University College London

Reinforcement Learning Routing Algorithm in Wireless Mesh IoT Networks

By

Yu Liu

A thesis submitted in partial fulfilment of the
requirements for the degree of

PhD

in Electronic and Electrical Engineering

Of

University College London

July 2019

Declaration of Authorship

I, Yu Liu, confirm that the thesis titled ‘Reinforcement Learning Routing Algorithm for Wireless Mesh IoT Networks’, and the work presented in it are my own original work. Where information has been derived from other sources, I confirm that this has been indicated in the work. No part of this dissertation contains material previously submitted to the examiners of this or any other university, or any material previously submitted for any other examination.

Signed:________________________

Date:__________________________

To Asinee and Anya.

Abstract

Internet of Things (IoT) has become a central part of our connected world. Apart from the devices in our home, many IoT devices are located in remote areas supporting all kinds of industrial, agricultural and scientific applications. Networks providing connectivity that cover in the scale of kilometre squares are crucial for these remote deployments. The extensively used star topology is not perfect for the rural environment as the coverage is limited by the placement of the central hub which also contributes to be a single point of failure. Mesh networks are clearly more appealing in this regard, but scalability has always been an issue for mesh networks, especially in terms of routing. Energy provisioning can also be challenging in the remote IoT deployments, as the devices can be left in isolated fields for a long period of time. In this thesis, we addressed the routing problem of mesh-based remote sensor IoT networks by introducing a distributive energy-aware reinforcement learning (RL) based routing algorithm. The proposed algorithm makes routing decisions by holistically considering the energy consumption of the network. This aims to maximise the durability of the entire network while preserving usability. Through the comparisons of simulated results in the failure rate, energy efficiency and carrier band usage rate of the networks supported by the proposed RL algorithm and the other applicable algorithms in the long-range remote IoT networks, we identified the strength of the RL routing algorithm for the remote sensor networks. This thesis also presents a detailed analysis of the RL routing algorithm progressively to demonstrate the effectiveness of the algorithm.

Impact Statement

The routing algorithm this thesis proposed provides possibilities for the deployment of fully meshed sensor networks in extremely remote areas. The networks can rely much less on the service operators provided and are more resilient to the changes of the environment as the introduced AI will adjust the routing of the network accordingly. The simulation results provide evidence and experience when applying the proposed method in other similar situations.

The performed research benefits the academia through the development on reinforcement learning, the appliance of AI can be expanded to more specific applications such as routing in wireless mesh networks. The remote monitoring mesh network itself can also be used in academia for research of other disciplines as the coverage and usability of the network can be expanded with the algorithm proposed in this thesis. With a more reliable and usable IoT sensor network with wider coverage, places that never been able to be studied can be reached. Hence, more studies can be conducted with the usage of the produce of this thesis.

When it comes to benefits outside academia, the energy-aware network can provide longer durability and better coverage outside the control of the network operators. This will reduce the cost for the users and organisations who need to deploy their devices in remote areas. Other than that, with the improved mesh network adaptability discussed in the proposed algorithm, different scales of WSNs can be deployed around our world, providing a better understanding of our living environment. Thus, make our world a little better.

Acknowledgement

This thesis could not be finished without all the great support and assistance I have received during these years. Firstly, I would like to show my great gratitude to my supervisor, Dr Kenneth Tong. He has guided me and supported me through this incredible journey, and his experience and expertise were invaluable to the research and implementation of the thesis.

I would also like to thank all the other academics at the department of Electronic and Electrical Engineering of UCL that have helped me and guided me during the course of the PhD. I would like to particularly thank Professor Kai-Kit Wong for his input and suggestions on the design and implementation of the algorithm and the simulation.

I would also like to acknowledge all my colleges in the department for their help and assistance during these four and a half years. It would be impossible to accomplish this journey.

Additionally, I would like to thank my family and friends who have supported me all these years. I could not become who I am without their help.

Last but not least, a special thanks to my wife Asinee and our daughter Anya. Thank you for being my inspiration for my work and my life.

List of Abbreviations

ACI	Adjacent-Channel Interference
AI	Artificial Intelligence
AODV	Ad hoc On-Demand Distance Vector
AS	Autonomous System
BATMAN	The Better Approach To Mobile Ad hoc Networking
BW	Bandwidth
CAR	Context-aware Adaptive Routing
CBUR	Carrier Band Usage Rate
CC	Charging Cycle
CCI	Co-Channel Interference
CCS	Chirp Spread Spectrum
CR	Coding Rate
CSPF	Centralised Shortest Path First
DN	Destination Node
DREAM	Distance Routing Effect Algorithm for Mobility
DSDV	Destination-Sequenced Distance Vector
DSR	Dynamic Source Routing Protocol
EPC	Evolved Packet Core
HWMP	Hybrid Wireless Mesh Protocol
IN	Intermediate Node
IoT	Internet of Things
IP	Internet Protocol
IPI	Inter-Packet Interval
ISM	Industrial, Scientific and Medical bands
LAN	Local Area Network
LPWAN	Low Power Wide-Area Networks
LTE	Long-Term Evolution
LTE-M	Long Term Evolution for Machines

M2M	Machine-to-Machine
MAC	Media Access Control
MDP	Markov Decision Process
ML	Machine Learning
MPR	Multipoint Relay
MR	Maximum Communication Range
MRP	Multiple Relay Points
NB-IoT	Narrowband IoT
NN	Next Node
NR	Number of Retry
OLSR	Optimized Link State Routing Protocol
OSI	Open Systems Interconnection
OSPF	Open Shortest Path First
PDR	Packet Delivery Rate
PQ	Path Quality
QoS	Quality of Service
RL	Reinforcement Learning
RM	Routing Metric
RSSI	Received Signal Strength Indication
SB	Success Bonus
SBV	Success Bonus Value
SF	Spreading Factors
SL	Supervised Learning
SN	Source Node
SINR	Signal to Interference and Noise Ratio
SNR	Signal to Noise Ratio
TC	Topology Control
TD	Temporal Difference
TDS	Total Dissolved Solids
TORA	Temporally Ordered Routing Algorithm
TQ	Transmission Quality

USL	Unsupervised Learning
VBF	Vector-Based-Forwarding
WMN	Wireless Mesh Network
WMSN	Wireless Mesh Sensor Network
WPAN	Wireless Personal Area Network
WRP	Wireless Routing Protocol
WSN	Wireless Sensor Network

Table of Contents

Chapter 1 Introduction

The first electromagnetic telegraph sent by Francis Ronalds in 1816 [1] marked a new era of communication using an invisible electromagnetic field instead of visible objects. By using the electronic pulses to convey messages, information can be passed on for a long distance in a short time. The adaptation of electromagnetic signals has removed the physical barriers that had limited humans from effectively performing remote area monitoring and communication. In the age of digitization, ALOHAnet, started in the 1970s, was the pioneer wireless network [2]. Over nearly 50 years after that, many wireless technologies have been proposed and implemented, such as the 802.11 WLAN protocols and mobile networks. Each of these technologies is designed to serve a certain purpose and there has not been any universal wireless solution so far, especially for the network that enables remote monitoring without human involvement. The concept of the Internet of Things (IoT) has been started since John Romkey developed the Internet-connected toaster in 1989 which could be switched on and off remotely using the Internet [3], as well as the famous remote monitoring IoT experiment was the Trojan Room coffee pot in the University of Cambridge [4]. Then the term IoT was put forward later by Kevin Ashton in 1999 [5].

Soon after that, widely deployed IoT devices have deeply changed how the industry of manufacturing, agricultural and numerous other applications, particularly those covering a wide remote area, work. Machine-to-Machine (M2M) communications and long-range wireless technologies enabled automatic data collection to data utilisation that is truly revolutionising the world. According to Business Insider Intelligence [6], there will be more than 40 billion IoT devices connecting to the Internet by 2023. The number will only drastically grow in the future as the demand accelerates and the industry progresses.

The type of IoT network discussed in this thesis is wireless sensor networks (WSNs). By employing different kinds of sensors, WSNs that can monitor the environment have been developed and used in different applications for many years [7]. The sensors collect information about the physical quantity of certain objects as well as events that happen in the environment. WSNs usually consist of nodes that have one or more sensors that can measure different environmental conditions like ambient temperature, movement, etc. The nodes in a WSN also have the other components particularly the radio transceiver with access to the antenna, internally or externally, the microprocessor and the energy source such as a battery, external power supply or energy harvesting device [8].

To connect the nodes together and to the Internet for data gathering and processing, WSN network technologies vary drastically according to the purpose and scale it serves. One of the most common technologies in this area is short-range wireless solutions, such as Wi-Fi, Bluetooth and Zigbee. These short-range technologies are great to provide connectivity to many IoT applications in urban areas. Wi-Fi has been widely used in indoor IoT applications as the coverage of WLAN can provide fast and stable connectivity for the devices that require considerable throughput and security [9]. One such application is security cameras for homes and offices. By using Wi-Fi as the connectivity solution, devices can have a secure, stable, and fast connection to the Internet or the intranet of the deployment [10]. Bluetooth has been used for peripheral devices for a long time because of its low power consumption [11]. ZigBee has been used for home automation IoT networks [12] by many established manufacturers such as Philips and Hive.

An alternative connectivity solution for IoT applications is the Low Power Wide-Area Networks (LPWANs). LPWANs are designed for long-range IoT communication scenarios from the ground up [13], which means they are positioned between the coverage of mobile and short-range wireless networks. They are suitable for M2M communications rather than directly interfacing with users [14]. LPWANs make the trade-off of data rate for lower power consumption and longer communication range. The GSMA, one of the driving forces of LPWANs, has launched the 'Mobile IoT Initiative' in June 2015 to promote LPWAN commercial solutions using licensed spectrum. Two major licensed-spectrum based LPWAN technologies supported by the GSMA are Narrowband IoT (NB-IoT) [15] and Long Term Evolution for Machines (LTE-M, also known as eMTC) [16]. Both technologies are based on the existing infrastructures owned by the mobile operators. The difference between NB-IoT and LTE-M networks is the former uses narrow maximum baseband for ultra-low complexity devices while the latter utilises wider bandwidth to provide higher data rates (up to 1 Mbps), lower latency and better location services [17]. The services of these networks are subscription-based which incurs a fee for using the networks, thus increasing the operational cost.

Other LPWAN technologies do not use the licensed spectrum to cover wide areas, one of such networks is LoRa [18]. By implementing chirp spread spectrum (CSS) modulation, LoRa trades data rate for sensitivity within a fixed channel bandwidth. The exceptional sensitivity enables LoRa devices to provide a long transmission range, with a stated range of more than 10 km using maximum spread factor 12 that has an indicated physically transmitting bit rate of 250 bps. It even reached an extreme 702 km point-to-point transmission in an experiment [19] as well as 3 km coverage in a field test in the dense suburban area [20]. Since LoRa uses license-free sub-gigahertz ISM bands, it also offers subscription-free services which can significantly reduce operation cost of IoT networks.

Sigfox is another prominent LPWAN solution using the sub-GHz ISM bands. Unlike LoRa's choice of spread spectrum approach, Sigfox uses Ultra Narrow Band modulation to exchange messages between the node and the base station. The bandwidth used to convey Sigfox messages is 100 Hz and have a data rate of 100 or 600 bps [21]. Using this technology, the Sigfox network is able to reach up to 40km in rural areas [22]. The topology of Sigfox is a one-hop star, and the network requires its operator to build the infrastructure (base stations) and the network is operated using a paid subscription model.

1.1 The Remote Monitoring Network Environment

In the remote areas described in this thesis, using the mobile network is considered as a challenge even in developed countries like the UK. Figure 1.1 [23] shows the mobile data network coverage offered by the only NB-IoT national provider in the UK, Vodafone [24]. It can be observed that there are large areas in the north of Scotland as well as some part of northern England showing no coverage. Another major mobile network operator, EE, also has limited 4G/5G coverages in those areas, as shown in Figure 1.2 [25].

In developing countries, remote areas are even less covered. During the field trip of testing the IoT networks [26] in BaZheXiang area, Sichuan in west China shown in Figure 1.3 in 2017, we found the coverage of the mobile network at the test site and villages around was very limited and unstable.

Figure 1.1 Vodafone mobile service availability in the UK [23].

Figure 1.2 EE mobile service availability in the UK [25].

Figure 1.3 Field test site in China [26].

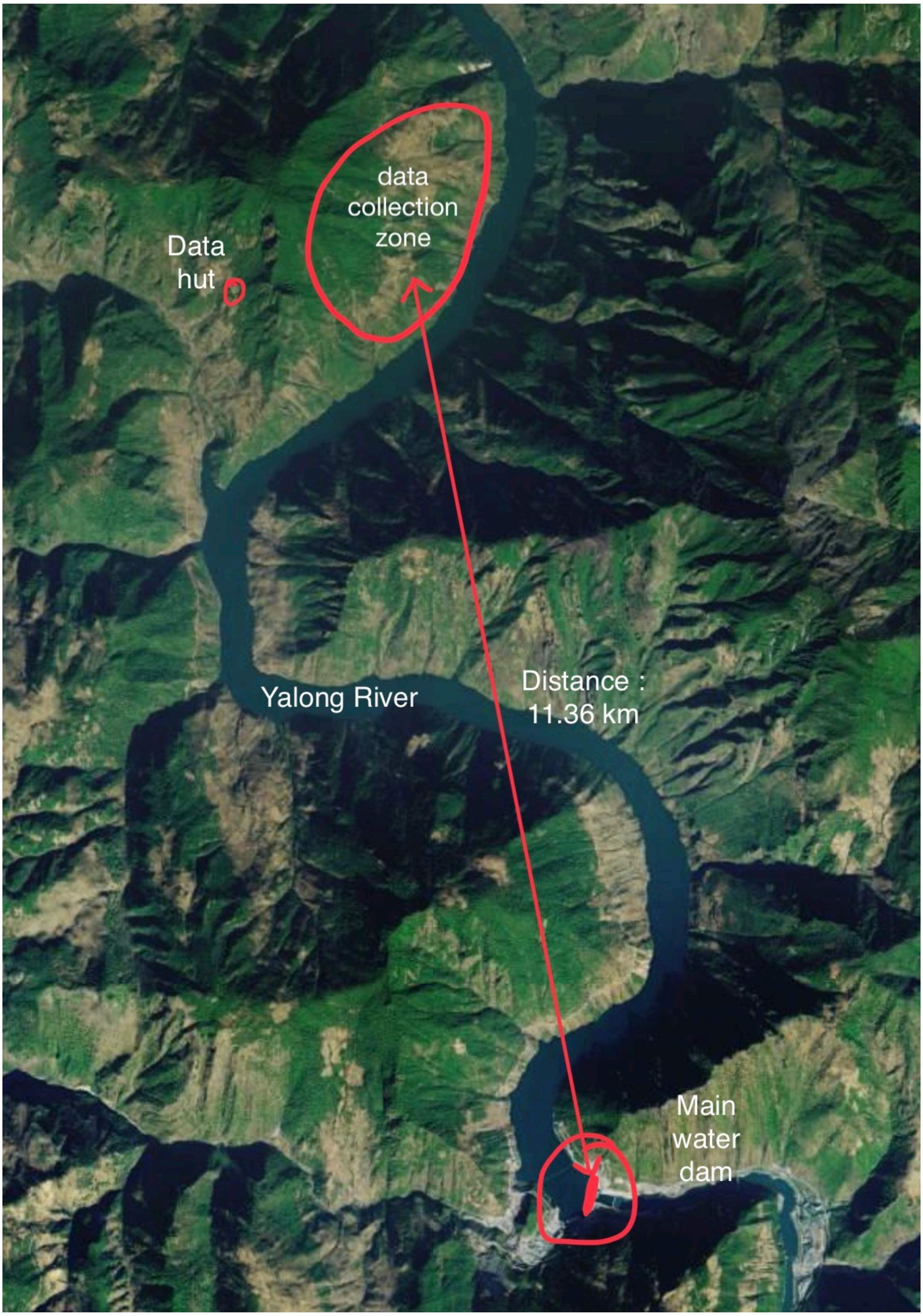


Figure 1.4 Terrain of monitoring sites during the field trip in China [26].

Beyond the coverage of the network, there are other critical challenges of the network that need to be addressed. Here listed four important ones in the order of importance that we keep our focus in this thesis:

- Firstly, power efficiency is one of the most important elements of remote monitoring networks. The site where the remote monitoring sensors deployed usually has no power supply from the grid which means the devices are sorely battery powered from the network perspective. Additionally, due to the remote geographically isolated location, the sensor nodes to be discussed in this thesis are usually left unattended and cannot be maintained regularly.
- Secondly, availability is crucial for all remote monitoring networks. The collected data from the sensors need to be wirelessly transmitted as other wired means, such as coaxial cables or optical fibres are considered expensive and labour intensive Besides, being an M2M communication network, the nodes may also need to be remotely configured or manipulated for the maintenance of the network.
- Thirdly, scalability is also important for such networks. As the coverage of the network grows, more devices can be deployed in the area for better monitoring. The network ought to be able to handle the change of the network configuration when new nodes are attached to the network as well as nodes with no battery are removed from the network at any time. This change of scale of the network happens all the time and the network will need the scalability to handle it.
- Finally, throughput and delay are sometimes important in the remote monitoring networks, since the purpose of such networks is monitoring certain areas, sometime the regular access and quality of service may be requisite, particularly in some of the nature disaster prevention and emergency applications. The data generated from the sensors can be stored and sent to the server regularly. The size of the data is also significantly smaller than other applications over the Internet, such as video conferencing. This characteristic of the network can ease the requirement for throughput and delay of the network.

During the field trip in China [26], we found the challenging geographical terrain for wireless communications in remote monitoring networks requires the technology to have a long communication range and deep penetration capability. The sheer distances between possible locations of monitoring sites where the sensor nodes are located impose such a challenge. As shown in Figure 1.4, the point-to-point distance between the monitoring site (data collection zone) to the data centre (the main water dam) is about 11.36 km. The distance between other monitoring sites can be even farther than this as the field test is for the transmission from the first data collection zone back to the main water dam. In other long-range environment monitoring networks in the remote areas, such as water monitoring [27] [28], and rural air monitoring [29]. The distances between monitoring nodes can be similar or even greater. This requirement of long range between nodes, in addition to the limited mobile network coverage and power grid supply, makes the automation of remote monitoring more challenging.

The climate of the deployment area is also crucial when the nodes in the network are battery powered and the only way to recharge the batteries is solar power. The sunlight hours of the test site will directly impact the charging cycles of the nodes, because of the more hours of sunlight per day, the shorter the battery recharging cycle. The national weather service of the UK, the Met Office, has the details of the number of hours of sunlight gathered from the weather stations across the country [30]. We have collected the monthly average sunlight hours across the UK from 1981 to 2010, as shown in Table 1.1:

Table 1.1 Average sunlight hours in the UK from 1981 to 2010

Period of Time	Average sunlight (hrs)	Average daily sunlight (hrs)
Annual	1372.8	3.76
Longest (May)	185.9	6.00
Shortest (December)	40.8	1.32

1.2 Limitations of other forms of wireless network

1.2.1 Short-range wireless networks

Short-range wireless networks are effective for what the name suggests, short-range communications in IoT networks. The coverage of these networks is very limited due to the focus on short range applications. Bluetooth, for example, uses frequencies in the 2.4 GHz ISM band and utilises frequency hopping in a predetermined order at regular intervals to highly resistant to narrow-band interference. This allows Bluetooth devices to have a maximum transmission range of 100m when using an output power of 20 dBm [31]. Zigbee has the ability to conduct cross-band communication across both 2.4GHz and sub-GHz bands. This allows the maximum transmission range up to 1km when the network operates in the sub-GHz bands. These technologies are more suitable for monitoring in urban areas when the population of the nodes is much denser and internet gateways can be easily found nearby.

People use boosted long range Wi-Fi repeaters for long distance communications for high bandwidth requirements. They can be used to connect widespread physical sites to the same network or fill in mobile network dead zones in remote areas. Lukac, et al. have used 23 dBm 2.4 GHz Wi-Fi radio, a 3m mast with a 15dBi gain YAGI antenna to reach 15-20 km communication range at ground level [32]. However, this setup is not feasible for the remote monitoring nodes in the context of this thesis as the 3m mast with 15dBi YAGI antenna can hardly be deployed in the remote areas for its size as well as a 23 dBm 2.4 GHz Wi-Fi radio will consume too much energy in the battery operated off-grid sites. Commercial Wi-Fi long range boosters have also been used to provide a line of sight long range transmission using directional antennas [33]. This type of solution is even harder to be deployed in remote monitoring sites as it not only requires line-of-sight connection with the uplink gateway but also needs to elevate antenna which is difficult when the monitoring site is in a valley or below the gateway.

Short-range network based IoT network solutions are more suitable for more densely populated areas where it has the power and infrastructure deployed beforehand. They are also great for higher data rate monitoring tasks. But in the context of this thesis, they are not the best choice.

1.2.2 Cellular mobile networks

Traditional mobile networks and the GSMA promoted licensed spectrum LPWANs can also be deployed for remote monitoring networks. These networks use the licensed spectrum to provide services, centred around ‘cells’ from base stations and other infrastructure provided by the network operators. As shown in Figure 1.1 and Figure 1.2, mobile networks have covered most of the area around the UK. By using the licenced frequency spectrum, the mobile networks have a much better channel performance when compared to technologies using the ISM bands as it is maintained by the operator [34]. This enables the network to have a better capacity, uses less power, and have a larger coverage area as base stations can be added and frequencies can be reused [35].

However, in the context of the thesis, remote areas that need monitoring are usually not inhabited. This means it is not cost efficient to provide data network services to these areas with traditional mobile (GSM/3G/4G/5G) networks, as the costs of building and maintaining the new base stations are high [36]. As shown in Figure 1.1 and Figure 1.2, some of the remote areas in the UK are not yet covered with 4G networks even after 7 years the first 4G network was commercially launched [37].

NB-IoT and LTE-M are designed to provide LPWAN services using licensed bands. NB-IoT is designed to enable support for IoT devices using new physical layer signals and channels. Commercial NB-IoT network has already been deployed nationally in the UK by Vodafone [24]. Adhikary, et al. evaluated that NB-IoT networks can provide the same target SNR with an extra 20 dB tolerance of maximum coupling loss over the existing LTE networks. This translates to a coverage enhancement of 20 dB for the network while co-existing with the LTE networks in the covered areas [38]. However, as NB-IoT is a subscription-based service, the cost of using the network can be significant when the network scales up and the number of sensor nodes increases. Additionally, NB-IoT and LTE-M share the same base stations of the existing mobile network, the coverage limitation still applies to both technologies. At the moment, LTE-M networks are not yet commercially available in the UK, it will not be discussed in detail here. Hence, extending remote monitoring and sensing networks in rural areas, NB-IoT and LTE-M are not the best solutions. Besides, both NB-IoT and LET-M support the mobility of the IoT devices, which is not applicable in the context of this thesis.

1.2.3 License exempt (ISM) band LPWANs

By employing sub-GHz ISM frequency bands (863–870 MHz in Europe), LoRa and Sigfox provide long range IoT services with low power consumption. Sigfox has a paid subscription business model [39] like the mobile network services, whereas LoRa provides a free to use radio technology with the freedom to establish own networks [18]. However, both networks use the star topology at its core, using a gateway to connect all the end nodes. This causes the same problem as the other forms of networks. In order to expand the network coverage, the gateway locations and the establishment of connections of the existing network are crucial. In the case of Sigfox, the coverage is all managed by the company itself. It becomes very difficult to predict if the remote sensing areas are covered by the network or not, as it is determined by the company. Whereas the coverage of LoRa networks can be established by the users, the gateways of the network must be connected to the Internet by either wired or other kinds of wireless networks, which also can be challenging in the place of remote areas.

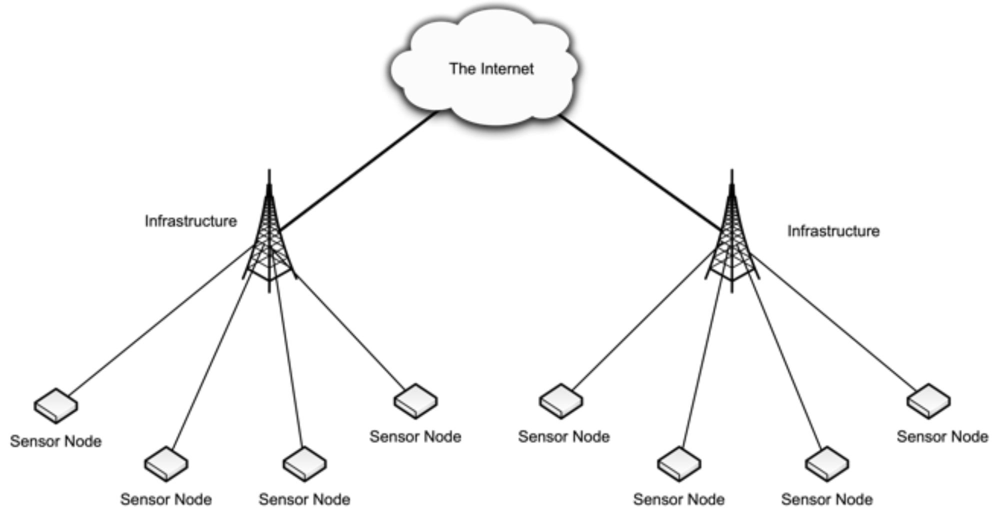
In all of these network technologies discussed in this section, the star topology is at the core. Challenges such as location and deployment of the centre hub of the network become pivotal in connecting all the remote sensor nodes. To accommodate the need for remote monitoring networks, the alternative mesh topology should be power efficient, widely available and easily scaled.

1.3 The need for the mesh topology

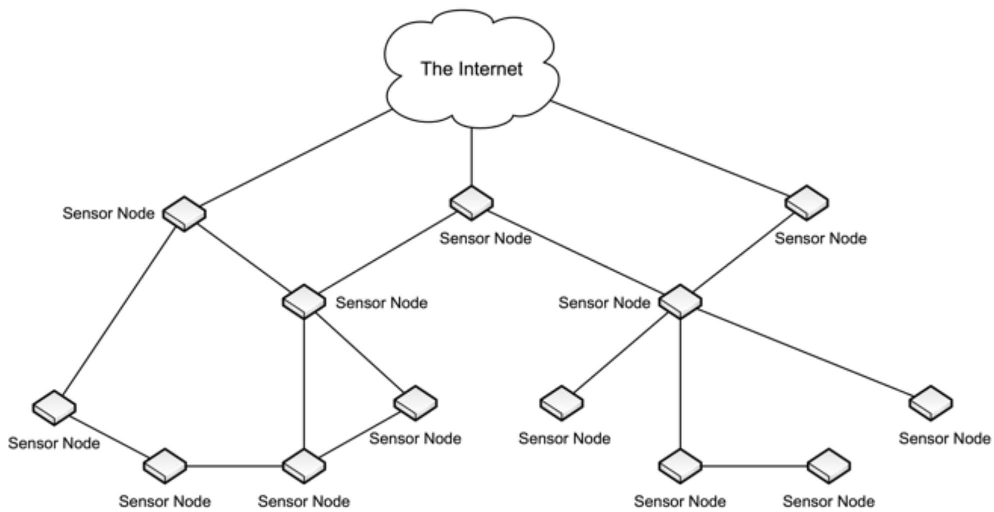
From the book of Wireless Mesh Networks [40] by I. Akyildiz, X. Wang, a wireless mesh network (WMN) is defined as a wireless communications network made up of radio nodes organised in a mesh topology instead of the star topology used in most of the networks. The nodes inside a WMN are connected to each other directly, unlike in the star networks, where all the devices are connected to a central hub. Interconnections between nodes inside a WMN are effective wireless ad hoc networks. Consequently, any wireless technology that supports multi-ad hoc connectivity can be used to establish a WMN.

WMNs are being used in many applications to provide connectivity services. The IEEE has established 802.15.4/802.15.5 working groups for Low-Rate WPAN and WPAN based mesh network as well as the 802.11s standard for Wi-Fi based mesh networks.

The difference between star topology and mesh topology networks in the remote monitoring scenario are displayed in Figure 1.5. Infrastructure is essential in the star networks, which has the disadvantages described in Section 1.3. Whilst in the mesh networks, the connectivity to the cloud can be provided by more than one sensor node in the network, and the interconnectivity can send all the data through those sensor nodes acting as a gateway.



*a. A s*tar topology network



b. A mesh topology network

Figure 1.5 Topologies of remote monitoring IoT networks

This decentralised and distributed nature of the WMN networks makes it very appealing for remote monitoring networks. The elimination of the centre point of failure also improves the availability of the network. The adaptability of nodes being gateways enables the self-forming nature of the network which works best in the scenarios which the uplink to the Internet is limited like the remote monitoring and sensing networks.

People have discussed employing mesh topology in wireless sensor networks to form wireless mesh sensor networks (WMSNs) for some years. But there are yet any established standards or workshops to establish one. Rodenas-Herraiz, et al. have reviewed a series of WMSN proposals, noticeably using the IEEE 802.15.5 Wireless Personal Area Network (WPAN) mesh network standards [41]. They have concluded that the WMSN proposal using IEEE 802.15.5 standard using the low-rate technologies using IEEE 802.15.4 specification stood out from a series of other WMSN solutions. However, this proposal relied on low-rate, short range networks, due to the personal network nature from the 802.15 WPAN working group. This rendered the proposal only working in short-range WMSN scenarios. The WMSN using LPWAN technology with multi-ad hoc connectivity support such as LoRa enables WMSN with a larger coverage and better efficiency. Additionally, the capability of self-organisation and self-configuration of the network can be used to support incremental deployment of sensing nodes inside a WMSN. However, the modelling of the long-range WMSNs remains to be done.

Another issue of the limitation of establishing WMSNs is its scalability. WMNs are often considered as non-scalable because they are likely to be suffered from interferences as well as suffering from changes in the network known as mobility [42] [43]. Nevertheless, this is not the case for applications discussed in this thesis. In the remote sensing and monitoring scenarios, such as soil quality monitoring networks in the farms and air quality monitoring networks in the urban areas, the monitoring sensors are mostly immobile in the network. Therefore, the topology of the network is relatively static with the only change in the network being the addition of nodes from an extra deployment or recovering from power drought as well as reduction due to battery power drought or sensors failure. The scalability problem of the scenario has no mobility involved. Therefore, with proper routing and network self-reestablishment, the WMN based remote monitoring network can be scalable. Adding nodes from the network can be considered attaching another network onto the existing network where the interconnecting nodes are gateways between the two networks. As there is little mobility occurred in the WMSN, once the connection has been established, the scale of the network has been expanded.

1.4 Routing in WMSNs

Routing in the star networks, such as the Ethernet, is done by using Short Path First algorithms. The Internet Protocol (IP) uses Open Shortest Path First (OSPF) to route its single autonomous system (AS). The OSPF works inside an autonomous system such as a local area network (LAN) to route the data packet to the right gateway. The OSPF implements Dijkstra's algorithm to calculate the shortest route to a destination through the network based on the cost of the route. It calculates the cost of the route based on weighted link-cost parameters, i.e. bandwidth, delay, and load. The weights are configured by the network administrators. [44] OSPF can also detect topology alterations caused by link failures and recalculate a new loop-free route in a short time [45]. However, it is designed to target the routes in a hierarchical network such as LAN.

When it comes to WMNs, OSPF is not applicable. Without the hierarchical in the topology, the shortest path cannot be easily calculated without centralised storage of the topology. Proactive link-state routing protocols such as Optimized Link State Routing Protocol (OLSR) are more suitable for decentralised mesh networks [46]. OLSR uses HELLO and topology control (TC) messages to discover and propagate the link-state of every node in the mesh network. The routing is done by calculating the shortest hop forwarding path between nodes using the link-state information on each node. However, the scalability of the OLSR is an issue as with each new node added into the network, more hello and TC messages need to be sent in the network to keep the link state information on each node updated. Furthermore, OLSR does not consider link quality when choosing a route. In the case of the remote monitoring networks, OLSR has no support for energy awareness, which is highly important. Additionally, as a proactive routing protocol, OLSR requires the usage of battery and computational power to send

Hello messages in order to propagate routing information. This further excludes OLSR from IoT mesh networks, as devices are usually kept sleeping while idling to conserve energy.

To perform routing for remote monitoring networks, the algorithm needs to be energy aware. This awareness is the key to finding an energy efficient route that not only benefits the sending and receiving nodes but also keeps the entire network more usable. It also needs to be dynamically updated and optimised when the packets are transmitting in the network. Lastly, it should be scaled with the addition and reduction of the nodes to support the scenario we describe in the earlier part of this section.

With all the requirement of the routing problem in WMSNs, an algorithm that has extra intelligence capability is a beneficiary, especially when considering the dynamic consumption model of different monitoring nodes. The introduction of using artificial intelligence to route the WMN IoT only comes naturally.

1.5 Machine Learning and Reinforcement Learning

Artificial Intelligence (AI) refers to the “cognitive” functions demonstrated by machines through learning from outside inputs instead of being programmed beforehand in the context of problem-solving. AI has been discussed and researched from the Dartmouth Workshop in 1956. Computer scientists demonstrated a computer that learns checkers strategies, solving word problems in algebra, proving logical theorems and speaking English [47]. However, the limitation of the computing power at the time slowed the progress down and led into an ‘AI winter’ era [48]. Not until the late 1990s and 21st century, AI has re-emerged as a useful tool in multiple areas including logistics, data mining, medical diagnosis and other areas [49]. The great evolution of computational power with Moore's law enables the AI algorithms to work much better. The demand for solving specific problems in different fields, i.e. statistics, economics and mathematics, using such technologies also greatly increased during that period of time. The possibility of application of AI greatly expanded since. Recently, AI has become a field with numerous applications.

As discussed in Section 1.4, AI can play an important role when deciding the optimum route of the packet. With the correct route selected, less stress will be imposed on the battery of the nodes that has a heavier duty as a sensor and a packet relaying node. The algorithm should be able to balance power consumption and network performance based on the need for the network.

Machine learning (ML) is an important research subset of AI. Without using pre-programmed explicit instructions, ML uses patterns and inference from the information input of the algorithm, producing statistical mathematically models and results that computers can use to perform similar tasks after that effectively [50]. The generalisation from experience or data is a core objective of a learner algorithm. This means that the learner algorithm will be able to handle new, unseen examples/tasks accurately based on the experience or data it gathered earlier during the learning period with certain training data set. The data used to train ML algorithms can be obtained from different sources, such as pre-defined training data sets, random related data sets or even previously used data can be re-used to retrain the AI for better accuracy and efficiency.

There are three main different approaches in the field of ML: supervised learning (SL), unsupervised learning (USL) and reinforcement learning (RL). Each one of the approaches has its unique requirement on input and output as well as the type of problem the algorithm is applied to solve.

The first method of the ML approach is SL. SL uses example input-output pairs to train the model to perform as expected [51]. The training data set contains examples input as well as output. Each example consists of the desired output value associated with a certain set of input data. The desired outputs are labelled for the algorithm to find the link between it and the input. Hence the name of the approach is called supervised learning. The SL algorithm produces an inferred function that maps the input and output data from the training data set. This function will then be used against unseen input of the same problem, and ideally, the function can produce a result output as expected.

Unlike SL, USL does not rely on labelled input/output pairs. It uses cluster analysis to discover commonalities among training data [52] The method will store the commonalities and act on oncoming new data based on the presence or absence of such commonalities to determine the action after receiving the data.

Neural networks are usually used in USL to provide artificial neurons that can be fired together using Hebbian theory to change the weighting of the connection. The change of the weighting will determine the action that will be taken by the algorithm. Since the data has not been labelled, the method is referred to as unsupervised learning because of the lack of human intervention of the training data.

Finally, people use RL to help make better decisions. Using dynamic programming, RL is about finding a balance between exploration and exploitation [53]. This means that the RL algorithms are both utilising the learnt experience from the past and the available options to find an optimal decision for certain problems. The reinforcement problem is usually modelled as a Markov decision process (MDP). MDP transforms stochastic decision-making situations involving random outcomes into a mathematical model that can be optimised to find the best solution [54]. RL can also use the Boltzmann exploration process to balance exploration and exploitation to add dynamic to the algorithm to avoid ignorance [55]. In RL, the algorithm is looking for the best outcome of each decision made. Each iteration of the running of the algorithm will contribute to that goal by providing appropriate feedback.

When considering the routing efficiency of the packet through the network, each node will need to decide of which the next node. This decision will not only affect the transmission of the current packet, but also the subsequent future packets as the nodes along the route will need to consume energy to transmit it. The entire routing decision in a mesh network can be seen as a collection of many small decisions that in the long term affect the usability and energy efficiency of the entire network. This decision-making process matches well with the RL model. Hence, by applying suitable RL into the routing algorithm will enable the nodes to learn from their experience and make a better decision in the long term. This utilises intelligence into the entire routing process and the operation of the network.

However, the RL algorithm has its own limitations. Firstly, RL takes time and effort before the decision-making functions. Without previously learnt knowledge, the node can also make random guesses and expect the feedback from that experience to make a difference in a short period of time. Therefore, when there is a large number of nodes in the mesh network, it will take a longer time and consume more energy of each node in the network before the network becomes energy aware, because of a lot of random choices have to be made before the sensible ones emerge. Nevertheless, the remote monitoring networks are not designed to work as a huge number of sensors in a confined area and are deployed for a long period of operation. This will eliminate the scalability concern of employing the RL algorithm into the network. Additionally, the added complexity and overhead to the nodes are also a concern of using RL for routing. But in the context of this thesis, the simple calculation to update the routing table will not bring about excessive overhead as the simplicity of the network ruled out the concern of the complication.

1.6 Energy Awareness in the Networks

Energy awareness of the network in this thesis means that the network has the ability to use the energy-related information gathered from nodes during the time of operation. This information typically includes the energy levels of each individual node, the energy consumption of the entire network, and the potential energy expenditure of selecting a certain route. Being a distributed network, energy awareness of the network suggests that the nodes can make routing decisions based on the energy consumption at their own knowledge. Each decision will add up to result in the usability of the entire network in the long term. This is very important in the context of this thesis, as it targets the remote networks that usually have a limited source of energy. These networks are deployed in rural areas where mostly are off-grid. Being energy aware means that the network is able to make energy efficient routing decisions.

The introduction RL enables remote monitoring networks to be energy-aware using AI. Over the course of the deployment and usage of the network, the AI will gradually store energy information about the network and optimise subsequent operations with better energy efficiency. Unlike SL and UL, AI created by RL learns the environment with experience instead of a large training set. The deployments of each network are unique to each case, providing an effective training set to fit all deployments is nearly impossible. Without the training, other

AI algorithms to predict and response to the changes in the network precisely is difficult. RL, with its balanced exploration and exploitation decision algorithm, can mitigate this problem, especially when the experience it gathers is sufficient over time. In the proposed algorithm, the routing table of each node is populated and updated whenever a transmission is happening to that node. This is performed by using a model-free RL algorithm based on temporal difference (TD) learning. The updated routing tables are able to produce better routes under the updated conditions.

1.7 Research Contribution

Wireless IoT networks and made using sensors to monitor remote areas possible. This capability can greatly benefit various disciplines of scientific areas from agriculture to geography. However, the current wireless network technologies used in IoT networks, such as cellular networks and WLAN networks, have constrained various potential applications being deployed. The coverage offered by these networks are usually limited to the deployment of the service providers, and in many cases, remote areas may not be in their best interest to be covered. In addition, the commonly utilised star topology in most existing wireless networks can also lead to reliability concerns when deploying the sensors to the rural areas. Maintaining the operation of the network is difficult because of the existence of a single point of failure in the network.

To resolve this research problem and create a more efficient wireless IoT network, we proposed a reinforcement learning routing algorithm in wireless mesh IoT networks. We first introduced mesh topology into wireless networks in the scenario of monitoring rural areas using IoT networks to provide better coverage. Instead of focusing on the throughput and delay in the network, we then tackled the energy efficiency quandary of the network by utilising reinforcement learning in the routing algorithm. Hence, the durability and usability of the network are greatly improved over traditional wireless mesh networks, as shown in the simulations conducted in Chapter 4 of this thesis.

The expansion of the coverage of WLAN networks and improvement of throughput are the main studied areas in most current studies on wireless mesh networks. However, the discussions of IoT mesh networks are relatively rare in the field, especially the work on the usability of the network. Because of the need for multiple relays when delivering a packet, the total energy consumption is higher in the WMNs than the traditional star networks. Therefore, by improving the energy efficiency of the network, the general durability and usability of the network can be considerably enhanced over a longer period of time. It also magnifies the benefit from this enhancement in the case of applications in remote wireless sensing and monitoring. By using RL instead of feedback loops like traditional routing algorithms, the algorithm is able to utilise the experience of the entire working life of the network compared to the short-term response from the feedback from the feedback loop that are commonly used in other routing algorithms. By taking the energy status of the entire network into account, the RL routing algorithm has shown improvement in all metrics including failure rates, energy efficiency and carrier band usage rate (CBUR) that we used in the simulations over the benchmark algorithm.

This improvement has proven the research contribution of this thesis in the field of remote sensing networks. The thesis presents how energy awareness can play a crucial part when deploying remote sensor networks and has pointed out a new method of energy aware routing for such IoT networks.

1.8 Research Publications

This proposed RL routing algorithm has produced the following research publications:

- Y. Liu, K.-F. Tong and K.-K. Wong, “Reinforcement learning based routing for energy sensitive wireless mesh IoT networks,” IET Electronics Letters, 2019.

- Y. Liu, K.-F. Tong, X. Qiu and Y. Liu, “Wireless Mesh Networks in IoT Networks,” in *Conference: 2017 International Workshop on Electromagnetics: Applications and Student Innovation Competition (iWEM)* , London, 2017.

- Y. Shen, L. Cai, Y. Liu, X. Ding, X. Qiu, Y. Liu and K.-F. Tong, “On the Antenna for Long Range Low Power Geographical Monitoring IoT Network,” in *European Conference on Antennas and Propagation*, Nagoya, 2018

1.9 Thesis Overview

The organisation of this thesis is as follows. After the introduction of this chapter, Chapter 2 reviewed other research works on WMNs, WSNs, AI and RL, as well as routing in WMNs. While in Chapter 3, we modelled the remote sensing network that the RL routing algorithm is targeted at. This model of the network included the wireless channel, the nodes, the links between the nodes, the transmissions and the feedback during the operation of the network. This model set the scene for the reinforcement learning to learn from which further discussed in Chapter 4. We also introduced the temporal difference learning, which this RL algorithm is based on, as well as some key parameters presented in the algorithm in the same chapter. A flowchart that described the routing decision process is also provided in this chapter for a better understanding of the process.

In Chapter 5, we presented the method we used in the simulations that evaluated the performance of the network. We assessed both an upper-bound and lower-bound method that are used in the simulation.

In Chapter 6, the thesis presented the comprehensive results of a series of simulations to examine different parameters related to RL algorithm and the environment of the network. By comparing the results of the RL routing algorithm against two benchmark algorithms, it established the effectiveness of the algorithm in different environments. It also showed how different parameters affect the performance of the algorithm. We concluded the chapter with a set of simulations with the optimal parameters gained from previous sections, to demonstrate the advantages of employing the RL routing in WMSNs.

Finally, Chapter 7 showed the conclusion drew from the building, testing and result of the algorithm. It also defined the possible future development on the algorithm after the completion of this thesis.

Chapter 2 Literature Review

While there have been many existing works in the field of WMN and WSN over the years, the routing algorithms which consider the energy constraint in remote monitoring sensor network are rare. The introduction of machine learning (ML) into routing algorithms is also not a new approach, however, few have set their focus on wireless mesh sensor networks (WMSN) networks. In this chapter, we reviewed the related works in 1. wireless mesh network (WMN), 2. wireless sensor network (WSN), 3. Machine learning (ML) and 4. routing solutions for WMSNs, to highlight the gap in the literature where the reinforcement learning routing algorithm aims to fill.

2.1 Wireless Mesh Networks

2.1.1 Wireless Mesh network standards

The IEEE 802 Standards Committee is responsible for networking standards development and maintenance [56]. Two major IEEE standardised WMN types are 802.11s Wi-Fi mesh networks and 802.15.5 Wireless Personal Area Network (WPAN) mesh networks. The former one based on the Wi-Fi standard, using ad hoc Wi-Fi as the connection between mesh stations. The 802.11s networks are MAC-based multi-hop solutions, hence it operates transparently to any higher-layer protocols [57]. However, as discussed in Section 1.2.1, using the short-range technology as the underlying backbone is not suitable for the networks with the constraint on energy, therefore mesh network using 802.11s standard is not considered the best suit for this thesis. On the other hand, 802.15.5 networks are based on the low power personal network technology 802.11.4, which has a limited single hop range, e.g. Zigbee has a range of 50-100 feet between devices [58]. This limitation on the single hop range constraints the application of these technologies in the context of the remote monitoring environment of this thesis, as described in Section 1.1 [41]. Therefore, a new WMN solution is specifically developed to support remote monitoring mesh networks.

2.1.2 Works on Wireless Mesh Networks

In 2004, Akyildiz et al. [40] researched different system architectures including Infrastructure/Backbone WMNs, Client WMNs and Hybrid WMNs. It was also summarised the characteristics of these WMN architectures as listed below:

1. Multi-hop wireless network: It can facilitate higher throughput in the same distance using multiple hops with shorter links, reducing interference and have better frequency reuse.
2. Self-forming, self-healing, and self-organization: The network can be built gradually and maintained easily, reducing the cost.
3. Mobility: It depends on the status node. Mesh networks can support both stationary and mobile nodes.
4. Multiple network access: The access to the internet from the nodes can be done using different connections.
5. Multiple power consumption constraints: The power constraints of the network depending on the type of application, the routing protocol needs to meet such constraints.
6. Compatibility and interoperability: The network needs to be compatible and interoperable with other networks to provide services.

It then listed some potential commercial applications of WMNs, such as broadband home networking, community and enterprise networking, building automation and health and security monitoring systems. However, it didn’t include WMNs in the WSNs forming the WMSNs. This is possible because when the paper was published WSN networks were not largely available due to the limitation of the long-range low power radio technology at the time. But one of the most important facts this paper pointed out is that WMNs have the ability of adapting and inter-operate with different network technologies. The paper has set the scene for further study of WMNs and inspired this thesis.

DaCosta described a brief history of WMNs in Chapter 15 First, Second and Third Generation Mesh Architectures in the book Emerging Technologies in Wireless LANs [59]. This history iterates the technologies has been used in past WMNs, as well as new technologies that have the potential to be deployed.

Especially the third generation WMN described in the chapter indicates a fully software oriented WMN without the regard of specific wireless technologies. This software-oriented approach pointed out the feasibility of employing WMN in simple sensor nodes.

The Serval project [60], developed by Gardner-Stephen, is an Android app using Wi-Fi equipped smartphones to form WMNs. A software-based routing algorithm is used in the Serval WMN to provide the network services. It was derived from the mesh potato device of The Village Telco project. [60] Initially developed as a mesh-based telephone network, data service was soon added into the network. The data was handled using an ad hoc transfer method called the MeshMS [61]. This method uses a store-and-forward protocol to achieve an infrastructure-free micro-blogging ready network service. It has demonstrated the feasibility of delivering text messages more than 10,000 km. The achievement of the project and the simple technology used have inspired the idea of the algorithm developed in this thesis. The simple routing or numbering for telephoning system in this project, namely Serval Distributed Numbering Architecture as shown in Figure 2.1, is not suitable for the use in the remote monitoring networks in this thesis because it is over complicated as each transmission requires a flood a query to the entire network.

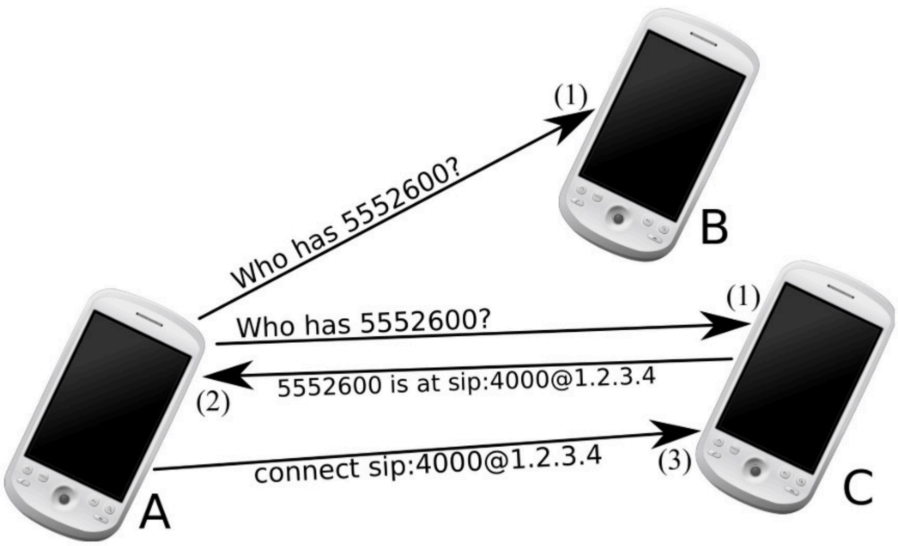


Figure 2.1 Call Resolution in the Serval Distributed Numbering Architecture [60]

One of the issues of WMNs is scalability, as discussed in Section 1.3. Sampaio, et al. reviewed the scalability and path stability issues in the IEEE 802.11s WMN networks [43]. It was reported that the multi-hop nature and the 802.11 network legacy limited the scalability of the 802.11s WMNs. The path selection protocol Hybrid Wireless Mesh Protocol (HWMP) defined in 802.11s is sensitive to the number of nodes which imposes the further limitation to its application in remote monitoring. Being a Wi-Fi-based standard, 802.11s and HWMP is a short-range network technology and not designed for the use in wide area monitoring networks. This hints at the possibility of introducing a new routing approach for such networks.

Jun, et al. [62] discussed the evaluation method for the exact capacity of a stationary WMN for network provisioning. The assumptions and scenarios described in the paper are very similar to the scene set for this thesis. But a chain topology was used to simplify the network structure in the model, it is not feasible in the network discussed in this thesis as the dynamic status of the nodes does impact the decision of the routing. However, the method discussed in the paper provided a useful reference for the modelling of the network in Chapter 3.

Many other works in the field of WMN have also been considered and are referenced for the implementation of this thesis. Bejerano [63] described a method that could connect the static WMNs to the backbone network efficiently. The method divided the network into several sub-trees for data delivery using clustering algorithms. This division of the network also considered the QoS factors of the transmission. The connections between the WMN and the infrastructure maximised the cluster throughput under the given QoS constraint. With the connections, the nodes in the remote monitoring networks could deliver the data to the Internet efficiently. Data transmission in the network could be performed using a publish/subscribe mechanism as described by Adi, et al. in [64]. This paper tackled the inefficiency of data collection problem using the traditional client/server model. The method decoupled certain data processors and collectors by applying the publish/subscribe model. The flowchart of the method is shown in Figure 2.2. This method has proven to be more reliable in the unstable network such as WMNs and has better scalability through parallel operations. This method can be employed in the remote monitoring networks for the collection because of these attributes described above.

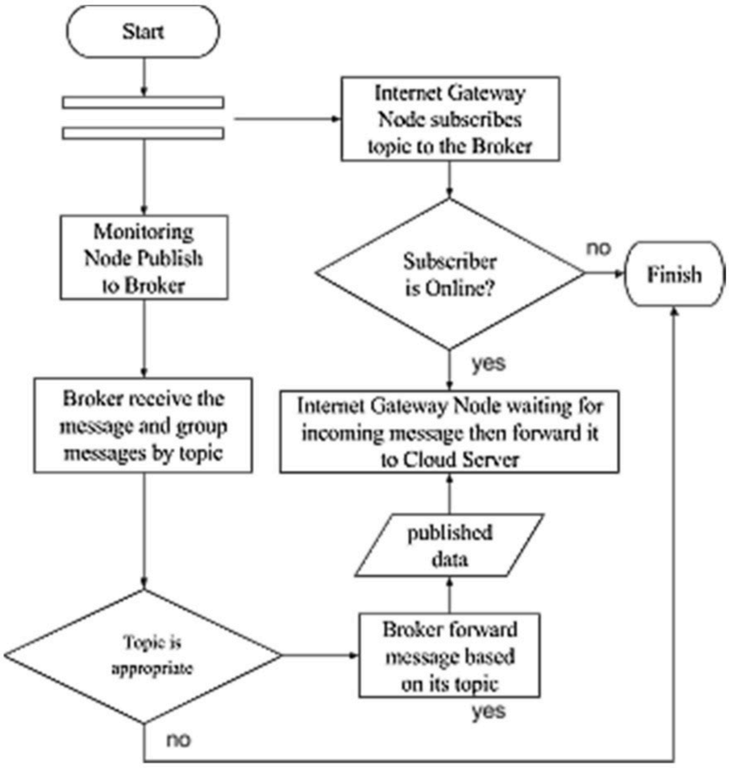


Figure 2.2 The push/subscribe model system flowchart [64]

Zhou-kangas [65] used NS-3 to model and simulate 802.11s Wi-Fi-based WMNs. The routing protocol described in the thesis used the 802.11s recommended HWMP. It was found out that the network efficiency indicator factors including throughput, delivery rate decreased with the increase of the higher application data rate, but the delay grew. The author also concluded that the antenna configuration played an important role in the performance of the networks. Jiao, et al. [66] have modelled the transmission in a two-hop network as shown in Figure 2.3. In the paper, it proposed a Multiple Relay Points (MRP) selection method to replace the Multipoint Relay (MPR) defined in OLSR for multiple hop design. The selection model was also based on the media access control (MAC) sublayer of the data link layer in the Open Systems Interconnection (OSI) model. The MRP selection method was compared with the MPR based method and it was concluded significant improvement in terms of throughput was achieved. The modelling of the transmission in this work has partially inspired the model of this thesis. Jain, et al. [67] has studied the impact of interference on multi-hop wireless network performance. In this thesis, we have considered the interference in the model of remote monitoring networks in Chapter 3, and in the simulations, we chose to put the focus on other critical parameters instead of the impact of interference in the remote monitoring network at the present stage as explained in Section Chapter 5.

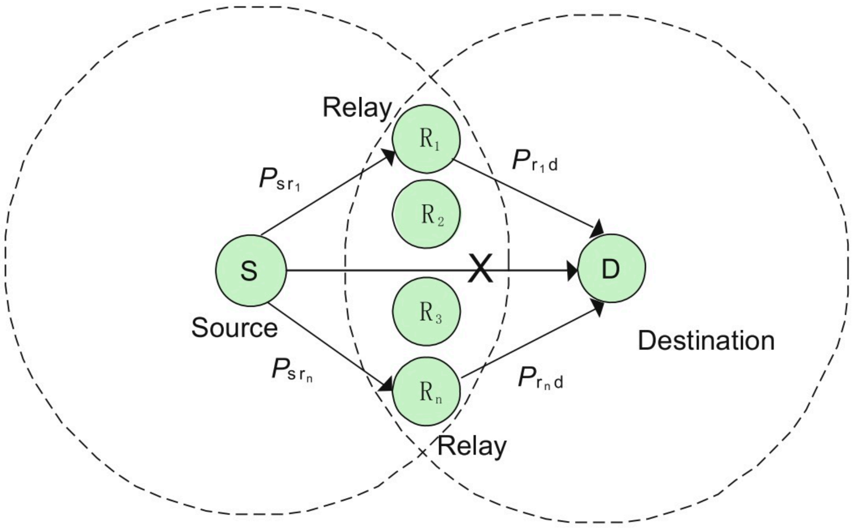


Figure 2.3 System model for two-hop cooperative communication [66]

2.2 Wireless Sensor Networks

2.2.1 Modelling WSNs

The remote monitoring network is one kind of WSNs that is dedicated to working under the environment described in Section 1.1. Modelling the network is one of the most important steps when designing the routing algorithm. The approach some other network technologies modelling the network and the environment is a great reference to the thesis. An open ISM band LPWANs that has been widely used and discussed is LoRa technology.

Bouguera, et al. [68] modelled the LoRa and LoRaWAN based Wireless sensor networks. In the work, it was focused on LoRaWAN protocol as the sensor nodes' battery life in the WMNs is significantly influenced by the wireless technology. The modelled the sensor node is shown in Figure 2.4. The selected transceiver chip is SX1272 from Simtech. It evaluated the impact of the parameters have on the energy consumption including spreading factors (SF), coding rate (CR) and payload size. It was concluded with a list of optimised parameters for energy consumption on different communication ranges. Additionally, transmission power is considered being more important than SF when it comes to the impact on energy consumption. Three different scenarios with a two-way transmission between the gateway and sensor nodes with uplink, downlink and error link situations were compared. It was found that a higher data rate with a shorter range resulted in the better autonomy of the nodes when it comes to LoRaWAN Modes. The proposed energy model was used to estimate the lifetime of the sensor nodes with the acknowledgement transmission on the LoRaWAN. The network model presented in the paper provided useful information for the modelling of the sensors and the structure of the network in this thesis.

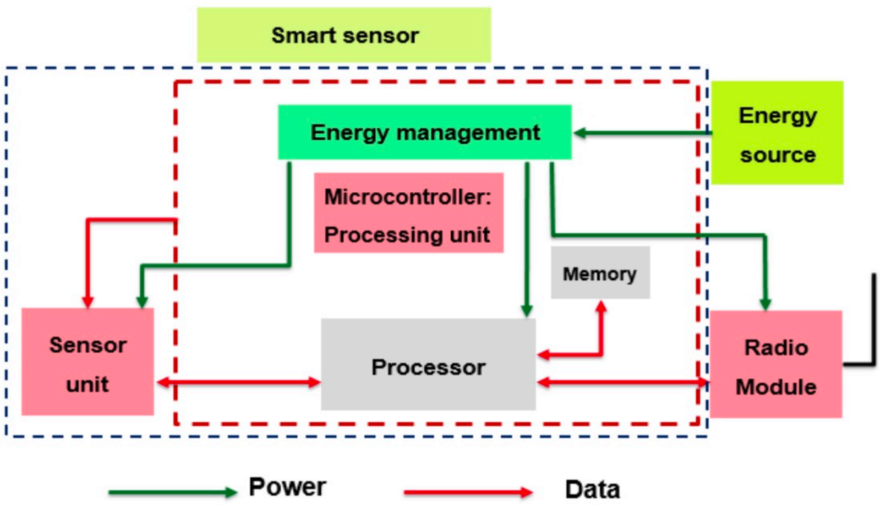


Figure 2.4 Sensor node architecture in Bouguera's model [68]

Another IoT network which based on LoRa and LoRaWAN model was proposed by Wixted, et al. [69] They have tested the LoRa network coverage in Glasgow city using two devices with the same antennas. One of the devices was deployed on the roof of the university building, while the other one equipped with a GPS receiver keeps moving and recording the location information and the communication quality. It was found that the transmission range in the city of Glasgow was about 2km, with the hill in the middle. The packet loss in the system was also measured. A field test was carried out in the mountainous area in China for this thesis [26].

A transmission power optimisation based on an energy consumption model for the WSN is proposed by Mohammed, et al. [70]. The model is to measure the total energy of a successfully transmitted one bit of data needed from a source node to a destination node. The calculated energy included the chance of the offset for the failure transmission using the possibility of success. It also included all the cost of overheads in different layers in the network to calculate the average energy consumed per bit by transmissions over the AWGN channel. An optimised transmission power strategy is then given using the model with different modulation approaches, including BPSK, M-PSK and M-QAM. As it is an analytical model for evaluating the energy consumption in the WSN, it is used as a reference for designing the cost function of the algorithm.

An analysis of the energy consumption model of energy harvesting network was conducted by Song, et al. in [71]. The paper used a ZigBee-based short-range WSN to analysis three major parameters in the WSN: 1. The consumption of powering up; 2. The consumption of acknowledgements; and 3. The consumption of routers. The results showed that the main power consumption in the ZigBee-based energy harvesting network was the first two power sources. This work is used in the consideration of the energy consumption model in this thesis as the powering up and acknowledgement are both included with the remaining power when the calculation of the cost of each leg of the transmission.

Path loss is one of the most significant figures to analyse the characteristics of wireless networks. Kurt et al. investigated several models of path loss and their uses in WSNs [72]. Particularly, they pointed out that the constraints of the propagation in WMNs can be summarised as 1. Low antenna heights; 2. Low transmission power; 3. Stationary network topology; and 4. Directivity of antennas.

They then analysed and compared different models under various scenarios and frequencies. They have noted that the free-space and two-ray parallel-polarization models are not suitable for the path loss estimation for WSNs. This conclusion is based on both models not providing accurate estimations of any metrics in their research. When it comes to the applications in 868 MHz, they have identified the best results have given by the one-slope model. We used a modified free-space propagation model in the process of this thesis as it has provided consideration of the effects of the channel rather than only the attenuation. Additionally, the model simplified the measure requirement in the one/two-slope log normal models for the ease of the calculation of the simulation. Hence, we considered the model as the model of choice in this thesis.

The interference of operational LoRa network can be found when transmission collides in time, frequency and spread factor but not with the expansion of the range, according to a study by Georgiou. O and Raza. U [73]. The degradation in performance of LoRa networks gauged in uplink outage probability when load increases is exponential, limiting the network scalability. It was concluded that one of the reasons for this degradation is the single gateway uplink system model which causes a considerable amount of collisions in due to interference the tested network. This thesis implemented the LPWAN network in a meshed fashion to avoid congested nodes to reduce such degradation happening. Another LPWAN network study was carried out by Gregora, et al. [74] in order to find the optimal location of deploying the LoRa gateway in a building by testing the packet loss rate of each location. Similarly, we considered packet loss in the failure rate in the simulations of this thesis and used it as an indicator when comparing the algorithm with the benchmarks.

A deterministic path loss model of the WSN networks in the urban environment was studied using satellite images of Cambridge, Massachusetts by Herring. K, et al [75]. By finding the most efficient placement of the transceivers, a reference path loss coefficient α with the range of 2 to 5 was tested. In addition, the air-to-ground links can also be modelled with $\alpha > 2$ with the Gaussian random components. In the wideband microcell outdoor propagation model [76] with different transmitter height and path conditions, the path loss coefficient was provided. In this thesis, we used $\alpha = 2.8$ in the simulation to reflect the rural environment that the remote monitoring network operates.

2.2.2 Energy Awareness in WSNs

When considering the energy consumption of the node in a WMN, Murali, et al. [77] have put forward a generalised model. The modelling of the consumption is crucial when considering energy awareness as it is how the energy is used in the calculation to be used in the algorithm. The structure of a sensor node in this paper is shown in Figure 2.5. The model in this paper is based on the network level in the OSI model which matches the design of the algorithm of the thesis, The purpose of the model in the paper is for use in the energy management systems of WMNs, however, we also used the idea of this model when building the model of this thesis.

In Murali's model, nodes belong to some particular clusters with a particular cluster head (CH) of each cluster. They differentiated the CH with other nodes in the network, similar to what the reinforcement learning routing algorithm considers the SN, DN, and INs. The energy consumed during entire action, such as transmission, is a weighted sum of nodes, as the cost functions used in the reinforcement learning routing algorithm.

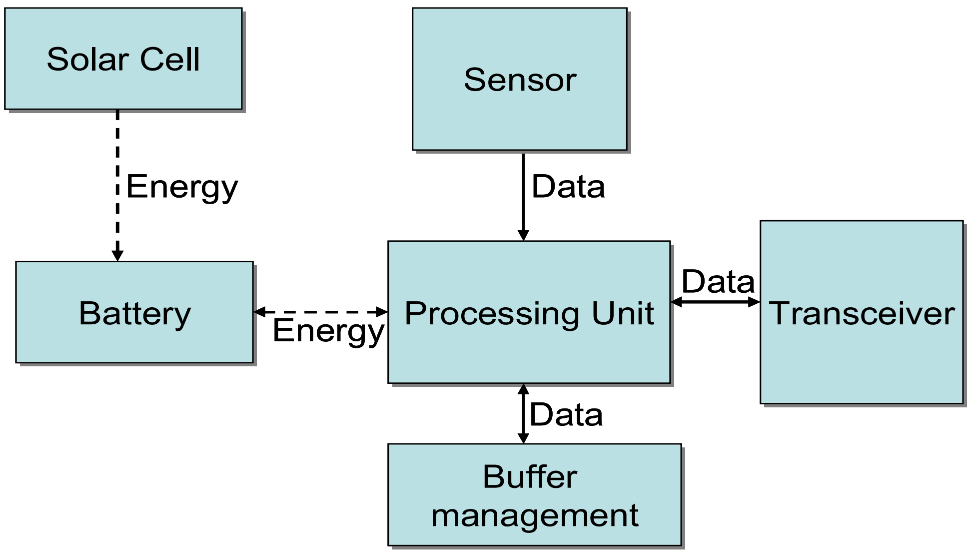


Figure 2.5 Sensor node architecture in Murali's model [77]

An energy harvesting powered sensor node model was developed by Ruan et al. [78]. This model used node-based design to manage energy consumption. By implementing the sleeping cycles with the energy harvesting cycles, the nodes manage the energy mismatch between the energy demands of the device and the energy gathered through harvesting. This is partly used in the consideration of the charging cycles and energy model design of this model as the nodes in this thesis are also relying on energy harvesting sources. However, the model of this thesis takes a simplified model of this method to give the focus on energy awareness at a network level. Another evaluation of WSN platform has been conduction by Antonopoulos et al. in [79] as a reference to the power consumption model as well.

2.2.3 WSN integration and other WSN works

The integration of WSNs technology and other wireless network technologies are also beneficial when considering the model of the edge the network, such as gateways. In the model of this thesis described in Chapter 3, we considered some of the sensor nodes to be the possible gateway. It is very important to understand the design of the gateway and how it is integrated with other networks. In this section, we studied some other works to help make that decision when modelling the network.

Navarro-Ortiz, et al. [80] discussed the integration of LoRaWAN and the cellular network for industrial IoT applications. The paper has shown an interesting idea to build a seamless fusion integration solution to merge the services between LoRaWAN networks as well as cellular networks. It modified the LoRaWAN gateway to work as an eNodeB in the Evolved Packet Core (EPC) framework on a 4G Long-Term Evolution (LTE) network by implementing LTE signalling on the gateway. The proof of concept testbed network enabled LoRaWAN end-node devices to send data through LoRaWAN and LTE core network while maintaining transparently as well as keeping its LoRa end-to-end security. The gateways of the remote monitoring network described in this thesis also need to be reliably connected to the internet. The possible remote location of deployment of the sensor nodes which operate as the gateway in this paper is assumed to have only the connectivity to the cellular network and have no access to other Internet connectivity. Hence, the work of this paper has proven the technological feasibility of the remote monitoring network with limited Internet connectivity and LPWAN coverage.

One of the motivations of performing the work in this thesis is to improve the energy efficiency of rural remote monitoring to provide a longer and deeper understanding of the environment without outage. It is as beneficial for the society as the smart city projects such as Smart WSN-based Infrastructural Framework for smart Transactions (SWIFT) architecture introduced by Nandury [81]. The work modelled the IT infrastructure of a smart city network working with IoT devices. Especially, the section of smart environment monitoring integration with the Node context analysis can be used in the future work of remote monitoring networks to help the responsiveness of the reaction of the data from the network.

Mehmood, et al. [82] surveyed (machine-to-machine) M2M communications using cellular LTE-A as well as future 5G networks. The existing and future cellular networks were assumed to be ready-to-use when it comes to mobile M2M communication. It has been identified that using cellular networks has challenges such as inefficiency in using the physical layer, manufacturing low-cost modules, and limited battery life of end nodes. It was suggested that relay-based data aggregation schemes can be used in cellular M2M networks to reduce overhead as well as the network congestion. This is similar to the scenario described in the context of this thesis with LPWAN networks.

The coverage of different IoT networks was studied by Lauridsen. M, et al. [83]. It was simulated a probability coverage model of a 7800 km2 area and compared the coverage of GPRS, NB-IoT, LoRa and SigFox based on the network operator Telenor. NB-IoT provides the best coverage while the GPRS is the worst. But the legacy LTE system will interfere with the NB-IoT when the distance between the sites increases, thus increasing the outage. We have considered a similar interference situation when modelling the network.

A LoRa based LPWAN architecture called OpenChirp [84] also inspired the modelling of the network when it comes to using a LoRa-like technology to simulate the real-world usage. In their proposal, Dongare, et al. have demonstrated a proof-of-concept of open LPWAN network.

Jones, et al. [85] have identified the power conservation requirement to be implemented at the higher levels in the protocol stack rather than the physical layer. This is because the physical layer technologies are designed with best power efficiency based on their designs. It was also mentioned that the viability of the wireless services depends crucially on the power efficiency of the protocol. In this thesis, we consider the energy consumption on the network layer to route the network as this paper suggested.

Bouguera, et al. [68] also mentioned that the processing power, such as speed, of the microcontroller and how often they are brought out from sleep mode in the nodes affects the energy consumption of the lifetime of the network. We take the approach that minimises the extra overhead for routing of the microcontroller when routing to accomplish a higher overall usability of the entire network.

An application of such monitoring networks is animal monitoring in the farms. Benaissa, et al. [86] identified that the fading model of the off-body sub-GHz wireless channel in the barns can be characterised by a one-slope log-normal path loss model. The path loss exponent α is about 2 is also gained inside the barn as low shadow fading effect in the barn.

Fan [87] proposed an EnergyxDelay value as a metric for the performance evaluation of high energy-efficiency and low transmission delay sensor network. The value is the product of energy consumption and transmission delay between the sensor nodes and the base station. The authors also discussed the application of such a metric in a distributed topology. However, as the main focus of this paper was of the base-station-based star network and one of the main considerations of the author was the delay which is considered less important in the context of this thesis, the EnergyxDelay value was not used in the algorithm of this thesis. The logic and idea behind still used as a reference here.

Many other works related to the WSN have been performed over the years, this section just can cover the most significant works that either inspired the work of this thesis or used as a reference when designing the models and the simulations in this thesis.

2.3 Machine Learning in Wireless Sensor Networks

The introduction of ML into wireless networks can benefit resource utilization in the network and increase the lifespan of the network. WSNs require more sophisticated routing solutions and be energy efficient at the same time. Jiang, et al. [88] has summarized some patterns that can incorporate ML in the Wireless Networks which includes routing. Mohammad, et al. [89] reviewed a number of different machine learning methods in WSNs in the period of 2002-2013. This review helps to guide the thesis in the direction of using the ML in routing.

One possible area path selection using ML in the WSN was in the MAC layer. Chu, et al. [90] presented a Q-Learning and ALOHA based MAC protocol called ALOHA-QIR. This protocol takes advantage of both ALOHA and Q-Learning to be a simple design and low energy consumption while keeping a low collision rate. It used Q-value stored in each node to reserve the air timeslot for the MAC of the network. However, the initial collision rate can be higher than normal ALOHA as the nodes are learning the network. Similarity can be found in the routing algorithm presented in this thesis.

Another work on the MAC layer was the Self-Adapting MAC Layer (SAML) by Sha, et al. [91]. The SAML is designed to address the compatibility issue of different MAC protocols in the WSNs by employing a reconfigurable MAC architecture, that uses ML to find the best suitable MAC protocol based on the network conditions. The learning process can be found in the decision tree shown in Figure 2.6. It uses a series of network indicators to decide which is the best MAC algorithm. The indicators include the inter-packet interval (IPI), the received signal strength indication (RSSI), the priority of QoS requirement (Energy, Latency, Packet Delivery Rate (PDR)) and the network traffic pattern. The work has shown an improvement of overall performance but introduced additional complexity. Hence, in this thesis, we consider our solution as network layer based to avoid this compatibility issue like this.

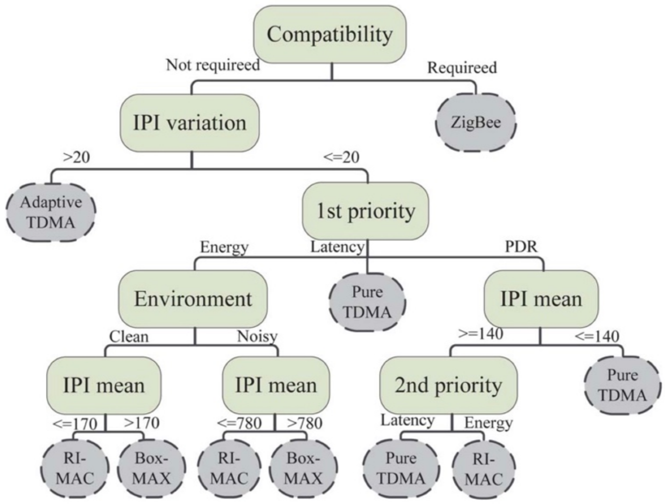


Figure 2.6 SAML MAC algorithm decision tree [91]

As early as 1994, Boyan and Littman has proposed a Q-Learning based routing algorithm for the wired network [92]. The algorithm did not require prior knowledge of the network topology and traffic patterns. It did not need a centralised controlling system. The simulation results have shown that this adaptive routing performed better than traditional routing algorithms in the dynamically changing wired network. It was also mentioned in the future work of the paper that the routing table values was replaced by a function approximation which is similar to what this thesis's routing metric likeliness approach.

Hu, et al. [93] proposed a Machine-Learning-Based routing protocol for underwater WSNs called QELAR. By employing Q-Learning, Q value tables instead of routing tables are stored in the nodes. The Q-Value is calculated based on the expected lifetime of the node. Similar to the idea of this thesis, it trades latency and throughput for energy efficiency. But the consideration of the approach is quite different and the comparison between a geographic routing Vector-based-forwarding (VBF) and the algorithm is not suitable for the simulation of this thesis, as the VBF requires the measurement of the incoming signal angle which requires an added-on device on every node.

Another Q-Learning based WSN Routing algorithm (AdaR) was proposed by Wang P, et al. [94]. The algorithm is configurable to different goals for different uses of the network. It achieved a better performance than the naïve Q-Learning mentioned in [92]. However, the added overhead of supporting different goals are not ideal for the limited resourced WSN nodes, and the selection of the next nodes are purely based on the Q-Value with simple ε greedy exploration whereas in this thesis the Boltzmann exploration is used for a better balance of exploration and exploitation. The AdaR can be found sensitive to the learning rate. Therefore, an algorithm that is less sensitive to the change in the learning rate is desirable.

Other works in WSN with ML utilisation include a biomedical sensor network routing protocol with QoS support by Liang, et al. [95]. Machine learning was used in the estimation of the QoS properties in all routes to find the best suitable route for the given QoS transmission based on the biomedical applications. Boushaba, et al. used a new routing metric called RLBPR to balance the load the gateways in WMNs using reinforcement learning [96]. These works are relevant to the algorithm proposed in this thesis. However, the protocol in [95] is limited to biomedical applications as the network model are tied to that scenario while the proposal in [96] of using RLBPR is similar to what has been used in this thesis, but the different emphasis constitutes the different idea of the way using the reinforcement learning, the one in [96] addressed on the load balance in the network while the one in this thesis on the energy efficiency and usability of the entire network.

Two exploration model for reinforcement learning-based cognitive radio spectrum sharing networks were introduced by Mitchell, et al. [97]. The main objective of this work was to efficiently use the available spectrum using software defined radio. By randomly reserving the spectrum and pre-partitioning the spectrum pool, both methods can increase the efficiency of spectrum efficiency. This idea of random reservation and pre-partitioning were considered in the selection of exploration and exploitation of the algorithm in this thesis. However, the traditional Boltzmann exploration can simplify the process and create less deviation from the focus of the thesis, hence, we still chose Boltzmann exploration as the method in this thesis.

In an expanding cellular network scenario, the management of the changing cells can be a challenge, Bennis et al. [98] proposed a reinforcement learning based solution to establish a decentralized and self-organizing mechanism for such objective. The base stations have only the local knowledge which gathered from the feedback from the connections. The idea is similar to the design of the reinforcement learning routing algorithm.

Another reinforcement learning method has been introduced for the wireless networks, notably the adaptive exploration strategies by Hwang, et al. [99] can benefit the efficiency of the learning process. However, the additional computation power is not suitable for the power constraint sensor network. Therefore, we used a fixed exploration strategy in the algorithm. The Q-Learning algorithm also used in routing in the map as shown in [100] by Guo, et al. The additional Q table upon the routing table is not ideal when it comes to the sensor network, so we use simpler TD-based learning in the algorithm. Another example of using Q learning in wireless mesh networks was by Vazifehdan et al. [101]. With consideration of the residual energy instead of the recharging cycle, we found the method too complex for the sensor network.

2.4 WMN Routing Approaches

2.4.1 Proactive routing protocols

The proactive routing protocols maintain the routing information by storing an up-to-date routing metric in a routing table in each node. One of the biggest challenges for these protocols is propagating the changes in the network. Protocols usually use broadcasting to conduct such tasks. However, different protocols differ in the approach to how the broadcast handled.

One of the most common proactive routing protocols in WMNs is Optimized link state routing protocol (OLSR) [102]. It is an ad hoc network optimized version of the link state protocol, hence the name. It uses periodical HELLO messages to sense the neighbours of the nodes. Topology Control (TC) packets are used to exchange topology information between neighbours. Then, Multi Point Relays (MPRs) are used to relay the information of the routes between certain nodes in the network which will be calculated using the shortest path first method.

However, for the links without the MPRs, the path is discovered over time.

Another early routing protocol is the Fisheye state routing [103]. The nodes in Fisheye state routing only maintain a limited knowledge of network, just like the reinforcement approach we used in this thesis. The link states are only updated in a smaller scope, as shown in Figure 2.7.

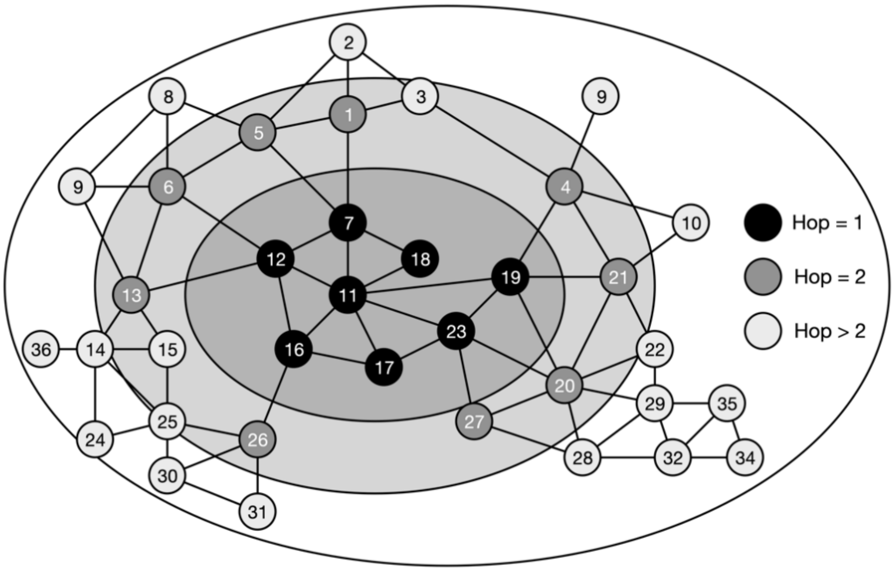


Figure 2.7 Scope of a fish eye routing system [103]

Another basic WMN routing protocol is called Wireless Routing Protocol (WRP) [104]. In WRP, each node keeps three tables: routes, distances and link costs. An additional message re-transmission list is also maintained in the network. These tables are periodically updated by update messages with its neighbours or when the link state changes. WRP is fast converging and involves fewer table updates. However, large memory storage is required in each node for maintaining the tables. Thus, scalability is impaired by this requirement. It also proves not fit for the WMSNs as the required large storage is not usually available in these devices in the network.

Destination-Sequenced Distance Vector (DSDV) [105] is also an early routing protocol for the WMNs. One similar point between the DSDV and the reinforcement learning routing algorithm is that both algorithms keep the first node on the shortest path to every destination node in the network. However, DSDV also keeps a set of current neighbours whereas the reinforcement learning routing algorithm keeps all the possible next-hops in the table. The DSDV, like other proactive routing protocols, is updated using periodically broadcasting messages. The update is triggered when the nodes changes status. It advertises the change to all neighbours. This can limit the mobility of the nodes where constant update is needed when one of the nodes is moving. The delay of propagation of information can also cause problems, due to the information is only updated periodically.

2.4.2 Reactive Routing protocols

Ad hoc On-Demand Distance Vector (AODV) [106] can be considered as the reactive adoption of the DSDV. Only the active paths routing information is maintained in the AODV routing table. The routing table maintains the possible next-hop at each node. This is to reduce the controlling messages in DSDV and overhead on traffic, which have impacts on scalability and performance. This brings about one disadvantage of the AODV, the efficiency can be affected as the path is not always up to date. The active neighbour nodes in a path are also periodically sending HELLO message to maintain the active status of the path. If one of the nodes does not receive the message, the path will be deemed inactive and be deleted. This creates another disadvantage of the AODV, which wastes the bandwidth for the HELLO messages. AODV has been used by default in the Zigbee ad hoc networks. The reinforcement learning routing algorithm overcomes these disadvantages by learning the information of the network over time, especially in a low mobility sensor network.

TORA (Temporally Ordered Routing Algorithm) is a source initiated on-demand routing protocol [107]. It aims to reduce the overhead of adaption of local topological changes in the network by limit message propagation. Each node only stores its adjacent nodes information and the local topology. The routing is done through a series of node searching. TORA also support multiple paths between two nodes. It works in a similar way to the reinforcement learning routing algorithm. However, reinforcement learning makes the algorithm inherently more adaptive than TORA.

Similarly, the Dynamic Source Routing Protocol (DSR) [108] also uses source routing to deliver packets. The network is completely self-governed with no administration supported. It is very simple and efficient as transmission only happens when there is no known route between two nodes, then the route discovery mechanism is initiated. This simplicity brings about the limitation of the maintenance of broken links. Besides, the set-up delay can be higher than the proactive routes as the packet need to find a route first. The reinforcement learning routing algorithm has the same problem of the set-up delay, however, in the non-delay sensitive WMSNs, it is not a problem. The maintenance of the broken links can be avoided, as it is learnt by the algorithm.

A distance routing effect algorithm for mobility (DREAM) [109] is an ad hoc network routing protocol taking distance effect and mobility rate as main metrics of the routing decisions. Distance effect is the effect of the distances between nodes on the routing decision. The fact that the respective speed reduces with distance, the frequency of routing table updates is determined by both distance effect and the mobility rate of the nodes involved in the route. The DREAM is aimed to provide a loop-free path, and with the support of multiple routes available, it is also robust.

The Better Approach To Mobile Ad hoc Networking (B.A.T.M.A.N.) [110] is a WMN routing protocol that has the decentralization design. No node has all the information about the network, therefore, no knowledge about the best route through the network is stored. The routes are dynamically created from node to node, just like the reinforcement learning routing algorithm.

Babel is another routing protocol based on DSDV, AODV with different loop avoidance method [111]. However, it uses the hop-count on wired networks and a variant of expected transmitted count on wireless links. It is simple and effective as the IETF has created a standard version of Babel and used it in the Homenet working group [111]. However, it is a very general routing protocol and does not learn the topology change over time, unlike the reinforcement learning routing algorithm.

2.4.3 Other works in WMN routing

Murray, et al. [112] investigated some popular routing protocols including OLSR, BATMAN, and Babel for multi-hop ad hoc networks. The paper discussed other factors that influence the performance of routing efficiency, such as addressing and scalability, the ability to run on restricted CPU, bandwidth and unreliable link resources. The study found that the overhead of the routing protocol is the most significant factor when it comes to small multi-hop ad hoc networks and WMNs.

Zeiger, et al. studied using wireless ad hoc networks to control robots by comparing four ad hoc routing protocols to find out the possibility of using the WMN in a real-world scenario with robot controlling system [113] This is based on the vision of the requirement of highly dynamic network topologies for the future robot operation. Four different routing protocols have been investigated for this purpose, namely, the reactive routing protocols AODV and DSR, the OLSR and B.A.T.M.A.N. In the paper, a tele-operation scenario has been used to compare the four ad hoc routing protocols. Alternative routes can hardly be found by B.A.T.M.A.N., AODV and OLSR when the controller connection needs to be rerouted. The DSR has shown the adaptability in this test, and similarly, the reinforcement learning routing algorithm may also perform impressively if included.

Tang, et al. [114] have proposed using a real-time deep learning method to conduct intelligent traffic control instead using routing protocols in WMNs. By exploiting deep Convolutional Neural Networks (deep CNNs) with uniquely characterized inputs and outputs of the network, lower average delay and packet loss rate has been achieved in simulation. However, the large dataset of transmissions in the network representing all possible scenarios needs to be prepared to train the CNN. In the remote monitoring network, the frequency of transmission is not very high in compare to other WMNs. The unexpected change of the network from both the nodes and the environment may also impair the effectiveness of the method. Hence, this method is more suitable for relatively stable WMNs with a high frequency of data exchange, which is not the type of the network discussed in this thesis.

Both More, et al. [115] and Nandakumar, et al. [116] proposed routing protocol for wireless Ad hoc network with energy efficient and Traffic and energy aware. These two methods are more suitable for high power WMNs such as mobile networks, so energy and traffic awareness can be used more frequently. While in the scenario of the remote monitoring sensor networks, the traffic and mobility are not main concerns when it comes to routing, hence, it is not as important as the methods described in these papers.

Han, et al. [117] used a software define WSN to solve the routing optimization problem. The network model they used is a hierarchical tree model as shown in Figure 2.8. The main focus of this algorithm is extending the network lifetime and improving energy efficiency, similar to the purposed of the proposed algorithm in this thesis. The method they used is by adopting a centralized control plane to apply software define network techniques to distribute the routing information to all nodes from the controller on the top of the hierarchy, effectively a gateway. They divided nodes into energy levels and the controller could alter the topology by dynamically changing the weight of the branches to change the routing behaviour. This centralised approach is suitable for the network with a clear gateway so that it can be characterised in a hierarchical fashion, which is not best suitable of the scenario of this thesis. However, the consideration of energy awareness is an important reference for this thesis.

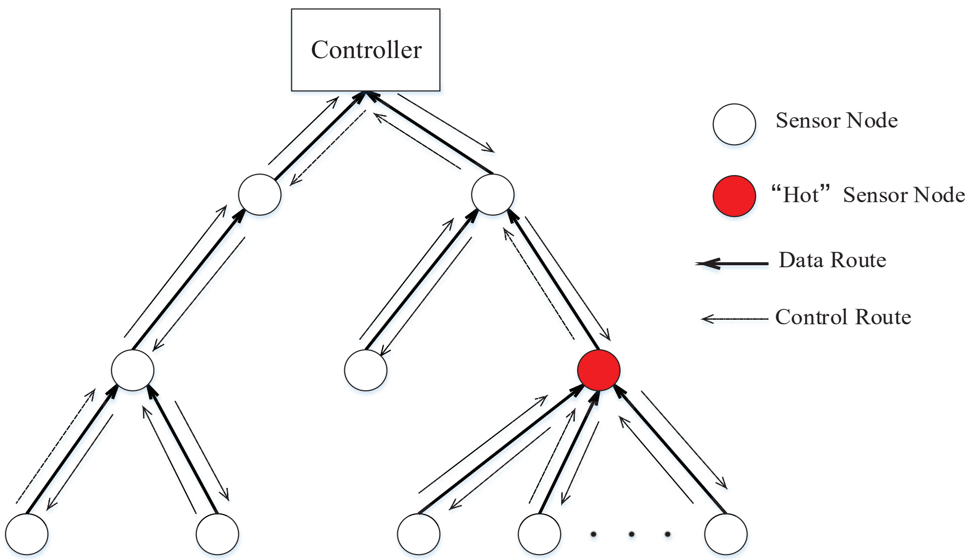


Figure 2.8 Hierarchical tree in Han's routing algorithm. [117]

Hao, et al. studied low duty-cycled WSNs [118]. The authors aimed at a minimum end-to-end energy consumption with the reasonable delay bound as the low duty cycles introduced large delays. The proposed geographic routing protocol uses packet holding before relaying and forwarding the packet to the destination to increase energy efficiency. The scenario described in the paper is closed to the one in this paper, but as the sensor node modelled in this thesis, the limited storage and processing power made this packet holding behaviour nearly impossible in the remote monitoring networks. However, the induction of Markov decision process as well as the reinforcement learning came after that described in this paper, influenced the method selection of this thesis.

By using 802.11s Wi-Fi-based WMNs, Sun, et al. proposed a routing metric for the Wi-Fi IP networks with load-balancing and energy-awareness [119]. This metric is aimed to be used in the wireless multimedia sensor networks that require high data throughput but in the coverage of the WLAN network. The energy-awareness in this metric and the modification of the routing metric for increased efficiency of the routing has influenced the approach of this thesis.

He, et al. [120] proposed a joint solution for using energy awareness with Simultaneous Wireless Information and Power Transfer (SWIPT) networks. By combining routing the packet with power information in the network layer; choosing the transmission mode in the MAC layer and energy allocation in the physical layer, the authors established a full stack of protocols for SWIPT networks. The work has been proven useful in the simulations and the idea of transmitting power in a multi-hop energy-constrained network was very different from other solutions. It has not been considered as useful in the situations of this thesis. The physical distances between the nodes in the context of this thesis is on the scale of kilometres, while in the [120] was metres. This difference made the power delivery wirelessly impossible, hence the method. But the idea of using routing not only to transmit the data but also guide the power delivery is certainly can be studied in the future work of remote monitoring networks.

Another method of routing selection of the WSNs is proposed by Zhang, et al. [121]. In their paper, a link weight and forward energy density based next hop selection mechanism has been utilised. The main consideration in their protocol is the Forward Energy Density (FED) calculated based on the energy value of the source node and all of its neighbours combined and the forward transmission area (FTA) of the node. FTA is defined by the nodes it will forward the packet to as shown in Figure 2.9. The introduction of FTA limited the possibility of the selection of the next hop to the direction of the selection of the next hop. This method is based on the geological knowledge of the nodes which is not in the consideration of the scenario of this thesis. However, the inclusion of energy provisioning is similar to the way the algorithm is designed.

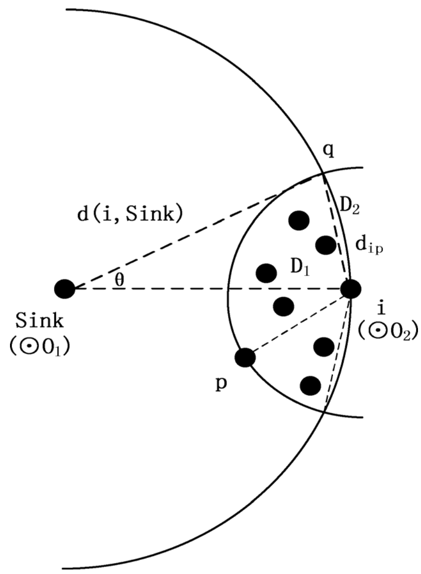


Figure 2.9 Forward transmission area of a node [121]

Other works also affected the choice of routing method used in the reinforcement learning routing algorithm. It includes the turn model for adaptive routing proposed by Glass. C, et al. [122]; The Context-aware Adaptive Routing (CAR) protocol by Micro Musolesi [123]. The CAR protocol is designed for mobile ad hoc networks using unicast communication for delay-tolerant usages. Additionally, the compassion between proactive and reactive routing approaches for wireless sensor networks by Koliousis, et al. [124]; The performance of a variety of routing protocols comparison by Broch, et al. [125] between DSDV, TORA, DSR, OLSR and AODV. The choice of simulation benchmarks in Chapter 5 is also considered the results of the works mentioned above.

2.5 Summary

In this chapter, we have reviewed a series of works on WMNs, WSNs, ML in wireless networks and current routing in WMNs. We have not yet identified any routing protocols nor algorithms that are designed for the remote monitoring sensor networks. This will verify the contributions of the proposed algorithm in the application scenario. The works discussed in this chapter have either inspired the design of the reinforcement learning routing algorithm or have a direct influence on certain decisions made when designing the algorithm.

Chapter 3 Network Modelling

The proposed reinforcement learning algorithm is designed for mesh topology from the ground up by the model of the network itself. We model the network operating in a finite area with a finite number of nodes distributed inside the area randomly in this algorithm. Hence, we defined the service area of the target network by assuming the finiteness of the network in the model. We then model the connection between the nodes in the network. In this model, we consider that any two nodes are adjacent when they are in a certain range and can establish an ad hoc connection between each other. This also models the connections in the network as we define the link as the connection between these nodes. However, not every two nodes in the range can establish a link between each other in this model because of the possible obstacle in between or zero battery power in either of the nodes.

3.1 The model of the wireless channel

As the network is set to be in the rural areas, we assume no coverage of all other kinds of wireless network services in the ISM band of choice. We also consider that the channel has certain interference (I) and background white noise (N). Hence, we can use the full capacity of the channel to reach the given rate of transmission. We can then calculate the maximum transmission rate (R) at the given Signal to Interference and Noise Ratio ($SINR$) of the channel based on Shannon–Hartley theorem and Shannon's noisy-channel coding theorem as shown in (3.1).

$$R = BW * log_2(1 + SINR) \tag{3.1}$$

where BW is the bandwidth of the carrier used in the transmission. To calculate the $SINR$ of the transmissions, we modelled the channel as a fading channel with the path loss exponent α as well as the channel gain coefficient h. We also define the distance between the transmitter and the receiver as d. We denoted the transmitting power used in the transmission as P_t. We then can calculate the $SINR$ using (3.2).

$$SINR = \frac{P_{\mathrm{t}} * |h|^2}{(I + N) * d^{\alpha}} \tag{3.2}$$

3.2 The model of the network

The networks discussed in this thesis consists of a certain number of nodes, with the links between them. There are no pre-defined gateway nodes, as we consider all nodes equal and the Internet connection can be available at any given nodes at any time. Hence the elimination of single point of failure. Figure 3.1 shows an example of a network modelled using this method.

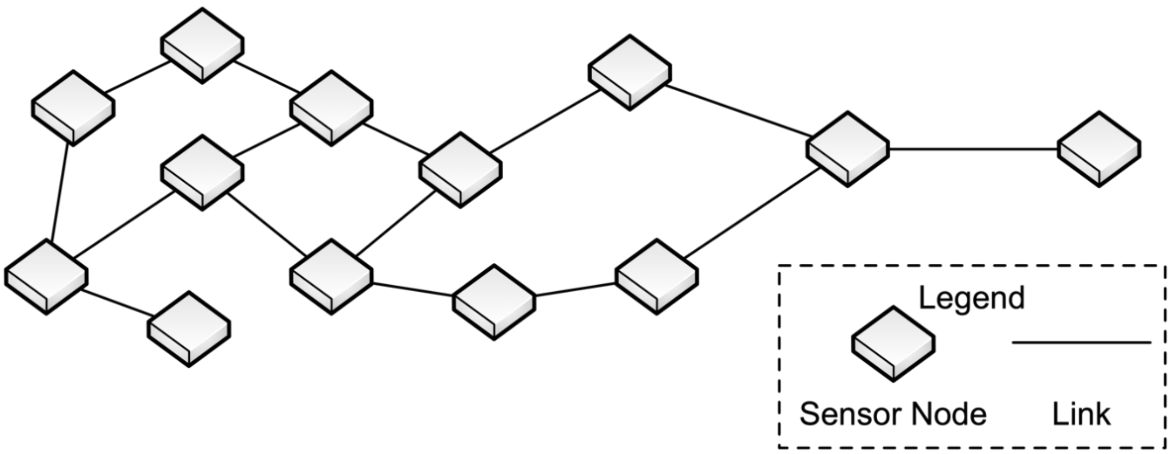


Figure 3.1 An Example of modelled network

3.2.1 Sensor nodes

A simplified model of the structure of a sensor node in the context of this thesis is shown in Figure 3.2. Each sensor node is considered being powered by an internal battery with a fixed capacity of P_{max}, as the sole power source. At any point, we consider that the remaining energy in the battery is known as P_{now}. For the sustainability of the node in the remote area, we assume all the nodes are connected to a solar-powered charger for recharging the battery. For the simplicity of the model, we emulate the charging process by resetting the P_{now} to its maximum value (P_{max}) after the period of a pre-defined charging cycle (CC) with the unit of time. The length of the CC depends on the given climate condition of the model and is calculated in Section 1.1 accordingly.

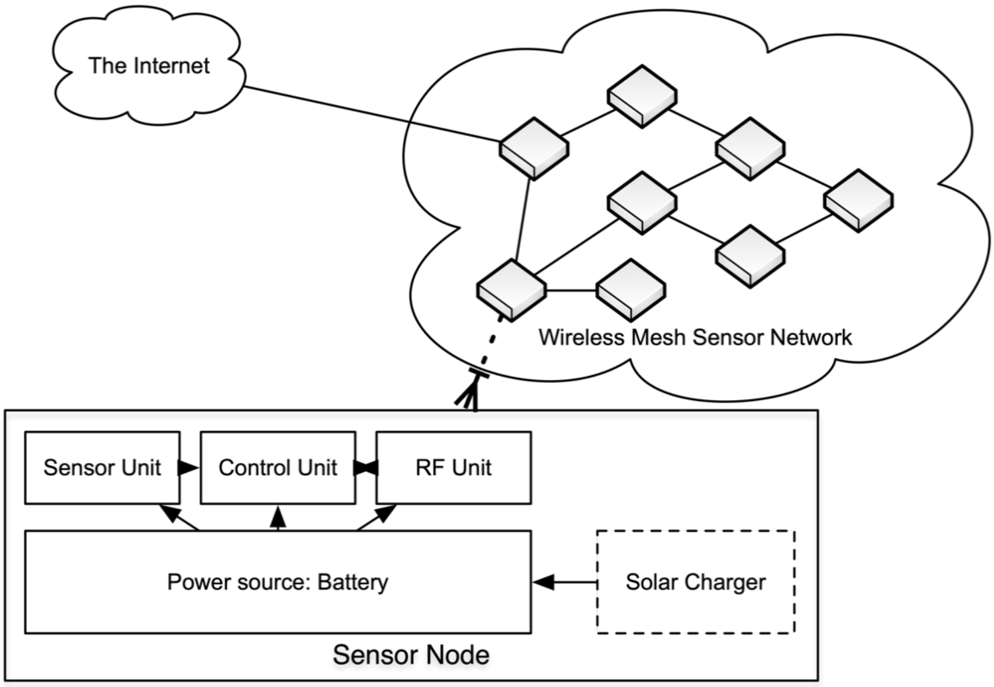


Figure 3.2 Structure of a sensor node

Current solar panels are small enough and capable of powering up the sensor nodes. Here we provide a commercial example of such solar panel that is accessible as well as affordable for such networks [126]. The size of this panel can be fitted onto most enclosures of the sensor nodes and the power output of the panel (P_{charge}) is sufficient to charge the nodes consistently. There are many better solar panels in the industry, but such one can serve as a reference for evaluating the charging cycle. The battery gets recharged t_{charge} hours each day. Nowadays, most mobile devices use Li-ion batteries which have a very high charging efficiency and have a small discharge loss [127]. We consider that the discharge loss is 0 because it is relatively small when compared with the operational discharge. We denote efficiency of the charging circuit as η_{charge}. We then can calculate the CC in the unit of days using (3.3):

$$CC = \frac{P_{\max}}{P_{\text{charge}} \times t_{\text{charge}} \times \eta_{Charge}} \tag{3.3}$$

During the operation of the nodes, there may have several possible phases that consume the battery power:

1. Data collection;
2. Data processing;
3. Sleeping;
4. Transmitting;
5. Receiving;

The data collection phase means the sensor is working to collect the data needed. However, due to the huge possible variety of sensor applications can be used in the remote monitoring network, it is difficult to precisely model the power consumption of this phase. Besides, the transmission power is much greater than the working power consumption of the sensors. For example, LoRa radio chip SX1278 requires a typical supply current in transmitting mode of 120mA when the RF gain is 20 dBm, while in the receiving mode, it requires 11.5mA [128]. Where an example of this total dissolved solids (TDS) water quality sensor [129] has a working current of the entire module, including the probe, is 3-6 mA. Moreover, most of the time, the sensor will be in the sleeping phase to conserve battery, and the power consumption is minimal. Thus, we consider mainly the transmitting power when calculating the cost of each point-to-point transmission in the algorithm. However, we do factor the percentage of remaining power of the node (the ratio of P_{now} / P_{max}) into the cost function of the algorithm which will be discussed in Section 3.2.2. Anyway, it is possible to include the power consumption of the sensors in the data collection phase in the calculation to reflect the cases of high-power sensor consumption when necessary. However, we put the focus of this thesis on the more dominate transmission power in the present stage to have a better understanding of the impact of the decision of routing on the operation of the network.

Each node has a routing table stored in its control unit. The routing table keeps track of all the possible routes from that node to all possible nodes in the mesh network. It starts of the destination column and also has a next node column as oppose to gateway column in the traditional IP routing tables [130]. The node keeps track of all its adjacent nodes using the next node column. As the network is not subdivided into subnets, there's also no need for a netmask column where can be found in IP routing tables. The Routing Metric (RM) values of each route can be found in the next column. The RM values indicate the likeliness of selecting the next node in the same row when routing to the designation node. Here we use the likeliness instead of the possibility of selection is because we use the Boltzmann exploration process to balance exploration and exploitation behaviour for the best result of the reinforcement learning, which will also be discussed in Section 4.2.1 of this chapter. A visited column keeps track of how many times the node being visited, and this information is also used in the reinforcement learning process. An example of a Routing table of node 1 is presented in Table 3.1.

Table 3.1 An example of the routing table

Destination	Possible Next Node	Routing Metric	Times Visited
3	4	1	131
4	3	1.42	22
4	5	1.71	31
4	6	3.18	80
5	4	2.13	68
5	6	0.13	2
6	4	4.98	186
6	5	0.05	3

3.2.2 Links between nodes

The connections between the nodes are called links in the context of this thesis. Each link in the topology is bidirectional which means transmission can be initiated from either end of the link. If there is a link between two certain nodes, they are called adjacent nodes. The power consumption of transmitting from the source node (SN) to the adjacent next node (NN) at a given rate R in the channel with the parameters of interference I, noise N, the channel gain coefficient h, and path loss exponent α can then be calculated using (3.4) based on equations (3.1) and (3.2):

$$P_t = \frac{(2^{\frac{R}{BW}} - 1) * (I + N) * d^{\alpha}}{|h|^2} \tag{3.4}$$

where d is the distance between the SN and the NN.

When a transmission happens from the SN to the Destination Node (DN), a route is decided on the fly as the network condition may change at any moment. The SN will first randomly pick an NN from its routing table in the column of the corresponding Destination with the possibility calculated by using the Boltzmann process (3.5):

$$p(NN_n) = \frac{e^{RM(NN_n)/\tau}}{\sum_{i=1}^{n} e^{RM(NN_i)/\tau}} \tag{3.5}$$

where $p(NN_n)$ represents the likeliness of the nth possible NN (NN_n) of the given DN being selected by the SN from the routing table. We also use a hyper-parameter τ to control the spread of the SoftMax distribution of all the possible routes, as discussed in Section 4.2.1. The NN then picked by the SN using a simple random process with all the possibilities to the same DN. The calculation of the cost (C) of this point-to-point transmission between SN and NN is calculated using (3.6):

$$C = W_1 P_t - W_2\, log(P_{SN}) - W_3\, log(P_{NN}) \tag{3.6}$$

where W_1, W_2, and W_3 are some pre-defined positive weights which depend on the required learning behaviour, and P_t represents the transmission power used in the corresponding leg of transmission calculated using (3.4). The weights are defined manually through a serial of trials. P_{SN} and P_{NN} are the power percentages of the transmitting and receiving nodes of SN to NN transmission, respectively. These two power percentages of the nodes are calculated by the fraction of the remaining power of each respective node P_{now} over their full capacity P_{max}. We then apply a log function on each power percentages to provide a value that is negative infinite when the P_{now} is 0 and the value is 0 when the power is full (P_{now} = P_{max}). In this fashion, the cost of going through the node which has a very low battery level is very high. By using this cost function, the algorithm takes not only the P_t but also the P_{SN} and P_{NN} into account in the learning process, which enables the learning algorithm to adapt to the changes of energy states of the nodes involved in this end-to-end transmission. This can result in the ability of the algorithm to adjust the routing strategy accordingly. Traditionally cost function of routing algorithms in IP networks only fact in only the number of hops of each route [130], the consideration of these power parameters is where machine learning ability improves over the standard IP routing when it comes to the performance of the network over time.

3.2.3 Transmissions

We define a transmission as an action of transferring one data packet from a source node (SN) to a destination node (DN) through several intermediate nodes (INs) along a certain route (in contrast to a link) in the mesh network [131]. As mentioned in Section 3.2.1, all nodes are treated equally and have an equal chance to be the SN, DN or IN in this scenario.

A leg of the transmission is defined as a point-to-point link between SN to the

IN or any IN to the next IN or the last IN to the DN. We attach a visited nodes variable in the header of the packet to keep track of the path that the transmission has taken to avoid looping. When moving forward from the SN to DN, every IN will be added to the visited nodes variable one at a time. The visited nodes in the header will be removed from the possible NN list in the routing table when one node decides the next node. We apply the same loop-detecting process in every leg of the transmission. If any IN cannot find any possible NN to the DN due to no possible NN have power or all possible NNs have been removed during the loop-detecting process, then this leg will be deemed unsuccessful.

We then roll back the transmission to the previous IN (or SN) and try another possible NN. We count the times of the retry (NR) and if the NR reaches a preset maximum number (NR_{max}) or all possible NNs from the SN has been tried before NR reaches NR_{max}, we then consider the entire transmission unsuccessful. A successful transmission means the packet reaches DN from SN with NR less than NR_{max}. After the transmission finished, we then can evaluate the total cost C_{total} of the entire transmission from SN to DN using (3.7):

$$C_{\text{total}} = \sum_{i=1}^{n} C_i \tag{3.7}$$

where C_i is the cost of the i^{th} individual point-to-point link of the entire transmission, where C_1 is the initial transmission from the SN to the IN_1 and C_n is the final leg of the transmission, which is from the IN_n to the DN. C_{total} is the cost of the entire route of the transmission. Then we can work out the Path Quality *(PQ)* value of this transmission which will use the reinforcement learning algorithm to update the routing table of each node in the path after the feedback from the DN to SN. The PQ value is calculated with a success bonus *(SB)* value depending on the success of the transmission using (3.8a):

$$PQ = SB - C_{\text{total}} \tag{3.8a}$$

$$SB = \begin{cases} SBV, & if\ transmission\ is\ successful \\ 0, & if\ transmission\ is\ unsuccessful \end{cases} \tag{3.8b}$$

The two-part PQ equation (3.8a) calculates the PQ value using the total route cost C_{total} as well as the SB If the transmission is successful a configurable pre-defined positive SB value (SBV) will be added to a successful transmission to compensate the total cost values. Zero SB will be given to the unsuccessful transmissions, and in this case, PQ will be negative to discourage the routing using the same path in the future.

3.3 Packet header and feedback

The only added parameter in the header of the IP packet [130] is the visited nodes variable. It can be added in the start of the data payload of each packet to keep the compatibility of the IP routing protocols used outside the mesh network. With minimum addition to every packet, the negative impact of mesh routing is minimised. As discussed in this chapter, the added header contains the nodes information are very small as each node in the visited nodes list can be represented as a number, the total overhead of such addition information can be only several bytes in size. Considering the IP packet carrying the sensor data are usually much bigger than this as the data, we decide to neglect the impact of such minor addition to the packet.

The feedback of the transmission is propagated from the DN back to the SN using the reverse order of INs along the route if the transmission is successful. The feedback information is the PQ value from the IN to the DN. The PQ value of that part of the transmission is calculated as if the transmission is from that IN to the DN. When the transmission is unsuccessful, the feedback with negative PQ value will be propagated from the node of failure back towards the SN in the same fashion. Finally, after each node in the transmission receives the feedback, it updates the RM value of the DN on the row. The actual NN used in the transmission in its routing table using reinforcement learning method will be described in Chapter 4.

3.4 Summary

In this chapter, we have modelled the crucial elements of the remote monitoring wireless mesh network. This includes the radio channel it uses; the sensor nodes work in the network, the links between sensor nodes, the routes of each packet travels and transmissions of the operation of the network. We also defined the calculation of some of the most important parameters that are used in the algorithm, such as $SINR$, Pt, CC and C. In addition, we discussed how the information of each transmission is stored and propagate throughout the operation of the network by defining the header of each packet and the feedback. With all these parameters and information in place, we can now focus on the reinforcement learning part of the algorithm, which makes this algorithm stand out from other routing algorithms.

Chapter 4 Reinforcement Learning in the Network

The updating of the routing table of the previous routing protocols for WMSNs is usually done by altering the next node after receiving the updated information from the link state messages such as periodical HELLO messages in the OLSR [46]. Another widely used mesh routing protocol, The Better Approach To Mobile Ad hoc Networking (B.A.T.M.A.N.) uses Transmission Quality (TQ) to determine route selection. As the authors explained in [132], TQ makes B.A.T.M.A.N. quality aware. However, the update of the TQ still uses a traditional way of overwriting, and the route selection is also by selecting the maximum TQ. This may produce a better result when the information of the network is present at the node, but the lack of adaptability from learning the unknown part of the network still renders it suboptimal when deployed in the ever-changing environment of the remote monitor networks.

As mentioned in Section 1.1, the environment of remote monitoring networks can be hard to predict. Not only are the available locales for the deployment of sensor nodes but also the changes of the power level of each node due to the weather and transmission over time. The algorithm needs the ability to adapt to the newest changes in the network as well as has the ability to use the previous knowledge of the network to make the best routing decision, in order to maximise the usability of the network. This requires a 'memory' of each node to keep track of what has happened over time and has the ability to absorb new information gathered from the feedback of each transmission. However, with the limited ability of the computational power and memory available at each sensor node, a large database is not feasible to be stored within the node. As discussed in Section 1.6, reinforcement learning is a good choice to be employed in the algorithm to enable such ability.

4.1 Temporal difference learning

In this thesis, we employed a temporal difference (TD) based learning method to conduct updating the routing metrics in the routing table of each node when a transmission finished and received the feedback from the terminal node (DN when the transmission is successful, or the last IN when it is unsuccessful). We also use the Boltzmann exploration method to balance exploration and exploitation when selecting the node based on the TD learning calculated routing metrics as, explained in Section 3.2.2.

TD learning is one of the most common reinforcement learning methods. It uses the differences between the expected value and the actual outcome value of a certain action during the process over successive steps to update the agent progressively [55]. This difference is called the temporal difference. The learning process is to gradually reduce the TD by setting appreciate parameter to reflect the learning process. TD learning does not require a training data set to train the machine learning algorithm like unsupervised learning. Instead, the learning agent, in this case, the sensor node, uses the result of a sequence of states to learn to predict the expected value of a variable occurring. This learnt value will also influence the environment, called the state, it is detailed in [133].

TD learning is an appropriate method to support the routing requirements in the remote monitoring networks as the best routing decisions needs to be predicted from the feedback of the previous transmissions while the decision will impact the energy level of each node along the route. Besides, by only storing the latest prediction value and update every time using the feedback of the transmission, this learning method requires a minimum extra storage as only two parameters are added to the routing table of sensor nodes. Apart from the RM value, another parameter used in the routing table of this algorithm is a time of visited variable storing the times that the route has been selected. This variable is used to confirm that all the possible NNs to the DN are tried before the calculation of the new RM value. So long as one or more of the variables for all the NNs to DN is 0, an equal chance random process will select the NN, instead of using the Boltzmann exploration. This is to make sure the learning is not biased towards the first selected NN. Thus, it saves both computational power and storage of the nodes that are also crucial for the usability of the network. (4.1) shows how the algorithm calculates a new RM value based on the previous prediction and the feedback from the transmission:

$$RM^*(NN) = RM(NN) + \beta(PQ + \gamma RM(NN') - RM(NN)) \quad (4.1)$$

In (4.1), $RM^*(NN)$ denotes the updated new RM value at the node for selecting the route toward DN via node NN; $RM(NN)$ represents the current RM for the same selection; PQ is the path quality value obtained from the feedback of the current transmission. It represents the actual outcome of the action of selecting the current route. $RM(NN')$ represents the expected RM value of the selected NN of the route. The $RM(NN')$ is calculated by averaging all the RM values of all the possible NNs in this leg of transmission. This represents the expected value of the action of selecting the route. We then use some control parameters to control the process of learning, where γ denotes the discount rate and β is the learning rate. We call the $PQ + \gamma RM(NN')$ part of the (4.1) the TD target. The TD target indicates the precision of how the algorithm has learnt the network. The smaller the value of the TD target gets, the more precise the learning algorithm has prediction the best route of the transmission.

4.2 Parameters of learning process

As can be seen in (4.1), there are two parameters that determine the outcome of the value the TD learning process. They are the updating discount rate γ, and the learning rate β. In this thesis, we also discuss another important parameter in the routing decision process, which is the exploring hyper-parameter τ. We use τ to control the balance between exploiting the learnt knowledge of the network and exploring the possibility of using other routes. By controlling this process, we can make sure that all the possible routes are considered over time. All 3 of these parameters have different impacts on how the reinforcement learning progresses over time and further influence the performance of the network.

4.2.1 Exploring hyper-parameter τ

As seen in (3.5), we used the Boltzmann exploration to balance the process of exploring and exploiting. The exploring hyper-parameter τ is used to control how the output the (3.5) by converting the RM values into the likeliness of the selection. The Boltzmann exploring process uses the spread of the SoftMax distribution of the output of the equation to conduct this purpose [55]. This is achieved by applying a SoftMax function to normalise the values and compress them into the values summed 1. [134] With a higher τ value ($\tau \to \infty$), all possibility tends to be the same, just like a random process, whereas with a lower τ value ($\tau \to 0$), the likeliness tends to reflect the real difference between the possible selection. Especially when τ = 0, the highest likeliness selection will have a chance of 1. By controlling the τ value in (3.5), we can adjust the balance between exploring new or low RM value nodes or exploiting the best-known node.

4.2.2 Discount Factor γ

The discount factor γ is multiplied to the expected RM value in the (4.1). Its value is in the range of $\gamma \in [0, 1)$. This is because if γ is greater than 1, the output values will diverge as a large expected value will be added to the actual value get from feedback. By directly acting with the expected RM value, γ is used to control the importance of future rewards versus the immediate ones [135], which exist in the form of PQ value. A larger γ value will result in a greater emphasis on the expected future value. Whereas a smaller γ value will lead the algorithm considering heavily on current immediate value and may ignore the future impacts of the selection. The balance of γ value will also change the balance of the performance of the algorithm and a larger value is expected as the purpose of the algorithm in the long-term usability of the network.

4.2.3 Learning rate β

The learning rate β is also known as the step size. β is multiplied to the new information calculated from the new PQ value as well as the temporal difference. It alters the extent of the new information added onto the current information. Hence, it determines the speed of learning. When β has a value of 0, the algorithm learns nothing as it keeps using the old value and ignores the newly learnt information. A small β value results in a slow learning process or may even permanently stuck if a high training error is associated. A high β value makes the algorithm emphasises on the new information but when the β value is too high, it may result in numerical overflow as the knowledge rapidly moves away from the origin [134]. An optimum β value paired with the appropriate balance between exploration and exploitation will make the learning algorithm most efficient.

4.3 The routing decision process

The routing decision process of the algorithm is illustrated in the flowchart in Figure 4.1. All the routing process happens with in one single node in the network. This local decision process makes the algorithm totally distributed. With this nature of the algorithm, each individual node learns the network individually and when one of more nodes becomes offline, the rest of the nodes in the network still have their own information about the network and will eventually learn the changes that happened in the network. This eliminated not only the problem of having a single point of failure in the star networks but also makes the network more adaptive.

The process begins from the SN choosing an NN from the routing table using the Boltzmann exploration described in (3.5). Then the SN will be added to the visited variable in the packet header. If the chosen NN is the DN, the transmission is successful as the data has been delivered to the DN, and the feedback and updating process will occur as described in section 3.3. Otherwise, all the nodes in the visited variable in the header will be removed from the routing table of NN before it selects its next NN. The process will continue repeating itself with new

NNs until the DN has been selected or when one of the NN's routing tables has no available nodes to select. When there are no more available nodes to be selected as the new NN, the algorithm will calculate the number of retries (NR) and compare it to the maximum allowed retry count (NR_{max}). If NR is smaller than NR_{max}, the algorithm will go back to the previous node and retry the process and 1 will be added to NR. If the NR is equal to or larger than NR_{max}, the routing will be considered failed. Once the routing process is finished, regardless of success or failure, feedback with the value calculated in (3.8a) will be transmitted to the nodes involved during the process.

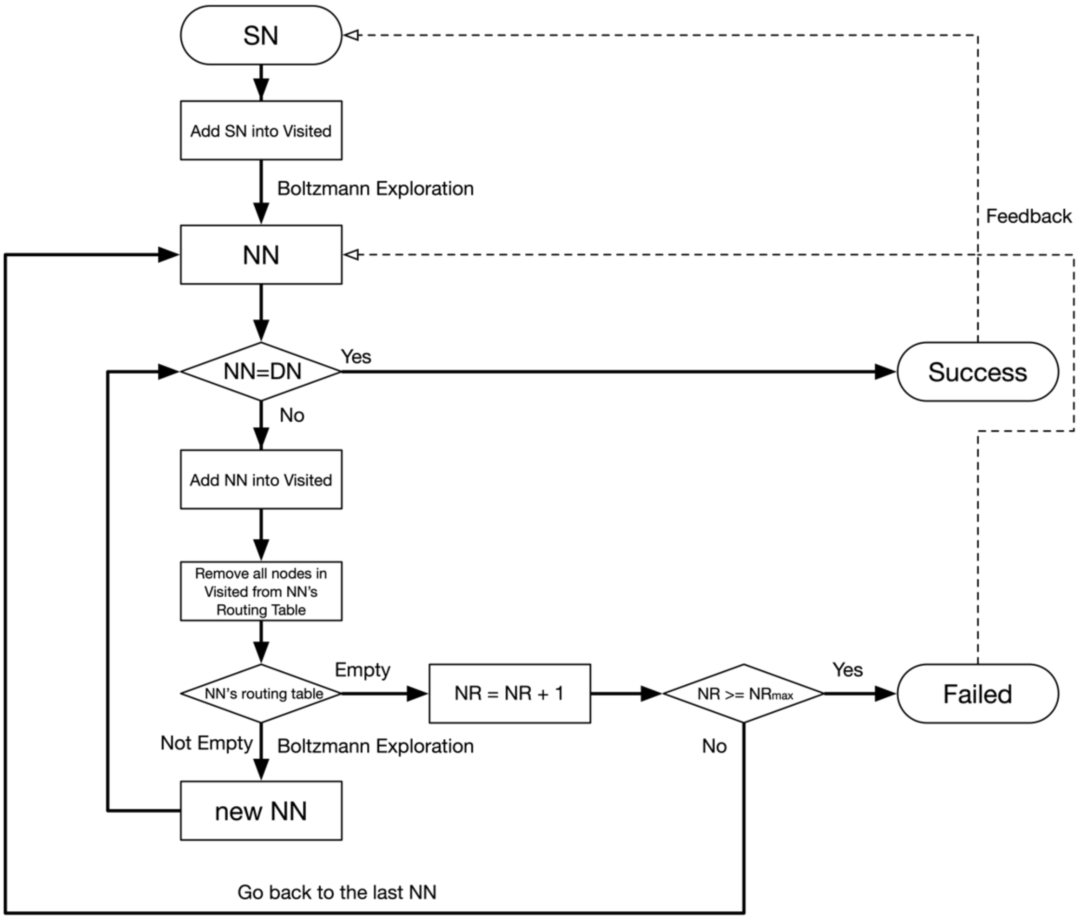


Figure 4.1 RL routing algorithm decision process

4.4 Summary

In this chapter, we discussed the reinforcement learning method utilised in this algorithm, namely temporal difference learning. We discussed several key variables play important roles in the process of conducting the reinforcement learning algorithm, such as exploring hyper-parameter, discount rate and learning rate. Furthermore, we have also discussed the implementation of reinforcement learning process in the algorithm and the workflow of the routing decision using the updated routing table.

Chapter 5 Network Simulations

To evaluate the performance of the reinforcement learning routing algorithm, we established a series of simulations with different learning variables to assess the performance of the reinforcement learning routing algorithm against the two benchmark routing algorithms.

First, we established a simulation environment that is based on that of the remote wireless mesh monitoring network model discussed in the Chapter 3. We started by defining the characteristics of the channel used in the simulations. When determining the path loss exponent α, we considered the network as a far-field communication scenario. Given that the network is being remotely deployed, the typical α value between of 2.7 to 3.5 was selected for describing the open area mobile radio environment [136] [76]. In the simulations in this thesis, we assumed the empirical α value of 2.8 for the environment of the remote monitoring network's low multi-path channel as describe in the Section 2.2.1. Similarly, the network is likely to have little fluctuation due to the infrequent human activities in the place where it is operational, we also assumed a fixed channel gain h of 2. The ambient noise (N) of the channel was assumed at the level of -130 dBm for the same reason. Besides, having a clear radio spectrum means that the adjacent-channel interference (ACI) does not exist, and the only interference can happen is co-channel interference (CCI) when two or more transmissions arrived at the same node at the same time. We can also utilise non-overlapping channels for simultaneous transmissions, this assumption is technically feasible. We also understand that these assumptions may not best describe the practical dynamic communication environment, therefore further detailed study of these specific parameters will be performed in the future work. In this thesis, we focus more on the failure rate, energy efficiencies and carrier band usage rate (CBUR) serving as the foundation for the future work.

We set up a series of simulations to compare the performance of the reinforcement learning routing algorithm against the lower bound benchmark, i.e.

the random routing algorithm and the upper bound benchmark, i.e. the centralised shortest path first routing algorithm. The computer we used for the simulation has been listed in Table 5.1.

Table 5.1 The computer used for the simulations in this thesis

Computer Model	iMac (27-inch, Mid 2011)
CPU	Intel® Core™ i5-2400 (4 Cores, @3.10 GHz)
RAM	16GB DDR3 @ 1333MHz
Graphics	AMD Radeon HD 6970M with 1GB GDDR5
Storage	128GB SSD + 2TB HDD
OS	Mac OS 10.13.5 High Sierra
MATLAB	MATLAB 2019a

For each parameter we investigated, we ran the simulations over 3 different scales of networks with 7, 20, 50 nodes within an area of 20*20 km^2 space to represent different scales of networks on the computer with the specification listed in Table 5.1. The choice of such area and scale was considering the accessibility of the internet from either cable or mobile networks in remote areas so that the data collected by the nodes can be uploaded. The 3 different scales of nodes represented low, medium, and high density of node coverage in that area. These 3 densities can be considered small, medium and large scale of the network, as the complexity of the network increases as more nodes are involved. Each simulation consisted of 52560 timeslots of 10 minutes each to simulate the one-year total operational time of the network. We executed each simulation of each scale of network for 5 times and averaged the results of these simulations for better precision. The transmissions happen randomly, with a 20% chance to initiate in any timeslot. Each timeslot was capped with initiating 3 maximum transmissions. This made a total approximate number of 31,500 transmissions in each simulation. The difference the maximum transmission range of each node in the simulation was studied in Section 6.3. However, in all other simulations, we fixed the range to 10km. We also studied the performance of the network under different charging cycles to reflect different weather conditions. In all other simulations, a fixed charging cycle of 5 days (720 timeslots) is implemented.

Regarding the fixed parameters about the node, we assumed that the battery size of each node (P_{max}) was 15 Wh. We chose this value based on the constraint of the size of the sensor node. The battery capacity of a smartphone with good a battery life, such as Huawei Mate 20 Pro [137], was used as a reference here. That smart phone has a typical battery capacity of 4200 mAh, operating at 3.7V, the energy it stores can be calculated by 4200mAh*3.7V = 15.54Wh.

We also chose the transmission bandwidth (BW) to be 125 kHz as it is the minimum transmission bandwidth of LoRa to reflect the extreme case. We also chose a fixed transmission rate (R) of 5 kbps for all the transmissions in the simulations. The choice of this transmission rate is in line with the LoRa network with spread factor 7 in 125 kHz bandwidth [18]. In order to avoid looping and infinite retries when finding routes, we also have defined the maximum number of retry $\mathrm{NR}_{\max}$ = 10 in all simulations. We also fixed the weight (W_1, W_2 and W_3) used in the cost function (3.6) to emphasize on the transmitting power first and then the remaining power of the receiver, before the power of the sending node. This will lead the algorithm to pick a route with less chance of failure, as the receiver has more power left for forwarding the packet on. We used these numbers based on our previous trials on the algorithm. All the empirical fixed parameters used in the simulation can be found in Table 5.2:

Table 5.2 Fixed parameters used in the simulation

BW (kHz)	N (dBm)	α	h	I	R (kbps)
125	-130	2.8	2	0	5
P_{max} (Wh)	NR_{max}	W_1	W_2	W_3	
15	10	1	0.1	0.3	

5.1 Comparison with the random routing method

Currently, one of the most common routing algorithms for wireless networks is Optimized Link State Routing (OLSR). OLSR uses Multipoint relays (MPRs) to relay messages between nodes [138]. MPRs are updated with the 2-hop neighbour information when receiving HELLO messages of the OLSR protocol periodically from other nodes. The HELLO messages with link-state information are usually flooded when the routing table is initialised, or new nodes have been added into the network. MPRs select intermediate nodes in the routes depending on the source. The selection usually is performed by selecting a random neighbour to pass the information onwards.

In this thesis, we used a random route selection process in the simulation to emulate the similar behaviour in the OLSR without the periodical HELLO message to update the routing table. This random routing algorithm can be considered as a simplified version of OLSR for the IoT network studied in this thesis. In each transmission, the SN and INs were picked a random next node from the routing table with equal probability. The visited variable in the header and the NR_{max} used in the reinforcement learning algorithm has also been applied to the random routing to avoid looping and too many retries. This created a fair comparison with the reinforcement learning algorithm.

This random routing was considered as a lower bound benchmark for the comparison because the OLSR it based is considered being the traditional wireless ad hoc routing algorithm. Hence, the comparison between the random routing and reinforcement learning based methods can demonstrate how the information from the feedback of previous route selection impact on the efficiency of the future node selections. In short, we considered the random routing algorithm as a reasonable lower bound benchmark for the performance of the reinforcement learning based algorithm.

5.2 Comparison with the centralised SPF methods

We used a centralised shortest path first (CSPF) routing as the upper bound benchmark in comparison with the proposed reinforcement learning based routing algorithm. Being a centralised method, this CSPF assumes all the information of the network is known to all nodes at all time, and the routing is done by using Dijkstra’s algorithm to calculate the shortest path from any SN to any DN using the same cost function as the reinforcement learning based routing. The CSPF method preloaded the best route every time the transmission is initialised and is expected to show a great margin when compared to the reinforcement learning based routing as well as the random routing. The CSPF is an ideal method, as it requires prior knowledge of the topology of the network in order to generate the routing table for each node. The updating of the routing tables can also be complicated for the low power consumption microprocessors in the sensor nodes because all the most effective routes need to be recalculated and updated to keep the network efficient as often as the network changes. Otherwise, nodes can be overloaded and get out of service due to battery drought if no rerouting is performed. This iterative updating of the routing tables of every node in the network will incur significant time and energy consumption.

Table 5.3 shows the comparison between initialising and update a routing table for reinforcement learning and updating the CSPF routing table by regenerate the entire table as mentioned in this section. Figure 5.1 demonstrates the time required to perform a single action of each method of 3 different scales of networks with 7, 20, 50 nodes respectively. With a longer computational time, it requires more energy for the microprocessor to perform the action, therefore, more gross energy will be consumed in the network. Though the differences between each action shown in Table 5.3 were not significant, as the total time needed was less than 1s. However, considering the CSPF routing need to update all the nodes in the network every time the transmission finished, while the RL routing just update the nodes along the route, the difference of total computational time required during the one-year operation simulation will be magnified with the number of transmissions and the number of nodes. We plotted this comparison in Figure 5.1 to demonstrate the vast difference in each scale of the network.

Table 5.3 Time of initialization and updating routing table for RL algorithms and generating SPF routing tables.

Scale of the network	Initializing a routing table for RL algorithm(s)	Update an entry for RL algorithm (s)	Regenerate a CSPF routing table (s)
7 nodes	0.033395	0.020031	0.068354
20 nodes	0.036594	0.018697	0.135501
50 nodes	0.078826	0.019745	0.789133

In the simulation of circa 30,000 transmissions we used in this thesis, with an assumed average number of links of each route of 10 in the 50-node network, the CSPF requires 1.183 million seconds of total update time as opposed to less than 6 thousand seconds spent using reinforcement learning method. This difference is so significant that renders the CSPF impractical to be used in real life for the remote monitoring WMSNs even it can give the absolute best route every single time.

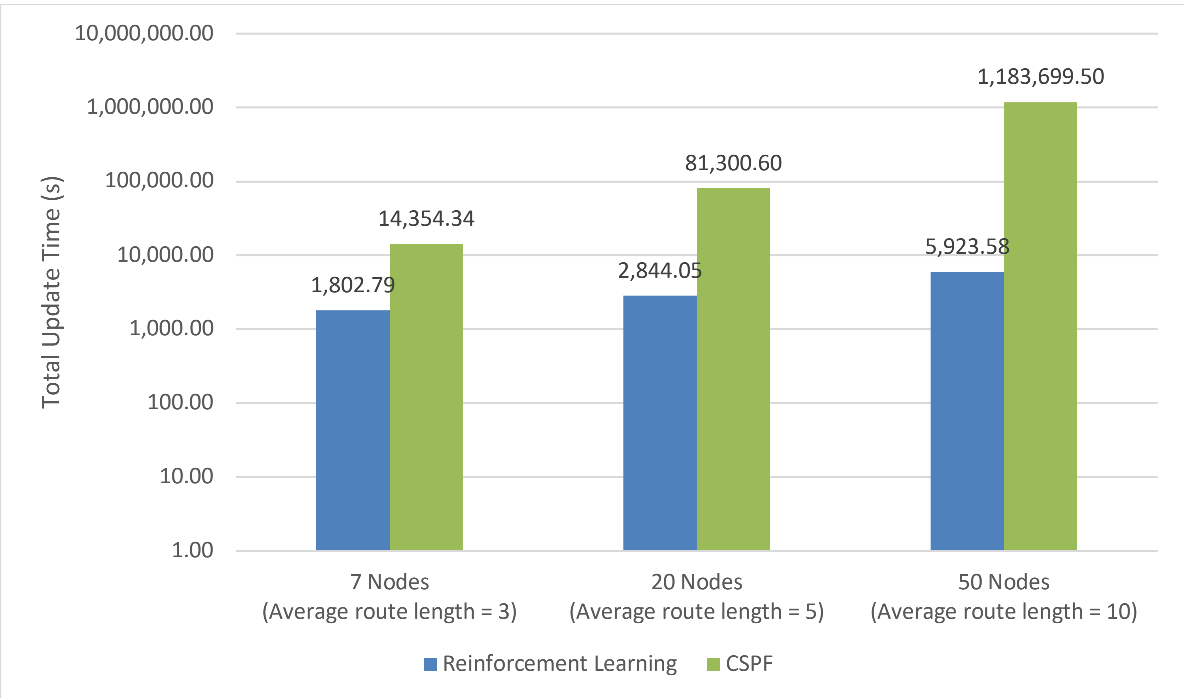


Figure 5.1 Average total computational time required for updating the routing tables (in logarithmic scale)

Despite the impossibility of using the CSPF method in the real IoT scenarios, we still used this method as the upper bound benchmark in the simulations as it represents the ideal case performance of the network can reach when all the constraints are removed from the physical layer.

The comparison between the reinforcement learning routing algorithm and the benchmarks over different scales network showed the strength and limitations of the algorithm in different situations of the IoT network. It also demonstrated how the configuration of the algorithm impacts network performance.

5.3 Summary

In this chapter, we set the scene for the simulations to prove the usefulness of the algorithm. We first listed all the parameters that needs to be used in the simulation, we then described the need for comparison between the reinforcement learning routing algorithm and the two benchmark algorithms, namely the Random Routing and Central Shortest Path First algorithms. Finally, we discussed the reason that the CSPF algorithm may have good results, but it is unrealistic to be used real world.

Chapter 6 Results and Discussion

In this chapter, we looked into several parameters in the reinforcement learning routing algorithm that might have an impact on the performance of the network under different circumstances. These parameters covered the learning process, node variations as well as network topologies. We compared the results of a series of simulations of reinforcement learning routing algorithm with different parameters against both the CSPF and the random routing benchmarks to establish the potential of the reinforcement learning routing algorithm.

We investigated the result of the average failure rate (%), the average carrier band usage rate (CBUR) (bit/Hz) and average energy efficiency (bit/kJ) of each set of simulations with difference scales of the network. The reason for the selection of these result outputs is that they present different aspects of how the algorithm is performing against benchmark algorithms. The failure rate shows how reliable the transmission is at the network level. The CBUR indicates how the transmission occupies the spectrum over sending data. This shows the spectral efficiency of the algorithm over a long period of time, and how much airtime it is required in the process. Finally, the energy efficiency shows how much energy needs to transmit a bit of data. As the total data transmitted in the network is fixed, the higher the energy efficiency, the longer the network lasts. Hence, the network is more sustainable.

The failure rate (λ_{fail}) is defined as (6.1):

$$\lambda_{fail} = \frac{N_{failed}}{N_{total}} \tag{6.1}$$

where N_{failed} represents the number of failed transmissions and N_{total} is the total number of transmissions. As this is the network level failure rate, unlike the failure rate in upper level in the stack, there will be no re-transmission on the network level, and it depends on the radio environment of the network. We consider 10%

for the large network is reasonable. For the smaller networks, we take 5% as the benchmark.

We also define the CBUR $\eta_{carrier}$ and energy efficiency η_{energy} in (6.2) and (6.3), respectively:

$$\eta_{carrier} = \frac{D_{\text{transmitted}}}{BW * NC_{\text{used}}} \tag{6.2}$$

$$\eta_{energy} = \frac{D_{\text{transmitted}}}{E_{\text{total}}} \tag{6.3}$$

where $D_{transmitted}$ is the total amount of data transmitted from the SNs to DNs of all successful transmissions during an entire simulation, NC_{used} represents the total number of used carriers, and E_{total} is the total energy consumed in the simulations for calculating the energy efficiency η_{energy}. E_{total} is evaluated by summing up the consumed energy of each node during each charging cycle, as shown in (6.4).

$$E_{\text{total}} = \sum_{0}^{N_{\text{charging cycle}}} \left(\sum_{0}^{N_{\text{nodes}}} (E_{\max} - E_{\text{left}}) \right) \tag{6.4}$$

where $N_{charging\ cycle}$ stands for the number of charging cycles has been passed since the start of the simulation, and N_{nodes} denotes the total number of nodes in the network in the simulation.

As we defined all the result outputs we investigated in this chapter, we can use them to investigate the impact of each parameter used in the network on its performance.

6.1 The impact of the network parameters

As we described in Section 3.2.2, the reinforcement learning routing algorithm uses the Boltzmann exploration to balance exploration and exploitation of the results of the learning process to balance the need for taking advantage of current knowledge of the network and explore for new routes in case of link failure. The Boltzmann exploration process that generates the possibility of selecting a particular next node $p(NN_n)$ was calculated using (3.5). Here we list it again as (6.5):

$$p(NN_n) = \frac{e^{RM(NN_n)/\tau}}{\sum_{i=1}^{n} e^{RM(NN_i)/\tau}} \tag{6.5}$$

During the learning process of the algorithm, we used the TD-learning based method to calculate the new routing metric (RM) value of each node towards the DN using the path quality (PQ) value obtained from the network feedback as (4.1) Here we list it again as (6.6):

$$RM^*(NN) = RM(NN) + \beta(PQ + \gamma RM(NN') - RM(NN)) \tag{6.6}$$

where $RM^*(NN)$ denotes the updated new RM; $RM(NN)$ represents the current RM; $RM(NN')$ represents the expected RM value of the selected NN of the route.

The parameter in the reinforcement learning we studied in this thesis included the Boltzmann exploration hyper-parameter (τ), the discount factor (γ) and the learning rate (β). We then established an optimal learning configuration of the algorithm for remote wireless mesh sensor network (WMSN) scenarios to evaluate the optimal performance of the reinforcement learning routing algorithm against the benchmarks.

6.1.1 Boltzmann exploration hyper-parameter (τ)

We used the exact same method to randomly generate networks for the simulations of each scale. The number of transmissions for the small-scale 7-node networks was 31611, for mid-sized 20-node networks was 31497, where for the large-scale 50-node networks was 31605. These simulations were also performed with 5 different τ values (0.1, 0.2, 0.5, 0.8 and 1).

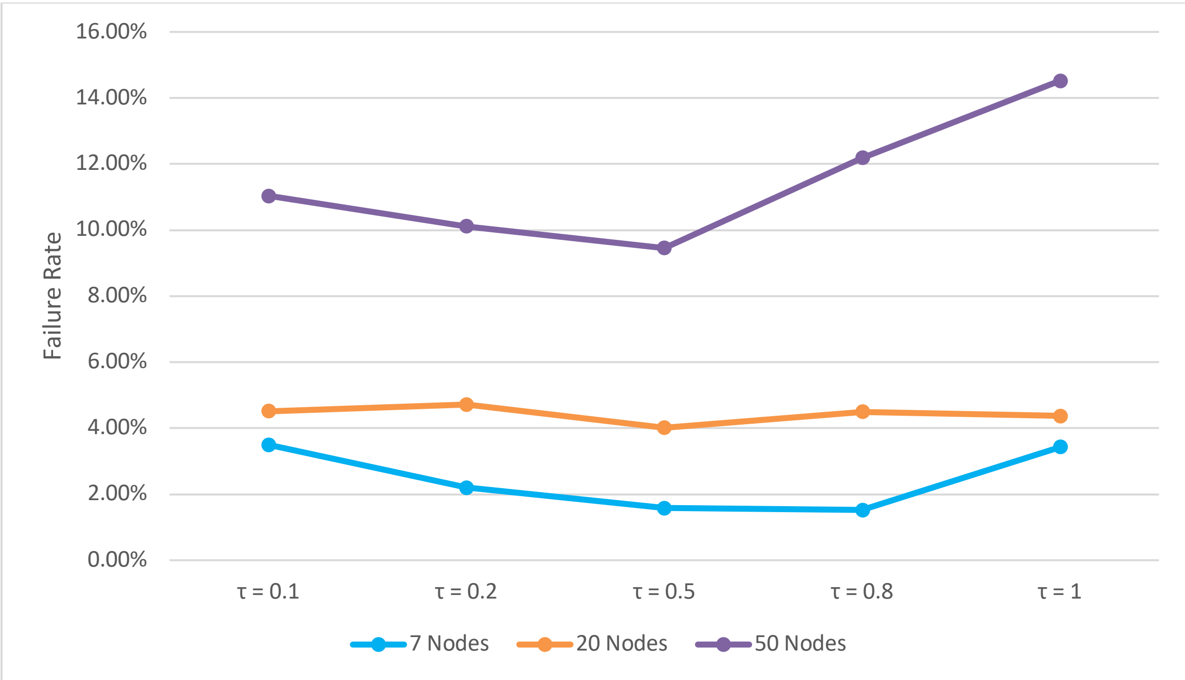


Figure 6.1 Average failure rates of simulations with different τ values

Figure 6.1 shows the results of the average failure rate using the same set of topologies and transmission settings with different Boltzmann exploration hyper-parameter (τ) values. We averaged all the failure rates results of a series of 5 simulations. As shown in Figure 6.1, the average failure rate of the transmissions was raised as the number of nodes increased. The result of the failure rate of 50-node large scare networks is much higher than the smaller scale networks. This phenomenon was expected as more nodes in the network will result in more legs in each transmission. Hence, there is a higher chance of generating route loops, which leads to failed transmissions. However, among all 3 different scale networks with the values we tested, we found that the optimum τ value was 0.5. It resulted in the lowest average failure rate in all 3 scales of networks. As the τ value controls the spread of possibility, this result can be explained with a small value of τ, the algorithm will be more exploitive of the learnt information, more likely to rely on the same route. This may exhaust the battery of the nodes on that route and is less responsive to the changes in the network. In contrast, a large τ value will even out the different of the likeliness routing metrics stored in the routing table. This will result in a more explorative behaviour, making the route selection more random rather than taking advantage of learnt information about the network. We found that the τ value of 0.5 strikes the best balance both exploration and exploitation in the tested networks with the best result. With the algorithm in place, all the failure rates met the target we set in the beginning part of this chapter, which is lower than 5% for the 7-node and 20-node network and 10% for the 50-node network.

We then compare the reinforcement learning routing algorithm with the τ value of 0.5 against the upper bound CSPF and lower random routing benchmarks in all 3 scales of networks. The results are shown in Figure 6.2.

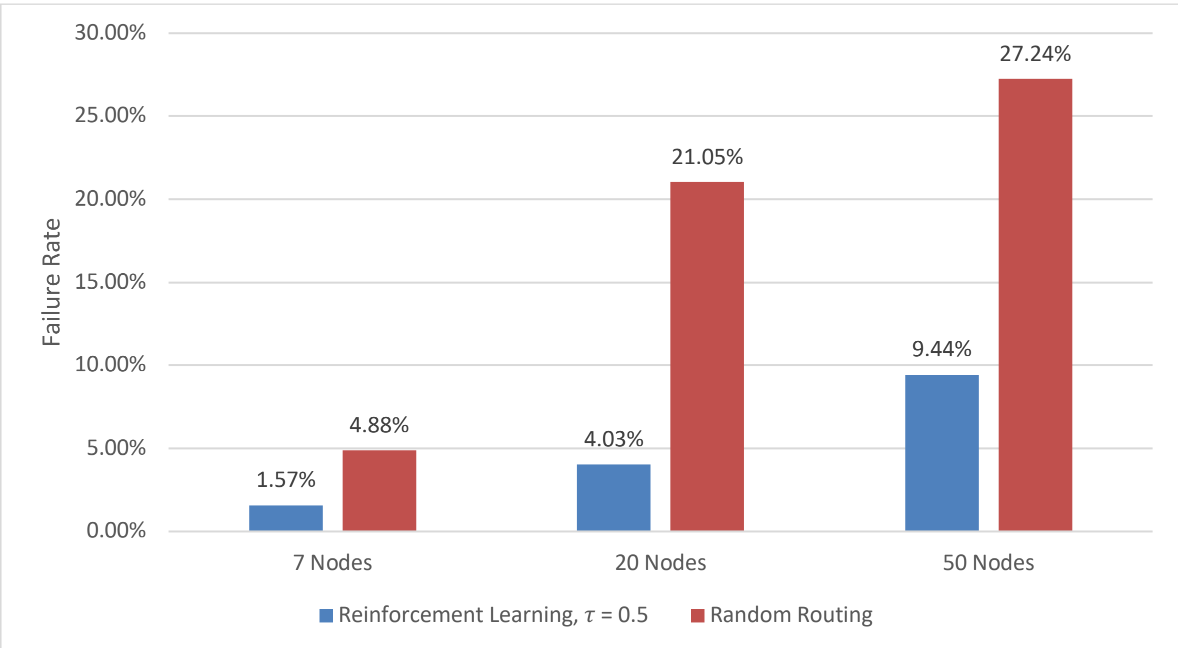


Figure 6.2 Average failure rates of simulations with reinforcement learning, random routing algorithms.

Figure 6.2 shows the comparison of the average failure rate between the RL routing algorithm with the τ value of 0.5 against the random routing algorithm. The result of the CSPF algorithm was omitted here as it scored a constant 0% failure rate at all times. We found that the average failure rate of reinforcement learning routing algorithm is much lower than that of the random routing algorithm in both situations of all 3 scales of networks, only at only around 1/5 for the 20-node networks and 1/3 for the 7-node and 50-node networks. This is because of the learning algorithm use the learnt information to avoid lower powered nodes as the RM values are updated with the cost function (3.6) where the remaining power of both nodes are also taken into account. The possibility of selecting the low powered route is reduced. Especially with a reasonable τ value to balance exploration and exploitation to take advantage of possible alternative routes. Without this knowledge of the network, random routing can only randomly pick the route, resulting in a higher failure rate. The CSPF algorithm eliminated the failed transmissions as it manages the entire network, and the best available route is always utilised. Figure 6.2 also shows the change the RL algorithm brings to the result by making the network meet the failure rate target.

Figure 6.3 shows the result of average energy efficiency of the series of τ values (0.1, 0.2, 0.5, 0.8 and 1) in the same set of simulations, where Figure 6.4 illustrates the comparisons between when the best result of τ = 0.5 from Figure 6.3 and Figure 6.1 are applied to the RL routing algorithm against benchmark algorithms in the simulations of networks with different scales.

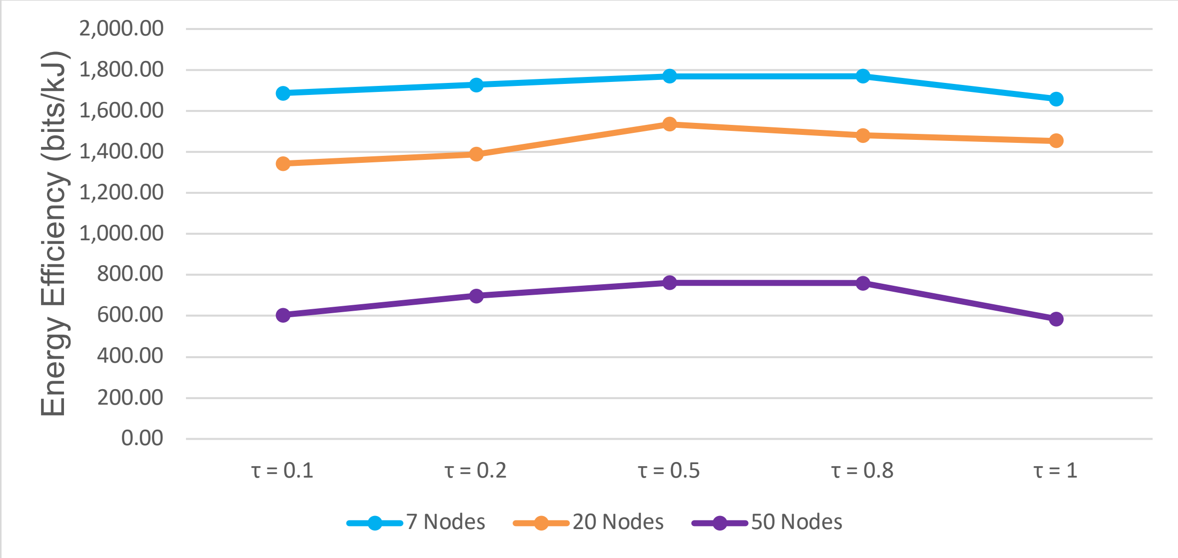


Figure 6.3 Average energy efficiencies of simulations with different τ values.

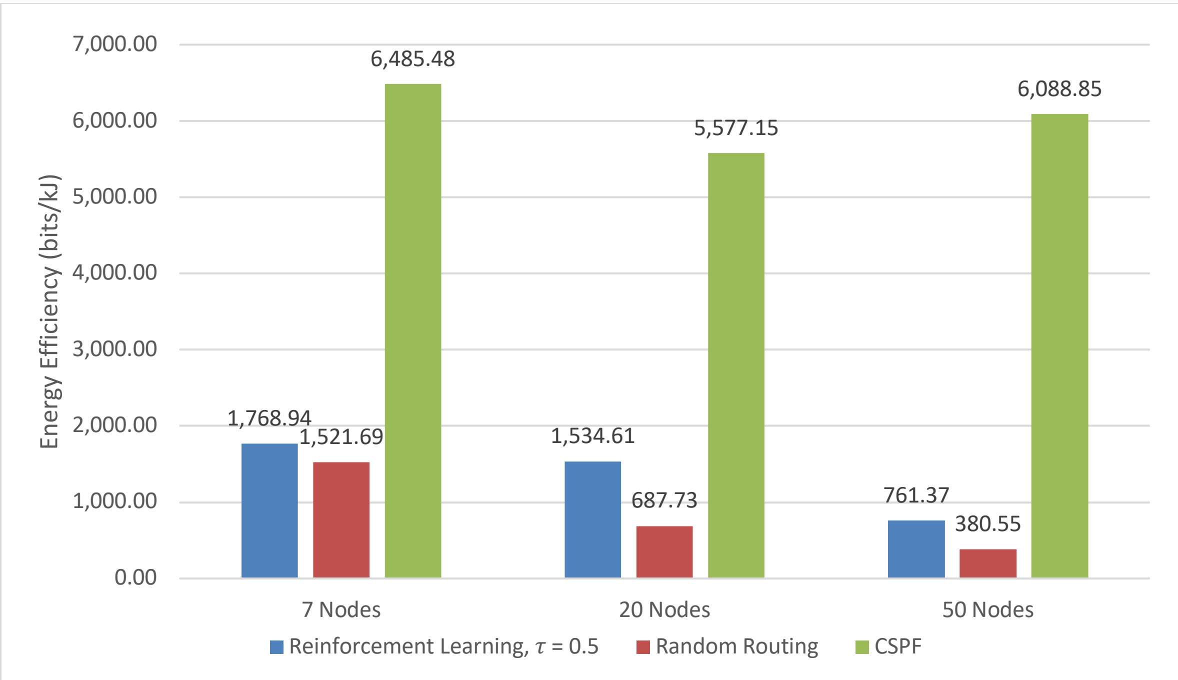


Figure 6.4 Average energy efficiencies of simulations with reinforcement learning, random routing and CSPF algorithms.

The results were similar to what can be found in the average failure rates. The upper bound benchmark CSPF algorithm had a significant advantage due to the ability to pick the best energy-efficient route at all time. This route may usually be the shortest route available and the lack of trial and learn progress from the RL algorithm, hence the much higher efficiency. Whilst the random routing algorithm, being the lower bound benchmark, was trailing behind both other algorithms. The difference between the random routing algorithm and the reinforcement learning routing algorithm was less prominent in the 7-node network. This is due to the less available links between nodes leads to a lack of choice for the routes for the RL routing algorithm to learn from. For the larger scale networks, the reinforcement learning routing algorithm had outperformed the lower benchmark. Among the different τ values in the reinforcement learning routing algorithm, the energy efficiency of both τ = 0.5 and τ = 0.8 were consistently higher than all other τ values. In the 7-node network scenario, the performance of all τ values were quite comparable, the same reason as the lack of choice mentioned when compared with random routing. In the large 50-node network simulation. τ = 0.1 and τ = 1 were much lower than other values as too much exploration may behave like random routing while too much exploitation will also exhaust nodes along a certain route, leading to worse results. Generally, from the result of this simulation the performance of all five simulations of reinforcement learning routing algorithm were better than the random routing benchmark in average energy efficiency.

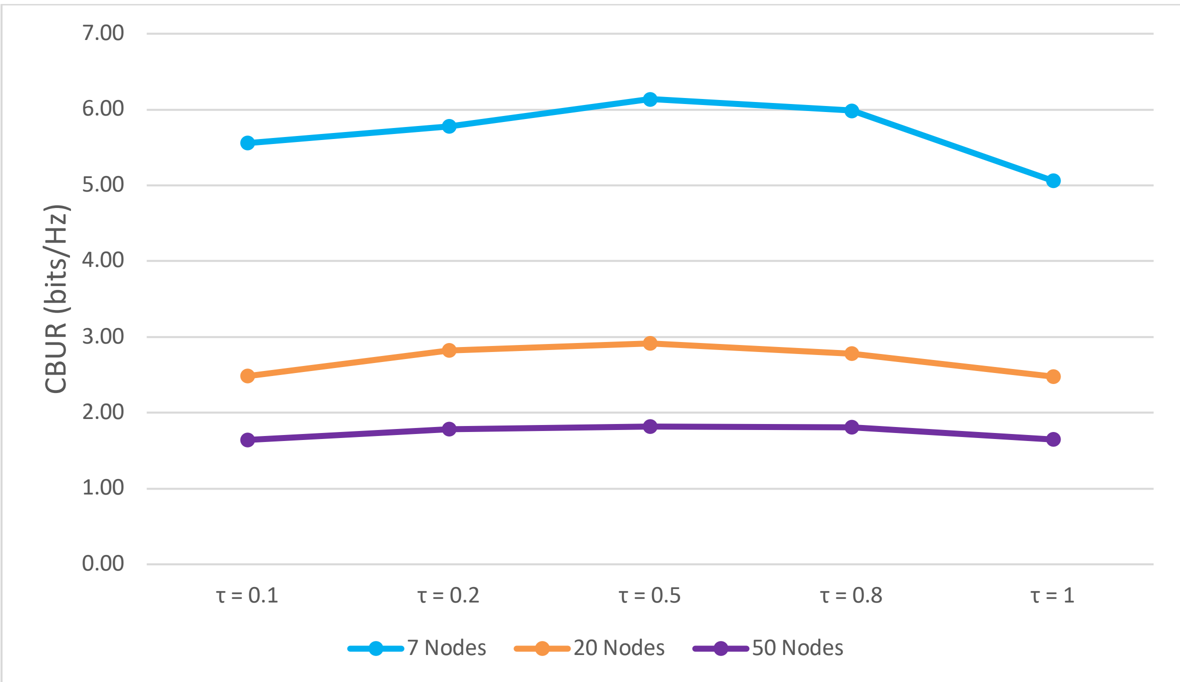


Figure 6.5 Average CBUR of simulations with different τ values.

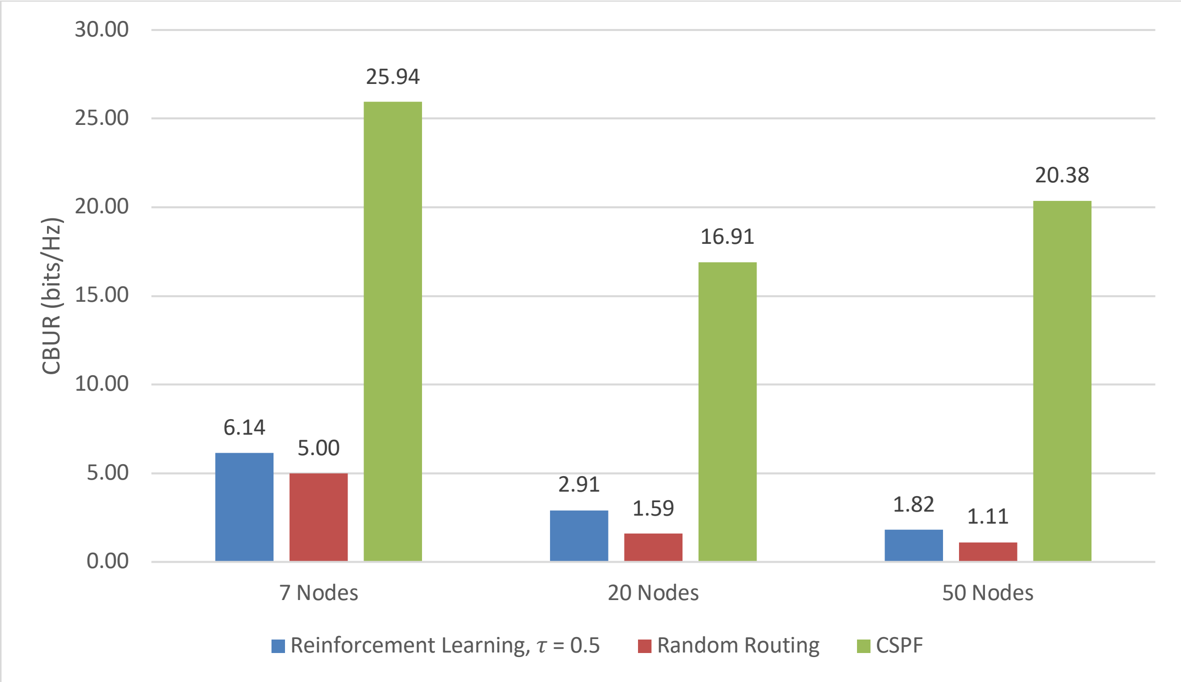


Figure 6.6 Average CBUR of simulations with reinforcement learning, random routing and CSPF algorithms.

Figure 6.5 and Figure 6.6 show the results of the average CBUR of the simulations to illustrate the spectral efficiency of the RL routing algorithm. Despite the great difference between the upper bound method and other methods shown in Figure 6.6, among all the learning methods with different τ values, the results for the τ value of 0.5 are still outperformed other values. However, a less noticeable improvement of the RL routing over the random in average CBUR than average energy efficiencies and average failure rates can be found in Figure 6.6. This can be explained by the focus of the route selection that the RL routing algorithm makes is more towards energy efficiency than spectral efficiency. This might result in the route not being the route with fewest hops, which has the highest CBUR. As the cost function of the RL routing algorithm is considering the impact of energy consumption and longevity of the nodes, the algorithm may choose a longer route to minimise that cost. In the network model, each hop requires a unit of bandwidth. Hence, the CBUR decreased as the longer routes are taken and the number of hops increased. In the network model, each hop requires a unit of bandwidth. This result can be observed in Figure 6.6. However, the CSPF has the knowledge to choose the most efficient route, and that selection is kept updated, therefore, result in the best average CBUR.

In all, the τ value did impact the performance of the reinforcement learning routing algorithms offered in the IoT networks. As found in all results, the exploration and exploitation balanced τ value of 0.5 was the better choice here for all 3 scales of networks, as discussed in Section 4.2.1.

6.1.2 Discount factor (γ)

As mentioned in Section 4.2.2, the discount factor (γ) controls the balance of the use of knowledge of reward from past and future in the reinforcement learning routing algorithm. It is considered as a measure of how far ahead in time the algorithm looks. The value of γ is between 0 and 1.

In this series of simulations, 31516 randomly generated point-to-point transmissions were used in 7-node small-scale networks. 31601 transmissions were used in 20-node mid-size networks, and 31433 transmissions were for the 50-node large networks.

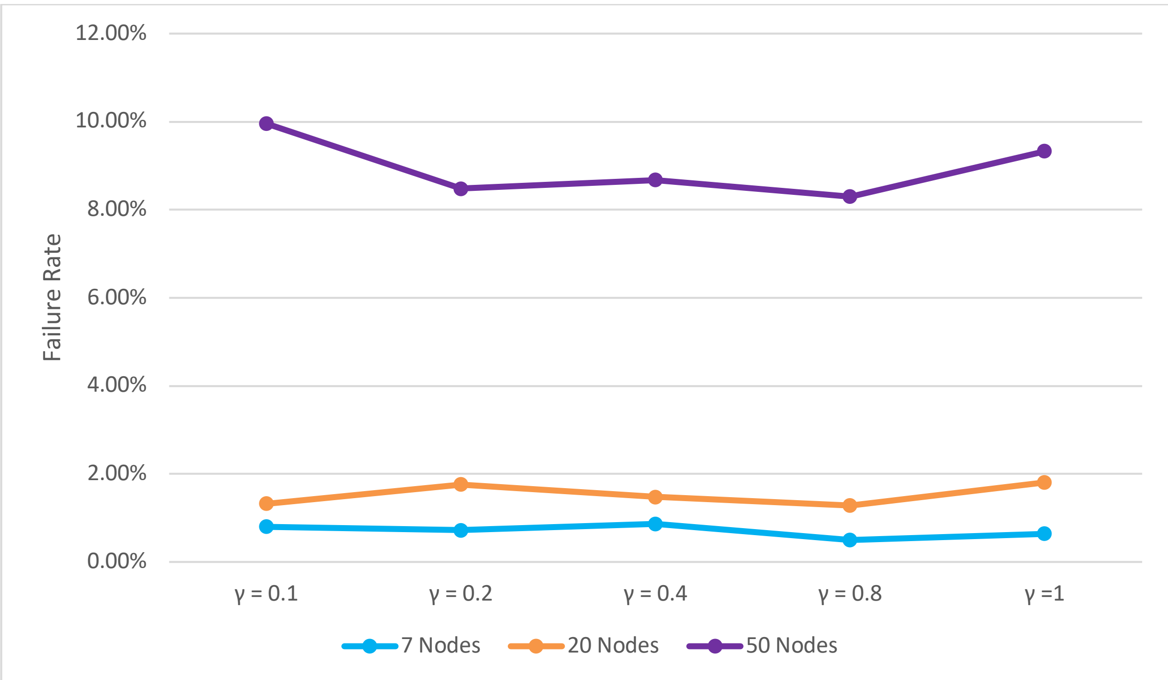


Figure 6.7 Average failure rates of simulations with different γ values

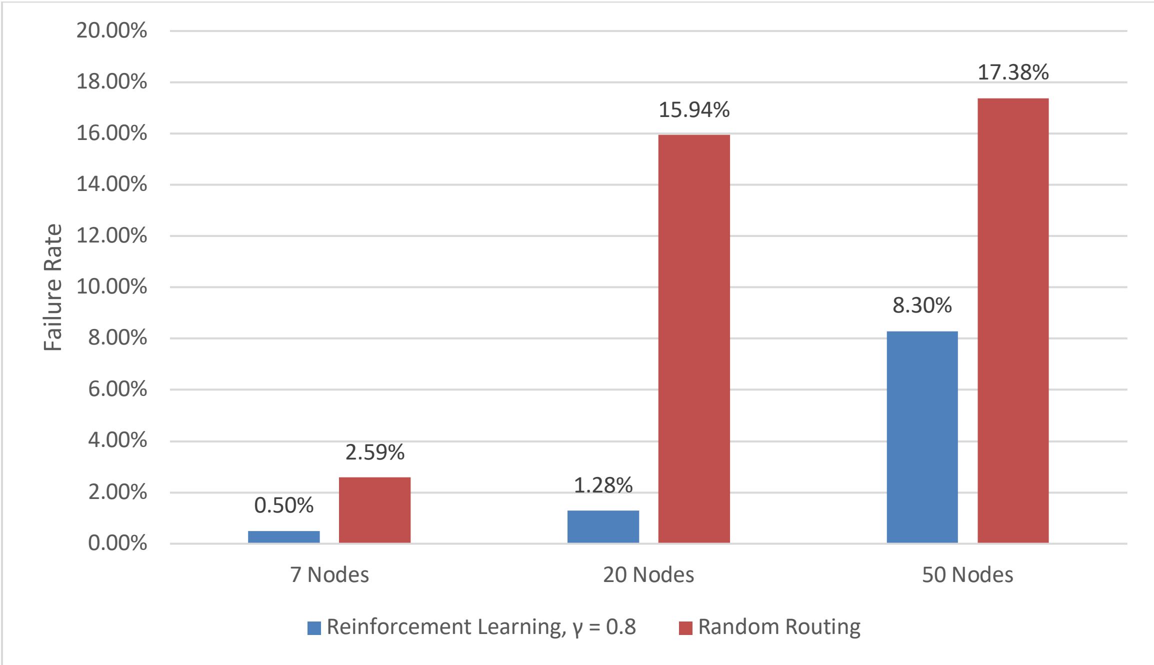


Figure 6.8 Average failure rates of simulations with reinforcement learning, random routing algorithms.

Similar to what can be found in the different τ values, Figure 6.7 and Figure 6.8 show the results of average failure rates related to γ values. With the 0 failed transmission in the CSPF method, the results of reinforcement learning routing algorithm with γ value of 0.8 had a lower failure rate than the random routing in all 3 scales of networks, as shown in Figure 6.8. Between different γ values, in the 7-node network, smaller values of γ posted better results than the larger values as the γ affects how much 'forward thinking' the learning has. A smaller value of γ determines a more 'memory' oriented approach, that means more of past results of feedback of previous transmissions are influencing the selection. This will slow down the progress of learning which will give the algorithm in the smaller networks an advantage of not overloading certain nodes too early due to convergence, consequently the better result shown in Figure 6.7. However, when the scale of network grows, the earlier convergence in the 'forward thinking' of the algorithm with higher γ values performed better. This can be found in the large network simulation as the value of $\gamma = 0.8$ presented a lower failure rate over the other configurations in 20- and 50-node networks. However, with a γ value of 1, the failure rate rises as too early convergence will cause more static route, leading to more failure caused by overloading. In the 20-node networks, the effect of such advantages of γ = 0.8 is less prominent as the smaller values of γ results are more comparable. However, γ = 0.8 still showed the best overall performance among tested γ values.

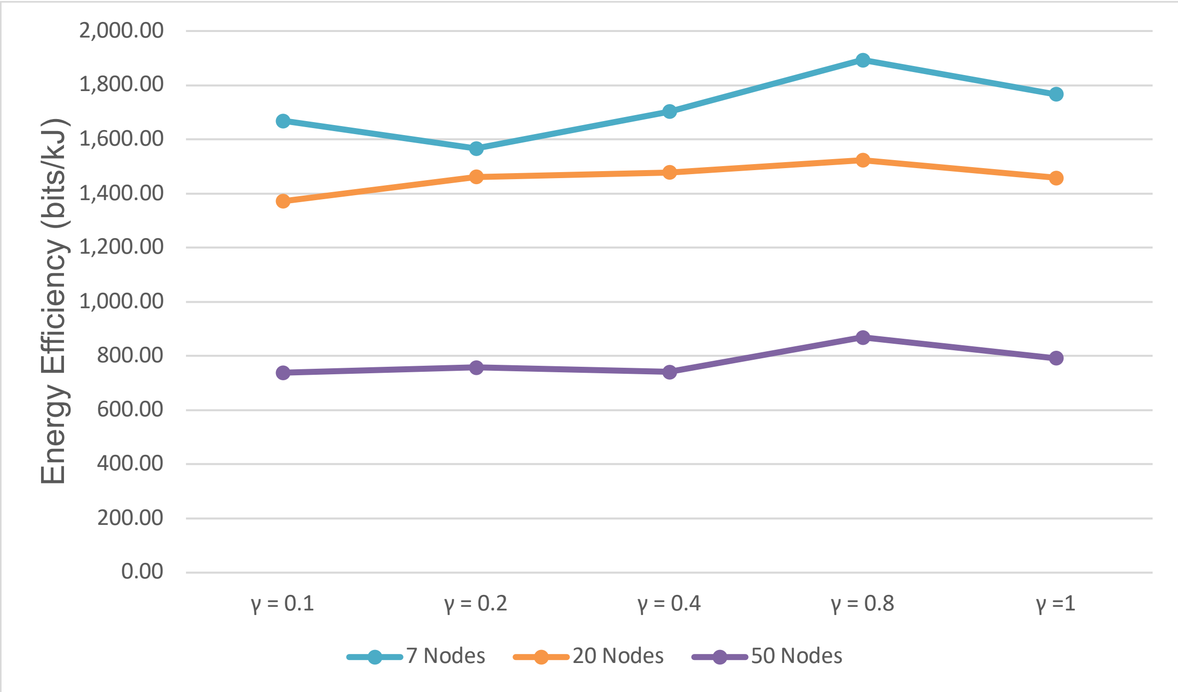


Figure 6.9 Average energy efficiencies of simulations with different γ values.

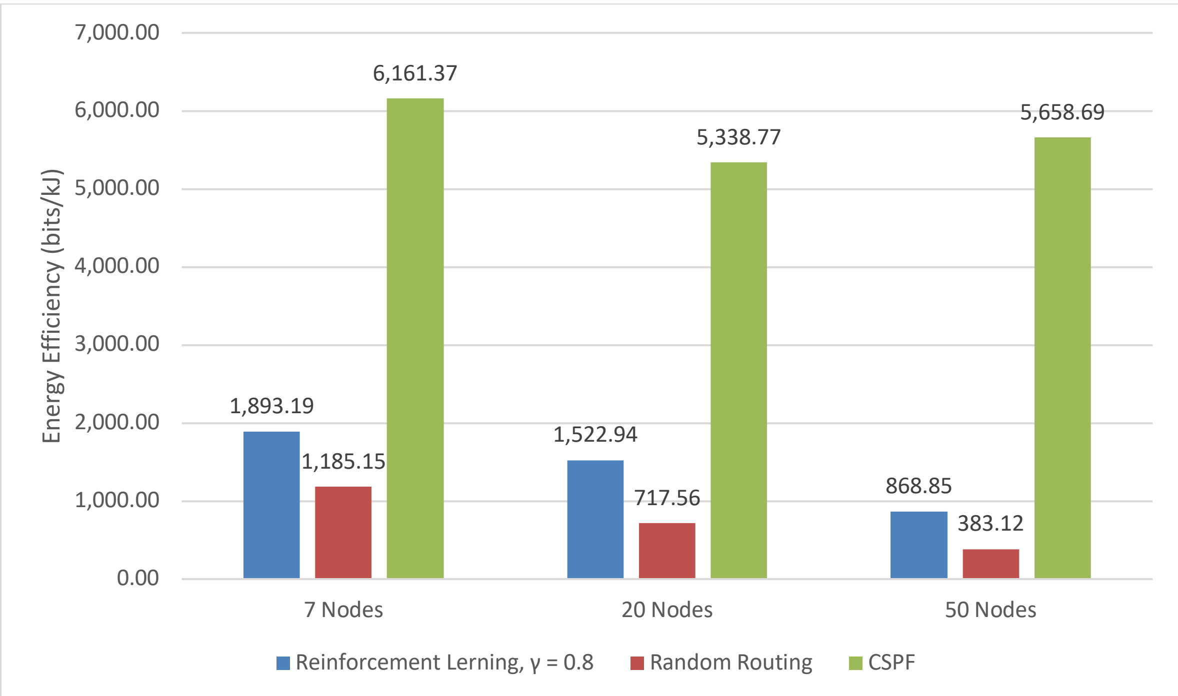


Figure 6.10 Average energy efficiencies of simulations with reinforcement learning, random routing and CSPF algorithms.

The energy efficiency results shown in Figure 6.9 and Figure 6.10 are quite similar to the failure rate results shown in Figure 6.7 and Figure 6.8. While the CSPF method kept its performance advantage, we noticed a significant improvement that the learning can bring about over the random routing algorithm from Figure 6.10. Moreover, the value of $\gamma = 0.8$ had also been proven that it is more energy efficient than all other γ values, especially in the large-scale networks for the same reason from the failure rates found in Figure 6.7. Whereas the smaller networks, the discount factor plays a less influential role as the convergence of route selection can be reached much quicker as there are fewer routes to choose from for each node. The influence from different balances between exploration and exploitation was much more prominent than here.

The results of simulations for CBUR of the network are shown in Figure 6.11 and Figure 6.12. Figure 6.12 also shows the advantage of having a centralised information over distributed knowledge. However, the efficiency advantage of the value of $\gamma = 0.8$ has dropped significantly in comparison to failure rate and energy efficiency compared to other γ values. This can be interpreted as the most energy efficient route may not be the most carrier band efficient as mentioned in the discussion in Section 4.2.1. However, it still preformed one of the best results in the simulations. The routing choice of the reinforcement learning routing algorithm was based on the cost function, which focuses on the energy consumption between the nodes. This value considers not only the energy consumption of the nodes involved in the transmission but also the remaining battery information of them. The variation of this remaining battery information during the simulation may divert alternative route with longer hops, hence higher carrier bandwidth usage. This may result a less efficient route but avoiding overloading certain nodes in the network which reduces usability. Hence, it keeps the entire network more useable in the long term.

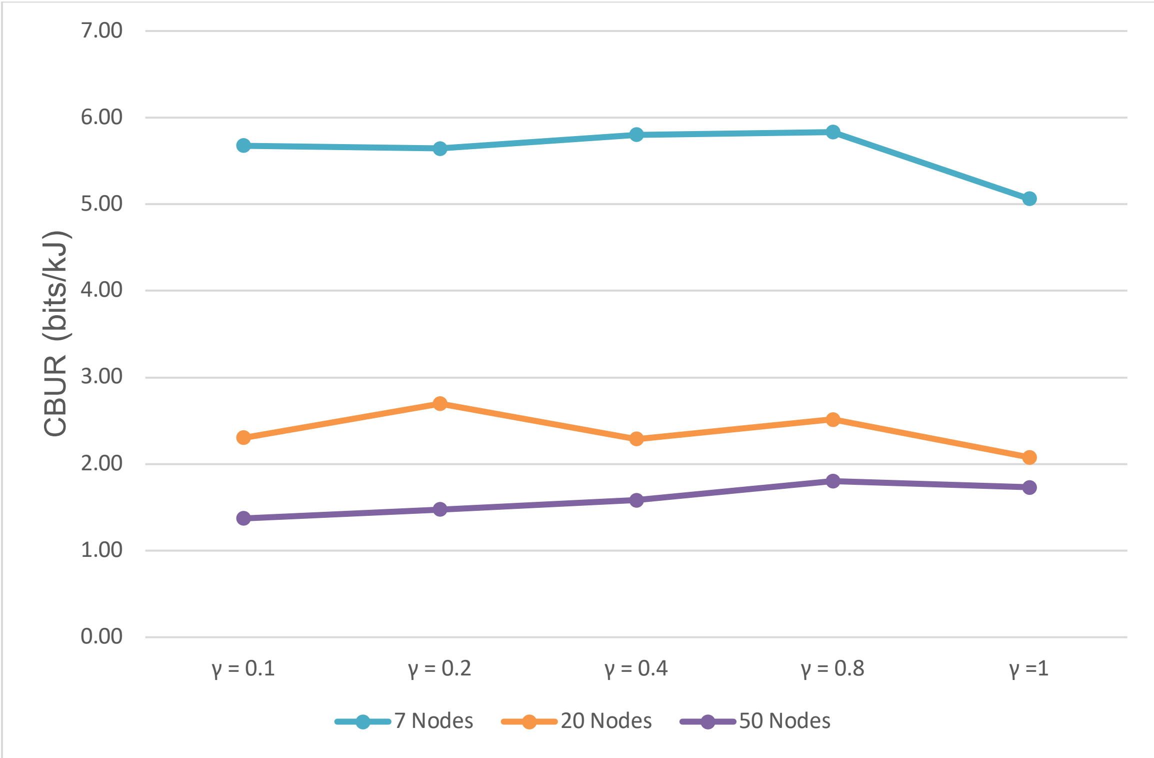


Figure 6.11 Average CBUR of simulations with different γ values.

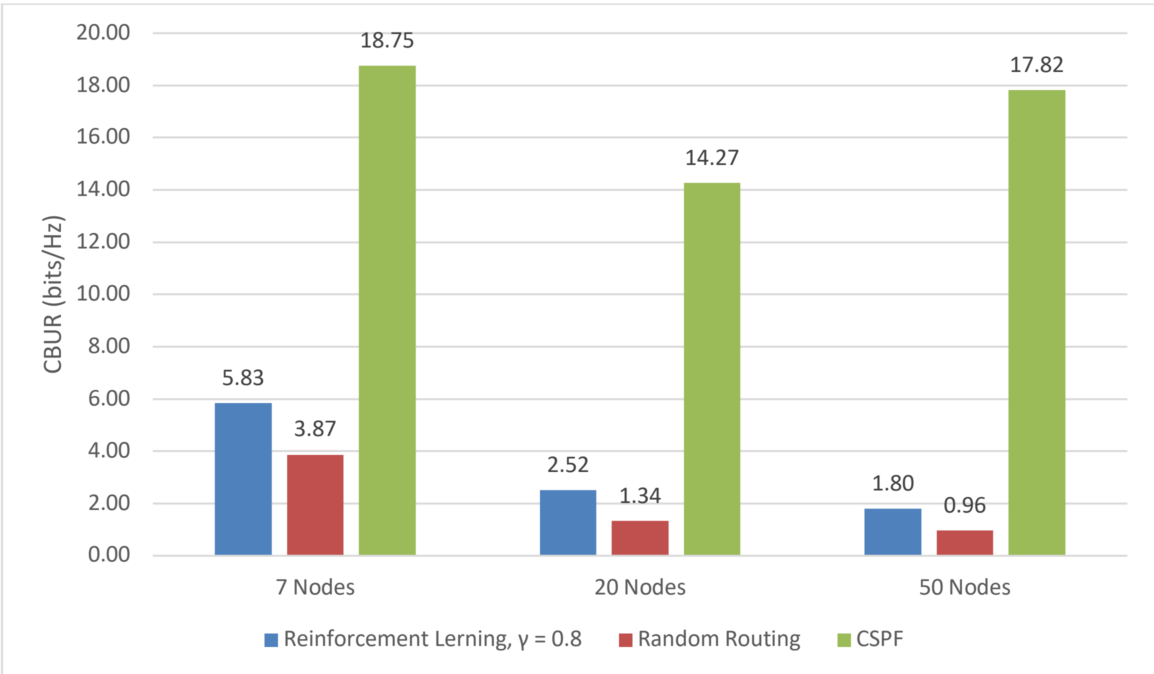


Figure 6.12 Average CBUR of simulations with reinforcement learning, random routing and CSPF algorithms.

In summary, among all γ values, 0.8 was found to be the most suitable for the performance of the RL routing algorithms in the context of remote monitoring IoT mesh networks, as discussed in Section 4.2.2.

6.1.3 Learning rate (β)

The learning rate β also plays a key role in the reinforcement learning algorithm used in this thesis. It determines the rate of newly acquired information replacing the known old information. This is done, as shown in (6.6), by multiplying β with the difference between new and old RM values. The RL routing algorithm is a time-based learning schedule as the learning rate alters how much old information are taken in each iteration when updating the routing table.

In this series of simulations, we used 31781 randomly generated point-to-point transmissions in 7-node networks. 31398 transmissions were used in 20-node networks, and 31611 transmissions were for the 50-node networks.

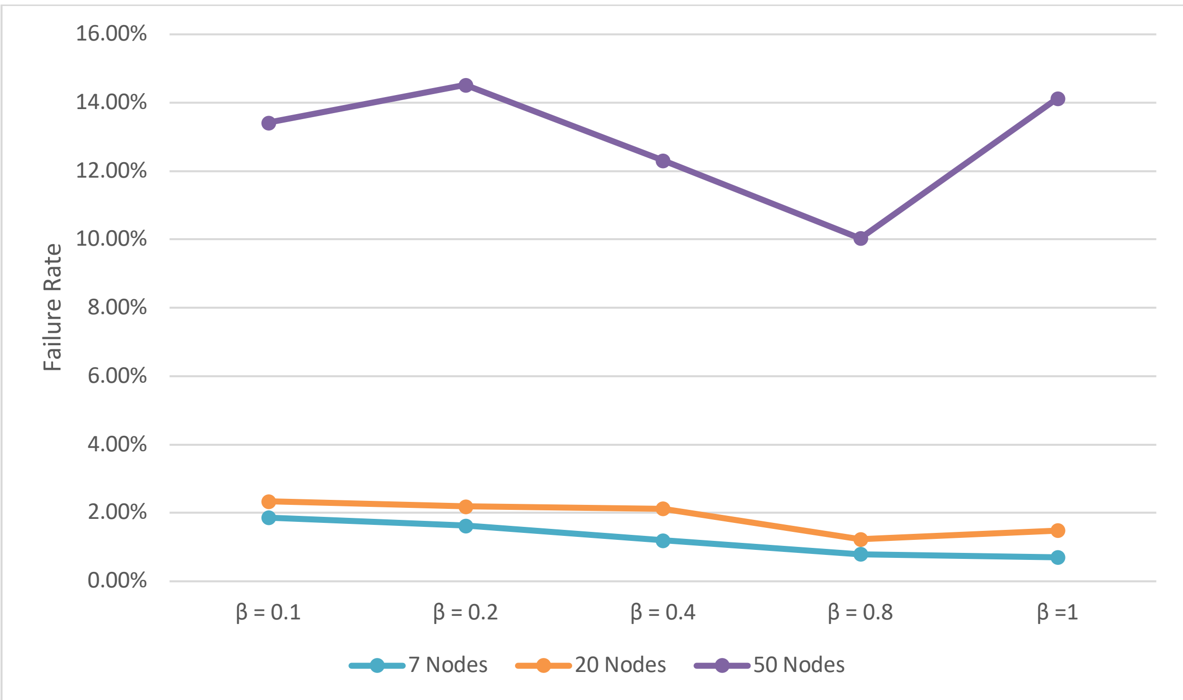


Figure 6.13 Average failure rates of simulations with different β values

As shown in Figure 6.13, a larger β value has a tendency of having a lower failure rate when the scales of networks are not too big. However, this cannot be applied to the 50-node network as the failure rate of β = 1 was almost the worst in the series of simulations. The larger learning rates changes the RM values in the routing table much quicker than the smaller β values. This leads to a numerical overflow as the RM value of earlier selection may be greatly changed from the original value. This can be an advantage when the scale of the network is smaller, as the limitation of the selection of routes will lead to a more efficient converged final route selection. With fewer choices of routes, this results a better performance when the network is really small. But in the bigger networks, this means the route selection is constantly changing like random routing, because of more choices of adjacent nodes for each node offer by the large networks. Each time one node being chosen, the corresponding route metric is updated, but with the new information will overflow the old information, leading away from convergence. Therefore, the extreme value of β = 1 was much worse than β = 0.8 in the large-scale networks. Another noteworthy trend can be seen in Figure 6.13 is that the failure rate when β = 0.1 is smaller than β = 0.2. This can be explained as very small β values do not affect the long-term route selection that significant, and the result more influenced by the random generated model in this situation.

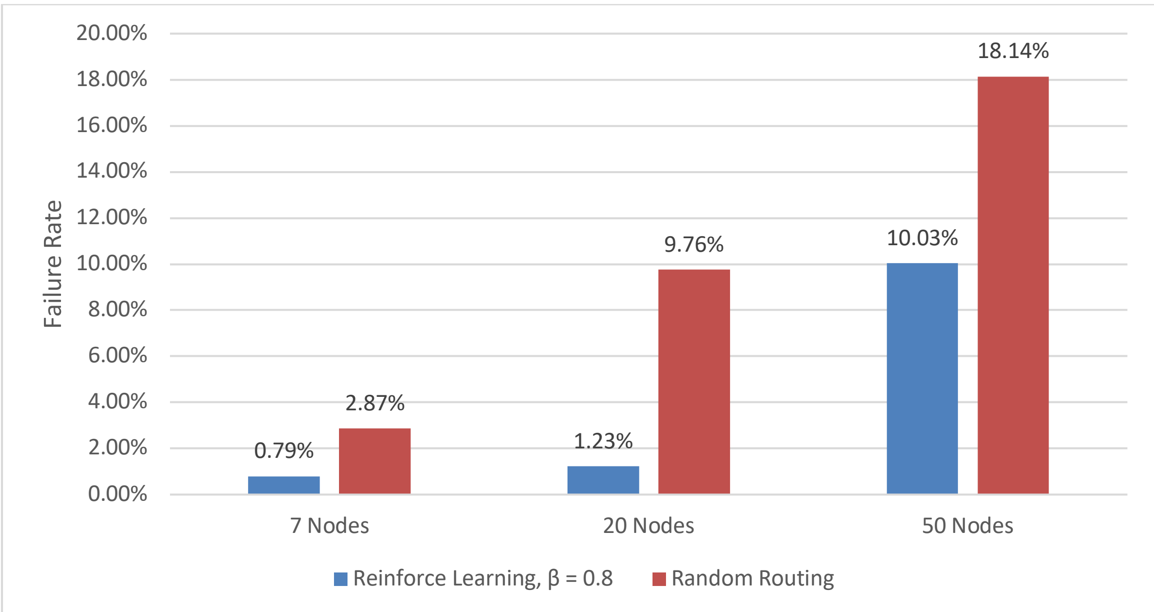


Figure 6.14 Average failure rates of simulations with reinforcement learning, random routing algorithms.

Figure 6.14 presented the comparison of the RL routing algorithm with β = 0.8 against the random routing benchmark. Similar to what can be found when discussing Boltzmann exploration hyper-parameters and discount factors, the RL routing algorithm with a reasonable parameter setting performed much better in failure rate in comparison to random routing with almost all situations met the target we set in the first part of this chapter. Even in the case of 50-nodes, the result is marginally lower than the target. However, considering the random routing had nearly double the failure rate of the RL routing algorithm, this can be caused by the network situation being more complicated than the previous simulations, which are also randomly generated. The CSPF continues to be the none-failing method which has also been omitted here.

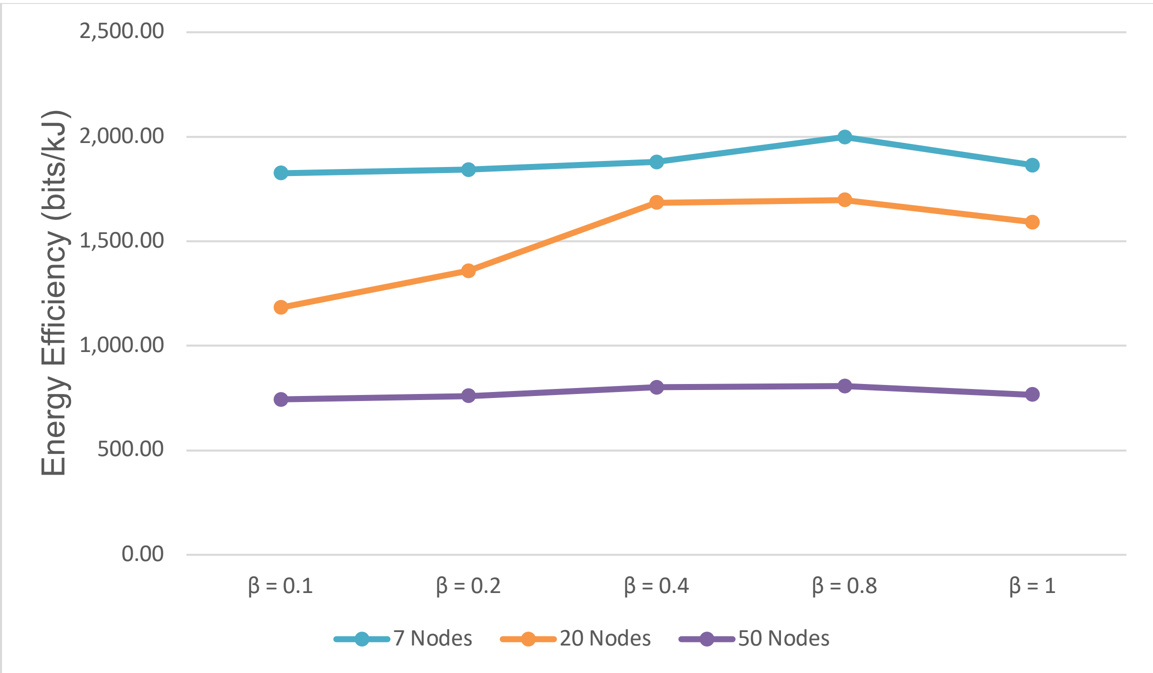


Figure 6.15 Average energy efficiency of simulations with different β values.

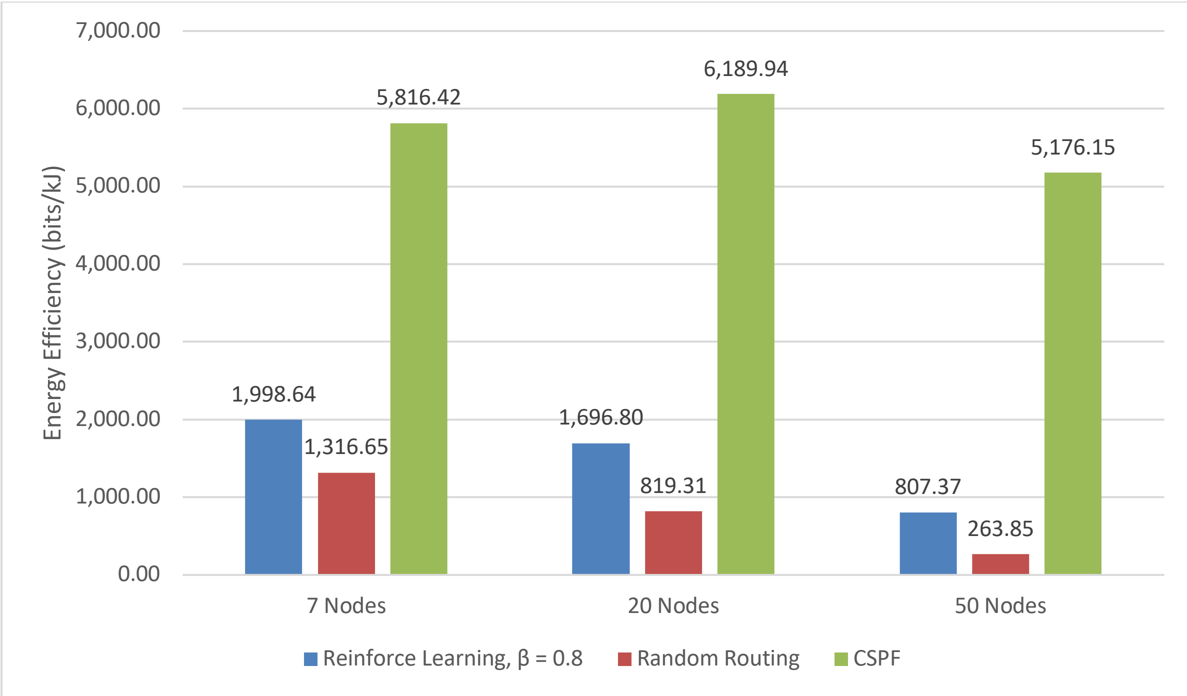


Figure 6.16 Average energy efficiencies of simulations with reinforcement learning, random routing and CSPF algorithms.

In Figure 6.15, β = 0.8 resulted in the best energy efficiency among all β values among the reinforcement learning algorithms. This can be described in the similar way as the failure rate. In the large-scale networks, the result of β = 1 is not as bad as it was in the failure rate in Figure 6.13. This is because even more transmissions had failed in this situation. The better-learnt successful route can be more energy efficient than slowly learning lower algorithms with small β values. When consider the total efficiency during a long-term simulation, the difference was eventually evened out. β = 0.4 results showed comparable numbers in terms of energy efficiency to the β = 0.8 results.

Figure 6.16 shows a very similar story of that of τ and γ values in results of previous sections. The numbers in the results in different simulations varies was expected, as the topologies in each set of simulations were randomly generated separately. But the reinforcement learning routing algorithm here has shown a steady advantage over the random routing as the other sets of simulations.

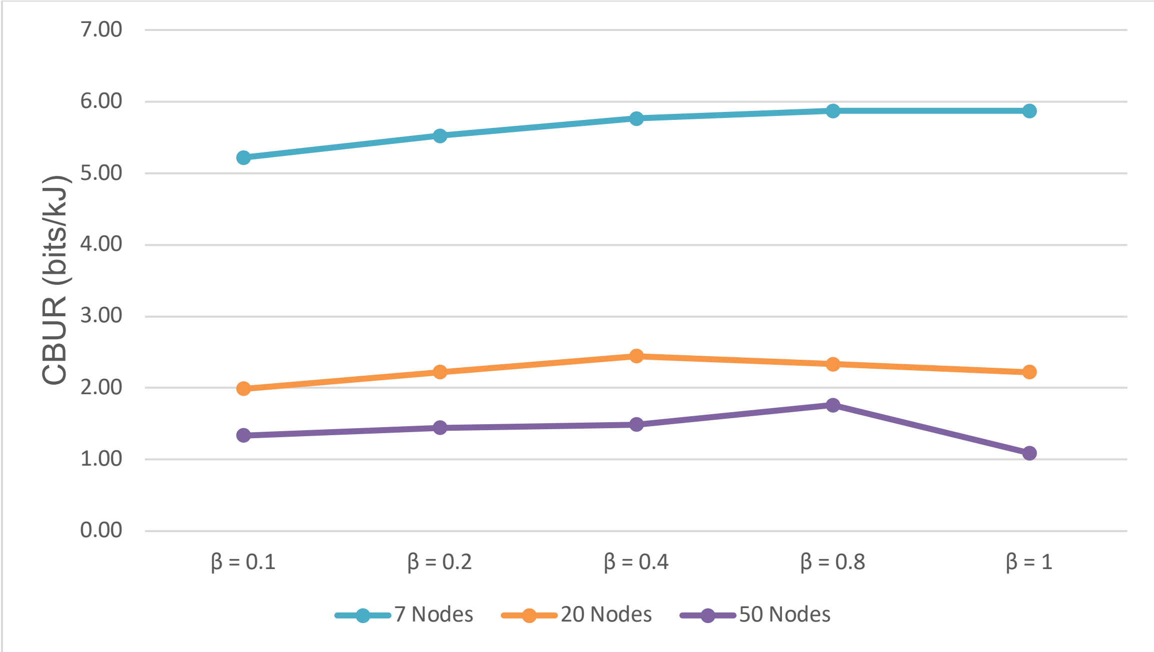


Figure 6.17 Average CBUR of simulations with different β values.

The CBUR of reinforcement learning routing algorithms with different β values can be found in Figure 6.17. We noticed that in the 20-node networks, β = 0.4 has shown the best CBUR among all β values, marginally higher than β = 0.8. However, at the other scales of the networks, β = 0.8 still demonstrated the best CBUR among all β values. The value of CBUR of β = 1 dropped again as the in the 50-node network, the RL algorithm failed to converge, resulting in drops in numbers just like in the energy efficiencies and the failure rates figures.

As shown in Figure 6.18, the RL routing algorithm performed better than the random routing but was much worse than the CSPF algorithm as expected. The RL routing algorithm stably outperformed the random routing nearly twice in terms of CBUR.

In summary, we concluded that the RL routing algorithm outperformed the random routing in all 3 measurements in all sizes of networks. The CSPF had the absolute best results in all networks, but it is impractical to be used in any real-life network. We have found that in the RL routing algorithm τ = 0.5, $\gamma = 0.8$, and β = 0.8 are the optimal parameters for the best result.

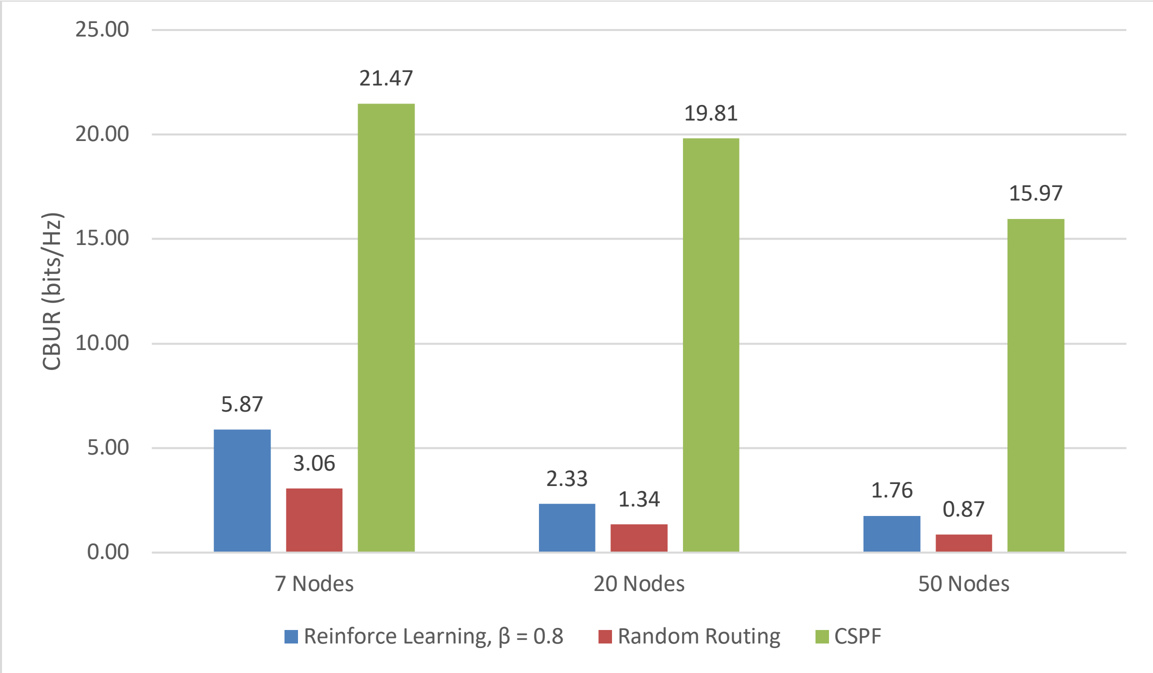


Figure 6.18 Average CBUR of simulations with reinforcement learning, random routing and CSPF algorithms.

6.2 The impact of the charging cycle of the nodes

The charging cycle also plays a crucial role when considering how the learning algorithm performs against the benchmarks. As the model of the node described in Section 3.2.1 suggests, every time the battery of a certain node is recharged, we assume its P_{now} is reset to P_{Max} instantly. This recharge behaviour not only impacts on how the availability of the routing nodes changes but also how the learning algorithm calculates the cost of each route over time as the power percentages of both transmission and receiving node of each node are considered in the cost function in the algorithm as discussed in Section 3.2.2. We reiterate the cost function (3.6) here as (6.7):

$$C = W_1 P_t - W_2 \, log(P_{TN}) - W_3 \, log(P_{RN}) \tag{6.7}$$

In the model of the simulation, every time the battery of each node gets recharged, the P_{TN} and P_{RN} are reset to 1. This behaviour will change the cost of each leg of the transmission accordingly when a recharge happens. The example solar panel [126] mentioned in Section 3.2.1 is able to provide a 1W charging output at 5.5V. The efficiency of typical battery charging circuit for laptop/palmtop computer during high charge current is 90% [139]. In this thesis, we considered the charging efficiency of the charger in the sensor node η_{charge}. as 80% as the charging process is not in a high charge current situation. We also assumed no other power consumption such as power leakage in the process for the ease of calculation in the simulation.

According to the climate data provided by the met office [30], we calculated the average daily sunlight hours in Table 1.1 in Chapter 1. Throughout a year, the average hour of sunlight in the UK is 3.76 per day, and 6.00 per day during the longest sunlight month (May), while only 1.32 per day when it is the shortest month (December). We can then use this data to calculate the charging cycle (CC) in the unit of days we use in the simulation for the environment of the UK using the equation (6.8):

$$CC_{\mathrm{days}} = \frac{P_{\max}}{P_{\mathrm{charge}} \times \eta_{Charge}} \tag{6.8}$$

Where CC_{days} stands for charging cycles measured in days, $P_{\max}$ was 15Wh for each node in the simulations in this thesis, P_{charge} was 1W in the case of the example solar panel mentioned earlier, η_{charge} for the charging efficiency, which was 0.8 in this case. We then calculated the charging cycle of the average, best case and worse case for the simulation is 3.125, 4.98 and 14.20 days respectively. To roundup for the ease of representation, we choose 3 days, 5 days and 15 days in the simulation. This selection of charging cycles represented the actual climate conditions in the context of deployment of remote monitoring devices for the rural areas in the U.K. We then converted the CC_{days} into $CC_{\mathrm{timeslots}}$ that can be used in the simulation using the following equation (6.9):

$$CC_{\text{timeslots}} = CC_{\text{days}} \times N_{\text{timeslot per day}} \tag{6.9}$$

where $N_{\text{timeslot per day}}$ represents the number of timeslots per day, and in this simulation is 144 for the 10 minute-timeslot mentioned in Chapter 5. Hence, the $CC_{\text{timeslots}}$ is 432, 720 and 2160 for the best, average and worst climate scenario, respectively. We then used these $CC_{\text{timeslots}}$ in the simulation with optimal learning parameters we concluded in Section 6.1. We plotted the results of the comparison between failure rates of reinforcement learning routing and random routing under different charging cycles in Figure 6.19.

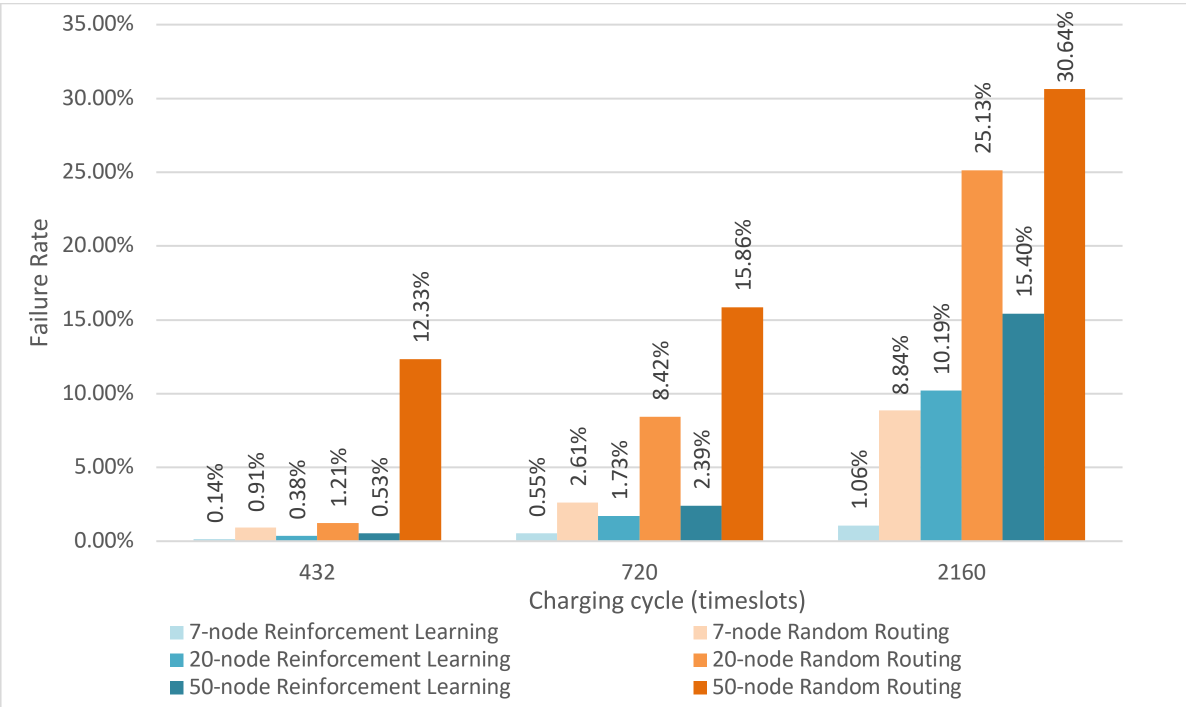


Figure 6.19 Average failure rates with different charging cycles

As shown in Figure 6.19, with the increase of the charging cycle, the failure rate of both the random routing and the reinforcement learning routing algorithm increased accordingly. However, the random routing algorithm suffered a greater impact on the 50-node network with more than 30 percent transmission failed with the 2160-timeslot charging cycle. It also failed 12.33% and 15.86% of all transmissions even with the 432 and 720-timeslot charging cycle respectively on the 50-node network. In the smaller networks, the failure rates were lower for the random routing, but with the 2160-timeslot charging cycle, it still had a 25.13% failure rate on the 50-node network. This made random routing unusable when the network is large and charging cycle is long, such as a large-scale monitoring network in the unpredictable climate of the rural areas. The same trend can also be found for the reinforcement learning routing algorithms as well, however, the reinforcement learning routing algorithm was significantly more resilient to such extension of the charging cycle. The worst result appeared in the 50-node network with a 2160-timeslot charging cycle. 15.4% of failure rate for the size of the network and the length of the charging cycle was much more usable than the random routing. In this case, the network become quite unusable as the failure rate is 50% higher than the target we set in the earlier part of this chapter. However, given the fact that 2160 timeslots are equivalent of 15 days, this failure rate is not that bad as the number suggested. The entire month of December has only 2 of this 15-day charging circles. Just like previous simulations, the CSPF algorithms always kept 0 failure of all simulation because of the up-to-date knowledge of the network which has been omitted in this figure. The results here demonstrated the advantage of the reinforcement learning routing over the traditional random routing for the remote monitoring IoT networks in the context of the climate of the U.K.

Figure 6.20 Average energy efficiency with different charging cycles

We then looked into the impact of charging cycles on the energy efficiency of the network with different routing options in different network scales. As can be seen on Figure 6.20, the effect of the charging cycles on the CSPF algorithm was minimal. As it is based on the centralised information, the CSPF can always work out the efficient route at any time. The energy efficiency has dropped with the increasing of the number of nodes. This is the same trend we observed in the section 6.1.1. The increase of complexity of the network will make the route longer for each transmission, hence, the decrease of the energy efficiency.

However, it can also be observed that the impact of the extended charging cycle impairs the random routing more than the reinforcement learning routing algorithm. When the charging cycle changed from 432 to 720 timeslots, the average energy efficiency of the random routing had dropped 31.6% while the reinforcement learning routing had only 8.1%. The decrease from the 720-timeslot and the 2360-timeslot were 62.4% and 50.1% respectively. The biggest decrease of the random routing was more than 58.4% from 432 to 720 timeslots in the 20-node network and 66.3% from 720 to 2360 timeslots in the 50-node network whilst for the reinforcement learning routing algorithm was only 15% from 432 to 720 timeslots in the 7-node network and 58.2% from 432 to 720 timeslots in the 7-node network. It is notable that the RL routing algorithm only better than the random routing marginally in 7-node network simulations, because of the lack of availability of selection of routes. The difference between the random routing and reinforcement learning routing algorithm in the larger networks and with longer charging cycles were much more significant when the learnt information is able to put into use. It is also noticeable that in case the 432-timeslot, 50-node network that the random routing has slightly better performance than the RL routing. This can be explained when the charging circle is so short and the network work is so simple, the network has not much information to be learned from. Nodes are mostly at its maximum power states and the routes are very simple. This resulted in this singularity. Except that, this general trend proves the benefit deploying RL routing over the random routing. The much less overhead RL routing brings about in comparison to the huge required computational power of the CSPF stated in Chapter 5 also proves the value of RL routing algorithm for the realistic remote monitoring sensing scenarios.

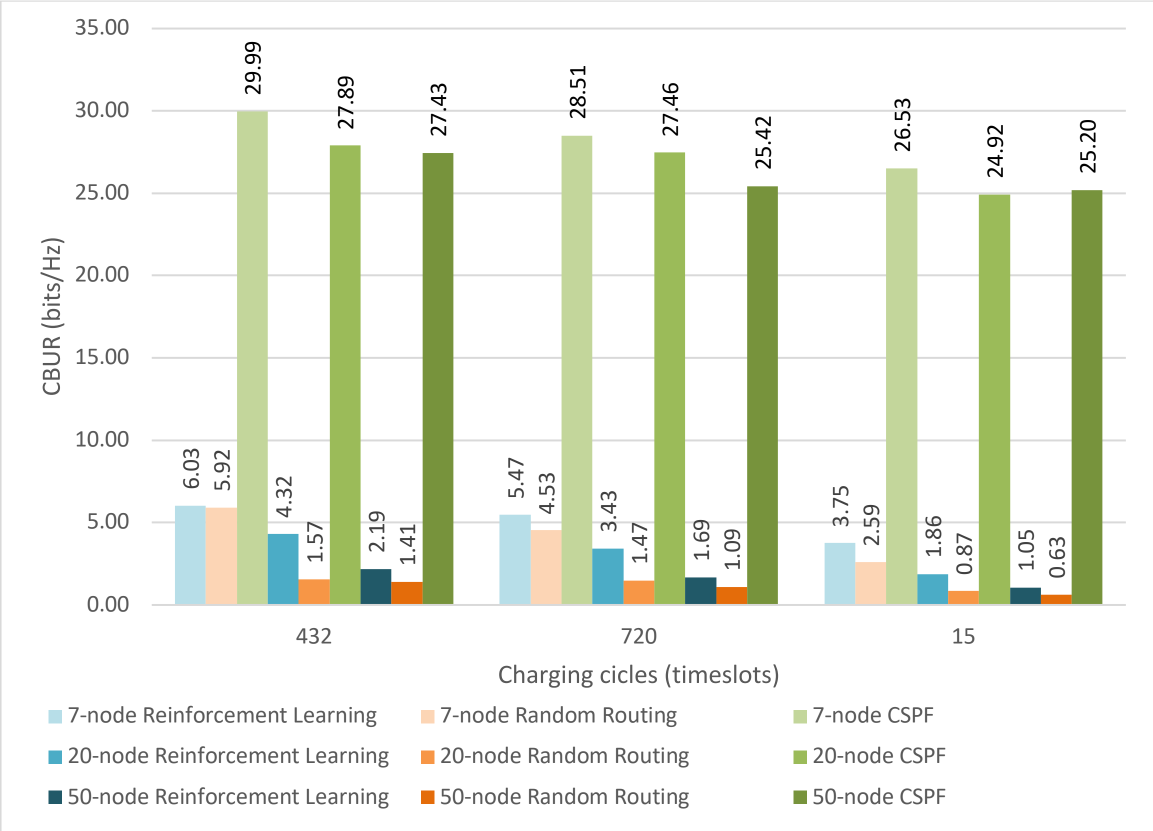


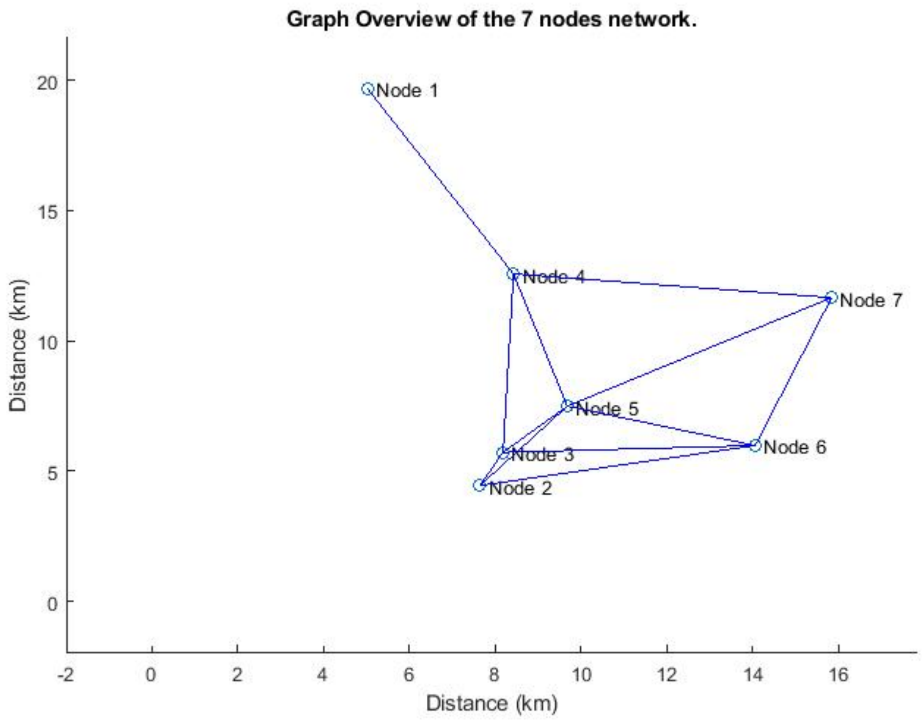
Figure 6.21 Average CBUR with different charging cycles

Similarities can also be observed from Figure 6.21 for the CBUR. Except had the little impact on the CSPF method, the longer charging cycle did influence the performances of both reinforcement learning routing algorithm and random routing algorithms. The reinforcement learning routing algorithm showed a stronger resilient to the change of the charging cycles while the random routing algorithm suffers from more fluctuation when the charging cycle condition changes.

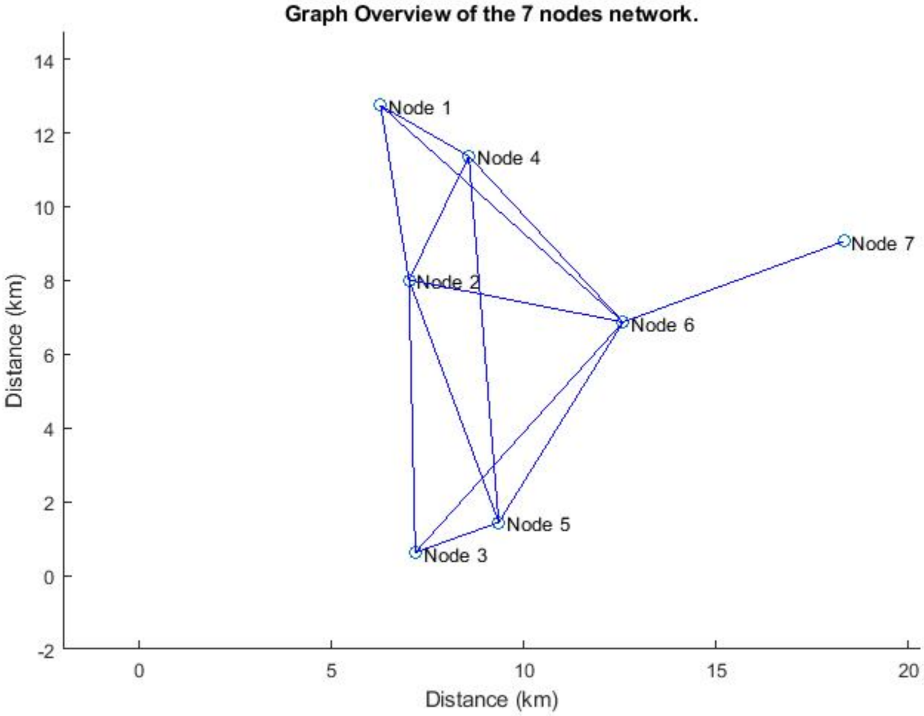
Thus, we can conclude that the proposed reinforcement learning routing algorithm shows better performance, in terms of failure rate, power efficiency and CBUR, than the random routing algorithm in the remote IoT environments where the capacity of the power supply is confined, and the recharging facility was inconsistent.

6.3 The impact of the maximum communication range of the nodes in the network

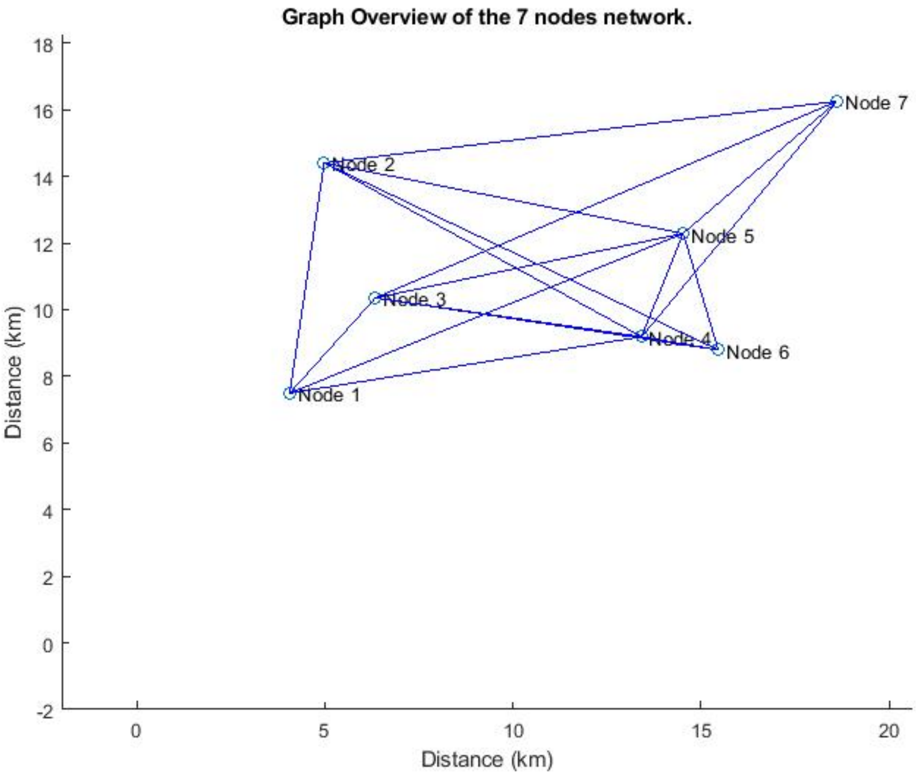
The different maximum communication range (MR) value of each node in the network affects the interconnectivity between nodes in a network. With a longer MR, a node can have more potential adjacent nodes. The complexity of the topology of the network will inherently change with different MRs. Given the same node, MR can be determined by the different environment the network operates in reality. Alternatively, the MR can also be presented in a fashion of limit the maximum transmission power of each node, but we chose using MR as the metric to study is because of the different MRs can deliver clearer representations of the change of topology of networks before the simulations start. Besides, the maximum transmission power can also be inferred by the maximum range as the range of the transmission is positively correlated with the transmission power as discussed in (3.4).



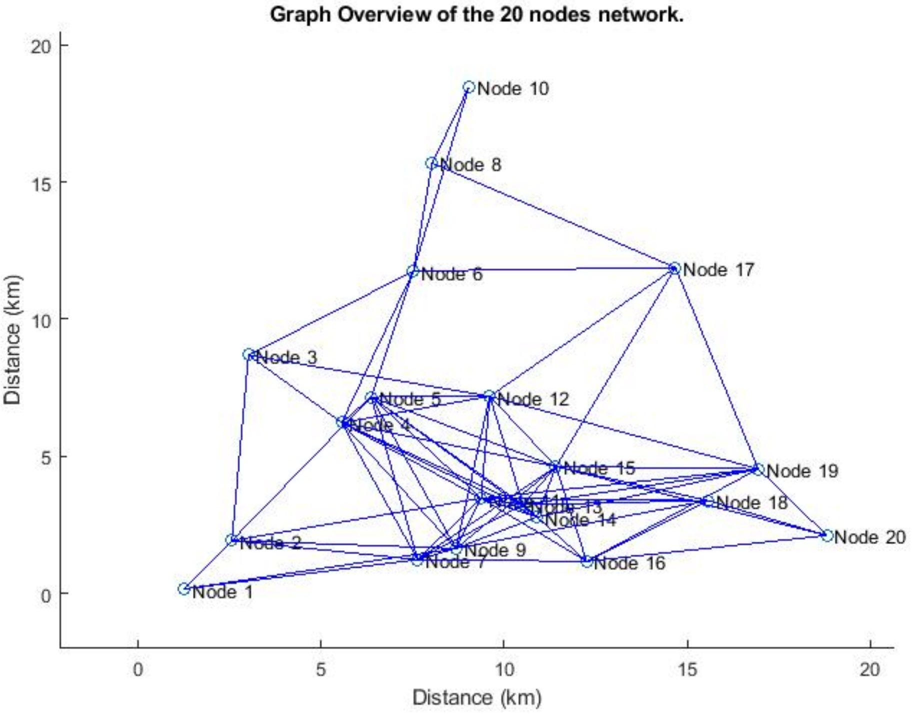
a. A 7-node network with maximum communication range of 8 km



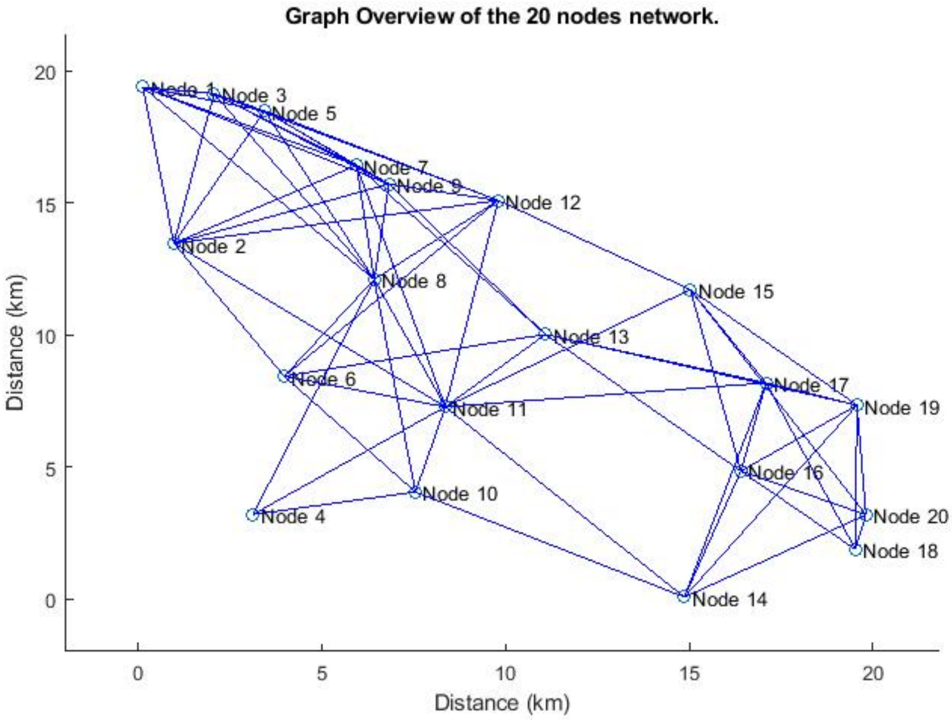
b. A 7-node network with maximum communication range of 10 km



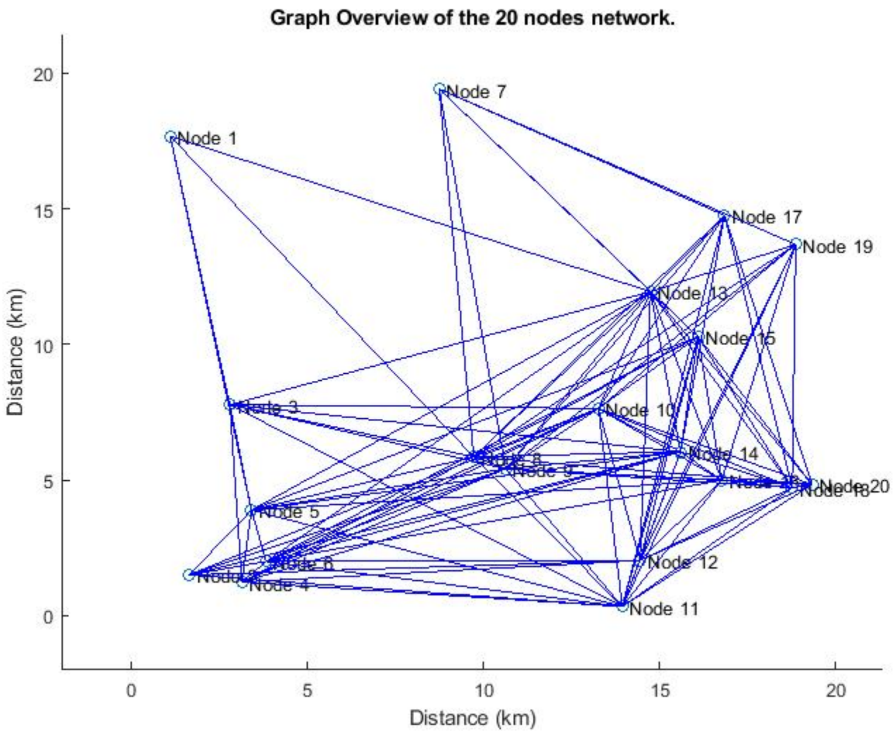
c. A 7-node network with maximum communication range of 15 km



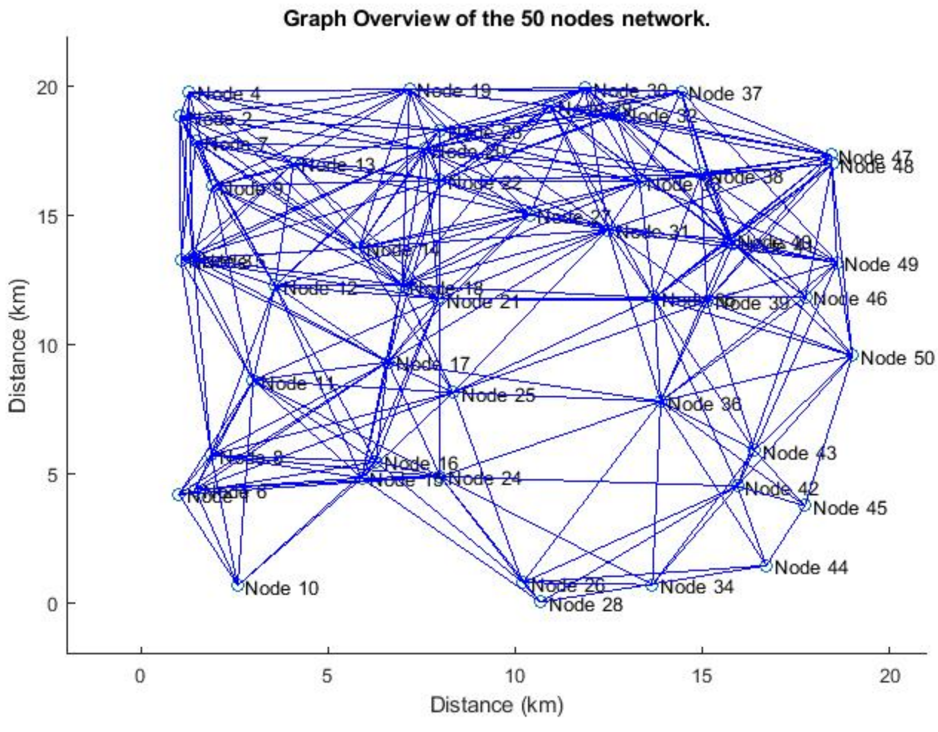
d. A 20-node network with maximum communication range of 8 km



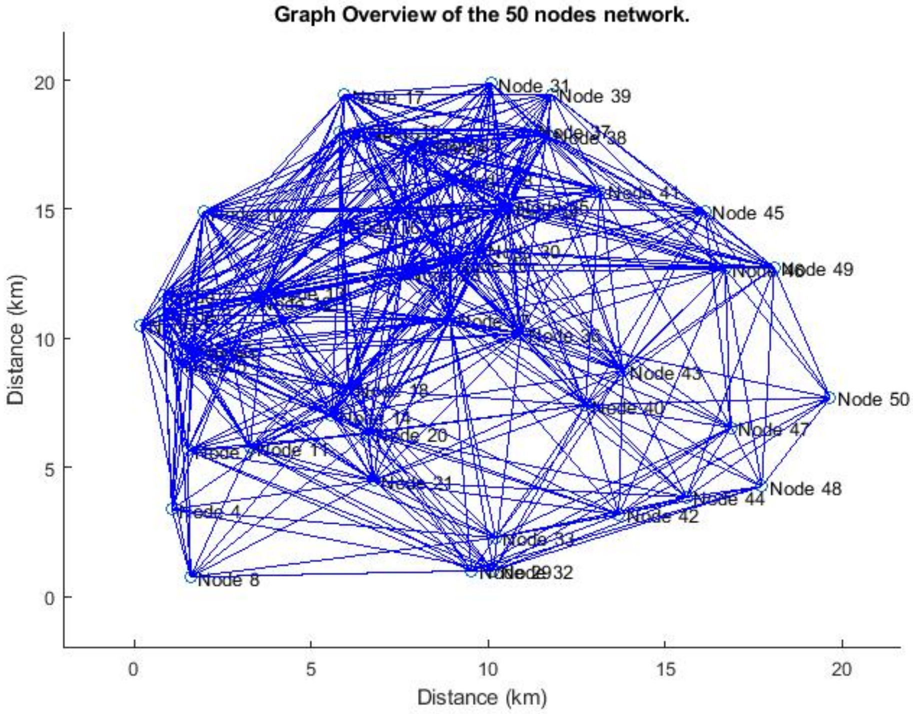
e. A 20-node network with maximum communication range of 10 km



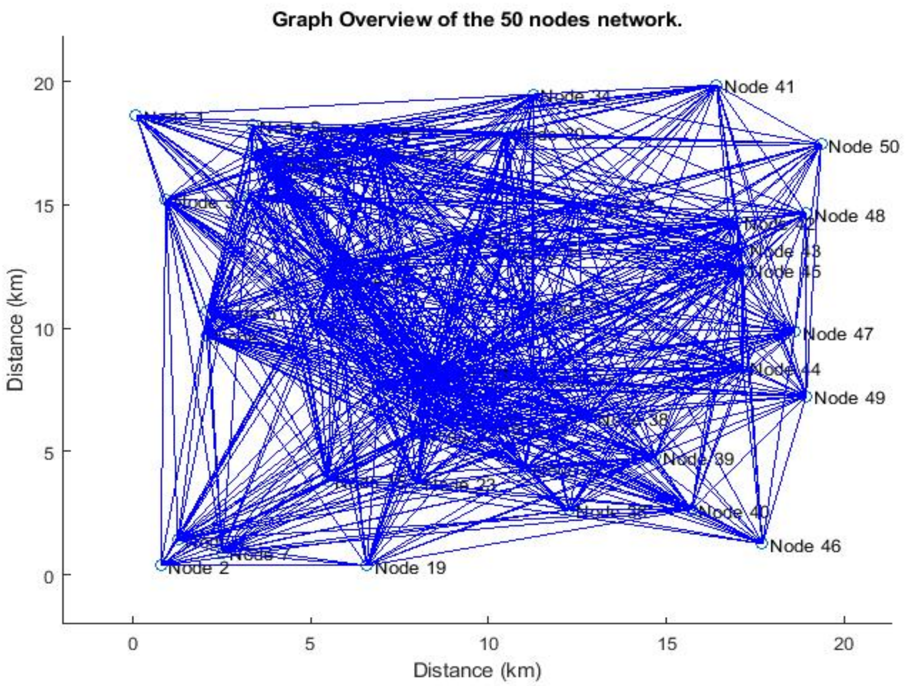
f. A 20-node network with maximum communication range of 15 km



g. A 50-node network with maximum communication range of 8 km



h. A 50-node network with maximum communication range of 10 km



i. A 50-node network with maximum communication range of 15 km

Figure 6.22 Examples of the different maximum communication ranges of randomly generated networks in a 20*20 km area.

As shown in Figure 6.22, we generated a series of 9 networks in the same area (20 km * 20 km) with different numbers (7, 20, 50) of nodes and different MR (8km, 10km and 15km) of connections between the nodes. The purpose of this study was to measure the adaptability of the algorithm to the scales of networks for the different environment against the benchmark networks. The complexity of networks with the same number of nodes differs from each other in the simulations. However, in the simulation, the network generator will connect every node into the network initially by relocating singleton nodes. As can be found in Figure 6.22a, the interconnectivity of the 7-node network with 8km MR is very limited. Some of the nodes even have only one single connection to other nodes, this means that if the only connected node fails, the node will become a singleton node and all transmissions to or from that node will fail until the next charging cycle. The networks with a larger MR or number of nodes are more interconnected, this phenomenon is less expected. In the network shown in Figure 6.22i the nodes are mostly interconnected, forming a nearly full-mesh network. This can make the route selection of the nodes difficult to balance so many possibilities, the data required to learn is much more comprehensive.

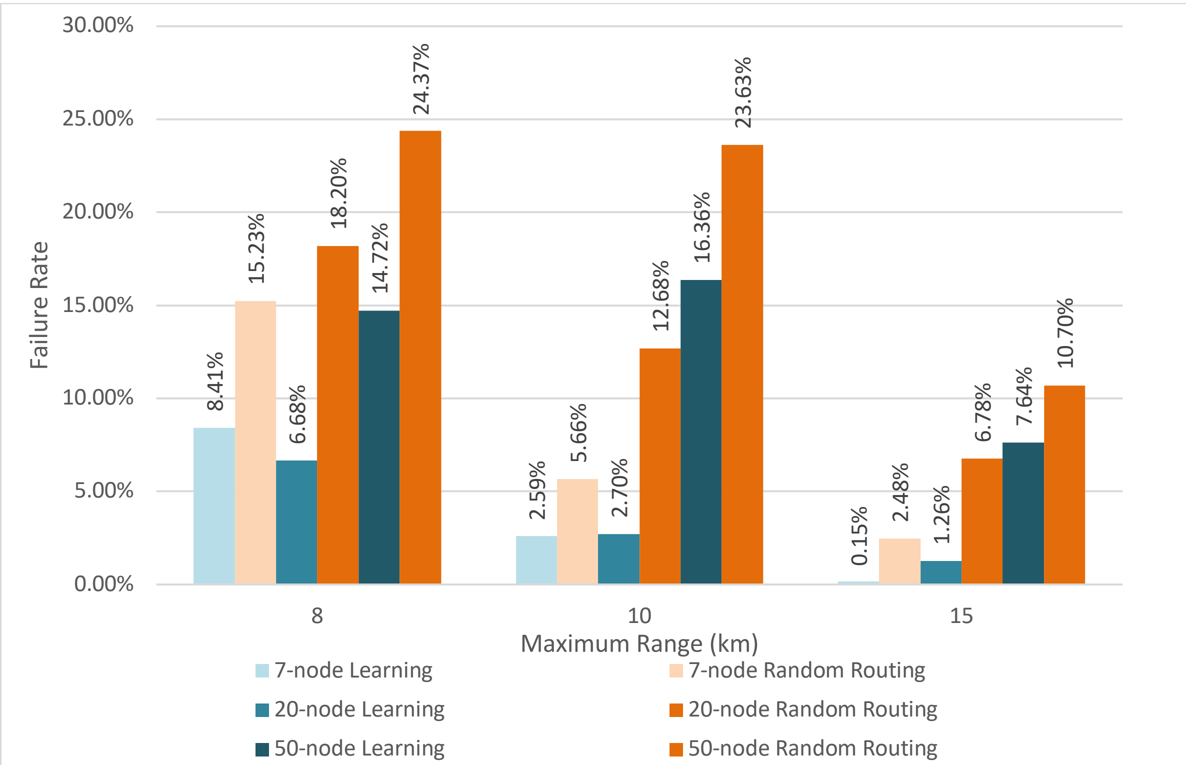


Figure 6.23 Average failure rates with different maximum communication ranges

From Figure 6.23, we observed that the networks with 8km of MR scored worst among all networks, regardless of the use of reinforcement learning routing algorithm when it comes to the average failure rate. Even in the 7-node small-scale network, this shortage of interconnectivity significantly reduced the ability of the network to deliver a low failure rate, hence the usability. This makes both algorithms failed to meet the target. However, the reinforcement learning routing algorithm did help to reduce the failure rate in comparison to random routing when the MR is 8km as the results of failure rates in 7-node and 50-node networks are only half of that of the random routing and only 1/3 in the 20-node network. Due to the lack of connectivity in the network, the limited choices of available adjacent nodes made the routing and re-routing when some nodes are failed very difficult. Therefore, nodes can be easily overloaded and become unavailable until the next charging cycle. While the reinforcement learning routing algorithm migrated this issue by avoiding nodes with less energy, but without this ability, the random routing suffered from a high failure rate in all the situations. With the addition of the maximum range, the connectivity increases significantly in both algorithms. However, the reinforcement learning routing algorithm still outperformed the random routing algorithm in all simulations and the difference in failure rate between the two increased as the interconnectivity becomes higher with the except the extreme interconnectivity shown in the 50-node network with the MR of 15km. This can be explained as the oversaturated choice of routes makes the learning process much slower.

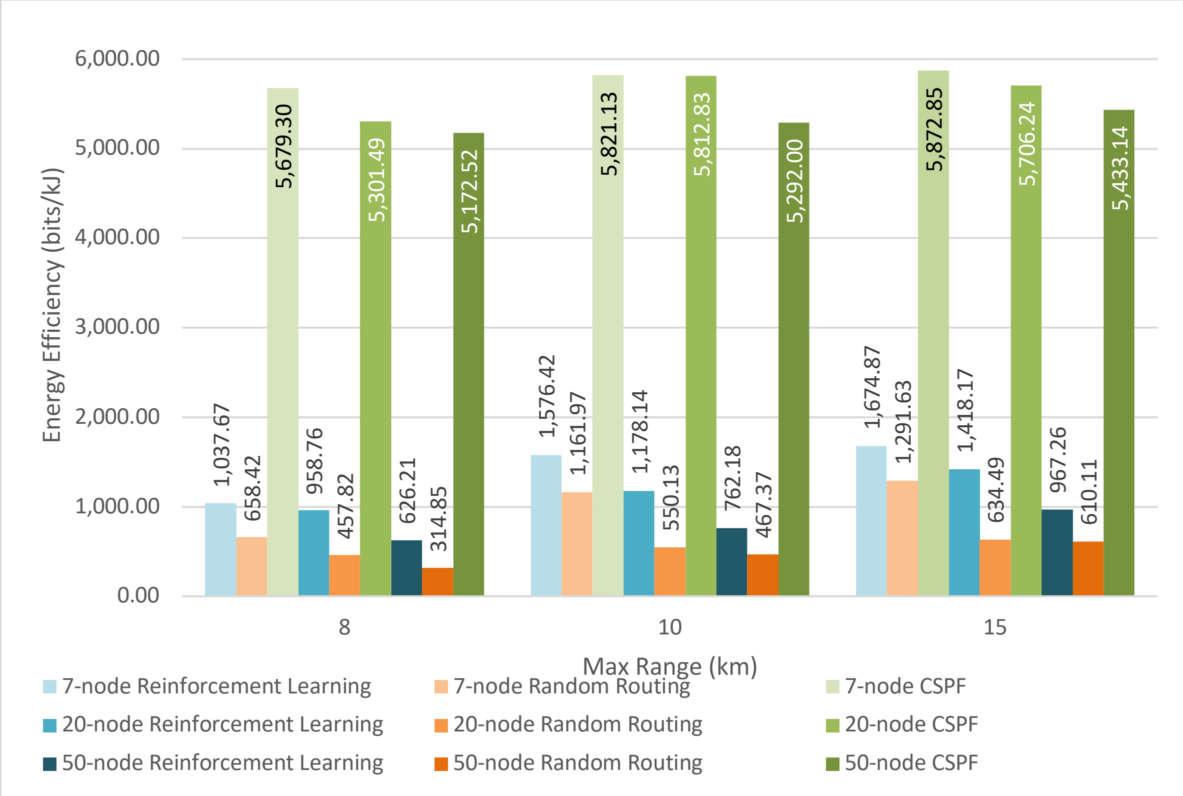


Figure 6.24 Average energy efficiencies with different maximum ranges

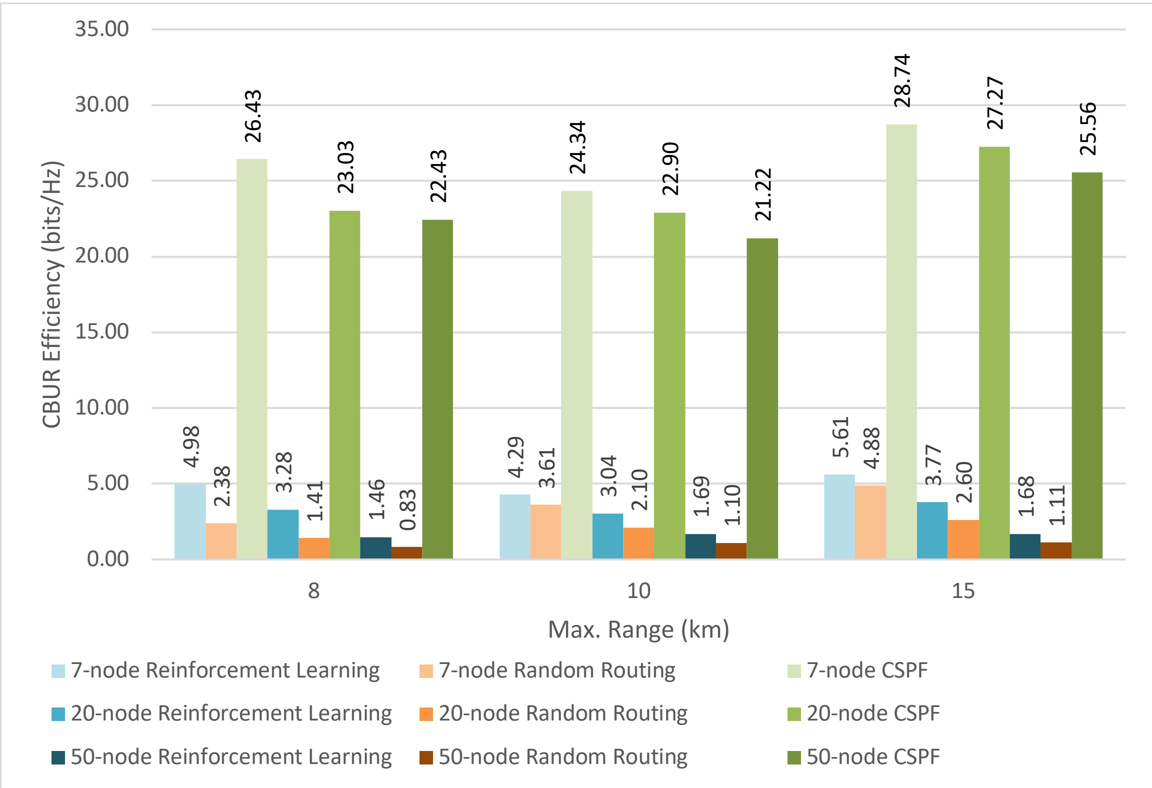


Figure 6.25 Average CBUR with different maximum ranges

Figure 6.24 and Figure 6.25 presents the average energy and CBUR of these simulations. As can be seen, the energy efficiency of both random routing and reinforcement learning routing algorithm increase as the MR increases in all 3 scales of the networks. Particularly, in the 7-node networks, both algorithms show great improvement from 8km to 10km in MR. The energy efficiency of the reinforcement learning routing algorithm increases 51.9% while the random routing raises 76.47%. This can clearly be understood as the decrease of the failure rate as well as more available routes contribute to this improvement. However, the CBUR do not have the same degree of improvement, especially in the reinforcement learning routing algorithm. When MR increased from 8km to 10km, all 3 scales of networks have shown a slight decrease in terms of CBUR. This is because that the more interconnected network may benefit the failure rate, but more complicated routes are available, the CBUR suffers as more legs are needed to accomplish a transmission. Hence, higher consumption of carrier bands occurs. The CSPF as it uses the best route all the time, therefore no positive nor negative effect can be noticed. The reinforcement learning routing algorithm showed an advantage over the random routing in all MR values of all networks, proved the effectiveness of the algorithm.

6.4 The reinforcement learning algorithm comparison

From the results above, we can find an optimised set of reinforcement learning parameters for the remote IoT mesh network simulation models considered in this study. The parameters from the simulations are listed in Table 6.1 as we found out in the earlier simulations. We also recap the other related parameters used in these series of simulations in the Table 6.2.

Table 6.1 The reinforcement learning parameters for the simulation with best results

τ	γ	β
0.5	0.8	0.8

Table 6.2 Other parameters used in the simulation

BW (kHz)	N (dBm)	I	α	h (fixed)	NR_{max}	P_{max} (Wh)	R (kbps)	CC (timeslots)	Max. Range (km)
125	-130	0	2.8	2	10	15	5	720	10

We then went through another set of 5 simulations of a year time duration (52560 timeslots) with all the best parameters we found in the simulations conducted in this chapter listed in Table 6.1 and Table 6.2 to establish how the reinforcement learning routing algorithm perform using an optimised set of parameters against the benchmark algorithms in the given remote IoT deployment environment. The total number of transmissions we used in this set of simulations was 31587 for the 7-node network, 31481 for the 20-node network, and 31871 for the 50-node network. We then plotted the results as follows:

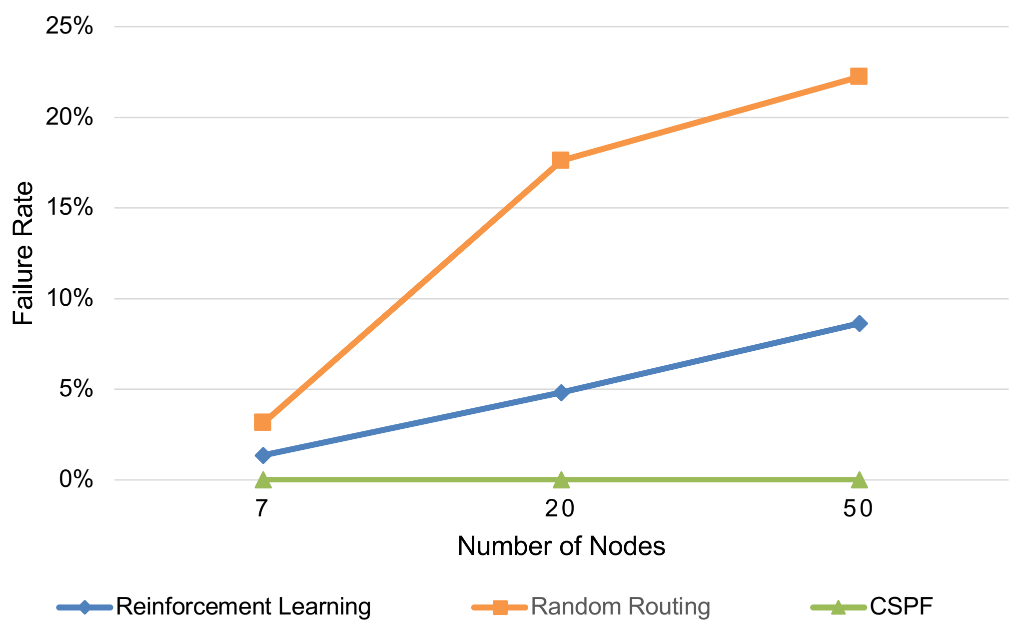


Figure 6.26 Average failure rates of the reinforcement learning, random routing and CSPF algorithms.

As demonstrated in Figure 6.26, the average failure rate of the RL routing algorithm is apparently lower than the random routing algorithm in all 3 different scales of networks. All these simulations, using RL routing algorithm has met the failure rate targets we set in the beginning of this chapter, whereas using the random routing, only in the case of 7-node network has met the same targets. With the scale of the network increases, the margin of average failure rate between the reinforcement learning algorithm and random routing algorithm increased, which has proven the value of employing reinforcement learning in routing the remote monitoring IoT networks as explained in Section 3.2.3.

We then investigated average energy efficiency and CBUR of the reinforcement learning routing algorithm in comparison to the benchmarks in the simulation. Figure 6.27 and Figure 6.28 present the results of the comparison, we can summarise that the RL routing algorithm performed better than the random routing algorithm in all 3 series of simulations in both energy efficiency and CBUR again. When it comes to average energy efficiency, the reinforcement learning routing algorithm shows great improvement over the random routing, especially in the 20-node and 50-node networks. Even in the simple small-scale 7-node network, the reinforcement learning routing algorithm still marginally more energy efficient than the random routing. This has also proved the RL routing algorithm is more energy efficient than the random routing algorithm. The reason for this improvement is that during the long period of simulation, the reinforcement learning algorithm learnt how the energy changes in the nodes in the network and route the transmission accordingly. This made the RL routing algorithm energy aware. As mentioned in Section 5.2, though the CSPF algorithm is much superior in all kind of performance, the low efficiency of updating the network and the complicity it involves ruled it out from implementation in any kinds of real world remote IoT networks. When it comes to the spectral efficiency from the CBUR chart, the improvement is much less profound as the RL routing algorithm is based on the energy information from the nodes. The benefit of the slight better spectral efficiency came with the great improvement in the side of energy. In both cases, the CSPF method has shown a lower result in the case of 20-node networks. This can also be explained as the complexity and the density of nodes helped the central knowledge the CSPF to perform better in the denser networks.

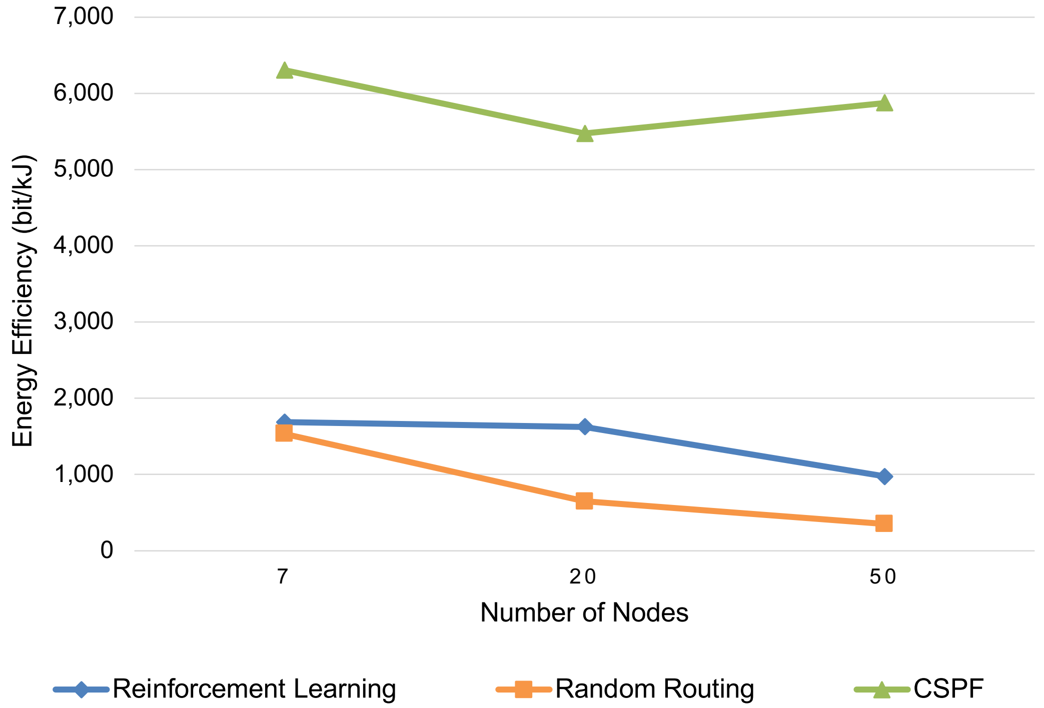


Figure 6.27 Average energy efficiencies of the reinforcement learning routing, random routing and CSPF algorithms.

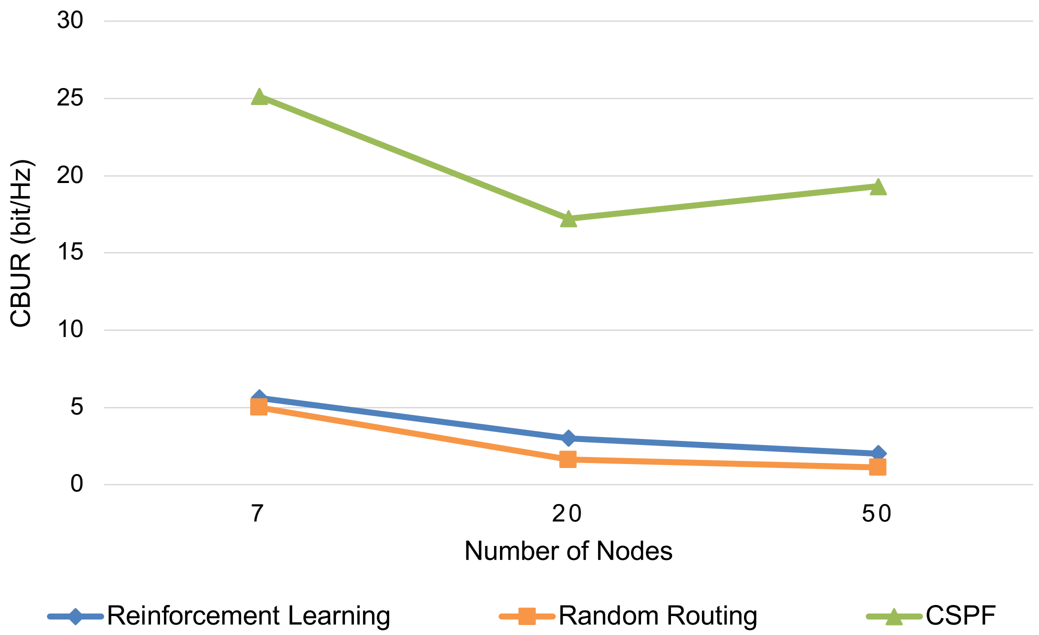


Figure 6.28 Average CBUR of the reinforcement learning, random routing and CSPF algorithms.

Finally, we plotted the progressive timeline of the failure rate, energy efficiency and carrier band efficient of the series of 5 simulations of 50-point large-scale network using the reinforcement learning routing algorithm conducted in Figure 6.29, Figure 6.30 and Figure 6.31. In addition to the actual number of the simulation results, we also plotted trendlines to show the continuity of the trend of that set of data. We can observe in Figure 6.29, the trend of the failure rate λ can be described as following a logarithmic trend with the time of t given in (6.10)

$$\lambda = -0.164 \ln(t) + 7.019 \tag{6.10}$$

As shown in the equation, the failure rate goes down gradually, which means over time, as the RL routing algorithm learns the network, the network becomes more reliable. Similarly, in Figure 6.30, we can draw (6.11) as its logarithmic trendline to conclude that the energy efficiency is increasing overtime. The energy efficiency starts improving quickly, but gradually slows down at a higher level. These trends show the progress of learning of the algorithm over the course of simulation or, in other words, the lifetime of the network.

$$\eta_{energy} = 35.764 \ln(t) + 762.85 \tag{6.11}$$

Finally, a linear trend following the (6.12) can be found in the CBUR chart shown in Figure 6.31.

$$\eta_{carrier} = 10^{-5}t + 2.1948 \tag{6.12}$$

The change of the CBUR is much less prominent. As we found in other simulation earlier in the network, the RL routing algorithm is learning on the energy pattern rather than the spectral efficiency can explain this phenomenon.

On the actual data of the failure rate shown in Figure 6.29, it starts with huge fluctuation in the early stage of the simulation. This is due to the RL routing algorithm has not yet learnt to manage the routing with sufficient data. As time progresses, the failure rate has reduced overtime and stabilised in the later part of the simulation as the RL routing algorithm gains a better knowledge of the network. The similar data pattern can also be found in both Figure 6.30 and Figure 6.31. The energy efficiency increased as the RL algorithm progressively learning the network providing a more consistent selection on route with better success rate. We can also observe that the CBUR has shown a similar pattern to the failure rate but with a lower degree of fluctuation. However, the improvement the RL routing algorithm provides for the spectral efficiency is limited as we explained earlier.

Additionally, we can also observe the data of the failure rate that the rate is also related to the charging cycle used here, 720 timeslots as the fluctuations of the data has a pattern of up and down every 720 timeslot. The failure rate went down every time the nodes gets recharged and up after that until next charging cycle. Energy efficiency also changed as the recharging process will make the route selection to the actual best route as all the previously exhausted nodes were back online after the recharge.

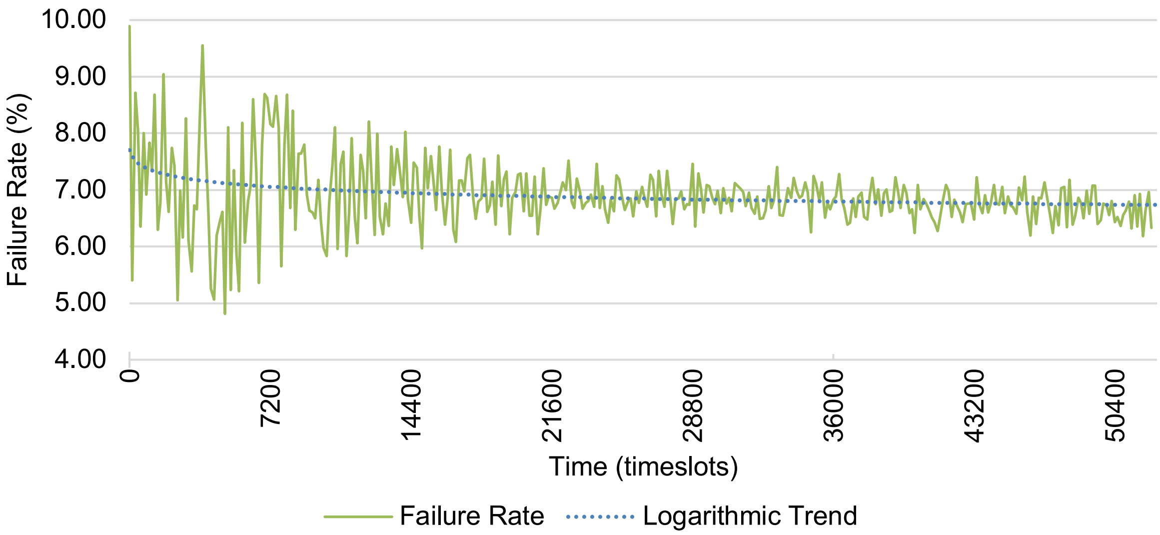


Figure 6.29 Time series and its trend of average failure rate for the 50-node mesh IoT networks using Reinforcement Learning Routing Algorithm.

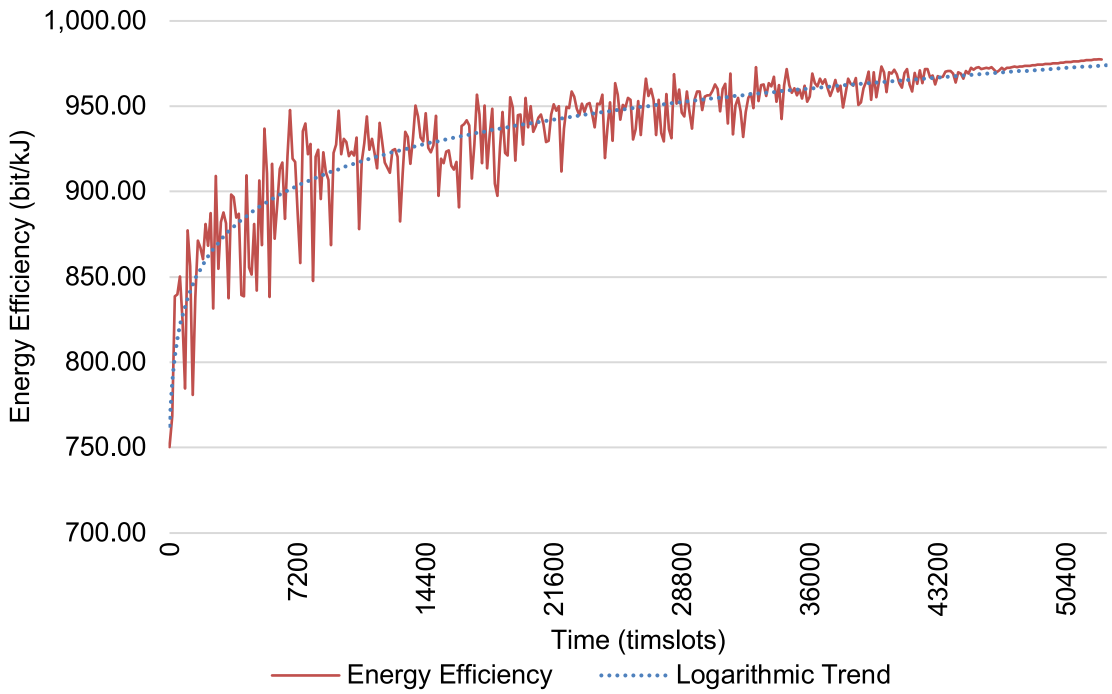


Figure 6.30 Time series and its trend of energy efficiency for the 50-node mesh IoT networks using Reinforcement Learning Routing Algorithm.

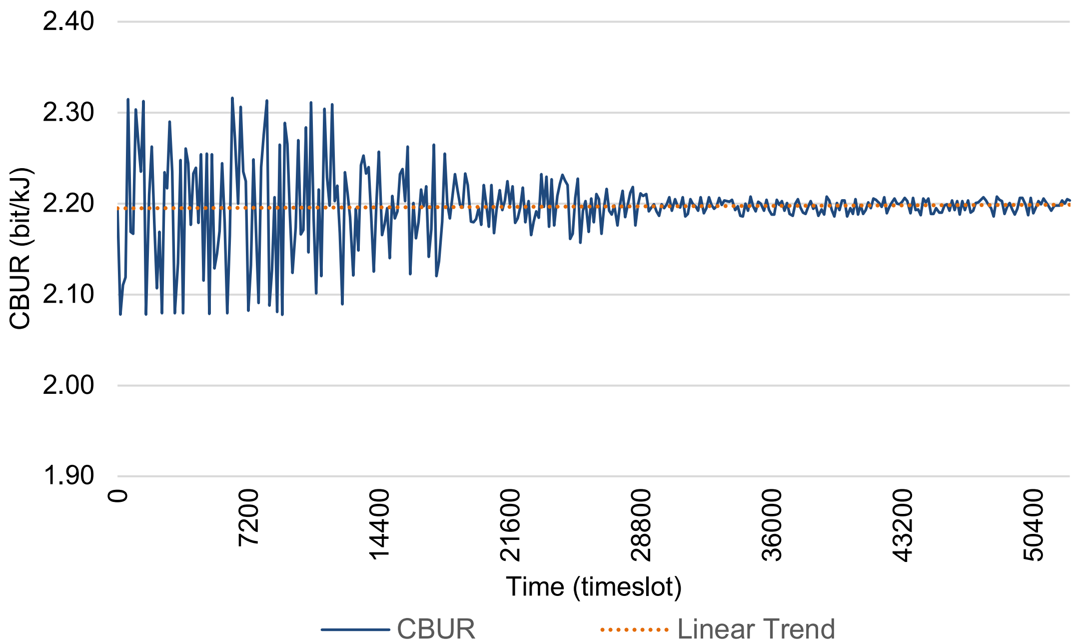


Figure 6.31 Time series and its trend of CBUR for the 50-node mesh IoT networks using Reinforcement Learning Routing Algorithm.

6.5 Summary

In this chapter, we have conducted a series of simulations to study the effects of 3 different parameters of reinforcement learning, namely the Boltzmann hyperparameter τ, the discount factor γ and the learning rate β. the performance of the network in three criteria including failure rate, energy efficiency and carrier band usage rate. 5 different possible values of each parameter are simulated in each episode of simulation. We then found the value with best result of each parameter.

We then compared the reinforcement learning routing algorithm directly against two benchmarks. We found that the reinforcement learning routing algorithm performed considerably better than the lower bound random routing algorithm. From all these data we found out the RL routing algorithm did help the network to meet the target we set in the chapter, whereas in most cases, the benchmark random routing algorithm failed to meet. Finally, we used the timeline of the simulation with the optimised parameters of the algorithm to illustrate the progress of the learning in failure rate, energy efficiency and CBUR. We concluded that employing RL in routing has a positive effect on the energy efficiency performance of the remote monitoring IoT networks.

Chapter 7 Conclusion

This thesis, we proposed a reinforcement learning routing algorithm that can reduce the failure rate, improve energy efficiency and carrier band usage rate of wireless mesh networks for rural environment monitoring. This kind of network addresses the need for a network that can provide connectivity effectively in the scale of kilometre squares in the rural without the pre-installed infrastructure. By introducing reinforcement learning, the algorithm is able to learn from the feedback information from each transmission and compare and store the information of the enviornment and the usage pattern of the power for better future routing decisions.

Firstly, we conducted extensive research and analysis of the environmental requirement for the remote monitoring networks and the available wireless technologies. We concluded that LPWANs with mesh topology were the choice for such tasks. We then introduced artificial intelligence into the routing using machine learning in the algorithm to enable long-term energy awareness. Such energy awareness can be reflected on the routing decision made by the algorithm as the algorithm learning about the network.

Secondly, we carried out a comprehensive literature review for related fields of study including wireless mesh networks, wireless sensor networks, machine learning in wireless sensor networks and other wireless mesh network routing solutions. We identified the need for a new routing algorithm that is specifically designed for remote monitoring sensor networks.

Furthermore, we modelled the remote monitoring networks with attributes of the key components including the channel, the nodes, the links, the transmissions, and the feedback that used for the machine learning. We also discussed and implemented the reinforcement learning method used in the algorithm. We also defined several key parameters to be studied in the further investigation.

Finally, we conducted a series of simulations to compare the effects of the key parameters on the effectiveness of the algorithm. We also compared the performance of the reinforcement learning routing algorithm with an upper bound and a lower bound benchmark. We proved the effectiveness of the algorithm by setting the performance target for algorithms to meet. We found that the RL routing has successfully met the target, whereas the benchmark algorithm has failed to meet the target.

We gathered a set of parameters with the best performance for the RL algorithm under the simulated environment. We found that the RL routing algorithm with these selected parameters to have a substantial improvement over the benchmark in all three criteria we studied. In summary, we concluded that the result has indicated that the proposed RL routing algorithm has addressed the need for the network.

7.1 Future work

Based on the research conducted in this thesis, there are several assumptions that can be studied further in the future. The assumptions include the interference, the channel parameters, charging cycles and transmission patterns. With the considerations on these assumptions, a more comprehensive model of the network can be built. Hence, more specified routing strategies can be developed and implemented for further researches.

Firstly, future works can consider the inclusion of both internal and external interferences. The interference has been considered zero in the simulations of this thesis, but in the real network situations, it exists. Internal Co-channel Interference can be studied to take simulation transmissions in adjacent nodes into consideration. The algorithm can divert the route selection to make better use the space in the network and even the energy consumption between all the nodes. Additionally, external Adjacent-Channel Interference should also be considered. Though the routing algorithm should be considered physical-layer technology-independent, the existence of congestion of the spectrum the underlying physical-layer technology will impact the energy consumption and performance of transmissions in the network. The learning process can consider that for more accurate channel prediction for the selection of routes.

Secondly, the fixed channel parameters used in the model, and the simulations also can be further studied. Location-specific parameters can be used in the simulation for more precise network planning. Dynamic channel parameters can also be introduced to the model to reflect the changing channel in the network.

Furthermore, the charging cycles we selected in this thesis is based on the average sunshine hours in the UK. The nodes may have other kinds of power sources, or different nodes may have different power cycles, such as grid-power nodes. With the introducing dynamic power model including the charging cycles, the different levels and the types of power source for each node, the power source planning can be carried out before the deployment of the network.

Finally, future researches of transmission patterns in the remote monitoring network can also benefit from the base of this thesis. We used randomly generated transmissions in the simulation with randomly picked SNs, DNs, and starting times. This may not reflect how data is flowing inside the network as the transmissions in the network in the real world usually have a certain pattern. This pattern may make the route projection more precise and the energy consumption planning more accurate as the learning algorithm can predict the direction of the transmission to provide better channel provisioning.

Many other types of works can be carried out on the basis of this thesis to make us build a better remote monitoring network and produce more meaningful data that benefit the world.

References

[1] B. F. Ronalds, “Francis Ronalds (1788–1873): The First Electrical Engineer?,” *Proceedings of the IEEE,* vol. 104, no. 7, pp. 1489 - 1498, 2016.

[2] N. Abramson, “THE ALOHA SYSTEM: another alternative for computer communications,” in *AFIPS Fall Joint Computing Conference*, Houston, 1970.

[3] J. Romkey, “Toast of the IoT: The 1990 Interop Internet Toaster,” *IEEE Consumer Electronics Magazine,* vol. 6, no. 1, pp. 116 - 119, 2016.

[4] D. Gordon, “The Trojan Room Coffee Machine,” Cambridge University, 22 August 2001. [Online]. Available: https://www.cl.cam.ac.uk/coffee/coffee.html . [Accessed 20 May 2019].

[5] K. Ashton, “That 'Internet of Things' Thing,” *RFiD Journal,* vol. 3, no. 1, p. 1, 2009.

[6] S. Dolan, “How the Internet of Things will transform consumerism, enterprises, and governments over the next five years,” Business Insider, 2018 19 07. [Online]. Available: https://www.businessinsider.com/iot-forecast-book-2018-7?r=US&IR=T. [Accessed 2019 05 15].

[7] I. Akyildiz, W. Su, Y. Sankarasubramaniam and E. Cayirci, “Wireless sensor networks: a survey,” *Computer Networks ,* vol. 38, no. 4, pp. 393-422, 2002.

[8] W. Dargie and C. Poellabauer, Fundamentals of Wireless Sensor Networks: Theory and Practice, Chichester: John Wiley & Sons, Ltd, 2011.

[9] S. L. Keoh, S. S. Kumar and H. Tschofenig, “Securing the Internet of Things: A Standardization Perspective,” *IEEE Internet of Things Journal,* vol. 1, no. 3, pp. 265 - 275, 2014.

[10] A. K. Simpson, F. Roesner and T. Kohno, “Securing vulnerable home IoT devices with an in-hub security manager,” in *IEEE International Conference on Pervasive Computing and Communications Workshops*, Kona, 2017.

[11] S. Raza, P. Misra, Z. He and T. Voigt, “Building the Interne tof Things with

bluetooth smart," *Ad Hoc Networks,* vol. 2017, no. 57, pp. 19-31, 2017.

[12] G. Vivek and M. Sunil, "Enabling IOT services using WIFI - ZigBee gateway for a home automation system," in *IEEE International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN)*, Kolkata, 2015.

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

[14] GSMA, "What is Mobile IoT?," GSMA, 19 06 2019. [Online]. Available: https://www.gsma.com/iot/mobile-iot/. [Accessed 19 06 2019].

[15] The GSMA, "NarrowBand – Internet of Things (NB-IoT)," The GSMA, 01 07 2019. [Online]. Available: https://www.gsma.com/iot/narrow-band-internet-of-things-nb-iot/. [Accessed 03 07 2019].

[16] The GSMA, "Long Term Evolution for Machines: LTE-M," The GSMA, 01 07 2019. [Online]. Available: https://www.gsma.com/iot/long-term-evolution-machine-type-communication-lte-mtc-cat-m1/. [Accessed 01 07 2019].

[17] S. Ullerstig, A. Zaidi and C. Kuhlins, "Know the difference between NB-IoT vs. Cat-M1 for your massive IoT deployment," Ericsson, 07 02 2019. [Online]. Available: https://www.ericsson.com/en/blog/2019/2/difference-between-NB-IoT-CaT-M1. [Accessed 01 07 2019].

[18] LoRa Alliance, "LoRaWAN® Specification v1.1," LoRa Alliance, 2017. [Online]. Available: https://lora-alliance.org/resource-hub/lorawanr-specification-v11. [Accessed 21 03 2019].

[19] The Things Network, "Ground breaking world record! LoRaWAN packet received at 702 km (436 miles) distance," The Things Network, 09 08 2017. [Online]. Available: https://www.thethingsnetwork.org/article/ground-breaking-world-record-lorawan-packet-received-at-702-km-436-miles-distance. [Accessed 03 07 2019].

[20] A. Augustin, J. Yi, T. Clausen and W. M. Townsley, "A Study of LoRa: Long Range & Low Power Networks for the Internet of Things," *Sensors (Basel),* vol. 16, no. 9, p. 1466, 2016.

[21] Sigfox, "Sigfox Technology Overview," Sigfox, 10 07 2019. [Online]. Available: https://www.sigfox.com/en/sigfox-iot-technology-overview. [Accessed 10 07 2019].

[22] K. Mekki, E. Bajic, F. Chaxel and F. Meyer, “A comparative study of LPWAN technologies for large-scale IoT deployment,” *ICT Express,* vol. 5, no. 1, pp. 1-7, 2019.

[23] Ofcom, “View mobile availability,” Ofcom, 10 07 2019. [Online]. Available: https://checker.ofcom.org.uk/mobile-coverage. [Accessed 10 07 2019].

[24] GSMA, “Deployment Map,” GSMA, 10 07 2019. [Online]. Available: https://www.gsma.com/iot/deployment-map/#GB. [Accessed 10 07 2019].

[25] EE, “CHECK YOUR MOBILE COVERAGE,” EE, 10 07 2019. [Online]. Available: https://ee.co.uk/why-ee/mobile-coverage. [Accessed 10 07 2019].

[26] Y. Shen, L. Cai, Y. Liu, X. Ding, X. Qiu, Y. Liu and K.-F. Tong, “On the Antenna for Long Range Low Power Geographical Monitoring IoT Network,” in *European Conference on Antennas and Propagation*, 2018, Nagoya.

[27] H. Wang, Z. Liu, L. Sun and Q. Luo, “Optimal Design of River Monitoring Network in Taizihe River by Matter Element Analysis,” *PLOS One,* vol. 10, no. 5, 2015.

[28] S.-Y. Park, J. H. Choi, S. Wang and S. S. Park, “Design of a water quality monitoring network in a large river system using the genetic algorithm,” *ecological modelling,* vol. 199, no. 2006, p. 289–297, 2006.

[29] S. Eaton, “Automatic Urban & Rural Network: Assessment of Site Classifications,” Air Information Resource, London, 2010.

[30] The Met Office, “UK Climate Averages - England,” The Met Office, 21 06 2019. [Online]. Available: https://www.metoffice.gov.uk/public/weather/climate/gcpsvg3nc. [Accessed 21 06 2019].

[31] Electronics Notes, “Bluetooth radio interface, modulation, & channels,” Electronics Notes, [Online]. Available: https://www.electronics-notes.com/articles/connectivity/bluetooth/radio-interface-modulation-channels.php. [Accessed 21 06 2019].

[32] M. Lukac, I. Stubailo, R. Guy, P. Davis, V. A. Puruhuaya, R. Clayton and D. Estrin, “First-class meta-data: a step towards a highly reliable wireless seismic network in Peru,” in *IPSN Workshop on Sensor Networks for Earth and Space Science Applications*, San Fransisco, 2009.

[33] A. World, “Customers who viewed SimpleWiFi Ultra Long Range WiFi Extender G2424 Directional Parabolic Grid… also viewed,” Amazon.com,

10 07 2019. [Online]. Available: https://www.amazon.com/Antenna-World-G2424-Directional-Parabolic/dp/B00NQGVMSE. [Accessed 10 07 2019].

[34] S. Kim, W. Choi, Y. Choi, J. Lee, Y. Han and I. Lee, “Downlink Performance Analysis of Cognitive Radio based Cellular Relay Networks,” in *International Conference on Cognitive Radio Oriented Wireless Networks and Communications*, Singapore, 2008.

[35] G. Miao, J. Zander, K. W. Sung and B. Slimane, Fundamentals of Mobile Data Networks, Cambridge: Cambridge University Press, 2016.

[36] K. K. Paliwal, S. Singh and S. Bajaj, “Rural communication enhancement using mobile ad-hoc network,” in *International Conference on Cloud Computing, Data Science & Engineering - Confluence*, 2017, Noida.

[37] J. Garside and S. Rogers, “EE launches UK's first 4G mobile network,” The Guardian, 29 10 2012. [Online]. Available: https://www.theguardian.com/technology/2012/oct/29/ee-launches-uk-4g-mobile-network. [Accessed 11 07 2019].

[38] A. Adhikary, X. Lin and Y.-P. E. Wang, “Performance Evaluation of NB-IoT Coverage,” in *IEEE 84th Vehicular Technology Conference*, Montreal, 2016.

[39] Sigfox, “Sigfox Services,” Sigfox, 11 07 2019. [Online]. Available: https://www.sigfox.com/en/solutions/sigfox-services. [Accessed 11 07 2019].

[40] I. F. Akyildiz and X. Wang, Wireless Mesh Networks, Chichester: John Wiley & Sons Ltd, 2009.

[41] D. Rodenas-Herraiz, A.-J. Garcia-Sanchez, F. Garcia-Sanchez and J. Garcia-Haro, “Current Trends in Wireless Mesh Sensor Networks: A Review of Competing Approaches,” *Sensors (Basel),* vol. 13, no. 5, p. 5958–5995, 2013.

[42] S. Srivathsan, N. Balakrishnan and S. Iyengar, “Scalability in Wireless Mesh Networks,” in *Guide to Wireless Mesh Networks*, London, Springer-Verlag, 2009, pp. 325-347.

[43] S. Sampaio, P. Souto and F. Vasques, “A review of scalability and topological stability issues in IEEE 802.11s wireless mesh networks deployments,” *International Journal of Communication Systems,* vol. 29, no. 4, pp. 671-693, 2016.

[44] M. P. Clark, Data Networks, IP and the Internet: Protocols, Design and Operation, New Jersey: John Wiley & Sons, 2003.

[45] O. S. U. C. O. U. M. Goyal Dept. of Comput. & Inf. Sci., K. Ramakrishnan and W.-c. Feng, “Achieving faster failure detection in OSPF networks,” in *IEEE International Conference on Communications*, Anchorage, 2003.

[46] Internet Engineering Task Force, *Optimized Link State Routing Protocol (OLSR), RFC 3636,* Internet Engineering Task Force, 2003.

[47] J. Moon, “The Dartmouth College Artificial Intelligence Conference: The Next Fifty years,” *AI Magazine,* vol. 27, no. 4, pp. 87-89, 2006.

[48] D. Crevier, AI: The Tumultuous Search for Artificial Intelligence, New York: BasicBooks, 1993.

[49] S. J. Russell and P. Norvig, Artificial Intelligence: A Modern Approach, New Jersey: Prentice Hall, 2003.

[50] C. Bishop, Pattern Recognition and Machine Learning, Cambridge: Springer, 2006.

[51] S. J. Russell and P. Norvig, Artificial Intelligence: A Modern Approach, New Jersey: Prentice Hall, 2009.

[52] G. Hinton and T. J. Sejnowski, Unsupervised Learning: Foundations of Neural Computation, The MIT Press: Cambridge, 1999.

[53] L. P. Kaelbling, M. L. Littman and A. W. Moore, “Reinforcement Learning: A Survey,” *Journal of Artificial Intelligence Research,* vol. 4, no. 1996, pp. 237-285, 1996.

[54] D. Bertsekas, Dynamic Programming and Optimal Control, Vol. II, Nashua: Athena Scientific, 2012.

[55] A. Barto and R. S. Sutton, Reinforcement Learning: An Introduction, Second Edition, London: The MIT Press, 2017.

[56] The IEEE, “IEEE 802 LAN/MAN Standards Committee,” The IEEE, 10 07 2019. [Online]. Available: http://www.ieee802.org/. [Accessed 20 07 2019].

[57] G. R. H. R. A. UNIV, D. Denteneer, S. Max, R. Taori, J. Cardona, L. Berlemann and B. Walke, “IEEE 802.11s: The WLAN Mesh Standard,” *IEEE Wireless Communications ,* vol. 17, no. 1, pp. 104-111, 2010.

[58] SmartThings Inc., “Z-Wave and ZigBee FAQ,” SmartThings Inc., 2017. [Online]. Available: https://support.smartthings.com/hc/en-gb/articles/208672926-Z-Wave-and-ZigBee-FAQ#Range. [Accessed 21 07 2019].

[59] F. Dacosta, “First, Second and Third Generation Mesh Architectures,” in *Emerging Technologies in Wireless LANs*, Cambridge, 2007.

[60] P. Gardner-Stephen, “The Serval Project : Practical Wireless Ad-Hoc Mobile Telecommunications,” 22 07 2011. [Online]. Available: http://developer.servalproject.org/files/CWN_Chapter_Serval.pdf. [Accessed 21 08 2018].

[61] P. Gardner-Stephen, “MeshMS: Ad Hoc Data Transfer within Mesh Network,” *Internationl Journal of Communications, Network and System Sciences,* vol. 05, no. 08, pp. 496-504, 2012.

[62] J. Jun and M. L. Sichitiu, “The Nominal Capacity of Wireless Mesh Networks,” *IEEE Wireless Communications,* vol. 10, no. 5, pp. 8-14, 2003.

[63] Y. Bejerano, “Efficient integration of multihop wireless and wired networks with QoS constraints,” *IEEE/ACM Transactions on Networking,* vol. 12, no. 6, pp. 1064-1078, 2004.

[64] H. Adi, K. Kom, P. S. Sakti, S. Eko Kom, M. Kom and S. T. K. Amron, “Data gathering in rural area using publish/subscribe over Wireless Mesh Network,” in *Proceedings - 2017 International Conference on Sustainable Information Engineering and Technology, SIET 2017*, Malang, Indonesia, 2018.

[65] Y. Zhou-Kangas, “Modeling of a wireless mesh network,” UNIVERSITY OF JYVÄSKYLÄ, Jyväskylä, 2014.

[66] H. Jiao and F. Y. Li, “Cooperative MAC design in multi-hop wireless networks-part II: When source and destination are two-hops away from each other,” *Wireless Personal Communications,* vol. 57, no. 3, pp. 351-363, 2011.

[67] K. Jain, J. Padhye, V. Padmanabhan and L. N. Qiu, “Impact of interference on multi-hop wireless network performance,” *Wireless Networks,* vol. 11, no. 4, pp. 471-487, 2005.

[68] T. Bouguera, J. F. Diouris, J. J. Chaillout, R. Jaouadi and G. Andrieux, “Energy consumption model for sensor nodes based on LoRa and LoRaWAN,” *Sensors (Switzerland),* vol. 18, no. 7, pp. 1-24, 2018.

[69] A. J. Wixted, P. Kinnaird, H. Larijani, A. Tait, A. Ahmadinia and N. Strachan, “Evaluation of LoRa and LoRaWAN for wireless sensor networks,” in *IEEE SENSORS*, Orlando, 2016.

[70] M. Abo-Zahhad, M. Farrag, A. Ali and O. Amin, “An energy consumption model for wireless sensor networks,” in *nternational Conference on Energy Aware Computing Systems & Applications*, Cairo, 2015.

[71] J. Song and Y. K. Tan, “Energy consumption analysis of ZigBee-based energy harvesting wireless sensor networks,” in *IEEE International Conference on Communication Systems (ICCS)*, Singapore, 2012.

[72] S. Kurt and B. Tavli, “Path-Loss Modeling for Wireless Sensor Networks: A review of models and comparative evaluations.,” *IEEE Antennas and Propagation Magazine ,* vol. 59, no. 1, pp. 18-37, 2017.

[73] O. Georgiou and U. Raza, “Low Power Wide Area Network Analysis: Can LoRa Scale?,” *IEEE Wireless Communications Letters,* vol. 6, no. 2, pp. 162-165, 2017.

[74] L. Gregora, L. Vojtech and M. Neruda, “Indoor Signal Propagation of LoRa Technology,” in *International Conference on Mechatronics - Mechatronika (ME)*, Prague, 2106.

[75] K. Herring, J. T. Holloway, D. W. Staelin and D. W. H. Bliss, “Path-Loss characteristics of urban wireless channels,” *IEEE Transactions on Antennas and Propagation,* vol. 58, no. 1, pp. 171-177, 2010.

[76] T. Rappaport, Wireless Communications: Principles and Practice, London: Prentice Hall, 2001.

[77] P. Murali, A. Challa, M. R. Kasyap and C. Hota, “A generalized energy consumption model for Wireless Sensor Networks,” in *International Conference on Computational Intelligence and Communication Networks*, Bhopal, 2010.

[78] T. Ruan, Z. J. Chew and M. Zhu, “Energy-Aware Approaches for Energy Harvesting Powered Wireless Sensor Nodes,” *IEEE Sensors Journal,* vol. 17, no. 7, pp. 2165-2173, 2017.

[79] C. Antonopoulos, A. Prayati, T. Stoyanova, C. Koulamas and G. Papadopoulos, “Experimental evaluation of a WSN platform power consumption,” in *IEEE International Symposium on Parallel & Distributed Processing*, Rome, 2009.

[80] J. Navarro-Ortiz, S. Sendra, P. Ameigeiras and J. M. Lopez-Soler, “Integration of LoRaWAN and 4G/5G for the Industrial Internet of Things,” *IEEE Communications Magazine,* vol. 56, no. 2, pp. 60-67, 2018.

[81] S. V. Nandury and B. A. Begum, “Smart WSN-based ubiquitous architecture for smart cities,” in *2015 International Conference on Advances in Computing, Communications and Informatics, ICACCI 2015*, Kochi, India, 2015.

[82] Y. Mehmood, C. Gorg, M. Muehleisen and A. Timm-Giel, “Mobile M2M communication architectures, upcoming challenges, applications, and future directions,” *Eurasip Journal on Wireless Communications and Networking,* vol. 2015, no. 1, pp. 1-37, 2015.

[83] M. Lauridsen, H. Nguyen, B. Vejlgaard, I. Kovacs, P. Z. Mogensen and M. Sorensen, “Coverage Comparison of GPRS, NB-IoT, LoRa, and SigFox in a 7800 km Area,” in *IEEE Vehicular Technology Conference*, Montréal, 2016.

[84] A. Dongare, C. Hesling, K. Bhatia, A. Balanuta, R. L. Pereira, B. Iannucci and A. Rowe, “OpenChirp: A Low-Power Wide-Area Networking architecture,” in *IEEE International Conference on Pervasive Computing and Communications Workshops*, Kona, 2017.

[85] C. E. Jones, K. M. Sivalingam, P. Agrawal and J. C. Chen, “A survey of energy efficient network protocols for wireless networks,” *Wireless Networks,* vol. 7, no. 4, pp. 343-358, 2001.

[86] S. Benaissa, D. Plets, E. Tanghe, J. Trogh, L. Martens, L. Vandaele, L. Verloock, F. Tuyttens, B. Sonck and W. Joseph, “Internet of animals: characterisation of LoRa sub-GHz off-body wireless channel in dairy barns,” *Electronics Letters,* vol. 53, no. 18, pp. 1281-1283, 2017.

[87] Z. Fan, “A distributed energy-efficient and delay-aware data gathering protocol for wireless sensor networks,” in *International Conference on Networks Security, Wireless Communications and Trusted Computing*, Wuhan, 2009.

[88] C. Jiang, H. Zhang, Y. Ren, Z. Han, K.-C. Chen and L. Hanzo, “Machine Learning Paradigms for Next-Generation Wireless Networks,” *IEEE Wireless Communications,* vol. 24, no. 2, pp. 98-105, 2017.

[89] M. A. Alsheikh, S. Lin, D. Niyato and H.-P. Tan, “Machine Learning in Wireless Sensor Networks: Algorithms, Strategies, and Applications,” *IEEE Communications Surveys & Tutorials,* vol. 16, no. 4, pp. 1996-2018, 2014.

[90] Y. Chu, P. D. Mitchell and D. Grace, “ALOHA and Q-Learning based medium access control for Wireless Sensor Networks,” in *International*

Symposium on Wireless Communication Systems (ISWCS), Paris, 2012.

[91] R. D. G. H. C. L. T.-S. K. T. P. Mo Sha, “Self-Adapting MAC Layer for Wireless Sensor Networks,” in *IEEE 34th Real-Time Systems Symposium* , Vancouver, 2013.

[92] J. A. Boyan and M. L. Littman, “Packet routing in dynamically changing networks: a reinforcement learning approach,” in *Proceedings of the 6th International Conference on Neural Information Processing Systems*, Denver, 1993.

[93] T. Hu and Y. Fei, “QELAR: A Machine-Learning-Based Adaptive Routing Protocol for Energy-Efficient and Lifetime-Extended Underwater Sensor Networks,” *IEEE Transactions on Mobile Computing ,* vol. 9, no. 6, pp. 796-809, 2010.

[94] P. Wang and T. Wang, “Adaptive Routing for Sensor Networks using Reinforcement Learning,” in *The Sixth IEEE International Conference on Computer and Information Technology* , Seoul, 2006.

[95] X. Liang, I. Balasingham and S.-S. Byun, “A reinforcement learning based routing protocol with QoS support for biomedical sensor networks,” in *First International Symposium on Applied Sciences on Biomedical and Communication Technologies*, Aalborg, 2008.

[96] M. Boushaba, A. Hafid and A. Belbekkouche, “Reinforcement learning-based best path to best gateway scheme for wireless mesh networks,” in *IEEE 7th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, Wuhan, 2011.

[97] P. Mitchell, T. Jiang and D. Grace, “Efficient exploration in reinforcement learning-based cognitive radio spectrum sharing,” *IET Communications,* vol. 5, no. 10, pp. 1309-1317, 2011.

[98] M. Bennis, S. Perlaza, P. M. Blasco, Z. Han and H. V. Poor, “Self-organization in small cell networks: A reinforcement learning approach,” *IEEE Transactions on Wireless Communications,* vol. 12, no. 7, pp. 3202-3212, 2013.

[99] “Adaptive exploration strategies for reinforcement learning,” in *International Conference on System Science and Engineering*, Ho Chi Minh City, 2017.

[100] M. Guo, Y. Liu and J. Malec, “A New Q-Learning Algorithm Based on the Metropolis Criterion Maozu,” *IEEE transactions on Sistems, Man, and*

Cybernetics, vol. 34, no. 5, pp. 2140-2143, 2004.

[101] J. Vazifehdan, R. V. Prasad and I. Niemegeers, “Energy-efficient reliable routing considering residual energy in wireless ad hoc networks,” *IEEE Transactions on Mobile Computing,* vol. 13, no. 2, pp. 434-447, 2014.

[102] P. Jacquet, P. Muhlethaler, T. Clausen, A. Laouiti, A. Qayyum and L. Viennot, “Optimized link state routing protocol for ad hoc networks,” in *IEEE International Multi Topic Conference, 2001*, Lahore, 2001.

[103] G. Pei, M. Gerla and T.-W. Chen, “Fisheye state routing: a routing scheme for ad hoc wireless networks,” in *IEEE International Conference on Communications.*, New Orleans, 2000.

[104] S. Murthy and J. J. Garcia-Luna-Aceves, “An efficient routing protocol for wireless networks,” *Mobile Networks and Applications - Special issue: routing in mobile communications networks,* vol. 1, no. 2, pp. 183-197, 1996.

[105] C. E. Perkins and P. Bhagwat, “Highly dynamic Destination-Sequenced Distance-Vector routing (DSDV) for mobile computers,” in *Proceedings of the conference on Communications architectures, protocols and applications*, London, 1994.

[106] C. Perkins, E. Belding-Royer and S.Das, “Ad hoc On-Demand Distance Vector (AODV) Routing,” RFC, 2003.

[107] V. Park and S. Corson, “Temporally-Ordered Routing Algorithm (TORA) Version 1 Functional Specification,” 20 07 2001. [Online]. Available: https://tools.ietf.org/html/draft-ietf-manet-tora-spec-04. [Accessed 12 07 2019].

[108] D. Johnson, “The Dynamic Source Routing Protocol (DSR) for Mobile Ad Hoc Networks for IPv4,” 02 2007. [Online]. Available: https://tools.ietf.org/html/rfc4728. [Accessed 11 07 2019].

[109] S. Basagni, I. Chlamtac, V. R. Syrotiuk and B. A. Woodward, “A distance routing effect algorithm for mobility (DREAM),” in *4th annual ACM/IEEE international conference on Mobile computing and networking*, Dallas, 1998.

[110] Open-mesh, “B.A.T.M.A.N.,” Open-mesh, 26 06 2019. [Online]. Available: https://www.open-mesh.org/projects/open-mesh/wiki. [Accessed 10 07 2019].

[111] IETF, “Babel routing protocol (babel),” 11 07 2019. [Online]. Available:

https://datatracker.ietf.org/wg/babel/about/. [Accessed 11 07 2019].

[112] D. Murray, “An experimental comparison of routing protocols in multi hop ad hoc networks,” in *Australasian Telecommunication Networks and Applications Conference*, 2010.

[113] N. K. K. S. Florian Zeiger, “Commanding Mobile Robots via Wireless Ad-Hoc Networks - A Comparison of Four Ad-Hoc Routing Protocol Implementations,” in *IEEE International Conference on Robotics and Automation*, Pasadena, 2008.

[114] F. Tang, B. Mao, Z. M. Fadlullah, N. Kato, O. Akashi, T. Inoue and K. Mizutani, “On Removing Routing Protocol from Future Wireless Networks: A Real-time Deep Learning Approach for Intelligent Traffic Control,” *IEEE Wireless Communications,* vol. 25, no. 1, pp. 154 - 160, 2018.

[115] P. R. More and S. V. Sankpal, “Energy aware routing using energy efficient routing protocol in wireless Ad hoc network,” in *International Conference on Electrical, Electronics, and Optimization Techniques, ICEEOT 2016*, 2016, Chennai.

[116] S. D. Nandakumar and K. A. Parthasarathi, “Traffic and energy aware routing in wireless ad hoc network,” in *Second International Conference on Current Trends In Engineering and Technology - ICCTET 2014*, Coimbatore, 2014.

[117] H. Bo, W. Muqing, Z. Min and L. Wenxing, “An energy aware routing algorithm for software defined wireless sensor networks,” in *IEEE/CIC International Conference on Communications in China (ICCC)*, Qingdao, 2017.

[118] J. Hao, Z. Yao, K. Huang, B. Zhang and C. Li, “An energy-efficient routing protocol with controllable expected delay in duty-cycled wireless sensor networks,” in *IEEE International Conference on Communications*, Budapest, 2013.

[119] W. Sun, Y. Song and M. Chen, “A load-balanced and energy-aware routing metric for wireless multimedia sensor networks,” in *IET 3rd International Conference on Wireless, Mobile and Multimedia Networks (ICWMNN 2010)*, Beijing, 2010.

[120] S. He, K. Xie, W. Chen, D. Zhang and J. Wen, “Energy-Aware Routing for SWIPT in Multi-Hop Energy-Constrained Wireless Network,” *IEEE Access,*

vol. 6, pp. 17996-18008, 2018.

[121] (. U. o. T. M. o. E. Degan Zhang Key Laboratory of Computer Vision and System, G. Li, K. Zheng, X. Ming and Z.-H. Pan, “An Energy-Balanced Routing Method Based on Forward-Aware Factor for Wireless Sensor Networks,” *IEEE Transactions on Industrial Informatics,* vol. 10, no. 1, pp. 766 - 773, 2014.

[122] C. J. Glass and L. M. Ni, “The turn model for adaptive routing,” *SIGARCH Computer Architecture News,* vol. 20, no. 2, pp. 278-287, 1992.

[123] M. Musolesi, *Context-aware Adaptive Routing for Delay Tolerant Networking Mirco Musolesi,* London: UCL, 2007.

[124] A. Koliousis and J. Sventek, “Proactive vs reactive routing for wireless sensor networks,” Department of Computing Science, University of Glasgow, Glasgow, 2007.

[125] J. Broch, D. Maltz and D. A. Johnson, “A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols,” in *International Conference on Mobile Computing and Networking*, Dallas, 1998.

[126] Amazon.co.uk, “SODIAL 1W 5.5V Solar Cell Epoxy Polycrystalline Solar Panel+Clip For Charging 3.7V Battery System Toy LED Light Study 95*95MM,” Amazon.co.uk, 21 06 2019. [Online]. Available: https://www.amazon.co.uk/SODIAL-Polycrystalline-Charging-Battery-System/dp/B07BGSTHP3/ref=pd_sbs_86_4/257-4218096-0714304?_encoding=UTF8&pd_rd_i=B07BGSTHP3&pd_rd_r=e3e8c144-8933-11e9-9a9a-49585dea68e8&pd_rd_w=c80Nk&pd_rd_wg=YZf4m&pf_rd_p=18edf98b-139a-41e. [Accessed 21 06 2019].

[127] Battery University, “BU-104a: Comparing the Battery with Other Power Sources,” Cadex Electronics Inc, 28 03 2019. [Online]. Available: https://batteryuniversity.com/learn/article/comparing_the_battery_with_other_power_sources. [Accessed 24 05 2019].

[128] Semtech, “Semtech SX1276/77/78/79 Datasheet,” Semtech, 2019.

[129] DFRobot, “Gravity: Analog TDS Sensor/Meter for Arduino,” DFRobot, 10 07 2019. [Online]. Available: https://www.dfrobot.com/product-1662.html?gclid=Cj0KCQjwpavpBRDQARIsAPfTwiyjV9W_WBCs5hRjLwyziil6wLh-VaMtiO0EHipt5qHlEWJAXaL5AqMaAlh4EALw_wcB. [Accessed 10

07 2019].

[130] J. F. Kurose and K. W. Ross, Computer Networking: A Top-Down Approach, New York: Pearson, 2012.

[131] Y. Liu, K.-F. Tong and K.-K. Wong, “Reinforcement learning based routing for energy sensitive wireless mesh IoT networks,” *IET Electronics Letters,* 2019.

[132] D. Furlan, “Improving B.A.T.M.A.N. Routing Stability and Performance,” University of Trento, Trento, 2011.

[133] J. L. McClelland, “Temporal-Difference Learning,” in *Explorations in Parallel Distributed Processing: A Handbook of Models, Programs, and Exercises* , San Jose, Standford University, 2015, pp. 193-216.

[134] I. Goodfellow, Y. Bengio and A. Courville, Deep Learning (Adaptive Computation and Machine Learning series), New York: The MIT Press, 2016.

[135] V. François-Lavet, R. Fonteneau and D. Ernst, “How to Discount Deep Reinforcement Learning: Towards New Dynamic Strategies,” in *Deep Reinforcement Learning Workshop*, Montreal, 2015.

[136] D. J. K. Anurag Kumar, “Wireless Communication: Concepts, Techniques, Models,” in *Wireless Networking*, London, Elsevier , 2008, pp. 15-51.

[137] Huawei, “HUAWEI Mate 20 Pro Specifications,” Huawei, 01 07 2019. [Online]. Available: https://consumer.huawei.com/uk/phones/mate20-pro/specs/. [Accessed 01 07 2019].

[138] P. M. Dang Nguyen, “Analysis of MPR Selection in the OLSR Protocol,” in *21st International Conference on Advanced Information Networking and Applications Workshops (AINAW'07)*, Niagara Falls, 2007.

[139] R. Dobkin, C. Nelson, D. O’Neill, S. Pietkiewicz, T. Skovmand and M. Wilcox, “Power conditioning for notebook and palmtop systems,” in *Analog Circuit Design - A Tutorial Guide to Applications and Solutions*, London, Elsevier, 2011, pp. 18-34.

[140] Y. Liu, K.-F. Tong, X. Qiu and Y. Liu, “Wireless Mesh Networks in IoT Networks,” in *Conference: 2017 International Workshop on Electromagnetics: Applications and Student Innovation Competition (iWEM)* , London, 2017.

[141] S. He, K. Xie, W. Chen, D. Zhang and J. Wen, “Energy-Aware Routing for

SWIPT in Multi-Hop Energy-Constrained Wireless Network," *IEEE Access,* vol. 6, pp. 17996-18008, 2018.