UNIVERSITY OF SOUTHAMPTON

FACULTY OF PHYSICAL SCIENCES AND ENGINEERING

Electronics and Computer Science

Adaptive Dynamic Packet Routing on Internet Networks based on Reinforcement Learning Approach

by

Tanyaluk Deeka

Thesis for the degree of Master of Philosophy

July 2017

UNIVERSITY OF SOUTHAMPTON

ABSTRACT

FACULTY OF PHYSICAL SCIENCES AND ENGINEERING
Electronics and Computer Science

Master of Philosophy

ADAPTIVE DYNAMIC PACKET ROUTING ON INTERNET NETWORKS BASED ON REINFORCEMENT LEARNING APPROACH

by Tanyaluk Deeka

In this thesis, we concern the problem of packet routing on the large scale networks like Internet which is a complex optimization due to a fast-growing, increasingly complex network of connected devices whereas the network models are conceptual. First, three synthesis Internet network models are proposed which are a random network, a random network with preferential attachment (PA) and a heuristically optimal topology (HOT) models. While Internet network models are constructed based on simplistic connections between nodes and connections formed sequentially by preferential attachment, the HOT model enhances to be more reflective of the Internet's router level topology. Since, all traffic on the network has to be transmitted by traveling through interconnected routers. In addition, the volume of traffic has an effect on traffic congestion on different network connectivity as a result of complex routing optimization problems. Hence, Reinforcement learning (RL) is applied in this thesis because it has been introduced to solve complex and adaptive optimization problems. In particular, Q-routing which is an application of RL, is interested in the routing problem, but it is successful in only small various distributed wireless networks. Hence, the size of network is extended to be more realistic, and connectivity properties as seen in the Internet is represented as the aim of Q-routing on these networks is to support massive number of users. In addition, the Q-routing in this thesis is also applied on realistic network; JANET. Therefore, the results of Q-routing on the large scale network like Internet are represented by dealing with adaptive packet routing is embedded on all nodes in these networks which aims to optimize routing problem. Furthermore, the effect of the different network connectivity is also represented in how much the Q-routing can improve the network performance when the networks are subject to increasing amounts of traffic.

Contents

List of Figures

List of Tables

Declaration of Authorship

I, Tanyaluk Deeka , declare that the thesis entitled *Adaptive Dynamic Packet Routing on Internet Networks based on Reinforcement Learning Approach* and the work presented in the thesis are both my own, and have been generated by me as the result of my own original research. I confirm that:

- this work was done wholly or mainly while in candidature for a research degree at this University;
- where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated;
- where I have consulted the published work of others, this is always clearly attributed;
- where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work;
- I have acknowledged all main sources of help;
- where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself;
- none of this work has been published before submission

Signed:.....

Date:.....

Acknowledgements

I would like to express my very great appreciation to my first supervisor, Professor Mahesan Niranjan, my research supervisor for his professional and constructive suggestions during the planning and development of this report and keeping my progress on schedule.

I would also like to thank my second supervisor, Dr Bing Chu, for his support, knowledge, and how to solve problems in an efficient manner.

Extending my thanks to all the colleagues: Chathurika Dharmagunawardana, Tayyaba Azim, Xin Liu, Yawwani Gunawardana, Jianhau Xiong, and Boriboon Deeka for supporting me to lose my fear of being wrong. In addition, I have the will power to continue with my higher education because I believe that my supervisors and my colleagues support me to continue doing good things.

Finally, I wish to thank my parents and my family for their support and encouragement throughout my study.

Nomenclature

RL	Reinforcement learning
UK	United Kingdom
$JANET$	Joint Academic Network
PA	Random network with Preferential Attachment
HOT	Heuristically Optimal Topology
PAC	Probability Approximately Correct
TD	Temporal-Difference
DP	Dynamic Programming
t	Time Steps of Simulation
$\mathbf{A}$	A Set of Actions
$\mathbf{S}$	A Set of States
d	Destination Node
δ	Transmission Delay
η	the Learning Rate
p_k	the Probability of a Node Having a Number of Link
N	a Number of Nodes on the Network
ISP	Internet Service Provider
L	Average Number of Packets in the Queueing Network
W	Average Waiting Time of a Packet in the Network
λ	Average Arrival Rate of Packets per Unit Time
$FCFS$	First Come First Served
$LCFS$	Last Come First Served
μ	the Exponential Service Rate
IID	Independent and Identically Distributed Random Variables
DST	the Deep Sea Treasure World
γ	the Discount Factor
ϵ	Probability of Random Action

Chapter 1

Introduction

1.1 Background

Routing is a common optimization problem with networked traffic systems, ranging from systems for delivery of goods over road networks to communicating packets of data over electronic networks. In communication networks, router level algorithms are usually designed to be static, with routing tables stored at every node fixed by computing shortest paths between pairs of source and destination nodes (Tanenbaum and Wetherall, 2011). Dijkstra's shortest path algorithm and the distance vector routing algorithm are common variants of optimization algorithms to compute such routing tables.

With the need to operate networks at ever increasing traffic loads, and the increasing use of more flexible network environments such as wireless communication systems (and the recently popularized notion of the'Internet of Things' (Atzori et al., 2010)), there is a need for more adaptive (or dynamic) approach to routing in which traffic and network connectivity changes can dynamically determine the routing table. Such thinking leads to more complex optimization problems.

Over the years, interest in the use of artificial intelligence techniques to solve complex and adaptive optimization problems has attracted much interest. Early work in this area includes the use of the Hopfield network as a basis to formulate the Travelling Salesman problem (Hopfield, 1984; Aiyer et al., 1990; Smith, 1999). Reinforcement learning, a branch of machine learning (Sutton and Barto, 2011), has been a particularly successful technique for formulating and solving difficult optimization problems. An example of this is the elevator arrival optimization formulated as a learning problem in (Crites and Barto, 1996). Moreover, it applied to group image elements on recurrent neural networks for providing a relationship between contour linking and curve-tracing (Brosch et al., 2015).

In the context of routing in communication networks, Boyan and Littman introduced Q-routing, an application of reinforcement learning, in its original formulation as Watkin's Q-learning Watkins and Dayan (1992). This work demonstrated that an adaptive routing table could be learnt and the performance of the network as measured by the average time delay to deliver packets can be improved under heavy loads. However, Boyan et al.'s demonstration was on a very small network of 36 nodes the topology which resembled a grid. Furthermore, the Q-routing algorithm has been proposed on small various distributed wireless networks which the size of these networks less than 250 nodes in order to improve network throughput and reduce path energy cost (Haraty and Traboulsi, 2012; Maleki et al., 2014; Bhorkar et al., 2012; Lin and van der Schaar, 2011; Santhi et al., 2011; Hu and Fei, 2010; Dowling et al., 2005).

While Boyan et al.'s work (Boyan and Littman, 1994) is over two decades old, subsequent work on the topic by several authors neither addressed larger networks nor topologies with different connectivities. Since, communication networks have to increase its sizes, and develop its connectivity structures in order to support massive number of users. Hence an empirical evaluation of the performance of Q-routing on networks of realistic sizes and connectivity properties as seen in the Internet is in order. This is the task undertaken in the present study.

In this thesis, we consider three types of network topologies with the number of nodes set at 500 and the number of connections in the network set at 5000. For a network of this size, we designed it based on IBM red book which claimed that 500 nodes are large size networks (Murhammer et al., 1999). We construct different network topologies with random connections between nodes and connections formed sequentially by preferential attachment (Batagelj and Brandes, 2005). We also consider a novel network architecture, known as a heuristically optimized topology due to Li et al (Li et al., 2004) which is designed to be more reflective of the Internet's router level topology than a preferential attachment network. Since, all traffic from the network edge has to transmit via interconnected routers. Further, we consider a realistic network, the Janet network, linking academic establishments in the United Kingdom. By doing these, we show in this thesis that the Q-routing approach scales to larger problems of adaptive routing. Our comparison also shows the effect of the different topologies in how much Q-routing can help improve performance when the networks are subject to increasing amounts of traffic.

1.2 Internet Routing Problem

A communication network consists of nodes which share information among them via packet transmission (Tanenbaum and Wetherall, 2011). However, all pairs of nodes are not directly connected as a result of taking time to deliver a number of packets because

they are delivered by taking a number of hops over intermediate nodes. Since, the packet is transmitted between source and destination node on the network. Hence, there are two major factors which have an effect on the total packet delivery time which are the waiting time and speed of the link (Tanenbaum and Wetherall, 2011). The first factor is waiting time especially in the intermediate nodes as many packets have to spend their time in the queues of these nodes while they are traveling from source to destination. Thus, the optimal routing should consider how to select nodes for packet transmission based on a minimum packet delivery time which those nodes should take a minimum packet delivery time, and also have small queues for waiting to be served. The latter factor depends on the speed of the links connecting between node and its neighbors which are not always equal (Tanenbaum and Wetherall, 2011). In addition, the length of the route also has an effect on packet traveling time through the network, and it will be a critical route because of traffic congestion.

Since, there could be multiple paths for packet transmission between source and its destination as a consequence of the total packet delivery time which reflects on routing policy selection. Thus, the optimal route has to consider the minimal total delivery time from a given source node to a given destination node in the network which should include the queue lengths of all their intermediate nodes. For example, the Dijkstra shortest path algorithm is applied to find the best route by selecting the minimal total packet delivery time of a packet from source to destination (Dijkstra, 1959). However, it is not an appropriate approach in practice way because the entire routing information of every nodes on the network being employed for making each routing decision.

Hence, each node on the network should have a process to compute optimal route to reach its destination before sending a packet out. In addition, an individual source node can select optimal routes to send packets, if it knows complete information on the entire network. In practice, some links or nodes on the Internet networks might go down and come up because the topology of the network is not fixed at all times. In addition, there is a significant overhead as each packet carries a lot of routing information. Hence, a selection of optimal routes based on knowing entire of the routing information on the network is not beneficial for the practical routing on the Internet networks.

Moreover, the intermediate nodes should make routing decisions based on their local routing information for forwarding packets to reach a given destination as quickly as possible via a believable neighboring node. There are some requirements that have been identified for this approach. For example, each of its neighboring nodes in the network can estimate delay time to send a packet via itself. A node should have mechanism to make routing decisions based on current traffic condition of the network that the updating of routing information mechanism should reflect in the overview traffic on the network.

Thus, it has led to adaptive routing algorithms such as Q-Routing for making local routing decisions to obtain optimal routes especially on dynamic changing networks (Boyan and Littman, 1994). In addition, the local routing information is applied to get overview of the traffic on the network, and make a routing decision in order to get a routing policy.

1.3 Motivation and Research Challenges

Although, the shortest path is the simplest routing algorithm to find minimum routes for packet transmission, it is suitable just only static network which should have not traffic congestion. However, if the number of packets are introduced increasingly into the network while nodes on the network can serve these packets at a constant rate which is lower than an arrival rate, and leads to have traffic congestion because the number of packets are hold on the intermediate nodes at the same route for packet serving, and leading to take a longer time for packet transmission.

Hence, adaptive routing algorithms are appropriate to be employed for avoiding traffic congestion along the popular route, and they should find multi-paths for packet transmission which should take a minimum time for packet transmission. Moreover, the selected route may be take a longer hops than the popular route, but the number of packets can be served suddenly without waiting time.

According to Boyan and Littman (1994), they suggested that each node in dynamic routing environment should have mechanism to maintain its routing information in order to make routing decisions for packet transmission, and its routing information should be updated based on its current traffic condition on the network.

Reinforcement learning (RL) is one of optimization methods which can achieve its goal by observing environment, and then giving reward feedback signal to make decision. There are many researchers interested in applying RL on various optimization problems such as routing optimization on distributed wireless network (Al-Rawi et al., 2015). For example, Bhorkar et al. (2012) claimed that a RL framework can apply on wireless ad hoc networks which

In this thesis, we will motivate to apply Q-routing on various large network connectivity like Internet networks which aims to determine optimal routes for packet forwarding which should learn to avoid traffic congestion while there are large number of packets are introduced to the network continuously.

1.4 Research Contributions

In this thesis, we propose the Q-routing on large scale synthetic Internet networks which has not been addressed, and apply to a real United Kingdom (UK) education network; Joint Academic Network (JANET). The main contributions of this thesis are summarized as follows:

- First, the Q-routing is proposed on small scale grid and random networks which a number of nodes is less than 80. These networks are employed to study and understand the process of routing making decision of the Q-routing for updating its routing tables.
- Second, the Q-routing can converge on the shortest paths under no traffic congestion after it learns and explores until it reaches convergence time.
- Third, the Q-routing is proposed on large scale Internet networks which are three different network connectivity; random network, random network with preferential attachment (PA), and Heuristically optimal topology (HOT). These networks are simulated for exploring possibility of the Q-routing to find optimal routes for packet transmission on different structural network connectivity. The results show that the Q-routing can decrease packet traveling time on these networks, whereas each pair of nodes on these network is connected by different ways.
- Fourth, the Q-routing is investigated on the real UK education network; JANET. Our studies show that the Q-routing is highly flexible for sustainability high number of incoming packets. In addition, it achieved its goal to reduce delay time consuming of packet, and having minimum queue length distribution against the shortest path Dijkstra.

1.5 Thesis Organization

The rest chapters are summarized as follows.

- Chapter 2 provides the background review of the adaptive routing algorithms, models of Internet network, and queueing theory which is used to simulate in this thesis. In more detail, there are many adaptive routing algorithms apply on communication networks which we are interested in the Q-routing. In addition, the Q-routing has been successful in routing schemes on communication networks more than two decades. Hence, the Q-routing is applied on this thesis for dealing routing problem on the Internet networks which has not been addressed. Moreover, the Internet network models are considered in order to built the network nearly the

realistic one which we consider three different network connectivity namely random network, random network with preferential attachment (PA), and heuristically optimal topology (HOT). In addition, basic queueing model is described in this chapter to understand a relationship among incoming packets, queue and server which are applied for analysis performance of the Internet networks by considering average amount of time for packet transmission.

- In chapter 3, we propose the Q-routing on two small grid and random network topologies under different traffic conditions for examining adaptive packet routing through the network. In addition, the small networks can help us to understand the process of the Q-routing using routing information feedback signal for changing routing tables based on traffic conditions. Moreover, the performance of the Q-routing in terms of delay time and distribution of queue length is represented in this chapter.

- In chapter 4, the network topologies are scaled up to be large scale networks which each pair of nodes is connected by different ways in order to study performance of the Q-routing. In more detail, we compare the performance of Q-routing in terms of delay time and distribution of queue length on three different network connectivity namely random network, random network with preferential attachment (PA), and heuristically optimal topology (HOT). Each network consists of 500 node, and 5000 links. In addition, we also consider the Q-routing under different traffic conditions.

- Chapter 5, we propose the Q-routing on the real UK education network; JANET under different traffic conditions. In addition, the performance of the Q-routing in terms of delay and distribution of queue length is represented in this chapter.

- Chapter 6, we propose the Pareto Q-learning on bi-objective problem; the Deep Sea Treasure World. In addition, the performance of the Pareto Q-learning in terms of minimize time consuming and maximize the value of treasure, is represented in this chapter.

- Finally, Chapter 7 summarizes this thesis and suggests some possible future research areas.

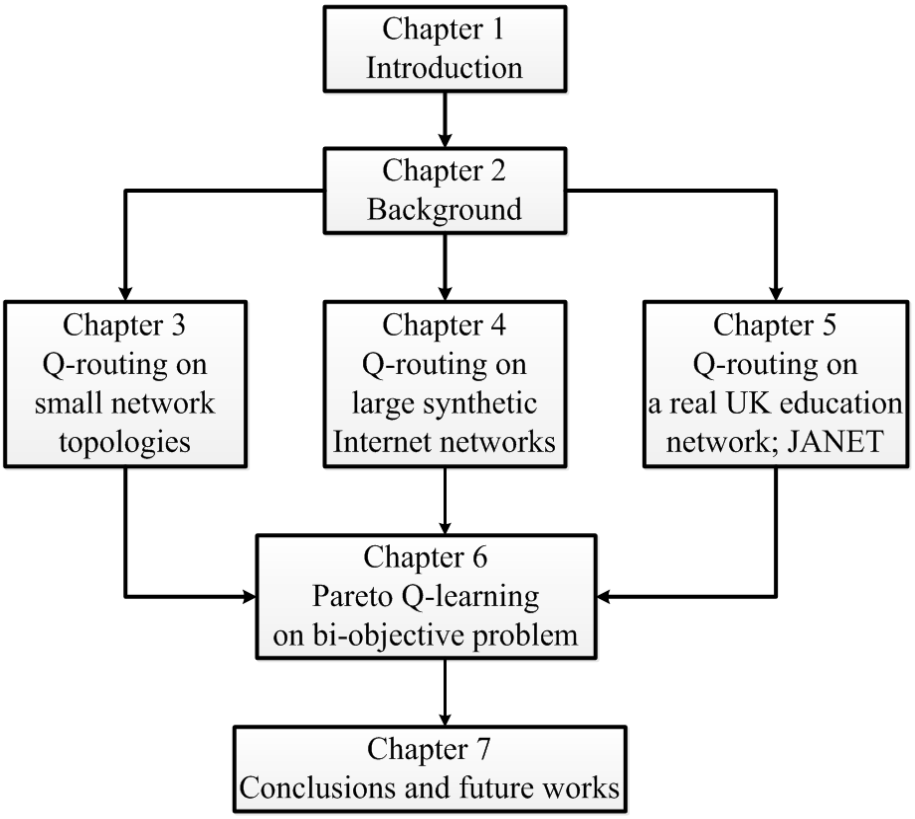


Figure 1.1: A summarizing flow chart of the thesis.

Concisely, Figure 1.1 summarizes the flow of this thesis where the Q-routing is proposed on three scenarios which are represented in chapter 3, chapter 4 and chapter 5, and then providing more information about Q-learning on multi-objective in chapter 6, and finally conclusions and future works are drawn in chapter 7.

Chapter 2

Literature Review

This chapter provides a background overview for Reinforcement Learning (RL), particularly in routing scheme as well as Q-routing. In addition, topological structure of Internet networks are also described in detail to understand how its connectivity has an effect on network performance. Finally, queueing network models are explained how packet arrivals, and departures including waiting and service times which are applied to simulate traffic routing in the Internet networks.

2.1 Introduction

Routing on communication networks is an optimize problem which still has been developed in order to achieve network performance, and it is also employed to solve vehicle and transportation problems (Geisberger et al., 2012). In this thesis, we consider only network layer on the Internet networks including forwarding and routing functions. In addition, forwarding function is different from routing function which the first one considers only a packet which is sent from an arriving link to a leaving link, and the latter one concerns with packet traveling between source and destination through entire routers on the network (Kurose and Ross, 2010). Since, the network layer involves packet forwarding, and determining optimal paths. Hence, a routing algorithm has an important in order to determine optimal paths for packet forwarding in order to achieve network performance. Considering intradomain routing on distributed networks, there are two main types of routing algorithms namely link-state and distance vector algorithms which are employed to determine optimal paths for packet forwarding (Kurose and Ross, 2010). Dijkstra's algorithm is an example of link state routing algorithm which has been employed more than a half century (Dijkstra, 1959), and its aim is finding the shortest path for packet traveling between source and destination entire the network as a result of a minimal amount of packet traveling time (Hayes, 2013). However, the Dijkstra's algorithm always sends packets via the shortest path as well as a popular path with causing

traffic congestion if there are large number of incoming packets increasing continuously, and wasting their time to be served. Hence, Boyan and Littman (1994) introduced Q-routing which packet routing can learn and adapt to changing environment such as topology or traffic conditions. In addition, the Q-routing has been employed on various type of networks as shown in Table 2.1 excluding large scale networks.

According to ability of the Q-routing, this algorithm and a general overview for RL are provided in next section. In addition, topological structure of Internet networks and the queueing network models which are applied in this thesis, are also described in the following sections.

Type of networks	**Authors**	**Type of networks topologies**	**Size of network (nodes)**
Static ad hoc networks	Bhorkar et al. (2012)	Grid	16, 36
		Random	16, 36
	Lin and van der Schaar (2011)	3-hop network	7
		4-hop network	9
Mobile ad hoc networks	Santhi et al. (2011)	Random	25
	Nurmi (2007)	Random	100
	Dowling et al. (2005)	Random	50
	Chang et al. (2004)	Centroid	26
		Random	26
Wireless sensor networks	Forster and Murphy (2007)	Random	50
	Hu and Fei (2010)	Random	125, 250
	Liang et al. (2008)	Random	40
	Dong et al. (2007)	Grid	100
		Random	100
	Zhang and Fromherz (2006)	Shooter localization	56
Cognitive radio networks	Xia et al. (2009)	Fixed	10
	Al-Rawi et al. (2014)	Fixed	10, 19
	Di Felice et al. (2010)	Random	100
Delay tolerant networks	Rolla and Curado (2013)	Random	20
		Urban-mobility	20
	Elwhishi et al. (2010)	Community based mobility model	100

Table 2.1: Summary of routing schemes on varied scale networks based on RL.

2.2 Reinforcement Learning

Reinforcement learning (RL) is a class of solution methods which solves its problems by learning, and mapping situations to actions in order to achieve its goal (Sutton and Barto, 2011). In addition, RL is one type of machine learning which differs from supervised and unsupervised learning because it can learn to make decisions after interacting with environment without predictive models or training descriptive models (Sutton and Barto, 2011). Since, RL has to use reward signal to make decision to achieve its goal. Hence, exploration and exploitation are important for finding many actions which have an effect on obtaining reward signals. For example, if RL uses only exploration to select actions, it takes a chance on getting suboptimal reward signals because it has not followed the effective actions which have been found in the past. In contrasting, if RL exploits only the already known actions, it has a few chances to get better reward signals in the future, and especially in dynamic environment. Hence, balancing between exploration and exploitation have an important role to obtain optimal rewards which have an effect on achieving goals.

2.2.1 Major Elements of Reinforcement Learning

Since, RL has the task of mapping situations to actions which involves fours major elements to represent relationship between a learner and decision maker. In addition, these elements of a RL system consist of a policy, a reward signal, a value function, and a model of the environment (Sutton and Barto, 2011), and they will be provided in a brief detail as follows:

- A policy defines a way of agent learning interacts with actions and environment at a given time until it has been officially agreed and chosen to represent an agent's behavior. For example, policy based on routing information where routers (agents) can provide multi paths under different traffic conditions in order to satisfy quality of service as a result of high network performance.
- A reward signal defined the goal in RL problem which can be a positive or a negative feedback from the environment to the agent depending on its objective. In addition, the maximum total rewards occurs when the agent hits its goal over the long run. For example, routing information under different traffic conditions is used to be a reward signal on a RL routing problem which has an affect on routers (agents) to make a routing decision in order to improve network performance in terms of minimize packet traveling time.
- A value function defines a function which represents a value of agent to predict an outcome by combining the value of current state with an estimated value of the next state.

- A model of the environment is defined based on methods for solving RL problems which is divided into two methods namely model-based and model-free methods.

In addition, a model-based method provides a model which represents how the situation changes from current state to another as known as the state transition, and the reward structure of the environment (Gläscher et al., 2010). Afterwards, actions will be evaluated by using this model. In contrast, a model-free method uses experience which has an affect on situation to build the form of a reward prediction in order to obtain actions (Gläscher et al., 2010). For example, Samejima and Doya (2007) suggested a model-based RL method in the area of neurophysiology. Strehl et al. (2006) applied a Probably Approximately Correct (PAC) model-free RL method which is called delayed Q-learning, and it shows that computation cost requirement is much less than a previous PAC based on model-based RL method.

2.2.2 Temporal-Difference Learning

Temporal-Difference (TD) learning is a method in the RL which combines ideas of Monte Carlo and dynamic programming (DP) for solving the prediction problem by learning directly from its experience without a model of the environment (Sutton and Barto, 2011). In addition, TD updates its estimated values based on the current values which are learned without using the ultimate outcome like DP (Sutton and Barto, 2011). Due to, TD is difference in time which can consider only a period of time between the previous event and the current event to predict what will happen next. Hence, some experience which had been learned in that period of time in TD learning process can provide a policy to solve the prediction problem as a brief detail will be given in the next section.

2.2.3 TD Prediction and Advantages

Since, the Monte Carlo method has to wait reward signals until the end of episode for returning information in order to update states of the system, and then provide an optimal policy (Sutton and Barto, 2011). Moreover, the DP method has to know a complete and accurate model of the environment to calculate actual values which is cause of computational expense, and it may be not suitable for large environments. Hence, the TD learning method have an advantage over the above mentioned methods which states can be updated by waiting only one time step to return information back to them. In addition, it can learn to find the optimal policy by using an estimated value function without a complete and accurate model of the environment. Due to, the TD learning methods can be classified according to the way of policy improvement which is called on-policy or off-policy. Sarsa which is an on-policy TD control method, was introduced by Rummery and Niranjan (1994) for policy improvement. Furthermore, it is called an

on-policy because it continually update the values of state-action pairs over changing the state transitions which the selected action has to return a maximum reward signal back to the state (Sutton and Barto, 2011). In this thesis, we are interested in an off-policy method which is focused on a state-value function rather than an action-value function. In addition, an well known off-policy TD control method is Q-learning (Watkins and Dayan, 1992). Furthermore, the advantage of Q-learning is only considering the maximum Q-value of the next state over all the possible actions during updated value iteration to provide optimal policy which is the minimal requirement to be guaranteed to discover an optimal behavior (Sutton and Barto, 2011). Since, the Q-learning can provide optimal policy, and discover optimal behavior without a complete and accurate model of environment which leads to many applications of Q-learning such as deep learning by LeCun et al. (2015). In this thesis, the Q-learning was applied for routing scheme on the Internet networks which the method is called the Q-routing, and it is introduced in the next section.

2.3 Q-routing

Q-routing was first introduced and successful over two decades by Boyan and Littman (1994) to solve routing problem in a small dynamically changing network. In addition, the results show that it can select a path which contains minimize total delivery time, and it is robust while the number of packets is increased continuously into the network. Moreover, the Q-routing has been employed on various network connectivity as mentioned at the table 2.1, but it has not been employed on large scale networks like Internet networks which inspires us to employ it on this network size.

Since, the aim of Q-routing is routing packets can learn routing policy through the network based on RL method for minimizing packet delivery time. Furthermore, the regular routing tables are replaced by the Q-value tables which reflect on current traffic information, and then use this information to be reward signal in order to make decision, and finally get a routing policy. In addition, the interaction between router on communication networks based on RL method can be represented as Figure 2.1.

Furthermore, the Figure 2.1 represents a straightforward framing of learning to solve the routing problem among routers on the network topology which every router on the network is an agent to learn, and make routing decision based on routing information to reduce packet delivery time to a minimum. Due to these interaction between router and its environment is ongoing that leads router selects its neighbors which respond to the network, and then provide new routing information back to the router. Moreover, determined reward signals as a task which respond to its environment, is one example of the RL problem (Sutton and Barto, 2011). In routing scheme in the RL problem, reward signal is routing information which receives from forthcoming routers.

More specifically, the interaction between the router and its network is ongoing based on a sequence of discrete time steps (t) which $t = 0, 1, 2, 3, ...$ At each time step t, the router receives forthcoming routing information from the environment's state, $S_t \in S$, where S is the set of possible states based on a selected action, $A_t \in A(S_t)$, where $A(S_t)$ is the set of actions available in state S_t. Furthermore, the state S_t is represented in terms of Q-values which is obtained from learning process, and then these values are employed to reflect on current traffic behavior through the entire network. Let consider how the routing information is employed at each router on the network at the following section.

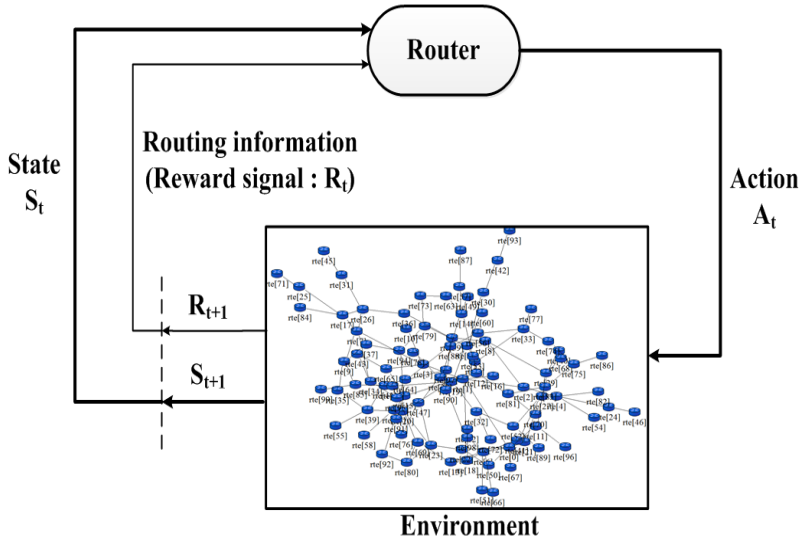


Figure 2.1: The interaction between router and network topology based on RL method

2.3.1 The Routing Information Employment

First, considering just only one router on the network, and its name is x which x has to know overview traffic behavior of the entire network via its Q-table (Q_x). Second, the router x has to select its neighbors which the routing information is employed for responding to the network by expecting to have a minimum packet delivery time. In addition, the forthcoming routing information depends on how far to spend time in packet traveling between the router x's neighbors (y) and destination (d). Moreover, router y and destination (d) are defined as $y \in N(x)$ where $N(x)$ is the set of all neighboring node x, and $d \in V$ where d is the set of all routers in the network. Hence, the table of Q-value is represented by $Q_x(d, y)$ which estimates a packet traveling time from a router y towards a destination d. Moreover, this value is sent it back to the router x as a reward signal to make a decision which its neighbor should be selected in the next round. According to Boyan and Littman (1994) and Kumar and Miikkulainen (1997), the table of $Q_x(:, y)$ is defined as three cases as follows:

- $Q_x(d, y)$ defines as an estimated packet traveling time between router y towards destination d including spending time in node x's queue, the total waiting time and transmission delays over the possible paths which is started from the router y. In addition, the estimated value of $Q_x(d, y)$ should be a minimum value in order to achieve the goal which minimize a packet traveling time to the destination.
- $Q_x(x, y)$ defines as infinity (∞) which means the packet arrives in the destination already and it should not be sent out to any node x's neighbors.
- $Q_x(y, y)$ defines as the amount of time will be sent the packet out only one hop from router x to router y because the router y is the destination, and its value is represented by δ. Thus, δ is the transmission delay over the link between router x and router y.

Moreover, if router y has to sent the packet more than a hop to the destination, it has to consider the estimated packet traveling time between its neighbors and the destination which is represented by $Q_y(d, z)$ where z is set of neighbors of router y. Due to, there are three quantities which router x has to consider before making decision, namely the waiting time q_x in the packet queue of router x, the transmission delay δ, and the amount packet traveling time $Q_y(d, z)$ (Boyan and Littman, 1994). Hence, the general equation of Q-value of router x can be shown in Equation 2.1.

$$Q_x(d, y) \leq q_x + \delta + Q_y(d, z) \qquad \forall y \in N(x) \; and \; \forall z \in N(y). \tag{2.1}$$

In addition, the equation 2.1 can be reduced a value if router y is the destination as shown in Equation 2.2.

$$Q_x(d, y) \leq q_x + \delta \qquad \forall y \in N(x). \tag{2.2}$$

However, its goal of Q-routing is to reduce a packet traveling time to a minimum. Hence, router y has to select its neighbor which contains the minimum packet traveling time between itself towards the destination, and it is represented by $Q_y(d, \hat{z})$. Therefore, the optimal packet traveling time value in case of the packet has to spend time in router x queue and full of traffic under unbounded router x's storage as shown in Equation 2.3.

$$Q_x(d, y) \leq q_x + \delta + Q_y(d, \hat{z}). \tag{2.3}$$

where

$$Q_y(d, \hat{z}) = \min_{\forall z \in N(y)} Q_y(d, z). \tag{2.4}$$

Furthermore, if the router x can serve a number of packets without traffic congestion and having router x's queue, and it has to take more than one hop in order to send the packet to the destination as a result of the optimal packet traveling time value as shown in Equation 2.5.

$$Q_x(d, y) \leq \delta + Q_y(d, \hat{z}). \tag{2.5}$$

2.3.2 The Routing Information Update Mechanism for Forwarding Packets

According to Boyan and Littman (1994), the routing information is used for estimated packet traveling time in terms of the Q-value function which should be close to the actual packet traveling time, and reflect the existing traffic condition of the network. Hence, there is a mechanism that is employed for updated the routing information as known as the Q-routing algorithm. In addition, Figure 2.2 is used to make a consideration how the router x makes a routing decision for forwarding the packet to the destination d. Moreover, the Q-routing is organized as follows:

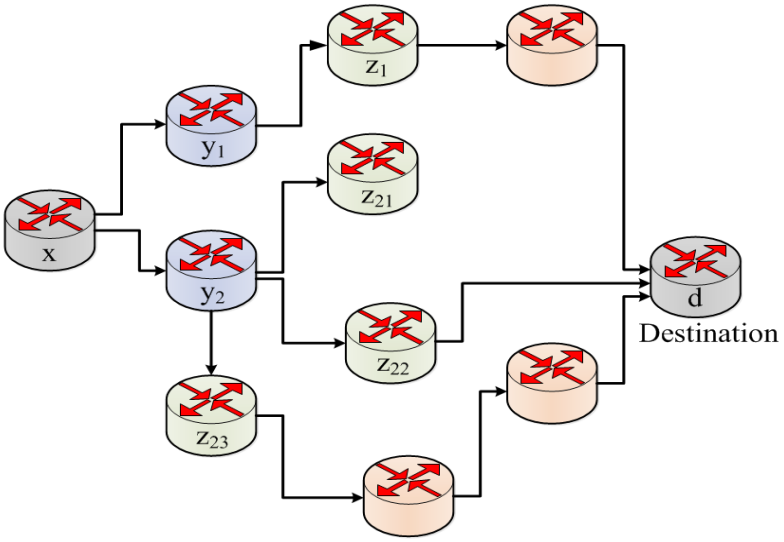


Figure 2.2: The network consists of eleven routers which router x would like to forward the number of packets to the destination d.

- First, router y_1, and router y_2 have to find its best neighbor for forwarding packets to the destination which router y_1 has only one neighbor, but router y_2 has to compare the packet traveling time between router z_{22} and router z_{23} to the destination, and then return the packet traveling time values back to the router y_2 to make a routing decision. Hence, the best estimated packet traveling time of neighbors of router y can be represented by Equation 2.6. Besides, this estimated

value is the forthcoming packet traveling time. Hence, router x has to compute the new estimated packet traveling time as shown in the next step.

$$Q_y(d, \hat{z}) = \min_{\forall z \in N(y)} Q_y(d, z). \tag{2.6}$$

- Second, the new estimated packet traveling time for the router x is computed based on three cases which have been been mentioned before, and shown in Equation 2.7 as follows:

$$Q_x(d, y)^{est} = q_x + \delta + Q_y(d, \hat{z}). \tag{2.7}$$

- Finally, the router x will update its value based on Equation 2.8 as follows:

$$Q_x(d, y)^{new} = Q_x(d, y)^{old} + \eta(Q_x(d, y)^{est} - Q_x(d, y)^{old}). \tag{2.8}$$

where η is the learning rate which is used to balance between the previous Q-value and the forthcoming estimated Q-value, and its value in the range of zero to one. If the learning rate η is set to one that means the new updated Q-value depends only on the forthcoming estimated Q-value. In contrasting, the new updated Q-value depends only on the previous Q-value if the learning rate η is zero. Moreover, the Equation 2.8 can be expanded as follows:

$$Q_x(d, y) = Q_x(d, y) + \eta(q_x + \delta + Q_y(d, \hat{z}) - Q_x(d, y)). \tag{2.9}$$

2.3.3 Summary of the Q-routing for the Routing Packet

In this thesis, we are interested in strategy for forwarding packet based on the Q-routing which every router on the network learns, and then makes a routing decision to avoid traffic congestion, and also keep a packet traveling time to a minimum under various traffic conditions. Since, the Q-routing involves forwarding packets by learning interaction among intermediate routers through the network, and shows its values in terms of the Q-value which aims to achieve its goal. However, the Q-value is not the actual packet traveling time values on the network, it is estimated as close to the actuals as possible. In addition, the Q-routing flowchart is shown in Figure 2.3 to clearly understand how the router selects its neighbor for forwarding packets by using the Q-routing algorithm. Moreover, the Figure 2.3 shows only the Q-routing embedded in the router x. However, our work employs the Q-routing which is embedded in every router on the network.

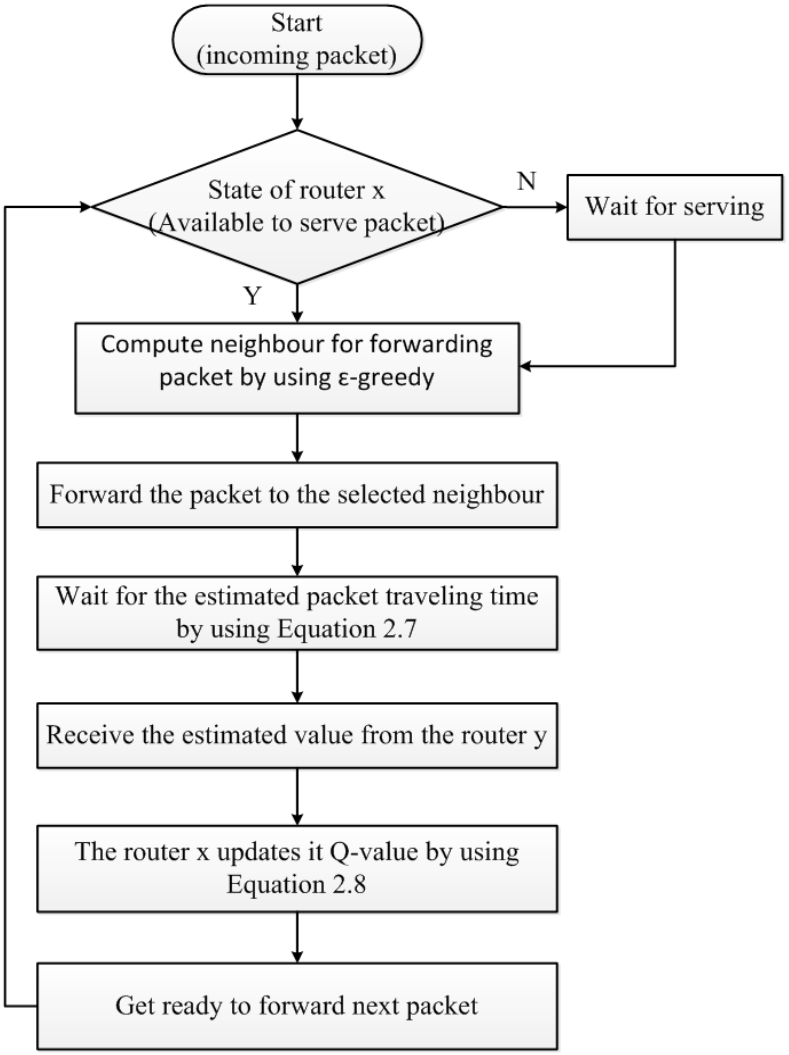


Figure 2.3: The flowchart of Q-routing algorithm which is embedded on every router through the network for forwarding packets, and this flowchart shows only the router x decides to select its neighbor router y for forwarding packets.

2.4 The Topological Structure of Internet Networks

Nowadays, spreading information has been enhanced as fast as possible to support a large number of users in the next generation of networking as fast as possible. In particular, future Internet is an extremely interesting topic which includes modeling of its topologies, security, mobility support and design methodologies in order to satisfy demand for upcoming Internet services. Due to, Internet network is an example for representative a large networking community which consists of a group of routers or switches under a single technical administration, and it also plays a fundamental role in modern societies and economies (Calvert et al., 1997; van der Ham et al., 2014). Over the past four decades, a rapidly growing number of routers leads to be more challenging to capture a complete and accurate of its topological structure nearly the realistic one in order to design and evaluate new protocols and algorithms for improved the performance

and traffic (Çetinkaya et al., 2013). Furthermore, the future Internet will not only expand from human to human, but also connect human to thing and thing to thing which leads to be heavily traffic demands as a result of complex Internet traffic problem.

Although, a much more integrated operation of networking, computing and storage devices have been developed for supporting demands on the future Internet, these components have to be managed and monitored in order to satisfied deliver services to applications and end users. In particular, topology is a basic knowledge for both the current and the upcoming future Internet platforms which provides information on the location of devices and on the connections between them. The topological structure of the Internet is a challenging issue that is investigated to find a new and more accurate structural Internet model for simulation purposes in order to design more efficient protocols, and predict how new protocols and external conditions could impact on its structure (Gregori et al., 2011). The first popular topology generator for networking simulation was the Waxman model which is connected between all pairs of nodes in the network by using probability given by a function of distance (Waxman, 1988). However, random network cannot use to explain characteristic of some real networks which exhibit certain hierarchical features as a result in suggesting non-random structures such as hierarchy and locality for generating these networks (Doar, 1996; Calvert et al., 1997; Zegura et al., 1997). In addition, network redundancy is insufficient to be achieved through the random network due to path failure and unavailability. Furthermore, power-law connectivity distribution which represents the relationship between the AS-level and router-level graph of the Internet were reported by Faloutsos et al. (1999a) in the mid of 1990s instead of the random network to provide a viewpoint of Internet's structure(Faloutsos et al., 1999a). The power-law distribution represents high number of nodes tolerance against node failures, and node attacks on the Internet AS-level topology (Cohen et al., 2001; Pastor-Satorras and Vespignani, 2001). Hence, the identification and explanation of power laws have become an increasingly significant issue which are commonly found in network topology literature (Yook et al., 2002; Chen et al., 2002; Medina et al., 2000). Although, the Internet topology generators such as the GT-ITM still have been developed, they were unsound to generate nearly the realistic Internet due to their connectivity generated based on random selection. Hence, they should be replaced the Internet topology generator by other topological models such as the Barabàsi and Albert (BA) model (Medina et al., 2001; Yook et al., 2002).

Since, the topological structure of Internet is represented by connections of routers or autonomous systems (*AS*) UCLA (2012). Hence, there are two interesting ways to study routing, resource reservation and administration on Internet topology which are router and AS levels, while the Internet topology is explosive growth. Over the last decade, the Internet topology was surprisingly discovered that random growth of incoming nodes follow power-law distributions as a result to revolutionize the current

research on the Internet topology Faloutsos et al. (1999b). In addition, the power-laws are employed to estimate important parameters such as the average neighborhood size, and facilitate the design and the performance analysis of protocols which has an advantage over simulated topologies in order to understand how to generate nearly realistic structure of the Internet Faloutsos et al. (1999b). Moreover, the power-law distribution is one of important roles to deeply understand how Internet topology is generated, so it should be studied to understand how it works.

Moreover, the power-law distribution was the first introduced by Pareto in 1896 in order to describe difference income distribution between wealthy and low income people where there are a large number of low income people contrast with wealthy people. In the mid 19^{th} century, Zipfian distribution which can be called Zipf's law was applied for trace the dynamic character of languages, and it follows a power-law distribution Dahui et al. (2005). The power-law is not only applied for understanding of social and biological phenomena, but also has been observed in communication networks. For example, the power-law of end-to-end network traffic is exploited to reconstruct the network which satisfied the technical constraints of compressive sensing Nie et al. (2013). In addition, the topology of the World Wide Web and peer-to-peer networks can be described by the power-laws.

Furthermore, power law degree distributions is one of the most important features of the networks that are generated according to one of the aforementioned probabilistic mechanisms as they tend to have a few centrally located and highly connected centers as well as hubs through which essentially most traffic has to flow. In addition, the central hubs of the networks generated by preferential attachment tend to be nodes added early in the generation process that means nodes with high expected degree have higher probability to attach to new incoming nodes. It can be clearly seen that highly connected central nodes in a network having a power law degree distribution have been a famous theme in the study of complex networks, especially among researchers inspired by statistical physics (Newman, 2003). However, this emphasis on power laws and resulting efforts to generate and explain topology only in general is not able to provide correct physical explanations for the overall network structure (Dorogovtsev et al., 2008). It is difficult to identify what mechanism of network deployment and growth is the causal drive affecting large-scale network properties and even more difficult to predict future trends in network evolution. Nevertheless, with the lack of concrete examples of such alternate models for large-scale Internet structure.

Furthermore, Li et al. (2004) introduced the heuristically optimal topology (HOT) which is designed based on combining the technological and economic issues in order to apply for the network core and the network edge planning. Due to all traffic from the network edged has to be transmitted through the network via intermediate routers which leads to have heavy traffic congestion on the core of the network. In addition, the transmission delay will be increased if the network edges far from its destination. Hence, the HOT

topology is also designed to reduce the distance between the network core and edge to be a minimum as a result to minimize packet transmission time. Furthermore, the HOT topology should represent a power-law distribution which shows relationship in the connectivity between AS-level and router-level. Due to the core of network has to contain heavily traffic congestion, so it should have low connectivity which its speed can be increased to improve network performance, and it also save cost to maintenance. Hence, Li et al. (2004) suggested that the HOT topology should be divided into three network layers: core, gateway and edge routers to facilitate maintenance.

However, there are five categories of network models and generators which can be broadly classified at present, namely random network models, preferential attachment models, optimization-based models, geographical models, and Internet-specific models (Chakrabarti and Faloutsos, 2012a). All in all, there are three network models: the random network, the random network with preferential attachment (PA), and the heuristically optimal topology (HOT) are built to represent the structural Internet networks in this thesis which the process of these networks are described in the next section.

2.4.1 The Erdös-Rényi Random Network Model

Random networks are simple network model which each node in the network can be connected its edges by using random probabilities as shown in Figure 2.4. Erdös and Rényi (1959) introduced the basic concept of random-graph theory which defined N labeled nodes connected by n edges which are selected randomly from the possible edges. Furthermore, there is an alternative way to create the random network which every pair of nodes will be connected with probability p (Gilbert, 1959). Therefore, many researchers employed the random graph theory for generating network because it is the simplest way to understand the network (Zhang et al., 2016; Yavuz et al., 2015; Costa and Farber, 2015).

In addition, Albert and Barabási (2002) suggested that random network is frequently employed in studying complex networks because it is visibly found in the network consisting of complex topology whether unknown organizing principles.

However, even though the random networks exhibit such interesting phenomena but their degree distribution is Poisson, and also have very different from the networks with power laws distributions which are claimed to be more likely real world networks such as WWW. networks. Hence, the random networks are suitable for studying the early generated networks, and then modeling of network generators should be developed in order to be close to the real network.

The basic network model which selects link to be connected between pair of nodes in the network by using random probability distribution is the Erdös-Rényi model (Erdös and Rényi, 1959; Chakrabarti and Faloutsos, 2012a). In addition, this model is the simplest

model for creating synthetic networks especially aim of study on network simulation in order to understand relationship between node and its number of connection as well as degree distribution.

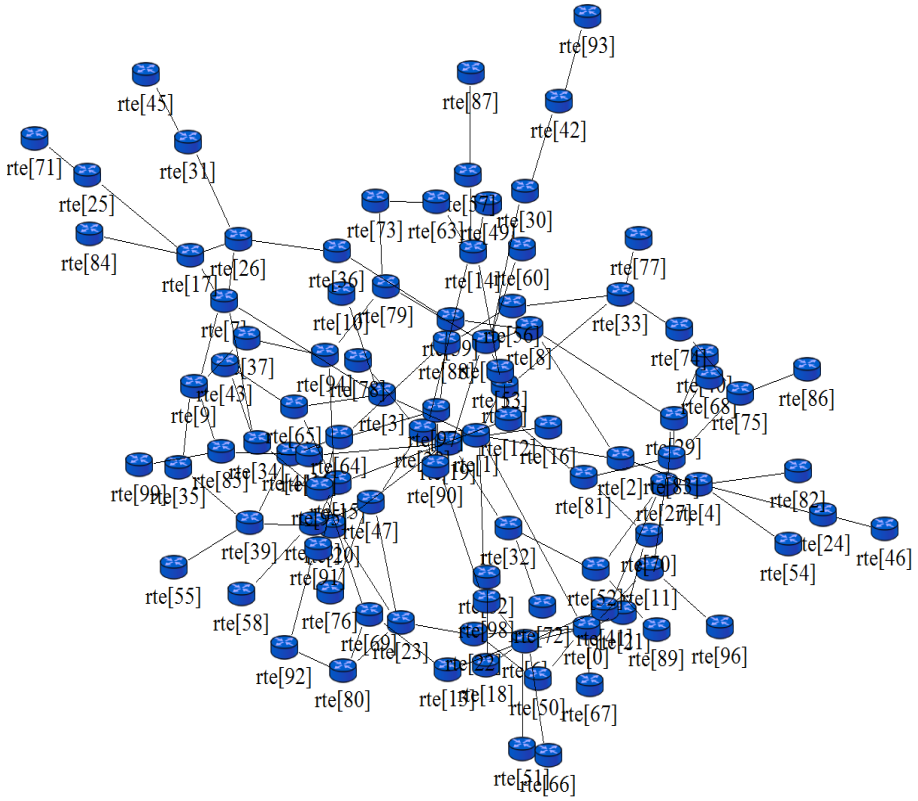


Figure 2.4: This is an example of random network which consists of 100 nodes, and they are connected each other by using random probability.

2.4.1.1 Degree Distribution

Degree distribution represents the act of sharing links among a number of nodes on the network in terms of probability as shown in Equation 2.10 follows:

$$p_k = \binom{N}{k} p^k (1-p)^{N-k} \tag{2.10}$$

where p_k is the probability of a node having a number of links k, and N is a number of nodes on the network. In addition, $p(N-1)$ can be replaced by z which leads Equation 2.10 to represent the Poisson model (Chakrabarti and Faloutsos, 2012a) as shown in Equation 2.11.

$$p_k \approx \frac{z^k e^{-z}}{k!} \tag{2.11}$$

Depending on Equation 2.11 which degree distribution on the random network is Poisson which has differing views on the real networks which should represent power-law degree distribution.

2.4.1.2 Clustering Coefficient

Clustering coefficient (CC_{random}) represents relationship between a group of node and its any two neighbors which are connected with the connection probability (Chakrabarti and Faloutsos, 2012a) as shown in Equation 2.12.

$$CC_{random} = \frac{< k >}{N} \tag{2.12}$$

where $< k >$ is the average a number of connection of the nodes.

2.4.1.3 Diameter

According to Chakrabarti and Faloutsos (2012a), the diameter of the network increases slowly in contrasting with rising the number of nodes. In addition, the Equation 2.13 uses to represent the diameter of the Erdös-Rényi random network.

$$\phi = \frac{\log N}{\log < k >} \tag{2.13}$$

Since, the degree distribution of the Erdös-Rényi random network exhibits a distinctive appearance from the degree distribution of many real-world networks. Hence, the random network with preferential attachment is introduced to build the network which its degree distribution exhibits the power-law degree distribution.

2.4.2 Random Network with Preferential Attachment

The power-law degree distribution observed in networks was addressed by Barabási and Albert (1999) which claimed that the property of this distribution is shared on many real networks such as the World Wide Web and citation networks. Moreover, Albert and Barabási (2002) provided more detail growing of these networks which the number of new nodes increased exponentially, and also connected an existing node on the network based on the reputation of the existing node in terms of the probability of node's degree. Furthermore, if the probability of connecting to a node relies on the node's degree, it is called a preferential attachment. For example, a new web page prefers to connect with popular hyper-links which should have high degrees or number of connection because highly connected these links can be found easily and lead the new

web page to be broadly well-known. Hence, there are two contributing factors namely growth and preferential attachment that will be exhibited when the network grows in a power-law degree distribution as follows:

- Growth means starting with a small number (m_0) of node, and then add a new node every time step with $m(\leq m_0)$ edges that link the new node to m different nodes already present in the system.
- Preferential attachment means the process of a new node prefers to connect with an existing node based on its reputation. Moreover, the probability P that a new node will be connected to node v relies on the degree k_i of node i is given by:

$$P_v = \frac{k(v)}{\sum_i k(i)} \tag{2.14}$$

After t time steps this procedure results in a network with $N = t + m_0$ nodes and mt edges.

2.4.2.1 The Barabàsi-Albert Model

The Barabàsi-Albert model is a minimal model which can capture the mechanisms of the power-law degree distribution. Compared to many real-world networks, it predicts a power-law distribution with a fixed exponent, while the exponents measured for real networks can vary according to its size and topology. In addition, the degree distribution of real networks can show having non power-law features such as exponential cutoffs (Amaral et al., 2000; Barabási et al., 2000). Hence, the description of the model on real networks leads to increase of interesting in addressing several basic aspects of network evolution especially classification of the network topology based on quantities beyond the degree distribution. Furthermore, the network community is still researching in order to discover new facts about how to model real networks which should show robustness which some research results are already available. These results indicate the emergence of a self-consistent theory of evolving networks that offers unusual insights into network evolution and topology.

In addition, a central issue of all models generating scale-free networks is preferential attachment as the probability of receiving new edges increases with the node's degree as shown in Figure 2.6. The Barabàsi-Albert model assumes that the probability P which a node attaches to node i is proportional to the degree k of node i according to Equation 2.14

- Degree Distribution

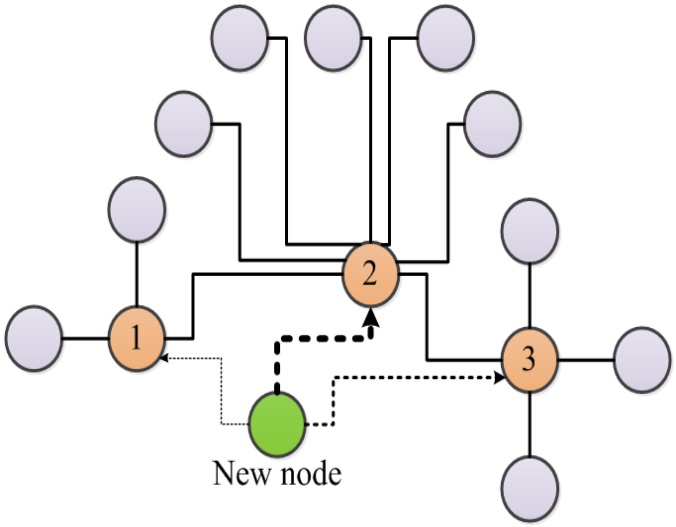


Figure 2.5: The Barabàsi-Albert model which a new node prefers to connect with node 2 more than the other nodes based on a preferential attachment because node 2 has the highest number of connection as the new node prefers connecting with node 2 shown in the thickest line.

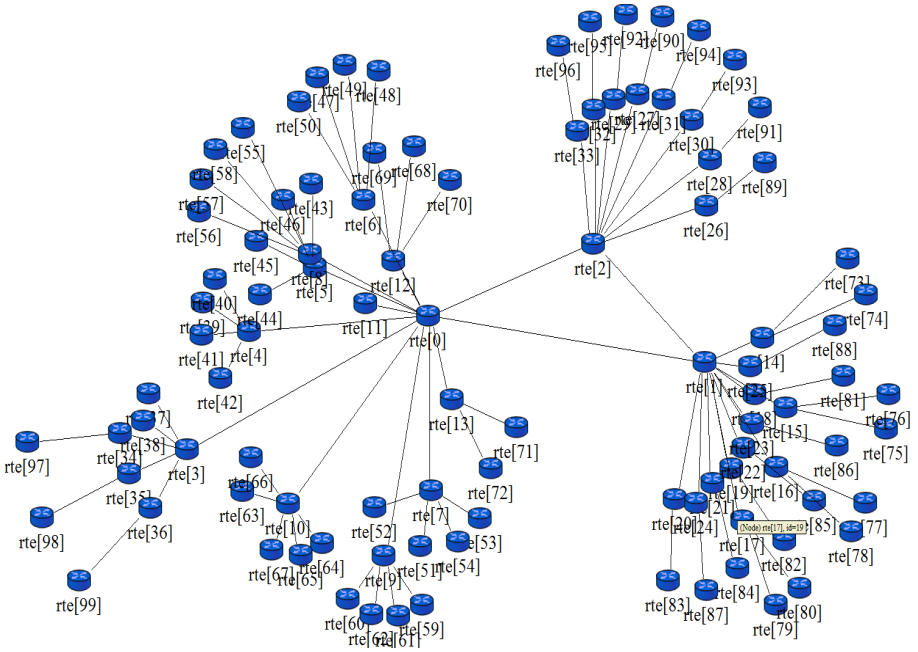


Figure 2.6: This is an example of random network with preferential attachment which consists of 100 nodes, and a new coming node prefers connecting with an existing node with high number of connections.

According to Chakrabarti and Faloutsos (2012a), the degree distribution of the Barabàsi-Albert model is given by:

$$p_k \approx k^{-3} \tag{2.15}$$

which Equation 2.15 represents a power-law tail degree distribution with exponent 3, and it does not rely on the number of existing nodes m. Moreover, many researchers claim that evolution of the social network of scientific collaborations relative with the degree distribution gets involved with the approximate exponent 3 (Newman, 2003; Boccaletti et al., 2006; Liben-Nowell and Kleinberg, 2007; Castellano et al., 2009).

- Diameter

  The diameter is a parameter which is able to see how far the distance between two nodes is Chakrabarti and Faloutsos (2012a). Moreover, there are two cases to consider the diameter of the network which depends on the number of starting node m. The first case, if $m = 1$, the diameter grows as follows:

$$\phi = O(\log N) \tag{2.16}$$

  The latter case is m has at least 2 for building the network which is given by the Equation 2.17 as follows:

$$\phi = O(\frac{\log N}{\log \log N}) \tag{2.17}$$

However, the original Barabàsi-Albert model with a power-law degree distribution alway got stuck in exponent 3 which still differs from some naturally occurring networks as a results to modify the model in order to flexible capture many real-world network especially Internet.

2.4.2.2 The modified Barabàsi-Albert Model

Since, many researchers would like to generate a network model which is nearly more realistic networks, and they found that the original Barabàsi-Albert model should be added extra parameter to let have flexible component in order to capture many real-world networks. In addition, the extra parameter represents an initial attractiveness A which leads the network growing up by gaining new edges, and Equation 2.14 changed to be Equation 2.18 as follows:

$$P_v = \frac{A + k(v)}{\sum_i (A + k(i))} \tag{2.18}$$

According to modify the original Barabàsi-Albert Model, leads to change the degree distribution from Equation 2.15 to be Equation 2.19 as follows:

$$\gamma = 2 + \frac{A}{m} \tag{2.19}$$

2.4.3 Heuristically Optimal Topology

Aim of network model generator is to generate the network nearly the real-world ones which exhibit power-law degree distribution. For example, a network model with preferential attachment exhibits a power-law behavior, and the network grows like rich get richer. However, a network which exhibits the power-law behavior should be designed based on resource optimizations. Carlson and Doyle (1999) proposed the optimized network model with existent power laws and tolerance, and its name is Highly Optimized Tolerance. The Highly Optimized Tolerance involves (n) possible events in minimizing the expected cost which each event has chance to occur $p_i(1 \leq i \leq n)$ (Chakrabarti and Faloutsos, 2012a). In addition, each event also has chance to get some loss (l_i) which can be defined as a function of the resources r_i as follows:

$$l_i = f(r_i) \tag{2.20}$$

Due to, the limitation of the total resources are $\sum_i r_i \leq R$. Hence, the minimum of the expected cost (j) of the Highly Optimized Tolerance is shown as follows:

$$J = \{\sum_i p_i l_i \mid l_i = f(r_i), \sum_i r_i \leq R\} \tag{2.21}$$

The Equation 2.21 helps to plan and run successful events with a minimum expected cost where limits the total available resources (Chakrabarti and Faloutsos, 2012a). However, the Highly Optimized Tolerance model requires globally optimal decision to manage resource allocation which contrasts with the Internet by using only local decisions. Hence, an alternative model is introduced by Fabrikant et al. (2002) which provides heuristic and local trade-offs. This model is called the Heuristically Optimized Tradeoffs model which uses to generate a network under two conflicting goals. The first goal is improving the edge of the network performance by connecting a new node of the edge network with the central of the network which prefers short distance between them. The latter goal is minimizing the transmission delays among nodes through the entire network based on the number of hops or distance (Chakrabarti and Faloutsos, 2012a). In addition, Fabrikant et al. (2002) suggested that a new node (i) should be connected with an existing node (j) by considering under two conflicting goals which can be defined as follows:

$$\alpha.d_{ij} + h_j(j \leq i) \tag{2.22}$$

According to Chakrabarti and Faloutsos (2012a) d_{ij} is the distance between node i and node j, h_j is measure of the centrality of node j, and the α is a constant control of two parameters d_{ij} and h_j.

Furthermore, Alderson et al. (2005) introduced the Heuristically Optimal Topology (HOT) model as shown in Figure 2.7 which is relative to the Highly Optimized Tolerance and the Heuristically Optimized Tradeoffs models. The HOT model is suggested that is a reasonably good design for an Internet Service Provider (ISP) network which core of the network connected with high speed and low connectivity to support the volume of traffic (Alderson et al., 2005).

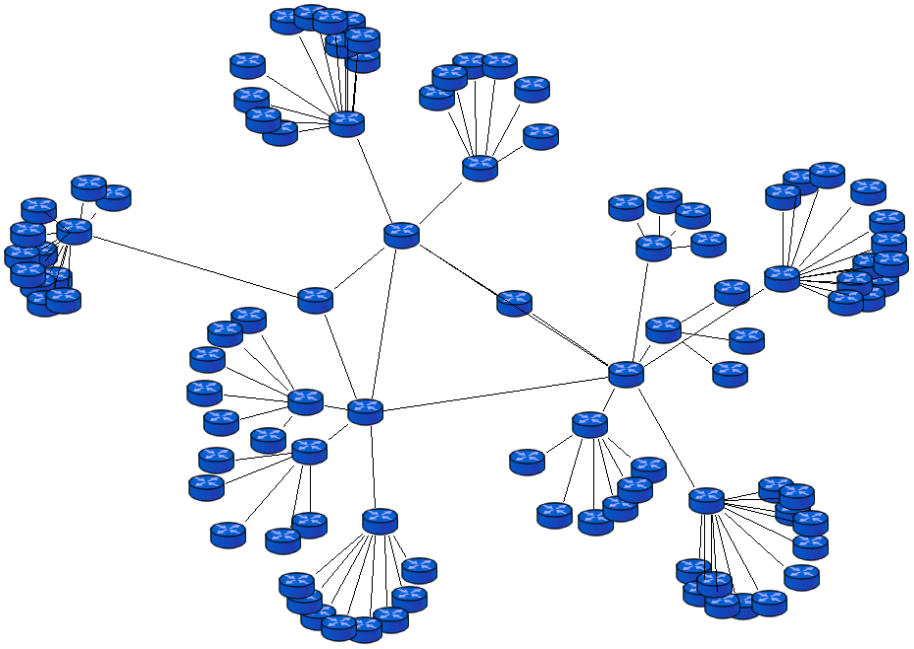


Figure 2.7: This is an example of heuristic optimal topology which consists of 100 nodes.

2.4.4 Validity the Network Model for Internet

According to Chen et al. (2002), the process of generating Internet networks has been summarized to validate as follows:

- Incremental Growth means the size of network is extended by adding nodes and edges gradually over time which lead the network grows incrementally.
- Preferential attachment means a new node prefers to connect with an existing node on the network which has high number of connection.
- Addition of Internet edges means increasing the number of internal edges of the network by connecting the new edge with a pair of existing nodes based on probability degree of vertex.

- Edge rewiring means rearrangement the number of connection of nodes on the network which aims to support the network engineering design. However, this parameter does not get involved in the Internet evolution (Chakrabarti and Faloutsos, 2012a).

Hence, the Internet network model should represent a power-law behavior and how it grows over the time especially adding new nodes with preferential attachment. In addition, it should be designed to support heavy traffic in the future, and easy maintenance to improve performance of the network.

2.5 Queueing Models

Since, packet routing on the Internet network involves arrivals, waiting, servicing, and departure on the routers through the network (Papoulis and Pillai, 2002). Hence, it should be modeled as an queueing model in order to observe behavior of packet routing especially on different types of Internet topologies.

Little's Law is a basic queueing network theorem which is explained relationship between arrivals and departures in terms of an average rate over the period of time (t) (Papoulis and Pillai, 2002). According to Papoulis and Pillai (2002), the basic Equation 2.23 shows relationship among average number of packets in the queueing network (L), average waiting time of a packet in the network (W), and average arrival rate of packets per unit time (λ) as follows:

$$L = \lambda W \tag{2.23}$$

In addition, the Equation 2.23 is simple and general because it does not need to specify number of servers, types of service time and inter-arrival time distributions, however this equation should be applied under steady state (Robertazzi, 2012).

Moreover, the number of customers in the system or the number of packets in the network as shown in Figure 2.8 can find from the difference of arrival rate ($A(t)$) and departure rate ($D(t)$) (Kleinrock, 1975) which is showed in Equation 2.24 as follows:

$$N(t) = A(t) - D(t) \tag{2.24}$$

Furthermore, λ is a parameter which depends on arrival rate of packets during a period of time (t), and it can be shown on Equation 2.25 as follows:

$$\lambda_t = \frac{A(t)}{t} \tag{2.25}$$

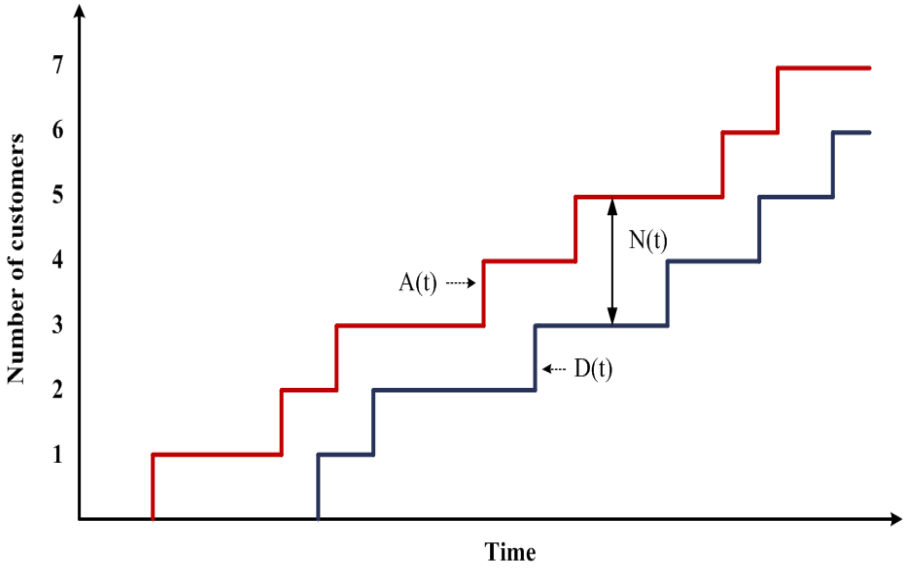


Figure 2.8: The relationship among number of customers, arrival and departure over the period of time in the network based on the Little's Law formula (Kleinrock, 1975).

According to Kleinrock (1975), the average time of a packet over all packets entire the network during the period of time (t) can be defined as a parameter (T_t) which represents the ratio of accumulated packets (γ) to the arrival rate up to the point of time as shown in Equation 2.26.

$$T_t = \frac{\gamma(t)}{A(t)} \tag{2.26}$$

Moreover, the average number of customers in the queueing system during the period of time t can be defined as $\bar{N}_t$ which is the ratio of the accumulated packets up to the time t as shown in Equation 2.27.

$$\bar{N}_t = \frac{\gamma(t)}{t} \tag{2.27}$$

Due to the relationship among Equation 2.25, Equation 2.26 and Equation 2.27, they lead the $\bar{N}_t$ can be calculated based on average packet arrival rate and the average system time which is shown in Equation 2.28.

$$\bar{N}_t = \lambda_t T_t \tag{2.28}$$

On the other hand, the Equation 2.28 is related with the basic Equation 2.23 when it is considered in case of a period of time refers to the average time spent waiting in the queue.

In addition, characteristics of queueing models have an important role to design a queueing network which the detail will be provided in the next section (Banks et al., 2005).

2.5.1 The Characteristics of Queueing Models

According to Banks et al. (2005), the characteristics of queueing models can be summarized as follows:

- Arrival Pattern: the form of packets have been arrived to the server which can be measured in terms of arrival rate or interarrival time.
- Service Pattern: the service form for serving packets on a server which may be deterministic or stochastic.
- Queueing Discipline: there are two popular ways namely First Come First Served (FCFS) and Last Come First Served (LCFS) to select incoming packets for service.
- Number of Service Channels: depending on the network model which can be parallel queues or single queue. In addition, parallel queues mean each server has a separate queue, in contrasting a single queue which has only one queue for all servers as shown in Figure 2.9.

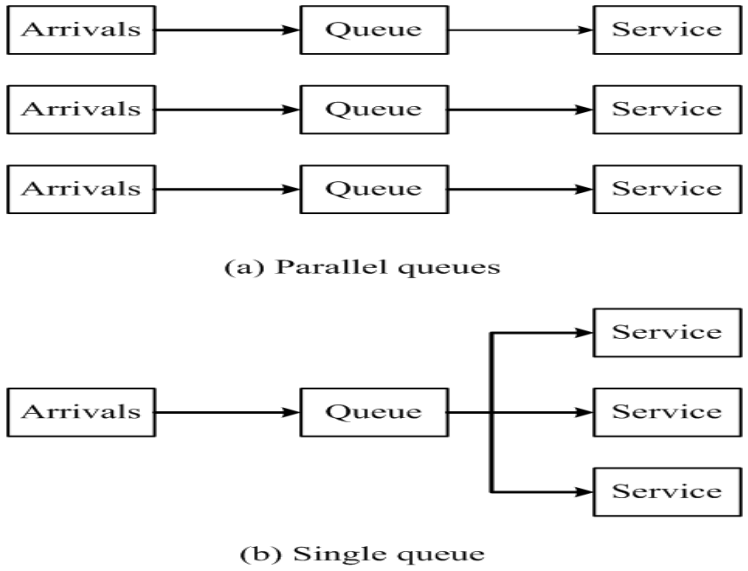


Figure 2.9: Number of service channels: (a). parallel queues, and (b). single queue.

However, it has to more specific types of arrival processes should consider the Poisson which is wildly applied for considering an arrival process in communication network. Hence, queueing on the network models in this thesis is simulated based on the M/M/1 queueing model which is a simple model, and arrival process is Poisson as described in the next section.

2.5.2 The M/M/1 Queueing Model

According to Kleinrock (1975), the M/M/1 queueing model is a classical queueing network model to analyze packet behavior where the arrival rate of these packets is Poisson process. In addition, a server is infinite storage capacity, and its service time distribution is an exponential. Moreover, the letter M stands for Markovian which the next event depends only upon the previous event, not on the sequence of all events in the past (Kleinrock, 1975).

Furthermore, the M/M/1 queueing model involves in a single server which the arrival rate (λ) should be less than the exponential service rate (μ) as a result of diminution of queue length in the network, otherwise it will be boundless growing. Considering case of interarrival time and service time are exponential distribution which lead the average interarrival time ($\bar{t}$) , and the average service time ($\bar{x}$) show in Equation 2.29 and 2.30 as follows:

$$\bar{t} = \frac{1}{\lambda} \tag{2.29}$$

$$\bar{x} = \frac{1}{\mu} \tag{2.30}$$

2.5.3 The M/M/1/K Queueing Model

More precise queueing network to be modeled nearly the real network, the server should have a limited storage capacity which the queueing network model changed from the M/M/1 to the M/M/1/K (Kleinrock, 1975). The K on M/M/1/K queueing network model stands for a total number of K packets holding in the server's storage capacity. Moreover, incoming packets are introduced continuously into the network by using Poisson process, and these packets have to find servers which can hold them in the storage capacity for waiting service (Kleinrock, 1975). However, these packets will be lost depending on no available storage capacity of a server for waiting service.

According to Kleinrock (1975), there are two cases of arrival rate as shown in Equation 2.31, and departure rate for finite storage capacity is shown in Equation 2.32.

$$\lambda_k = \begin{cases} \lambda & k < K \\ 0 & k \geq K \end{cases} \tag{2.31}$$

where K is the finite storage capacity in the server.

$$\left\{\mu_k = \mu \qquad k = 1, 2, 3, ..., K \right. \tag{2.32}$$

Furthermore, the equilibrium probability (p_k) of holding k packets in the storage capacity can be defined as Equation 2.33, 2.34 as follows:

$$p_o = \frac{1 - \frac{\lambda}{\mu}}{1 - \left(\frac{\lambda}{\mu}\right)^{K+1}} \tag{2.33}$$

$$p_k = \begin{cases} p_0 \left(\frac{\lambda}{\mu}\right)^k & k \leq K \\ 0 & k > K \end{cases} \tag{2.34}$$

2.6 Conclusion

In this chapter, we have discussed into three main parts of literature which consist of Q-routing algorithm, Internet network models, and queueing network models in order to simulate the Internet network models based on various traffic conditions, and then the Q-routing will be examined how it can improve the network performance in terms of decreasing packet delivery time on these various Internet network models.

The first part concerns with the RL and its application; Q-routing which has been introduced over two decades for solving routing on small distributed wireless networks. In addition, it is successful to improve these networks performance in terms of decreasing packet delivery time while the number of packet is steadily increasing. However, it has not been subjected to large scale networks like Internet which consists of different connectivity to build the networks. Furthermore, it would be great to apply the Q-routing on the Internet networks which these networks have grown dramatically in sizes, and leads to have traffic congestion.

Hence, the second part introduced the Internet network models based on three different construction of network connectivity. Firstly, a random network is introduced because of a simplification. However, the real network is much more complicated than this as a result of a random network with preferential attachment (PA). The PA exhibits a power-law degree distribution which can be captured from real networks. In addition, the process of building this network is also interesting because the network can grow only one side due to a new coming node prefers to connect with the famous existing nodes which show in terms of high number of connections. However, the PA network is not considerate of economic such as maintenance costs and time. Therefore, the heuristically optimal topology is introduced to consider in this thesis because it is claimed to be relative to real networks, and its designed based on engineering and economy. Moreover, the

heuristically optimal topology is also relative to real-world networks like Joint Academic Network (JANET) in the United Kingdom. Thus, three synthesis network models are studied in this thesis which the difference growth of these networks have an affect on the performance of Q-routing on various traffic conditions. Furthermore, the queueing models are also considered in this thesis in order to generate various traffic conditions as a result of traffic congestion on these networks.

The third part introduced the queueing network models which are M/M/1 and M/M/1/K. We started using M/M/1 on these synthetic Internet networks based on assumption of unlimited storage capacity because it is simplicity to examine how the Q-routing can sustain the high traffic condition. However, it should be relative to real networks which it must be a limit to the storage capacity. Hence, the M/M/1/K is employed on the network model which improves the network simulation to be more realistic.

According to the final part involves queueing network models which our networks are built based on the M/M/1 queueing model and then it will be expanded to be the M/M/1/K queueing model which related to real queueing networks. In addition, the queueing models can help us to introduce the various packet arrival rates which leads the network getting to grips with traffic congestion.

Thus, the evaluation of Q-routing on small sizes of networks based on the M/M/1 queueing model, will be represented in the next chapter which helps us to easily understand the process of routed learning and updating. In addition, it is a basic idea to understand how it works before applying it on large scale Internet networks. Then, the Q-routing is also applied for routing optimization on three synthetic Internet network models as shown in Chapter 4 which their connectivity has an effect on traffic congestion. Moreover, the basic queueing network model M/M/1 is extended to be the M/M/1/K which is introduced in Chapter 5 to employ in the realistic network model; JANET, and the Q-routing is examined how it can sustain the high traffic conditions.

Chapter 3

Adaptive dynamic packet routing on small network topologies using reinforcement learning

During the years, reinforcement learning which a branch of machine learning has been successful applied to optimize problems solving on various contexts such as routing optimization problem in communication networks. For example, Q-routing which is an application of reinforcement learning, was introduced by Boyan and Littman (1994) to find optimal paths under high traffic congestion on a 36-irregular grid network.

In this chapter, the Q-routing is evaluated the effectiveness of routing information feedback under different traffic conditions against Dijkstra's algorithm on small network topologies.

3.1 Connectivity design

The small network topologies are interested in this chapter to build networks because it is simple to examine how the Q-routing works through the network. The network sizes are built below 80 nodes based on IBM redbooks which claimed that a small network is classified to be below 80 users. Due to connectivity has an effect on network performance such as delay time, so grid network and random network are studied under different traffic conditions. In addition, the Q-routing and Dijkstra's algorithm are applied to forward packets through the network which aim to minimize delay time under high traffic congestion, and should more flexible approach to traffic conditions.

3.2 Datagram networks

Communication networks can be classified by using process of information exchange between pair of nodes as shown in Figure 3.1. In this thesis, we are interested in datagram network which provides only a connectionless service at network layer to transfer data without connected session requirement (Kurose and Ross, 2012). In addition, packet transmission on datagram network uses routing table on each router to specify which neighboring router should be selected to forward packet, and the routing table can be modified based on routing algorithm (Kurose and Ross, 2012). When packet starts transmitting between pair of nodes, and it is related to transmission time, propagation delay and processing delay. Figure 3.2 represents main delays in datagram network which are considered to build routing tables.

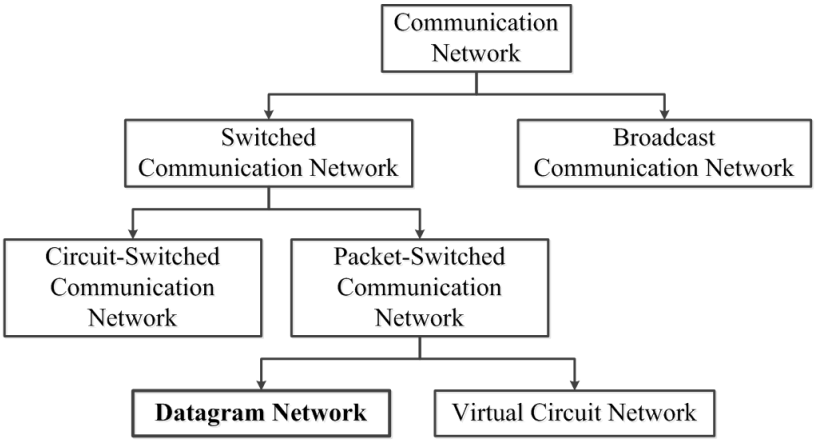


Figure 3.1: A taxonomy of communication networks

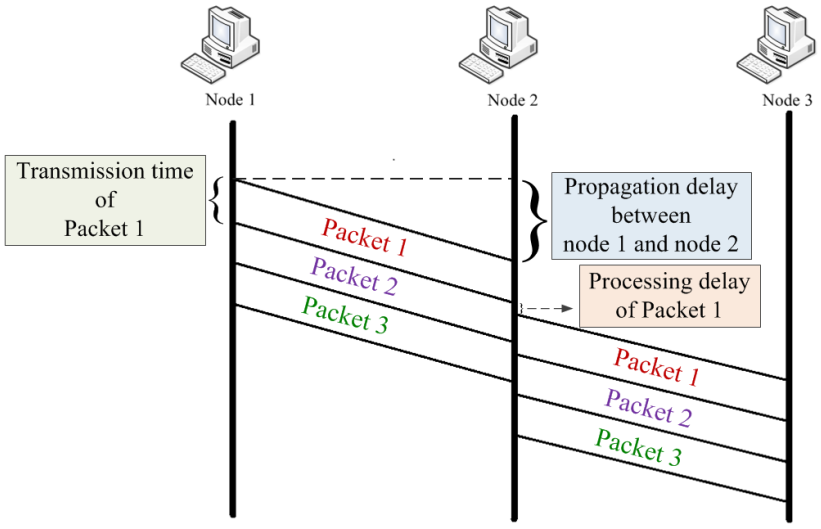


Figure 3.2: datagram network

3.3 Queueing model

Queueing model is applied for analysis performance of complex systems such communication networks by considering average amount of time which packet spends in the network (Filipowicz and Kwiecień, 2008). Basically, a queueing network model consists of incoming packets, queue and server which have a relationship between them (Filipowicz and Kwiecień, 2008).

To build a queueing network model, there are parameters which are described as follows to specify how they represent characteristics of the network:

1. Arrival process

Basically, incoming packets are arrived by using Poisson distribution with rate (λ), and the different time between each generated incoming packet is called interarrival time (τ). It is represented by a sequence of Independent and Identically distributed random variables (IID) and exponentially distribution (Jain, 2008). For example, if packets arrive at times $t_1, t_2, ..., t_j$, the random variables $\tau_j = t_j - t_{j-1}$ are called the interarrival times.

2. Service time

Normally, it is always assumed to be random variables (IID) and exponential distribution to describe how long each packet has been served which is represented by parameter (μ) (Jain, 2008).

3. Router capacity

It is available space to contain maximum number of packets which should be finite capacity as known as buffer. However, it can be infinite capacity to formulate queueing network simulation easily.

4. Queueing discipline

The queueing discipline represents how packet in queue is served which the simple one is applied without considering priorities such as Fist Come, First Served (FCFS) and Last Come, First Served (LCFS).

3.3.1 M/M/1 queueing network model

The simplest queueing model $M/M/1$ which is the abbreviation for Markov arrivals, Markov services, and single server, is applied in this chapter to simulate a queueing network model. In addition, each node in the network has single server with unlimited

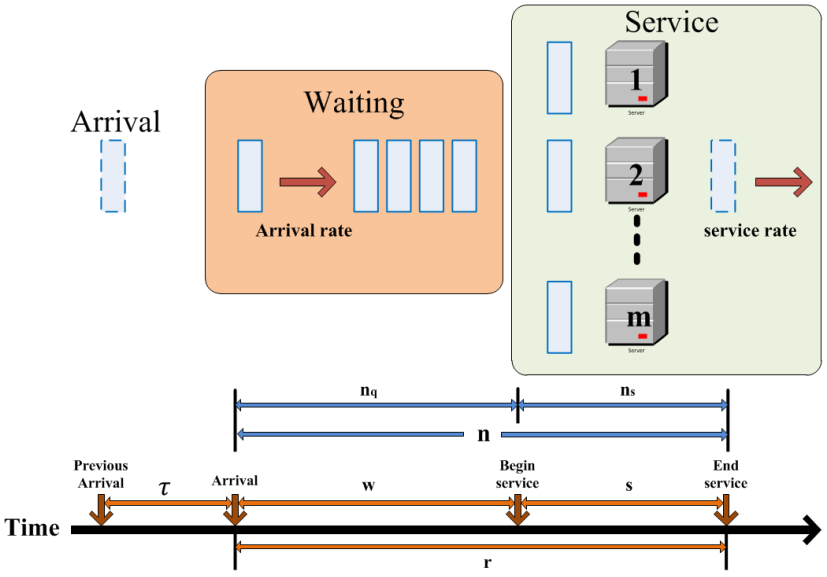


Figure 3.3: Components of a queueing network model consist of interarrival time (τ), waiting time (w), service time (s), the time in the system (r), number of packets receiving packets (n_s), number of packets waiting to serve (n_q), and number of packets in the system (n).

buffer size, and its interarrival times and service time are exponential distribution. Moreover, FCFS is employed for a service policy whereby the first incoming packet arrives in a server, it also can be the first served (Jain, 2008).

Consider a network which each node consists of a $M/M/1$ queueing network model, and a packet with constant size is generated continuously by Poisson distribution. In addition, time between each incoming generated packet is exponential distribution. Transmission capacity and propagation delay of each link specify based on Cisco 300 series for supporting small business which are 100 Mb/s and 0.5 μs, respectively. Due to propagation delay is relative with length of physical link and propagation speed in medium which is $2{\times}10^8$ m/s in fiber. Hence, the link distance between a pair of router is calculated on propagation delay multiplied by propagation speed which is $(0.5\ \mu s) \times (2 \times 10^8$ m/s) = 100 m. The specified link distance is relative to the maximum cabling distance of Cisco 300 series which is 100 meters or 328 feet. However, increasing and decreasing the interarrival times have an affect on traffic loads as well as the utilization which the maximum utilization approaches 1. For example, assuming average packet size is 1,526 bytes or 12,208 bits, and traffic load of link is 80% so 80% of 100 Mbps is 80 Mb/s. In addition, considering traffic load is 80% of link capacity which can generate packet 80 Mb/s, but the packet size is 12,208 bits so it can generate $\frac{80{,}000{,}000}{12{,}208} = 6,553$ packets/sec, and interarrival time should be $\frac{1}{6{,}553} = 0.15$ ms. Table 3.1 shows different traffic loads and its interarrival times when the packet size is 12,208 bits. Moreover, service rate (μ) depends on arrival rate (λ) and traffic load (ρ) which is $\rho = \frac{\lambda}{\mu}$ so $\mu = \frac{\lambda}{\rho}$. Hence, if

arrival rate (λ) is 6,553 packets/sec and traffic load (ρ) is 0.8, so service rate (μ) is 8,191 packets/sec.

Traffic load (%)	**Packet size (bits)**	**Arrival rate (Packets/sec)**	**Interarrival time (ms)**
10	12,208	819	1.22
20	12,208	1,638	0.61
30	12,208	2,457	0.41
40	12,208	3,276	0.31
50	12,208	4,095	0.24
60	12,208	4,914	0.20
70	12,208	5,733	0.17
80	12,208	6,553	0.15
90	12,208	7,372	0.13
95	12,208	7,781	0.12

Table 3.1: Traffic loads and interarrival times for $M/M/1$ queueing model which the packet size is 1,526 bytes (12,208 bits), and transmission capacity is 100 Mbps

3.3.2 Routing algorithms

Every router needs routing algorithm to make routing decision in order to guarantee best path for packet transmission between source and its destinations. In addition, routing algorithms use routing information from routing protocols such as number of hops or delivery time to compute the best path which can be classified as shown in Figure 3.4. In this thesis, we consider two major routing algorithms which are link-state Dijkstra's algorithm and distance vector Q-routing. The aim of Dijkstra's algorithm algorithm is finding minimal path cost path between source and it destinations as known as shortest path algorithm. However, it has to know entire routing information of the network in order to compute shortest paths which is not flexible when traffic of the network has been changed. The strategic aim of Q-routing is to find optimal paths for packet transmission between source and it destinations based on estimated function of routing information which it can adapt to new environment such as traffic conditions and network topologies.

3.3.3 Experimental Settings

These experiments are intended to demonstrate the ability of the Q-routing algorithm for packet transmission in term of average packet delay time and distribution of queue lengths, and how they are tolerant of congestion under different traffic conditions on small network topologies.

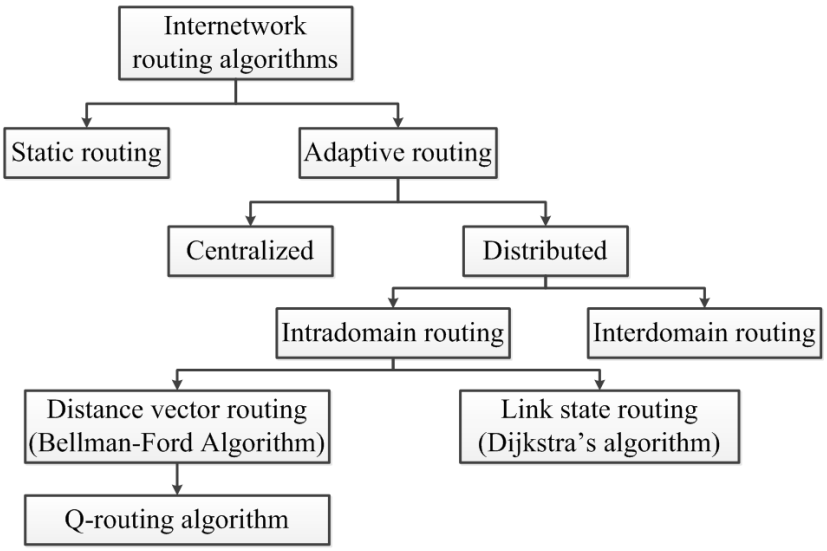


Figure 3.4: Diagram of internetwork routing algorithms

In this chapter, a size of packet is specified based on standard IEEE 802.3 which is 1526 bytes. However, size of packet frames has an effect on transmission delay in Ethernet link. Hence, different sizes of packet and interarrival time are considered which both of them are cause of delay time. Since, number of packets which is introduced to the network, has an effect on traffic congestion. Hence, the traffic congestion in this simulation happened while increasing packet arrival rate and its frame size.

Furthermore, each node generates packets are periodic which are sent to all over nodes in the network. Each packet specifies its destination, and it is sent out following routing tables. Moreover, the simplest queueing model M/M/1 is embedded in each node to store multiple packets with unbounded FCFS queue. Delay time and distribution of queue length under different traffic conditions are observed to describe how long the packet has to spend time in the queue until it can be transmitted over the link in the network.

There are two sizes of irregular grid network topologies which are employed to study Q-routing. These networks consist of 36 nodes and 72 nodes. In addition, a 36-irregular grid network is designed same as Boyan and Littman (1994) as shown in Figure 3.5, and a 72-irregular grid network is designed as shown in Figure 3.6.

In this thesis, we consider different traffic loads which each link has limited 100 Mb/s transmission capacity, and the packet sizes vary from 1,526 bytes to 4,578 bytes which its interarrival time is a leading cause of traffic congestion on the network. Table 3.1 is represented arrival rate and interarrival time when packet size is 1,526 which is a minimum packet size considering how it has effect on traffic congestion when traffic loads vary from 10% to 95%.

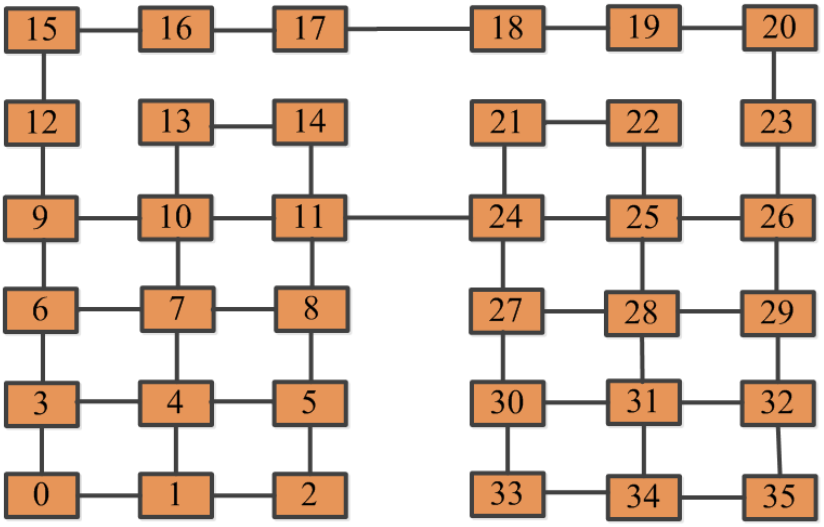


Figure 3.5: a 36-irregular grid network which every node generates packets, and sends them throughout the entire of network. In addition, left cluster (node 0 - node17) can send packets to right cluster (node18 - node35) via node 11 and node 17 which node 11 prefers to use for packet transmission because it takes small number of hops to send packets between left and right clusters (Boyan and Littman, 1994)

In addition, Table 3.2 - Table 3.5 are represented arrival rate and interarrival time when packet sizes vary from 2,289 bytes to 4,578 bytes which traffic load is increased 10% to 95% in order to study the effect of traffic congestion on the network.

Traffic load (%)	**Packet size (bits)**	**Arrival rate (Packets/sec)**	**Interarrival time (ms)**
10	18,312	546	1.83
20	18,312	1,092	0.91
30	18,312	1,638	0.61
40	18,312	2,184	0.45
50	18,312	2,730	0.36
60	18,312	3,276	0.30
70	18,312	3,822	0.26
80	18,312	4,368	0.23
90	18,312	4,914	0.20
95	18,312	5,187	0.19

Table 3.2: Traffic loads and interarrival times for $M/M/1$ queueing model which the packet size is 2,289 bytes (18,312 bits), and transmission capacity is 100 Mbps

Moreover, Table 5.1, and Figure 3.7 are summarized interarrival time under different traffic loads 10%, 50% and 90% which represent low, medium, and high traffic load levels on the network. In addition, packet sizes vary from 1,526 bytes to 4,578 bytes are also considered how it has effect on traffic congestion.

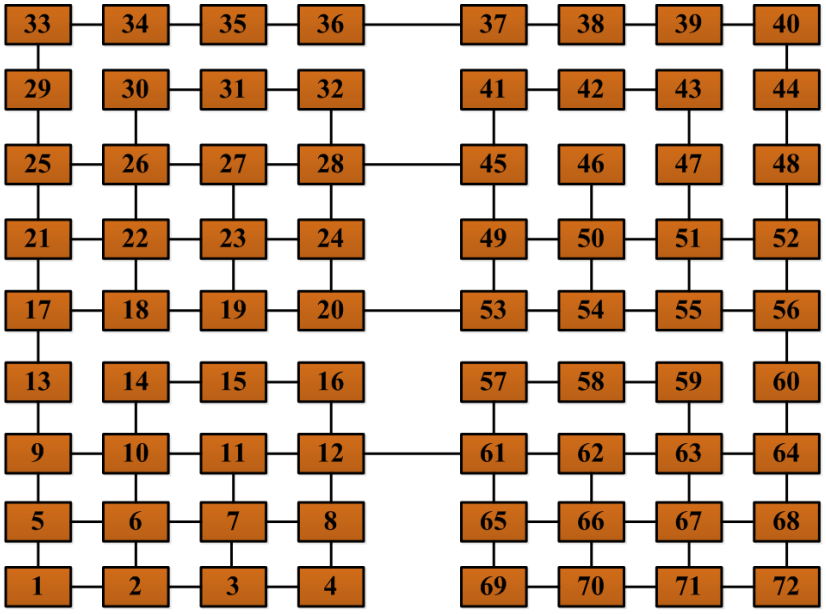


Figure 3.6: a 72-irregular grid network is designed relative to a 36-irregular grid network which every node generates packets, and sends them throughout the network. However it is extended from an originality of Boyan et. al.'s network. Hence, connected paths between left cluster (node 1 - node 36) and right cluster (node 37 - node 72) are increased to be 4 which are reasonable to support packet transmission. The packets can be transmitted via node 12, node 20, node 28, and node 36 depending on routing policy.

Traffic load (%)	**Packet size (bits)**	**Arrival rate (Packets/sec)**	**Interarrival time (ms)**
10	24,416	409	2.44
20	24,416	819	1.22
30	24,416	1,228	0.81
40	24,416	1,638	0.61
50	24,416	2,047	0.48
60	24,416	2,457	0.41
70	24,416	2,866	0.35
80	24,416	3,276	0.31
90	24,416	3,686	0.27
95	24,416	3,890	0.25

Table 3.3: Traffic loads and interarrival times for $M/M/1$ queueing model which the packet size is 3,052 bytes (24,416 bits), and transmission capacity is 100 Mbps

3.3.4 Experimental Results

Considering a 36-irregular grid network and a 72-irregular grid network as a small network which each link has 100 Mb/s transmission capacity limit, and routing algorithm is embedded in each node on these networks to find optimal routing policies for forwarding

Traffic load (%)	**Packet size (bits)**	**Arrival rate (Packets/sec)**	**Interarrival time (ms)**
10	30,520	327	3.05
20	30,520	655	1.52
30	30,520	982	1.02
40	30,520	1,310	0.76
50	30,520	1,638	0.61
60	30,520	1,965	0.51
70	30,520	2,293	0.43
80	30,520	2,621	0.38
90	30,520	2,948	0.34
95	30,520	3,112	0.32

Table 3.4: Traffic loads and interarrival times for $M/M/1$ queueing model which the packet size is 3,815 bytes (30,520 bits), and transmission capacity is 100 Mbps

Traffic load (%)	**Packet size (bits)**	**Arrival rate (Packets/sec)**	**Interarrival time (ms)**
10	36,624	273	3.66
20	36,624	546	1.83
30	36,624	819	1.22
40	36,624	1,092	0.91
50	36,624	1,365	0.73
60	36,624	1,638	0.61
70	36,624	1,911	0.52
80	36,624	2,184	0.45
90	36,624	2,457	0.41
95	36,624	2,593	0.38

Table 3.5: Traffic loads and interarrival times for $M/M/1$ queueing model which the packet size is 4,578 bytes (36,624 bits), and transmission capacity is 100 Mbps

packets. In addition, packet arrival rate is increased in order to introduce traffic congestion. Furthermore, Q-routing and shortest path are compared under various traffic conditions. Average delay time and distribution of queue length are studied as a performance of routing algorithm while the traffic is steadily increasing until congestion on the network. Since, there are two clusters would like to transmit packets throughout the network as shown in the Figure 3.5 and 3.6. However, it will get traffic congestion easily if packet is transmitted only via the popular path which is the shortest way to connect between two clusters. Hence, the Q-routing is employed to avoid traffic congestion by using routing information to be feedback signal, and it can reflect on current traffic in order to make routing decision for forwarding packet.

Figure 3.8 shows the effective of Q-routing on a 36-irregular grid network under traffic

Traffic load (%)	Packet size (bytes)	Interarrival time (ms)
10	1,526	1.22
	2,289	1.83
	3,052	2.44
	3,815	3.05
	4,578	3.66
50	1,526	0.24
	2,289	0.36
	3,052	0.48
	3,815	0.61
	4,578	0.73
90	1,526	0.13
	2,289	0.20
	3,052	0.27
	3,815	0.34
	4,578	0.41

Table 3.6: Summary of interarrival time under different traffic loads 10%, 50%, and 90% which each link has limited 100 Mbps transmission capacity, and the packet sizes vary from 1,526 bytes to 4,578 bytes

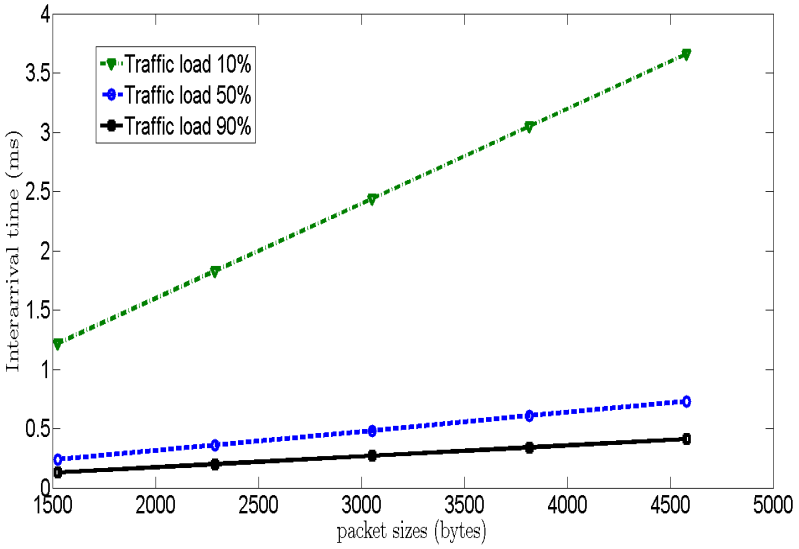


Figure 3.7: Summary of interarrival time when the packet sizes vary from 1,526 bytes to 4,578 bytes under different traffic loads 10%, 50%, and 90% which each link has limited 100 Mbps transmission capacity

load 10% as a low load level where the packet sizes vary from 1,526 bytes to 4,578 bytes. The experimental result shows that it is a slightly different between the Q-routing and the shortest path under low load level because the Q-routing can learn to find an optimal path which is the shortest path to send packets due to it is no traffic congestion. However, the Q-routing can send packets by reducing average delay time when compared with the shortest path where the sizes of packet have an affect on waiting time for serving which

is cause of delay time.

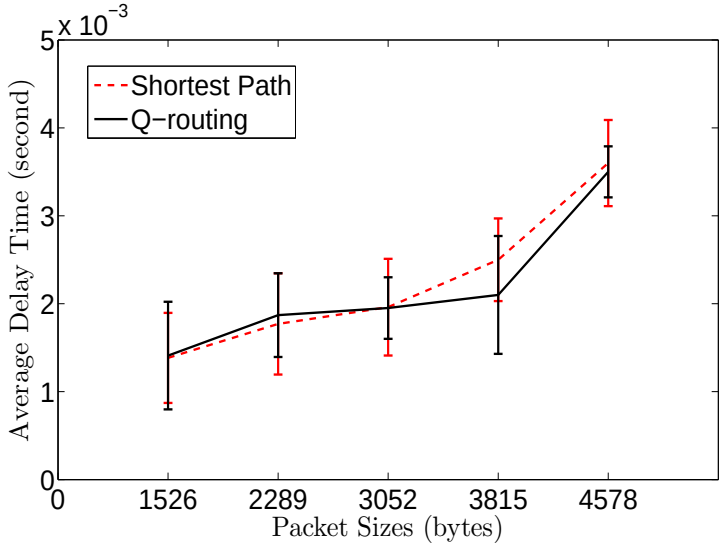


Figure 3.8: Comparing of average delay time between Shortest Path and Q-routing on a 36-grid network which the packet sizes vary from 1,526 bytes to 4,578 bytes under traffic load 10%, and each link has limited 100 Mbps transmission capacity.

Figure 3.9 shows the effective of Q-routing on a 36-irregular grid network under traffic load 50% as a medium load level where the packet sizes vary from 1,526 bytes to 4,578 bytes. The experimental result shows that the average delay time is reduced by using the Q-routing when compared with the shortest path under medium load level because the Q-routing can learn to find optimal paths by using the routing information feedback which reflects on current traffic condition in order to select it neighbor for forwarding packets without traffic congestion.

Figure 3.10 shows the effective of Q-routing on a 36-irregular grid network under traffic load 90% as a high load level where the packet sizes vary from 1,526 bytes to 4,578 bytes. The experimental result shows that the average delay time is significantly reduced by using the Q-routing when compared with the shortest path under high traffic congestion because the Q-routing can learn to find optimal paths by using the routing information feedback which reflects on current traffic condition in order to select it neighbor for forwarding packets without traffic congestion.

Figure 3.11 shows the effective of Q-routing on a 72-irregular grid network under traffic load 10% as a low load level where the packet sizes vary from 1,526 bytes to 4,578 bytes. The experimental result shows that it is a slightly different between the Q-routing and the shortest path under low load level because the Q-routing can learn to find an optimal path which is the shortest path to send packets due to it is no traffic congestion. However, the Q-routing can send packets by reducing average delay time when compared with the shortest path where the sizes of packet have an affect on waiting time for serving which is cause of delay time.

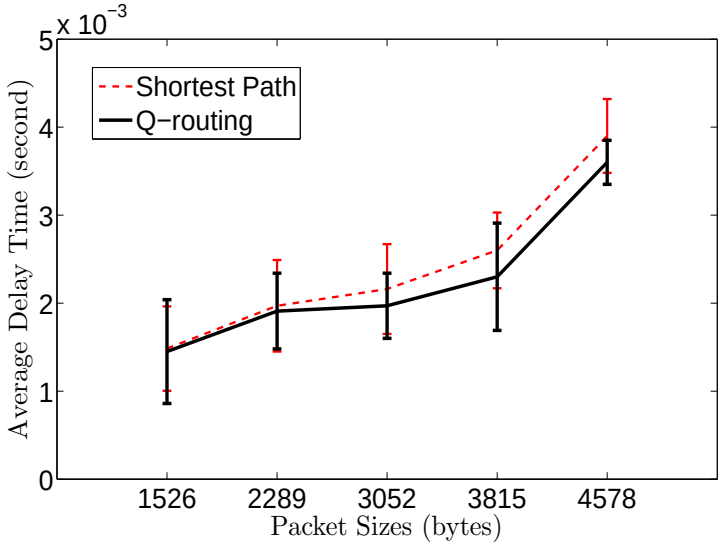


Figure 3.9: Comparing of average delay time between Shortest Path and Q-routing on a 36-grid network which the packet sizes vary from 1,526 bytes to 4,578 bytes under traffic load 50%, and each link has limited 100 Mbps transmission capacity.

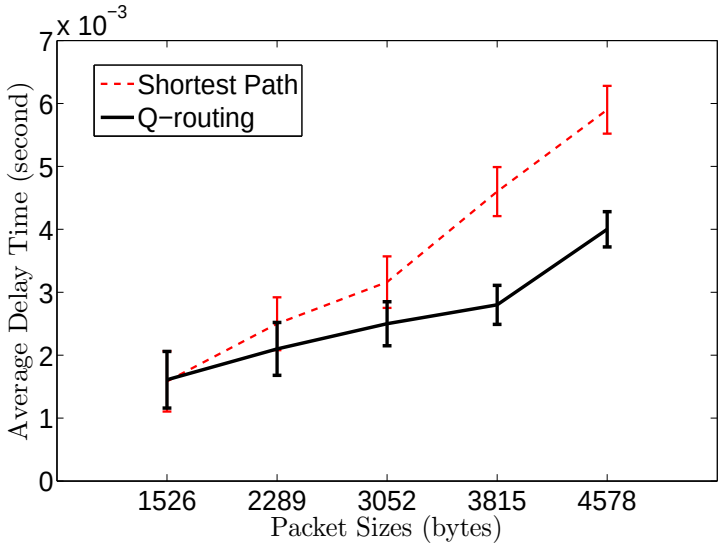


Figure 3.10: Comparing of average delay time between Shortest Path and Q-routing on a 36-grid network which the packet sizes vary from 1,526 bytes to 4,578 bytes under traffic load 90%, and each link has limited 100 Mbps transmission capacity.

Figure 3.12 shows the effective of Q-routing on a 72-irregular grid network under traffic load 50% as a medium load level where the packet sizes vary from 1,526 bytes to 4,578 bytes. The experimental result shows that the average delay time is reduced by using the Q-routing when compared with the shortest path under medium load level because the Q-routing can learn to find optimal paths by using the routing information feedback which reflects on current traffic condition in order to select it neighbor for forwarding packets without traffic congestion.

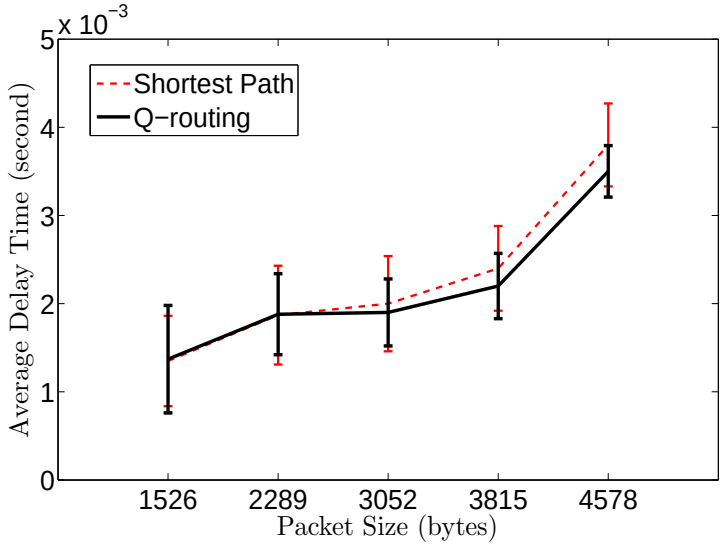


Figure 3.11: Comparing of average delay time between Shortest Path and Q-routing on a 72-grid network which the packet sizes vary from 1,526 bytes to 4,578 bytes under traffic load 10%, and each link has limited 100 Mbps transmission capacity.

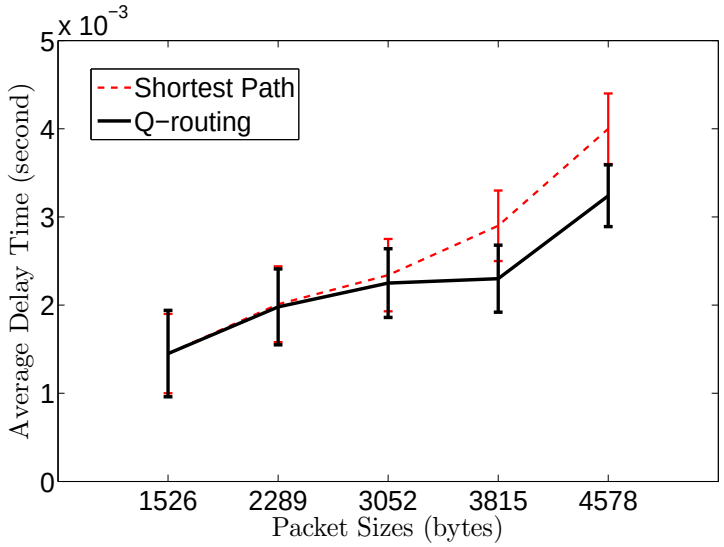


Figure 3.12: Comparing of average delay time between Shortest Path and Q-routing on a 72-grid network which the packet sizes vary from 1,526 bytes to 4,578 bytes under traffic load 50%, and each link has limited 100 Mbps transmission capacity.

Figure 3.13 shows the effective of Q-routing on a 72-irregular grid network under traffic load 90% as a high load level where the packet sizes vary from 1,526 bytes to 4,578 bytes. The experimental result shows that the average delay time is significantly reduced by using the Q-routing when compared with the shortest path under high traffic congestion because the Q-routing can learn to find optimal paths by using the routing information feedback which reflects on current traffic condition in order to select it neighbor for forwarding packets without traffic congestion.

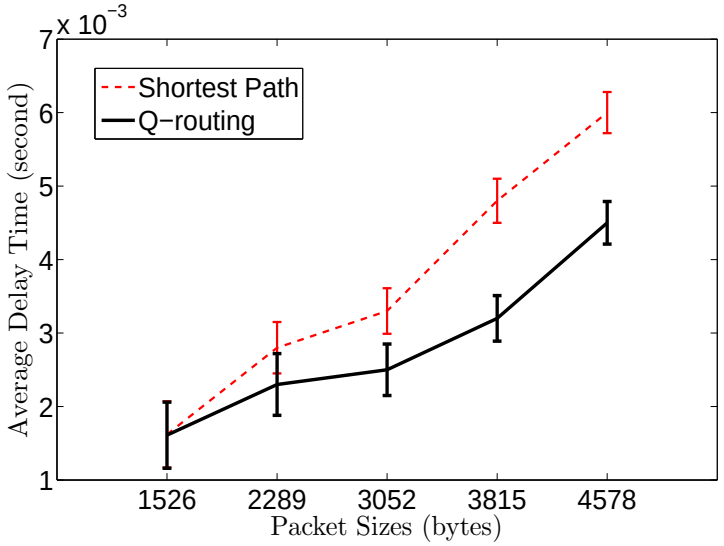


Figure 3.13: Comparing of average delay time between Shortest Path and Q-routing on a 72-grid network which the packet sizes vary from 1,526 bytes to 4,578 bytes under traffic load 90%, and each link has limited 100 Mbps transmission capacity.

Figure 3.14 shows the pdf of queue length on a 36-irregular grid network under traffic load 10% which the queue length between the Q-routing and the shortest path is slightly different because of no traffic congestion. However, the Q-routing holds smaller queue length than the shortest path under high traffic load 90% because the Q-routing discovers multi-path for forwarding packets which these paths are also considered to avoid traffic congestion as shown in Figure 3.15.

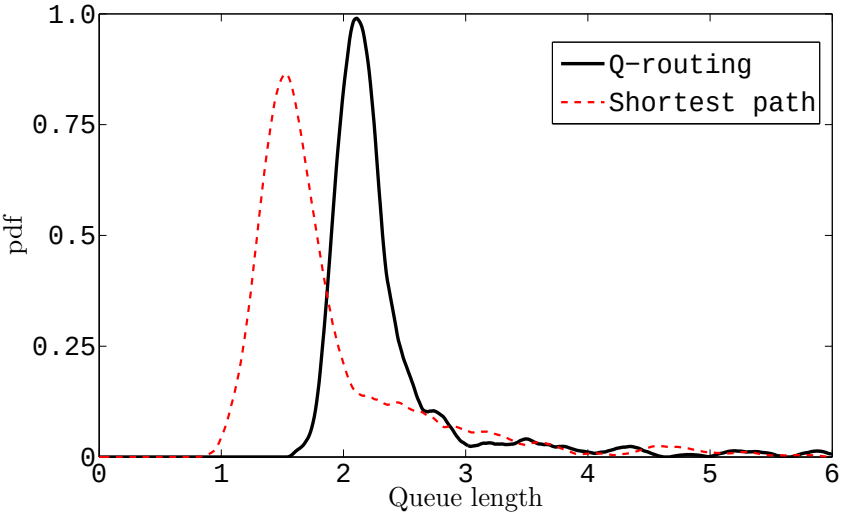


Figure 3.14: Comparing of queue length between Shortest Path and Q-routing on a 36-grid network which the packet sizes is fixed at 1,526 bytes under traffic loads 10%, and each link has limited 100 Mbps transmission capacity.

Figure 3.16 shows the pdf of queue length on a 72-irregular grid network under traffic

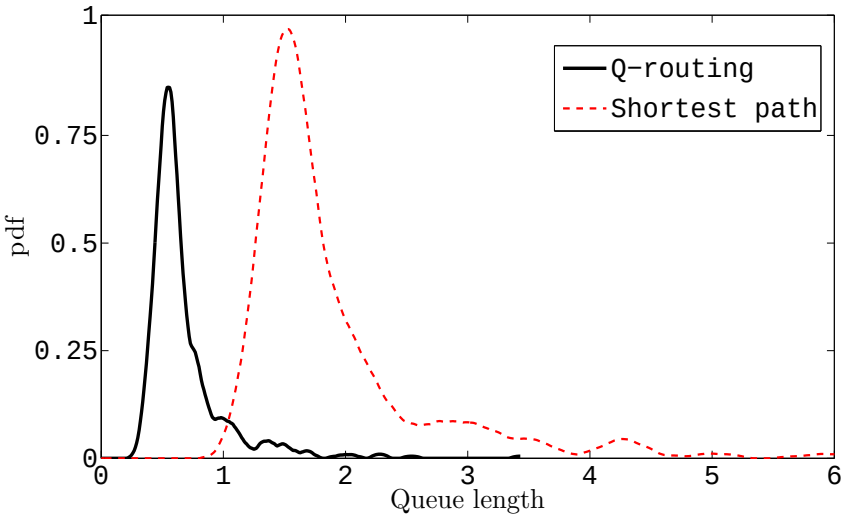


Figure 3.15: Comparing of queue length between Shortest Path and Q-routing on a 36-grid network which the packet sizes is fixed at 4,578 bytes under traffic loads 90%, and each link has limited 100 Mbps transmission capacity.

load 10% which the queue length between the Q-routing and the shortest path is slightly different because of no traffic congestion. However, the Q-routing holds smaller queue length than the shortest path under high traffic load 90% because the Q-routing discovers multi-path for forwarding packets which these paths are also considered to avoid traffic congestion as shown in Figure 3.17.

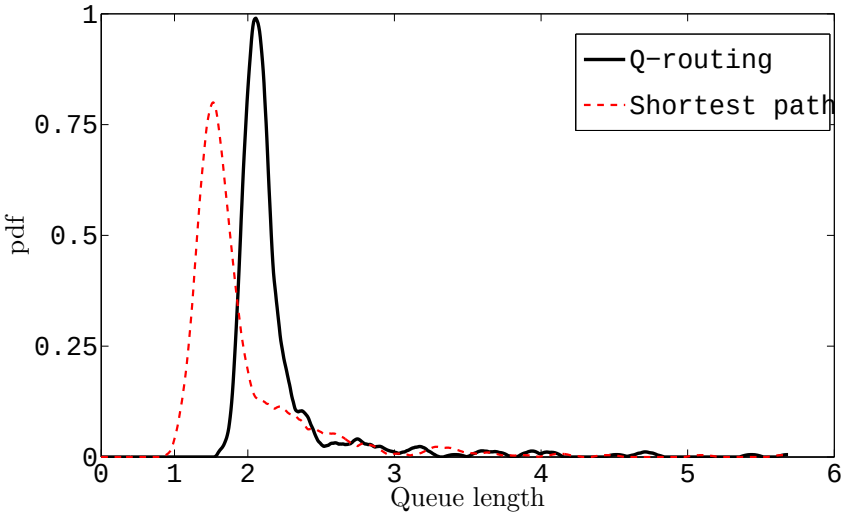


Figure 3.16: Comparing of queue length between Shortest Path and Q-routing on a 72-grid network which the packet sizes is fixed at 1,526 bytes under traffic loads 10%, and each link has limited 100 Mbps transmission capacity.

In addition, the number of hops is compared between Shortest Path and Q-routing on a 36-grid network which the packet sizes is fixed at 1,526 bytes under traffic load 10%, and each link has limit100 Mbps transmission capacity limit. The result in Figure 3.18

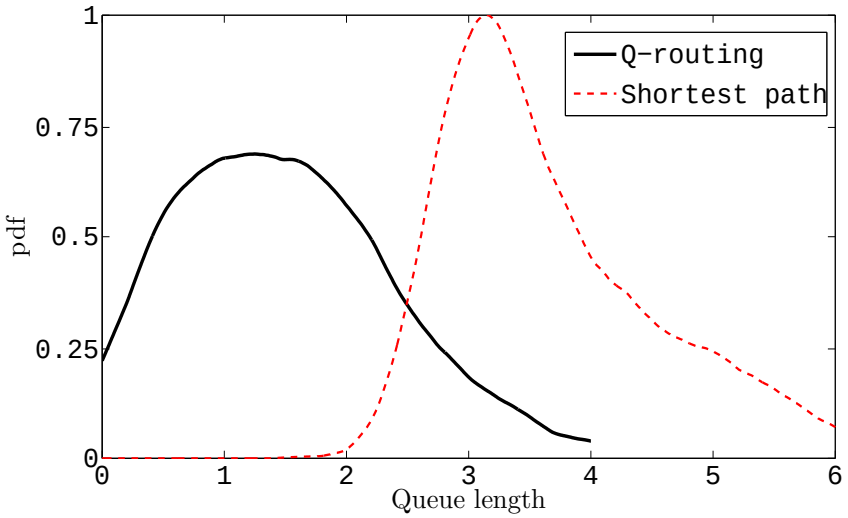


Figure 3.17: Comparing of queue length between Shortest Path and Q-routing on a 72-grid network which the packet sizes is fixed at 4,578 bytes under traffic loads 90%, and each link has limited 100 Mbps transmission capacity.

shows that it is slightly different the number of hops because the Q-routing can converge to the shortest path which is an optimal policy for forwarding packets. However, the Q-routing can take higher number of hops for forwarding packets under high traffic load 90% to avoid traffic congestion as show in Figure 3.19.

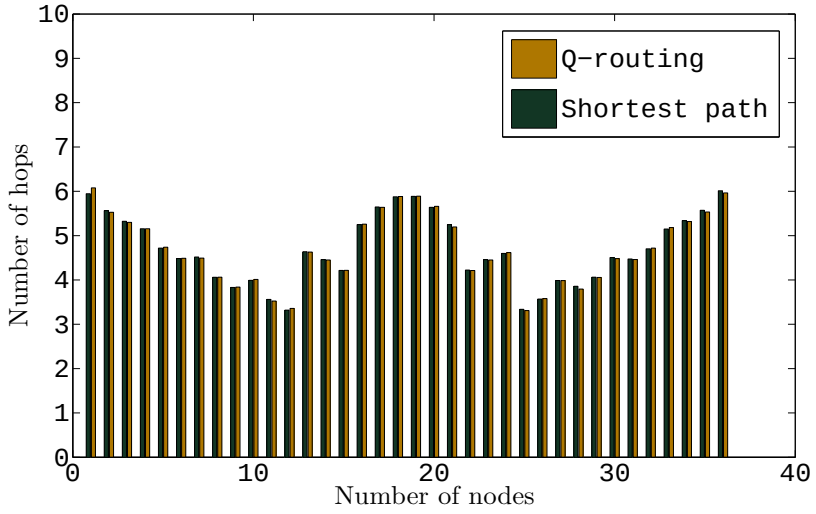


Figure 3.18: Comparing number of hops between Shortest Path and Q-routing on a 36-grid network which the packet sizes is fixed at 1,526 bytes under traffic loads 10%, and each link has limited 100 Mbps transmission capacity.

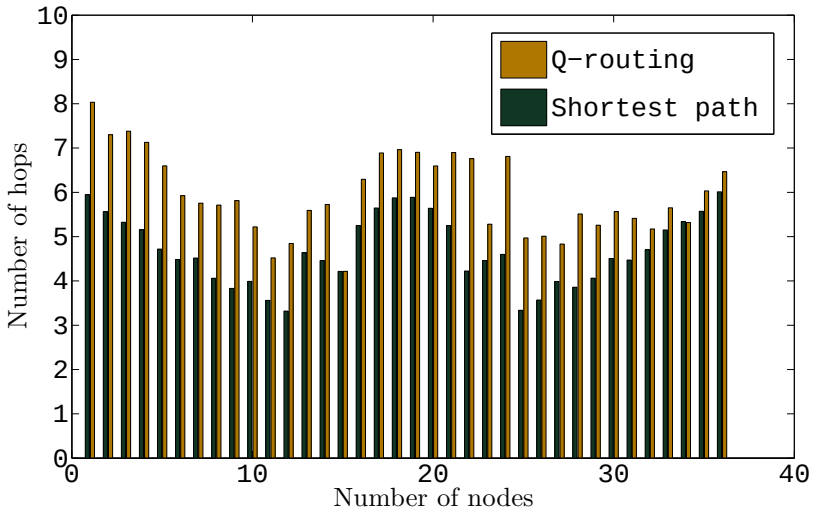


Figure 3.19: Comparing number of hops between Shortest Path and Q-routing on a 36-grid network which the packet sizes is fixed at 1,526 bytes under traffic loads 90%, and each link has limited 100 Mbps transmission capacity.

3.4 Conclusions

In this chapter, the Q-routing on the irregular grid topologies is employed to study the performance of Q-routing when it is employed for finding routing policies, and then compared with the shortest path. The performance of Q-routing is measured in terms of average delay time under different traffic conditions where the number of packets are generated, and then forwarded to its destination. In addition, the number of packets can be increased continuously under various sizes of packet routing. The experimental results show that the Q-routing can find optimal routes for forwarding number of packets as a results of a minimum of average delay time. Moreover, it can be clearly seen that the queue length when Q-routing is employed for finding routing policy is smaller than the shortest path because the Q-routing can find optimal routes which avoids traffic congestion. However, this chapter shows the performance of Q-routing on small networks which guarantees that it is successful in reducing packet delay time as a result of improving network performance. However, it is more interesting and challenging to employ the Q-routing on large scale sizes of networks like Internet. Hence, the Q-routing on the large scale Internet networks is studied in the next chapter.

Chapter 4

Adaptive dynamic packet routing on large scale Internet networks

According to an adaptive packet routing research on Boyan et al.'s work over the past two decades (Boyan and Littman, 1994), there are many researchers get inspiration for solving routing problems by applying Q-routing on various networks. However, following of these work neither addressed larger networks nor topologies with different connectivities. Moreover, a growing communication networks trend towards increasing its sizes, and develops its connectivity structures in order to support massive number of users. Hence, the aim of this chapter is to represent an empirical evaluation of the performance of Q-routing on synthesis Internet networks of realistic sizes and connectivity properties.

In addition, we consider several network topologies with the number of nodes set at 500 and the number of connections in the network set at 5000 based on IBM red book which claimed that 500 nodes are large size networks (Murhammer et al., 1999). Different network topologies were constructed which are random connections between nodes and connections formed sequentially by preferential attachment (Batagelj and Brandes, 2005). We also consider a novel of Internet network architecture, known as a heuristically optimized topology which Li et al. (2004) claimed that it is more reflective of the Internet's router level topology than a preferential attachment network. Since, it is designed to support high traffic demand and considering the network maintenance which all traffic from the network edge has to transmit through the network, but just only core of the network has to be increased bandwidth for maintain heavily traffic. By doing these, the Q-routing is represented approach scales to larger problems of adaptive routing when different network connections are subject to increasing amounts of traffic.

4.1 Synthesis Internet network models

Three representative sample of structural network models namely random network, random network with preferential attachment and heuristically optimal topology are introduced in this section.

4.1.1 Random network

The random network as shown in Figure 4.1 is a basic network model which is given a fixed number of nodes and connected each link between pairs of node with probability p. A connectivity process of random network creates a giant component which has attracted a lot of networking research to study its phase transition properties (Chakrabarti and Faloutsos, 2012b; Newman et al., 2002).

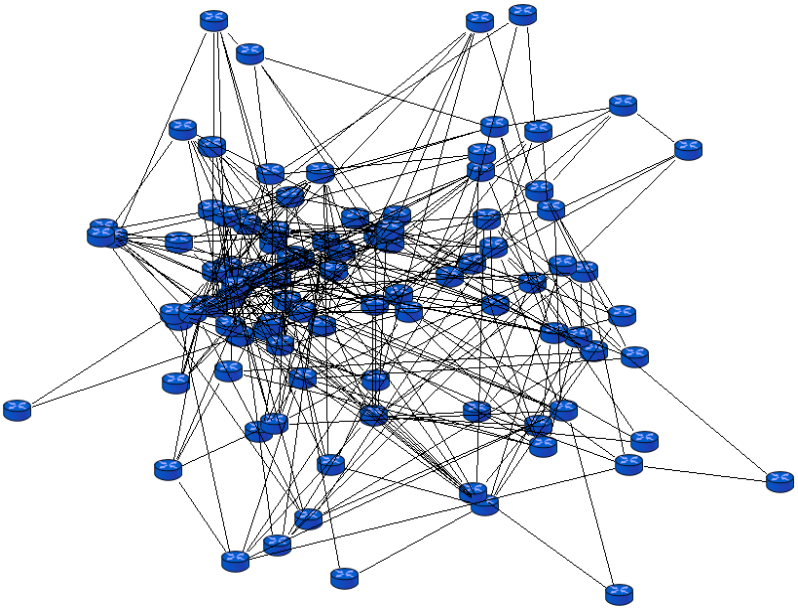


Figure 4.1: The infrastructure of the random network is generated by connecting each link between pairs of node with probability p.

4.1.2 Random network with preferential attachment

In most real networks such as the collaboration and citation networks continually grow a network size by adding nodes and edges according to a power-law distribution (Barabâsi et al., 2002; Dorogovtsev and Mendes, 2002; Newman, 2003). In these networks, new nodes prefer to connect with an existing node which has high number of connections as new nodes are added to the network depending on probability proportional to the current node number of connections. This process is called preferential attachment as shown in Figure 4.2, and the pseudo code of these network construction is given in (Batagelj and Brandes, 2005).

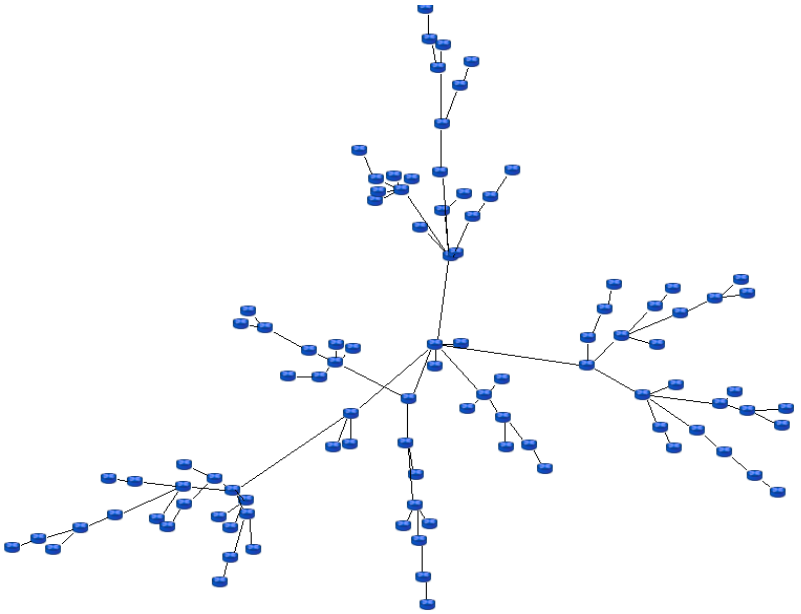


Figure 4.2: The infrastructure of the random network with preferential attachment represents a power-law distribution between number of nodes and its number of connections which few nodes have high number of connections contrasting the other nodes have a small number of connections.

4.1.3 Heuristically optimal topology

The heuristically optimal topology (HOT) as shown in Figure 4.3 is designed based on combining the technological and economic issues in order to apply for the network infrastructure planning (Li et al., 2004). Due to all traffic from the network edged has to be transmitted through the network via interconnected routers which leads to have heavy congestion on core of the network. In addition, the transmission delay will be increased if the network edges far from its destination. Hence, the HOT topology is also designed to minimize the distance between the network core and edge in order to minimize transmission time. Li et al. (Li et al., 2004) suggested that the HOT topology is structural three network layers: core, gateway and edge routers. Furthermore, the HOT topology should represent a power-law distribution which shows relationship in the connectivity between AS-level and router-level. Hence, the first step to create HOT topology is to generate a random network with preferential attachment, and then rewire the network connectivity in order to create three structural network layers. Due to the core of network has to contain heavily congestion, so it should have low connectivity which its speed can be increased to improve network performance, and it also save cost to maintenance. The gateway routers are connected with the core of network by selecting the other higher-degree nodes, and then connected the edge of network according to the degree of each gateway. Since this is not common in network literature, we give here the construction algorithm as pseudo code in algorithm.

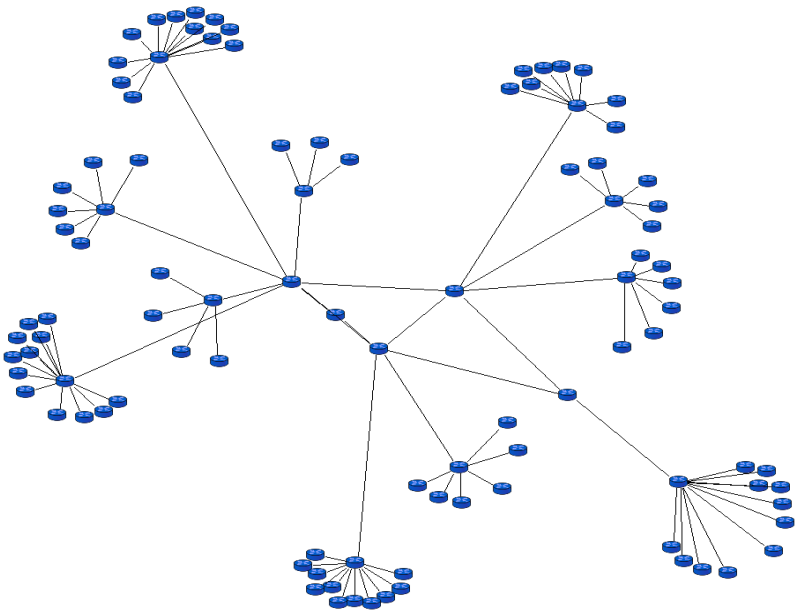


Figure 4.3: The infrastructure of the heuristically optimal topology is designed based on supporting demand of traffic in the future which considers cost of network maintenance by increased bandwidth only core of the network.

4.2 Experimental Settings

These experiments are intended to demonstrate the ability of the adaptive packet routing when Q-routing is employed for packet transmission in terms of average packet delay time, distribution of queue lengths, and how they are tolerant of different traffic conditions on three network topologies.

In this chapter, we set a size of packet based on Ethernet jumbo frames which expanded frame sizes from the original standard IEEE 802.3 in order to reduce the effect of TCP frame overhead. The frame size of packet starts from 1526 bytes and should less than 11,455 bytes because of limit of Ethernet's error checking. However, size of packet frames has an effect on transmission delay in Ethernet link.

Furthermore, each node generates packets are periodic which are sent through entire nodes in the network. Each packet specifies its destination, and it is sent out according to its routing table. Moreover, the simplest queueing model M/M/1 is embedded in each node to store multiple packets with unbounded FCFS queue. In this chapter, we observed queueing delay time which is described how long the packet has to spend time in the queue until it can be transmitted over the link in the network. The parameters for experimental setting are shown in 4.2 and the performance of using Q-routing is compared with the shortest path algorithm.

Traffic load (%)	**Packet size (bytes)**	**Interarrival time (ms)**
10	1,526	1.22
	3,052	2.44
	4,578	3.66
	6,104	4.90
	7,630	6.13
	9,156	7.35
50	1,526	0.24
	3,052	0.48
	4,578	0.73
	6,104	0.97
	7,630	1.22
	9,156	1.46
90	1,526	0.13
	3,052	0.27
	4,578	0.41
	6,104	0.54
	7,630	0.67
	9,156	0.81

Table 4.1: Summary of interarrival time under different traffic loads 10%, 50%, and 90% which each link has limited 100 Mb/s transmission capacity, and the packet sizes vary from 1,526 bytes to 9,156 bytes

Parameters in the experiments			
Categories	**Parameters**	**Values**	**Meanings**
Network	N	500	number of nodes
	N_{link}	5000	number of links
	p	0.04	probability of random connectivity
	m_0	7	initial number of nodes to build PA network
	m	5	number of new links added to PA network at a time
Q-routing algorithm	η	1.0	learning rate
	ϵ	0.1	exploration rate
	t	2000 s	simulation time
	it	10	number of iterations
Data traffic modeling	τ	exp{0.13,...,7.35}	inter-arrival time (ms)
	P_k	{1526,...,9526} bytes	size of a packet
	t_{pd}	0.5 μs	propagation delay
	datarate	100 Mbps	transmission speed
	buffer	unlimited	queue capacity

Table 4.2: Summary of parameters for the experiments which consists of Internet network model, Q-routing and data traffic modeling.

4.3 Experimental Results

Figure 4.5 is a comparison of average delay time between Q-routing and shortest path algorithm while the number of packets is increased until traffic congestion happens. It can be clearly seen that the Q-routing can decrease maximum delay time at load level 6 on three network topologies, and it has slightly different on queueing delay time at load level 1 because of no traffic congestion. Moreover, it can decrease average delay time 60.33% and 58.30% at load level 6 when the PA and the HOT are compared with the random network respectively. In addition, the PA network contains highest average delay because some nodes on the network connected with large number of connections, contrasting with some nodes has only a single way to transmit packets as a result of traffic congestion.

Furthermore, comparing average delay time at load level 6 as shown in Figure 4.5 which compared a high load level between the shortest path and the Q-routing algorithms on three network topologies, it can be clearly seen that the Q-routing algorithm can decrease average queueing delay time 59.46%, 37.93%, and 40.78% on the Random, PR, and HOT networks respectively because the Q-routing algorithm is embedded on each node which reflects current traffic condition by using its Q-values table for making routing decision, and then selects optimal paths for reducing traffic congestion.

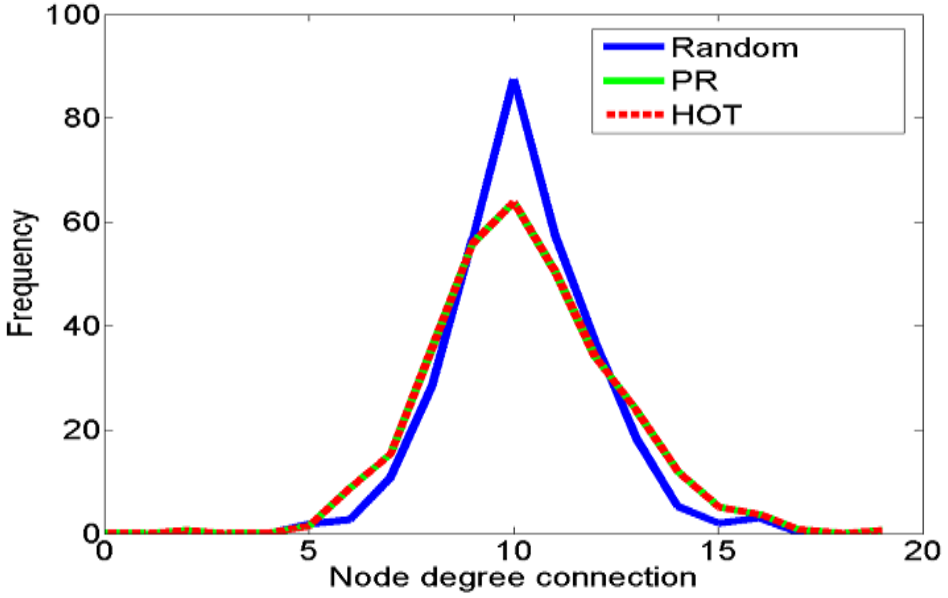


Figure 4.4: The number of node's connections on three network topologies.

Figure 4.6 shows distribution of queue lengths between load levels 1 and 6 where the Q-routing algorithm is employed for packet transmission on three network topologies, and it is clearly seen that distribution of queue length for each link on random network holds smallest number of queue length at both of load levels because the random network is built by connected each node with the same probability 0.04, and leads it has the

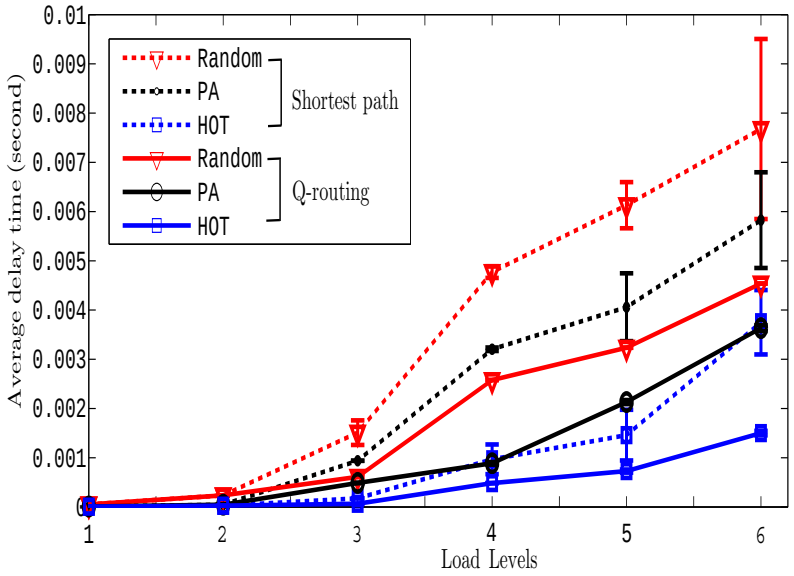


Figure 4.5: Observed average delay time between shortest path and Q-routing while the number of packets is increasing steadily in terms of load levels on three network topologies which each network consists of 500 nodes and 5000 links. The Q-routing can decrease average queueing delay time 59.46%, 37.93%, and 40.78% on the random network, the random network with preferential attachment and the heuristically optimal topology, respectively.

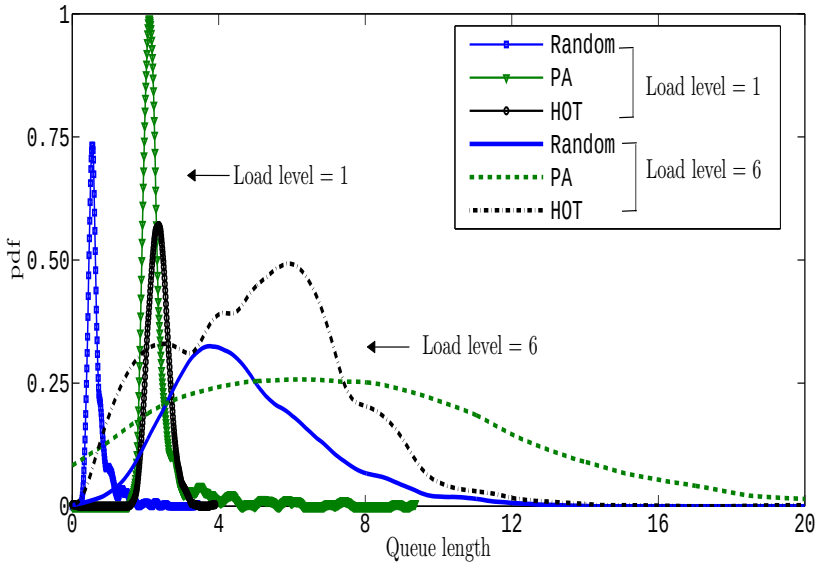


Figure 4.6: Distribution of queue lengths between load levels 1 and 6 on three network topologies when the Q-routing algorithm is employed for packet transmission, and it is clearly seen that the Q-routing algorithm holds smaller queue length at both of load levels because the Q-routing algorithm can find multi paths for forwarding packet which leads to reduce traffic congestion by distributed traffic among links.

same number of node degree connection. However, the PA network is different from the random network with new node prefers to connect existing nodes with higher degree of connection, so it leads this network has been wildly growing up only one side, and this is cause why this network holds highest number of queue length at both load levels when compared with the rest of networks. In addition, the HOT network is constructed from rewiring node degree connection of the PA network, so it can reduce traffic congestion and contains lower queue lengths when compared with the PA network, but it holds higher queue length than the random network because the traffic will congest at core of routers which connected with lowest node degree connection.

Figure 4.7 is a comparison of average delay time on every link between the shortest path and Q-routing on three network topologies which considers only load levels 1 and 6. There are highest number of links which contain highest queueing delay time on the PA network, and in contrast with the random network which has a few number of links contained the high delay time on both load levels. Hence, the random network can contain more load levels, and also has lowest delay time when compared with the PA and the HOT networks.

Figure 4.8 shows fan-out of a node on a random network at low load level which the queue length between the shortest path and the Q-routing is slightly different because of no traffic congestion at the low load level.

Figure 4.9 shows fan-out of a node on a random network at high load level which the queue length of the Q-routing is lower than the shortest path because it can avoid traffic congestion at the high load level.

Figure 4.10 shows fan-out of a node on a random network with preferential attachment at low load level which the queue length between the shortest path and the Q-routing is slightly different because of no traffic congestion at the low load level.

Figure 4.11 shows fan-out of a node on a random network with preferential attachment at high load level which the queue length of the Q-routing is lower than the shortest path because it can avoid traffic congestion at the high load level.

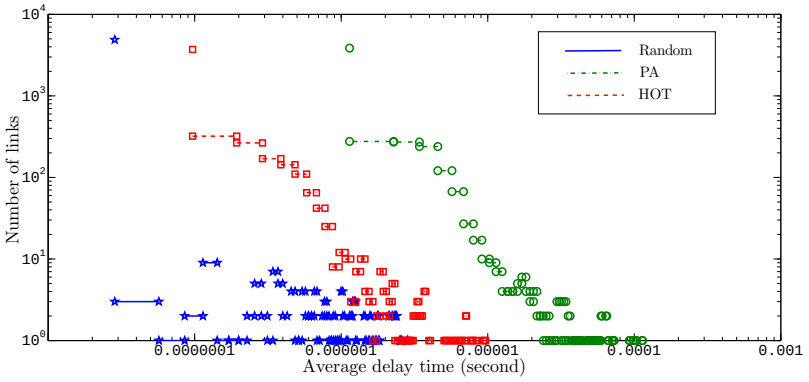
Figure 4.12 shows fan-out of a node on a heuristically optimal topology at low load level which the queue length between the shortest path and the Q-routing is slightly different because of no traffic congestion at the low load level.

Figure 4.13 shows fan-out of a node on a heuristically optimal topology at high load level which the queue length of the Q-routing is lower than the shortest path because it can avoid traffic congestion at the high load level.

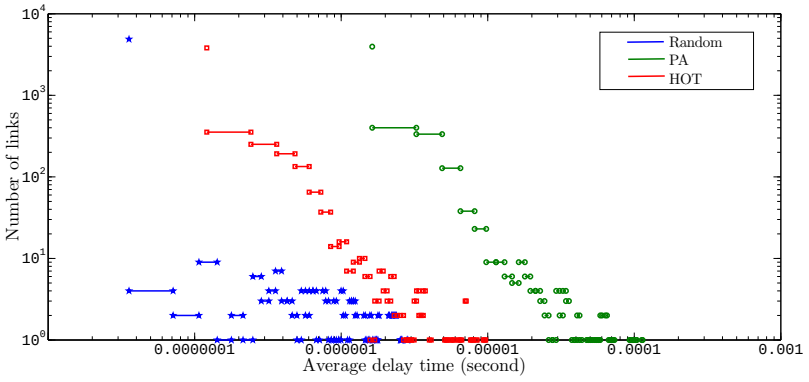
4.4 Conclusions

In summary, the shortest path algorithm is not appropriate for forwarding packets if there are a large number of packets would like to be sent into the network because it uses static routing table for packet transmission. In addition, it also leads to easily get traffic congestion since it always used the same path for packet transmission. Moreover, the Q-routing can find the same routes as the shortest paths if it learns until convergent time. Hence, the Q-routing algorithm is appropriate for forwarding packets especially if the number of packets is steadily increasing because its routing table can be updated to select suitable it's neighboring node. Since, it can avoid congested paths to send the packet to its destination as a result of decreasing average delay time, and also contains large number of packets as shown good results on the experimental results section.

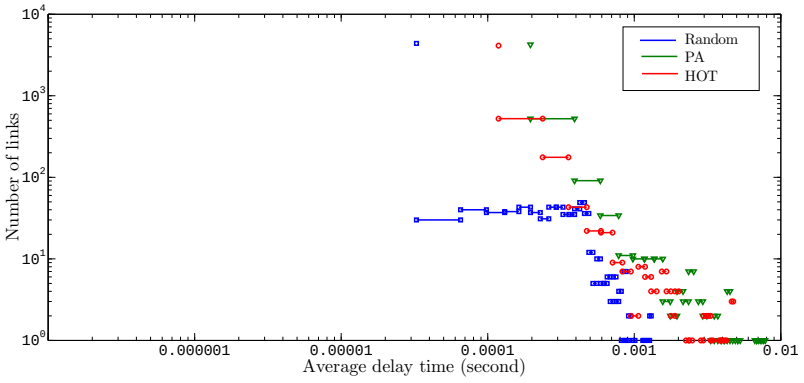
In addition, it will be more powerful if it is employed on real network topology because the Q-routing can contain higher traffic loads than the shortest path. Furthermore, it can decrease average delay time while more number of packets is steadily increasing into the network. Hence, the Q-routing in this thesis is also employed on the JANET which is a real network for supporting education in UK, and it is described in next chapter.



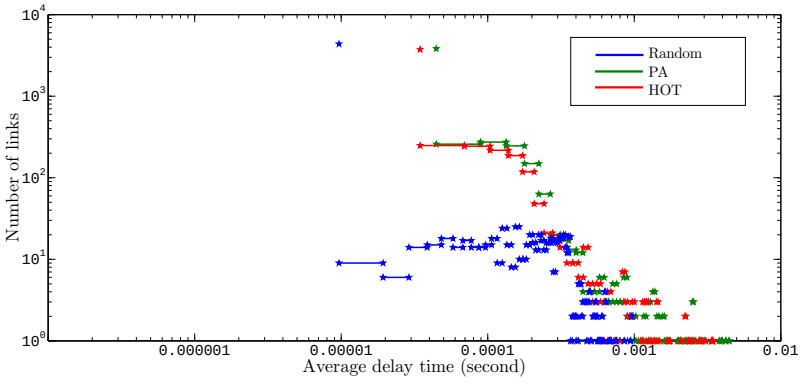
(a) Shortest path at load level = 1



(b) Q-routing at load level = 1



(c) Shortest path at load level = 6



(d) Q-routing load at level = 6

Figure 4.7: Comparing average delay time on three network topologies at load level 1 and 6 when the shortest path and Q-routing were employed for packet transmission.

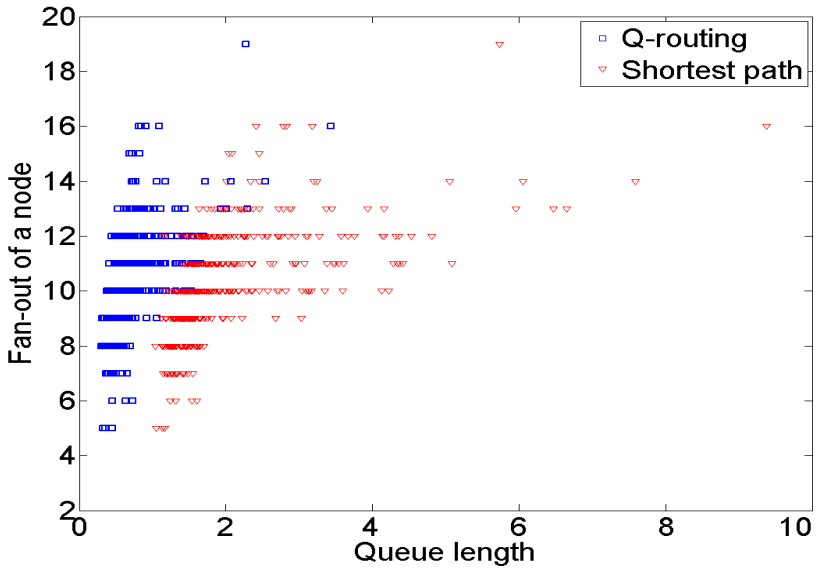


Figure 4.8: Fan-out of a node on a random network at low load level.

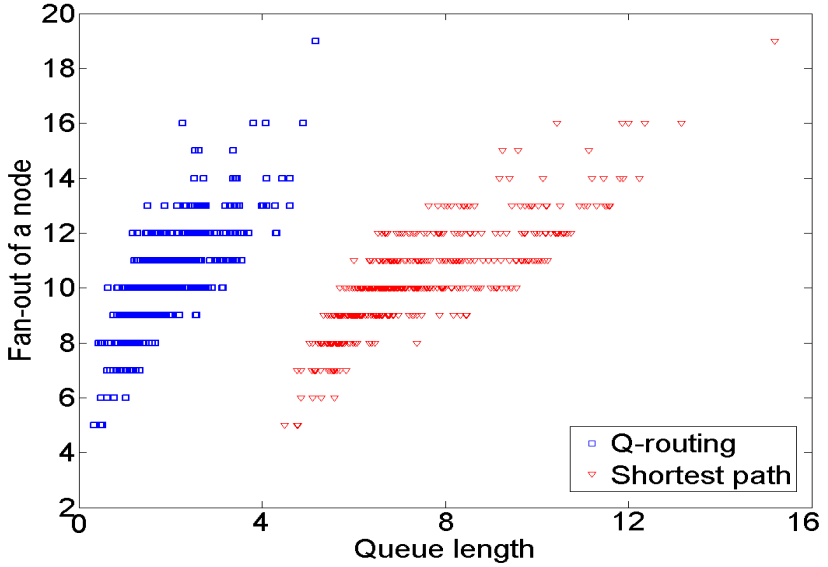


Figure 4.9: Fan-out of a node on s random network at high load level.

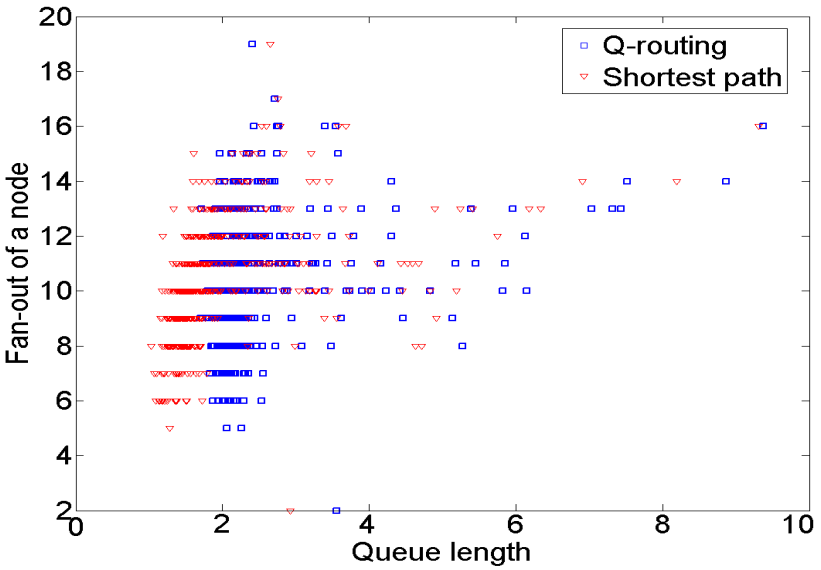


Figure 4.10: Fan-out of a node on a random network with preferential attachment at low load level.

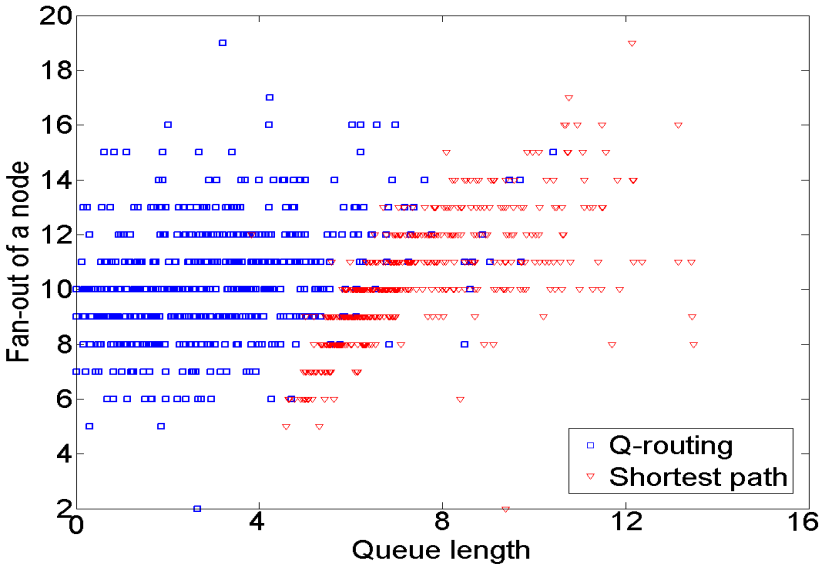


Figure 4.11: Fan-out of a node on a random network with preferential attachment at high load level.

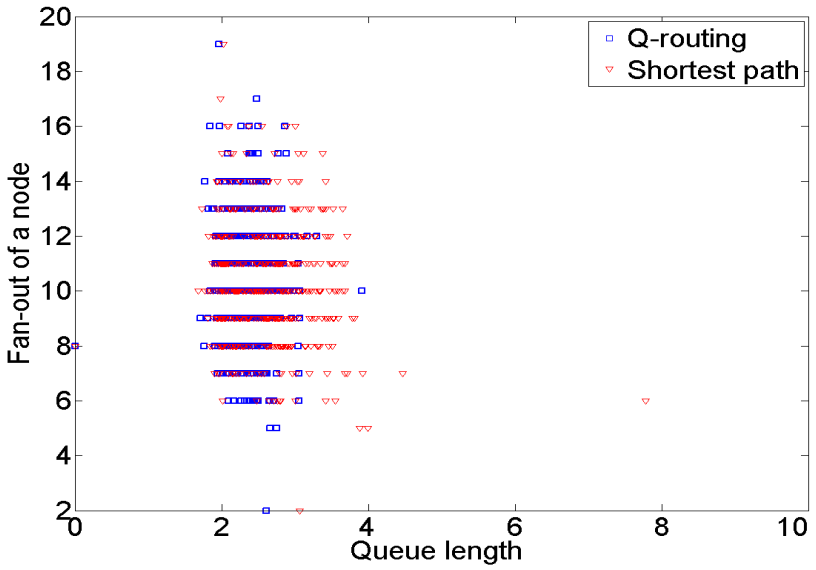


Figure 4.12: Fan-out of a node on a heuristically optimal topology at low load level.

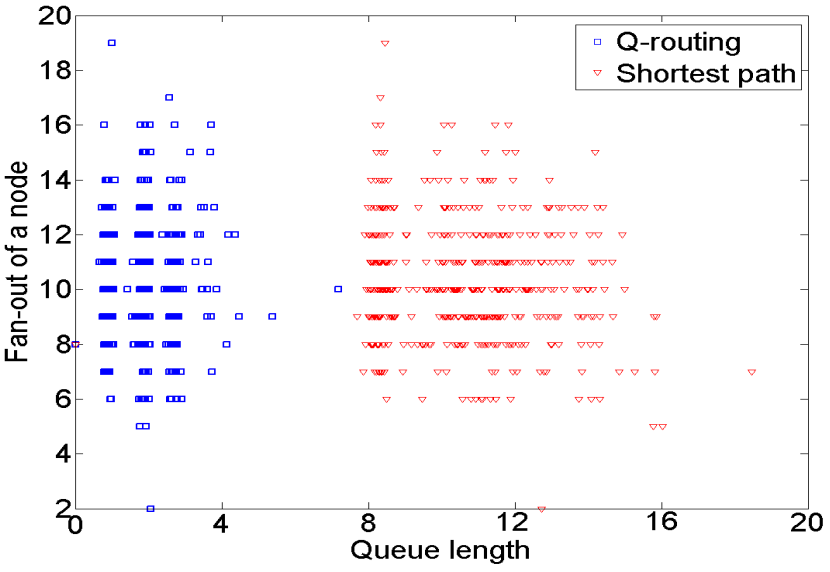


Figure 4.13: Fan-out of a node on a heuristically optimal topology at high load level.

Chapter 5

Adaptive dynamic packet routing on JANET network topology

During the years, reinforcement learning which a branch of machine learning has been successful applied to optimize problems solving on various contexts such as routing optimization problem in communication networks. In previous chapters, the Q-routing is successful to employ for finding optimal routing paths in order to forward packets under different traffic conditions on small and large synthesis Internet networks. The previous experimental results show that the Q-routing can find optimal routing paths which avoids traffic congestion as a result of decreasing average delay time and queue length. However, it has not been applied on real network topology which it is a good chance to apply on real network communication for supporting high traffic demand in the future.

Hence, the Q-routing is evaluated the effectiveness of routing information feedback under different traffic conditions against Dijkstra's algorithm on real United kingdom (UK) network topology; JANET (Joint Academic Network) network, in order to explore the possibilities of adaptive routing algorithm according to support high traffic demand.

5.1 JANET connectivity

Since the UK is famous for educational systems including corporate reputation research centers, many journals have been published to keep abreast of development. Hence, exchanging knowledge information among research and educational centers has an important for supporting economy, society, and environment in the future. JANET network is established for educational serving between UK research and education community which provides high speed connection, and covers the UK from lands end to John O'Groats and everywhere in between. The network backbone runs at 100 Gbit/s,

and capacity of interconnection is around 40 Gbit/s as shown in 5.1. In addition, the university of Southampton is one of three universities across the UK is collaborating on fiber infrastructure which aims to develop photonics and optical systems for supporting future network architectures.

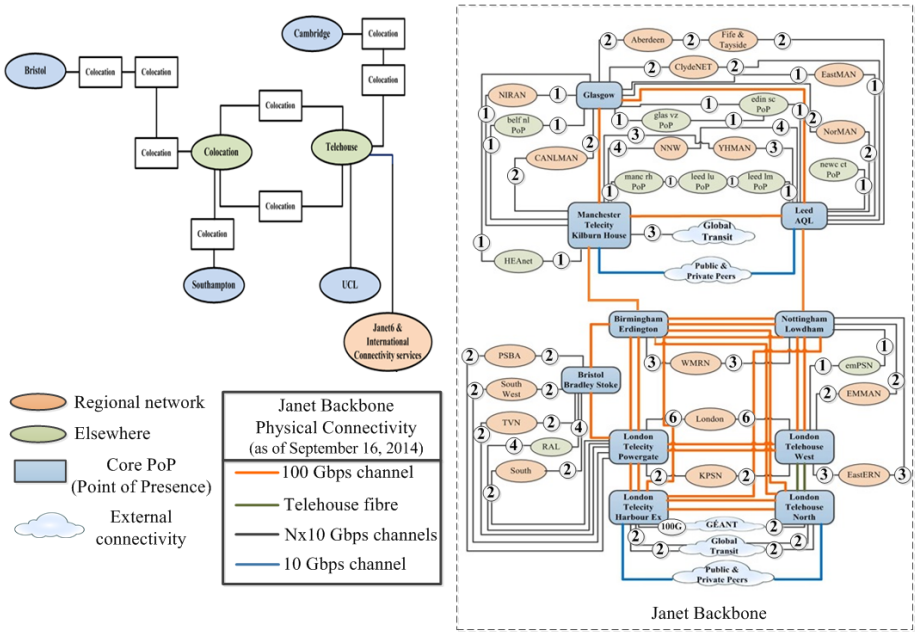


Figure 5.1: JANET network is the network for supporting UK research and education which is operated by United Kingdom Education and Research Networking Association (UKERNA) and the Joint Information Systems Committee (JISC).

5.1.1 Experimental Settings

These experiments are intended to demonstrate the ability of the Q-routing algorithm for packet transmission in term of average packet delay time and distribution of queue lengths, and how they are tolerant of congestion under different traffic conditions on the JANET network.

In this thesis, we consider different traffic loads which each link has limited 100 Mb/s transmission capacity, and the packet sizes vary from 1,526 bytes to 4,578 bytes which its interarrival time is a leading cause of traffic congestion on the network. Table 3.1 is an example of arrival rate and interarrival time when packet size is 1,526 which is a minimum packet size considering how it has effect on traffic congestion when traffic loads vary from 10% to 95%.

Moreover, Table 5.1 is summarized interarrival time under different traffic loads 10%, 50% and 90% which represent low, medium, and high traffic load levels on the network.

In addition, packet sizes vary from 1,526 bytes to 4,578 bytes are also considered how it has effect on traffic congestion.

Traffic load (%)	**Packet size (bytes)**	**Interarrival time (ms)**
10	1,526	1.22
	2,289	1.83
	3,052	2.44
	3,815	3.05
	4,578	3.66
50	1,526	0.24
	2,289	0.36
	3,052	0.48
	3,815	0.61
	4,578	0.73
90	1,526	0.13
	2,289	0.20
	3,052	0.27
	3,815	0.34
	4,578	0.41

Table 5.1: Summary of interarrival time under different traffic loads 10%, 50%, and 90% which each link has limited 100 Mb/s transmission capacity, and the packet sizes vary from 1,526 bytes to 4,578 bytes

5.1.2 Experimental Results

Considering a real network topology; JANET which each link has limited transmission capacity, and routing algorithm is embedded in each node to find optimal routing policy for forwarding packets. Packet arrival rate is increased in order to introduce traffic congestion. In addition, Q-routing and shortest path are compared when traffic load is increased on the network. Average delay time and distribution of queue length are studied a performance of routing algorithm while traffic is increasing until congestion on the network.

Figure 5.2 shows the relationship between number of nodes and its connection which represents power-law degree behavior by a few number of nodes connected with high number of connection, and contrasting with the other nodes on the network. In addition, the highest number of connection on the JANET network is 8.

Figure 5.3 shows the relationship between number of hops and number of nodes on the JANET network where the packet size 1526 bytes is sent through the entire network. In addition, it is slight different between number of hops when the Shortest path and Q-routing are employed for packet transmission because they always use the same routing table for forwarding packets.

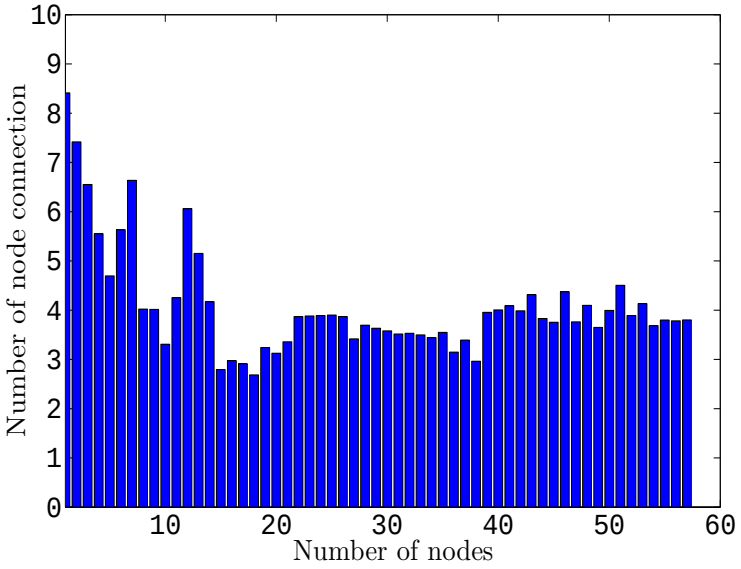


Figure 5.2: Average number of node connection on JANET network.

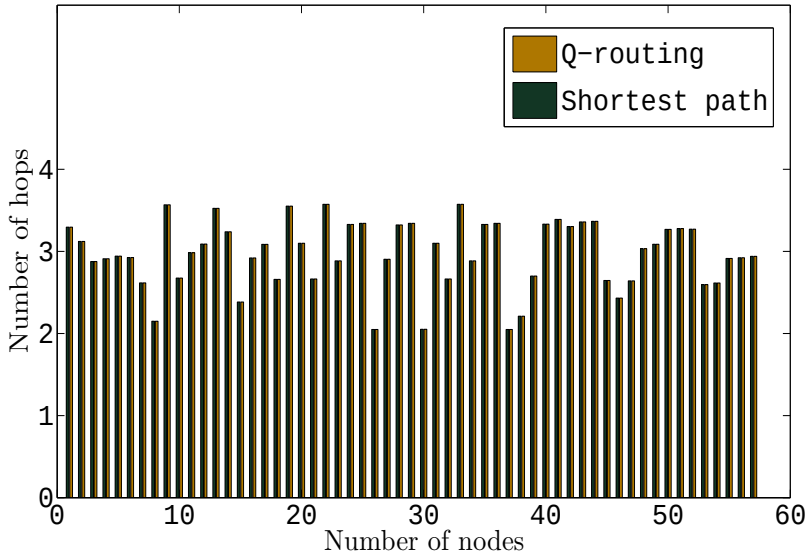


Figure 5.3: Average number of hops at low load level on JANET network.

Figure 5.4 shows the relationship between number of hops and number of nodes on the JANET network where the packet size 4578 bytes is sent through the entire network. In addition, it is different between number of hops when the Shortest path and Q-routing are employed for packet transmission because the Q-routing prefers to send packets out with longer paths because of avoiding traffic congestion.

Figure 5.5 shows the average delay time between Shortest path and Q-routing while the number of packets is increasing steadily in terms of load levels on JANET network. It

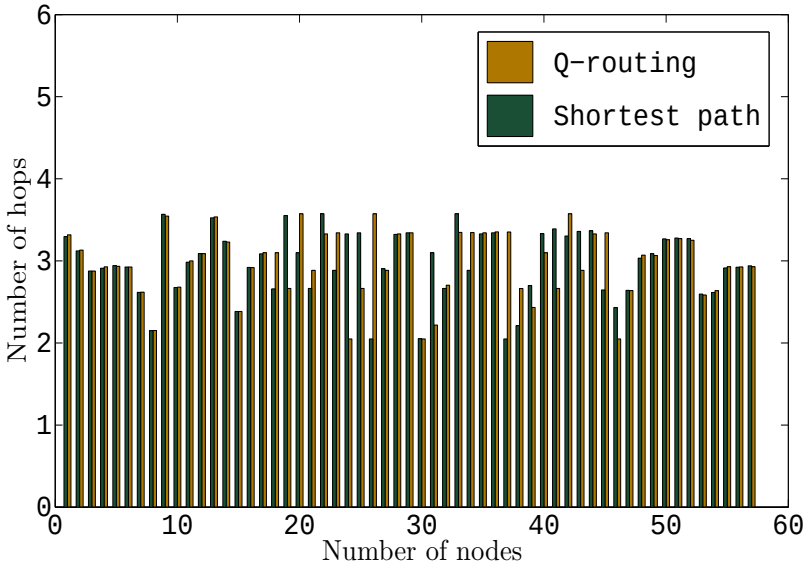


Figure 5.4: Average number of hops at high load level on JANET network.

is clearly seen that, the Q-routing has significantly decreased the average delay time on the JANET network at high load level.

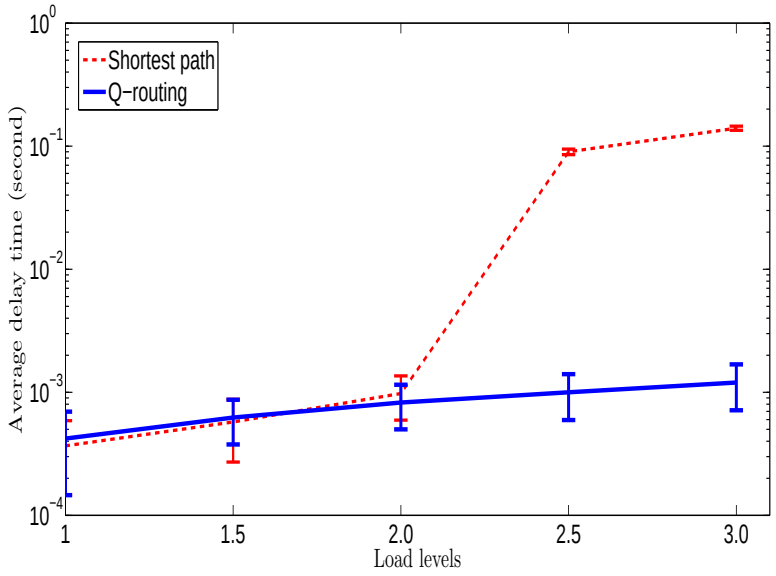


Figure 5.5: Average delay time between Shortest path and Q-routing while the number of packets is increasing steadily in terms of load levels on JANET network.

Figure 5.6 shows the distribution of queue lengths between shortest path and Q-routing at load levels 1 and 3 on JANET network where the Q-routing contains lower queue length than the Shortest path because it can find optimal paths to transmit packets with avoiding traffic congestion.

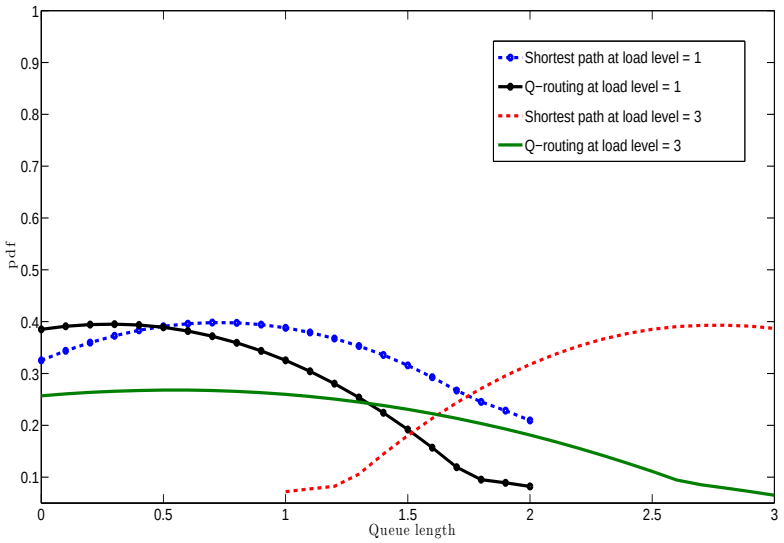


Figure 5.6: Distribution of queue lengths between shortest path and Q-routing at load levels 1 and 3 on JANET network.

In addition, the second part of experiment in this chapter is applied the M/M/1/K queueing model on every node in the JANET network in order to observe how the Q-routing can manage routing tables which helps to avoid dropping packets. In addition, we consider extremely case which K can store only 1 packet.

Figure 5.7 shows the comparing number of nodes for packet drop between Shortest path and Q-routing where the packet sizes vary from 1526 bytes to 4578 bytes, and on JANET network. It is clearly seen that the Q-routing can decrease the number of nodes which drop packets on the JANET network.

Figure 5.8 shows the comparing percentage of number of nodes for packet drop between Shortest path and Q-routing where the packet sizes vary from 1526 bytes to 4578 bytes, and on JANET network. It is clearly seen that the Q-routing can decrease more than half of the number of nodes which drop packets on the JANET network.

Figure 5.9 shows the comparing amount of packet drop (bytes) between Shortest path and Q-routing where the packet sizes vary from 1526 bytes to 4578 bytes, and on JANET network. In addition, it is slightly different amount of packet drop at the packet size 1526 bytes. In contrasting with the packet size 4578 bytes, the Q-routing can decrease the amount of packet drop 68% .

However, the percentage of decreasing amount of packet drop between Shortest path and Q-routing where the packet sizes vary from 1526 bytes to 4578 bytes on JANET network as shown in Figure5.10 shows that the Q-routing can decrease the amount of packet drop more than 50%. Hence, the Q-routing is suitable for packet transmission on the real network because it can avoid traffic congestion as a result of decreasing average

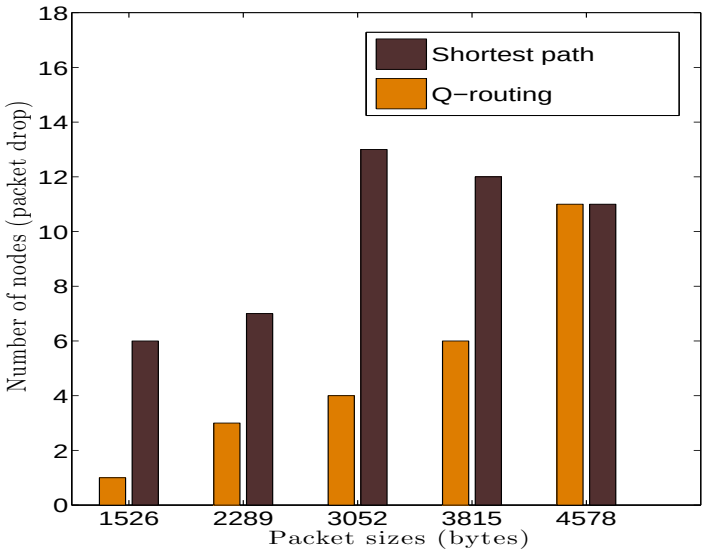


Figure 5.7: Comparing number of nodes for packet drop between Shortest path and Q-routing where the packet sizes vary from 1526 bytes to 4578 bytes, and on JANET network.

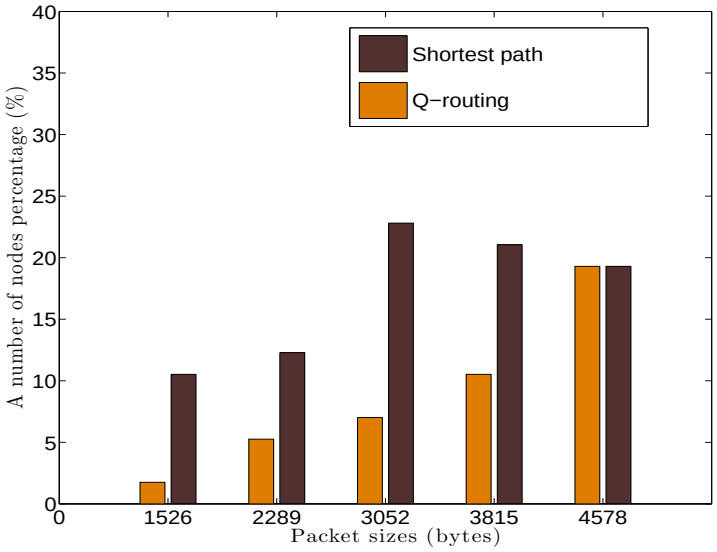


Figure 5.8: Comparing percentage of number of nodes for packet drop between Shortest path and Q-routing where the packet sizes vary from 1526 bytes to 4578 bytes, and on JANET network.

delay time, and it can decrease the amount of packet drop when the router has capacity limit.

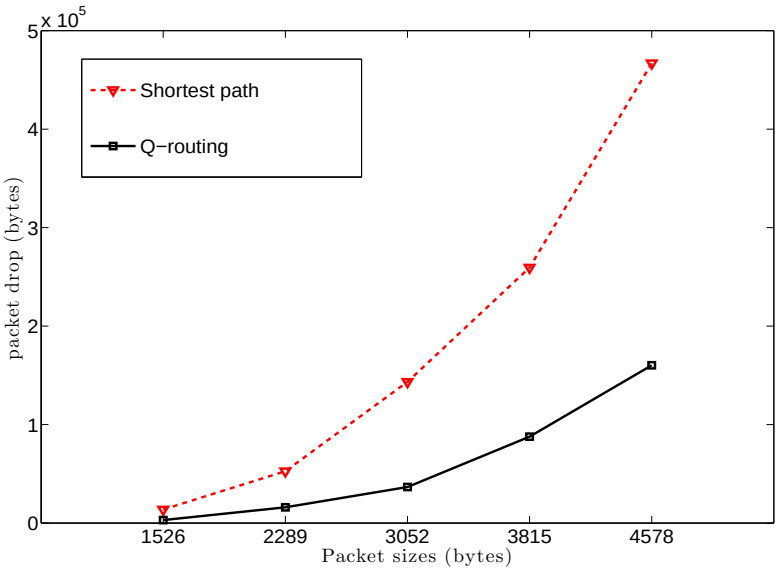


Figure 5.9: Comparing amount of packet drop (bytes) between Shortest path and Q-routing where the packet sizes vary from 1526 bytes to 4578 bytes, and on JANET network.

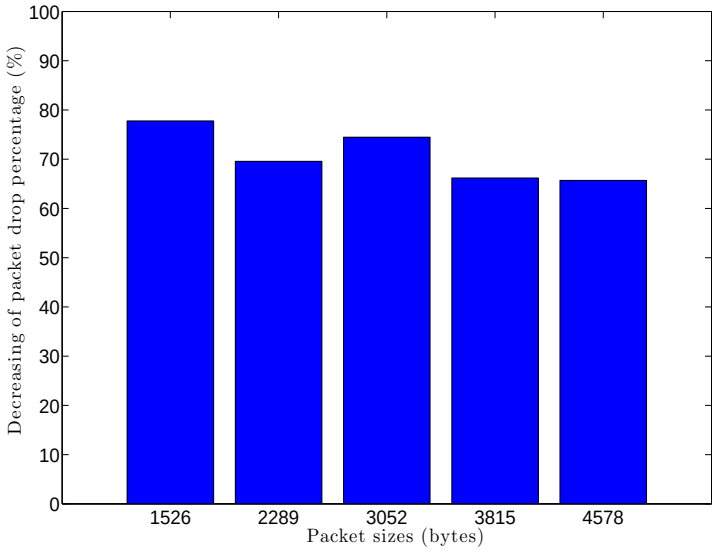


Figure 5.10: Percentage of decreasing amount of packet drop between Shortest path and Q-routing where the packet sizes vary from 1526 bytes to 4578 bytes on JANET network.

5.2 Conclusions

In this chapter, the Q-routing is employed on a real network architecture; the JANET network. The JANET network is a high-speed network infrastructure as know as the Joint Academic Network (JANET) has been cooperated between the U.K. academic

and research network, which is aimed to be driven mass productive research, and provided excellent service for tons of users. In addition, the JANET has been designed to support vastly users by increasing capacity, resilience and flexibility. Moreover, the JANET network aurora2 has been emphasized by using its own dedicated dark fibers for supporting researchers to develop the highlight communications technologies for the future Internet without breaking the current Internet. However, it has to ensure that its flexible architecture should appropriately respond to arising new technologies as come with massive users, and the quality of service measurements such as packet delivery time have to develop a strategy for user satisfaction. Hence, routing algorithms play an important role in sorting the best path for packet transmission which should avoid congestion paths. The Q-routing is one of routing algorithms which can select multi-path for packet transmission by avoiding traffic congestion while the large number of packets is increasing in the network as the results showed in the experimental results section. Furthermore, it will be interesting to apply the Q-routing on real network topology such as the JANET network to see how the Q-routing can sustain the high traffic conditions. In this thesis, we employed two routing algorithms which are the shortest path and the Q-routing for packet transmission while the number of packets are steadily increasing into the network.

The experimental results show that the Q-routing can show a good performance to decrease average delay time, and contain lower distribution of queue length than the shortest path on a real network JANET while the number of packets is increasing into the network. Hence, we confirm that the Q-routing not only suitable for decreasing average delay time and queue length on synthetic Internet networks, but it also works as well on the real network like JANET.

Chapter 6

Pareto Q-learning based on the Deep Sea Treasure World Case Study

In this chapter, the RL method namely Q-learning has been employed for studying multi-objective problems by combining with the Pareto front in order to get the optimal solution. In addition, we are interested in the deep sea treasure world (DST) case study which is proposed by (Vamplew et al., 2011). Moreover, it is claimed that the optimal path of each treasure is a part of the Pareto front which is learned by the Q-learning. Furthermore, the purpose of DST is discovering the highest treasure value while taking a minimum number of hops as well as a minimum time consuming. Hence, the Pareto-Q-learning is the one method which is interesting for solving multi-objective problems.

6.1 Multi-objective Reinforcement Learning

According to Vamplew et al. (2011), multi-objective reinforcement learning (MORL) is divided into two classes depending on how many policies are learned which are called single policy class and multiple-policy class. In addition, the single policy class is learned from the set of objectives as a result of getting the best single policy. In contrasting with multiple-policy, it is learned in order to get the set of policies which is close to the Pareto front. Gábor et al. (1998) applied the reinforcement learning on multi-criteria decision problems by setting a threshold to get the optimal policy which responds to a set of objectives. In addition, Gábor et al. (1998)'s work is an example of the single policy class which requires only an optimal policy to achieve multi-goal of the system. However, the single policy class can lead to a sub-optimal solution depending on the method of defining threshold (Vamplew et al., 2011). Hence, the multi-policy class is

introduced to solve multi-objective problems which a set of policies should provide more flexible answers rather than only fixed with a single policy.

In this chapter, the multi-policy class based on Q-learning is interested in solving multi-objective problems by extending from the original Q-learning which the Q-values vector is used to represent a feedback on a set of objectives. In addition, the Pareto Q-learning is a method of multi-policy class which was applied on the Deep Sea Treasure World. Furthermore, it was introduced by Vamplew et al. (2011) which motivated us to understand how it works, and inspire us to extend this work into multi-objective in communication network like Internet.

6.2 Pareto Q-learning

The original Q-learning was introduced by Watkins and Dayan (1992) which aims to solve a single objective problem based on reward signaling from the next state to estimate an optimal policy. In addition, an action selection is chosen at each time step based on selected action mechanisms such as ϵ-greedy which selects a possible action based on its probability. Moreover, the Q-learning can learn to improve the network without knowing the complete model of the network. However, the single objective will achieve only one goal which cannot improve overview of the network. Hence, a single selected action is extended to a set of actions which responds to multi objectives which these selected actions should rely on Pareto front to provide optimal policies. In addition, the Q-values of original Q-learning which is used to observe the reward signal between state and its action, is also extended to a set of Q-vectors (Van Moffaert and Nowé, 2014). Furthermore, the algorithm of Q-learning is described in more detail by Sutton and Barto (2011) where the Pareto Q-learning will provide the flowchart of learning process as shown in Figure 6.1 as follows:

6.3 Experiments

The DST simulation is an example of multi-objective based on Pareto Q-learning which has a set of Q-vectors to find optimal policies. In addition, there are two objectives which the submarine would like to discover; the highest value of treasure and the minimum time consuming. Furthermore, the selected action mechanism in this chapter is ϵ-greedy which is applied to explore and select possible actions by random with probability ϵ. The maximum time step of DST simulation (t) is 2000 which is used to observe when the simulation converges into optimism. Moreover, the learning rate (α) is specified 0.8, discount factor (γ) is 0.1, and probability of random action ϵ is 0.3. The reward signal is provided 1 for every step to find the treasure, but a reward could be 100 if the submarine goes into the rock undersea region. If the submarine reaches the goal state, the value of

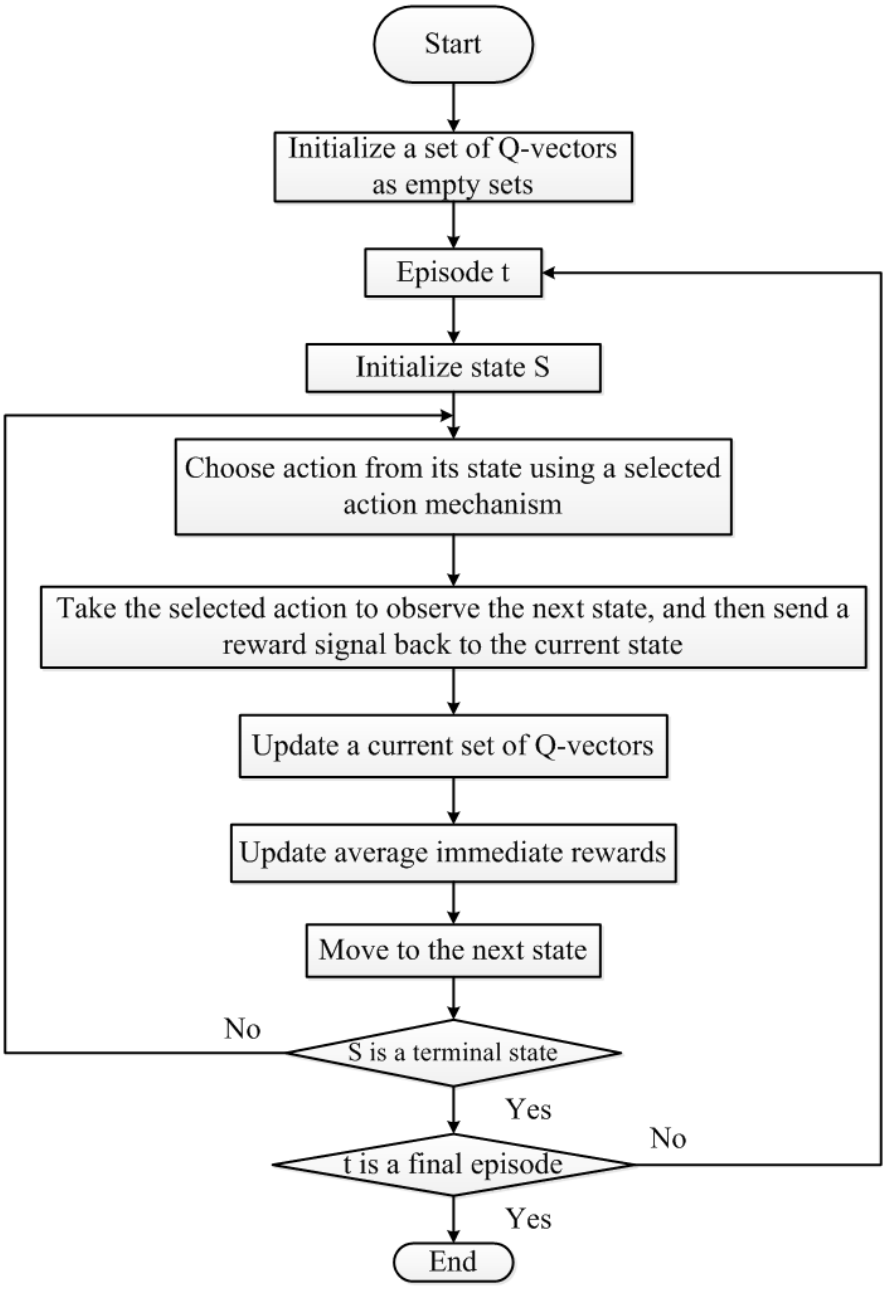


Figure 6.1: The flowchart of Pareto Q-learning.

treasure and time consuming will be provided back to the state of submarine. For an initial state, the submarine starts to find the treasure at the top left corner, and it will discover the highest value by moving to the next states until reach its goal. Hence, the highest value of treasure and the minimum time consuming which the submarine can discover, are 124 and 19, respectively.

In addition, Figure 6.2 represents the environment of deep sea treasure world which

consists of ten states of treasure values, the rock seabed, and the submarine starts discovering the treasure from the top left corner. In addition, the optimal policies of this problem should rely on the Pareto front. Moreover, the parameters of learning process such as α, and a probability of selected action ϵ are also examined how they have an affect on the DST environment as the experimental results shown in the next section.

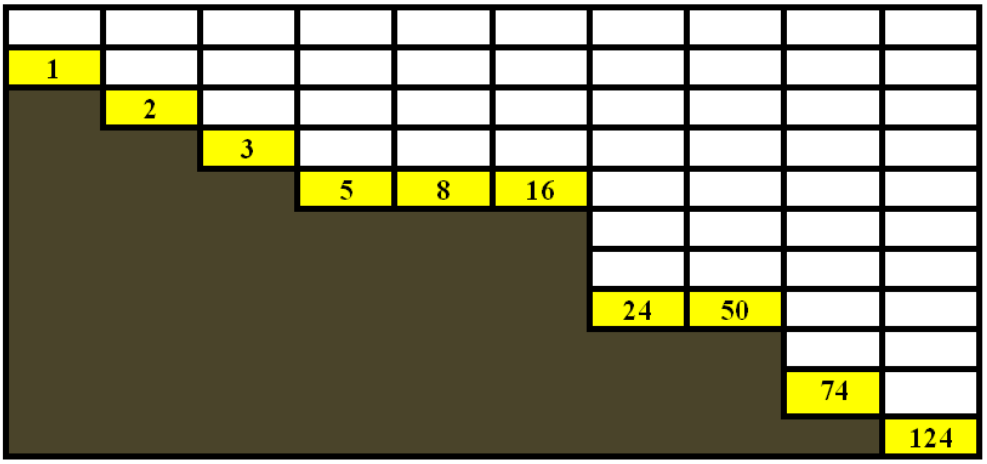


Figure 6.2: The *DST* environment consists of the white cells, the darker tan cells and the yellow cells which represent possible states to find treasure, the rock seabed, and the goal states, respectively.

6.4 Experimental Results

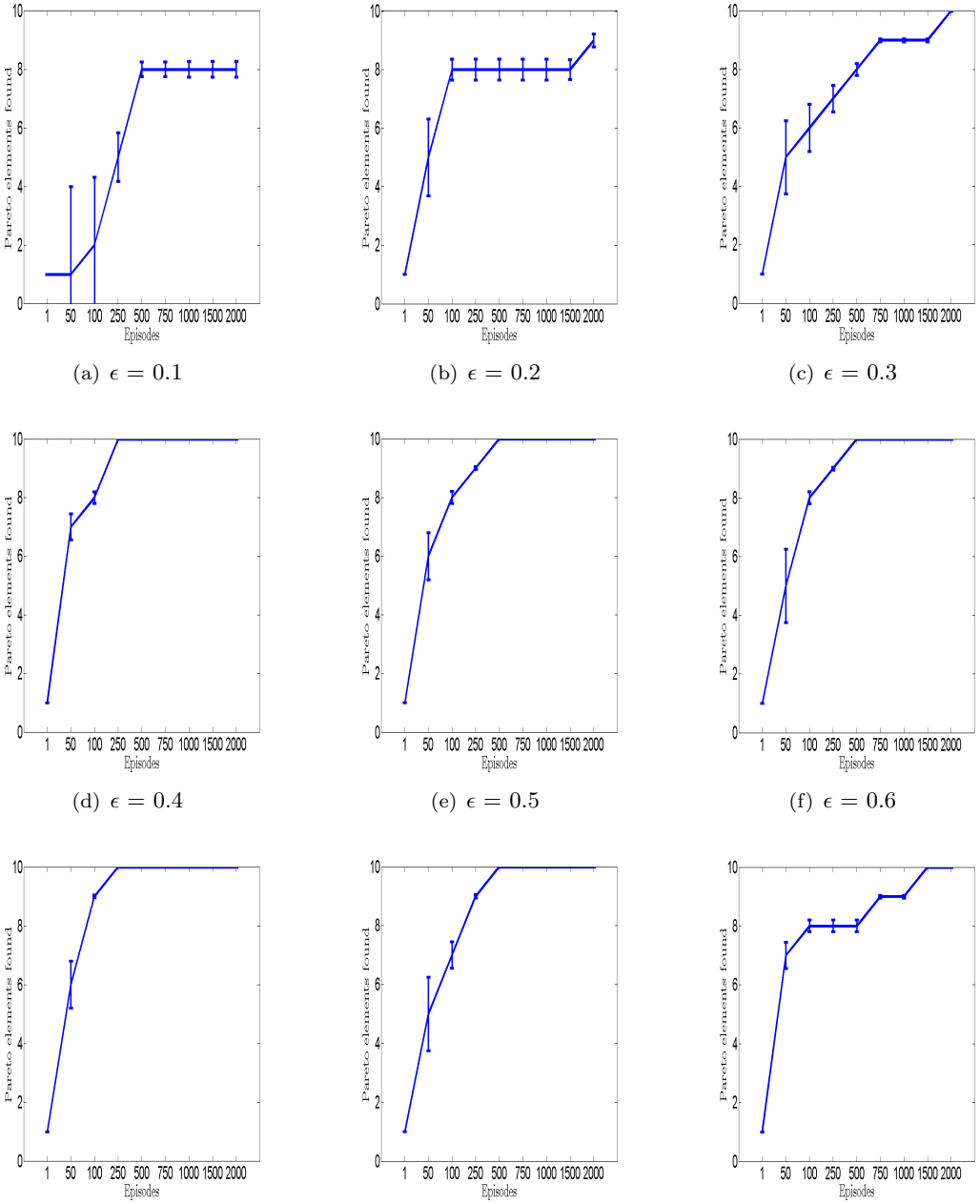
Since, the Q-learning consists of learning rate, discount factor, episode and probability of selected action which these parameters get involved in a learning process, and having a relationship among them to provide optimal policies. Hence, they are examined in order to understand how to tunning these parameter to achieve the goals.

Firstly, the The time step of DST simulation (t) varies from 1 to 2000 in order to observe convergence time which allows the submarine running this algorithm to quickly and reliably converge. Secondly, the ϵ also varies from 0.1 to 0.9 in order to explore and take a random action according to the probability which $\epsilon = 0.1$ means the submarine has chanced to explore only 10% for selecting an action from all possible actions, and 90% for selecting the action depending on the greatest estimated values as well as the greedy action (Sutton and Barto, 2011). However, the action should be explored before selecting because Sutton and Barto (2011) claimed that at least one of these possible actions probably is actually better than the greedy action. Finally, the learning rate (α) and the discount factor (γ) should be varied in order to study how they have an affect on the DST bi-objective environment.

Figure 6.3 shows the Pareto front on the DST bi-objective environment which the ϵ varies from 0.1 to 0.9, and the time step of simulation varies from 1 to 2000. In addition, the learning rate (α) is specified 0.9, discount factor (γ) is 0.1. It is clearly seen that the ϵ = 0.1, the Pareto front can be found only 8 goals states, and it will be found all 10 goal states when the ϵ is more than 0.2 due to the exploring selected action is actually better than greedy action. However, the increasing of ϵ until it is nearly 1 has not guaranteed the optimal policies because the action will be selected based on random probability rather than greediness. Hence, the $\epsilon = 0.3$ is selected to apply on the DST environment in this thesis because it is the first probability of random action which is found all 10 goal states.

Furthermore, Figure 6.4 shows the probability of goal states visiting on the DST bi-objective environment which the ϵ also varies from 0.1 to 0.9, and the time step of simulation is 2000. It is clearly seen that all of goal states will be visited when the ϵ more than 0.2, and the probability of higher value of treasure (> 50) visiting will be arisen in the ϵ range of 0.2 to 0.5. However, the probability of highest value of treasure (124) will be visited decreasingly where the ϵ is more than 0.5 because it prefers to visit the goal states which the values are less than 50.

In addition, Figure 6.5 shows the results of convex hull on the DST bi-objectives environment which the ϵ varies from 0.1 to 0.9 where x bar represents time consuming, and y bar represents the value of treasure. It is clearly seen that if the ϵ is less than 0.4, the highest value of treasure is found by taking minimum time consuming. In contrasting, if the ϵ is more than 0.4, the value of treasure will be found by taking more time consuming which is the reasonable reason why the $\epsilon = 0.3$ is selected to apply on the DST bi-objectives environment.

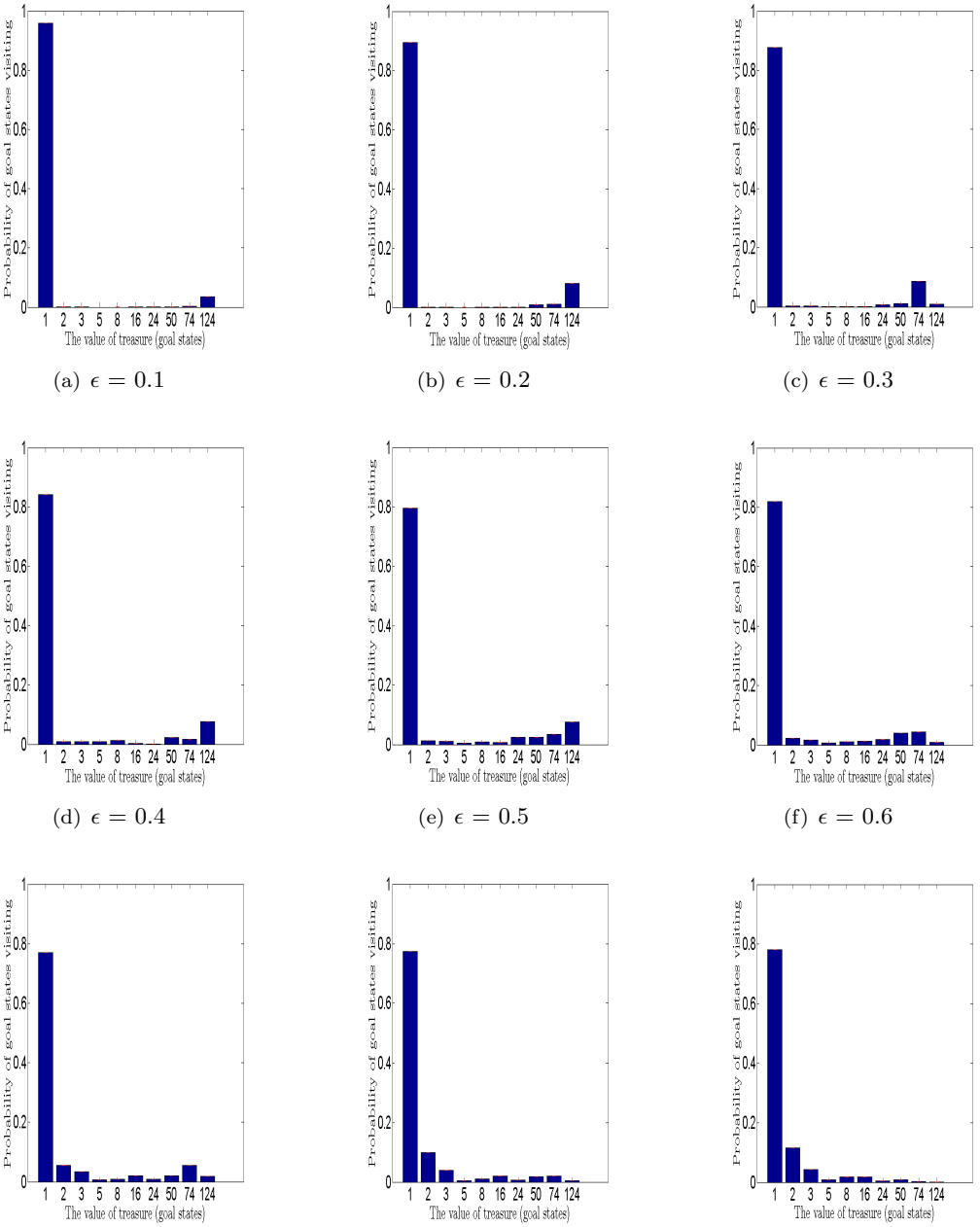


(a) $\epsilon = 0.1$ (b) $\epsilon = 0.2$ (c) $\epsilon = 0.3$

(d) $\epsilon = 0.4$ (e) $\epsilon = 0.5$ (f) $\epsilon = 0.6$

(g) $\epsilon = 0.7$ (h) $\epsilon = 0.8$ (i) $\epsilon = 0.9$

Figure 6.3: The results of Pareto front on the DST bi-objective environment which the ϵ varies from 0.1 to 0.9, and the time step of simulation varies from 1 to 2000.

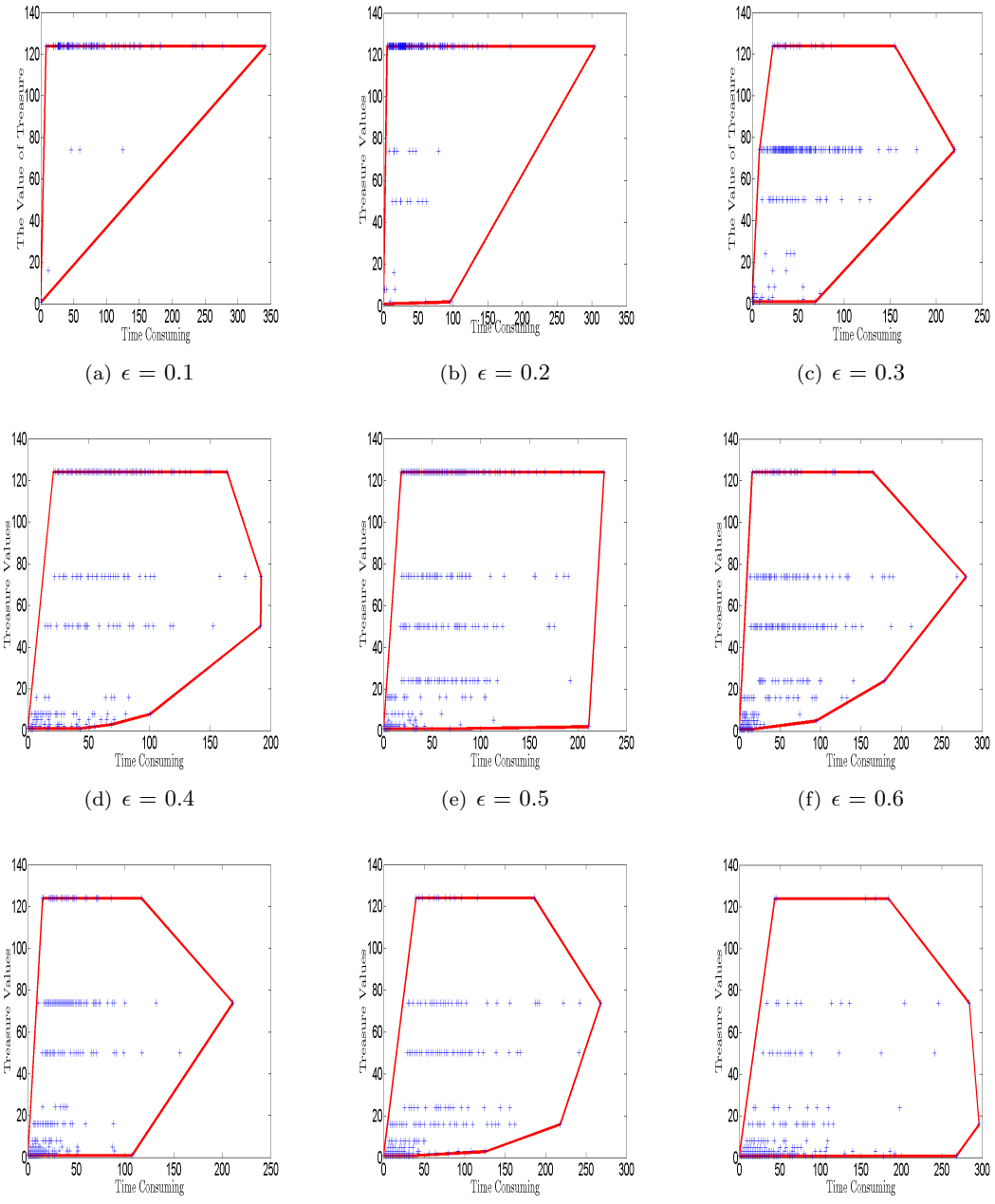


(a) $\epsilon = 0.1$ (b) $\epsilon = 0.2$ (c) $\epsilon = 0.3$

(d) $\epsilon = 0.4$ (e) $\epsilon = 0.5$ (f) $\epsilon = 0.6$

(g) $\epsilon = 0.7$ (h) $\epsilon = 0.8$ (i) $\epsilon = 0.9$

Figure 6.4: The results of probability of goal states visiting on the DST bi-objective environment which the ϵ varies from 0.1 to 0.9, and the time step of simulation is 2000.



(a) $\epsilon = 0.1$ (b) $\epsilon = 0.2$ (c) $\epsilon = 0.3$

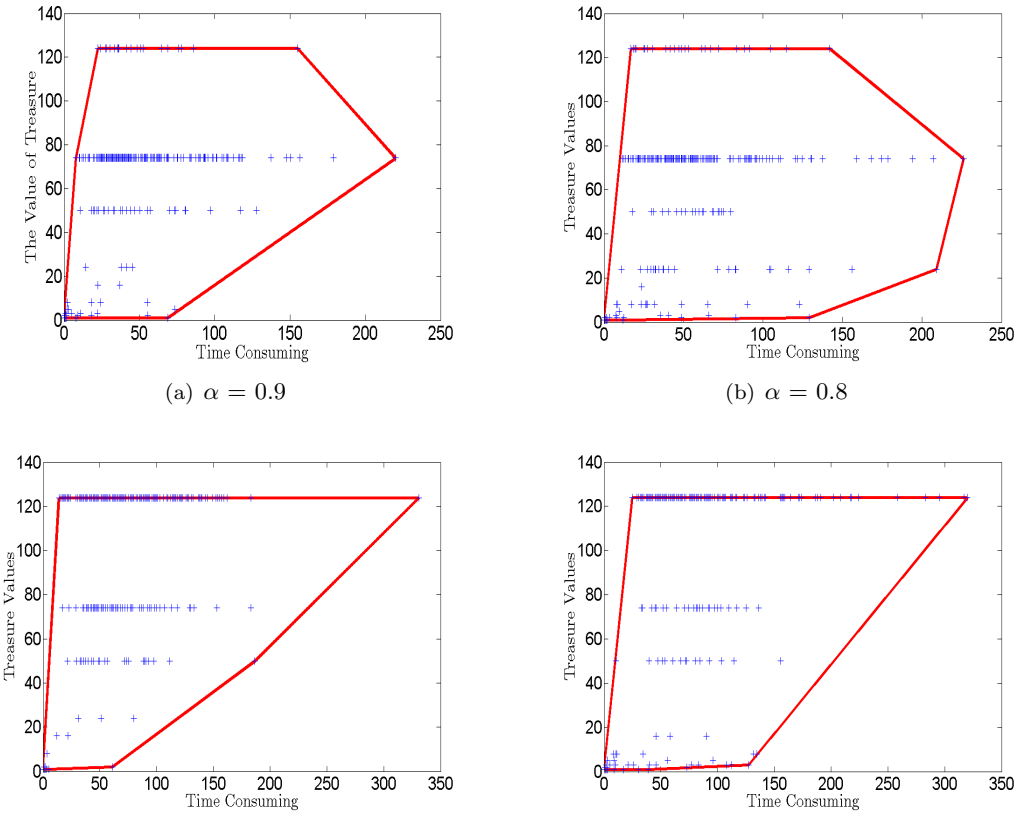
(d) $\epsilon = 0.4$ (e) $\epsilon = 0.5$ (f) $\epsilon = 0.6$

(g) $\epsilon = 0.7$ (h) $\epsilon = 0.8$ (i) $\epsilon = 0.9$

Figure 6.5: The results of convex hull on the DST bi-objective environment which the ϵ varies from 0.1 to 0.9.

However, the learning process of Pareto Q-learning concerns with the learning rate (α), then the convex hull and the probability of goal states visiting on the DST bi-objective environment are examined when the (α) is decreased from 0.9 to 0.1. In addition, the discount factor (γ) is specified 0.1, the $\epsilon = 0.3$, and the time step of simulation is 2000.

Figure 6.6 shows the results of convex hull on the DST bi-objective environment which the α is decreased from 0.9 to 0.1. It is clearly seen that if the α is 0.8, the value of treasure will be found by taking a minimum time consuming. In contrasting with the α is 0.1, the submarine will take more time consuming to discover the valuable treasures. Hence, the α is fixed at 0.8 in order to provide optimal policies.



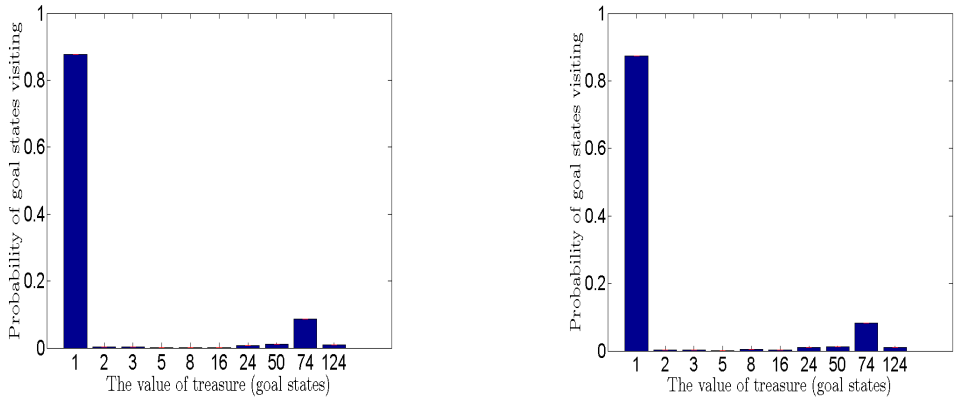
(a) $\alpha = 0.9$ (b) $\alpha = 0.8$

(c) $\alpha = 0.5$ (d) $\alpha = 0.1$

Figure 6.6: The results of convex hull on the DST bi-objective environment which the α is decreased from 0.9 to 0.1.

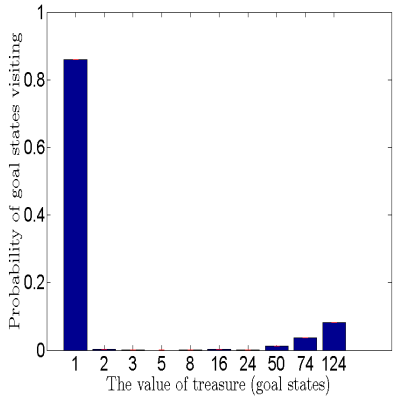
Figure 6.7 shows the results of probability of goal states visiting on the DST bi-objective environment which the α is decreased from 0.9 to 0.1. Although, the probability of the highest value of treasure visiting of the α 0.8 is less than the α 0.1, the submarine takes a minimum time consuming for the highest valuable treasure discovering. Hence, the α is fixed at 0.8 in order to provide optimal policies.

Moreover, the discount factor *gamma* is also increased from 0.1 to 0.9 in order to study how it has an affect on the DST bi-objective environment where the learning rate (α) is

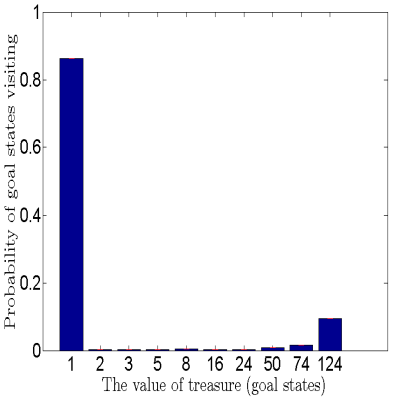


(a) $\alpha = 0.9$

(b) $\alpha = 0.8$



(c) $\alpha = 0.5$



(d) $\alpha = 0.1$

Figure 6.7: The results of probability of goal states visiting on the DST bi-objective environment which the α is decreased from 0.9 to 0.1.

specified 0.8, the $\epsilon = 0.3$, and the time step of simulation is 2000.

Figure 6.8 shows the results of convex hull on the DST bi-objective environment which the γ is increased from 0.1 to 0.9. It is clearly seen that if the γ is nearly 1, the submarine will discover the only lower valuable treasures with taking more time consuming.

Furthermore, Figure 6.9 shows the results of probability of goal states visiting on the DST bi-objective environment which the γ is increased from 0.1 to 0.9. Although, the probability of the highest value of treasure visiting of the γ 0.1 is less than the γ 0.5, the submarine takes a minimum time consuming for the highest valuable treasure discovering as shown in Figure 6.8. Hence, the γ is fixed at 0.1 in order to provide optimal policies.

In addition, the Pareto front on the DST bi-objective environment which the γ is fixed at 0.9 is an average found only 6 goal states as shown in Figure 6.10, and it should take time more than 2000 episodes to find optimal policies which should provide all 10 goal states.

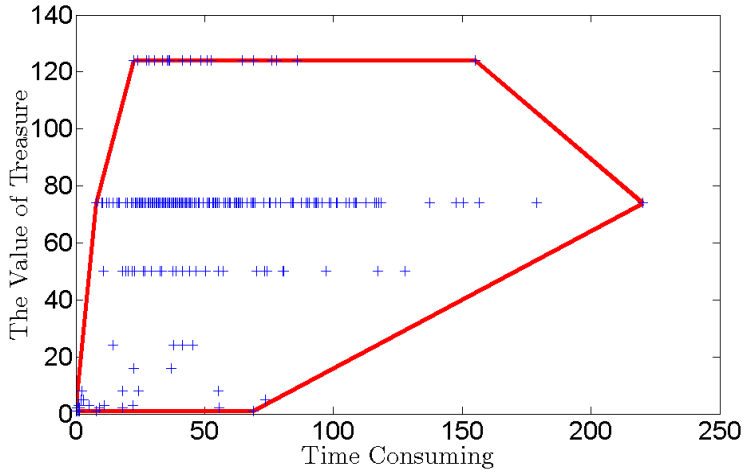
Hence, Figure 6.12 shows the Pareto front of the DST environment which consists of bi-objective; time consuming and the value of treasure which the Q-learning algorithm is embedded in the submarine in order to achieve all the 10 goal states. In addition, three parameters; the learning rate (α), the discount factor (γ), and the ϵ have been studied in order to study how they have an affect on the DST bi-objective environment which they should be specified 0.8, 0.1, and 0.3 respectively in order to discover all the value of treasure by taking minimum time consuming.

6.5 Conclusions

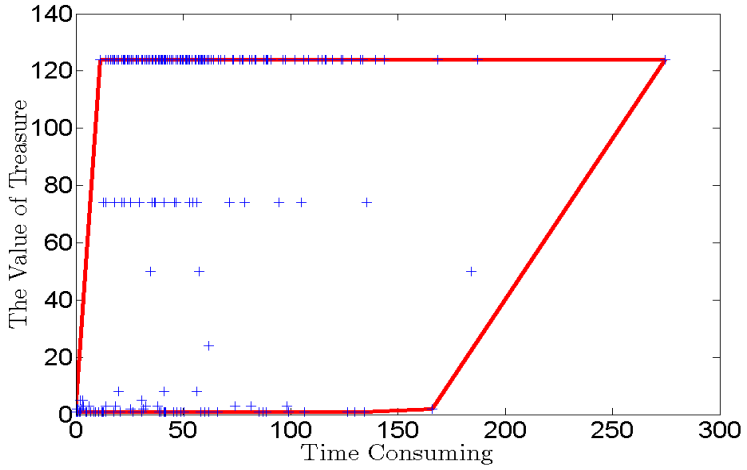
In this chapter, the experimental results are presented bi-objective optimization and reinforcement learning approaches for providing optimal policies which the first objective is to minimize time consuming for discovering the valuable treasure, and the second objective is to maximize the value of treasure. In addition, the DST bi-objective environment has the maximum 10 undersea valuable treasures which the value of treasure depends on its distance. It can be clearly seen that there are 10 policies to find the undersea treasures as showed in Figure 6.12, and the optimal path of each treasure is a part of the Pareto front. Moreover, number of episodes for running simulation have an effect on discovering Pareto front which the Pareto front can be found less than five at the beginning of episodes, in contrast with the nearly end of episode which the Pareto front can be found all of the goal states. In addition, three parameters; the learning rate (α), the discount factor, and the ϵ have a relationship among them which should be tuned before applied on the DST bi-objective environment because they have an affect on providing optimal policies.

Hence, the DST is one of multi-objective reinforcement simulations which has shown the optimal path of each treasure is an element of the Pareto front. In addition, the exploration and exploitation trade-off is crucial problem in a reinforcement learning which should have developed an evaluation mechanisms in order to select an efficient action and provide optimal policies.

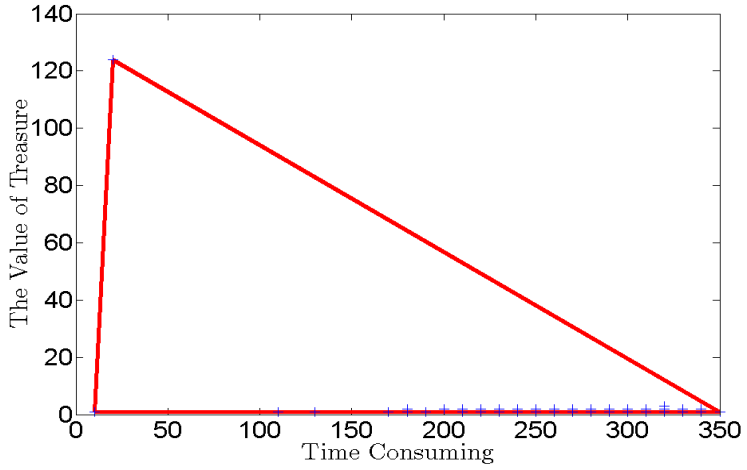
In addition, this chapter also provided an idea to apply the Pareto Q-learning for multi-objective routing optimization problems on communication network. For example, it can be applied to reduce congestion, and improve quality of service on Network-on-Chip which its topology is a grid network like the DST case study.



(a) $\gamma = 0.1$

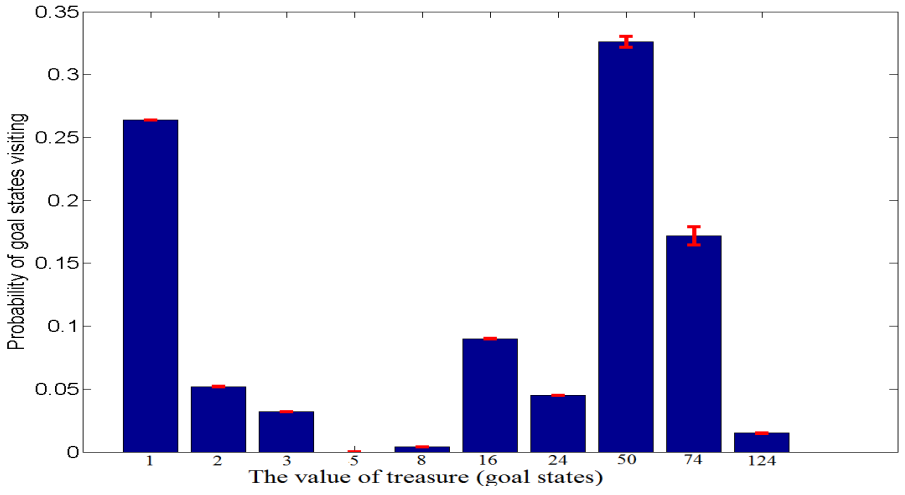


(b) $\gamma = 0.5$

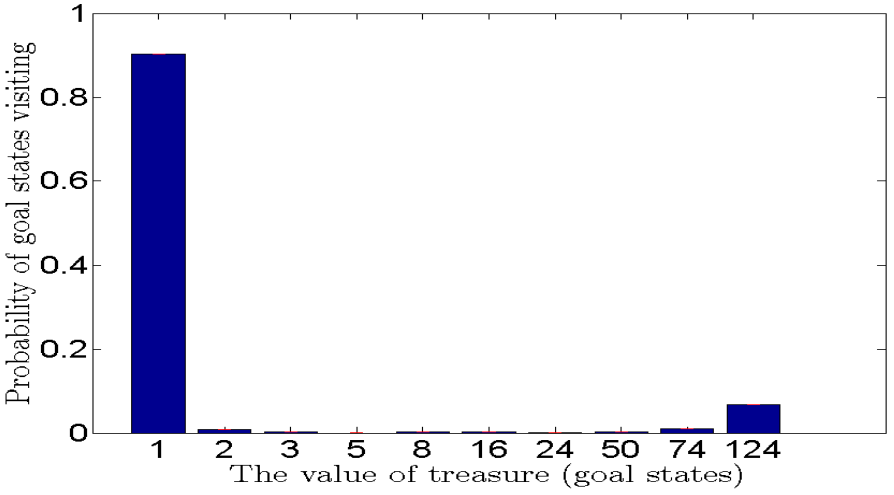


(c) $\gamma = 0.9$

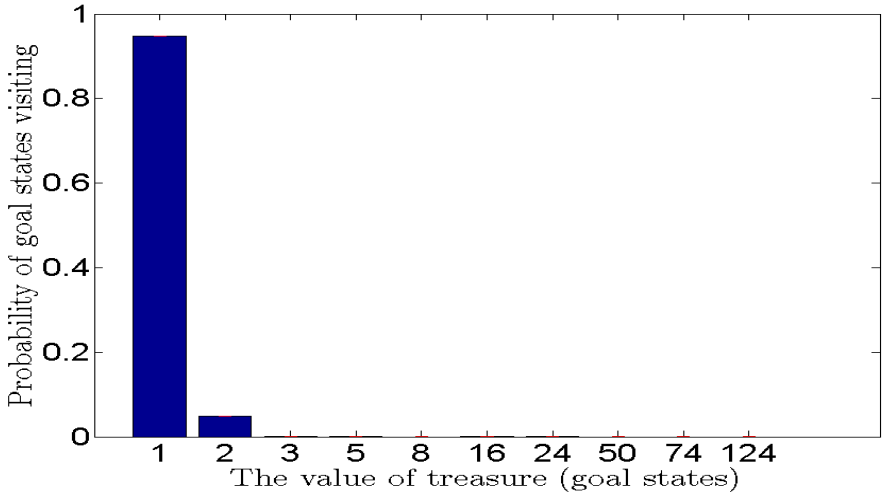
Figure 6.8: The results of convex hull on the DST bi-objective environment which the γ is increased from 0.1 to 0.9.



(a) $\gamma = 0.1$



(b) $\gamma = 0.5$



(c) $\gamma = 0.9$

Figure 6.9: The results of probability on goal states visiting on the DST bi-objective environment which the γ is increased from 0.1 to 0.9.

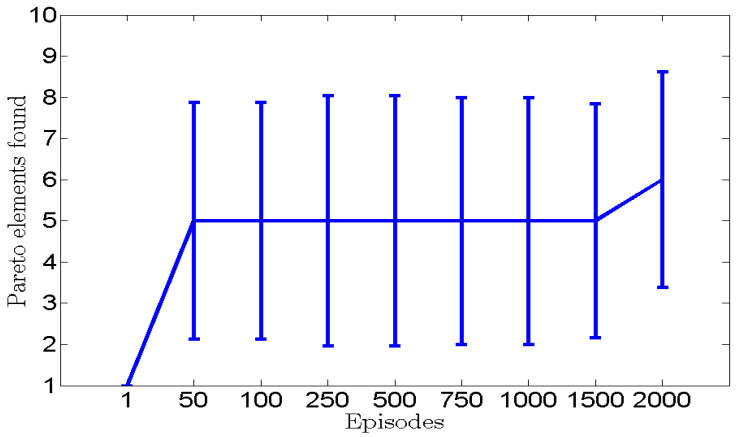


Figure 6.10: The Pareto front of the DST bi-objective environment which the learning rate (α) is 0.8, the discount factor (γ) is 0.9, and the ϵ is 0.3.

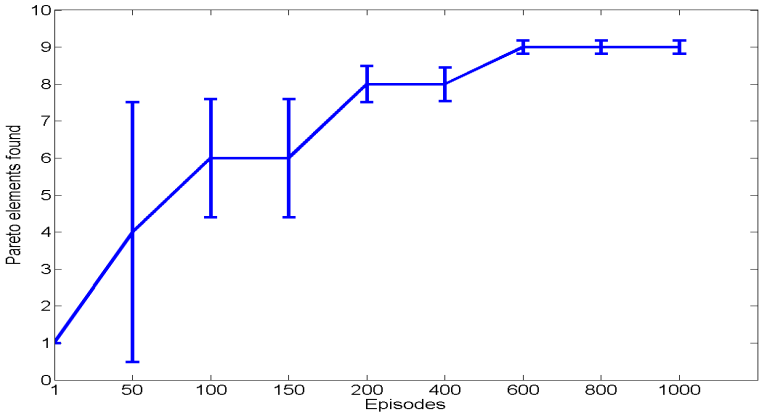


Figure 6.11: The Pareto front of all 10 goal states on the DST bi-objective environment which three parameters; the learning rate (α) is 0.8, the discount factor (γ) is 0.1, and the ϵ is 0.3.

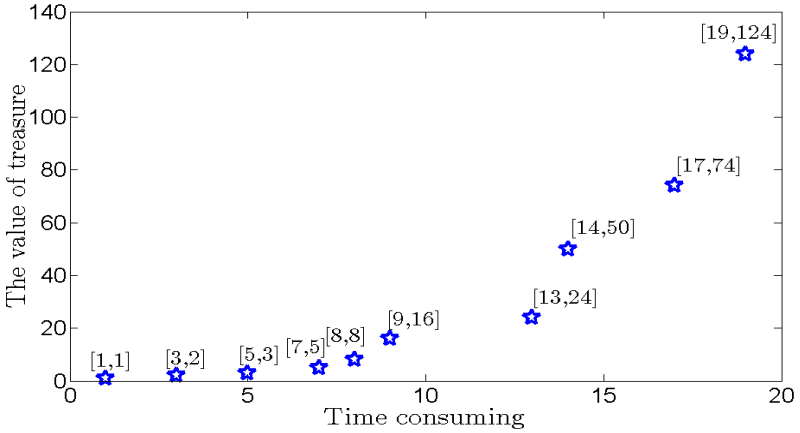


Figure 6.12: The Pareto front of DST bi-objective environment which consists of time consuming and the value of treasure of all 10 goal states.

Chapter 7

Conclusions and Future works

7.1 Conclusions

In dynamically changing communication networks, an efficient routing policy of packet routing should be adapted depending on various traffic conditions, traffic patterns, and changing in connectivity of the network. However, making globally optimal routing decision is not realistic because it would require a central controller which contains complete routing information in term of the state of all nodes and links on the entire network. For this reason, the routing decision has to build local routing information in individual nodes which should estimate packet delivery time to other nodes via its neighbors, or estimate queue lengths of intermediate nodes in the network. In addition, an adaptive routing algorithm should have an efficient mechanism to explore and update its routing tables in order to reflect the current routing information before forwarding packets in order to improve network performance in terms of decreasing packet delay time.

Hence, an adaptive routing strategy for communication networks based on the machine learning methodology of reinforcement learning have been explored in this thesis. Moreover, the Q-routing which is an application of reinforcement learning, was introduced over twenty years ago and it is also successful in packet transmission. However, all work citing it has been on small toy example networks which are not relative to real sizes of Internet networks. Hence, three synthesis topologies namely random topology, random topology with preferential attachment, and heuristically optimal topology have been shown that the Q-routing approach can scale up to realistic router level networks with 500 nodes and 5000 links between them. Furthermore, a real-world network architecture; the JANET is included in our studies. Moreover, statistical connectivity properties have been further explored. While the preferential attachment (PA) construct is seen as the popular model of several natural and man-made networks including the Internet, a recent suggestion of router level topology is the HOT topology. In comparing networks of

random preferential attachment and HOT network topologies with respect to adaptive routing, the experimental results demonstrate how a random network achieves the best improvement in reducing average delay at high loads 59.46% because it is easier to find alternated routes. The HOT topology, being a more realistic model of Internet routing is able to reduce average delay 40.78% which outperforms the PA architecture significantly. In addition, the PA architecture can reduce average delay time 37.93% because of its connectivity which tends to have a few centrally located and highly connected centers as a result of most traffic has to flow. Hence, suggesting adaptive routing is a strategy which may be deployed on real networks operating under heavy loads.

Furthermore, we are interested in exploring adaptive routing strategies for ad hoc mobile networks including routing in the context of the Internet of Things'. We are also interested in more efficient algorithms in the class of RL, such as the SARSA algorithm Rummery and Niranjan (1994) and performance optimization strategies with resource limitations (e.g. finite buffer sizes at nodes).

7.2 Future Works

In this thesis, there are three synthetic network topologies which Q-routing algorithm is applied for reducing average delay time. The Q-routing algorithm has been shown enhancement network performance by using the Q-values. Since, the Q-values store routing information which is a feedback signal in order to find routing policies while there are large number of packets increased in the network. However, there is only one objective function which is studied in this thesis that is to be minimize average delay time when packet is transmitted to its destination. Hence, we are interested in applied reinforcement learning for multi-objective problems in our future work. In addition, buffer size of router is one of problems which we are interested because it has an effect on network performance in terms of packet loss rate. Dhamdhere and Dovrolis (2006) showed that small buffers can lead to excessively high packet loss rate when the link carries many flows which has an effect on throughput of the network. Moreover, Wischik and McKeown (2005) and Lakshmikantha et al. (2011) claimed that buffers in routers play an important role for the Internet performance. By the way, we are interested the algorithm namely Pareto Q-learning which is suggested by Van Moffaert and Nowé (2014). The Pareto Q-learning is applied in multi-objective reinforcement learning by using sets of Pareto dominating policies that Van Moffaert and Nowé (2014) claimed that the Pareto Q-learning is able to learn the entire Pareto front under the assumption that each state and action pair is sufficiently sampled. In addition, the Pareto Q-learning is the first temporal difference-based multi-objective reinforcement learning (MORL) which does not use the linear scalar function so it is no limited to the convex hull (Van Moffaert and Nowé, 2014). However, the Pareto Q-learning has not employed on routing scheme and also network topology. Hence, it is more attractive to

apply Pareto Q-learning on routing scheme for solving multi-objective problems which are to be minimize average delay time, minimize buffer size in the router, and maximize throughput. The details of each ideas are described as follows.

7.2.1 Multi-objective reinforcement learning

Due to multiple objectives are suitable for network system design because multi-optimization can be solved at the same time, and it is more interesting, if we can find out how these objectives have an effect on each other. We have got an idea from (Van Moffaert and Nowé, 2014) which purposed Pareto Q-learning and showed that it is useful in an on-line setting, and it is able to find the optimal paths which are elements of the Pareto front. Hence, we can apply Pareto Q-learning in network routing scheme which the first objective is to minimize delay time between source and destination, and the latter objective is to maximize throughput in terms of successful transmitted packets. Moreover, the experiments on the deep sea treasure world (DST) problem in appendix B have shown that all Pareto optimal policies can be found by using Q-learning within a short learning period. However, there is another way to develop the Pareto Q-learning by improving exploration method. Hence, we are interested in action selection techniques in order to balance the exploration and the exploitation which we will suggest 3 techniques.

7.2.1.1 Hypervolume Set Evaluation

The first action selection technique is hypervolume which is used to measure and evaluate the Q-values set which is suggested by Van Moffaert and Nowé (2014). They claimed that the hypervolume is suitable for finding actions because it is able to be stricly monotonic with the Pareto dominance, and it provides a scalar measure of the quality of a set of vector. The hypervolume set evaluation starts by initial the list of each evaluated action, and then calculates the Q-values set of each action which its hypervolume will be added to the list. In addition, an action can be selected similarly to the single-objective case. For example, the greedy action can be selected if it relates to the Q-values set with contains the highest value of hypervolume, and it can be empty if the hypervolume of each action is 0, otherwise an action can be selected by random. Moreover, the hypervolume which is applied to the Pareto Q-learning, can be called $HV - PQL$.

7.2.1.2 Cardinality Set Evaluation

Van Moffaert and Nowé (2014) claimed that this evaluation mechanism closely relates to the cardinality indicator in multi-objective optimization which can be called C-PQL. The selected action process related to provide a degree of domination one action which

should have over other actions. Moreover, it is expected that these actions with larger probability should be cover Pareto dominating solutions.

7.2.1.3 Pareto Set Evaluation

The last selected action technique is Pareto which is a simplified version of the cardinality metric (Van Moffaert and Nowé, 2014). The selected action process is considerate a non-dominated vector of action across every other action which it is expected to remove any dominated actions, and then the non-dominated action can be selected by random. In addition, Van Moffaert and Nowé (2014) claimed that this technique is related to the Pareto.

7.2.2 Reinforcement Approach to Virus Propagation Models

In addition, it will be more interesting to consider virus propagation models (Kim et al., 2004; Yang et al., 2012) on complex communication networks like Internet which the Q-routing should be deployed for finding optimal policies in order to achieve a network performance. According to Yang et al. (2012) suggested a model of computer virus spreading under a reasonable assumption. In addition, it is more challenging to reduce infected computer virus by using an adaptive Q-routing to specify the area of virus infection in order to avoid those paths, and recovery of network performance. Furthermore, Zou et al. (2003) introduced a modeling E-mail based worms propagation which the Q-routing can be deployed on this model to be network monitoring in order to detect malicious nodes. Not only will it be successful in communication networks, but also the biological networks have benefited from the virus learning.

References

Aiyer, S. V., Niranjan, M., and Fallside, F. (1990). A theoretical investigation into the performance of the Hopfield model. *IEEE Transactions on Neural Networks*, 1(2):204–215.

Al-Rawi, H. A., Ng, M. A., and Yau, K.-L. A. (2015). Application of reinforcement learning to routing in distributed wireless networks: a review. *Artificial Intelligence Review*, 43(3):381–416.

Al-Rawi, H. A., Yau, K.-L. A., Mohamad, H., Ramli, N., and Hashim, W. (2014). Reinforcement learning for routing in cognitive radio ad hoc networks. *The Scientific World Journal*, 2014.

Albert, R. and Barabási, A.-L. (2002). Statistical mechanics of complex networks. *Reviews of modern physics*, 74(1):47.

Alderson, D., Li, L., Willinger, W., and Doyle, J. C. (2005). Understanding internet topology: principles, models, and validation. *IEEE/ACM Transactions on Networking (TON)*, 13(6):1205–1218.

Amaral, L. A. N., Scala, A., Barthelemy, M., and Stanley, H. E. (2000). Classes of small-world networks. *Proceedings of the National Academy of Sciences*, 97(21):11149–11152.

Atzori, L., Iera, A., and Morabito, G. (2010). The internet of things: A survey. *Computer networks*, 54(15):2787–2805.

Banks, J., CARSON II, J. S., Barry, L., et al. (2005). *Discrete-event system simulation-fourth edition*. Pearson.

Barabási, A.-L. and Albert, R. (1999). Emergence of scaling in random networks. *science*, 286(5439):509–512.

Barabási, A.-L., Albert, R., and Jeong, H. (2000). Scale-free characteristics of random networks: the topology of the world-wide web. *Physica A: Statistical Mechanics and its Applications*, 281(1):69–77.

Barabâsi, A.-L., Jeong, H., Néda, Z., Ravasz, E., Schubert, A., and Vicsek, T. (2002). Evolution of the social network of scientific collaborations. *Physica A: Statistical mechanics and its applications*, 311(3):590–614.

Batagelj, V. and Brandes, U. (2005). Efficient generation of large random networks. *Physical Review E*, 71(3):1–13.

Bhorkar, A. A., Naghshvar, M., Javidi, T., and Rao, B. D. (2012). Adaptive opportunistic routing for wireless ad hoc networks. *IEEE/ACM Transactions on Networking (TON)*, 20(1):243–256.

Boccaletti, S., Latora, V., Moreno, Y., Chavez, M., and Hwang, D.-U. (2006). Complex networks: Structure and dynamics. *Physics reports*, 424(4):175–308.

Boyan, J. A. and Littman, M. L. (1994). Packet routing in dynamically changing networks: A reinforcement learning approach. *Advances in neural information processing systems*, pages 671–671.

Brosch, T., Neumann, H., and Roelfsema, P. R. (2015). Reinforcement learning of linking and tracing contours in recurrent neural networks. *PLOS Computational Biology*, 11(10):1–36.

Calvert, K. L., Doar, M. B., and Zegura, E. W. (1997). Modeling internet topology. *IEEE Communications magazine*, 35(6):160–163.

Carlson, J. M. and Doyle, J. (1999). Highly optimized tolerance: A mechanism for power laws in designed systems. *Physical Review E*, 60(2):1412.

Castellano, C., Fortunato, S., and Loreto, V. (2009). Statistical physics of social dynamics. *Reviews of modern physics*, 81(2):591.

Çetinkaya, E. K., Broyles, D., Dandekar, A., Srinivasan, S., and Sterbenz, J. P. (2013). Modelling communication network challenges for future internet resilience, survivability, and disruption tolerance: A simulation-based approach. *Telecommunication Systems*, 52(2):751–766.

Chakrabarti, D. and Faloutsos, C. (2012a). *Graph mining : laws, tools, and case studies*. Synthesis lectures on data mining and knowledge discovery. Morgan & Claypool.

Chakrabarti, D. and Faloutsos, C. (2012b). Graph mining: laws, tools, and case studies. *Synthesis Lectures on Data Mining and Knowledge Discovery*, 7(1):1–207.

Chang, Y.-H., Ho, T., and Kaelbling, L. P. (2004). Mobilized ad-hoc networks: A reinforcement learning approach. pages 240–247.

Chen, Q., Chang, H., Govindan, R., and Jamin, S. (2002). The origin of power laws in internet topologies revisited. In *INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 2, pages 608–617. IEEE.

Cohen, R., Erez, K., Ben-Avraham, D., and Havlin, S. (2001). Breakdown of the internet under intentional attack. *Physical Review Letters*, 86(16):3682.

Costa, A. and Farber, M. (2015). Homological domination in large random simplicial complexes. *arXiv preprint arXiv:1503.03253*.

Crites, R. H. and Barto, A. G. (1996). Improving elevator performance using reinforcement learning. pages 1017–1023.

Dahui, W., Menghui, L., and Zengru, D. (2005). True reason for zipf's law in language. *Physica A: Statistical Mechanics and its Applications*, 358(2):545–550.

Dhamdhere, A. and Dovrolis, C. (2006). Open issues in router buffer sizing. *ACM SIGCOMM Computer Communication Review*, 36(1):87–92.

Di Felice, M., Chowdhury, K. R., Wu, C., Bononi, L., and Meleis, W. (2010). Learning-based spectrum selection in cognitive radio ad hoc networks. In *International Conference on Wired/Wireless Internet Communications*, pages 133–145. Springer.

Dijkstra, E. W. (1959). A note on two problems in connexion with graphs. *Numerische mathematik*, 1(1):269–271.

Doar, M. B. (1996). A better model for generating test networks. In *Global Telecommunications Conference, 1996. GLOBECOM'96.'Communications: The Key to Global Prosperity*, pages 86–93. IEEE.

Dong, S., Agrawal, P., and Sivalingam, K. (2007). Reinforcement learning based geographic routing protocol for uwb wireless sensor network. pages 652–656.

Dorogovtsev, S. N., Goltsev, A. V., and Mendes, J. F. (2008). Critical phenomena in complex networks. *Reviews of Modern Physics*, 80(4):1275.

Dorogovtsev, S. N. and Mendes, J. F. (2002). Evolution of networks. *Advances in physics*, 51(4):1079–1187.

Dowling, J., Curran, E., Cunningham, R., and Cahill, V. (2005). Using feedback in collaborative reinforcement learning to adaptively optimize manet routing. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 35(3):360–372.

Elwhishi, A., Ho, P.-H., Naik, K., and Shihada, B. (2010). Arbr: Adaptive reinforcement-based routing for dtn. pages 376–385.

Erdös, P. and Rényi, A. (1959). On random graphs i. *Publ. Math. Debrecen*, 6:290–297.

Fabrikant, A., Koutsoupias, E., and Papadimitriou, C. H. (2002). Heuristically optimized trade-offs: A new paradigm for power laws in the internet. In *International Colloquium on Automata, Languages, and Programming*, pages 110–122. Springer.

Faloutsos, M., Faloutsos, P., and Faloutsos, C. (1999a). On power-law relationships of the internet topology. In *ACM SIGCOMM computer communication review*, volume 29, pages 251–262. ACM.

Faloutsos, M., Faloutsos, P., and Faloutsos, C. (1999b). On power-law relationships of the internet topology. *SIGCOMM Comput. Commun. Rev.*, 29(4):251–262.

Filipowicz, B. and Kwiecień, J. (2008). Queueing systems and networks. models and applications. *Bulletin of the Polish Academy of Sciences. Technical Sciences*, 56(4).

Forster, A. and Murphy, A. L. (2007). Froms: Feedback routing for optimizing multiple sinks in wsn with reinforcement learning. pages 371–376.

Gábor, Z., Kalmár, Z., and Szepesvári, C. (1998). Multi-criteria reinforcement learning. In *ICML*, volume 98, pages 197–205.

Geisberger, R., Sanders, P., Schultes, D., and Vetter, C. (2012). Exact routing in large road networks using contraction hierarchies. *Transportation Science*, 46(3):388–404.

Gilbert, E. N. (1959). Random graphs. *The Annals of Mathematical Statistics*, 30(4):1141–1144.

Gläscher, J., Daw, N., Dayan, P., and O'Doherty, J. P. (2010). States versus rewards: dissociable neural prediction error signals underlying model-based and model-free reinforcement learning. *Neuron*, 66(4):585–595.

Gregori, E., Improta, A., Lenzini, L., and Orsini, C. (2011). The impact of ixps on the as-level topology structure of the internet. *Computer Communications*, 34(1):68–82.

Haraty, R. A. and Traboulsi, B. (2012). MANET with the Q-routing protocol. *ICN The Eleventh International Conference on Networks*, pages 187–192.

Hayes, J. (2013). *Modeling and analysis of computer communications networks*. Springer Science & Business Media.

Hopfield, J. (1984). Neurons with graded response have collective computational properties like those of two-state neurons. *Proceeding of the National Academy of Sciences of the United states of America*, 81:3088–3092.

Hu, T. and Fei, Y. (2010). Qelar: a machine-learning-based adaptive routing protocol for energy-efficient and lifetime-extended underwater sensor networks. *IEEE Transactions on Mobile Computing*, 9(6):796–809.

Jain, R. (2008). *The art of computer systems performance analysis*. John Wiley & Sons.

Kim, J., Radhakrishnan, S., and Dhall, S. K. (2004). Measurement and analysis of worm propagation on internet network topology. In *Computer Communications and Networks, 2004. ICCCN 2004. Proceedings. 13th International Conference on*, pages 495–500. IEEE.

Kleinrock, L. (1975). Queueing systems, volume i: theory.

Kumar, S. and Miikkulainen, R. (1997). Dual reinforcement q-routing: An on-line adaptive routing algorithm. In *Proceedings of the artificial neural networks in engineering Conference*, pages 231–238.

Kurose, J. F. and Ross, K. W. (2010). *Computer networking: a top-down approach*. Addison Wesley.

Kurose, J. F. and Ross, K. W. (2012). Computer networking: A top-down approach.

Lakshmikantha, A., Beck, C., and Srikant, R. (2011). Impact of file arrivals and departures on buffer sizing in core routers. *IEEE/ACM Transactions on Networking (TON)*, 19(2):347–358.

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature*, 521(7553):436–444.

Li, L., Alderson, D., Willinger, W., and Doyle, J. (2004). A first-principles approach to understanding the internet's router-level topology. *ACM SIGCOMM Computer Communication Review*, 34(4):3–14.

Liang, X., Balasingham, I., and Byun, S.-S. (2008). A multi-agent reinforcement learning based routing protocol for wireless sensor networks. pages 552–557.

Liben-Nowell, D. and Kleinberg, J. (2007). The link-prediction problem for social networks. *journal of the Association for Information Science and Technology*, 58(7):1019–1031.

Lin, Z. and van der Schaar, M. (2011). Autonomic and distributed joint routing and power control for delay-sensitive applications in multi-hop wireless networks. *IEEE Transactions on Wireless Communications*, 10(1):102–113.

Maleki, M., Hakami, V., and Dehghan, M. (2014). A reinforcement learning-based bi-objective routing algorithm for energy harvesting mobile ad-hoc networks. *IST The Seventh International Symposium on Telecommunications*, pages 1082–1087.

Medina, A., Lakhina, A., Matta, I., and Byers, J. (2001). Brite: An approach to universal topology generation. In *Modeling, Analysis and Simulation of Computer and Telecommunication Systems, 2001. Proceedings. Ninth International Symposium on*, pages 346–353. IEEE.

Medina, A., Matta, I., and Byers, J. (2000). On the origin of power laws in internet topologies. *ACM SIGCOMM computer communication review*, 30(2):18–28.

Murhammer, M. W., Lee, K.-K., Motallebi, P., Borghi, P., and Wozabal, K. (1999). *IP Network Design Guide*. IBM.

Newman, M. E. (2003). The structure and function of complex networks. *SIAM review*, 45(2):167–256.

Newman, M. E., Watts, D. J., and Strogatz, S. H. (2002). Random graph models of social networks. *Proceedings of the National Academy of Sciences*, 99(1):2566–2572.

Nie, L., Jiang, D., and Guo, L. (2013). A power laws-based reconstruction approach to end-to-end network traffic. *Journal of Network and Computer Applications*, 36(2):898–907.

Nurmi, P. (2007). Reinforcement learning for routing in ad hoc networks. pages 1–8.

Papoulis, A. and Pillai, S. U. (2002). *Probability, random variables, and stochastic processes*. Tata McGraw-Hill Education.

Pastor-Satorras, R. and Vespignani, A. (2001). Epidemic spreading in scale-free networks. *Physical review letters*, 86(14):3200.

Robertazzi, T. G. (2012). *Computer networks and systems: queueing theory and performance evaluation*. Springer Science & Business Media.

Rolla, V. G. and Curado, M. (2013). A reinforcement learning-based routing for delay tolerant networks. *Engineering Applications of Artificial Intelligence*, 26(10):2243–2250.

Rummery, G. A. and Niranjan, M. (1994). *On-line Q-learning using connectionist systems*. University of Cambridge, Department of Engineering.

Samejima, K. and Doya, K. (2007). Multiple representations of belief states and action values in corticobasal ganglia loops. *Annals of the New York Academy of Sciences*, 1104(1):213–228.

Santhi, G., Nachiappan, A., Ibrahime, M. Z., Raghunadhane, R., and Favas, M. (2011). Q-learning based adaptive qos routing protocol for manets. *Recent Trends in Information Technology (ICRTIT), 2011 International Conference on*, pages 1233–1238.

Smith, K. A. (1999). Neural networks for combinatorial optimization: a review of more than a decade of research. *INFORMS Journal on Computing*, 11(1):15–34.

Strehl, A. L., Li, L., Wiewiora, E., Langford, J., and Littman, M. L. (2006). Pac model-free reinforcement learning. In *Proceedings of the 23rd international conference on Machine learning*, pages 881–888. ACM.

Sutton, R. S. and Barto, A. G. (2011). *Reinforcement learning: An introduction*. Cambridge Univ Press.

Tanenbaum, A. S. and Wetherall, D. J. (2011). *Computer networks*. Pearson.

UCLA, I. (2012). Internet topology collection. *Avaliable: http://irl. cs. ucla. edu/topology*.

Vamplew, P., Dazeley, R., Berry, A., Issabekov, R., and Dekker, E. (2011). Empirical evaluation methods for multiobjective reinforcement learning algorithms. *Machine learning*, 84(1-2):51–80.

van der Ham, J., Ghijssen, M., Grosso, P., and de Laat, C. (2014). Trends in computer network modeling towards the future internet. *arXiv preprint arXiv:1402.3951*.

Van Moffaert, K. and Nowé, A. (2014). Multi-objective reinforcement learning using sets of pareto dominating policies. *Journal of Machine Learning Research*, 15(1):3483–3512.

Watkins, C. J. and Dayan, P. (1992). Q-learning. *Machine learning*, 8(3-4):279–292.

Waxman, B. M. (1988). Routing of multipoint connections. *Selected Areas in Communications, IEEE Journal on*, 6(9):1617–1622.

Wischik, D. and McKeown, N. (2005). Part i: Buffer sizes for core routers. *ACM SIGCOMM Computer Communication Review*, 35(3):75–78.

Xia, B., Wahab, M. H., Yang, Y., Fan, Z., and Sooriyabandara, M. (2009). Reinforcement learning based spectrum-aware routing in multi-hop cognitive radio networks. pages 1–5.

Yang, L.-X., Yang, X., Wen, L., and Liu, J. (2012). A novel computer virus propagation model and its dynamics. *International Journal of Computer Mathematics*, 89(17):2307–2314.

Yavuz, F., Zhao, J., Yağan, O., and Gligor, V. (2015). Toward-connectivity of the random graph induced by a pairwise key predistribution scheme with unreliable links. *IEEE Transactions on Information Theory*, 61(11):6251–6271.

Yook, S.-H., Jeong, H., and Barabási, A.-L. (2002). Modeling the internet's large-scale topology. *Proceedings of the National Academy of Sciences*, 99(21):13382–13386.

Zegura, E. W., Calvert, K. L., and Donahoo, M. J. (1997). A quantitative comparison of graph-based models for internet topology. *IEEE/ACM Transactions on Networking (TON)*, 5(6):770–783.

Zhang, X., Moore, C., and Newman, M. (2016). Random graph models for dynamic networks. *arXiv preprint arXiv:1607.07570*.

Zhang, Y. and Fromherz, M. (2006). Constrained flooding: a robust and efficient routing framework for wireless sensor networks. 1:6–pp.

Zou, C. C., Towsley, D., and Gong, W. (2003). Email virus propagation modeling and analysis. *Department of Electrical and Computer Engineering, Univ. Massachusetts, Amherst, Technical Report: TR-CSE-03-04*.