabito, and S. Marano, "A Two-Levels Hierarchy for Low-Energy Adaptive Clustering Hierarchy (TL-LEACH)," in VTC-2005-Fall. 2005 IEEE 62nd Vehicular Technology Conference, 2005., vol. 3, Sept 2005, pp. 1809–1813.

[60] F. Xiangning and S. Yulin, "Improvement on LEACH Protocol of Wireless Sensor Network," in 2007 International Conference on Sensor Technologies and Applications (SENSORCOMM 2007), Oct 2007, pp. 260–264.

[61] R. E. Mohemed, A. I. Saleh, M. Abdelrazzak, and A. S. Samra, "Energy-efficient routing protocols for solving energy hole problem in wireless sensor networks," Computer Networks, pp. –, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1389128616304261

[62] S. Poolsanguan, C. So-In, K. Rujirakul, and K. Udompongsuk, "An Enhanced Cluster Head Selection Criterion of LEACH in Wireless Sensor Networks," in 2016

13th International Joint Conference on Computer Science and Software Engineering (JCSSE), July 2016, pp. 1–7.

[63] M. Sharma and K. Sharma, "An Energy Efficient Extended LEACH (EEE-LEACH)," in Communication Systems and Network Technologies (CSNT), 2012 International Conference on, May 2012, pp. 377–382.

[64] 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, Oct 2013, pp. 158–163.

[65] A. Roshini and H. Anandakumar, "Hierarchical Cost Effective LEACH for Heterogeneous Wireless Sensor Networks," in Advanced Computing and Communication Systems, 2015 International Conference on, Jan 2015, pp. 1–7.

[66] V. Sharma and D. S. Saini, "Performance Investigation of Advanced Multi-Hop and Single-Hop Energy Efficient LEACH Protocol with Heterogeneous Nodes in Wireless Sensor Networks," in Advances in Computing and Communication Engineering (ICACCE), 2015 Second International Conference on, May 2015, pp. 192–197.

[67] A. Manjeshwar and D. P. Agrawal, "TEEN: a Routing Protocol for Enhanced Efficiency in Wireless Sensor Networks," in Proceedings 15th International Parallel and Distributed Processing Symposium. IPDPS 2001, April 2001, pp. 2009–2015.

[68] A. Manjeshwar and D. P. Agrawal, "APTEEN: a Hybrid Protocol for Efficient Routing and Comprehensive Information Retrieval in Wireless," in Proceedings 16th International Parallel and Distributed Processing Symposium, April 2002, pp. 8 pp–.

[69] O. Abouelkhair and A. Elsaadny, "Multipath Adaptive Periodic Threshold-Sensitive Energy Efficient Protocol for Wireless Sensor Networks," in 2014 Sixth International Conference on Computational Intelligence, Communication Systems and Networks, May 2014, pp. 33–37.

[70] Anjali, A. Garg, and Suhali, "Distance Adaptive Threshold Sensitive Energy Efficient Sensor Network (DAPTEEN) Protocol in WSN," in Signal Processing, Computing and Control (ISPCC), 2015 International Conference on, Sept 2015, pp. 114–119.

[71] G. Smaragdakis, I. Matta, and A. Bestavros, "SEP: a Stable Election Protocol for Clustered Heterogeneous Wireless Sensor Networks," in Second International workshop on sensor and actor network protocols and applications (SANPA 2004), 2004.

[72] S. Iqbal, S. B. Shagrithaya, S. G. G. P, and M. B. S, "Performance Analysis of Stable Election Protocol and its Extensions in WSN," in 2014 IEEE International Conference on Advanced Communications, Control and Computing Technologies, May 2014, pp. 744–748.

[73] A. A. Malluh, K. M. Elleithy, Z. Qawaqneh, R. J. Mstafa, and A. Alanazi, "EM-SEP: An Efficient Modified Stable Election Protocol," in American Society for Engineering Education (ASEE Zone 1), 2014 Zone 1 Conference of the, April 2014, pp. 1–7.

[74] D. Singh and C. K. Panda, "Performance Analysis of Modified Stable Election Protocol in Heterogeneous WSN," in Electrical, Electronics, Signals, Communication and Optimization (EESCO), 2015 International Conference on, Jan 2015, pp. 1–5.

[75] L. Qing, Q. Zhu, and M. Wang, "Design of a Distributed Energy-Efficient Clustering Algorithm for Heterogeneous Wireless Sensor Networks," Computer communications, vol. 29, no. 12, pp. 2230–2237, 2006.

[76] S. B. Alla, A. Ezzati, A. Mouhsen, A. B. Hssane, and M. L. Hasnaoui, "Balanced and centralized distributed energy efficient clustering for heterogeneous wireless sensor networks," in 2011 3rd International Conference on Next Generation Networks and Services (NGNS), Dec 2011, pp. 39–44.

[77] Z. Kaili, Z. Lihua, and W. Xu, "An improved DEEC algorithm with isolated nodes," in IET International Conference on Information and Communications Technologies (IETICT 2013), April 2013, pp. 228–232.

[78] M. Bagouri, S. Chakkor, and A. Hajraoui, "Improving Threshold Distributed Energy Efficient Clustering Algorithm for Heterogeneous Wireless Sensor Networks," in 2014 Third IEEE International Colloquium in Information Science and Technology (CIST), Oct 2014, pp. 430–434.

[79] M. A. Hasnat, M. Akbar, Z. Iqbal, Z. A. Khan, U. Qasim, and N. Javaid, "Bio Inspired Distributed Energy Efficient Clustering for Wireless Sensor Networks," in 2015 5th National Symposium on Information Technology: Towards New Smart World (NSITNSW), Feb 2015, pp. 1–7.

[80] T. Tiwari and N. R. Roy, "Modified deec: A varying power level based clustering technique for wsns," in 2015 International Conference on Computer and Computational Sciences (ICCCS), Jan 2015, pp. 170–176.

[81] M. Shaji and A. S., "Distributed Energy Efficient Heterogeneous Clustering in Wireless Sensor Network," in 2015 Fifth International Conference on Advances in Computing and Communications (ICACC), Sept 2015, pp. 130–134.

[82] J. Ma, W. Lou, Y. Wu, X. Y. Li, and G. Chen, "Energy Efficient TDMA Sleep Scheduling in Wireless Sensor Networks," in IEEE INFOCOM 2009, April 2009, pp. 630–638.

[83] Z. Yuan, L. Wang, L. Shu, T. Hara, and Z. Qin, "A Balanced Energy Consumption Sleep Scheduling Algorithm in Wireless Sensor Networks," in 2011 7th International Wireless Communications and Mobile Computing Conference, July 2011, pp. 831–835.

[84] Y. Wu, S. Fahmy, and N. B. Shroff, "Optimal QoS-aware Sleep/Wake Scheduling for Time-Synchronized Sensor Networks," in 2006 40th Annual Conference on Information Sciences and Systems, March 2006, pp. 924–930.

[85] H. Yoo, M. Shim, and D. Kim, "Dynamic Duty-Cycle Scheduling Schemes for Energy-Harvesting Wireless Sensor Networks," IEEE Communications Letters, vol. 16, no. 2, pp. 202–204, February 2012.

[86] J. Deng, Y. S. Han, W. B. Heinzelman, and P. K. Varshney, "Balanced-Energy Sleep Scheduling Scheme for High Density Cluster-Based Sensor Networks," in Applications and Services in Wireless Networks, 2004. ASWN 2004. 2004 4th Workshop on, Aug 2004, pp. 99–108.

[87] D. Shuman and M. Liu, "Optimal Sleep Scheduling for a Wireless Sensor Network Node," in Signals, Systems and Computers, 2006. ACSSC '06. Fortieth Asilomar Conference on, Oct 2006, pp. 1337–1341.

[88] B. Nazir and H. Hasbullah, "Dynamic Sleep Scheduling for Minimizing Delay in Wireless Sensor Network," in 2011 Saudi International Electronics, Communications and Photonics Conference (SIECPC), April 2011, pp. 1–5.

[89] Z. Xinlian and Q. Wenhao, "Sensor network energy saving sleep scheduling algorithm research," in 2014 International Conference on Information and Communications Technologies (ICT 2014), May 2014, pp. 1–5.

[90] G. Pradeebaa and N. Lavanis, "Network lifetime improvement using routing algorithm with sleep mode in wireless sensor network," in 2016 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET), March 2016, pp. 1572–1575.

[91] M. I. Khan, W. N. Gansterer, and G. Haring, "Static vs. Mobile Sink: The Influence of Basic Parameters on Energy Efficiency in Wireless Sensor Networks ," Computer Communications, vol. 36, no. 9, pp. 965–978, May 2013.

[92] A. Nayak and I. Stojmenovic, Wireless Sensor and Actuator Networks: Algorithms and Protocols for Scalable Coordination and Data Communication. New York, NY, USA: Wiley-Interscience, 2010.

[93] X. Xu and W. Liang, "Placing Optimal Number of Sinks in Sensor Networks for Network Lifetime Maximization," in 2011 IEEE International Conference on Communications (ICC), June 2011, pp. 1–6.

[94] B. Paul and M. A. Matin, "Optimal Geometrical Sink Location Estimation for Two-Tiered Wireless Sensor Networks," IET Wireless Sensor Systems, vol. 1, no. 2, pp. 74–84, June 2011.

[95] C. Tunca, S. Isik, M. Y. Donmez, and C. Ersoy, "Ring Routing: An Energy-Efficient Routing Protocol for Wireless Sensor Networks with a Mobile Sink," IEEE Transactions on Mobile Computing, vol. 14, no. 9, pp. 1947–1960, Sept 2015.

[96] V. Kumar, A. Kumar, Gaurav, and M. Singh, "Improving Network Lifetime Reporting Delay in Wireless Sensor Networks Using Multiple Mobile Sinks," in 2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom), March 2016, pp. 1675–1678.

[97] M. Abo-Zahhad, S. M. Ahmed, N. Sabor, and S. Sasaki, "Mobile Sink-Based Adaptive Immune Energy-Efficient Clustering Protocol for Improving the Lifetime and Stability Period of Wireless Sensor Networks," IEEE Sensors Journal, vol. 15, no. 8, pp. 4576–4586, Aug 2015.

[98] H. Karl and A. Willig, Protocols and Architectures for Wireless Sensor Networks. John Wiley & Sons, 2005.

[99] C. F. Cheng and H. Lee, "Data Gathering in Wireless Sensor Networks with Uncontrolled Sink Mobility," in 2016 IEEE 83rd Vehicular Technology Conference (VTC Spring), May 2016, pp. 1–5.

[100] F. Restuccia and S. K. Das, "Lifetime optimization with qos of sensor networks with uncontrollable mobile sinks," in 2015 IEEE 16th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM), June 2015, pp. 1–9.

[101] K. Akkaya, F. Senel, A. Thimmapuram, and S. Uludag, "Distributed Recovery from Network Partitioning in Movable Sensor/Actor Networks via Controlled Mobility," IEEE Transactions on Computers, vol. 59, no. 2, pp. 258–271, Feb 2010.

[102] O. Cayirpunar, E. Kadioglu-Urtis, and B. Tavli, "Optimal Base Station Mobility Patterns for Wireless Sensor Network Lifetime Maximization," IEEE Sensors Journal, vol. 15, no. 11, pp. 6592–6603, Nov 2015.

[103] A. Abuarqoub, "Optimal Mobile Sink Selection Scheme for Multiple Sink Mobile Wireless Sensor Networks," in 2015 European Intelligence and Security Informatics Conference, Sept 2015, pp. 169–172.

[104] M. T. Nuruzzaman and H. W. Ferng, "A low energy consumption routing protocol for mobile sensor networks with a path-constrained mobile sink," in 2016 IEEE International Conference on Communications (ICC), May 2016, pp. 1–6.

[105] W. Stallings, Foundations of Modern Networking: SDN, NFV, QoE, IoT, and Cloud. Pearson Education, 2016.

[106] Open Networking Foundation, https://www.opennetworking.org/, [Accessed on April 2017].

[107] J. Doherty, SDN and NFV Simplified: A Visual Guide to Understanding Software Defined Networks and Network Function Virtualization, 1st ed. Addison-Wesley Professional, 2016.

[108] T. D. Nadeau and K. Gray, SDN: Software Defined Networks. "O'Reilly Media, Inc.", 2013.

[109] F. Hu, Network Innovation Through OpenFlow and SDN: Principles and Design, 1st ed. Boca Raton, FL, USA: CRC Press, Inc., 2014.

[110] P. A. Morreale and J. M. Anderson, Software Defined Networking: Design and Deployment. Boca Raton, FL, USA: CRC Press, Inc., 2014.

[111] D. F. Macedo, D. Guedes, L. F. M. Vieira, M. A. M. Vieira, and M. Nogueira, "Programmable Networks - From Software-Defined Radio to Software-Defined Networking," IEEE Communications Surveys Tutorials, vol. 17, no. 2, pp. 1102–1125, Secondquarter 2015.

[112] T. Luo, H. P. Tan, and T. Q. S. Quek, "Sensor OpenFlow: Enabling Software-Defined Wireless Sensor Networks," IEEE Communications Letters, vol. 16, no. 11, pp. 1896–1899, November 2012.

[113] A. Mahmud, R. Rahmani, and T. Kanter, "Deployment of Flow-Sensors in Internet of Things' Virtualization via OpenFlow," in 2012 Third FTRA International Conference on Mobile, Ubiquitous, and Intelligent Computing, June 2012, pp. 195–200.

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

[115] Y. J. Chen, Y. H. Shen, and L. C. Wang, "Traffic-Aware Load Balancing for M2M Networks Using SDN," in 2014 IEEE 6th International Conference on Cloud Computing Technology and Science, Dec 2014, pp. 668–671.

[116] A. D. Gante, M. Aslan, and A. Matrawy, "Smart Wireless Sensor Network Man-

agement Based on Software-Defined Networking," in 2014 27th Biennial Symposium on Communications (QBSC), June 2014, pp. 71–75.

[117] H. Kim, J. Kim, and Y. B. Ko, "Developing a Cost-Effective OpenFlow Testbed for Small-Scale Software Defined Networking," in Advanced Communication Technology (ICACT), 2014 16th International Conference on, Feb 2014, pp. 758–761.

[118] 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 Computer Communications (INFOCOM), 2015 IEEE Conference on, April 2015, pp. 513–521.

[119] S. Han and S. Lee, "Implementing SDN and Network-Hypervisor Based Programmable Network Using Pi Stack Switch," in Information and Communication Technology Convergence (ICTC), 2015 International Conference on, Oct 2015, pp. 579–581.

[120] F. Olivier, G. Carlos, and N. Florent, "SDN Based Architecture for Clustered WSN," in Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), 2015 9th International Conference on, July 2015, pp. 342–347.

[121] M. Ariman, G. Seçinti, M. Erel, and B. Canberk, "Software Defined Wireless Network Testbed Using Raspberry Pi of Switches with Routing Add-on," in Network Function Virtualization and Software Defined Network (NFV-SDN), 2015 IEEE Conference on, Nov 2015, pp. 20–21.

[122] S. Salsano, P. L. Ventre, F. Lombardo, G. Siracusano, M. Gerola, E. Salvadori, M. Santuari, M. Campanella, and L. Prete, "Hybrid IP/SDN Networking: Open Implementation and Experiment Management Tools," IEEE Transactions on Network and Service Management, vol. 13, no. 1, pp. 138–153, March 2016.

[123] T. Huang, F. R. Yu, C. Zhang, J. Liu, J. Zhang, and J. Liu, "A Survey on Large-scale Software Defined Networking (SDN) Testbeds: Approaches and Challenges," IEEE Communications Surveys Tutorials, vol. PP, no. 99, pp. 1–1, 2016.

[124] R. Chayapathi, S. F. Hassan, and P. Shah, Network Functions Virtualization (NFV) with a Touch of SDN, 1st ed. Indianapolis, Indiana, USA: Pearson Education, Inc., 2016.

[125] R. Mijumbi, J. Serrat, J. L. Gorricho, N. Bouten, F. D. Turck, and R. Boutaba, "Network Function Virtualization: State-of-the-Art and Research Challenges," IEEE Communications Surveys Tutorials, vol. 18, no. 1, pp. 236–262, Firstquarter 2016.

[126] Q. Duan, N. Ansari, and M. Toy, "Software-Defined Network Virtualization: an

Architectural Framework for Integrating SDN and NFV for Service Provisioning in Future Networks," IEEE Network, vol. 30, no. 5, pp. 10–16, September 2016.

[127] Y. Li and M. Chen, "Software-defined network function virtualization: A survey," IEEE Access, vol. 3, pp. 2542–2553, 2015.

[128] C. Mouradian, T. Saha, J. Sahoo, R. Glitho, M. Morrow, and P. Polakos, "NFV Based Gateways for Virtualized Wireless Sensor Networks: A Case Study," in 2015 IEEE International Conference on Communication Workshop (ICCW), June 2015, pp. 1883–1888.

[129] N. Bizanis and F. A. Kuipers, "Sdn and virtualization solutions for the internet of things: A survey," IEEE Access, vol. 4, pp. 5591–5606, 2016.

[130] J. Rittinghouse and J. Ransome, Cloud Computing: Implementation, Management, and Security, 1st ed. Boca Raton, FL, USA: CRC Press, Inc., 2009.

[131] D. C. Marinescu, Cloud Computing: Theory and Practice, 1st ed. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2013.

[132] K. Hwang, J. Dongarra, and G. C. Fox, Distributed and Cloud Computing: From Parallel Processing to the Internet of Things, 1st ed. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2011.

[133] A. Flammini and E. Sisinni, "Wireless Sensor Networking in the Internet of Things and Cloud Computing Era," Procedia Engineering, vol. 87, no. Supplement C, pp. 672 – 679, 2014, EUROSENSORS 2014, the 28th European Conference on Solid-State Transducers.

[134] S. Smoot and N. Tan, Private Cloud Computing: Consolidation, Virtualization, and Service-oriented Infrastructure. Morgan Kaufmann, 2012.

[135] D. McLoughlin, E. O'Connell, W. Elgenaidi, J. Coleman, and T. Newe, "Review and Evaluation of WSN Simulation Tools in a Cloud Based Environment," in 2016 10th International Conference on Sensing Technology (ICST), Nov 2016, pp. 1–6.

[136] C. Zhu, V. C. M. Leung, L. T. Yang, and L. Shu, "Collaborative Location-Based Sleep Scheduling for Wireless Sensor Networks Integratedwith Mobile Cloud Computing," IEEE Transactions on Computers, vol. 64, no. 7, pp. 1844–1856, July 2015.

[137] ZigBee Alliance, "Zigbee Overview," http://www.zigbee.org, 2003, [Accessed on April 2017].

[138] T. R. Banniza, D. Boettle, R. Klotsche, P. Schefczik, M. Soellner, and K. Wuenstel, "A European Approach to a Clean Slate Design for the Future Internet," Bell Labs Technical Journal, vol. 14, no. 2, pp. 5–22, Summer 2009.

[139] W. Ding, Z. Yan, and R. H. Deng, "A Survey on Future Internet Security Architectures," IEEE Access, vol. 4, pp. 4374–4393, 2016.

[140] L. M. Correia, H. Abramowicz, M. Johnsson, and K. Wnstel, Architecture and Design for the Future Internet: 4WARD Project, 1st ed. Springer Publishing Company, Incorporated, 2011.

[141] G. Tselentis and A. Galis, Towards the Future Internet: Emerging Trends from European Research. IOS Press, 2010.

[142] S. Tennina, A. Koubâa, R. Daidone, M. Alves, P. Jurčík, R. Severino, M. Tiloca, J. Hauer, N. Pereira, G. Dini et al., IEEE 802.15.4 and ZigBee as Enabling Technologies for Low-Power Wireless Systems with Quality-of-Service Constraints, ser. SpringerBriefs in Electrical and Computer Engineering. Springer Berlin Heidelberg, 2013.

[143] C. Buratti, M. Martalo, R. Verdone, and G. Ferrari, Sensor Networks with IEEE 802.15.4 Systems: Distributed Processing, MAC, and Connectivity. Springer Publishing Company, Incorporated, 2013.

[144] D. D. Guglielmo, F. Restuccia, G. Anastasi, M. Conti, and S. K. Das, "Accurate and Efficient Modeling of 802.15.4 Unslotted CSMA/CA through Event Chains Computation," IEEE Transactions on Mobile Computing, vol. 15, no. 12, pp. 2954–2968, Dec 2016.

[145] T. Cooklev, Wireless Communication Standards: A Study of IEEE 802.11, 802.15, 802.16. Wiley Publishing, 2011.

[146] Queueing Networks and Markov Chains: Modeling and Performance Evaluation with Computer Science Applications.

[147] G. Giambene, Queuing Theory and Telecommunications. Springer, 2005.

[148] A. Faridi, M. R. Palattella, A. Lozano, M. Dohler, G. Boggia, L. A. Grieco, and P. Camarda, "Comprehensive Evaluation of the IEEE 802.15.4 MAC Layer Performance With Retransmissions," IEEE Transactions on Vehicular Technology, vol. 59, no. 8, pp. 3917–3932, Oct 2010.

[149] M. Zuniga and B. Krishnamachari, "Analyzing the Transitional Region in Low Power Wireless Links," in Sensor and Ad Hoc Communications and Networks, 2004. IEEE SECON 2004. 2004 First Annual IEEE Communications Society Conference on, Oct 2004, pp. 517–526.

[150] M. Zamalloa and B. Krishnamachari, "An Analysis of Unreliability and Asymmetry in Low-Power Wireless Links," ACM Transactions on Sensor Networks (TOSN), vol. 3, no. 2, p. 7, 2007.

[151] P. Bremaud, Markov Chains: Gibbs Fields, Monte Carlo Simulation, and Queues, ser. Texts in Applied Mathematics. Springer New York, 2013.

[152] D. Chaturvedi, Soft Computing: Techniques and its Applications in Electrical Engineering, ser. Studies in Computational Intelligence. Springer Berlin Heidelberg, 2008.

[153] B. Muller and J. Reinhardt, Neural Networks: An Introduction. Springer, 2014.

[154] J. Sarangapani, Neural Network Control of Nonlinear Discrete-Time Systems. Boca Raton, FL, USA: CRC Press, Inc., 2006.

[155] M. Hagan, H. Demuth, and M. Beale, Neural Network Design. Campus Pub. Service, University of Colorado Bookstore, 2002.

[156] S. Sivanandam and S. Deepa, Introduction to Genetic Algorithms. Springer Berlin Heidelberg, 2010.

[157] M. Clerc, Particle Swarm Optimization, ser. ISTE. Wiley, 2013.

[158] L. M. Borges, F. J. Velez, and A. S. Lebres, "Survey on the Characterization and Classification of Wireless Sensor Network Applications," IEEE Communications Surveys Tutorials, vol. 16, no. 4, pp. 1860–1890, Fourthquarter 2014.

[159] J. Ren, Y. Zhang, K. Zhang, A. Liu, J. Chen, and X. S. Shen, "Lifetime and Energy Hole Evolution Analysis in Data-Gathering Wireless Sensor Networks," IEEE Transactions on Industrial Informatics, vol. 12, no. 2, pp. 788–800, April 2016.

[160] G. Xie and F. Pan, "Cluster-Based Routing for the Mobile Sink in Wireless Sensor Networks With Obstacles," IEEE Access, vol. 4, pp. 2019–2028, 2016.

[161] M. I. Chidean, E. Morgado, M. Sanromán-Junquera, J. Ramiro-Bargueño, J. Ramos, and A. J. Caamaño, "Energy Efficiency and Quality of Data Reconstruction Through Data-Coupled Clustering for Self-Organized Large-Scale WSNs," IEEE Sensors Journal, vol. 16, no. 12, pp. 5010–5020, June 2016.

[162] Y. Gu, F. Ren, Y. Ji, and J. Li, "The Evolution of Sink Mobility Management in Wireless Sensor Networks: A Survey," IEEE Communications Surveys Tutorials, vol. 18, no. 1, pp. 507–524, Firstquarter 2016.

[163] J. N. Al-Karaki and A. E. Kamal, "Routing Techniques in Wireless Sensor Networks: A Survey," IEEE Wireless Communications, vol. 11, no. 6, pp. 6–28, Dec 2004.

[164] N. Mukherjee, S. Neogy, and S. Roy, Building Wireless Sensor Networks: Theoretical and Practical Perspectives. CRC Press, 2016.

[165] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-Efficient Communication Protocol for Wireless Microsensor Networks," in System Sciences,

2000. Proceedings of the 33rd Annual Hawaii International Conference on, Jan 2000, pp. 1–10.

[166] D. S. Deif and Y. Gadallah, "Classification of Wireless Sensor Networks Deployment Techniques," IEEE Communications Surveys Tutorials, vol. 16, no. 2, pp. 834–855, Second 2014.

[167] C. Zhu, H. Wang, X. Liu, L. Shu, L. T. Yang, and V. C. M. Leung, "A Novel Sensory Data Processing Framework to Integrate Sensor Networks With Mobile Cloud," IEEE Systems Journal, vol. 10, no. 3, pp. 1125–1136, Sept 2016.

[168] S. Singh, A. Malik, and R. Kumar, "Energy efficient heterogeneous {DEEC} protocol for enhancing lifetime in {WSNs}," Engineering Science and Technology, an International Journal, pp. –, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S2215098616301562

[169] A. Badi, I. Mahgoub, M. Slavik, and M. Ilyas, "Investigation of the effects of network density on the optimal number of clusters in hierarchical wireless sensor networks (wsns)," in 7th International Symposium on High-capacity Optical Networks and Enabling Technologies, Dec 2010, pp. 171–177.

[170] Y. Weng, W. Jia, and G. Wang, "Energy Efficient Data Gathering Scheme for Wireless Sensor Networks with Static and Mobile Sinks," in 2010 2nd IEEE InternationalConference on Network Infrastructure and Digital Content, Sept 2010, pp. 748–752.

[171] H. Salarian, K. W. Chin, and F. Naghdy, "An Energy-Efficient Mobile-Sink Path Selection Strategy for Wireless Sensor Networks," IEEE Transactions on Vehicular Technology, vol. 63, no. 5, pp. 2407–2419, Jun 2014.

[172] A. Lara, A. Kolasani, and B. Ramamurthy, "Network Innovation using OpenFlow: A Survey," IEEE Communications Surveys Tutorials, vol. 16, no. 1, pp. 493–512, First 2014.

[173] B. A. A. Nunes, M. Mendonca, X. N. Nguyen, K. Obraczka, and T. Turletti, "A Survey of Software-Defined Networking: Past, Present, and Future of Programmable Networks," IEEE Communications Surveys Tutorials, vol. 16, no. 3, pp. 1617–1634, Third 2014.

[174] Arduino, https://www.arduino.cc/, [Accessed on April 2017].

[175] J. Batalle, J. F. Riera, E. Escalona, and J. A. Garcia-Espin, "On the Implementation of NFV over an OpenFlow Infrastructure: Routing Function Virtualization," in Future Networks and Services (SDN4FNS), 2013 IEEE SDN for, Nov 2013, pp. 1–6.

[176] Arduino pico IPv6 stack (pIPv6), https://github.com/telecombretagne/Arduino-pIPv6Stack, [Accessed on April 2017].

[177] IETF, "Neighbor Discovery for IP version 6 (IPv6)," https://tools.ietf.org/html/rfc4861, 2007, [Accessed on April 2017].

[178] IETF, "Neighbor Discovery Optimization for IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs)," https://tools.ietf.org/html/rfc6775, 2012, [Accessed on April 2017].

[179] Internet-of-Things Platform, https://www.thingspeak.com/, [Accessed on April 2017].

[180] N. Akkari, P. Wang, J. M. Jornet, E. Fadel, L. Elrefaei, M. G. A. Malik, S. Almasri, and I. F. Akyildiz, "Distributed Timely Throughput Optimal Scheduling for the Internet of Nano-Things," IEEE Internet of Things Journal, vol. 3, no. 6, pp. 1202–1212, Dec 2016.

[181] O. Kaiwartya, A. H. Abdullah, Y. Cao, A. Altameem, M. Prasad, C. T. Lin, and X. Liu, "Internet of Vehicles: Motivation, Layered Architecture, Network Model, Challenges, and Future Aspects," IEEE Access, vol. 4, pp. 5356–5373, 2016.

[182] S. Abdelwahab, B. Hamdaoui, M. Guizani, and A. Rayes, "Enabling Smart Cloud Services Through Remote Sensing: An Internet of Everything Enabler," IEEE Internet of Things Journal, vol. 1, no. 3, pp. 276–288, June 2014.