Cross-Layer Design for QoS Routing in Multi-Hop Wireless Networks

Ahed Alshanyour

A Thesis

In the Department

of

Electrical and Computer Engineering

Presented in Partial Fulfillment of the Requirements

For the Degree of

Doctor of Philosophy (Electrical and Computer Engineering) at

Concordia University

Montreal, Quebec, Canada

September 2011

CONCORDIA UNIVERSITY
SCHOOL OF GRADUATE STUDIES

This is to certify that the thesis prepared

By: Ahed Alshanyour

Entitled: Cross-Layer Design for QoS Routing in Multi-Hop Wireless Networks

and submitted in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY (Electrical & Computer Engineering)

complies with the regulations of the University and meets the accepted standards with respect to originality and quality.

Signed by the final examining committee:

Dr. B. Jaumard — Chair

Dr. S. Cherkaoui — External Examiner

Dr. L. Narayanan — External to Program

Dr. A. K. Elhakeem — Examiner

Dr. Y. R. Shayan — Examiner

Dr. A. Agarwal — Thesis Supervisor

Approved by

Chair of Department or Graduate Program Director

September 7, 2011

Dean of Faculty

ABSTRACT

Cross-Layer Design for QoS Routing in Multi-Hop Wireless Networks

Ahed Alshanyour, Ph.D.

Concordia University, 2011

Mobile Ad Hoc Networks (MANETs) are gaining increasing popularity in recent years because of their ease of deployment. They are distributed, dynamic, and self-configurable without infrastructure support. Routing in ad hoc networks is a challenging task because of the MANET dynamic nature. Hence, researchers were focused in designing best-effort distributed and dynamic routing protocols to ensure optimum network operations in an unpredictable wireless environment. Nowadays, there is an increased demand on multimedia applications (stringent delay and reliability requirements), which makes a shift from best-effort services to Quality of Services.

Actually, the challenge in wireless ad hoc networks is that neighbor nodes share the same channel and they take part in forwarding packets. Therefore, the total effective channel capacity is not only limited by the raw channel capacity but is also limited by the interactions and interferences among neighboring nodes. Thus, such factors should be taken in consideration in order to offer QoS routing. While, some of the distributed QoS route selection algorithms assume the availability of such information, others propose mechanisms to estimate them.

The goals of this thesis are: (i) to analyze the performance of IEEE 802.11 MAC mechanism in non-saturation conditions, (ii) to use the analysis in the context of multi-hop ad hoc networks, (iii) to derive theoretical limits for nodes performance in multi-hop ad hoc networks, (iv) to use the multi-hop analysis in QoS route selection.

We start the thesis by proposing a discrete-time 3D Markov chain model to analyze the saturation performance of the RTS/CTS access mode. This model integrates the backoff countdown process, retransmission retry limits, and transmission errors

into one model. The impact of system parameters (e.g., number of nodes, packet size, retry limits, and BERs) are analyzed. Next, we extend the 3D model to analyze the performance under non-saturation conditions and finite buffer capacity using two different approaches. First, we extend the 3D model into a 4D model to integrate the transmission buffer behavior. Second, we replace the 4D model by an M/G/1/K queueing system model with independent samples from the saturation analysis. The latter model gives similar results as the former but with a reduction in the analysis complexity. Next and by means of the non-saturation analysis, we proposed an approximate mathematical model for multi-hop ad hoc networks. Furthermore, we proposed an iterative mechanism to estimate the throughput in the presence of multiple flows. Finally, we used the multi-hop analysis to propose a QoS route selection algorithm. In this algorithm, we concentrate on the throughput as a QoS parameter. However, the proposed algorithm is valid to be used with other QoS parameters, such as packet delay, packet loss probability, and fairness. Analytical and simulation results show the deficiency of the current route selection algorithm in AODV and at the same time verifies the need for QoS route selection algorithms.

Acknowledgments

I have had the good fortune of having Dr. Anjali Agarwal as my thesis advisor. She has not only offered invaluable assistance and supported me financially but also given me the freedom to explore and to find my own research topic.

Deepest thank also to the members of the supervisory committee, Dr. Ahmed Elhakeem, Dr. Yousef Shayan, Dr. Lata Narayanan and my external examiner Dr Soumaya Cherkaoui without their knowledge and assistance this thesis would not have been successful.

I owe my deepest thanks to my father, brothers and sisters for deep love and encouragements they have given me. Also, I owe my deepest thanks to my deceased mother, she gone now but never forgotten. I will miss her always and love her forever. Thanks for all she did for me.

Next, I would like to thank all the people in my lab for making it a friendly and lively working place.

Lastly, I would like to express my deep gratitude to my wife Rula for her assistance in many ways for the successful completion of this thesis. During the many late nights and long weekends I spent working on this thesis; her patience was tried, but never failed. Most importantly, she has provided me with three beautiful children, Omar, Lina, and Zaid, who have brought more happiness into my life than I deserve.

To My Late Mother

Table of Contents

List of Tables

List of Figures

List of Abbreviations

3D	Three Dimensional
4D	Four Dimensional
ABR	Associative Based Routing
ACK	Acknowledgment
ADQR	Adaptive Dispersity QoS Routing
AHPC	Ad Hoc Broadcast Protocol
AMDR	Adaptive Mean Delay Routing
AODV	Ad-hoc On-Demand Vector
AQOR	Ad-Hoc QoS On-Demand Routing
BEB	Binary Exponential Backoff
BER	Bit Error Rate
BIP	Broadcast Increment Power
BLU	Broadcast Least-Unicast-cost
BPSK	Binary Phase Shift Keying
CCK	Complementary Code Keying
CEDAR	Core-Extracted Distributed Algorithm Routing
CI	Confidence Interval
CPThresh	CaPturing Threshold
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance
CSThresh	Carrier Sensing Threshold
CTS	Clear-To-Send
CW	Contention Window

DCF	Distributed Coordination Function
DIFS	DCF InterFrame Space
DLAR	Dynamic Load Aware Routing
DSR	Dynamic Source Routing
DSSS	Direct Spread Sequence Spectrum
EIFS	Extended InterFrame Space
FHSS	Frequency Hopping Spread Spectrum
FIFO	First-In First-Out
HMB	Highest Minimum Bandwidth
IEEE	Institute of Electrical and Electronic Engineering
IFS	InterFrame Space
IP	Internet Protocol
LAKER	Location-Aided Knowledge Extraction Routing
LAR	Location-Aided Routing
LBAR	Load-Balanced Ad-hoc Routing
LMST	Local Minimum Spanning Tree
LSR	Load Sensitive Routing
MAC	Medium Access Control
MANET	Mobile Ad-hoc NETwork
MPR	MultiPoint Relay
MST	Minimum Spanning Tree
NAV	Network-Allocation Vector
NP	Nondeterministic Polynomial
NS	Number of Slots
ns	Network Simulator
OLSR	Optimized Link State Routing
OSI	Open System Interconnection
PANDA	Positional-Attribute-based Next-hop Determination Approach
PHY	Physical layer

PLBQR	Predictive Location-Based QoS
QAM	Quadrature Amplitude Modulation
QoS	Quality-of-Service
QPSK	Quadrature Phase Shift Keying
RBOP	RNG Broadcast Oriented Protocol
RNG	Relative Neighborhood Graph
RPSF	Routing Protocol with Selective Forwarding
RRD	Random Rebroadcast Delay
RREP	Route REPly
RREQ	Route REQuest
RRS	RNG Relay Subset
RTCP	RNG Topology Control Protocol
RTS	Request-To-Send
RxThresh	Receiving Threshold
SBA	Scalable Broadcast Algorithm
SIFS	Short InterFrame Space
SINR	Signal to Interference and Noise Ratio
SLRC	Station Long Retry Count
SNR	Signal-to-Noise Ratio
SSA	Signal Stability Adaptive
SSRC	Station Short Retry Count
TBP	Ticket-Based Probing
TCP	Transmission Control Protocol
TDR	Trigger-based Distributed-QoS Routing
TTL	Time-To-Live
UDP	User Datagram Protocol
VoIP	Voice over IP
WLAN	Wireless Local Area Network

Chapter 1

Introduction

There has been a rapid increase in wireless local-area network (WLAN) deployment in recent years. WLANs offer convenience, low cost, expandable, and integral solution that helps setup a network quickly in situations where no network setup exists. Moreover, WLAN ad hoc mode allows wireless devices to communicate directly with no central access points involved. This mode of operation increases the popularity of WLANs especially when setting up a fixed infrastructure network is infeasible.

Mobile Ad Hoc Network (MANET) is a distributed, dynamic, and self-configurable network of mobile devices connected by wireless links. In MANET, each mobile device may function as both a host and a router. Routing in ad hoc networks is a challenging task because the network topology changes rapidly and unpredictably. The connectivity among the nodes varies with time due to mobility, high error rates, channel fading, congestion, and interference. Hence, routing failure problem is common in MANET networks. Usually, routing protocols resolve the route failure problem by selecting an alternative route, constructing a partial route, or rediscovering a new route to the destination. These solutions enlarge the scarce network resources consumption problem.

Extensive simulations [1, 2] using the ad-hoc on-demand distance vector (AODV) routing protocol show that an appropriate route-failure interpretation and control overhead minimization produce noticeable enhancement in the system performance.

In addition, the simulations show that most of the route failure problems are related to the quality of the discovered routes. The simple flooding mechanism used in reactive routing protocols does not provide quality of service (QoS) route selection. QoS route selection should take into consideration several metrics instead of a single metric. For example, selecting the shortest route does not yield a QoS route. Short route metric has many problems:

1. The short route is always the heavy loaded route especially when multiple TCP/UDP sessions run simultaneously.

2. Hidden node and exposed node problems reduce the possibility of parallel transmissions over the short route.

3. The short route always has common nodes with other routes, which produces the power depletion problem for these common nodes.

4. The short route implies using high transmission power. However, high transmission powers increases interference in adjacent channels or systems, which decreases the possibility of concurrent transmissions.

Also, routes that minimize total power consumptions or maximize network lifetime alone are not a good selection metric. For example, minimizing total power consumption implies:

1. Increasing the number of hops between the source and destination nodes in order to reduce the transmission power of each node. However, increasing the number of hops increases the frequency of route failure due to nodes mobility.

2. Using routes with minimum power consumption results in frequent use of certain links, which enlarges the power depletion problem.

On the other hand, maximizing the network lifetime metric alone is not a good choice because it could lead to more power consumption. Therefore, efficient broadcasting mechanisms are needed to select the QoS route between the source and destination nodes in order to avoid the problems that may occur during data transmission.

To discover QoS routes, a cross-layer route discovery framework is needed in which the source node automatically creates appropriate routing strategies as per application requirements, then intermediate nodes adapt the routing strategy according to the available resources. Further, during the route discovery phase, adapting a single routing metric and ignoring others is not an appropriate solution. All or most of the routing metrics should be considered in order to minimize or avoid as possible the induced routing problems.

To achieve this, efficient routing discovery strategies are needed. Hence, any proposed routing discovery mechanism must take into consideration more than one of the following parameters during the route selection phase: (a) the scarce network resources (bandwidth and power), (b) the channel information (bit error rate, signal strength, and channel utilization), (c) the TCP layer parameters (throughput and packet loss information), (d) the application requirement (bandwidth, delay, packet loss, and user priority), (e) the link states (link lifetime, link bandwidth, and link stability), (f) the location and position information (node coordinates, mobile speed, and neighborhood distribution), (g) the transmission power levels. Therefore, a cross-layer interaction is needed where each layer provides other layers with its own parameters and consequently a better decision can be taken.

1.1 Problem Statement

Nowadays, multimedia services play a central role for many social and entertainment applications. Provision of QoS guarantees by MANETs is a challenging task due to node mobility, multi-hop communication, unreliable wireless channel, lack of central coordination, and limited device resources [3]. Hence and for proper operation of multimedia services in MANETS, the QoS routing is essential instead of best-effort routing. Different QoS metrics can be considered to satisfy QoS requirements in route selection: e.g., minimum required throughout, maximum tolerable delay, maximum tolerable delay jitter, and maximum tolerable packet loss ratio [4]. In this thesis,

we focus on providing the QoS based on throughput because most of voice or video applications require some level of guaranteed throughput in addition to their other constraints.

In order to offer bandwidth-guaranteed routing, bandwidth information is needed. Some QoS routing protocols, e.g., core-extracted distributed algorithm routing (CEDAR) protocol [5], ticket-based QoS routing protocol[6] and trigger-based distributed-QoS routing (TDR) protocol [7] assume that the available bandwidth is known. Others propose techniques to estimate the available bandwidth, such as OLSR-based QoS routing protocol [8] and adaptive dispersity QoS routing (ADQR) protocol [9]. However, such available bandwidth estimation is imprecise because different factors affect bandwidth availability such as network size, transmission power, channel characteristics, and the interaction and interference among neighboring nodes. Therefore, we believe that QoS routing based on an accurate analysis of IEEE 802.11 MAC protocol will provide better bandwidth information than the estimation techniques.

1.1.1 Objectives

The main objective of this research is threefold:

1. To analyze the saturation and non-saturation performance of IEEE 802.11 MAC protocol in a single hop network.

2. To extend the non-saturation analysis for single hop network to multi-hop network.

3. To propose a cross-layer route discovery framework for QoS route selection for multi-hop ad hoc networks.

These objectives can be summarized as follows:

1. Study the performance of IEEE 802.11 MAC protocol analytically for single hop ad hoc networks using a three-dimensional Markov process model, which models the IEEE 802.11 DCF under saturated conditions. The analytical model will

integrate the backoff countdown process, transmission errors, and data/control retry limits into one model. To model transmission errors, Gaussian wireless error channel with constant bit errors will be used. In addition to collision probability, packet error rate will be used as transition probabilities in the model.

2. The model for saturation analysis will be extended to analyze the IEEE 802.11 MAC in non-saturation conditions. First, we will extend the model into a 4D model where the fourth state is used to model the transmission buffer and an additional state, namely the idle state, to be added to model the state in which the station resides when its transmission buffer is empty. Second, we will simplify the complexity of the 4D model by using the M/G/1/K queueing system with independent sample from the saturation analysis and show that both models give similar results.

3. The non-saturation model will be used for multi-hop analysis to analyze the end-to-end performance metrics of multi-hop wireless network.

4. The multi-hop analysis will be used to propose a cross-layer distributed route discovery mechanism for QoS route selection. In this distributed mechanism, the link performance metrics (throughput, delay, packet loss, and fairness) will be propagated through the entire network using the flooding mechanism. Using this information, intermediate nodes can classify the received route requests (RREQs) message from neighbors into good and bad candidates and then broadcast the best candidate. Moreover, the destination node uses the propagated information to select the QoS route.

5. Simulation using network simulator will be implemented to verify the correctness of our analytical analysis at each phase. For QoS route selection, we will implement our proposed cross-layer mechanism over the AODV routing protocol.

1.2 Contributions

1. We propose a discrete-time Markov chain model to analyze the saturation performance of IEEE 802.11 MAC. In this model:

   (a) We integrate the error recovery mechanism into our proposed model by extending the 2D Markov chain model to 3D. The first state models the backoff countdown process, the second state models the station short retry count, and the third state models the station long retry count.

   (b) We derive the main performance metrics: throughput, packet transmission delay, packet discard delay, packet discard probability, packet success probability, and packet mean service time.

   (c) We show that transmission errors as well as collisions impact the performance of IEEE 802.11 MAC and thus long and short retry limits are important metrics.

   (d) We show that our proposed model for RTS/CTS access mode is a general model and can be used to study the performance of basic access mode by setting values of certain parameters.

2. We extend the saturation analysis to a non-saturation analysis by extending the 3D Markov model into 4D model.

3. We show that we can reduce the complexity of non-saturation analysis by using an M/G/1/K queueing system with independent samples from saturation analysis instead of the 4D Markov chain model.

4. We show that the two models, namely the 4D model and the M/G/1/K model, give similar results.

5. In addition to the previous performance metrics, we derive the queueing delay, blocking probability, and packet loss probability.

6. We use the M/G/1/K analysis to propose an approximate analytical model for multi-hop wireless network.

7. We derive the end-to-end performance metrics of the multi-hop network.

8. We propose an iterative algorithm to find the non-saturated throughput of a multi-hop's links when it is surrounded by a random number of active nodes.

9. We show that stations with high flow rates can monopolize the channel and degrade the performance of the multi-hop path.

10. We propose a mechanism that utilize the multi-hop analysis in route selection process.

11. We show the detailed implementation of our proposed mechanism over AODV routing protocol.

12. We show how our proposed mechanism enhances the quality of the selected routes. In our implementation and simulation, we concentrate in maximizing throughput as a QoS metric.

1.3 System Model

The topology of a wireless ad hoc network can be represented by an undirected graph $G = (V, E)$. A graph G is a set of vertices (mobile nodes) v connected by edges (wireless links) e. An edge e exists between two nodes u and v if and only if for a certain acceptable bit error rate (BER), the SINR exceeds an appropriate threshold.

$$\frac{P_u G\left(L_u, L_v\right)}{\sum_{l \in N} P_l G\left(L_l, L_v\right) + N_v f} > \gamma_v \tag{1.1}$$

where P_u denotes the transmitted power of the transmitter node u, L_v denotes the location of the receiver v, $G\left(L_u, L_v\right)$ denotes the channel attenuation from node u to node v, N denotes the set of simultaneously active wireless links, N_v denotes the

power spectral density of the thermal noise at the receiver node v, f denotes the frequency bandwidth of the channel, and γ_v is an SINR threshold corresponding to an acceptable BER.

For the node v, its exact k-hop neighbor set, $H_k(v)$, is the set of nodes that are exactly k-hops away from v, $N_k(v) = \{v\} \bigcup H_1(v) \bigcup H_2(v) \bigcup \ldots \bigcup H_k(v)$, is the set of nodes that is at most k hops away from the node v. The k-hop neighborhood of a set of nodes is $N_k(A) = \bigcup_{v \in A} N_k(v)$. Every node $v \in V$ be assigned a unique identifier (id) and its degree is the number of nodes in $N_1(v)$. The density of the graph is the average degree for each node. The number of nodes in the network is $n = |V|$.

1.4 Assumptions

There are a variety of assumptions that can be made about the operation of MANET and the amount of topology, power, location, velocity, direction, distance, and area information that is needed for efficient algorithm implementations. Here is a simple summery of those assumptions:

1. **No control messages.** Each node reacts to the incoming broadcasting message without being aware of its neighbors. Blind flooding works under such assumption where each node rebroadcasts the first received RREQ message after some random rebroadcast delay (RRD) period and discards all subsequent duplicate RREQ packets.

2. **Availability of neighborhood information.** Many routing protocols use neighborhood information to minimize routing overhead. For example, by the help of the one-hop neighborhood information, nodes can decide whether to rebroadcast or discard the received RREQ message. On the other hand, the two-hop neighborhood information can be used to find the dominant forward node set[1]. While one-hop neighborhood information can be collected and maintained

[1]The minimum set of adjacent nodes whose broadcast cover the all 2-hop neighbors

by using periodic HELLO messages. the second-hop neighborhood topology can be built-up by including the list of one-hop neighbors in the broadcasted HELLO messages or using a TTL value of *2* when broadcasting HELLO messages.

1.5 Outline

We first present mobile ad hoc networks with some of problems that limit its scalability, cross-layer networking technology, and IEEE 802.11 MAC protocol in Chapter 2. Some of the proposed mechanism to enhance the route discovery mechanism are presented in Chapter 3. Those strategies are classified into two groups, namely better quality and lower overhead strategies.

In Chapter 4, we introduce a 3D Markovian framework that we use to model the backoff and error recovery in the idealized 802.11 DCF protocol. We provide a simple and accurate analysis using Markov chain modeling to compute the IEEE 802.11 DCF performance in a single hop network, in the presence of transmission errors and saturation conditions. This mathematical analysis calculates in addition to the throughput, the average packet delay, the packet drop probability and the average time to drop a packet for RTS/CTS, basic and hybrid access modes.

In Chapter 5, we use our Markovian framework to evaluate the performance of the protocol in non-saturation conditions. We first extend the 3D model into a 4D model to integrate the transmission buffer stochastic process. In the 4D model, we disallow packet arrivals between transmissions and instead introduce arrivals with a probability depending on the packet mean service time. Second, we notice that arrivals and service time in our model are independents, and thus the complexity of the 4D model can be reduced by using an M/G/1/K queue with independent samples from the saturation model (the service time). Our analysis show that both models give similar results.

In Chapter 6, we use the M/G/1/K analysis to provide an approximate mathematical model to analyze the end-to-end performance of IEEE 802.11 DCF in a

multi-hop network. The interference and carrier sensing ranges models are used to divide the single chain multi-hop network into a congregation of interleaved single hop sub-networks. Furthermore, we provide an iterative algorithm to estimate the throughput in presence of multiple flows.

In Chapter 7, we use the multi-hop analysis to propose a QoS route selection algorithm. This algorithm can select the QoS route subject to the throughput, delay, packet loss ratio, or fairness. We show through simulation and analytical analysis the deficiency of the route discovery mechanism in the AODV routing protocol. Moreover, we implement our algorithm over AODV and compare its performance to the performance of the current AODV.

Finally, we discuss and present possible extensions and conclude our work in Chapter 8.

Chapter 2

Background

2.1 Wireless Networks

There is a rapid increase in wireless local-area network (WLAN) deployment in the recent years. WLANs offer convenience, low cost, expandable, and integral solution that helps to setup a network fast in situations where there is no existing network setup. Moreover, WLAN ad hoc mode allows wireless devices within range of each other to communicate directly without involving central access points. This mode of operation increases the popularity of WLANs especially when setting up a fixed infrastructure network is considered infeasible.

Mobile Ad Hoc Network (MANET) is a distributed, dynamic, and self-configurable network of mobile devices connected by a wireless links. In MANET, each mobile device may function as both a host and a router. Routing in ad hoc networks is a challenging task because the network topology is not fixed and nodes cannot detect the activity of all other nodes. Moreover, the connectivity between terminals may vary with time due to the mobility, interference, fading, high bit error rate, and congestion. Hence, dynamic routing protocols are necessary for such networks to function properly. In MANET, all mobile nodes must cooperate in order to configure temporary network topologies to forward traffic between non-neighbor nodes.

2.2 Problems in Multi-hop Wireless Networks

Multi-hop networks encounter several problems that limit its scalability. In this section, we will present the following three problems: fairness problem, hidden node problem, and exposed node problem.

2.2.1 Fairness Problem

In multi-hop networks and within a given area, a channel can be reused, therefore the CSMA-CA protocol does not function well because nodes cannot coordinate their channel accesses and it is up to each node to decide when and how long it will access the channel. This can lead to a situation where a certain station monopolizes the channel for a long time while other stations starve. This situation is referred to as the fairness problem. For example consider, a wireless network of four nodes A, B, C, and D, see Figure 2.1. Consider that node C wants to send data to node D, therefore node C sends an RTS packet to node D to reserve the channel. If the channel is idle, node D responds with a CTS packet. Node B which is out of node's D transmission range but within node's C transmission range receives the RTS packet and enters the virtual carrier sensing mode. Now, assume station A wants to send data to station B. While station A cannot detect the RTS or CTS packet from node D or node C, it assumes that the channel is idle. Thus, it sends an RTS packet to node B. But node B will not respond with a CTS packet because it knows that the channel is busy. As a result, node A will backoff and double it contention window. When transmission between node C and D is over, nodes B, C, and D will have small contention windows while node A have a large one. Due to the contention windows sizes, nodes B, C, and D have a higher chance to capture the channel compared to node A. Which means that throughput of nodes B, C, and D will be higher than node A. Such scenario is simulated by [10] under infinite traffic load. Table 2.1 shows the results of their simulation. From this table, we see that the first link has only a fourth of the throughput of the other two links.

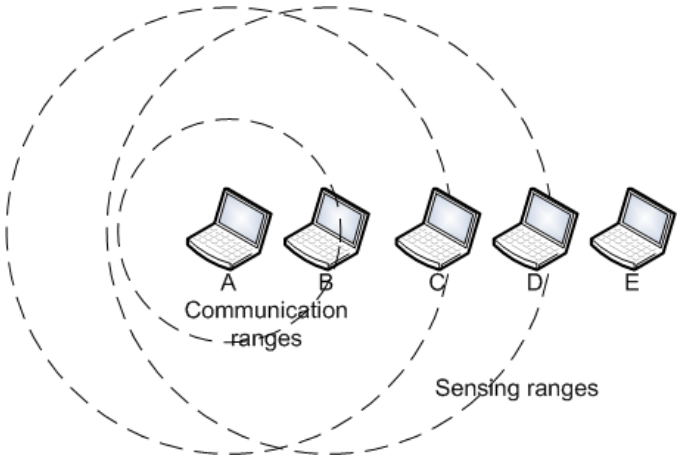

Figure 2.1: Redundant broadcast

Table 2.1: Throughput (Mbps) at infinite load

link A-B	link B-C	link C-D
0.1568	0.6733	0.6878

2.2.2 Hidden Node Problem

Hidden node problem occurs because sender could not hear as far as the receiver. In Figure 2.1, if there is a transmission between node D and node E, node A cannot hear this transmission because this transmission is out of its carrier sensing area. To node A, the channel is idle. Therefore, if node A sends an RTS packet to node B, node B will not respond with a CTS packet because it knows that the channel is busy.

2.2.3 Exposed Node Problem

The exposed node problem occurs when a node (the exposed node) is blocked by hearing an RTS packet and thus it cannot initiate a transmission, although its transmission will not interfere with ongoing transmission. In Figure 2.1, consider that there a transmission from node B to node A and node D wants to transmit to node

E. Although transmission from node D to node E will not interfere ongoing transmission from node B to node A, node D cannot initiate such transmission because it is blocked by node's B transmission.

2.3 Cross-Layer Networking Technology for Wireless Communications

The Open Systems Interconnection (OSI) reference model divides the network protocol into seven independent layers, which are designed separately. Each layer defines the hierarchy of services to be provided for the layers directly located above and below it. Various protocols are designed in different layers to realize these services. The main goal of the OSI model was to allow heterogeneous computer systems to interact and communicate by transmitting pure data traffic. This type of rigid design demonstrates the scalability and effectiveness of the layering principle and it achieves a great success in wired networks. However, in wireless networks, wireless links create several problems for protocol design that cannot be handled well in the strict layering architecture. Some of these problems include:

- **Wireless networks are interference limited.** The network throughput depends mainly on the level of interference. In the IEEE 802.11 [11] medium access control (MAC) protocol, only one station is allowed to transmit within a circular area whose radius is about twice the node's communication range, the carrier sensing range. Therefore, efficient routing protocols are needed to minimize the level of interference through using adaptive transmission powers or avoiding interfered routes.

- **Congestion in wireless networks has different notion**. In wireless networks, congestion may result from a high level of interference or an excessive traffic load at a node/link's buffers. Therefore, efficient mechanisms are needed to recognize the reason of congestion and then to resolve it.

- **Wireless networks operate in a broadcast medium**. Packet flooding or broadcasting is essential function for establishing a communication path from a source node to a destination node. However, broadcasting operation consumes power and bandwidth resources and enlarges the packet collision and contention problems, which reduces the success rate of packet transmissions and consumes energy. Therefore, there is a need for efficient protocols that minimize routing overheads and power consumptions.

- **Wireless links are not stable**. Link instability results from mobile movement, high levels of interference, high error rate, channel fading, packet collision, and contention problems. Thus, route maintenance mechanisms, multi-path solutions, robust and reliable routes, and efficient scheduling algorithms are needed to minimize the instability problem.

- **Wireless networks are power capacity limited**. Depletion of some nodes' power may cause a network partitioning. Therefore, there is a need to maximize the network lifetime by proposing mechanisms for minimum power consumption, balanced load distribution, adaptive power control, adaptive coding, and adaptive error control.

Further, these days, both wired and wireless networks need to support new applications, namely real-time multimedia applications (multimedia video conferences and voice over IP (VoIP)). In such type of applications, user demand is shifted from high data rates to more complex requirements in terms of Quality of Service (QoS) and energy efficiency.

These aspects and others make the strict layer design approach not suitable and does not function efficiently in the current communication networks especially wireless networks. This motivates the network designers to abuse the layered architecture and to introduce a layerless structure [12] or a cross-layer approach [13, 14, 15] as an alternative solution.

The layerless structure suggests merging all layers into a single layer and then optimizing all stack parameters for efficient performance. In layerless communication system, there is a single protocol instead of several intercommunicated protocols, which will offer a great flexibility in the protocol design, as different layers will exist together. Further, the communication protocol can be modeled as a single mathematical model, which simplifies the process of optimizing the protocol operations in order to provide a reliable and a high-quality end-to-end performance in multimedia communications. Unfortunately, layerless approach is not operable and not compatible with the existing layered networks, which makes it difficult to adapt such approach in the design of the network communication protocols.

Some cross-layer solutions have been proposed in recent years to improve the performance of networks operating in an error-prone wireless environment. This approach suggests using signaling between layers such that the information between layers is shared to optimize the performance of the network protocol. Thus, only limited modification is required to the existing stack. For example, the existence of a link layer's mechanism, which discovers the cause of route failure and then signals the appropriate stack's layer to take an appropriate action, will enhance the overall system performance (mobility-related route failure can be handled by the routing layer by selecting an alternative route or initiating a new route discovery cycle, traffic-related route failure can be handled by the transport layer through reducing the transmission rate, and interference-related route failure can be handled by the link layer through decreasing the transmission power level).

2.4 Routing Protocols

Routing protocols in ad hoc networks can be categorized into two classes, proactive or table-driven and reactive or on-demand routing protocols. In proactive routing protocols [16], each node collaborate to maintain consistent, up-to-date routing information to every other node in the network even before it is needed. Routing

information is kept in routing tables and updated if there is any topology change in order to keep the network in a consistent view. In large networks, proactive protocols are not a suitable routing choice because they consume network bandwidth due to the excessive routing overhead traffics, which are needed to maintain the route entries in the routing tables. On the other hand, reactive routing protocols [17, 18] utilize the network bandwidth by creating the needed routes on-demand. If a node needs to communicate with another node, the routing protocol searches for the route in an on-demand manner. The operations of reactive routing protocols are divided into three stages, route discovery, packet delivery, and route maintenance. On-demand routing protocols are distinguished by the different strategies that are used in the route discovery and the route maintenance stages.

On-demand routing protocols, such as AODV [18] and DSR [17], often use flooding technique to search for new routes. Consider a source node S, which has data packets to send to a destination node D. If the node S does not have in its routing table a valid route to the node D, it initiates a route discovery process by broadcasting a route request (RREQ) packet, which is flooded throughout the entire network. Each node, upon receiving the RREQ packet, rebroadcasts the RREQ packet to its neighbors if it has not forwarded it before, provided that the node is not the destination node and the RREQ packet's time-to-live (TTL) counter has not been exceeded. Each RREQ carries a unique sequence number, namely a broadcast ID, which is generated by the source node. A node, upon receiving the RREQ packet, checks the sequence number of the packet before forwarding it. The packet is forwarded only if it is not a duplicate RREQ. The node D, after receiving the first RREQ packet, replies to the source node through the reverse path the RREQ packet had traversed. This type of flooding is called pure flooding or blind flooding because all nodes except the destination forward once the RREQ packet and ignore the duplicate copies of the same RREQ packet during the route construction phase.

Blind flooding guarantees that broadcast will cover the entire network if the network is not partitioned and no packet loss is observed. But flooding consumes scarce

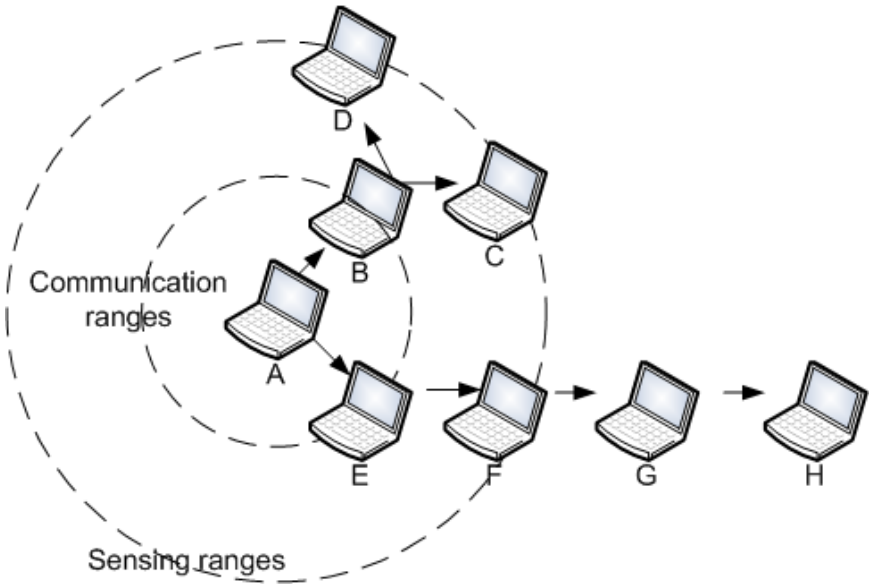

Figure 2.2: Redundant broadcast

network resources, power and bandwidth, due to the unnecessary routing overheads. For a network with n nodes, flooding requires $n-1$ RREQs transmissions because each node except the destination node retransmits the received RREQ packet. However, in most cases, route discovery does not require this number of RREQ transmissions especially when the physical distance between the source and the destination nodes is short. In Figure 2.2, although the physical distance between the source node A and the destination node D is two hops, the RREQ packet is broadcasted seven times as a result of flooding ($A \rightarrow (B, E)$, $B \rightarrow (C, D)$, $E \rightarrow F$, $F \rightarrow G$, and $G \rightarrow H$. Indeed, minimizing the redundant transmissions decreases the total power consumptions. Hence, efficient routing discovery protocols are needed to maintain the network resources.

Different cross-layer optimization techniques are presented in literature to enhance the flooding process as well as enhancing the quality of the discovered routes. Some of these optimization techniques are:

- Discover multiple paths instead of a single path to the destination node [19], sequence numbers are generated by the source node and the path it has tra-

versed. Based on a predefined cost function, the destination replies on all or some of the received RREQ packets.

- Minimize flooding overheads, different approaches can be used to minimize routing overheads. For example, route caches are used during the route construction phase. An intermediate node, which have a route to the requested destination node in its cache, replies to the source node instead of rebroadcasting the received RREQ packet to its neighbors. Another example, the geographical position information is included in the RREQ messages to limit the searching area to a smaller zone instead of flooding the route requests into the entire network.

- Avoid network partitioning and route failure problems, nodes with low residual energies should not be selected as routers.

- The reliability and availability of the discovered routes can be enhanced by utilizing interference and congestion information from the link layer.

- Nodes can also learn about the neighboring routes traversed by data packets if they operated in the promiscuous mode (the mode of operation in which a node can receive the packets that are neither broadcasted nor addressed to itself)

In the next chapter, we review some of the cross-layer techniques, which are used to enhance the quality of the discovered routes by utilizing the feedback information from other layers.

2.5 IEEE 802.11 MAC Protocol

IEEE 802.11 medium access control (MAC) [11] protocol is currently the most popular random access MAC layer protocol used in wireless ad hoc networks. It is used for coordination and scheduling of transmissions among competing stations in order to minimize collisions. It defines two medium access methods, the compulsory

distributed coordination function (DCF) and the optional point coordination function (PCF). DCF is based on a carrier sense multiple access with collision avoidance (CSMA/CA) technique. CSMA/CA uses a binary exponential backoff (BEB) mechanism as a contention resolution technique. BEB resolves the contention problem by randomizing moments at which stations can access the channel. DCF defines two modes to access the channel, the basic access mode and the request-to-send/clear-to-send (RTS/CTS) mode. RTS/CTS is an optional scheme, which uses small RTS/CTS control packets to reserve the medium before large packets are transmitted in order to reduce the duration of a collision.

In IEEE 802.11 DCF, priority levels for accessing the channel are provided through the use of interframe spaces (IFs) such as, short interframe space (SIFS), DCF interframe space (DIFS) and extended interframe space (EIFS). A station with new packet to transmit monitors the channel, if the channel is sensed to be idle for an interval larger than DIFS period, the station transmits the packet. Otherwise, if the channel is sensed busy (either immediately or during the DIFS), the station defers its transmission and keeps monitoring the channel until it becomes idle for a DIFS period. Then, the station generates a random backoff period before transmitting the packet. To avoid channel capture problem, the random backoff period is selected between successive transmissions. The contention window (CW) value depends on the number of retransmissions. It starts with a minimum value (CW_{min}) and doubles after each unsuccessful transmission up to a maximum value $CW_{max} = 2^m CW_{min}$, (m is a positive-integer number, which limits the value of CW). The backoff time is randomly and uniformly chosen from the range $(0, CW - 1)$ time slots. It is a slotted time, the duration of each time slot (σ) is carefully set equal to the time needed by any station to detect the transmission of other stations within a certain range. The backoff time is decremented once every time slot for which the channel is detected idle, frozen when a transmission is detected on the channel, and resumed when the channel is sensed idle again for a DIFS period. The station transmits when the backoff time reaches zero. Time duration between successive empty time slots is

variable and depends on the status of the medium. Two successive empty time slots should be proceeded by an idle DIFS period. DCF sets a threshold for the number of retransmissions, as the number of retransmission exceeds this threshold, the frame is dropped from the MAC queue. More, as CW reaches it is maximum value, it keeps on this value in subsequent retransmission attempts.

In the basic access mode, as the backoff time equals zero, the source node transmits a data frame and waits for a timeout period in order to receive an acknowledgment packet (ACK) from a destination node. The destination node waits for a SIFS period immediately following the successful reception of the data frame and replies with the ACK to indicate that the data packet has been received correctly. While the data frame is being transmitted, other nodes hearing the data frame transmission adjust their network-allocation vector (NAV), which is used for virtual carrier sense at the MAC layer, correctly based on the duration field value in the data frame received. This includes the SIFS and the ACK frame transmission time, which are following the data frame.

In the RTS/CTS access mode, two small control packets, RTS and CTS, are handshaked between a source and a destination nodes prior to the transmission of an actual data frame in order to capture the channel, to prevent other nodes from transmission and to shorten the collision time interval. A node that needs to transmit a packet follows the rules of backoff mechanism. As the backoff counter reaches zero, the source node sends an RTS frame. As the destination receives the RTS frame, it responds with a CTS frame after a SIFS period. The source node is allowed to transmit its data frame if and only if it received the CTS frame correctly. Successful data transmission is acknowledged by the destination node. RTS and CTS used by other stations to update their NAVs using duration fields information. If a collision occurs with two or more RTS frames, less bandwidth is wasted as compared to the situation when larger data frames are collided.

In RTS/CTS, collisions and bit transmission errors cause unreliable transmissions. The behavior of a sender station when a receiver station received a corrupted packet

is similar to the case when a collision is occurred. In both situations, the sender will not be acknowledged by the receiver. Packet corruption may occur for the RTS, CTS, data, and ACK. Therefore, the station behavior when it received a corrupted packet is similar to the case if it did not receive that packet.

IEEE 802.11 achieves reliability through retransmissions. It associates a retry counter with each MAC packet, a station short retry counter (*ssrc*) for a packet whose length is less than or equal to the *RtsThreshold* and a station long retry counter (*slrc*) for the packet longer than *RtsThreshold*. The two retry counters have two predefined limits, *ShortRetryLimit* (R_1) for *ssrc* and *LongRetryLimit* (R_2) for *slrc*. The associated retry counter is incremented after each failed retransmission attempt until it reaches its limit. Moreover, the associated retry counter should be reset to *0* when the packet transmission is succeeded or dropped. Specifically, after sending a packet, if an acknowledgment is not received in a timely fashion, the sender increments the associated retry counter and retransmits the packet. In RTS/CTS, two retry counters are used: *ssrc* for RTS packet and *slrc* for data packet. When the RTS or data packet transmission is failed, the corresponding retry counter is incremented up to a predefined limit. The station shall discard the data packet when either of the two retry counters reaches its limit first.

Although RTS/CTS increases the overheads associated with transmitting the extra physical (PHY) headers for the RTS/CTS packets but it can be used to improve the performance of IEEE 802.11 wireless networks. Specifically, the RTS/CTS is widely deployed in wireless networks in order to reduce collision time and, thus, achieve high throughput [20]. Also, in a highly congested network, numbers of collisions and retransmissions increase. Hence, stations spend more energy on retransmission and sensing the wireless channel. Thus, RTS/CTS mode results in energy depletion less than that in basic mode because when collisions occurs, only short RTS packets are collided instead of lengthy data packets [21]. Moreover, in rate-adaptive MAC algorithms, RTS/CTS control packets are used to measure the channel condi-

tion at the receiving side to solve the blind probing[1] problem [22]. In addition to that, RTS/CTS is helpful to reduce the number of retransmissions if hidden node problem persists in multihop ad hoc networks [23].

In both access modes, if the ACK frame is received correctly, the transmitting node resets its CW to CW_{min} and reenters the backoff process if it has further frames in its MAC queue. If the source node does not receive the ACK, the data frame is assumed to be lost and the source node doubles its CW and reschedules the frame retransmission according to the backoff rules. The transmitted data packet is dropped from the MAC queue after specific number of retransmission attempts.

The MAC is an essential block in wireless ad hoc network. It directly affects the utilization of the channel capacity and the system performance. Indeed, there are main issues that should be considered when designing the MAC layer protocol such as, utilizing the scarce network resources (bandwidth and power), avoiding hidden and exposed terminal problems (these problems occur due to the simultaneous transmissions of some nodes that are not within the transmission range of the sender, but are within the transmission range of the receiver), minimizing collision between stations by scheduling the channel access among the competing nodes, minimizing routing overheads that are needed to coordinate between the competing nodes, and finally, minimizing the impact of mobility on the network performance.

Hence, the accurate feedback information from the MAC protocol to the routing layer could help the latter to select optimal routes. In this work, we will propose an analytical model for the IEEE 802.11 multi-hop network, which considers the network, channel, and node state information. The model will be analyzed to derive the IEEE 802.11 performance metric. These performance metrics will be used as feedback information for the routing layer during the route discovery phase in order to enhance the quality of the selected routes.

[1]The sender keeps trying sending a DATA packet at a higher data rate from time to time, even though the receiver cannot actually handle a faster transmission

2.6 Summary

Ad hoc wireless networks have several characteristics that distinguish it from wired network. The strict layer design approach is not suitable and does not function efficiently in wireless networks. As a solution, there has been currently a proliferation in the use of cross layer design. There are two approaches for cross layer design, either merging all stack layers in a single layer and optimize all stack parameters for efficient performance or using signaling between layers and thus only limited modification would be required to the layered stack. In this chapter, we discussed the routing induced problems in multihop ad hoc networks. Furthermore, IEEE 802.11 MAC protocol is discussed in details. In this thesis, a cross layer design approach mainly between the MAC and the routing layer will be investigated. A general model for the IEEE 802.11 MAC layer will be proposed and analyzed to calculate the IEEE 802.11 performance in multihop ad hoc network. The calculated performance metrics will be fed backed to the routing layer to use them during the route selection phase.

Chapter 3

Literature Review of Routing Discovery Strategies in MANET

Although blind flooding is a simple mechanism, it overwhelms the network with constant broadcast traffic, which results in high contention and collision in the network. Such phenomenon is called a broadcast storm problem [24]. However, rebroadcasting can provide only 0-61% additional coverage over that already covered by the previous broadcasting [24]. Contention occurs when two or more hosts around a transmitter are likely to be close enough and contend with each other on the wireless medium. Analytical analyses where two hosts are around a transmitter host show that the probability of contention is around 59% [24]. This probability is expected to be higher as the number of surrounding hosts increases. Finally, collision occurs due to an inefficient or an absence of collision prevention and avoidance mechanisms, which results in simultaneous rebroadcast by two or more hosts.

In [24], a random rebroadcast delay (RRD) is proposed to prevent the broadcast storm problem. In RRD, the received RREQ message is delayed before it is rebroadcasted. The delay time is uniformly distributed between 0 and 10 milliseconds. Although RRD reduces the contention and the congestion problems, it is not a proper solution because it causes the next-hop racing problem [25], in which the worst next-hop candidate in terms of link lifetime is chosen instead of the best candidate one. In

figure 3.1, while node B is in communication with node C, node A needs to establish data communication with node G. As a result, node A broadcasts a RREQ message to its neighbors. The two intermediate nodes, B and D, receive the RREQ packet from A almost at the same time. Since RRD is uniformly distributed and routing traffic is prioritized over data traffic, it is possible that B rebroadcasts the RREQ packet before D. In such scenario, nodes E and F relay and broadcast the packet that was sent by B and node E cancels the RREQ that was sent by D. Finally, G receives the RREQ packet from E or F and responds by a RREP packet via the reversed route to the source node A. Since node B is already involved with communication with C, it will not serve traffic from node A efficiently. Although, D is a better candidate than B, node B is selected as the next-hop node due to the deficiency of the RRD mechanism.

To solve next-hop racing problem, RRD should be accompanied with some positional attributes (velocity and location) or some power attributes in order to assign a high rebroadcast priority for good next-hop candidates to reduce the next hop racing problem, to prevent bad candidates from rebroadcasting, and to alleviate the rebroadcast redundancy.

Recently, many new routing discovery strategies have been proposed to alleviate the next hop racing and redundant broadcasting problems by taking the advantage of cross layer information exchanges. In these mechanisms signal strength, position, velocity, load, bandwidth, delay, and neighborhood information are collected by the link layer, the application layer, or the transport layer and then passed to the routing layer in order to select routes that satisfy the QoS requirements. Further, based on this information, routing layer may control the operations of the link layer by adjusting nodes' transmission power. In this Chapter, we categorize such cross-layer strategies into categories, namely better quality strategy and minimum routing overhead strategy.

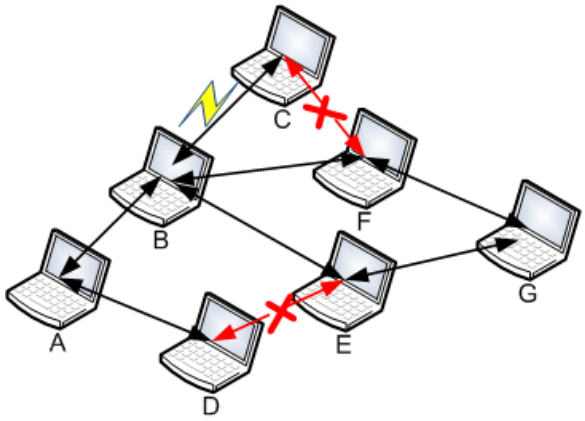

Figure 3.1: Next-hop racing problem

3.1 Better Quality Strategy

The quality of a route implies searching for a feasible path between a source node and a destination node, which satisfies the quality of service (QoS) requirements for each admitted connection and optimizes the use of network resources. QoS requirements may include fairness, route stability, maximum bandwidth, minimum delay, reliability, minimum loss rate, and minimum delay jitter. Several algorithms are proposed to enhance these quality metrics.

3.1.1 Maximum Bandwidth

The core-extraction distributed algorithm (CEDAR) [5] selects a set of nodes in a distributive and a dynamic manner to form the core of the network, which maintains local topology, performs route computation, and performs route maintenance. Each core node propagates bandwidth availability information of stable high bandwidth links to all core nodes, while information of dynamic links or low bandwidth is kept locally. To establish a route between a source and a destination, a core path is established first from the source node dominator to the destination node dominator by using the up-to-date topology information. The source-dominator node finds a path satisfying the requested QoS from the source to a furthest possible core node. This

furthest core node then becomes the source of the next iteration. The above process is repeated until the destination node is reached or the computation fails to find a feasible path. Link failure or destination node movement re-initiates the QoS computation. However, CEDAR is suitable for small and middle size networks. Further, the discovered route is a sub-optimal route and core nodes become bottlenecked.

The ticket-based probing (TBP) algorithm [6] selects multiple paths using imprecise link state information. The bandwidth and delay information are also assumed available. This algorithm tries to limit the flooding by issuing a limited number of tickets based on the available state information. The tickets are distributed amongst the neighbors based on their available resources. However, TBP needs a global state information maintenance performed by a distance vector protocol and it incurs huge control overhead. Further, queuing delay and processing delay of the nodes are not taken into consideration.

The predictive location-based QoS routing (PLBQR) protocol [26] is a link state algorithm, which assumes that each node has information about the whole topology of the network. Each node broadcasts its position and resource information periodically or when a considerable change has occurred. Thus, the future position of nodes and the corresponding delay can be predicted based on the previous location updates. The candidate route, which satisfies the QoS requirements is discovered and added to the transmitted data packet. However, link state algorithms are not suitable for high dynamic networks. Further, the inaccuracy in delay prediction affects the performance of PLBQR.

The proactive QoS routing protocol [8] is proposed for static networks. It integrates the QoS feature into the OLSR protocol. The channel's idle time is calculated by each node and used as a measure for the available bandwidth. The optimal path is the path with a maximum bandwidth. However, the protocol introduces additional protocol overhead, which may affect its performance.

3.1.2 Shortest Routes

The optimized link state routing (OLSR) protocol [27] for mobile Ad-Hoc networks is an optimization of the link state algorithm. It substantially reduces the message overhead by selecting certain nodes as multipoint relays (MPRs), which are forward broadcast messages during the flooding process. MPR node periodically broadcasts a message that contains information about the set of nodes who select it as an MPR. This message is received and processed by all neighbor nodes and rebroadcasted only by the nodes of its MPR set. Thus, each node calculates its routing table using shortest hops path based on the partial network topology it has. The MPR set is selected such that it covers all nodes that are two hops away.

The positional attribute based next-hop determination approach (PANDA) [25] addresses the next hop racing problem. PANDA uses positional attributes such as location and velocity information to set an appropriate value for the RRD. The RRD value depends on the route metric that will be considered, a shortest-hop path, a longest-lived path with a small number of hops, or a minimal power consumption path. Velocity and location information are carried by the RREQ message. Thus, each node compares its own location and velocity with that of the previous-hop's node and then determines the RRD value according to which cost metric it will use.

3.1.3 Longer-Lived Routes

Other efforts have been made to find stable or longer-lived routes. The associativity-based routing (ABR) protocol [28] selects longer-lived routes which helps in reducing the cost of route reconstruction in case of route failure. The location stability or the associativity between nodes is used as the route metric instead of the shortest hop count. Each node determines the link stability by counting the number of beacons, which were sent by its neighbors. Given that beacons are transmitted periodically by nodes to signify their existence with their neighbors, links between nodes are classified into stable and unstable links based on the count of beacons. However,

periodic beaconing consumes network resources. Further, the selected path may not be the shortest path.

Signal stability-based adaptive (SSA) routing protocol [29] extends ABR protocol. It selects longer-lived routes based on the signal strength and location stability. In addition to beacon count, each node keeps a record of the signal strength of its neighbors. The signal strength criterion allows the protocol to differentiate between strong and weak links while the location stability criterion helps the protocol to select long-lived links. Thus, the RREQ packets received from strong or stable links are forwarded while the ones received from weak or unstable links are discarded. However, together these two criterion put a restrict condition on the forwarding RREQ, which result in large setup time if no stable links are found.

3.1.4 Load-Balancing

Network traffic load balancing is another proposed approach, which is used as a metric for optimal routes. Dynamic Load-Aware Routing (DLAR) [30] considers the load of intermediate nodes (the number of packets buffered in their interfaces) as the main route selection metric. It monitors the congestion level of active routes in order to reconstruct the least-loaded routes. Flooding is the main mechanism for route discovery where intermediate nodes add their load information to the RREQ messages. The destination node waits to receive multiple RREQ messages and then chooses the least-loaded path to send over it a RREP message to the source node. However, DLAR does not optimally reflect the actual load since buffered packets may vary in size.

The load-sensitive routing (LSR) protocol [31] uses the network information as the main route selection criterion. The proposed algorithm assumes the nearby paths' traffics interfere with each other and influence the routing performance. Therefore, the network load depends on the traffic passing a mobile host as well as the traffic passing the neighbor nodes. LSR defines the load metric of a node as the total number

of packets buffered in the node interface and its neighbors. However, LSR does not take into account the different sizes of the buffered packets.

The load-balanced ad hoc routing (LBAR) protocol [32] is an on-demand routing protocol proposed for delay sensitive applications. LBAR assumes that a least delay path is the path with minimum load and minimum interference. LBAR considers the load metric of a node is the total number of routes flowing through the node and its neighbors. The destination selects the least congested path based on the aforementioned metric. However, this method is not optimal since it does not account for the various traffic sizes in each route.

3.1.5 Minimum Power Consumption

Nodes in MANET have the capacity to modify the area of coverage by adjusting their transmission powers. Indeed, controlling the transmission power levels significantly reduces energy consumption and increases the lifetime of the network. However, adjustment of transmission signal strength generally implies alteration in the network topology and leads to the loss of network connectivity. Hence, nodes have to manage their coverage area while maintaining the connectivity of the network. Most of the energy-aware proposed protocols are based on the minimum spanning tree (MST). They are global because computing the MST requires a global information about the network. Recently, localized protocols have been proposed using the relative neighborhood graph (RNG) [33] and the local minimum spanning tree (LMST) [34]. The optimization criterion for such category is to minimize the total transmission powers while preserving the complete connectivity of the network. Such optimization problem is an NP-hard for tree-dimensional [35] and k-dimensional [36] spaces where $k \geq 2$. Heuristic global solutions are proposed to implement such control topology protocols.

The topology control protocol based on MST [37] considers the total energy of the broadcast tree as the sum of the energy expended by all transmitting nodes of the

tree. Therefore, to minimize the total energy consumption, each node should adjust its transmission power to a level at which it remains connected with its neighbors in the MST.

In [37] two greedy centralized protocols were proposed, namely broadcast incremental power (BIP) and broadcast least-unicast-cost (BLU). BIP construction goes through four steps. In the first step, a source node searches for a first node that can be reached with a minimum expenditure of power. In the second step, the source searches for a second node that can be added to the tree at a minimum additional cost either by increasing its transmission power to reach the second node or determining the necessary transmission power needed to connect the first node with the second one. The alternative with a minimum incremental power increase is chosen. Now there are three nodes that joined the tree. In the third step, the incremental cost to reach a fourth node is calculated by the previous three nodes and a node with a minimum incremental cost is added to the tree. Finally, the procedure is continued until all nodes are included in the tree. On the other hand, BLU uses a straightforward approach that modifies Bellman-Ford or Dijkstra algorithm cost functions for constructing the minimum-power paths. In BLU, the cost function is modified from a minimum distance to a minimum consumption power.

The RNG topology control protocol (RTCP) [38] is a localized approach, which replaces the MST by the RNG. The RNG can be deduced locally by each node by using only the distances to its neighbors. Distances to neighbors can be explicitly or implicitly determined. The distances can be determined explicitly by using the location information transmitted periodically from the neighbor nodes. The distances can be determined implicitly by analyzing the signal strength or the time delay information of the received packets. The connectivity of RNG assures that all nodes receive the message for any choice of a source node. Thus, the RTCP provides an efficient energy saving even if blind flooding is used as a broadcasting mechanism.

In RTCP, energy consumption can be enhanced furthermore by ignoring covered RNG neighbors from previous transmissions when deciding to transmit or by discard-

ing the received RREQ message. By injecting these enhancements to the RTCP, the RTCP becomes a RNG broadcast oriented protocol (RBOP) [38]. In RBOP, a source node transmits it message with a determined range $R(u)$, which is defined by applying the RTCP. When a node receives a new broadcast message, it checks the transmitter whether it is a RNG neighbor or a non-RNG neighbor. If the transmitter is a RNG neighbor, the node calculates the distance to the furthest RNG neighbor that did not receive this message and then sends the message according to this calculation. If all RNG neighbors have received the transmitted message, then the message transmission is ignored. If the transmitter is a non-RNG neighbor, the node generates a list of the RNG neighbors that did not receive this message. After a certain timeout, if the neighbor list is not empty, the node retransmits the message with a range that reaches the furthest neighbor in the associated list.

The local minimum spanning tree (LMST) protocol [34] is a minimum spanning tree-based topology control algorithm. LMST is proposed for multihop wireless networks with limited mobility. To construct the topology, each node builds its local MST independently by using the position information of its neighborhood, which is carried by the periodic HELLO messages. Then, each node determines the specific power levels it needs to reach all neighbors based on the received signal strength. Then, the topology is constructed using only the bidirectional links either by enforcing all the unidirectional links to become bidirectional or by deleting all the unidirectional links.

3.2 Lower Routing Overhead

Flooding consumes scarce network resources, power and bandwidth, due to unnecessary routing packet transmissions. Therefore minimizing the overhead is one of the main design issues in the broadcasting protocols.

The location-aided routing (LAR) protocol [39] utilizes location information to improve the performance of ad hoc wireless networks. It tries to minimize the number of routing messages by limiting the search for a new route to a smaller zone. Indeed,

the operation of LAR depends mainly on the assumption that the source node knows the previous location L and the speed v of the destination node at time t_0. LAR proposes two schemes for limiting the searching zone. In the first scheme, LAR defines the expected-zone (a circular region of radius $v(t_1 - t_0)$ centered at L) which is the region that the source node expects to find the destination node in it at time t_1. In addition to the expected-zone, a request-zone is defined as the zone that contains both the source node and the expected-zone. When the source starts the route discovery phase, it includes the requested region boundaries in the RREQ message. When an intermediate node receives the RREQ message, it decides whether to discard the RREQ message if it is out of the requested-zone or to rebroadcast it. In the second scheme, the source node includes the information of the destination node location (x, y), the velocity, and its distance from the destination in the RREQ message. An intermediate node forwards the RREQ if it is closer to the destination than the sender node.

In [40], the two heuristic algorithms, self-pruning and dominant-pruning were proposed, which flood packets more efficiently than the blind flooding. Both protocols reduce the unnecessary transmissions by adapting the neighborhood information exchange mechanism between mobile nodes. In the self-pruning, each node exchanges its neighborhood information with its neighbors through the RREQ message. A node that receives the RREQ compares its neighbor list to the senders' neighbor lists and it refrains from rebroadcasting if its neighbor list is included in one of the senders' neighbor list. In the dominant-pruning, the range of neighborhood information is extended to include two-hop neighborhood. The sender analyzes the neighbor lists of its neighbor nodes and forms a forward list, which contains the nodes that should relay the packet to complete the broadcast. Forward list is the minimal set of adjacent neighbors that their transmission will cover all the 2-hop neighbors. Their simulation results show that both pruning mechanisms outperform the blind flooding mechanism. In fact, the dominant-pruning has a greater performance gain but it has the larger overheads especially when the host mobility increases. Thus, the self-pruning is

suitable for high mobility environments while the dominant-pruning is a good choice for low and moderate mobility environments. The ad hoc broadcast protocol (AHPC) [41] and the connected-dominating set-based broadcast [42] protocol are similar to the dominant-pruning algorithm.

In the scalable broadcast algorithm (SBA) [43], each node builds 2-hop topology centered at it. When the node receives a RREQ packet it excludes the neighbors of the RREQ's senders from its local topology. If there are additional neighbors that did not receive the RREQ, it schedules the transmission of the received RREQ after a certain backoff delay. During the backoff delay period, if the node hears the same packet transmission, it determines if there are additional nodes that received the RREQ and should be eliminated from the list. This process continues until either the backoff timer expires, or the packet transmission is canceled.

The multipoint relaying (MPR) [44] is a deterministic method for a reliable broadcasting. It requires two-hop topology information to select a minimal set of nodes from its 1-hop neighbors that covers completely the 2-hop neighbors. Since the computation of the minimal set is NP-complete problem [44], a heuristic greedy algorithm to find the minimal set was proposed.

The RNG relay subset (RRS) [33] is a source-dependent broadcasting protocol based on self selection and neighborhood elimination mechanisms. A source node, which desires to begin a broadcast, sends its message. When an intermediate node receives the message, it generates a list of RNG neighbors, which contains the nodes that did not receive the message. If this list is empty, the node drops the message and ignores any duplicate message. Otherwise, the node sets up an RRD timeout and starts eliminating its neighbor nodes, which have received the same broadcast during the RRD from that list. If the RRD expires and the list is not empty, the message is rebroadcasted.

The location-aided knowledge extraction routing (LAKER) protocol [45] was introduced to reduce the flooding overhead during the route discovery phase. LAKER is a descendant of the DSR and the LAR and it can discover gradually the knowledge

of the topological characteristics such as the nodal population density distribution of the network. It limits the LAR request-zone to a zone that only covers the guiding-routes (a series of locations along the route where there are many nodes around. These guiding information narrows the search space in the route discovery process and overcomes the problem of "void" area in the network. This kind of information is discovered and cached during the route discovery process and then can be used in the subsequent route discovery rounds). For proper operation of LAKER, each node requires to know its location, the number of its neighbors, and the destination node's location. In the route discovery phase, the guiding-routes are used to direct the search for a forward route to the destination. If the intermediate node lay outside the LAKER's request-zone, it discards the RREQ. Otherwise, it forwards the RREQ and updates the guiding-routes in the RREQ.

The routing protocol with selective forwarding (RPSF) [46] is a novel algorithm for route discovery that tries to minimize the propagation of the redundant RREQs by limiting the number of nodes that forward any RREQ packet. RPSF chooses only a subset of nodes as forwarding nodes and ensures that the RREQ message eventually reaches the destination unless the network is partitioned. The source node selects three nodes as forward nodes while other nodes select only two nodes. This selection criterion divides the surrounding region into three broadcast areas. The distance between the selected nodes and the node itself is as far as possible and the angle between the node and its selected neighbors is $\leq 120^o$. The neighbor nodes selection is done in such a way to ensure that the RREQ message propagation covers the entire network. The route discovery cycle is initiated by the source node, which elects a list of forwarding nodes, stores the forwarding node list in the RREQ, and broadcasts the RREQ message. Only nodes that are available in the list are handling the RREQ by forwarding it in the same way in the other directions. As the destination node or an intermediate node with fresh enough route to the destination receives the RREQ, it broadcasts a RREP packet to the source node. RPSF floods both the RREQ and the RREP. Thus, the source node can maintain multiple routes to the destination.

3.3 Summary

In this chapter, we reviewed some of the proposed mechanisms to enhance the route discovery process in wireless ad hoc networks, which are categorized into two classes: better quality strategy and lower overhead strategy. These proposed mechanisms utilize cross-layer information to enhance the quality of the discovered routes but each mechanism adapts a single routing metric. For example, CEDAR, TBP, PLBQR and proactive QoS mechanisms try to maximize the bandwidth. ABR and SSA use signal strength and neighborhood information to select long-lived routes. DLAR, LSR, and LBAR use load and network information to balance the traffic load over the selected routes. RTCP, RBOP, LMST, BLU, and BIP use position and neighborhood information to minimize the total transmission power. Finally, MPR, RRS, SBA, LAR, LAKER, and RPSF use neighborhood and location information to minimize routing overheads.

Chapter 4

Saturation Performance Analysis of the IEEE 802.11 DCF

4.1 Introduction

This Chapter addresses the performance of the 802.11 DCF with RTS/CTS access mode in error-prone channel. Specifically, the error recovery mechanism in RTS/CTS is implemented using two independent retry counters to control the number of data and control packet transmissions. To the best of our knowledge, there is no analytical model that integrates the two retry counters in one model. Here, we address this issue and provide a 3-D Markov chain model to evaluate the saturation performance of the RTS/CTS in error-prone channel. The 3-D model enables us to accurately capture the important DCF performances metrics: throughput, packet discard probability, packet delay, and packet discard delay. Moreover, our 3-D model is able to capture the stochastic behavior of the basic access mode in which a single retry counter is used to implement the error recovery mechanism.

Analytical models for performance analysis of IEEE 802.11 RTS/CTS access mode have widely been reported in the literature. To obtain approximate expressions for the saturated throughput[1], [47, 48, 49] used the Markov-like chain approach and

[1]The maximum load that the system can carry in a saturation condition (i.e., stations always

[50, 51] used the elementary probability theory. Other analytical models studied the performance of IEEE 802.11 DCF under finite buffer and load [52, 53, 54, 55, 56, 57]. However, none of the models reported in literature consider the two retry counters together. In particular, the stochastic behavior of the RTS/CTS is either modeled in error-free channel, which revokes the necessity for the data retry counter, or error-prone channel with a single data transmission attempt. Actually, the two assumptions are not realistic. In imperfect channels, bit transmission errors may corrupt not only large packets but also small ones. Therefore, the lack of the data retransmission model will not provide an appropriate estimate for the IEEE 802.11 performance metrics.

This Chapter is organized as follows: Section 4.2 presents our new analytical model for the DCF saturation performance in error-prone channel. Section 4.3 shows the applicability of our model to analyze the saturation performance of the basic access mode. The saturation performance of the hybrid system in which the packet is transmitted by means of the RTS/CTS mechanism only if its payload size exceeds a predefined threshold (*RtsThreshold*), is discussed in Section 4.4. Simulation, verification, and numerical investigation are discussed in Section 4.5. Finally, Section 4.6 summarizes the Chapter.

4.2 System Performance Analysis

This section introduces and analyzes the discrete-time 3-D Markov chain model for the RTS/CTS access mode. The transmission probability is derived and used to study the performance of the IEEE 802.11 system.

4.2.1 Assumptions

The stochastic behavior of the RTS/CTS access mode is modeled using a discrete-time 3-D Markov chain. The stochastic behavior of a single station (the tagged station) is studied based on the following assumptions:

have packets for transmission in their transmission buffers).

(i) The network has a finite number of homogeneous stations (n), which run IEEE 802.11 DCF mechanism, use the same RTC/CTS access mode, and hear each other (i.e., single hop communications). In Section 4.3 and Section 4.4, this assumption will be released, i.e., the basic access only mode and hybrid mode will be addressed.

(ii) All stations work in saturation mode, where they always have packets to transmit in their infinite transmission buffers.

(iii) The packet's collision probability is constant and independent of the packet retransmission history. Collision occurs if at least one station in addition to the tagged station transmit in the same slot time. Assuming that each node transmits with a probability τ and collides with a constant and independent probability p_c, then

$$p_c = 1 - (1 - \tau)^{n-1} \tag{4.1}$$

(iv) error-prone channel is modeled as a Gaussian wireless error channel in which bit errors are identically and independently distributed (i.i.d.) over the whole packet thus, each bit has the same bit error rate (BER). Although the Gaussian channel model cannot capture the multipath fading, it is widely used due to its simplicity. The BER can be estimated by measuring the signal-to-noise ratio (SNR) of the received signal. From SNR, the bit-energy-to-noise (E_b/N_0) ratio can be calculated:

$$E_b/N_0 = (SNR)(W/R_b) \tag{4.2}$$

where R_b is the transmission bit rate and W is the channel bandwidth.

For M-ray QAM ($M = L^2$, which equals to *16* or *64* in 802.11a), BER can be calculated using the following formula:

$$BER = \frac{2(1-L^{-1})}{log_2 L} Q\left(\sqrt{\frac{3Log_2 L}{L^2-1}\frac{2E_b}{N_0}}\right) \tag{4.3}$$

where $Q(x) = \int_x^\infty \frac{1}{\sqrt{2\pi}} e^{-t^2/2} dt$.

In 802.11 systems, control packets use the most reliable modulation, i.e. BPSK, while data packets generally use much higher modulation rates, e.g. QPSK, CCK, or QAM. This results in huge differences between the BER for control packets (RTS, CTS and ACK) and data packets [58]. Control packets loss probability is typically *1000,000* times smaller than the data packet loss. Consequently the loss effect due to imperfect channel is 99.9999% dominated by data loss rather than control packet loss (contributing to less than 0.0001% loss). As a result, it is safe to ignore the control packet loss in our model. Now, assume the length of the data packet is constant and equal to l_{data} bits, the data packet error probability (p_e) is equal to:

$$p_e = 1 - (1 - BER)^{l_{data}} \tag{4.4}$$

(v) In each channel reservation attempt, regardless the number of reservations, the channel has a constant and independent reservation failure probability p. Since the control packet loss due to transmission errors is ignored in the model, then this probability equals the collision probability given in (4.1). Also, in each data packet transmission attempt, regardless the number of retransmissions, each data packet has a constant and independent success or failure probability, α_s or α_f respectively. Given that all data packet transmissions should be proceeded by a successful RTS/CTS dialogue, α_s and α_f can be expressed as:

$$\begin{cases} \alpha_s = (1-p)(1-p_e) \\ \alpha_f = (1-p)p_e \end{cases} \tag{4.5}$$

4.2.2 System Model

The backoff procedure can be modeled as a parallel set of 2-D Markov chains. The number of 2-D chains is equivalent to the value of the data retry limit (R_2). Each 2-D chain is similar to the one proposed in [49] with the necessary change in transition probabilities and the initial CW size.

Let $z(t)$, $y(t)$, and $x(t)$ be stochastic processes that represent at time t the station long retry count $slrc$ with state space $\{0, 1, \ldots, R_2 - 1\}$, the station short retry count ($ssrc$) with state space $\{0, 1, \ldots, R_1\}$, and the backoff counter with state space $\{0, 1, \ldots, W_{j,i} - 1\}$, respectively. The $W_{j,i}$ represents the CW value when $slrc = j$ and $ssrc = i$. For $0 \leq j \leq R_2 - 1$, $W_{j,i}$ is equal to:

$$W_{j,i} \quad = \quad \min\left\{CW_{max}, 2^i W_{j,0}\right\} \tag{4.6}$$

$CW_{max} = 2^m CW_{min}$, where m is a positive-integer number that limits the value of CW. The initial value of CW for the current data transmission attempt is twice its last value in the previous transmission attempt; up to CW_{max}. Given that $W_{0,0} = CW_{min}$ and $1 \leq j \leq R_2 - 1$, $W_{j,0}$ is equal to:

$$W_{j,0} \in \min\left[W_{max}, 2 \times \{W_{j-1,0}, W_{j-1,1}, \ldots, W_{j-1,ssrc}\}\right] \tag{4.7}$$

To keep on the memoryless property of the Markov process, $W_{j,0}$ will be weighted average over its all possible values. Thus, $W_{j,0}$ is equal to:

$$W_{j,0} \quad = \min\left\{CW_{max}, 2 \times \left\lceil \left(\sum_{i=0}^{R_1} p^i \times W_{j-1,i}\right) \Big/ \sum_{i=0}^{R_1} p^i \right\rceil \right\} \tag{4.8}$$

The process $\{z(t), y(t), x(t)\}$ is a discrete-time 3-D Markov process with state space of $(j, i, w) : j \in [0, R_2 - 1], i \in [0, R_1], w \in [0, W_{j,i} - 1]$ and the time scale is discrete and integral, where t and $t+1$ correspond to the beginning of two consecutive slot times.

4.2.3 Transition Probabilities

Figure 4.1 shows the 2-D Markov model for the j^{th} data transmission attempt ($slrc = j$). The first row, indexed from $(j, 0, 0)$ to $(j, 0, W_{j,0})$, is the stage-*0* backoff states, which represents the first RTS transmission attempt ($ssrc = 0$). The second row of states, indexed $(j, 1, 0)$ to $(j, 1, W_{j,1} - 1)$ is the stage-*1* backoff states, which represents the second RTS transmission attempt ($ssrc = 1$). The last row, indexed from $(j, R_1, 0)$

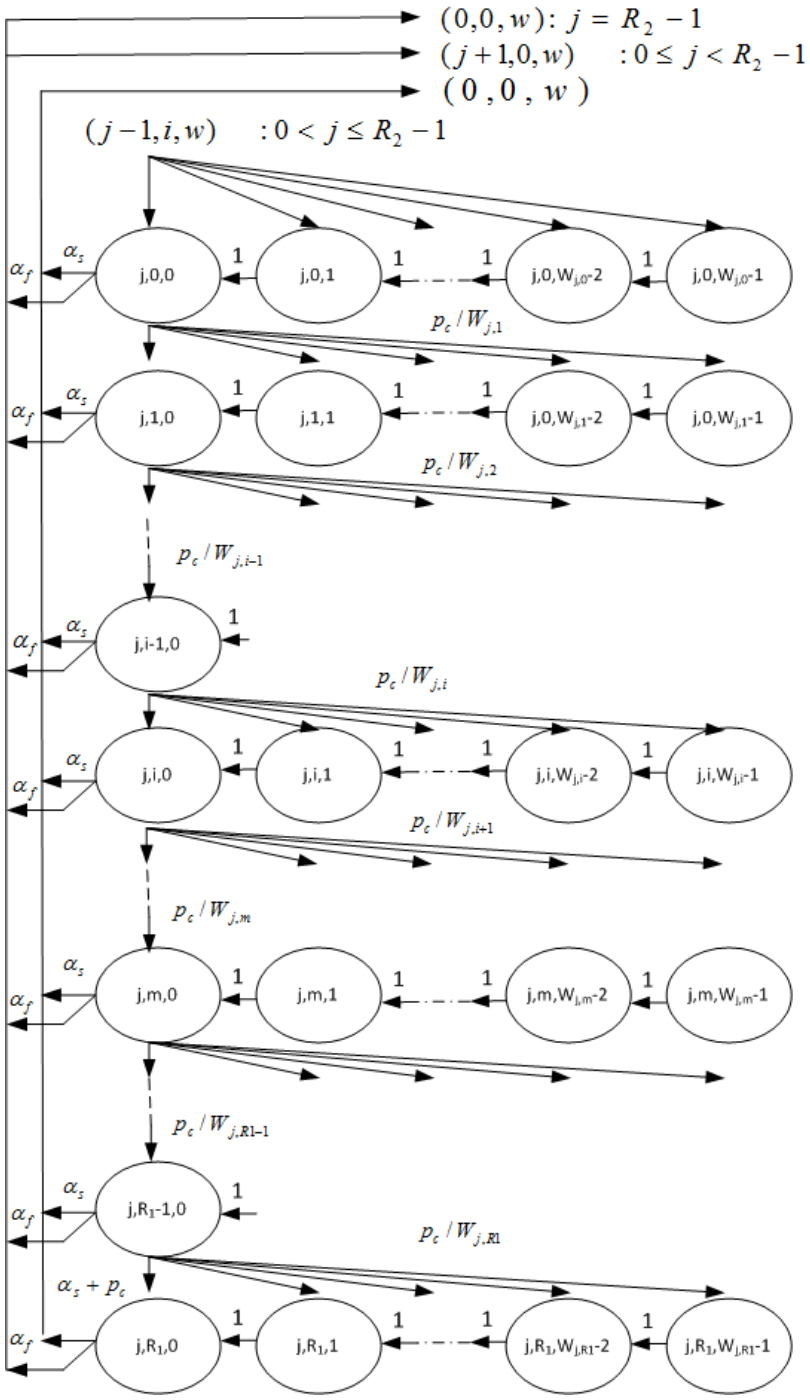

Figure 4.1: The 2-D Markov for the j^{th} data transmission attempt ($slrc = j$).

to $(j, R_1, W_{j,R_1} - 1)$, is the stage-$(R_1 + 1)$ backoff states, which represents the last RTS transmission attempt ($ssrc = R_1$).

Consider a packet transmission event at the tagged station, where the tagged station starts the initial backoff by randomly and uniformly choosing one of the stage-*0* backoff states, $(0, 0, w)$. At state $(0, 0, w)$, the station decrements the backoff counter and enters the state $(0, 0, w - 1)$ with a probability 1 if the channel remains idle for a DIFS period. The backoff countdown process continues until it reaches the state $(0, 0, 0)$. In general, for $j \in [0, R_2 - 1], i \in [0, R_1], w \in [0, W_{j,i} - 2]$, the backoff countdown process can be expressed as:

$$P\Big[(j, i, w) \,|(j, i, w + 1)\Big] = 1 \tag{4.9}$$

When the station reaches the state $(j, i, 0)$, it shall try to reserve the channel by transmitting an RTS packet. The RTS transmission may success or fail with probability $(1 - p)$ or p, respectively. If the RTS transmission fails, the station doubles the CW value, increments the $ssrc$, and then transits the backoff process randomly and uniformly to the state $(j, i + 1, w)$ for another channel reservation attempt.

$$P\Big[(j, i + 1, w) \,|(j, i, 0)\Big] = p/W_{j,i+1} \tag{4.10}$$

At state $(j, R_1, 0)$, there is no more RTS transmission attempts. If the station cannot successfully reserve the channel, it shall discard the data packet and transit the backoff process to the state $(0, 0, w)$ in order to transmit the next head-of-queue packet. Always when the station discards a data packet, it shall initialize the $ssrc$, $slrc$, and CW.

$$P\Big[(0, 0, w) \,|(j, R_1, 0)\Big] = p/W_{0,0} \tag{4.11}$$

Otherwise, if the station reserves the channel (that is, the tagged station receives the CTS correctly), it shall reset the $ssrc$ and transmit the data packet.

The transmitted data packet may be corrupted or sent correctly. If the packet is corrupted by bit transmission errors, the station shall increment the $slrc$, double the

CW size, and transit to the state $(j+1, 0, w)$ for a new transmission attempt.

$$P\Big[(j+1,0,w)\,|(j,i,0)\Big] = \alpha_f / W_{j+1,0} \tag{4.12}$$

The station continues transmitting the data packet after each failed attempt until the number of attempts is exhausted ($j > R_2 - 1$). After that, the station shall discard the data packet and initialize the $slrc$ and CW. Next, the station shall start a new backoff process to transmit the next head-of-queue data packet.

$$P\Big[(0,0,k)\,|(R_2-1,i,0)\Big] = \alpha_f / W_{0,0} \tag{4.13}$$

However, if the data packet has been sent properly at any attempt (that is, the station receives the ACK correctly), the station shall initialize $slrc$ and CW. Next, the station shall start a new backoff process to transmit the next head-of-queue data packet.

$$P\Big[(0,0,w)\,|(j,i,0)\Big] = \alpha_s / W_{0,0} \tag{4.14}$$

4.2.4 Transmission Probability

The non-null transition probabilities (4.9)-(4.14) can be expressed in a highly reduced mathematical form in terms of $S_{0,0,0}$ and the two conditional probabilities p and α_f, where $S_{0,0,0}$ is the probability of being in the state $(0,0,0)$. Let $\gamma = \alpha_f \times \sum_{i=0}^{R_1} p^i$. then for $w \in [0, W_{0,0} - 1]$,

$$S_{0,0,w} = \frac{W_{0,0} - w}{W_{0,0}} \left(\alpha_s \sum_{j=0}^{R_2-1} \sum_{i=0}^{R_1} S_{j,i,0} + p \sum_{j=0}^{R_2-1} S_{j,R_1,0} + \alpha_f \sum_{i=0}^{R_1} S_{R_2-1,i,0} \right) \tag{4.15}$$

For $j \in [1, R_2 - 1]$ and $w \in [0, W_{j,0} - 1]$,

$$S_{j,0,w} = \frac{W_{j,0}-w}{W_{j,0}} \left(\alpha_f \sum_{i=0}^{R_1} S_{j-1,i,0} \right) \tag{4.16}$$

For $j \in [0, R_2 - 1]\,, i \in [1, R_1]$,

$$\begin{aligned} S_{j,i,w} &= \frac{W_{j,i}-w}{W_{j,i}} \Big(p \times S_{j,i-1,0} \Big) \\ &= \frac{W_{j,i}-w}{W_{j,i}} \Big(p^i \times S_{j,0,0} \Big) \end{aligned} \tag{4.17}$$

Now, based on the fact that transmission is only allowed when the backoff timer value reaches zero ($w = 0$), the transmission probability τ, the probability that the station transmits in a randomly chosen slot time, is:

$$\begin{aligned} \tau &= \sum_{j=0}^{R_2-1} \sum_{i=0}^{R_1} S_{j,i,0} \\ &= S_{0,0,0} \times \sum_{j=0}^{R_2-1} \gamma^j \sum_{i=0}^{R_1} p^i \end{aligned} \tag{4.18}$$

$S_{0,0,0}$ can be determined by imposing the normalization condition:

$$\begin{aligned} 1 &= \sum_{j=0}^{R_2-1} \sum_{i=0}^{R_1} \sum_{w=0}^{W_{j,i}-1} S_{j,i,w} \\ &= S_{0,0,0} \times \sum_{j=0}^{R_2-1} \gamma^j \sum_{i=0}^{R_1} \left(p^i \frac{W_{j,i}+1}{2} \right) \end{aligned} \tag{4.19}$$

Taken in consideration that $p = p_c$, (4.1) can be inverted to express $\tau^*(p)$ as:

$$\tau^*(p) = 1 - (1-p)^{\frac{1}{n-1}} \tag{4.20}$$

Equations (4.18) and (4.20) represent a nonlinear system of equations with two unknowns p_c and τ, which can be solved using numerical techniques.

4.2.5 Throughput

At any chosen slot time, the channel is distinguished as idle, success, or fail, depending on whether a slot on the channel is idle, a data packet is successfully transmitted during the slot, or a failed transmission has occurred during that slot, respectively. Let the pair (P_i, T_i) represent the probability and time duration of an idle slot time. Let the pair (P_s, T_s) represent the probability and time duration of a successful data transmission. A successful transmission occurs if only one station transmits during the slot and bit transmission errors do not corrupt the packet. Let the pair (P_c, T_c) represent the probability and time duration when any two or more stations start transmission in a same slot time. Let (P_e, T_e) pair represents the probability and

time duration of unsuccessful transmission when only one station transmits in a time slot and the packet is corrupted by transmission errors. Equations (4.21) and (4.22) show these different probabilities and their corresponding time durations (letter superscripts, r, b, and h, indicate difference among *RTS/CTS*, basic, and hybrid time durations respectively):

$$\begin{cases} P_i &= (1-\tau)^n \\ P_s &= n\tau\,(1-\tau)^{n-1}\,(1-p_e) \\ P_c &= 1-(1-\tau)^n - n\tau\,(1-\tau)^{n-1} \\ P_e &= n\tau\,(1-\tau)^{n-1}\,p_e \end{cases} \tag{4.21}$$

$$\begin{cases} T_i^r &= \sigma \\ T_s^r &= l_{rts}/R_b + \delta + SIFS + l_{cts}/R_b + \delta + SIFS + l_{data}/R_d + \delta + SIFS \\ &\quad + l_{ack}/R_b + \delta + DIFS \\ T_c^r &= l_{rts}/R_b + \delta + EIFS \\ T_e^r &= l_{rts}/R_b + \delta + SIFS + l_{cts}/R_b + \delta + SIFS + l_{data}/R_d + \delta + EIFS \end{cases} \tag{4.22}$$

where l_{rts}, l_{cts}, l_{data}, and l_{ack} represent the length of the RTS, CTS, data, and ACK packets inclusive of the required overheads respectively. Also, δ, R_b, and R_d represent the propagation delay, basic bit rate, and data bit rate respectively.

The normalized throughput is defined as the fraction of time the channel is used to successfully transmit the useful payload bits (l_P) during any chosen slot time. Thus, the normalized throughput can be expressed as:

$$S = \frac{l_p}{R_d} \times P_s/t_{slot}^r \tag{4.23}$$

where (t_{slot}^r) is the average length of a slot time in RTS/CTS mode. If there is no activity on the channel, then t_{slot}^r is the system slot time (σ). Otherwise, it would be the time to complete a successful transmission (T_s^r), the time to recover from a

collision (T_c^r), or the time to recover data packet corruption (T_e^r). Using (4.21) and (4.22), t_{slot}^r is equal to:

$$t_{slot}^r = P_i\sigma + P_sT_s^r + P_cT_c^r + P_eT_e^r \tag{4.24}$$

4.2.6 Packet Discard Probability

If $ssrc > R_1$ or $slrc > R_2 - 1$, the data packet is discarded from the transmission buffer. Specifically, the data packet is discarded if at state $(j, R_1, 0)$, $0 \leq j \leq R_2 - 1$, the station could not reserve the channel or at state $(R_2 - 1, i, 0)$, $0 \leq i \leq R_1$, the station reserved the channel but bit transmission errors corrupt the transmitted data packet. Hence, packet discard probability ($P_{discard}$) can be expressed as:

$$P_{discard} = p^{R_1+1} \sum_{j=0}^{R_2-1} \gamma^j + \alpha_f \times \gamma^{R_2-1} \sum_{i=0}^{R_1} p^i \tag{4.25}$$

4.2.7 Packet Delay

Packet delay (D_{trans}) is defined as the time duration from the instant that the packet starts to contend for the channel to the successful transmission instant, this time includes all transmission attempts. The average delay is conditioned on the event that the packet is not discarded. To find D_{trans}, we need to find the average number of slots ($E[X]$) that are required to transmit the data packet successfully. The packet may be transmitted successfully at state $(j, i, 0)$, $0 \leq i \leq R_1$ and $0 \leq j \leq R_2 - 1$. Therefore, $E[X]$ is the probability that the transmission succeeds in state $(j, i, 0)$ times the average number of slots that are required to reach that state; conditioned on the event that the packet will not be discarded $(1 - P_{discard})$. Thus,

$$D_{trans} = E[X] \times t_{slot}^r \tag{4.26}$$

$$E[X] = \frac{\alpha_s}{1 - P_{discard}} \times \left(\sum_{j=0}^{R_2-1} \gamma^j \sum_{i=0}^{R_1} \left(p^i \times \overline{NS}_{j,i} \right) \right) \tag{4.27}$$

where $\overline{NS}_{j,i}$ is the average number of slots that are required to reach that state, which is equal to,

$$\overline{NS}_{j,i} = \sum_{k_1=0}^{j} \sum_{k_2=0}^{i} \frac{W_{k_1,k_2}+1}{2} \tag{4.28}$$

4.2.8 Packet Discard Time

Packet discard time ($D_{discard}$) is defined as the time duration from the instant that the packet starts to contend for the channel to the packet discard instant. The discard delay is conditioned on the event that the packet is discarded. The packet is discarded if $ssrc > R_1$ or $slrc > R_2 - 1$. Thus, the packet may be discarded from the state $(j, R_1, 0)$, $0 \leq j \leq R_2 - 1$, with a probability p or from the state $(R_2 - 1, i, 0)$, $0 \leq i \leq R_1$, with a probability α_f. To find $D_{discard}$, we need to find the average number of slots ($E[Y]$) that are required to discard the packet. Thus, $E[Y]$ is the probability that the packet is discarded in state $(j, R_1, 0)$ or $(R_2 - 1, i, 0)$ times the average number of slots that are required to reach that state; conditioned on the event that the packet will be discarded ($P_{discard}$). Thus,

$$D_{discard} = E[Y] \times t^r_{slot} \tag{4.29}$$

$$E[Y] = \frac{1}{P_{discard}} \times \left(p^{R_1+1} \sum_{j=0}^{R_2-1} \left(\gamma^j \overline{NS}_{j,R_1} \right) + \alpha_f \times \gamma^{R_2-1} \sum_{i=0}^{R_1} \left(p^i \overline{NS}_{R_2-1,i} \right) \right) \tag{4.30}$$

4.3 Basic Access Mode

In the basic access mode, if the tagged station senses the channel idle for a DIFS period, it proceeds with data transmission. When the destination receives the data frame correctly (the data packet does not collide or corrupt by transmission errors), it waits for a SIFS period before responding with an ACK frame to confirm a correct reception. If the receiver does not send the ACK, the sender starts the error recovery mechanism after a *data_timeout* period.

Unlike the RTS/CTS mode, to recover from collision and transmission errors, the station requires only one retry counter to count the number of data transmissions. if the packet size is larger than *RtsThreshold*, it uses the *slrc*. Otherwise, it uses the *ssrc*. After each failed transmission, the sender doubles the CW; up to CW_{max}, increments the associated retry counter, and then transmits the data frame. Retransmissions are done until the number of attempts reaches the corresponding retry limit value or the frame has been successfully transmitted. When the retry counter reaches its limit, the frame is discarded from the system. Our 3-D model can be smoothly used to model the basic mode by adjusting the values of certain control parameters. Specifically, for small size data frames ($\leq$ *RtsThreshold*), we set $R_2 = 1$, $\alpha_f = 0$, $\alpha_s = (1-p)(1-p_e)$, and $p = 1 - \alpha_s$. For lengthly data frames ($>$ *RtsThreshold*), we set $R_1 = 0$, $p = 0$, $\alpha_s = (1-p)(1-p_e)$, and $\alpha_f = 1 - \alpha_s$.

To calculate the average length of a slot time in the basic access mode (t^b_{slot}), the cost of handshaking the RTS and CTS frames should be excluded from (4.22). Equations (4.31) and (4.32) show the time durations and t^b_{slot} for the basic access mode:

$$\begin{cases} T^b_i &= \sigma \\ T^b_s &= l_{data}/R_d + \delta + SIFS + l_{ack}/R_b + \delta + DIFS \\ T^b_c &= l_{data}/R_d + \delta + EIFS \\ T^b_e &= l_{data}/R_d + \delta + EIFS \end{cases} \tag{4.31}$$

$$t^b_{slot} = P_i\sigma + P_sT^b_s + P_cT^b_c + P_eT^b_e \tag{4.32}$$

However, (4.23), and (4.25)-(4.30) can be used to calculate S, $P_{discard}$, D_{trans}, and $D_{discard}$ for the basic mode.

4.4 Hybrid Mode

In hybrid system, packets are transmitted using RTS/CTS only if their payload lengths exceed *RtsThreshold* ($l_p >$ *RtsThreshold*), otherwise they are transmitted

using the basic mode. Let $F(.)$ be the probability distribution function of the payload size, then $F(\overline{P})$ ($\overline{P}$ represents the $RtsThreshold$) is the probability that the packet is transmitted using the basic mode when its payload size $\leq RtsThreshold$ and $(1 - F(\overline{P}))$ is the probability that the packet is transmitted using the RTS/CTS mode.

In such environment, data packets may collide with RTS packets. Hence, T_s^h, T_c^h, and T_e^h calculations should take this fact in consideration as shown in [47]. Accordingly, t_{slot}^h, S, $P_{discard}$, D_{trans}, and $D_{discard}$ can be expressed as:

$$\begin{cases} t_{slot}^h & = P_i\sigma + P_sT_s^h + P_cT_c^h + P_eT_e^h \\ S & = (E(l_p)/R_d)P_s/t_{slot}^h \\ P_{discard}^h & = F(\overline{P})P_{discard}^b + (1 - F(\overline{P}))P_{discard}^r \\ D_{delay}^h & = F(\overline{P})D_{delay}^b + (1 - F(\overline{P}))D_{delay}^r \\ D_{discard}^h & = F(\overline{P})T_{discard}^b + (1 - F(\overline{P}))T_{discard}^r \end{cases} \tag{4.33}$$

where $E(l_p)$ is the average packet payload size.

4.5 Verification and Performance Investigation

For simplicity, in this section, we restrict our simulation and analytical analysis to the case of fixed length packet size where $E(p) = l_p$ and $l_p > RtsThreshold$. We will evaluate the performance of the system in which all stations operate according to the RTS/CTS mode.

4.5.1 Verification

We implement a simulation model using the network simulator (ns) [59] to validate the results obtained from the analytical model. The free space propagation model is used to predict the signal power received by the receiver. The signal strength is used to determine if the frame is received successfully or not. Specifically, ns

uses three thresholds, the carrier sensing threshold ($CSThresh$), reception threshold ($RxThresh$), and capturing threshold ($CPThresh$) to determine whether a frame is received correctly or not. If the frame's signal strength is less than $CSThresh$, the frame is discarded in the PHY module without being visible to the MAC layer. Otherwise, the frame is passed to the MAC, which in turn may discard the frame if its strength is less than $RxThresh$. To combat the propagation effect, the simulation area is chosen such that the received signal strength is always greater than $RxThresh$. In fact, the station may not discard all collided frames but it may accept one of them if its signal strength to the total strength of the others is greater than $CPThresh$; otherwise, the MAC ignores them all. Our simulation settings guarantee that in case of collision, all collided frames will be discarded. On the other hand, *ns* does not consider transmission errors. Thus, we modify the *ns wireless-phy* module to consider transmission errors by using the constant bit error rate model; the frame error probability, p_e, of the received frame is calculated and then used to predict whether the frame is corrupted or not.

In our simulation settings, all nodes run the RTS/CTS access mode of the IEEE 802.11 DCF and they are static and within communication range of each other (single hop communications). The channel data rate is *1Mb/s*. All nodes transmit UDP/IP frames and work in saturation mode. In each simulation-iteration, we generate a random scenario, where sender-receiver pairs are randomly chosen. For low number of station, the probability of collision is usually small and negligible. Therefore, we analyze the performance of the network for large number of stations ($n \geq 10$) to study the impact of collision on the network performance. The simulation time lasts for 200 seconds and then terminated. The simulation results reported in the next section represent the average results over 1200 different scenarios. The results collected are the average values over 30 runs for each simulation setting. Unless otherwise specified, Table 4.1 shows the system parameters used in the simulation and analytical analysis. The system values are those specified for frequency hopping spread spectrum (FHSS) PHY layer [11] and used in [47].

Table 4.1: FHSS systems parameters and additional parameters used to obtain numerical results

Parameter	Value	Parameter	Value
l_P	1KB	SIFS	$28\mu sec.$
BER	10^{-5}	σ	$50\mu sec.$
CW_{min}	16	DIFS	$156\mu sec.$
CW_{max}	1024	EIFS	$460\mu sec.$
H_{rtr}	160 bits	Transmission power	-15dBm
H_{mac}	272 bits	CPThresh	10.0
H_{phy}	192 bits	CSThresh	-232.5 dBm
l_{ack}(bits)	$112 + H_{phy}$	RXThresh	-102.5 dBm
l_{rts}(bits)	$160 + H_{phy}$	Transmission range	107 m
l_{cts}(bits)	$112 + H_{phy}$	Simulation region	67m $\times$ 67m
R_d	1Mbps	R_b	1Mbps
$rts\_timeout$	$300\mu sec.$	$data\_timeout$	$300\mu sec.$
δ	$1\mu sec.$	Simulation time	200 sec

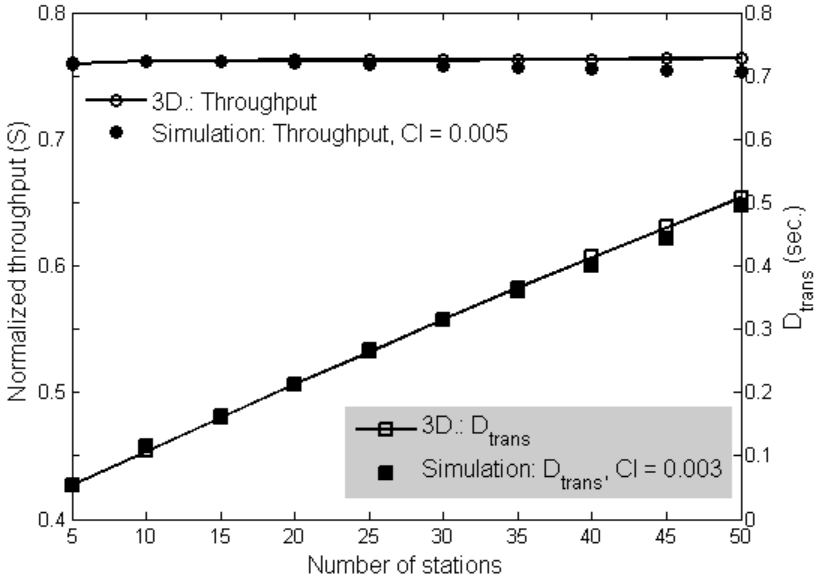

Figure 4.2: Throughput and packet time delay (D_{trans}), $BER = 1 \times 10^{-5}$: analysis versus simulation.

Figures 4.2 and 4.3 show that simulation results fairly match analytical performance results (saturation throughput, packet delay, packet discard time, and packet discard probability). The 95% confidence intervals (CI) of the reported simulation results are shown in the figures.

4.5.2 Performance Investigation

In situations where the wireless channel experiences a high bit error rate and the network is highly-congested, the station will face a large number of collisions before it can acquire the channel. Furthermore, after channel acquisition, the data frame may be discarded due to bit error transmissions. Therefore, to improve the reliability, the retransmission policy should be adopted for data frames as well as for RTS frames. But statistically, to send a single frame properly, the average number of transmission attempts is equivalent to the average number of frames that are sent to have one successful frame transmission. Therefore, we may expect that adding the data

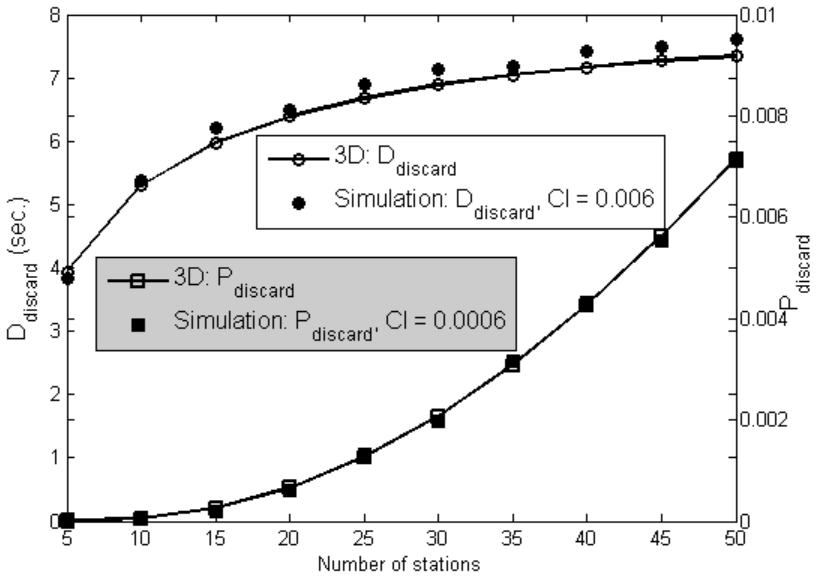

Figure 4.3: Packet discard time ($D_{discard}$) and packet discard probability ($P_{discard}$), $BER = 1 \times 10^{-5}$: analysis versus simulation.

retransmission model will not affect the system throughput calculation.

To validate the necessity for the data retransmission policy, we compare the performance of our 3-D model with the Bianchi [47] and Wu [49] models. Bianchi and Wu did not consider bit transmission errors because they assumed error-free channels. For comparison purposes, we release the ideal channel condition from Bianchi and Wu models by adding the frame error probability to their transition probabilities. As a result, the data frame may be discarded after the first transmission attempt with a probability $(1 - p) \times p_e$.

Figure 4.4 shows that while Bianchi has higher throughput than Wu, the 3-D has the highest throughput. In fact, the 3-D has better throughput because stations work with large CW sizes compared to Bianchi and Wu (at $n = 350$, the throughput is equal to 0.474, 0.456, and 0.385 in 3-D, Bianchi, and Wu models respectively). This is because the station doubles the CW after each unsuccessful attempt to transmit an RTS or data frame. In Bianchi and Wu, the data frame is discarded and the CW is reinitialized with CW_{min} after the first failed data transmission attempt. There-

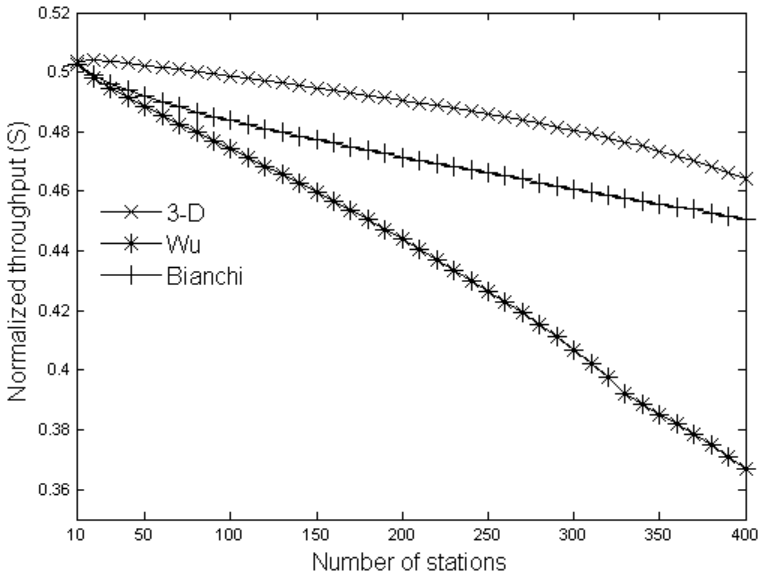

Figure 4.4: The throughput over the three analytical models, $BER = 5 \times 10^{-5}$.

fore, stations in these two models have small CW sizes compared to the 3-D model. In general, a large CW size reduces number of collisions and the adoption of the data retransmission policy enhances the reliability of the channel. Furthermore, the Bianchi model shows better performance than the Wu model because the probability of unsuccessful channel acquisition in the former is negligible due to the unlimited channel reservation attempts. Figures 4.5 and 4.6 show how the error recovery mechanism in the 3-D model can improve the $P_{discard}$ and $D_{discard}$ calculations. Specifically, the pair (R_1, R_2), which represents the RTS and data retry limits, is equal to $(6, 4)$, $(6, 1)$, and $(6, \infty)$ in the 3-D, Wu, and Bianchi respectively. By applying these retry limit values to (4.25), we see that, with $BER = 5 \times 10^{-5}$, the 3-D model has the smallest packet discard probability, the Bianchi model has a constant packet discard probability ($P_{discard} = p_e$), and finally, $P_{discard}$ in the Wu model depends on the p, which increases as the number of stations increases. The smallest packet discard in the 3-D model is attributed to the fact that it has less number of collisions, which makes its p less than that in the other two models. Moreover, since $R_2 > 1$ in the 3-D model, the probability that bit transmission errors corrupt the transmitted data

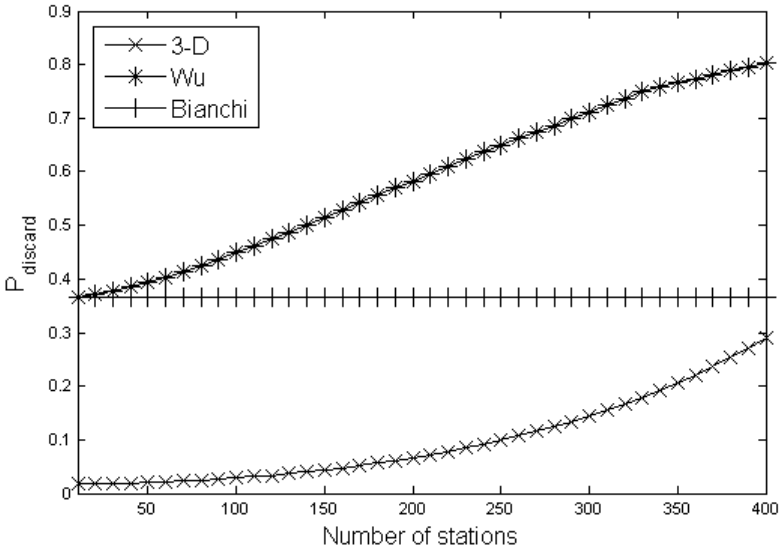

Figure 4.5: The packet discard probability over the three analytical models, $BER = 5 \times 10^{-5}$.

frame is reduced from p_e to $(p_e)^{R_2}$.

Figure 4.6 shows that the 3-D model has the highest $D_{discard}$; this is due to the smallest $P_{discard}$ it experiences, which implies that the packet will stay longer before it is discarded. In the Bianchi model, a frame is discarded only due to transmission errors, but in the Wu model, unsuccessful channel reservation and transmission errors cause packet discard. However, when the number of contending stations is small, both models almost have the same $D_{discard}$ because the main cause of packet discard is bit transmission errors. But when the number of contending stations is large, the Bianchi model has higher $D_{discard}$ because the station spends more time to reserve the channel due to the infinite number of reservation attempts.

The D_{trans} is conditioned on the event that the frame will not be discarded. Figure 4.7 shows that the 3-D model experiences the highest delay because the packet has several transmission attempts ($R_2 > 1$). In the other two models, the packet has only one transmission attempt ($R_2 = 1$) and it experiences a discard if its transmission is failed. On the other hand, $D_{trans}^{Bianchi} > D_{trans}^{Wu}$ because of infinite retry limit in the

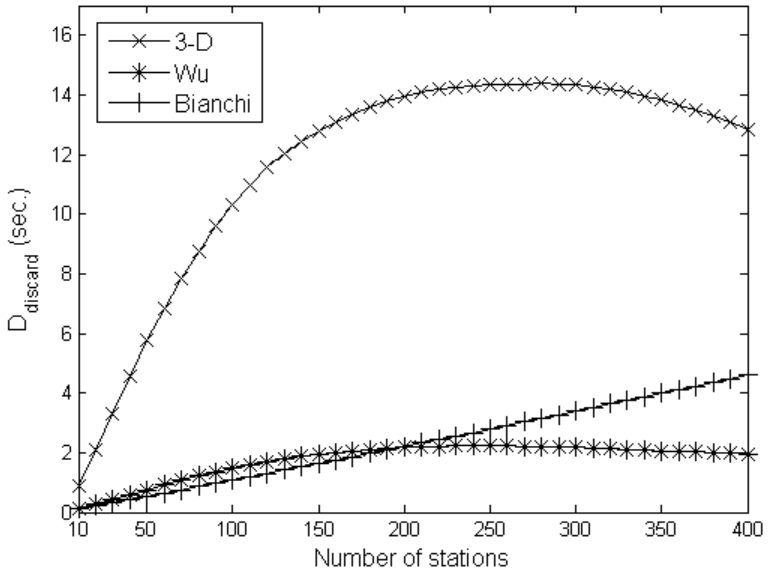

Figure 4.6: The average time to discard a packet over the three analytical models, $BER = 5 \times 10^{-5}$.

Bianchi model.

Equation (4.4) shows p_e is proportionally depends on BER. Further increase in BER not only makes it difficult to send a data frame correctly but also vanishes the possibility of channel acquisition because transmission errors may corrupt control frames, RTS and CTS, even their sizes are small compared to the data frame. We can notice from Figure 4.8 that when $BER > 3 \times 10^{-4}$, the system throughput reaches zero. In cases where $BER \leq 3 \times 10^{-4}$, the possibility of successful data transmission can be increased by sending the corrupted data frame several times instead of discarding it. Moreover, Figure 4.8 shows that the optimal number of transmission attempts is 4. On the other hand, Figure 4.9 plots $P_{discard}$ versus BER. As expected, we notice that when BER increases, $P_{discard}$ increases too. Thus to reduce $P_{discard}$, R_2 should be increased (for $BER > 2 \times 10^{-4}$, $P_{discard} \approx 1$).

Figure 4.10 plots the throughput versus CW_{max} over different R_2 values. The RTS/CTS mode is robust to the number of contending stations such that the throughput is almost similar for large networks as well as small ones. For low CW_{max}, through-

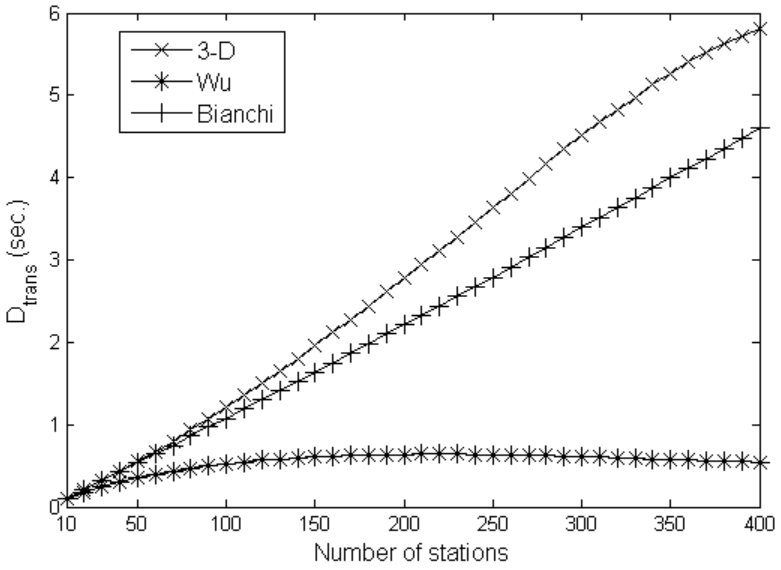

Figure 4.7: The average time to successfully transmit a packet over the three analytical models, $BER = 5 \times 10^{-5}$.

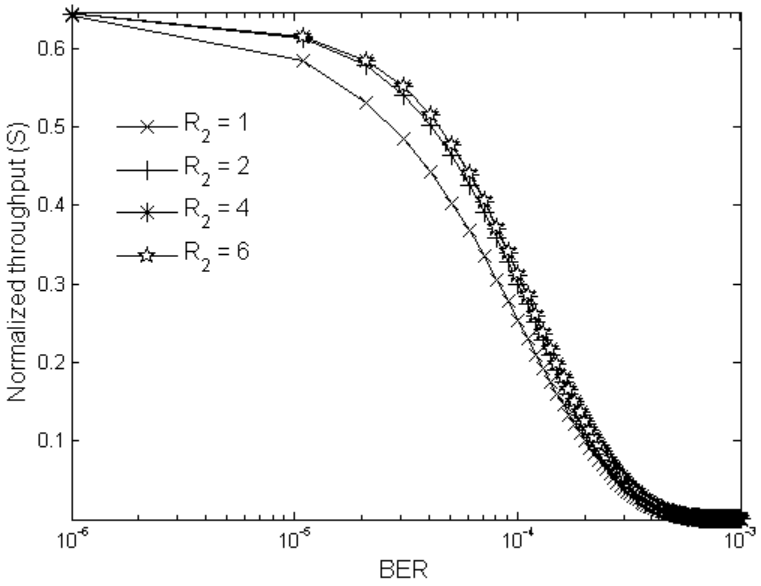

Figure 4.8: The saturated throughput versus bit error rate, $n = 100$ stations.

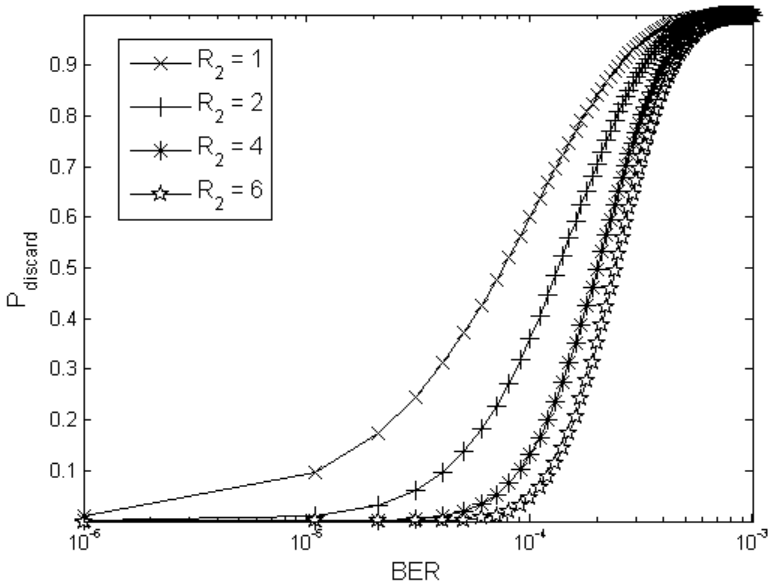

Figure 4.9: The packet discard probability versus BER, $n = 100$ stations.

put drops off due to the high number of collisions. Although, increasing the CW_{max} enhances the system throughput, the maximum achievable throughput is reached at $CW \geq 1024$ regardless of the value of R_2.

4.6 Summary

In this Chapter, we present a new analytical model to analyze the performance of the RTS/CTS access mode of the IEEE 802.11 DCF. In imperfect channel conditions, the stochastic behavior of the DCF algorithm is analytically modeled by a discrete time 3-D Markov chain. The 3-D Markov is modeled as a parallel set of 2-D Markov chains where the number of 2-D chains is equivalent to the number of data retransmission attempts. The saturation throughput, packet drop probability, average packet delay, and average packet drop time are calculated under saturation and infinite transmission buffer conditions. In fact, our analysis show that in situations where the wireless channel experiences a high bit error rate and the network is highly-congested, both RTS and data retry models are necessary to predict the saturation performance of

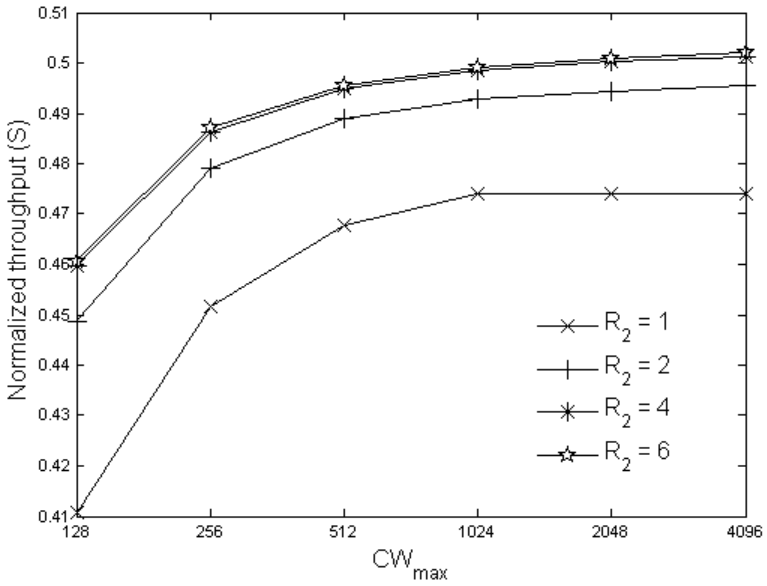

Figure 4.10: The saturated throughput over different R_2, $BER = 5 \times 10^{-5}, n = 100$ stations.

RTS/CTS access mode. To the best of our knowledge, our 3-D model is the first model that deals with this issue. In the next Chapter, the model will be extended to study the non-saturation performance as well.

Chapter 5

Non-Saturation Performance Analysis of the IEEE 802.11 DCF

5.1 Introduction

Recently, a number of analytical models for non-saturation performance analysis of the IEEE 802.11 RTS-CTS access mode have been reported in the literature. Models in [57, 60, 61] extended the saturated discrete-time Markov chain model, which is introduced in [47] and extended in [49]. Specifically, a post backoff states are added to model the non-saturation mode. The post backoff states represent the states where the station resides when it has no data packets to transmit. In [62], a detailed analytical analysis for the queueing behavior of IEEE 802.11 DCF was introduced. In [62] and based on the discrete time $G/G/1$ queue, the throughput and delay are evaluated with the assumptions of a general traffic pattern, an arbitrary number of users, and an infinite transmission buffer size. Although [57, 60, 61, 62] modeled the unsaturated traffic conditions, [57, 60, 61] assumed that the queue length of the MAC layer is zero and [62] assumed an infinite queue length, which are not practical. Furthermore, they cannot provide accurate delay analysis.

In contrary to the 2-D Markov models, the finite capacity of the queue is studied

in [52, 53, 54] using 3-D Markov models to integrate the contention resolution and queueing processes. Specifically, the third dimension is used to model the capacity of the queue. In these models, non-saturation throughput and delay are analyzed with the assumptions of finite retry limit, finite buffer, and finite load.

However, none of the [47, 49, 52, 53, 54, 57, 60, 61, 62] models study the behavior of the RTS-CTS mode in imperfect channel conditions. They either modeled the RTS-CTS in perfect channel conditions, which revokes the necessity of the data retransmission model, or in imperfect channel conditions with a single data transmission attempt.

In this Chapter, we extend our earlier work in Chapter 4 and analyze the performance of IEEE 802.11 in non-saturation conditions using two different approaches. In the first approach, the 3-D Markov process is replaced by a a discrete-time 4-D Markov chain model. The 4D model integrates in addition to the data and control retransmission limits, the finite load, finite buffer capacity, and quality of the received data. Using the assumption of Poisson packet arrival distribution, the non-saturation condition is modeled by adding an additional state, namely the idle state, in which the station resides when its transmission buffer is empty. By exploiting the theoretical framework, we derive the transmission probability. Using transmission probability, we can derive the system throughput, length of the transmission buffer, buffer blocking probability, packet discard probability, packet delay, packet discard time, and packet service time. In the second approach, we simplify the 4D model and use an M/G/1/K queue with independent samples from the saturation analysis to model the MAC transmission buffer capacity instead of using the queue length as a state variable. Analysis results for the two models are similar. Due to its simplicity, the second model will be used in multi-hop performance analysis as we will discuss in Chapter 6.

The rest of the Chapter is organized as follows. In Section 5.2, we introduce the 4D model with its analysis for the RTS/CTS access mode. Section 5.3 introduces the M/G/1/K analysis. In Section 5.4, we discuss the simulation and numerical results.

Finally, Section 5.5 summarizes the Chapter.

5.2 4D Markov Chain Model Analysis

This Section introduces and analyzes the discrete-time 4D Markov chain model for the RTS/CTS access mode. The transmission probability (τ) is derived and used to study the QoS performance of the IEEE 802.11 system.

5.2.1 Assumptions

In addition to the assumptions discussed in Section 4.2.1, we assume that each station has a first-in first-out (FIFO) transmission buffer of finite length (K). Packet arrivals follow a Poisson distribution with a known arrival rate (λ) and during a time interval (t). The probability $\Lambda(k,t)$ of k packet arrivals in a time interval t is:

$$\Lambda(k,t) \quad = (\lambda t)^k \, e^{-\lambda t}/k! \tag{5.1}$$

when the transmission buffer reaches its maximum capacity, the incoming packets are lost.

5.2.2 System Model

Let $s(t)$, $z(t)$, $y(t)$, and $x(t)$ be the stochastic processes that represent at time t the number of packets in the transmission buffer with state space $\{0, 1, \ldots, K\}$, the $slrc$ with state space $\{0, 1, \ldots, R_2 - 1\}$, the $ssrc$ with state space $\{0, 1, \ldots, R_1\}$, and the backoff counter with state space $\{0, 1, \ldots, W_{j,i} - 1\}$, respectively. From (4.8), $W_{j,i}$ represents the CW value when $slrc = j$ and $ssrc = i$.

The process $\{s(t), z(t), y(t), x(t)\}$ is a discrete-time 4-D Markov process with the state space of $idle \cup \{(k, j, i, w)$, where $1 \leq k \leq K$, $0 \leq j \leq R_2 - 1$, $0 \leq i \leq R_1$, and $0 \leq w \leq W_{j,i} - 1$. The state $idle$ is the idle state at which the station resides when it does not have any packet to transmit.

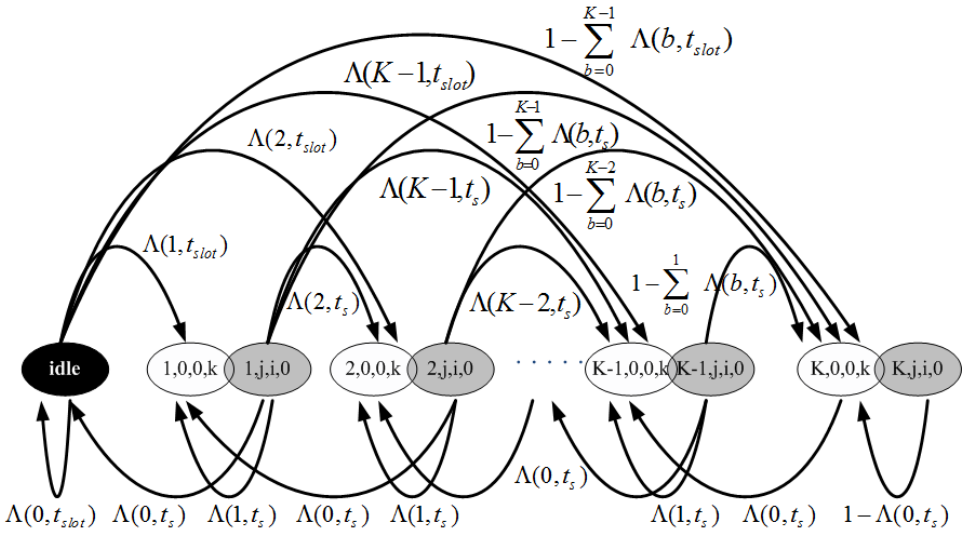

Figure 5.1: Inter-chain transition probabilities

For more clarification, the backoff procedure is modeled by a 3-D Markov chain, which models the backoff procedure of the tagged station when it has a specific number of packets in its transmission buffer. The 3-D chain is a set of distinct 2-D Markov processes. Each 2-D process models one of the data transmission attempts. Finally, the set of 3-D chains forms the 4-D model.

5.2.3 Transition Probabilities

Figure 5.1 shows the inter-chain transition probabilities. Initially, the tagged station resides in the idle state (*idle*) because it has no data packets to transmit. As long as no new packets arrive from the upper layer, the station remains in the state *idle* with a probability $\Lambda(0, t_{slot})$.

$$P\left[idle\,|idle\right] \quad = \Lambda(0, t_{slot}) \tag{5.2}$$

from (4.23), t_{slot} is the average length of a slot time in RTS/CTS mode.

If k packets ($1 \leq b < K$) arrive in the transmission buffer during t_{slot}, the station shall transit to the k^{th} backoff chain with a probability $\Lambda(k, t_{slot})$. In the k^{th} chain,

the tagged station shall uniformly and randomly select the backoff state $(k, 0, 0, w)$, where $0 \leq w \leq (W_{0,0} - 1)$.

$$P\left[(k, 0, 0, w) \,|idle\right] \;\; = \Lambda(k, t_{slot})/W_{0,0} \tag{5.3}$$

If $k \geq K$, the station shall transit to the K^{th} chain with a probability $1 - \sum_{c=0}^{K-1} \Lambda(c, t_{slot})$.

$$P\left[(K, 0, 0, w) \,|idle\right] \;\; = \Big(1 - \sum_{c=0}^{K-1} \Lambda(c, t_{slot})\Big)/W_{0,0} \tag{5.4}$$

As the tagged station enters the k^{th} chain, it starts the backoff procedure to transmit the next head-of-queue packet. The backoff procedure continues until the packet either transmitted successfully or discarded. During the packet's service time (t_s), which is the time to transmit the packet successfully or to discard it, the tagged station counts the number of new packet arrivals (b) and based on this number it shall transit to the chain $k - 1$, k, or $k + b$, with a probability $\Lambda(0, t_s)$, $\Lambda(1, t_s)$, or $\Lambda(b + 1, t_s)$ respectively (t_s is the packet mean service time and it will be discussed and derived in the end of this Section).

$$P[(k - 1, 0, 0, w) \,|(k, j, i, 0)] \;\; = \Lambda(0, t_s)/W_{0,0} \tag{5.5}$$

$$P[(k, 0, 0, w) \,|(k, j, i, 0)] \;\; = \Lambda(1, t_s)/W_{0,0} \tag{5.6}$$

$$P[(k + b, 0, 0, w) \,|(k, j, i, 0)] \;\; = \Lambda(b + 1, t_s)/W_{0,0} \tag{5.7}$$

If the number of new arrivals in addition to the ones in the transmission buffer is $\geq K$, the tagged stations shall transit to the K^{th} chain with a probability $1 - \sum_{c=0}^{K-k} \Lambda(c, t_s)$.

$$P[(K, 0, 0, w) \,|(k, j, i, 0)] \;\; = \Big(1 - \sum_{c=0}^{K-k} \Lambda(c, t_s)\Big)/W_{0,0} \tag{5.8}$$

If the tagged station is in the first chain, it shall transit back to the state *idle* with a probability $\Lambda(0, t_s)$ if no packets arrive during t_s.

$$P[idle \,|(1, j, i, 0)] = \Lambda(0, t_s) \tag{5.9}$$

Figure 5.2 shows the 2-D Markov model for the j^{th} data transmission attempt $(slrc = j)$ and the number of packets in the queue is k. The first row, indexed

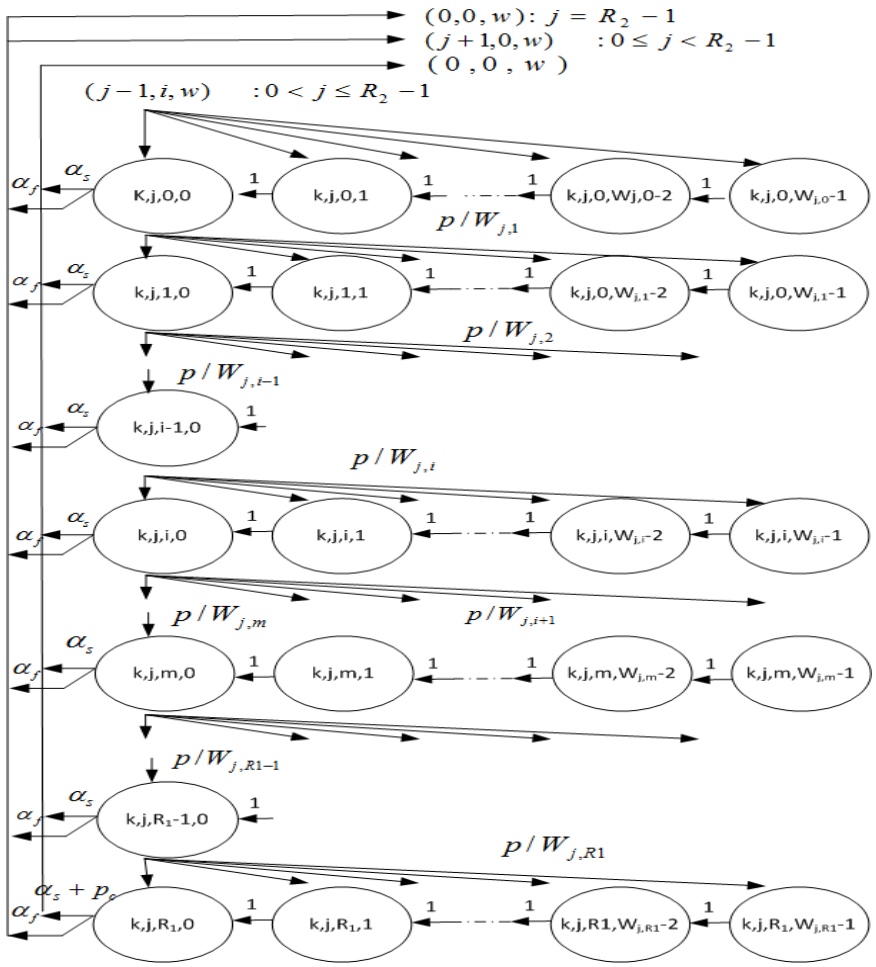

Figure 5.2: The 2-D Markov chain when $slrc = j$ and k packets are in the transmission buffer

from $(k, j, 0, 0)$ to $(k, j, 0, W_{j,0})$, is the stage-*0* backoff states, which represents the first RTS transmission attempt ($ssrc = 0$). The second row of states, indexed from $(k, j, 1, 0)$ to $(k, j, 1, W_{j,1}-1)$ is the stage-*1* backoff states, which represents the second RTS retransmission attempt ($ssrc = 1$). The last row, indexed from $(k, j, R_1, 0)$ to $(k, j, R_1, W_{j,R_1} - 1)$, is the stage-$(R_1+1)$ backoff states, which represents the last RTS retransmission attempt ($ssrc = R_1$).

Consider a packet transmission event at the tagged station, where the tagged station starts the initial backoff by randomly and uniformly choosing one of the stage-*0* backoff states, $(k, 0, 0, w)$. At state $(k, 0, 0, w)$, the station decrements the backoff

counter and transits to the next state $(k, 0, 0, w-1)$ with a probability 1 if the channel remains idle for a single t_{slot} time. The backoff countdown process continues until it reaches the state $(k, 0, 0, 0)$. In general, for $0 \leq j \leq R_2 - 1$, $0 \leq i \leq R_1$, and $0 \leq w \leq W_{j,i} - 1$, the bachoff countdown process can be expressed as:

$$P[(k, j, i, w) \,|(k, j, i, w + 1)] = 1 \tag{5.10}$$

When the station reaches state $(k, j, i, 0)$, it shall transmit an RTS packet. The RTS transmission may success or fail with probability $(1-p)$ or p, respectively. If the RTS packet transmission fails, the station shall double the CW, increment the $ssrc$, and transit the backoff process randomly and uniformly to a state $(k, j, i + 1, w)$ for another channel reservation attempt.

$$P[(k, j, i + 1, w) \,|(k, j, i, 0)] = p/W_{j,i+1} \tag{5.11}$$

At state $(k, j, R_1, 0)$, there is no more RTS transmission attempts. If the station cannot successfully reserve the channel, it shall discard the data packet and transit the backoff process to the state $(k + b - 1, 0, 0, w)$ in order to transmit the next head-of-queue packet, where $b \geq 0$ represents the number of packet arrivals (if $k+b-1 = 0$, the station transits back to the idle state). When the station drops a data packet, it shall initialize the $ssrc$, $slrc$, and CW.

$$P[(k + b - 1, 0, 0, w) \,|(k, j, R_1, 0)] = p\Lambda(b, t_s)/W_{0,0} \tag{5.12}$$

Otherwise, if the station reserves the channel (that is, the tagged station receives the CTS correctly), it shall reset the $ssrc$ and transmit the data packet. The transmitted data packet may be corrupted or sent correctly with probability p_e or $(1 - p_e)$ respectively, p_e is given in (4.4). If the packet is corrupted, the station shall increment the $slrc$, double the CW, and transit to the state $(k, j + 1, 0, w)$ for a new transmission attempt.

$$P[(k, j + 1, 0, w) \,|(k, j, i, 0)] = \alpha_f/W_{j+1,0} \tag{5.13}$$

from (4.5), α_f is the packet transmission failure probability per transmission attempt.

The station continues transmitting the data packet after each failed attempt until the number of attempts is exhausted ($j > R_2 - 1$). After that, the station shall discard the data packet and initialize the $slrc$ and CW. Next, the station shall start a new backoff process to transmit the next head-of-queue data packet.

$$P[(k+b-1,0,0,w)\,|(k,R_2-1,i,0)] = \alpha_f \Lambda(b,t_s)/W_{0,0} \tag{5.14}$$

However, if the data packet has been sent properly at any attempt (that is, the station receives the ACK correctly), the station shall initialize the $slrc$ and CW. Next, the station shall start a new backoff process to transmit the next head-of-queue data packet.

$$P[(k+b-1,0,0,w)\,|(k,j,i,0)] = \alpha_s \Lambda(b,t_s)/W_{0,0} \tag{5.15}$$

from (4.5), α_s is the packet success transmission probability per transmission attempt.

5.2.4 Transmission Probability

The non-null transition probabilities (5.2)-(5.15) can be expressed in a highly reduced mathematical form in terms of S_{idle} (S_{idle} is the probability that the backoff process is being in the state $idle$) and the two conditional probabilities p and α_f. Let $\gamma = \alpha_f \sum_{i=0}^{R_1} p^i$, then $S_{h,j,i,0}$ (the probability that the backoff process is being in the state $(k,j,i,0)$) is equal to:

$$S_{k,j,i,0} \quad = S_{k,0,0,0}\gamma^j p^i \tag{5.16}$$

Equation (5.7) shows that the probability of transiting from the state $(k,j,i,0)$ to the state $(k+b,0,0,w)$ depends on the number of new packet arrivals (b). Given that the backoff process leaves the current chain regardless if the packet is successfully transmitted or lost, (5.7) can be rewritten as:

$$S_{k+b,0,0,w} \quad = \frac{\Lambda(b+1,t_s)}{W_{0,0}} \left(\alpha_s \sum_{j=0}^{R_2-1} \sum_{i=0}^{R_1} S_{k,j,i,0} + p \sum_{j=0}^{R_2-1} S_{k,j,R_1,0} + \alpha_f \sum_{i=0}^{R_1} S_{k,R_2-1,i,0} \right) \tag{5.17}$$

By plugging (5.16) into (5.17),

$$S_{k+b,0,0,w} = \frac{\Lambda(b+1,t_s)}{W_{0,0}} S_{k,0,0,0} \tag{5.18}$$

Now by means of (5.10) and(5.18), for $1 \leq k \leq K$ and $0 \leq w \leq W_{0,0} - 2$, $S_{k,0,0,w}$ can be expressed as follows:

$$S_{k,0,0,w} = S_{k,0,0,w+1} + \Lambda(k, t_{slot}) S_{idle} + \sum_{b=1}^{k+1} \Lambda(k-b+1, t_s) S_{b,0,0,0} \tag{5.19}$$

Based on the chain regularities, $S(k,0,0,w)$ can be expressed as:

$$S_{k,0,0,w} = \frac{(W_{0,0}-w)}{W_{0,0}} \left(\Lambda(k, t_{slot}) S_{idle} + \sum_{b=1}^{k+1} \Lambda(k-b+1, t_s) S_{b,0,0,0} \right) \tag{5.20}$$

For $w = 0$, equation (5.20) can be written as:

$$S_{k,0,0,0} = \Lambda(k, t_{slot}) S_{idle} + \sum_{b=1}^{k+1} \Lambda(k-b+1, t_s) S_{b,0,0,0} \tag{5.21}$$

Using (5.21), let us define a $K \times K$ upper triangular matrix $\mathbf{T}$, called the state probabilities matrix, where $\mathbf{T}_{i,j}$ is the transition probability from state i to state j and K is the transmission buffer capacity. We can express $S_{k,0,0,0}$ in terms of S_{idle} as shown in algorithm (1).

Now, based on the fact that transmission is only allowed when the backoff timer value reaches zero, the transmission probability τ (the probability that the station transmits in a randomly chosen slot time) is:

$$\begin{aligned} \tau &= \sum_{k=1}^{K} \sum_{j=0}^{R_2-1} \sum_{i=0}^{R_1} S(k,j,i,0) \\ &= \sum_{k=1}^{K} \mathbf{U}[k+1] \sum_{j=0}^{R_2-1} \gamma^j \sum_{i=0}^{R_1} p^i \end{aligned} \tag{5.22}$$

S_{idle} can be determined by imposing the normalization condition:

$$\begin{aligned} 1 &= S_{idle} + \sum_{k=1}^{K} \sum_{j=0}^{R_2-1} \sum_{i=0}^{R_1} \sum_{w=0}^{W_{j,i}-1} S_{k,j,i,w} \\ 1 &= S_{idle} + 0.5 \sum_{k=1}^{K} \mathbf{U}[k+1] \sum_{j=0}^{R_2-1} \gamma^j \sum_{i=0}^{R_1} p^i (W_{j,i}+1) \end{aligned} \tag{5.23}$$

Algorithm 1 Discrete-time Markov chain state probabilities

1: Let $\mathbf{I}$ be an $K \times K$ identity matrix.
2: Let $\mathbf{U}$ be a state probability vector of the discrete-time Markov chain with $K+1$ probability states $[S_{idle}, S_{1,0,0,0}, \ldots, , S_{K,0,0,0}]$ initialized with zeros.
3: Let t_s represent the packet service time and $\Lambda(0, t_s)$ represent the number of zero packet arrivals during t_s
4: $\mathbf{U}(1)$=1
5: **for** $k = 1$ to K **do**
6: **for** $i = 1$ to k **do**
7: $\mathbf{U}_{k+1} = \mathbf{U}_{k+1} + \mathbf{(I\text{-}M)}_{i,k} \times \mathbf{U}_i$
8: **end for**
9: $\mathbf{U}_{k+1} = \mathbf{U}_{k+1}/\Lambda(0, t_s)$
10: **end for**
11: $\mathbf{U} = S_{idle} \times \mathbf{U}$

Now, p from (4.4) can be inverted to express $\tau^*(p)$ as:

$$\tau^*(p) \quad = 1 - (1-p)^{\frac{1}{n-1}} \tag{5.24}$$

Equations (5.22) and (5.24) represent a nonlinear system of equations with two unknowns p and τ, which can be solved using numerical techniques.

5.2.5 Normalized Throughput

The normalized throughput, S, is defined as the fraction of time the channel is used to successfully transmit useful payload bits (l_P) during any chosen slot time (t_{slot}).

$$S \quad = \frac{\frac{l_p}{R_d} P_s}{t_{slot}} \tag{5.25}$$

from (4.21) P_s, is the probability of successful transmission in a chosen slot time. R_d is the data transmission rate.

5.2.6 Buffer Length

In the 4-D model, the transmission buffer length is represented by a stochastic process $s(t)$ with a state space $\{0, 1, \ldots, K\}$ that represents the number of buffered packets at time t. Assume the probability that the backoff process is being in the chain k is denoted by S_k, where k indicates that there are k packets in the transmission buffer, then the average length of the transmission buffer ($\overline{B}$) can be expressed as:

$$\overline{B} = \sum_{k=1}^{K} k S_k \tag{5.26}$$

The probability S_k is the sum of the backoff state probabilities over the chain k ,

$$\begin{aligned} S_k &= \sum_{j=0}^{R_2-1} \sum_{i=0}^{R_1} \sum_{w=0}^{W_{j,i}-1} S_{k,j,i,w} \\ &= 0.5\mathbf{U}\,[k+1] \sum_{j=0}^{R_2-1} \gamma^j \sum_{i=0}^{R_1} p^i (W_{j,i}+1) \end{aligned} \tag{5.27}$$

5.2.7 Buffer Blocking Probability

The buffer blocking probability (P_{block}) is the probability that the arriving packet will not join the transmission buffer when the buffer reaches its maximum capacity. If the backoff process resides in the state *idle* and more than K packets arrive during t_{slot}, then only K packets enter the buffer and the remaining ones are lost. Regardless whether the next head-of-queue packet is successfully transmitted or dropped, if the backoff process resides at chain $1 \leq k \leq K$ and $b > K-k+1$ packets arrive during t_s, then only $K-k+1$ packets enter the buffer and the remaining ones are lost. Hence, P_{block} can be expressed as:

$$P_{block} = \left(1 - \sum_{k=0}^{K} \Lambda(k, t_{slot})\right) S_{idle} + \sum_{k=1}^{K} \left(1 - \sum_{b=0}^{K-k+1} \Lambda(b, t_s)\right) S_k \tag{5.28}$$

5.2.8 Packet Discard Probability

The data packet is discarded from the transmission buffer if the station fails to reserve the channel in $R_1 + 1$ consecutive attempts or it fails to successfully send the data

packet in R_2 attempts. This implies that for $1 \leq k \leq K$, $0 \leq j \leq R_2 - 1$, and $0 \leq i \leq R_1$, the packet may be discarded at the backoff state $(k, j, R_1, 0)$ with a probability p or it may be discarded at the backoff state $(k, R_2 - 1, i, 0)$ with a probability α_f. Given that all backoff chains have the same packet discard probability $P_{discard}$ and is conditioned on the event that there is a packet under transmission, $P_{discard}$ can be expressed as:

$$\begin{aligned} P_{discard} &= \sum_{k=1}^{K} \left(p \sum_{j=0}^{R_2-1} S(k, R_1, j, 0) + \alpha_f \sum_{i=0}^{R_1} S(k, R_2 - 1, i, 0) \right) \\ &= p^{R_1+1} \sum_{j=0}^{R_2-1} \gamma^j + \gamma^{R_2} \end{aligned} \tag{5.29}$$

5.2.9 Packet Delay

The average packet delay (D_{succ}) for a packet that is successfully transmitted is the sum of the queueing delay (D_{queue}) and the transmission delay (D_{trans}). The queueing delay is the total time from the instance the packet joined the transmission buffer until it reaches the head-of-queue. The transmission time is the time duration from the instant that the packet started the backoff process to the successful transmission instant, this time includes all retransmission attempts and it is conditioned on the event that the packet will not be discarded $(1 - P_{discard})$.

$$D_{succ} = D_{queue} + D_{trans} \tag{5.30}$$

5.2.9.1 The Queueing Delay

The transmission buffer can be modeled as a FIFO M/G/1/K queueing system. The queueing delay for a randomly arriving packet depends on the average number of packets in the buffer ($\overline{B}$) that the packet finds and the average packet's service time (t_s). Thus,

$$D_{queue} = \overline{B} t_s \tag{5.31}$$

5.2.9.2 Transmission Delay

To find D_{trans}, we need to find the average number of slots ($E[X]$) that are required to transmit the data packet successfully. For $1 \leq k \leq K$, $0 \leq j \leq R_2 - 1$, and $0 \leq i \leq R_1$, the packet may be transmitted successfully from any state $(k, j, i, 0)$. Conditioned on the events that there is at least one packet in the transmission buffer and that packet will not be dropped, $E[X]$ is the probability that the packet succeeds in state $(k, j, i, 0)$ times the average number of slots that are required to reach that state. Thus,

$$D_{trans} = E[X]\, t_{slot} \tag{5.32}$$

$$E[X] = \left(\alpha_s \sum_{j=0}^{R_2-1} \gamma^j \sum_{i=0}^{R_1} p^i \overline{NS}_{j,i}\right)/(1 - P_{discard}) \tag{5.33}$$

where $\overline{NS}_{j,i}$ is the average number of slots that are required to reach that state, which is equal to,

$$\overline{NS}_{j,i} = \sum_{k_1=0}^{j} \sum_{k_2=0}^{i} \frac{W_{k_1,k_2} + 1}{2} \tag{5.34}$$

5.2.10 Packet Discard Delay

The total packet discard delay (D_{fail}) for a packet that is discarded from the system due to unsuccessful transmission is the sum of the queueing delay (D_{queue}) and the average discard delay ($D_{discard}$). The average discard delay is the time duration from the instant that the packet started the backoff process to the packet discard instant, this time includes all retransmission attempts and it is conditioned on the event that the packet will be discarded.

$$D_{fail} = D_{queue} + D_{drop} \tag{5.35}$$

The packet is discarded if the *ssrc* or *slrc* exceeds R_1 or $R_2 - 1$ limit, respectively. Thus, for $1 \leq k \leq K$, $0 \leq j \leq R_2 - 1$, and $0 \leq i \leq R_1$, the packet may be discarded from the state $(k, j, R_1, 0)$ with a probability p or from the state $(k, R_2 - 1, i, 0)$ with a probability α_f. To find $D_{discard}$, we need to find the average number of slots ($E[Y]$)

that are required to discard the packet. Thus, conditioned on the events that there is at least one packet in the transmission buffer and that packet will be discarded, $E[Y]$ is the probability that the packet is discarded at state $(k, j, R_1, 0)$ or state $(k, R_2 - 1, i, 0)$ times the average number of slots that are required to reach that state. Thus,

$$D_{discard} = E[Y]\, t_{slot} \tag{5.36}$$

$$E[Y] = \left(p_r^{R_1+1} \sum_{j=0}^{R_2-1} \gamma^j \overline{NS}_{j,R_1} + \alpha_f \gamma^{R_2-1} \sum_{i=0}^{R_1} p^i \overline{NS}_{R_2-1,i} \right) / P_{discard} \tag{5.37}$$

5.2.11 Packet Service Time

The packet service time (t_s) is the duration from the time instant that the packet starts the backoff procedure to the successful transmission or packet lost instant including all retransmission attempts. Since there are two possibilities for any packet transmission: discarded with a probability $P_{discard}$ or transmitted successfully with a probability $(1 - P_{discard})$, then t_s can be expressed as:

$$t_s = (1 - P_{discard}) \times D_{trans} + P_{discard} \times D_{discard} \tag{5.38}$$

5.3 M/G/1/K Queueing Model

Here, we simplify the analytical analysis for the non-saturated finite buffer capacity DCF using a finite capacity M/G/1/K queue with multiple vacations model. This service model provides an exhaustive service, as the station cannot go for vacation (the idle state) until all packets present in the queue have been served. Moreover, the station, on returning from the idle state, either returns back to the idle state if it finds the queue is still empty or it resumes normal service if it finds one or more packets waiting in the queue. Packet arrivals follow Poisson distribution with parameter λt, where λ is the station's packet arrival rate. The interarrival time t could be considered as packet mean service time ($\overline{X}$) or the mean vacation interval ($\overline{V} = \frac{1}{\lambda} + \overline{X}$). From

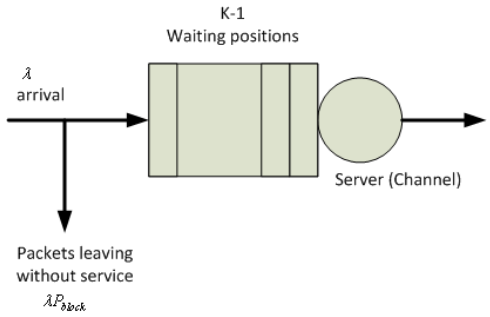

Figure 5.3: A finite capacity single server M/G/1/K queue

the non-saturation analysis, $\overline{X}$ is the packet mean service time. Using (4.25), (4.26), and (4.29), the packet mean service time equals,

$$\overline{X} \quad = (1 - P_{discard}) \times D_{trans} + P_{discard} \times D_{discard} \tag{5.39}$$

Mean service and vacation times are generally distributed and i.i.d. random variables and stations have equal arrival rates λ. The embedded point is chosen to be the point at which a packet has been completely served or a vacation has been ended. The state at a certain embedded point is represented by the number of packets in the station (waiting and in-service) just after that embedded point, see Figure 5.3. Let q_k, $k = 0, 1, \ldots, K$ be the probability of being at state k just after the embedded point, where K is the maximum buffer capacity. Also, Let f_i, $i = 0, 1, \ldots, \infty$ be the probability of i packet arrivals during the vacation interval, which is given by

$$f_i \quad = \frac{(\lambda \overline{V})^i}{i!} e^{-\lambda \overline{V}} \tag{5.40}$$

Let r_i, $i = 0, 1, \ldots, \infty$ be similarly defined as the probability of i packet arrivals during the service time, which is given by

$$r_i \quad = \frac{(\lambda \overline{X})^i}{i!} e^{-\lambda \overline{X}} \tag{5.41}$$

Considering the station just after the embedded points, the following transition equations can be written as

$$q_k \quad = \begin{cases} q_0 f_k + \sum_{i=1}^{k+1} q_i r_{k-i+1} & 0 \leq k < K \\ q_0 \sum_{i=K}^{\infty} f_i + \sum_{s=1}^{K-1} q_s \sum_{i=K-s+1}^{\infty} r_i & k = K \end{cases} \tag{5.42}$$

By summing probabilities of all possible states, we get,

$$\sum_{k=0}^{K} q_k \quad = 1 \tag{5.43}$$

Using (5.42) and (5.43) along with the appropriate calculated values of f_i and r_i, $i = 0, 1, \ldots, \infty$, we can solve for q_k, $k = 0, 1, \ldots, K$. The mean of the time interval between successive embedded points, i.e. departure instants, is given by,

$$T_{em} \quad = \overline{V} q_0 + \overline{X}(1 - q_0) \tag{5.44}$$

where $\overline{V} = \frac{1}{\lambda} + \overline{X}$ is the mean vacation time when the queue is empty at the previous departure instant and $\overline{X}$ is the mean service time when the queue is non-empty at the previous departure instant.

5.3.1 Non-saturation Service Time

For the non-saturation mode the new mean service time $\overline{t_s}$ can be expressed in terms of the saturation mean service time t_s and the idle state probability q_0, so that $\overline{t_s}$ is equal to

$$\overline{t_s} \quad = (1 - q_0)\overline{X} \tag{5.45}$$

5.3.2 Blocking Probability

The blocking probability (P_{block}) is the probability that the arriving packet will not join the transmission buffer because the buffer reaches its maximum capacity. Let ρ_c be the carried load, which is the probability that the server is busy at an arbitrary time, then ρ_c can be expressed as

$$\rho_c \quad = \frac{(1 - q_0)\overline{X}}{q_0\overline{V} + (1 - q_0)\overline{X}} \tag{5.46}$$

The total offered load ρ is defined as

$$\rho \quad = \lambda\overline{X} \tag{5.47}$$

Now using (5.46) and (5.47), the blocking probability P_{block} is equal to

$$P_{block} \quad = \frac{\rho - \rho_c}{\rho} \tag{5.48}$$

5.3.3 Throughput

Since a fraction P_{block} of the arrived packets will be blocked, the normalized throughput (S) of the system can be given by

$$S = \Lambda(1 - P_B)P_{succ}/R \tag{5.49}$$

where R is the channel bit rate and Λ is the total offered load in the network, which is considered as the aggregate of the individual loads offered by all stations. Assume stations have equal arrival rates and they generate fixed length data packets. Let l_p be the useful packet payload size, then Λ is equal to

$$\Lambda = n\lambda l_p \quad \text{bits/second} \tag{5.50}$$

5.3.4 Packet Loss Probability

Packet losses are caused by collisions, transmission errors, or buffer overflow. In CSMA-CA, collision occurs when two or more stations transmit at the same time. While transmission errors may corrupt packet transmission and cause packet losses, buffer flow losses are related to the buffer size. IEEE 802.11 DCF resolves collisions and transmission errors problems by means of retransmissions. Since, the throughput represents the time average traffic transmitted successfully by that station, then P_{loss} can be expressed in terms of the throughput and the total arrival rate as follows,

$$P_{loss} = \frac{\Lambda - S}{\Lambda} \tag{5.51}$$

5.3.5 Discard Probability

As defined by the IEEE 802.11 standard [11] for RTS/CTS access mode, a packet is discarded after R_1 retries of the RTS transmission without receiving a CTS or after R_2 retries of data transmission, preceded by an RTS/CTS handshake, without receiving an ACK. Given that the source of packet losses is either packet blocking or

discarding, then packet discard probability is,

$$P_{discard} = (P_{loss} - P_{block})/(1 - P_{block}) \tag{5.52}$$

5.3.6 Packet Delay

In multi-hop network, the delay for a successfully transmitted UDP packet is defined as the difference between the time a packet arrives at the node without being blocked and the time the packet is successfully received by the final destination node. This time is the sum of queueing and transmission delays of non-blocked and non-discarded packet.

5.3.6.1 Queueing Delay

The queueing delay is defined as the difference between the time a non-blocked packet arrives at the node and the time the packet reaches the head-of-queue and starts transmission. This time mainly depends on whether the queue is empty or not empty at the instant of arrival and the average queue length. The mean number of $\overline{B}$ of packets in the system can be determined by means of the M/G/1/K analysis. Using (5.42), the average queue length can be expressed as,

$$\overline{B} = \sum_{k=0}^{K} k.q_k \tag{5.53}$$

Further, the mean time D_{queue} spent in the queue by a packet that is not blocked is given by

$$D_{queue} = (1 - P_{block}) \left[q_0 \times 0 + (1 - q_0) \times \overline{B} \times \overline{X} \right] \tag{5.54}$$

5.3.6.2 Transmission Delay

The transmission delay at each node is the time delay measured from the moment that the packet reaches the head of the queue to the time that sender receives an ACK confirming its successful reception. It mainly consists of three parts: the time

to successfully transmit the packet, the backoff time, and the retransmission time. It equals,

$$D_{trans} = E[X] \times t_{slot} \tag{5.55}$$

where $E[X]$ is the expected number of slots to successfully transmit a packet is given in (4.27).

Now, the average packet delay equals,

$$D_{succ} = D_{trans} + D_{queue} \tag{5.56}$$

5.4 Performance Evaluation and Verification

5.4.1 Verification

We implemented a simulation model using the network simulator *ns* [59] to validate the results obtained from the analytical model. The free space propagation model is used to predict the signal power received by the receiver. The signal strength is used to determine if the packet is received successfully or not. Here, we use the same simulation settings from Table 4.1 to verify our analytical analysis. In our simulation, all nodes run the RTS-CTS access mode of the IEEE 802.11 DCF and they are static and within the communication range of each other (single hop communications). The channel data rate is *1Mb/s*. All nodes transmit UDP/IP packets and work in the non-saturation mode. In each simulation-iteration, we generate a random scenario, where sender-receiver pairs are randomly chosen. The simulation time lasts for 200 seconds and then terminated. Unless otherwise specified Table 6.1 shows the system parameters used in the simulation and analytical analysis. Most of the reported values are similar to the ones used in [47].

Figures 5.4 - 5.9, provide simulation as well as analytical performance results. Simulation results indicate that our analytical results are fairly accurate.

The non-saturation packet service time ($\overline{t_s}$) given by (5.45) is a function of the packet arrival rate (λ) and number of stations (n). It is composed from the time

needed to drop the packet ($D_{discard}$) and the time needed to successfully transmit the packet (D_{trans}) times the probability of non-empty queue. Thus, the higher λ is, the longer $\overline{t_s}$ is. Also, the greater the number n of the contending stations is, the longer $\overline{t_s}$ is. Figure 5.4 that plots $\overline{t_s}$ versus λ for different network sizes, we noticed that as λ increases, $\overline{t_s}$ increases too. Then, $\overline{t_s}$ becomes constant at the point where $q_0 \approx 0$, which means that stations become saturated. Before the saturation point, $\overline{t_s}$ is very small (< 0.05 second) but when the station approaches the saturation, $\overline{t_s}$ suddenly increases and then remains constant. The sudden increase in $\overline{t_s}$ is attributed to the fact that at saturation, all stations have packet to transmit, thus they always contend on the channel and consequently the number of collisions is increased. Moreover, Figure 5.4 shows that as the number of stations increases, $\overline{t_s}$ increases and the station becomes early saturated, this can be attributed to the increase in the number of collisions. Therefore, not only the number of channel acquisition attempts is increased but also that packet may be discarded from the transmission buffer if the $ssrc$ exceeds R_1. Since the bit error rate is relatively small ($BER = 1.0 \times 10^{-5}$), the main reason for packet discard is collision rather than the bit transmission errors.

Figure 5.5 shows that as long as the station is non-saturated, the throughput (S) is increased linearly with increasing (λ). Moreover, we notice that as the number n of stations increases, not only the throughput is early saturated but also the maximum throughput is reduced. Actually, the increase in n increases the number of collisions and consequently the time needed to acquire the channel is increased, thereby, the packet service time ($\overline{t_s}$) is increased. Large $\overline{t_s}$ value means that more packets enter the transmission buffers, and this makes the system reaches saturation early. Figure 5.6 shows that at saturation points, the transmission buffers quickly build up until they reach their maximum capacities. Additionally, analysis shows an existence of an effective maximum throughput that can be reached at $\lambda \approx 100/n$.

The blocking probability (P_{block}) depends on $\overline{t_s}$. Specifically, the number of packets that each station can serve is ($1/\overline{t_s}$) packets/second. Thus, as long as $\lambda \leq 1/\overline{t_s}$), $P_{block} \approx 0$. As $\lambda \geq (1/\overline{t_s})$, packet arrival rate exceeds packet service rate. Con-

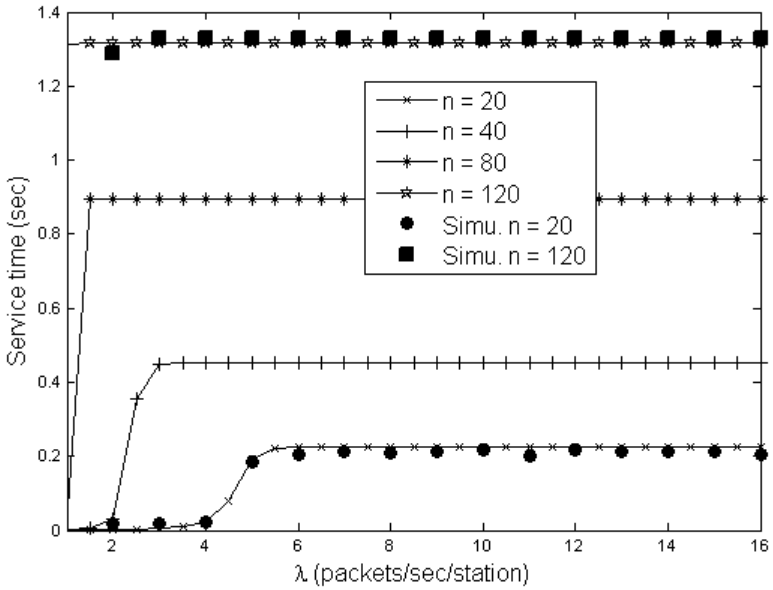

Figure 5.4: Packet service time ($\overline{t_s}$) for different number of stations (n), $K = 16$, $R_1 = 6$, $R_2 = 4$, and $BER = 1 \times 10^{-5}$.

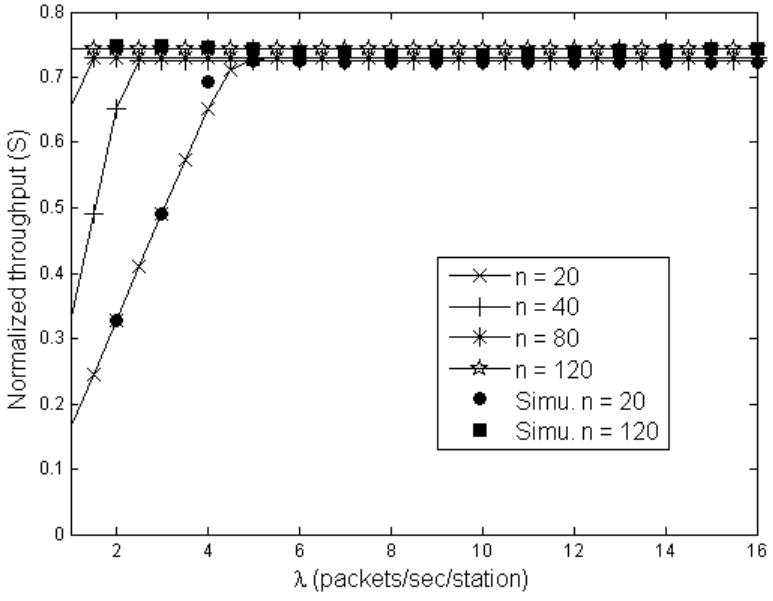

Figure 5.5: Normalized throughput (S) for different number of stations, $K = 16$, $R_1 = 6$, $R_2 = 4$, and $BER = 1 \times 10^{-5}$.

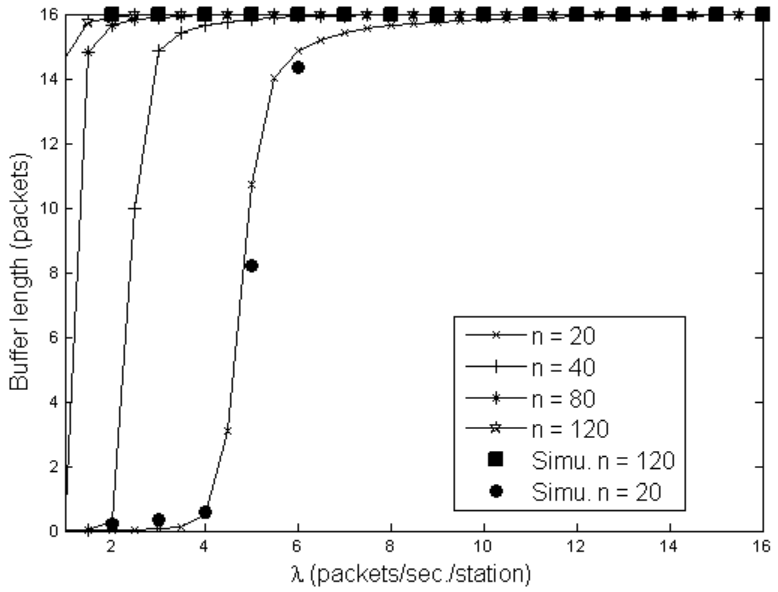

Figure 5.6: Buffer length ($\overline{B}$) for different number of stations, $K = 16$, $R_1 = 6$, $R_2 = 4$, and $BER = 1 \times 10^{-5}$.

sequently, packets start accumulating in the transmission buffers. When the buffer reaches its maximum capacity ($K = 16$), the station blocks the new arrivals. Figure 5.7 shows that for large n, the P_{block} is higher compared to the cases with a small number of stations. In fact, when n is large, $\overline{t_s}$ is large due to the large number of collisions as shown in the Figure 5.4.

The packet delay (D_{succ}) given by (5.30) is composed of the time spent in the buffer before a target packet reaches the head-of-queue and the time needed to transmit the packet successfully including all retransmission attempts. The queueing delay depends on λ as well as buffer capacity (K). As λ and K increase, more packets are proceeding the target packet, which implies that the target packet will experience more queueing delay in the buffer before it reaches the head-of-queue. The second part of D_{succ} is D_{trans}, which depends on the number of contending stations (n) and BER. While increasing n increases the number of attempts to acquire the channel, increasing BER increases the number of data retransmission. For $n = 20$ and $BER = 10^{-5}$, D_{succ}

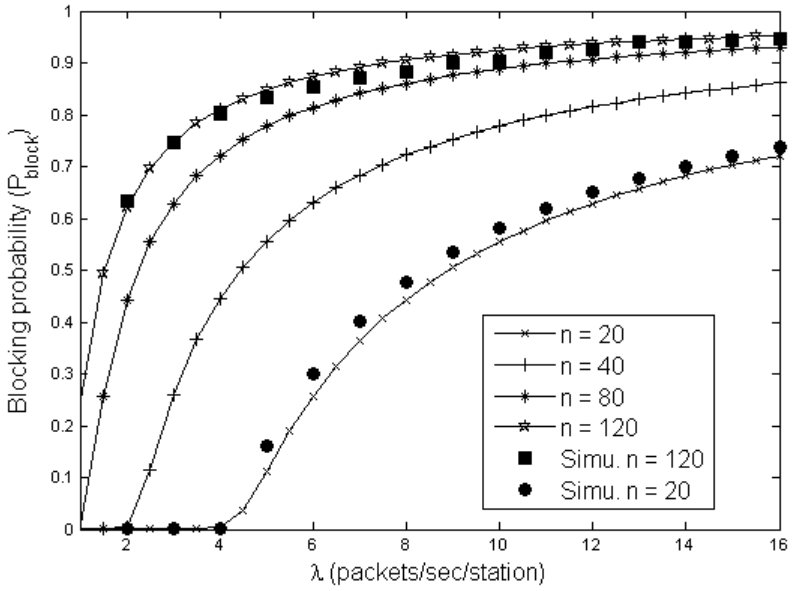

Figure 5.7: Blocking probability (P_{block}) for different number of stations, $K = 16$, $R_1 = 6$, $R_2 = 4$, and $BER = 1 \times 10^{-5}$.

is relatively small because the impact of collision and transmission errors, respectively, are low. Figure 5.8 shows that as λ is less than the service rate ($1/\overline{t_s}$), the transmitted packet faces an empty buffer, which means that the queueing delay is negligible. However, as λ exceeds the service rate, D_{queue} is exponentially increasing. Finally, when the buffer becomes full, $D_{succ} > K \times \overline{t_s}$.

The packet drop delay (D_{fail}) given by (5.35) is composed of the queueing delay and the delay that the packet experiences before the packet is discarded. The queueing delay (D_{queue}) is the same for a packet that is transmitted successfully and a packet that is dropped. Figure 5.9 shows that $D_{discard} >> D_{trans}$. This difference in the delay is expected because for the same n and BER values, the packet drop probability ($P_{discard}$) is very small compared to the packet success probability ($1 - P_{discard}$) as shown in Figure 5.9 (for $n = 20$ and $BER = 10^{-5}$, $P_{discard} = 0.002$).

For $\lambda = 8$ packets/second and $BER = 5 \times 10^{-5}$, Figure 5.10 shows the impact of data retransmission limit (R_2) on the packet discard probability ($P_{discard}$). At

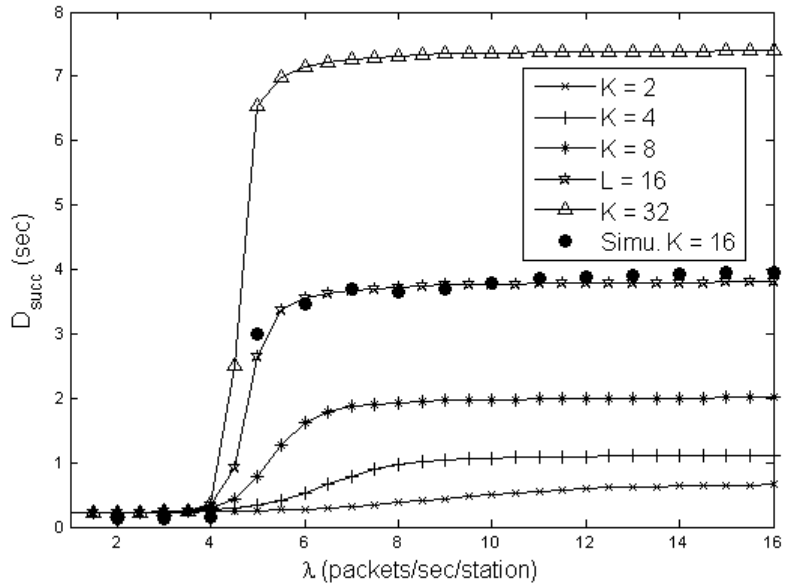

Figure 5.8: Packet delay (D_{succ}) for different buffer capacities, $n = 20$, $R_1 = 6$, $R_2 = 4$, and $BER = 1 \times 10^{-5}$.

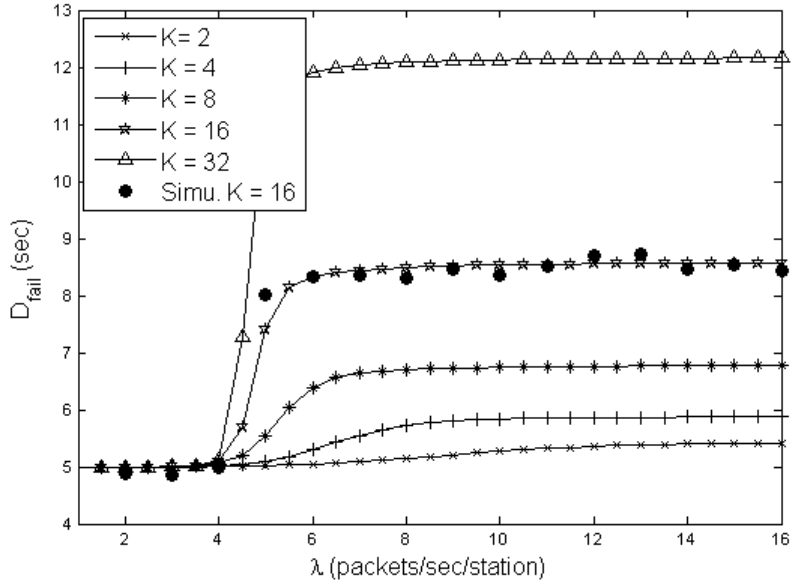

Figure 5.9: Drop delay (D_{fail}) for different buffer capacities, $n = 20$, $R_1 = 6$, $R_2 = 4$, and $BER = 1 \times 10^{-5}$.

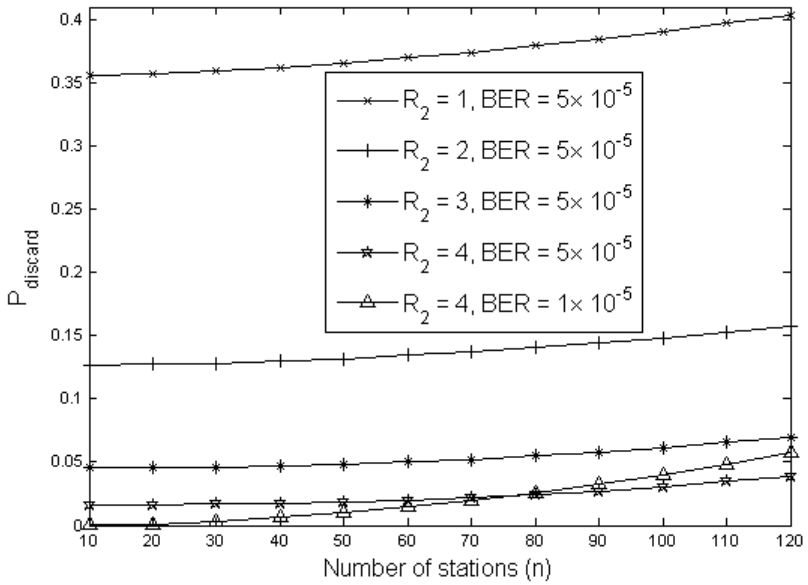

Figure 5.10: Packet drop probability ($P_{discard}$) for different BER and slrc values, $\lambda = 8$ packets/second, $R_1 = 6$, $K = 16$.

$BER = 5 \times 10^{-5}$, the probability that transmission errors corrupt the data packet (p_e) is equal to 0.3659. With such high probability, the station should retransmit the same data packet several times to recover from transmission errors ($P_{discard} \approx p_e^{R_2}$). Moreover, as the number n of the contending stations increases, the number of collisions increases too, and this reduces the chance to capture the channel.

The two main reasons that affect $P_{discard}$ are collisions and transmission errors. At $n \leq 10$, $P_{discard} \approx p_e^{R_2}$ because the impact of collisions ($p \approx 0$) on $P_{discard}$ compared to p_e is negligible. Therefore, as the BER increases, $P_{discard}$ increases too, e.g. at $BER = 1 \times 10^{-5}$ and $BER = 5 \times 10^{-5}$, $p_e \approx 0$ and $p_e = 0.0179$ respectively. But, when n starts increasing, the number of collisions starts increasing too as shown in Figure 5.11. As a result, the transmitted packet may be discarded due to the bit transmission errors or the inability to acquire the channel. Although both p and p_e have strong influence on the $P_{discard}$ when $BER = 5 \times 10^{-5}$, Figure 5.10 shows that $P_{discard}$ at $BER = 1 \times 10^{-5}$ is larger. This unexpected trend in $P_{discard}$ can be

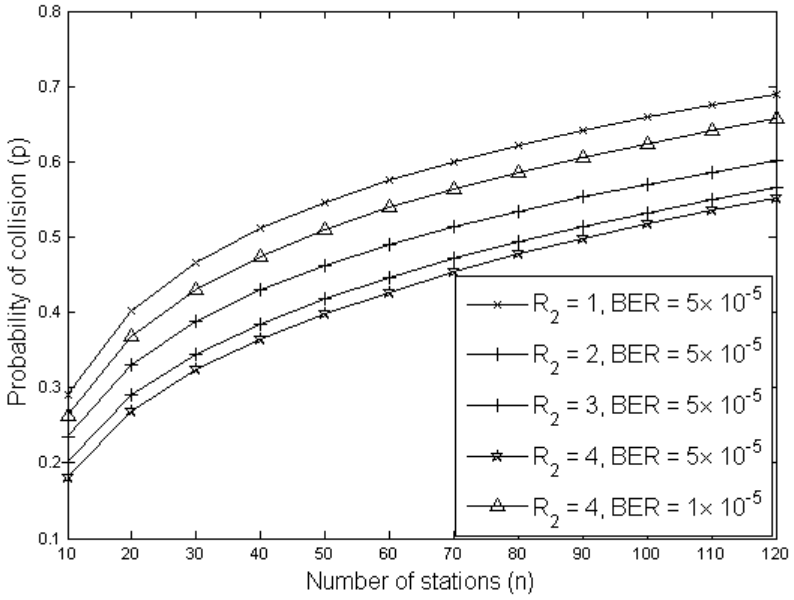

Figure 5.11: Collision probability (p_c) for different BER and slrc values, $\lambda = 8$ packets/second, $R_1 = 6$, $R_2 = 4$, and $K = 16$.

attributed to the fact that when p_e is high, the contention window (CW) sizes of the contending stations are relatively high because stations double their CW in case of unsuccessful transmissions. Consequently, large CW sizes reduces collision between the contending stations as shown in Figure 5.11.

5.5 Summary

Existing Markov chain models of IEEE 802.11 systems studied the QoS performance and queueing behavior by integrating the IEEE 802.11 contention resolution and queueing processes into one model. However, additional queueing processes increase the number of model state variables and parameters and make the model difficult to understand and analyze. In this Chapter, we show how to reach the same objective without increasing the computational complexity. Specifically, the new packet arrivals are disallowed during backoff countdowns and retransmissions. This assump-

tion makes arrivals and service time independent and thus a finite capacity M/G/1/K queue with independent samples from the saturation analysis is used to model the queue length. This allows us to accurately capture important QoS measures such as delay, loss, throughput, and queue length for 802.11 systems with finite buffer under finite load. Our queue analysis points to the existence of an effective maximum throughput and shows its relationship with station offered load. Extensive simulation and analysis results show that our model captures the system dynamics over a wide range of traffic load, buffer capacity, network size, and channel condition. In the next Chapter, the M/G/1/K queueing model will be used to analyze the performance of the multi-hop wireless network.

Chapter 6

Performance Analysis of IEEE 802.11 in Multi-hop Wireless Networks

6.1 Introduction

The IEEE 802.11 MAC protocol works well in single hop wireless local area networks (WLANs) because all nodes are in direct communication and each node is able to detect the activity of other nodes. Therefore, running the protocol locally by a node is enough to regulate its access to the shared communication channel. Although, in a multi-hop wireless network, nodes might not be able to communicate directly with others due to their limited radio range, the IEEE 802.11 protocol is used there. The inability to detect the activity of others gives rise to use intermediate nodes as relays to achieve an end-to-end communication, and this produces the well known hidden node problem [63]; the transmission of a node may collide at the intended receiver with a transmission of another node that is hidden from the transmitter.

Most of earlier analytical works for IEEE 802.11 concentrated on the WLAN setting and often relied explicitly on assumptions of saturated single hop networks

and basic collision models. They often used Markov chain model to analyze the saturated throughput [47, 49]. In addition, M/M/1 and M/G/1 models [53, 54, 61, 62] have been used for unsaturated throughput analysis. However, analytical modeling of IEEE 802.11 in multi-hop ad hoc networks is not straight forward because the channel can be reused for multiple transmissions at the same time and no central/distributive scheduler that coordinates nodes access to the channel. In [64], throughput of each node of a multi-hop chain network is analyzed using traffic-based-analysis approach. In [65], a cycle time approach shows that all nodes receive equal throughput regardless of their data rates but the model does not take the effect of capture into account. In [66], the performance of multi-hop network has been analyzed using signal to noise and interference ratio (SINR) model. The model is subject to multiple signal reception and it takes into account the occurrence of packet captures at receiving nodes. Conflict graph and independent set approach [67] is used to analyze the maximum end-to-end throughput for both nodes that are optimally or randomly placed.

In this Chapter, we provide an approximate mathematical model to analyze the end-to-end throughput, average delay, fairness, and packet loss probability in the DCF in multi-hop wireless network. Our model mainly utilizes the non-saturation performance analysis of the single hop WLAN and extends it for a multi-hop analysis. Shortly, the interference and carrier sensing ranges model is used to divide the multi-hop wireless network into a congregation of interleaved single hop subnetworks. Throughput, average delays, fairnesses, and packet loss probabilities for those subnetworks are analyzed and used to analyze the performance of the multi-hop network.

The rest of the Chapter is organized as follows: in Section 6.2, we describe our system model for the multi-hop wireless network and provide extensions to calculate the end-to-end throughput, average delay, fairness, and packet loss probability of the multi-hop path. We present the simulation results and discussions in Section 6.3, and summarize the Chapter in Section 6.4.

6.2 Multi-hop Ad Hoc Network Analysis

In multi-hop ad hoc network, we consider a system of n nodes that are randomly distributed over an area. All nodes are equipped with omni-directional antennas and with IEEE 802.11 cards that use the RTS/CTS access mechanism. For each node we define three radio ranges as following:

- the transmission range (R_t) is the range from the transmitter node (u) within which u's transmission can be successfully received or overheard at node v.
- the carrier sensing range (R_{cs}) is the range within which a transmission can be sensed at node u, even though correct packet reception may not be available.
- the interference range (R_i) is the range within which node u's transmission can collide at node v with other concurrent transmissions.

Unlike single hop network, wireless nodes in multi-hop network cannot detect the activities of all other nodes. Therefore, and due to the hidden node problem, a transmission from node u to node v that exists in u's transmission range may fail even though no other transmissions is sensed by u in that slot time. This may happen because (i) node v receives u's transmission successfully, however in CSMA/CA protocol, the physical carrier sensing performed at each node before it starts transmission and this disallows v to transmit a CTS if there is a transmission by another node w within v's carrier sensing area, (ii) another transmission by an interferer node w that exists in v's interference area and outside the u's carrier sensing area may corrupt packet reception at node v because SINR is less than the power capturing threshold ($CPThreshold$, which is usually set to 10). However, and by using the two-ray ground path-loss model, if transmission is going from u to v, $R_i = \sqrt[\kappa]{CPThreshold} \times d = 1.78d$, where κ is the path-loss factor, and $d \leq R_t$ is the separation distance between nodes u and v. Given, R_i depends on d and $R_{cs} = 2 \times R_t$, then the interference area is included in the sensing area. To simplify our analysis, we assume that if the separation distance between the multi-hop nodes u and v are long

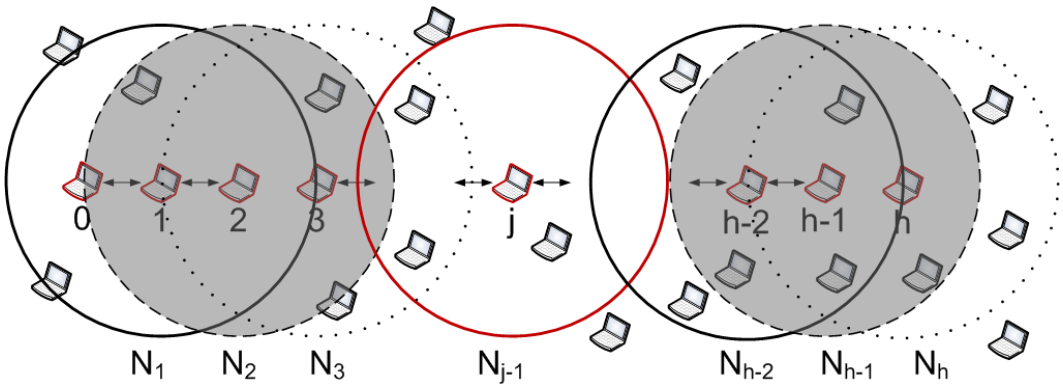

Figure 6.1: A wireless multi-hop flow with k hops/links

enough ($d \cong R_t$), RTS/CTS handshake and physical carrier sensing do not function well if v is in a transmission state or if there is an interferer node w that is $\leq 2 \times R_t$ away from node v.

In this Chapter, we analyze the performance of a multi-hop path in a network that has multiple active nodes. An active node is a node on a network that sends or forwards traffic to other nodes. Our analysis is mainly based on using the non-saturation performance analysis of a single hop network to analyze the performance of the multi-hop network. To achieve this, the network is divided into a set of interleaved single hop subnetworks. For example, let's assume that $h + 1$ nodes participate in a certain flow f (h hops), nodes are numbered sequentially from the source (node number 0) to the destination (node number h), see Fig. 6.1. The hop between nodes j and $j + 1$ is denoted by h_j. Based on our earlier assumption that RTS/CTS handshake cannot function well if node u transmits to node v while node v is in a transmission state or if there is an interferer node w that is $\leq 2 \times R_t$ away from node v, the network can be divided into k interleaved and cascaded single hop subnetworks ($N_{1,2,...,k}$), where N_j centers around the node j, for $1 \leq j \leq k$, and extends up to $2 \times R_t$ as shown in Fig. 6.1.

We further assume:

- The network is operating in non-saturation mode and nodes have buffers of

finite capacity K. Packet loss may occur due to collisions, transmission errors, or buffer overflow.

- In addition to the multi-hop path, the network has a random number of active nodes that are distributed randomly. The aggregate packet arrival rate at node j is denoted by $\lambda_j = \lambda_j^{int} + \lambda_j^{ext}$ packets per second. Here, λ_j^{int} is the internal traffic load if the node serves as a source node and λ_j^{ext} is the sum of all traffic loads flow through this node if it serves as an intermediate node.

- Each subnetwork (N_j) has a set of nodes. By assuming the availability of 2-hop neighborhood information, nodes identities and their arrival rates, let n_j be the set of active nodes in N_j. Nodes can use local exchange of HELLO messages so that each node can determine the presence and information of its 1-hop and 2-hop neighbors.

6.2.1 End-to-End Throughput

Based on the neighborhood information, the successor node determines the throughput of its predecessor node, i.e., for flow f, see Fig. 6.1, node j determines the throughput of node $j-1$ and then its arrival rate from that flow, $\lambda_{j,f}$. To calculate the throughput, we have to take into consideration:

(a) The existence of other active nodes.

(b) The possibility that nodes can serve multiple flows at the same time.

Active nodes may be saturated or non-saturated and this implies the non-saturated nodes do not utilize their total estimated throughput. The non-utilized throughput can be used by the other nodes. However, node j uses the node $(j-1)$'s aggregate arrival rate to determine the throughput of node $j-1$. Therefore, to determine the contribution from flow f in that estimated throughput, the throughput should be multiplied by a fraction, i.e., ratio of the flow's arrival rate to the aggregate arrival

rate at node $j-1$. As a result, throughput calculation requires an iterative procedure. In other words, node j estimates the saturation throughput S^s_{j-1} for node $j-1$ using saturation analysis. Second, by using non-saturation analysis, the non-saturated throughput S^u_i is estimated for $\forall i \in n_j$ with $\lambda_i \leq S^s_i$. The throughput of those nodes are extracted from the total channel saturation capacity, $S^s = n_j \times S^s_{j-1} - \sum_{\forall i \in n_{ex}} S^u_i$ (where n_{ex} is the set of excluded nodes) and they removed from the set $n_j = n_j - n_{ex}$. Third, we determine the new saturation throughput for the remaining nodes $S^s_{j-1} = S^s/n_j$. Fourth, we initialize the set n_{ex} and repeat the second and third steps until either (a) $\lambda_{j-1} \leq S^s_{j-1}$ or $n_{ex} = \phi$ and at this point, we determine S^u_{j-1} by means of non-saturation analysis. Finally, we determine $\lambda_{j,f} = a \times S^u_{j-1}$, where $a = (\lambda_{j-1,f})/(\lambda_{j-1})$. This process starts with node 1 and ends at node k. The throughput for the multi-hop flow $T_f = S^u_{k-1,f}$ because, in a any multi-hop flow, the throughput of a node cannot exceed that of its predecessor node. Algorithm 2 shows the details of throughput calculation.

6.2.2 Packet Loss Probability

In multi-hop ad hoc network, packet losses are caused by collisions, transmission errors, or buffer overflow. In CSMA-CA, collision occurs when either two stations, at least, transmit at the same time although they have a common sensing range or due to the hidden node problem. The former type of collision occurs due to the distributed nature of CSMA-CA mechanism. As well, transmission errors may corrupt packet transmission and cause packet losses. IEEE 802.11 DCF resolves collisions and transmission errors problems by means of retransmissions. On the other hand, buffer flow losses are related to the buffer size and the packet mean service time. Retransmissions due to collisions and transmission errors sharply increase the packet mean service time as well as the packet end-to-end delay.

Th station's throughput represents the time average traffic transmitted successfully by that station. Thus, the throughput S^u_{j-1} of station $j-1$ and its aggregate

Algorithm 2 Iterative method to calculate a throughput of multi-hop flow

Require:

- Flow f with $h+1$ nodes and h hops;
- n_j, $\forall j \in [1,h]$, $\{id, \lambda_i, \lambda_{i,f}\}$, $\forall i \in n_j$;

Ensure: T_f;

$j \leftarrow 1$;
while $j \leq h$ **do**
  $n \leftarrow n_j$;
  Find $\overline{X}$;
  $S^s_{j-1} \leftarrow 1/\overline{X}$;
  $S^s \leftarrow |n| \times S^s_{j-1}$;
  if $\lambda_{j-1} \leq S^s_{j-1}$ **then**
    find_throughput;
    Continue;
  end if
  $done \leftarrow false$;
  while $done = false$ **do**
    find_new_saturation_throughput;
    if $n_{ex} = \emptyset$ **then**
      find_throughput;
      $done \leftarrow true$;
    end if
  end while
end while

1: **procedure** find_new_saturation_throughput
2: $n_{ex} \longleftarrow \emptyset$;
3: **for all** $i \in n$ **do**
4: **if** $\lambda_i \leq S^s_{j-1}$ **then**
5: Find S^u_i;
6: $S^s \leftarrow S^s - S^u_i$;
7: $n_{ex} \longleftarrow i$;
8: **end if**
9: **end for**
10: $n \longleftarrow n - n_{ex}$;
11: $S^s_{j-1} \leftarrow S^s / |n|$;
12: **end procedure**

1: **procedure** find_throughput
2: Find S^u_{j-1};
3: $a \leftarrow \lambda_{j-1,f} / \lambda_{j-1}$
4: $\lambda_j \leftarrow \lambda_j + a \times S^u_{j-1}$;
5: $T_f \leftarrow S^u_{j-1}$;
6: $j \leftarrow j + 1$;
7: **end procedure**

arrival rate λ_{j-1} can be used to estimate the packet loss probability at node $j-1$ as follows,

$$P_{loss,j-1} = (S^u_{j-1} - \lambda_{j-1})/\lambda_{j-1} \tag{6.1}$$

After determining the packet loss probabilities for the all nodes ($0 \leq j < h$) that participate in flow f, the end-to-end packet loss probability of flow f can be calculated as follow,

$$P_{loss} = 1 - \prod_{j=0}^{h-1}(1 - P_{loss,j}) \tag{6.2}$$

6.2.3 Blocking Probability

The buffer blocking probability is the probability that a new arriving packet will not join the transmission buffer because the buffer reaches its maximum capacity. By means of M/G/1/K analysis and at state $k = K$, arrivals coming are blocked and denied entry into the system. Thus, blocking probability at node $j-1$ is

$$P_{block,j-1} = q_K \tag{6.3}$$

Similarly, the packet blocking probability at flow f is equal to,

$$P_{block} = \sum_{j=0}^{h-1} P_{block,j} \prod_{s=0}^{j-1}(1 - P_{block,s}) \tag{6.4}$$

6.2.4 Discarding Probability

As defined by the IEEE 802.11 standard [11] for RTS/CTS access mode, a packet is discarded after R_1 retries of the RTS transmission without receiving a CTS or after R_2 retries of data transmission, preceded by an RTS/CTS handshake, without receiving an ACK. Given that the source of packet losses is either packet blocking or discarding, then packet discard probability at node $j-1$ is,

$$P_{discard,j-1} = (P_{loss,j-1} - P_{block,j-1})/(1 - P_{block,j-1}) \tag{6.5}$$

The total discard probability at flow f is equal to,

$$P_{discard} = (P_{loss} - P_{block})/(1 - P_{block}) \tag{6.6}$$

6.2.5 End-to-End Delay

In multi-hop network, the end-to-end delay for a successfully transmitted UDP packet is defined as the difference between the time a packet arrives at the node without being blocked and the time the packet is successfully received by the final destination node. This time is the sum of queueing and transmission delays of non-blocked and non-discarded packet over the whole multi-hop route.

6.2.5.1 Queueing Delay

The queueing delay is defined as the difference between the time a non-blocked packet arrives at the node and the time the packet reaches the head-of-queue and starts transmission. This time mainly depends on whether the queue is empty or not empty at the instant of arrival and the average queue length of the non-empty queue. The mean number of $\overline{B_{j-1}}$ of packets in the system can be determined by means of the M/G/1/K analysis. Using (5.42), the average queue length in node $j-1$ can be expressed as,

$$\overline{B_{j-1}} = \sum_{k=0}^{K} k.q_k \tag{6.7}$$

Further, the mean time $D_{queue,j-1}$ spent in the queue by a packet which is not blocked is given by

$$\begin{aligned} D_{queue,j-1} &= (1 - P_{block,j-1}) \times \\ &\quad (q_0 \times 0 + (1 - q_0) \times \overline{B_{j-1}}/S^u_{j-1}) \end{aligned} \tag{6.8}$$

6.2.5.2 Transmission Delay

The transmission delay at each node is the time delay measured from the moment that the packet reaches the head of of the queue to the time that sender receives an ACK confirming its successful reception. It mainly consists of three parts: the time to successfully transmit the packet, the backoff time, and the retransmission time. For example, the transmission delay $D_{trans,j-1}$ at node $j-1$ is equal to,

$$D_{trans,j-1} = E[X] \times t_{slot} \tag{6.9}$$

where $E[X]$ is the expected number of slots to successfully transmit a packet as given in (4.27).

Now, the average packet delay at node $j-1$ is equal to,

$$D_{succ,j-1} = D_{trans,j-1} + D_{queue,j-1} \tag{6.10}$$

Finally, the end-to-end delay of $h-$hops route can be expressed using (6.10) and packet loss probabilities as follow,

$$D_{succ} = \sum_{j=0}^{h-1}(D_{queue,j} + D_{trans,j}) \tag{6.11}$$

6.2.6 Fairness

In CSMA-CA protocol, nodes that are suffered from collisions backoff by doubling their contention windows. This gives a higher chance to win access to the channel by nodes that lately captured the channel and it gives rise to unfairness problem. In this section we will use Jain's Fairness Index to study the fairness in h-hops route.

Definition *Let S_j be the throughput of the directed link j. The link's throughput fairness index FI of the $h-$hops route is*

$$FI = \frac{(\sum_{j=0}^{h-1} S_j)^2}{h\sum_{j=0}^{h-1} S_j^2} \tag{6.12}$$

The maximum fairness index is $FI = 1$. It corresponds to a network where all links have similar throughput. To find the fairness of the selected multi-hop route, we need to know the expected throughput S_j^u for each node $j \in [0, h-1]$.

6.3 Performance Evaluation and Validation

We implemented a simulation model using the network simulator *(ns)* [59] to validate the results obtained from the analytical model under error-free channel. The SINR model of the *ns* is modified such that any concurrent transmission within the receiver's interference area will corrupt packet's reception at that receiver. In our simulation,

all nodes run the RTS-CTS access mode of the IEEE 802.11 DCF and they are static. The channel data rate is $1Mb/s$ and all nodes transmit UDP/IP packets and work in the non-saturation mode. We use the no ad hoc routing protocol (NOAH) [68] to setup static multi-hop routes without sending any routing related packets. The same traffic profile and node positions scenarios are used for both simulation and analytical environments setup. Our analysis and simulation mainly covers two scenarios: single chain disjoint multi-hop flow with and without active nodes. The disjoint flow is defined as the flow whose participating nodes only serve that flow. The network has 200 nodes that are randomly distributed over the simulation area. Active nodes other than the ones being participated in the disjoint path are configured to communicate with their direct neighbors (single hop communication) and they have different arrival rates ($4 \leq \lambda_{active} \leq 180$ packets/second). The simulation time lasts for 200 seconds and then terminated. Unless otherwise specified, Table 6.1 shows the system parameters used in the simulation and analytical analysis, which are the default values used for MAC 802.11 in the ns simulator.

Figure 6.2 shows the simulated throughput of multi-hop flow for three different scenarios, single chain network, single chain with variable-rate active nodes, and single chain with equal-rate active nodes. We setup our simulation such that 10% of nodes are active and have single hop communication with their direct neighbors and their flow rates range between 6 to 180 packets/second. We can notice from Figure 6.2 that the throughput of the multi-hop flow is dramatically decreased under active nodes scenarios due to the increase in number of collisions and the corresponding retransmissions. While the throughput in the case of variable rates is about 1.95 packets/second, it is about 11.2 packets/second for equal-rates scenario at $\lambda = 20$ packets/second. This considerable drop in throughput for variable-rate scenario can be attributed to the fact that high rate flows are monopolizing the channel because they have smaller contention windows. Actually, the absence of a congestion control mechanism in UDP makes short distances and high rates flows monopolize the available bandwidth which means that high rates single hop flows impact the performance

Table 6.1: DSS system parameters

Parameter	Value	Parameter	Value
n	200 nodes	R_1	7
R_2	4	CW_{min}	32
CW_{max}	1024	Data + Headers	$1000 + 58$KB
CTS	38B	RTS	44KB
ACK	38B	σ	$20\mu sec.$
SIFS	10μsec.	DIFS	50μsec.
EIFS	364μsec.	δ	$2\mu sec.$
R_t	250 m	R_i	500 m
R_d	1Mbps	R_{cs}	500 m
λ	$1 - -32$pkts/sec.	K	16pkts
Simulation time	200 sec	Simulation area	2000m $\times$ 500m

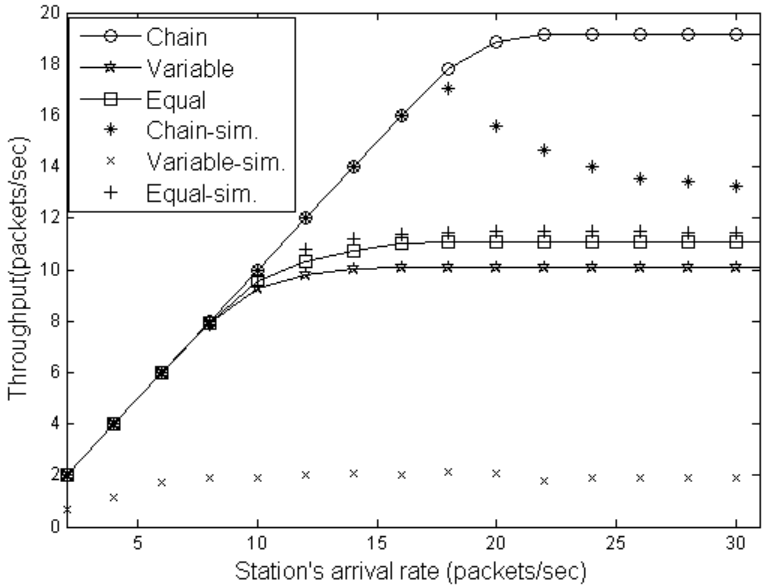

Figure 6.2: Flow throughput in packets/sec versus the station's arrival rate, analytical and simulation (packet size 1000 Byte and BER = 0).

of the multi-hop flow. In the scenario of equal-rates, arrival rates for active nodes are set to 8 packets/second and as a result the throughput of the multi-hop flow is enhanced from 1.95 packets/second to 11.2 packets/second. By comparing simulation results to the analytical ones, we see that they are fairly matching each other for the first and third scenarios. Results for variable-rates scenario do not match because the analytical analysis does not take in consideration the monopolization effect and this explain why the analytical results for both variable and equal rates are similar. Monopolization effect can be clearly noticed in the single chain scenario when $\lambda \geq 18$ packets/second. Beyond this rate, the traffic is generated at the first node in a saturated manner while traffic at later nodes originates from the first node is not saturated.

Figure 6.3 shows the average packet delay versus the arrival rate. The increase in the number of active nodes increases the delay due to the retransmission policy used to combat collisions. In single chain scenario and for $\lambda \geq 16$, node 1 is saturated and

its transmission buffer is almost full. This increases the number of packets waiting in the transmission buffer which in turn increases the delay ($\overline{B} \propto D_{trans}$) because the non-blocked packets need to stay longer in the transmission buffer until they reach the head-of-the-queue. by comparing the delay of the two active scenarios, we see that equal-rates scenario reduces the delay by about 42% at $\lambda = 32$ packets/second because in such scenario the channel is available for all nodes and thus their transmission buffers are not full. Also, it can be noticed from Figure 6.4 that packet loss probability increases as arrival increases. This is can be attributed to the fact that at saturation points, buffers become full and as a result packet blocking problem starts to appear ($P_{loss} \propto P_{block}$). Active nodes scenarios have higher packet loss probabilities compared to the single chain scenario because the number of nodes access the channel is higher and thus nodes need to wait longer time before they start transmitting. This increases the delay and as a result increases the blocking probability. Actually, Figures 6.2-6.4, show that to prevent a high packet loss rate and to enhance the throughput and delay for multi-hop flows, the offered load must be controlled not only for the multi-hop source node but also in the other active nodes. Algorithm 2 with some type of communication between neighbor nodes can be used to set the optimal offered load for all sources. Also, the existance of an optimal load for multi-hop flows were also pointed out in [69, 70, 71].

Figures 6.5-6.7 show the throughput, delay, and packet loss probability versus the long retry limit (R_2) under two different BER (10^{-5} and 5×10^{-5}). We can see how increasing the BER impacts the performance of the multi-hop flow, e.g., for single chain and at $R_2 = 4$, the throughput is reduced by 39%, delay increased by 26%, and packet loss probability increased by 42.2%. Moreover, under the same BER, the figure shows how R_2 impacts the performance, e.g., while increasing R_2 from 1 to 4, for single chain scenario, enhances the throughput and loss probability by 73% and 36% respectively, it deteriorates delay by 54%. This deterioration can be attributed to the fact that increasing number of retransmission produces more delay and increases the waiting time for new packets that joins the buffer. In spite the fact

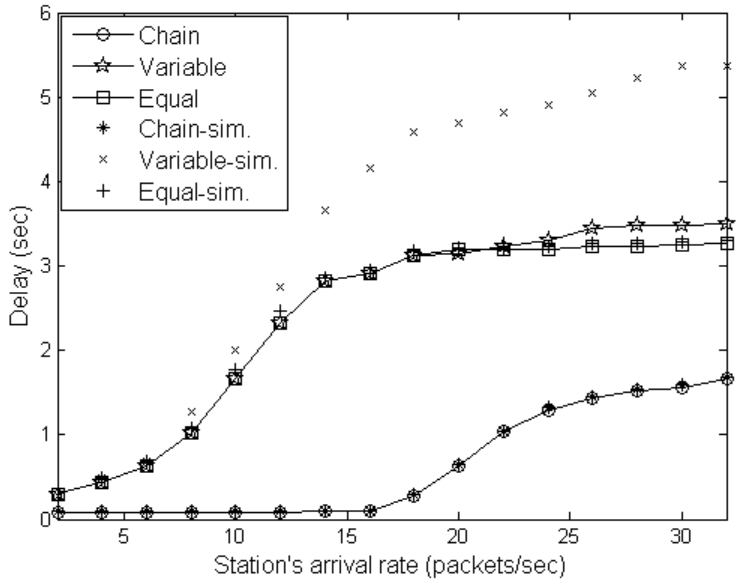

Figure 6.3: Average delay in packets/sec versus arrival rate under BER=0 (analytical and simulation).

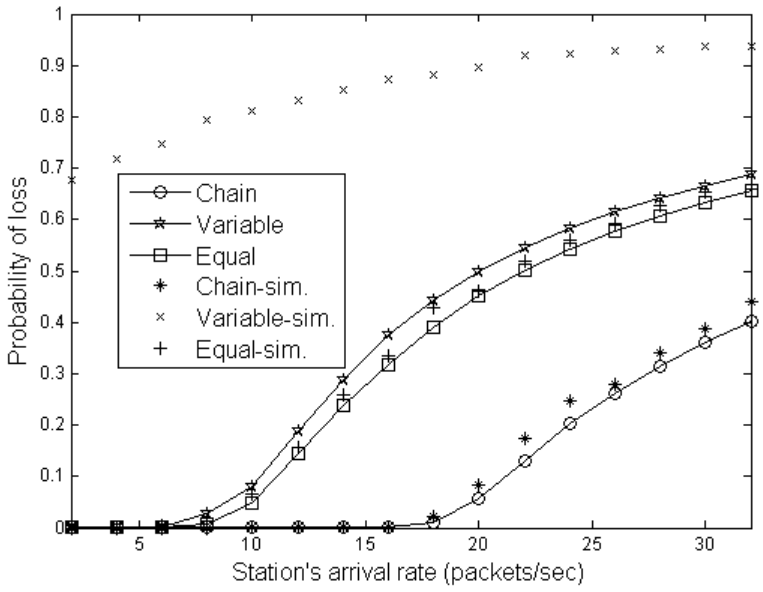

Figure 6.4: Packet loss probability versus arrival rate under BER=0 (analytical and simulation).

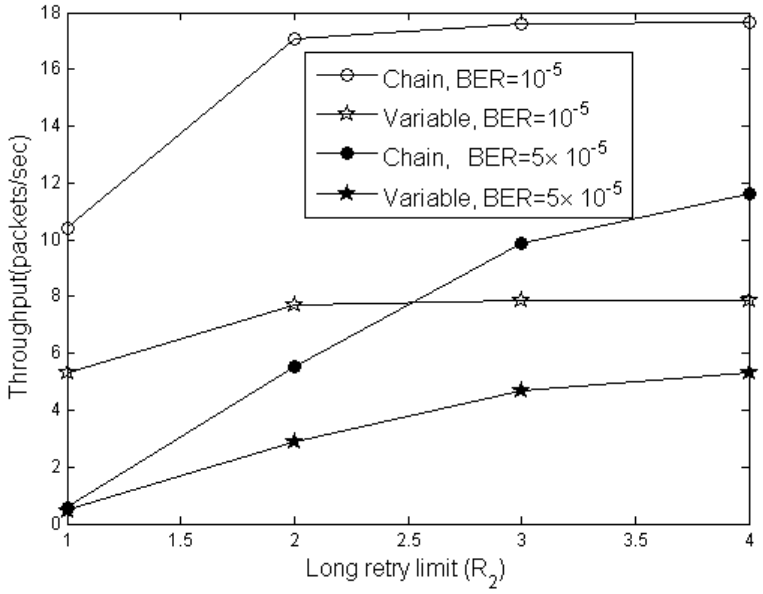

Figure 6.5: Flow throughput in packets/sec versus long retry limit.

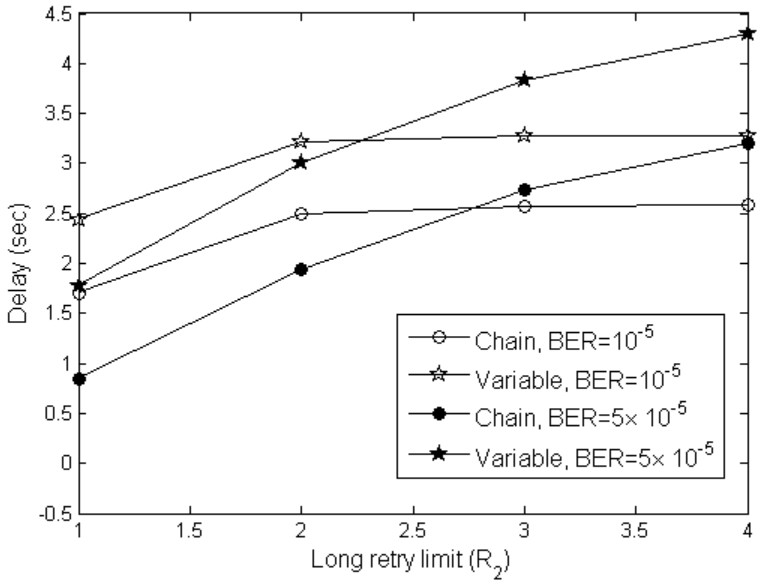

Figure 6.6: Average delay in sec. versus long retry limit.

that $P_{block} \propto D_{trans}$ and $P_{loss} \propto P_{block}$, packet discarding dominates blocking and this explains why increasing R_2 improves loss probability instead of deteriorating it.

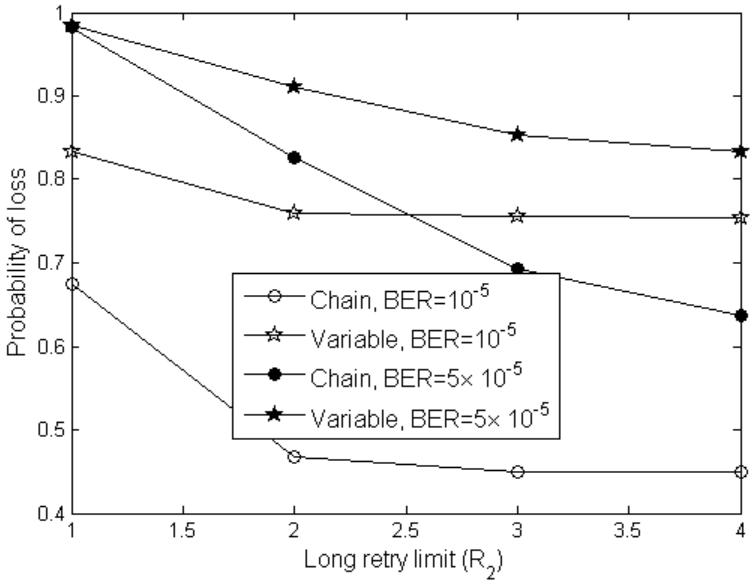

Figure 6.7: Packet loss probability versus long retry limit.

Figure 6.8 shows the multi-hop flow has lower throughput at small short retry limit (R_1) values compared with large R_1 values. This can be attributed to the fact that increasing R_1 reduces $P_{discard}$ and reduces p_c due to the large CWs sizes and this will enhance the throughput.

Figure 6.9 shows the throughput at each hop versus the arrival rate. The reduction in the throughput usually occurs on the first three hops because nodes 1 and 2 pump more packets to the following nodes than they can forward. While this results in an excessive packet loss at nodes 2 and 3, nodes 4, 5, and later ones have low arrival rates than their maximum capacity and thus their packet loss almost reaches zero and their throughputs are equal. The same behavior can be noticed for variable-rate scenario if the active nodes are relatively uniform distributed. This phenomenon gives rise that selecting longer routes could avoid areas of the high number of active nodes and achieve better throughput.

Figure 6.10 shows fairness against arrival rates. The fairness is calculated using (6.12). We see that at low arrival rates, $FI \approx 1$ and then starts decreasing as the arrival rate exceeds the maximum throughput. This again gives rise to the need of

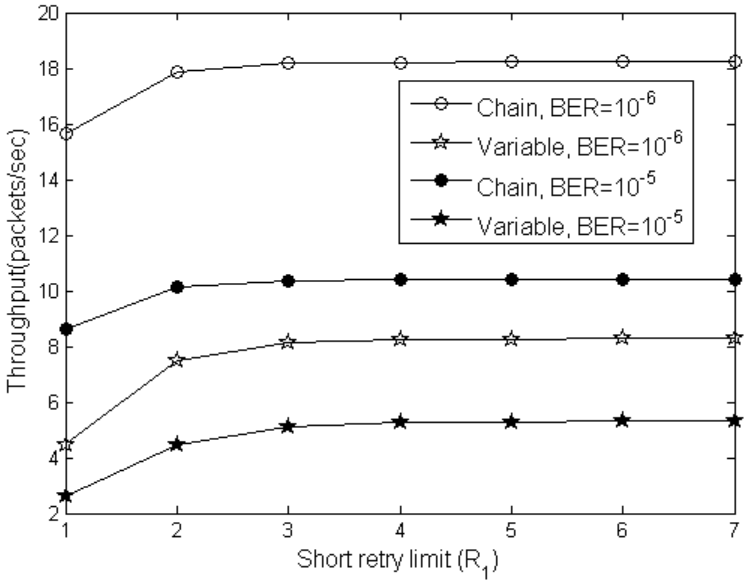

Figure 6.8: Flow throughput in packets/sec versus short retry limit.

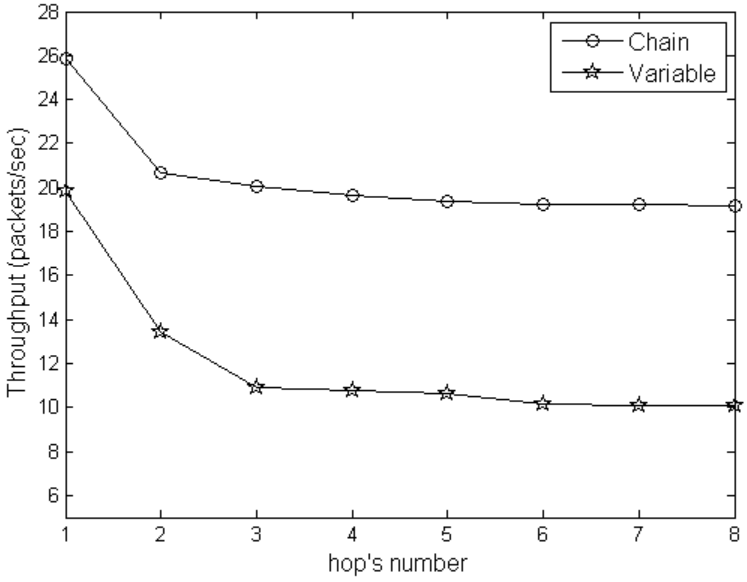

Figure 6.9: Flow throughput in packet/sec versus number of nodes in multi-hop network with unsaturated traffic sources.

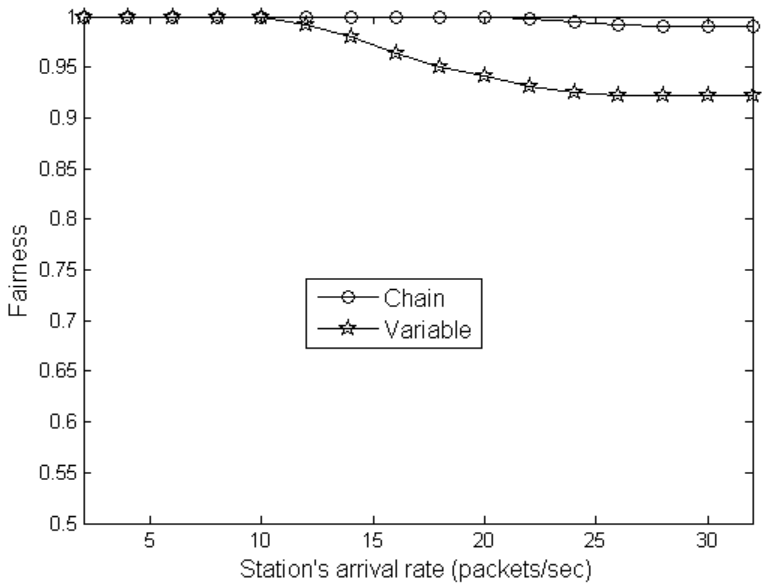

Figure 6.10: Fairness versus packet arrival rate

an efficient mechanism to control traffic loads.

6.4 Conclusion

IEEE 802.11 network uses the physical carrier sensing and RTS/CTS handshake as the main two techniques to combat interference and hidden node problem. But both techniques do not function well if the interferer or hidden node is beyond the transmission range of receivers. In our model, we take this fact into consideration when analyzing the performance of IEEE 802.11 in multi-hop wireless network. Instead of analyzing the network based on the behavior of the transmitter node, we take the receiver node as the point of reference. Mainly, we divide the network around the multi-hop path into a congregation of interleaved single hop subnetworks, i.e., each subnetwork covers up to 2-hop neighbors of one of the reference nodes. We use the finite capacity M/G/1/K queue with multiple vacations model that uses independent samples from saturation analysis to analyze the non-saturation performance of the single hop wireless network. By means of non-saturation analysis from Chap-

ter 5, we present a general analytical model and an iterative method to analyze the performance of the multi-hop path. Our model is accurate as the analytical fairly match the simulation results. In our analysis, the impact of retry limit, BER, path length, and active node intensity on the throughput are discussed. From analytical and simulation results, we conclude that there is a need to find the optimal offered load for the multi-hop flow and other active nodes to avoid high packet loss rates. We proposed an iterative mechanism that can be used with means of communication between adjacent nodes to determine the maximum achievable throughput. In the Chapter 7, we will implement and use our multi-hop analysis in the route selection process to discover high QoS routes.

Chapter 7

Cross-Layer QoS Route Selection

7.1 Introduction

Nowadays, multimedia services play a central role for many social and entertainment applications. Provision of QoS guarantees by MANETs is a challenging task due to node mobility, multi-hop communication, unreliable wireless channel, lack of central coordination, and limited device resources [3]. Hence and for a proper operation of multimedia services in MANETS, the QoS routing is essential instead of the best-effort routing. Different QoS metrics can be considered to satisfy QoS requirements in route selection: e.g., minimum required throughout, maximum tolerable delay, maximum tolerable delay jitter, and maximum tolerable packet loss ratio [4]. In this Chapter, we focus on providing the QoS based on throughput because most of voice or video applications require some level of guaranteed throughput in addition to their other constraints.

On-demand routing protocols, such as AODV [18] and DSR [17], often use blind flooding technique to establish and maintain communication routes. The source node broadcasts a RREQ message with the time-to-live (TTL) value equals 1. The RREQ message is uniquely identified by the source node and a sequence number. When intermediate nodes receive the RREQ for the first time, they register its identification in their route broadcasting tables. The route broadcasting table is vital to control

the flooding process (i.e., further requests of the same identification information are discarded) and to prevent the formulation of routing loops. If the RREQ reaches an intermediate node, the intermediate node rebroadcasts the RREQ to its neighbors. This process continues until the RREQ reaches the destination node. The route details (the sequence of hops that the packet is to follow on its way to the destination) can be implemented by either listing the address of each intermediate node through which the RREQ has been forwarded [17] or creating a reverse route in intermediate nodes's routing tables [18] (a reverse route is a route setup to forward a RREP packet back to the source node). Upon receiving the RREQ by the destination, it replies with a route reply (RREP) message. The RREP messages propagates via the reverse route [18] or implicitly encoded in the transmitted RREP [17]. As an optimization for flooding process, if an intermediate node has a fresh enough route to the destination, it cancels the RREQ and sends back a RREP to the source node.

Blind flooding sends a RREQ message to every node of the ad hoc network. A RREQ message can be flooded by an iterated use of broadcast. However, flooding consumes scarce network resources, power and bandwidth, due to the unnecessary routing overheads. Additionally, flooding causes a broadcast storm problem [24] that leads to a significant network performance degradation due to the high contention and collision in the network. To solve this problem, nodes need to use a random rebroadcast delay (RRD) [25] to randomly delay the RREQ transmission. But still, replying to the first received RREQ results in the next hop racing problem [25], in which the worst next-hop candidate in terms of link lifetime, throughput, or delay is chosen instead of the good one. For example, in Figure 7.1, assume node E is in communication with node I ($E \rightarrow H \rightarrow I$) and at the same time the source node A broadcasts a RREQ message to establish a connection with the destination node G. At time t_1, nodes B and C receive the RREQ packet from node A. Since RRD is randomly selected, it is possible that node B rebroadcasts the RREQ packet before node C. in such scenario, nodes E and F receive the RREQ at time t_2. At t_3, nodes D and E receive the RREQ from node C but E cancels this RREQ because it has

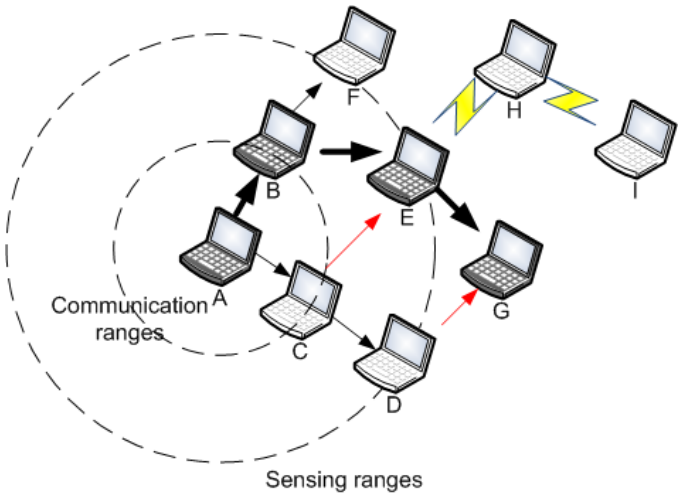

Figure 7.1: Next-hop racing problem

received the same RREQ from node B. Given that the routing traffic is prioritized over data traffic, node E may broadcast the received RREQ from node B before node D. As a result, node G will receive the RREQ from node E at time t_4 and send back a RREP message to the source node A. Since node E is already involved in another communication with node I, it has less capacity to serve node's A traffic than node D. Although, node D is a better candidate than node E, nodes B and E are selected as next-hops due to the deficiency of the RRD mechanism. Moreover, if we assume that nodes are not static and node B moves outward faster than node C, then the destination selects the worst route in terms of route lifetime and reliability.

To discover and select QoS routes, nodes need to evaluate their capacity to serve the future incoming traffic. Based on this evaluation, they assign a high rebroadcast priority to broadcast the RREQ if they have the capacity to provide the requested QoS. As a result, good candidates choose small RRDs and bad ones choose large RRDs. Also, during the chosen RRD period, nodes have to evaluate the received RREQ messages instead of responding to the first one. For example, subject to the throughput, delay, packet loss, and/or buffer space constraints, the node can classify its previous hops into good or bad candidates. Accordingly, the received RREQ from the best candidate is considered for re-broadcasting. By following these two

techniques, better end-to-end routes in terms of the desired quality of service metrics are discovered.

To investigate the deficiency of the current route discovery mechanism in AODV routing protocol, we setup a simulation scenario for a single chain network and analyze all candidate routes in each route discovery cycle. We noticed that the sequence of hops that are participated in the candidate routes are similar except the last two or three hops. For example, for different five scenarios (number of nodes equal 200), we fix the source and destination nodes at specific locations (we set the distance between the source and destination nodes to 1400m) and then run the simulation. Randomly, we choose one of the route request cycles and analyze the all candidate routes in that cycle. We observed that in average the number of possible candidate routes is about 10 and the length of a route is ranged between 6 and 9 hops where the first five hops are common in all of them. This observation emphasizes that the current route discovery mechanism is not enough efficient and only 10% out of 199 broadcasted RREQs are lived while others are ignored or discarded. Evenmore, the number of the received RREQs by the destination are not enough such that the destination can select the QoS route.

In this Chapter, we will propose a QoS route selection algorithm. The algorithm utilizes the non-saturation performance analysis of IEEE 802.11 DCF to select the best RREQ that can satisfy the requested QoS.

Section 7.2 discusses the related work. In Section 7.3, we will propose a cross layer QoS route selection algorithm that selects QoS routes subject to the requested throughput, delay, packet loss, and/or buffer space constraints. This algorithm enables nodes to evaluate the received RREQs and then choose the best RREQ for broadcasting. In Section 7.3, we present the implementation details of the proposed algorithm over AODV routing algorithm. Section 7.5 discusses the simulation model and results. Finally, Section 7.6 summarizes the Chapter.

7.2 Related Work

To achieve QoS routes, the QoS parameters (e.g., throughput, delay, delay jitter, packet loss ratio, power, and energy) should be included in the transmitted routing packets. In the bandwidth-aware routing protocols, the bandwidth information or prediction techniques are needed [72]. Some of the proposed bandwidth-aware routing protocols assume the availability of the bandwidth information as in CEDAR [5], and TDR [7]. Others proposed appropriate techniques for the bandwidth estimation. For example, the OLSR-based QoS routing protocol [8] uses the channel's busy and idle times to estimate the available bandwidth. In ADQR [9], nodes handshake the bandwidth consumption information with neighbors. In highest minimum bandwidth (HMB) routing protocol [73], the network is inferred by source nodes using statistics which are collected locally.

Ticket-based QoS-aware routing protocol [6], adaptive mean delay (AMDR) routing protocol [74] and ad hoc QoS on-demand (AQOR) routing protocol [75] are examples of delay aware routing protocols. They use the route discovery latency as an estimation for the route delay but they do not consider the contention and interaction between neighbors in the delay calculation.

In delay jitter[1] and power loss aware routing protocols [76, 77], the receiver node monitors the received packets over a period of time to calculate these two QoS parameters. If the current path does not satisfy the requested QoS requirement, the receiver initiates a new route discovery cycle.

Energy-aware routing protocols [78, 79, 80, 81, 82] try to maximize the network lifetime and avoid network partition. On the other hand, power-aware routing [83, 84] protocols try to minimize the total power consumptions.

[1]Delay jitter is defined as the variation of delay over a period of time

7.3 QoS Route Selection Algorithm

Bandwidth estimation [8, 9, 73] is imprecise because different factors affect bandwidth availability such as network size, transmission power, channel characteristics, and the interaction and interference among neighboring nodes. Therefore, we believe that QoS routing based on an accurate analysis of IEEE 802.11 MAC protocol will provide better bandwidth information than the estimation techniques. Hence, in this section, we use the non-saturation analysis for wireless multi-hop networks to propose a QoS route selection algorithm. Moreover and due to the availability of the other performance metrics, our proposed algorithm can be used as a delay, packet loss, or fairness aware route selection protocol.

An efficient route discovery process should take into consideration two aspects. (a) To avoid broadcast storm problem and next-hop racing by using the RRD approach that minimizes contentions and collisions. As well, it priorities the next-hop candidates into good and bad candidates. (b) for a QoS route selection and during the RRD period, nodes should evaluate the received RREQ messages using the non-saturation performance analysis to select the best RREQ for future broadcast or route selection. The former one is presented and implemented in many previous works where the RRD value is prioritized based on power, throughput, load, delay, neighborhood, or/and active nodes calculations. In this work, we will study the RREQ evaluation and selection.

Finding QoS routes will reduce average cost of the flooding route discovery scheme in the traditional MANETs routing protocols. In this chapter, we propose a distributed route discovery algorithm that supports QoS requirement for MANETs. In this distributed algorithm, intermediate stations as well as the destination station utilize cross-layer information for QoS route selection. To implement the algorithm, the RREQ message should include additional information that helps the destination and intermediate stations to be involved in the QoS route selection. The RREQ message should include the QoS requirement from the application layer, the flow information

from the transport layer, and the end-to-end QoS information from the routing layer. The application and transport layer pass their information to the routing layer via the transmitted data packets.

The sender triggers the route discovery cycle by broadcasting a RREQ message. In addition to the route information (the source and destination addresses, source and destination sequence numbers, a broadcast ID, and a hop count), the sender adds to the RREQ the received information from the application and transport layers and initializes the end-to-end QoS information. Upon receiving the broadcasted RREQ by an intermediate node, the MAC layer of that node uses in addition to the neighbourhood information and channel information, the transport layer information to analyze the performance of its sub-network. Based on this analysis, the MAC updates the end-to-end QoS information of the received RREQ packet and forwards it to its upper layer, namely the routing layer. Based on the QoS requirements, the routing layer uses the end-to-end QoS information to evaluate the received RREQ message. After evaluating all candidate RREQs, the intermediate node rebroadcasts the best RREQ. This process continues until the destination receives all candidate RREQ messages and decides which route will be established. Figure 7.2 shows the interaction between different network stack layers in the QoS route selection process. Local resources availability are determined in the MAC layer. Specifically, the MAC starts with saturation analysis to determine the saturated throughput and the packet mean service time. Then, it uses M/G/1/K analysis with independent samples from the saturation analysis (the packet mean service time) to estimate and updates the QoS information in the each received RREQ message.

Our proposed algorithm helps in QoS route selection. For example, in Figure 7.1 node G evaluates the received RREQs from nodes E and D and it finds that node D provides higher throughput and less delay than node E. Therefore, it discards the RREQ from node E and replies to the one received from node D.

We implemented our approach using the network simulator (ns) by changing the route discovery mechanism for the AODV routing protocol. Also, the non-saturation

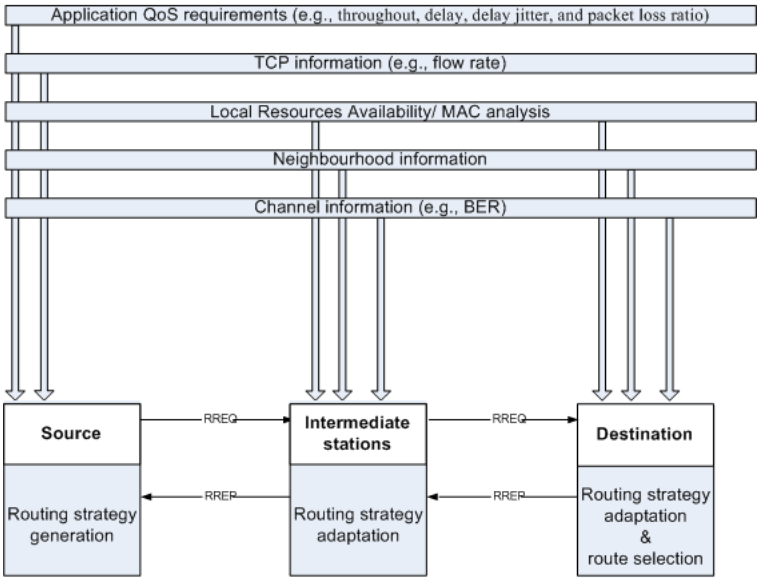

Figure 7.2: The cross-layer route discovery framework

analysis is coded into the mac-802._11 agent. Next section presents the implementation details of our QoS route selection algorithm.

7.4 Implementation

The implementation requires some modification to the mac_802.11 agent and the route discovery mechanism in AODV routing protocol. In the MAC layer side, we need to add functions for the saturation and non-saturation analysis. Those functions are executed whenever a RREQ packet is received. While the saturation analysis depends on the number of active neighbors and non-saturation analysis depends on flow rates for the active nodes, we need to cache the first and second hop neighborhood information (identities and flow rates). Neighborhood information can be collected by broadcasting local HELLO messages which is already implemented in the AODV protocol. However, to collect flow rates and second hop neighborhood information, the HELLO message should be modified such that its TTL value is increased from 1 to 2 and the node's flow rate is added to it. To keep track of neighborhood infor-

mation, we need to implement a neighborhood cache in the MAC layer. To optimize number of HELLO messages, only active nodes send HELLO messages in a periodic manner. In the routing layer side, the node needs to track the received RREQs in order to select the best one, therefore another cache is needed to store the good RREQ message. In addition, the RREQ packet should be modified to carry the QoS information. In addition to the node's aggregate traffic rate, current flow's traffic rate, and QoS requirement, additional fields should be added to calculate the end-to-end throughput, delay, packet loss ratio, packet discard ratio, and fairness information. We implemented our algorithm and tested it using UDP traffic type.

The route discovery mechanism is initiated when a route to new destination is needed by broadcasting a route request (RREQ) message. The source node prepares the RREQ message (it adds its aggregate arrival rate and current flow arrival rate to the message and initializes the QoS parameters) and broadcasts it to its neighbors. At the intermediate node side and upon receiving the RREQ, the MAC starts its calculation to estimate how many nodes should be added to its active neighborhood list. This number depends on the information carried in the RREQ, specifically the destination and previous hop identities, and if those nodes are part of neighborhood list (this number of additional nodes could be 0, 1, 2, 3, or 4). Now and based on the number of active nodes and their flow rates, the node performs the non-saturation analysis to estimate the throughput of the previous hop. Then using the estimated throughput and the previous hop's arrival rate, the node can find other performance metrics such as delay, packet loss probability, packet discard probability. Then, the MAC updates the QoS information in the received RREQ and forwards it to the routing layer.

In the routing layer, the node compares between the cached RREQ and the new received one. if the new RREQ has a better QoS, the new one replaces the cached one. The intermediate node keeps receiving and processing the incoming RREQs during the RRD period. After that, the node creates a reverse route to the source node, updates the previous hop and flow rate information, and broadcasts the cached RREQ. This

process continues until the RREQ reaches the destination node or an intermediate node with an active route to the destination. In the same manner the destination node processes the received RREQs and selects the good one and then responds with a route reply (RREP) message via the reverse route. The RREP message propagates between intermediate nodes until it reaches the source node. The route maintenance in our modified AODV algorithm is similar to the one used in AODV but it follows the previous guidelines.

7.5 Simulation and Discussion

We use AODV routing protocol from the network simulator *(ns)* [59] to implement our QoS routing protocol. Furthermore, the mac_802.11 protocol is modified to include the non-saturation performance analysis. In our simulation, all nodes run the RTS-CTS access mode of the IEEE 802.11 DCF and they are static. The channel data rate is $1Mb/s$ and all nodes transmit UDP/IP packets and work in the non-saturation mode. The number of nodes is equal to 200 and the simulation area is $2000 \times 1000m^2$. The source and destination nodes are fixed in the same locations in the all generated scenarios (the source node located at $(x, y) = (237, 243)$ and the destination node located at $(x, y) = (1423, 127)$). The percentage of active nodes is 5% and they are randomly distributed over the simulated area and have equal flow rates of 8 packets/second (active nodes communicate with direct neighbors). The simulated multi-hop path is a disjoint path. We compare the performance of the modified AODV with AODV routing protocol. The simulation time lasts for 300 seconds and the collected results are averaged over 10 different scenarios. The same simulation parameters' values listed in Table 6.1 are used here.

We start our simulations by showing the deficiency of the current route discovery mechanism in AODV protocol. Therefore, for one of the randomly generated network scenarios, we generated a random traffic profile and ran the simulation for 2000 seconds. After that, we analyzed the route components (nodes that participated in

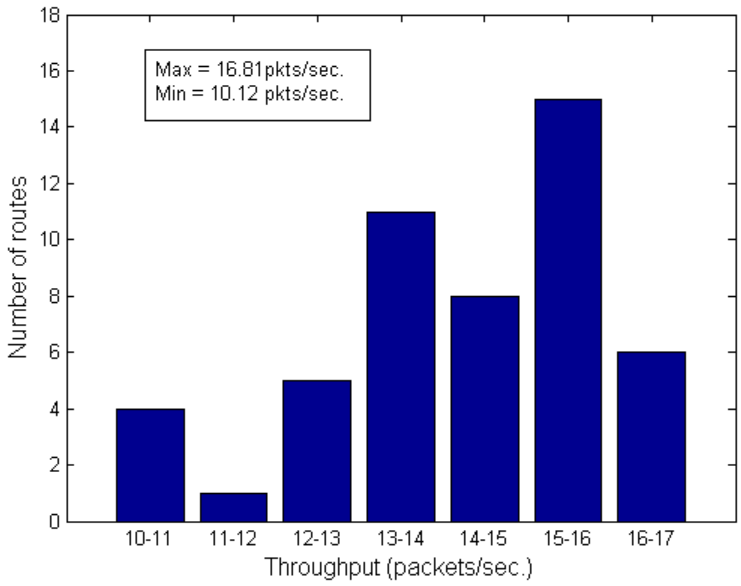

Figure 7.3: Routes distribution versus throughput

the route) for the first 50 route discovery cycles. Given that the first and second hops neighborhood information can be extracted from the movement and traffic files, we evaluated these routes analytically to calculate their end-to-end performance parameters. Figure 7.3, shows the distribution of the calculated throughput versus the number of the candidate routes. We noticed that only 12% of the candidate routes acheive the maximum throughput ($16 - 17$ packets/second).

Also, Figure 7.4 shows routes distribution versus packet delay. We see that the maximum packet delay is 1.06 seconds and the minimum is 0.46 seconds. The percentage of routes that have delays less than the average is 34%, which means that the possibility of selecting an unsuitable route is high. Hence, there is a need for a QoS mechanism that can filter these candidate routes to choose the appropriate one. In AODV, the destination sends back a RREP to the first received RREQ message. But this does not mean that the first received RREQ leads to a minimum delay. Taking in consideration that routing traffic is prioritized over data traffic, nodes broadcasts the received RREQ even though they serve other flows. However, if the route is chosen based on the first received RREQ, packet transmission will suffer delay in the

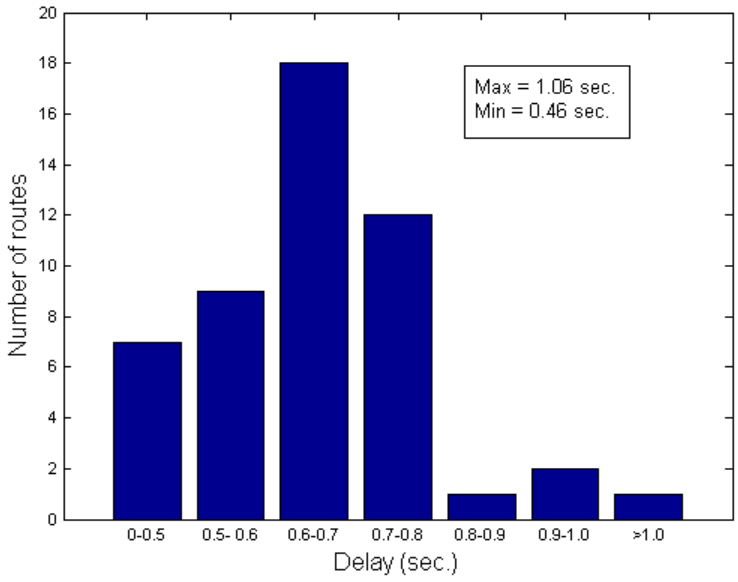

Figure 7.4: Routes distribution versus delays

intermediate nodes that serve multiple flows.

Figure 7.5 shows the distribution of candidate routes versus the packet loss probability. We notice that 12% of candidate routes have minimum loss ratios and about 56% of routes their loss ratios is less than the average. In AODV, loss probability cannot be predicted from the received RREQs. To predict or estimate loss ratios, the node needs to monitor the traffic for a period of time and count the number of the received packets and the number of transmitted packets. This implies that several route discovery cycles are needed to test the chosen routes and then decide if the selected routes acheive the QoS requirements.

Figure 7.6 shows the distribution of routes versus their lengths. We notice that the routes' lengths ranged between $6 - 13$ hops and the percentage of short routes is 32% (≤ 7 hops). Moreover, the figure shows that short routes are not necessary the best routes in terms of throughput, delay, or packet loss. For example, the shortest route has 6 hops but the maximum throughput is achieved when the route's length is 8 hops. Also, a 7-hops route gives better performance in terms of delay and packet loss compared to the shortest ones. The AODV route discovery mechanism does

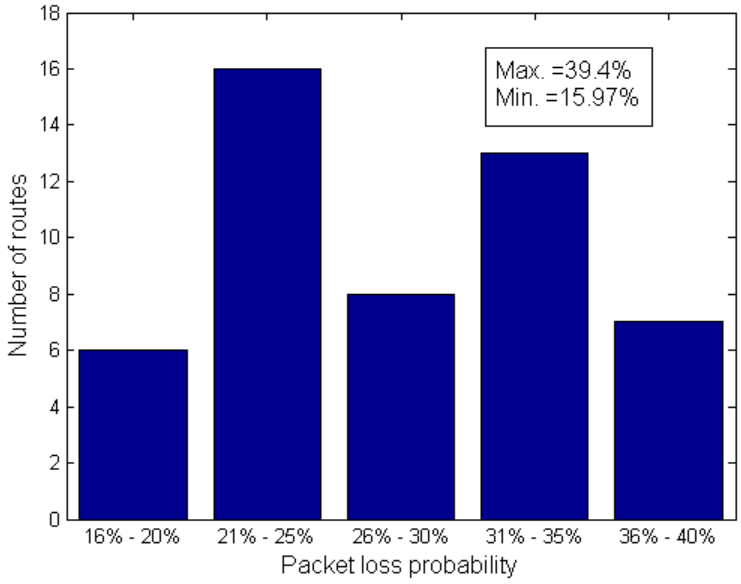

Figure 7.5: Routes distribution versus packet loss probabilities

not consider avoiding high congested areas when selecting the route. High congested areas cause many packet losses and increase delay and consequently degrade the throughput.

These results emphasize the strong need for an efficient routing mechanism that takes into consideration system information during route selection process.

Figure 7.6 plots the end-to-end throughput versus the flow rate for the AODV and modified AODV routing protocols. For low flow rates, both protocols have similar performance because the source node is not saturated so that it has an enough bandwidth capacity to transmit its data packets. At high flow rates, simulations show that the modified AODV has higher throughput than AODV. Also, the modified AODV reaches the saturation state at a higher flow rate than AODV. This enhancement in the throughput can be attributed to the quality of the selected route. Since the non-saturation performance analysis takes in consideration most of the system information (e.g., the number of nodes, channel BER, packet loss, queue length, and packet size) in throughput calculation, this implies that the selected route is the best one among all candidates routes. Even more, while AODV randomly selects the route and the

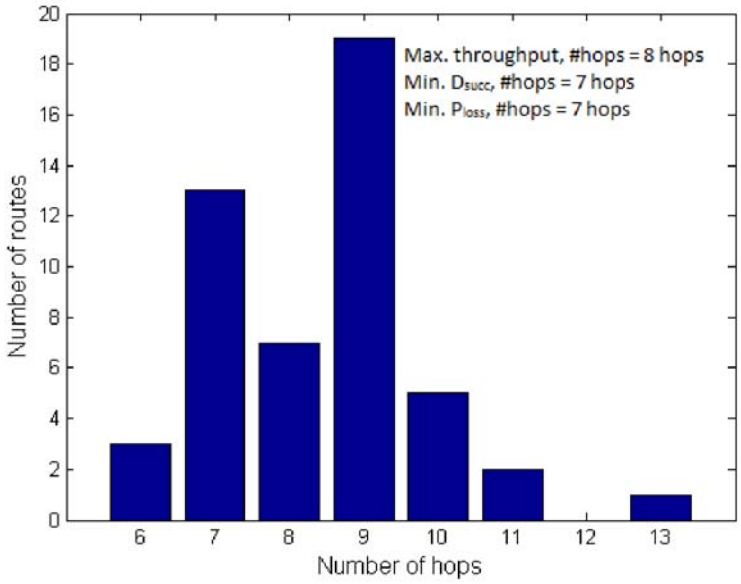

Figure 7.6: Routes distribution versus routes' lengths

selected route could be changed every route discovery cycle, the modified AODV has the ability to select the same route in different route discovery cycles. Additionally, we notice that the number of route discovery cycles in AODV is 0.776 cycles/second while in the modified AODV it is reduced to 0.47 cycles/second. This reduction in the number of route discovery cycles verifies the quality of the discovered routes.

Figures 7.7 and 7.9 plot the packet loss probability and packet delay versus the flow rate, respectively, for AODV and the modified AODV protocols. We notice that the modified AODV has better performance in terms of packet loss and delay compared to AODV. In this simulation, the percentage of active nodes is 5%, this implies that the collision probability is small and can be ignored. As a result, $P_{discard} \approx 0$ and $P_{loss} \approx P_{block}$. Given that $(P_{loss})\frac{1}{\propto}(S)$, then enhancing the end-to-end throughput will enhance the packet loss probability because it reduces the blocking probability as given in (5.51). Lower blocking probability implies less queueing delay and consequently a better end-to-end delay.

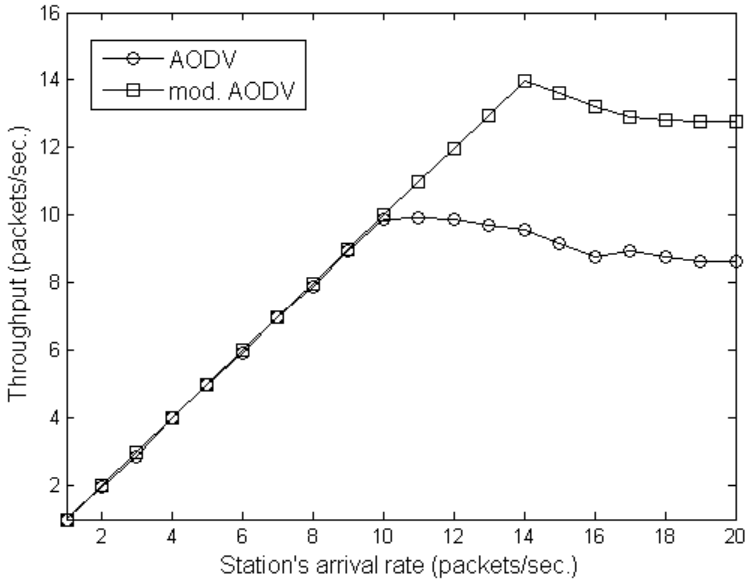

Figure 7.7: the end-to-end throughput versus the flow rate

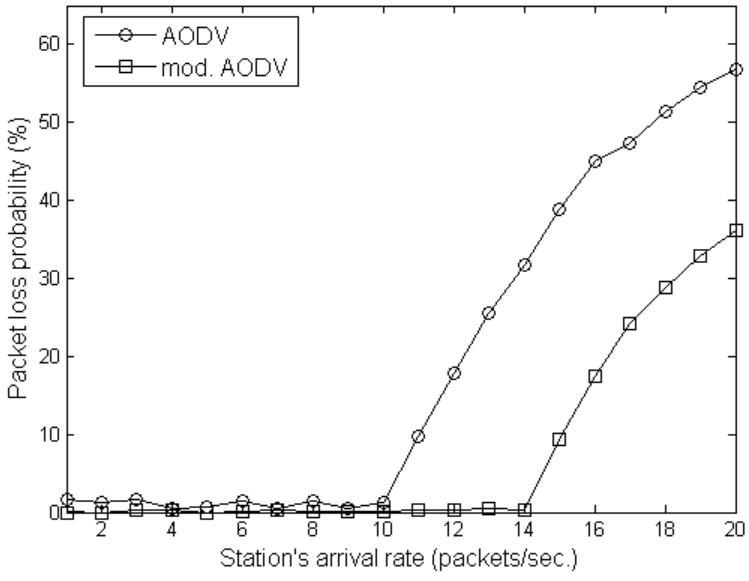

Figure 7.8: the end-to-end packet loss probability versus the flow rate

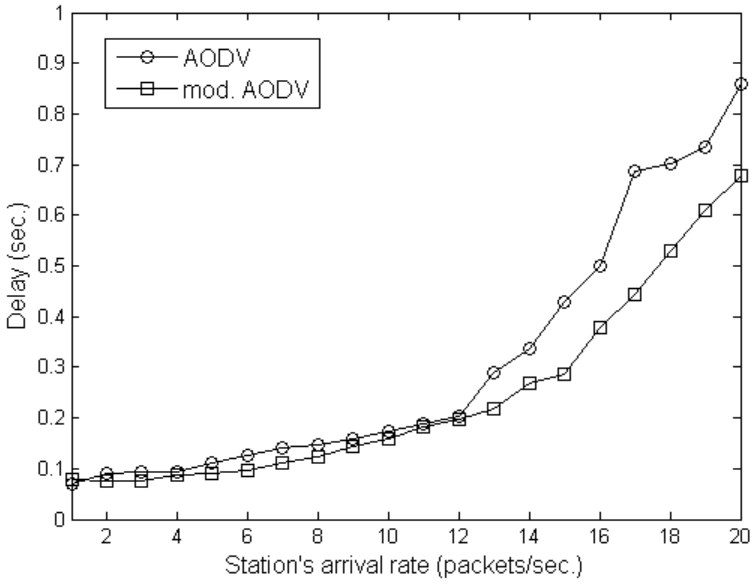

Figure 7.9: the end-to-end packet delay versus the flow rate

7.6 Summary

In this Chapter, we propose a QoS route selection algorithm that utilizes the non-saturation performance to compute the QoS parameters. Based on the calculated performance parameters, intermediate nodes can evaluate the received RREQs and select the best one for broadcasting. The broadcasted RREQ contains the calculated QoS parameters which are updated by intermediate nodes. The destination node uses the QoS information to select a QoS route. The route discovery mechanism in AODV is modified based on our proposed QoS route selection algorithm. Simulation results show that the new protocol has better performance than AODV.

Chapter 8

Conclusion and Future Work

8.1 Conclusion

Throughout this thesis we have used the discrete-time Markov chain process to study the saturation performance of RTS/CTS access mode in imperfect channel conditions. Indeed, the proposed model integrates transmission errors, the backoff countdown process, and transmission retry limits into one model. Also, we used the M/G/1/K queuing system with independent samples from the saturation analysis to analyze the non-saturation performance. After that, we used the non-saturation analysis to propose an approximate analytical model for multi-hop ad hoc networks and we proposed an iterative algorithm for throughput calculation in presence of multi-traffic flows. Finally, the multi-hop analysis helps in proposing a QoS route selection algorithm. The route selection algorithm depends on the feedback information from the MAC layer. Indeed, the MAC layer performs the non-saturation analysis upon receiving a RREQ packet from a previous hop. The results of analysis are used to update the QoS parameters included in the received RREQ. At destination side, the received RREQ carries the end-to-end QoS performance information. Based on this information, the destination can select the QoS route. The proposed algorithm is implemented in AODV routing algorithm. Simulation results show that the QoS route selection algorithm achieves better performance compared to the default one

in AODV. In this work, we show that the quality of route can be enhanced if intermediate nodes evaluate the received RREQs and broadcast the best one. Adapting such approach will enhance the quality of the candidate routes at the destination side. By analyzing the current route discovery mechanism in AODV and evaluating the performance of the possible routes, we noticed that all possible routes, per route discovery cycle, have the same quality because they share the first hops. Our route selection algorithm can provide the destination with different routes with different qualities such that the destination node can select the appropriate one. By measuring the frequency of route failures, we noticed that the QoS route selection algorithm reduces it by 40% to 50%.

8.2 Future Work

Following the previous discussion, it would be interesting to extend our work:

1. To verify the applicability of our analysis for TCP traffic type because in TCP there is a bidirectional communication between nodes. Moreover, collisions may occur between the TCP data packets and TCP-ACK packets.

2. To verify how TCP can use the MAC analysis to enhance the sliding window flow control mechanism.

3. To extend the iterative throughput calculation algorithm in order to determine the maximum load that can be used by the communicating nodes.

4. To see how other QoS parameters such as power and energy can be added to the proposed QoS route selection algorithm.

5. We strongly believe that an experimental validation for our results is necessary.

Bibliography

[1] Ahed M. Alshanyour and Uthman Baroudi. Bypass-AODV: improving performance of ad hoc on-demand distance vector (AODV) routing protocol in wireless ad hoc networks. In *Proceedings of the 1st international conference on Ambient media and systems*, Ambi-Sys '08, pages 17:1–17:8, 2008.

[2] Manoj Pandey, Roger Pack, Lei Wang, Qiuyi Duan, and Daniel Zappala. To repair or not to repair: Helping ad-hoc routing protocols to distinguish mobility from congestion. In *Proceedings of the 26th Annual Joint Conference of the IEEE Computer and Communications Societies*, pages 2311–2315, 2007.

[3] L. Hanzo-II and R. Tafazolli. A survey of QoS routing solutions for mobile ad hoc networks. *IEEE Communications Surveys Tutorials*, 9(2):50–70, 2007.

[4] Jiwa Abdullah. QoS routing solutions for mobile ad hoc network. In *Mobile Ad-Hoc Networks: Protocol Design*, pages 417–454, 2011.

[5] R. Sivakumar, P. Sinha, and V. Bharghavan. CEDAR: a core-extraction distributed ad hoc routing algorithm. *IEEE Journal on Selected Areas in Communications*, 17(8):1454–1465, 1999.

[6] Shigang Chen and K. Nahrstedt. Distributed quality-of-service routing in ad hoc networks. *IEEE Journal on Selected Areas in Communications*, 17(8):1488–1505, 1999.

[7] Swades De, Sajal K. Das, Hongyi Wu, and Chunming Qiao. Trigger-based distributed QoS routing in mobile ad hoc networks. *Mobile Computing and Communications Review*, 6:22–35, 2002.

[8] Ying Ge, T. Kunz, and L. Lamont. Quality of service routing in ad-hoc networks using OLSR. In *Proceedings of the 36th Annual Hawaii International Conference on System Sciences*, page 9 pp., 2003.

[9] Youngki Hwang and Pramod Varshney. An adaptive QoS routing protocol with dispersity for ad-hoc networks. In *in Proceedings of the 36th Hawaii International Conference on System Sciences*, pages 302–311, 2003.

[10] T. Ozugur, M. Naghshineh, P. Kermani, and J.A. Copeland. Fair media access for wireless LANs. In *Proceedings of the IEEE Global Telecommunications Conference*, volume 1B, pages 570–579, 1999.

[11] IEEE 802.11 Working Group. *IEEE Standards Board, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, 2006.

[12] J. Liu and S. Singh. ATCP: TCP for mobile ad hoc networks. *IEEE Journal on Selected Areas in Communications*, 19(7):1300–1315, 2001.

[13] V. Kawadia and P. R. Kumar. A cautionary perspective on cross layer design. *IEEE Journal on Wireless Communications*, 12(1):3–11, 2005.

[14] S. Shakkottai, T. S. Rappaport, and P. C. Karlsson. Cross-layer design for wireless networks. *IEEE Communications magazine*, 41(10):74–80, 2003.

[15] V. Srivastava and M. Motani. Cross-layer design: a survey and the road ahead. *IEEE Communications magazine*, 43(12):112–119, 2005.

[16] Charles E. Perkins and Pravin Bhagwat. Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers. In *Proceedings of the*

conference on Communications architectures, protocols and applications, pages 234–244, 1994.

[17] David B. Johnson, David A. Maltz, and Josh Broch. *Ad Hoc Networking*. Addison-Wesley, 2001.

[18] Charles E. Perkins and Elizabeth M. Royer. Ad-hoc on-demand distance vector routing. In *Proceedings of the Second IEEE Workshop on Mobile Computer Systems and Applications*, WMCSA '99, pages 90–102, 1999.

[19] Xu Yi, Cui Mei, Yang Wei, and Xan Yin. A node-disjoin multipath routing in mobile ad hoc networks. In *Proceedings of the International Conference on Electric Information and Control Engineering*, pages 1067–1070, 2011.

[20] S. Ray, J.B. Carruthers, and D. Starobinski. RTS/CTS-induced congestion in ad hoc wireless LANs. In *Proceedings of the IEEE Wireless Communications and Networking*, volume 3, pages 1516–1521, 2003.

[21] Wen-Kuang Kuo. Energy efficiency modeling for IEEE 802.11 DCF system without retry limits. *Computing and Communications*, 30:856–862, 2007.

[22] Yuanzhu Peter Chen, Jian Zhang, and Anne N. Ngugi. An efficient rate-adaptive MAC for IEEE 802.11. In *Proceedings of the 3rd international conference on Mobile ad-hoc and sensor networks*, MSN'07, pages 233–243, 2007.

[23] H. Jasani and N. Alaraje. Evaluating the performance of IEEE 802.11 network using RTS/CTS mechanism. In *Proceedings of the IEEE International Conference onElectro/Information Technology*, pages 616–621, 2007.

[24] N. Wisitpongphan, O.K. Tonguz, J.S. Parikh, P. Mudalige, F. Bai, and V. Sadekar. Broadcast storm mitigation techniques in vehicular ad hoc networks. *IEEE Wireless Communications*, 14(6):84–94, 2007.

[25] Jian Li and P. Mohapatra. A novel mechanism for flooding based route discovery in ad hoc networks. In *Proceedings of the IEEE Global Telecommunications Conference*, volume 2, pages 692–696, 2003.

[26] S.H. Shah and K. Nahrstedt. Predictive location-based QoS routing in mobile ad hoc networks. In *Proceedings of the IEEE International Conference on Communications*, volume 2, pages 1022–1027, 2002.

[27] P. Jacquet, P. Muhlethaler, T. Clausen, A. Laouiti, A. Qayyum, and L. Viennot. Optimized link state routing protocol for ad hoc networks. In *Proceedings of the IEEE International Multi Topic Conference*, pages 62–68, 2001.

[28] Chai-Keong Toh. Associativity-based routing for ad hoc mobile networks. *Wireless Personal Communications*, 4:103–139, 1997.

[29] R. Dube, C.D. Rais, Kuang-Yeh Wang, and S.K. Tripathi. Signal stability-based adaptive routing (SSA) for ad hoc mobile networks. *IEEE Personal Communications*, 4(1):36–45, 1997.

[30] S.-J. Lee and M. Gerla. Dynamic load-aware routing in ad hoc networks. In *Proceedings of the IEEE International Conference on Communications*, volume 10, pages 3206–3210, 2001.

[31] K. Wu and J. Harms. Load-sensitive routing for mobile ad hoc networks. In *Proceedings of the 10th International Conference on Computer Communications and Networks*, pages 540–546, 2001.

[32] Chai Keong Toh, Anh-Ngoc Le, and You-Ze Cho. Load balanced routing protocols for ad hoc mobile wireless networks. *IEEE Communications Magazine*, 47(8):78–84, 2009.

[33] Julien Cartigny, Franois Ingelrest, and David Simplot. RNG relay subset flooding protocols in mobile ad-hoc networks. *International Journal of Foundations of Computer Science*, 14(2):253–265, 2003.

[34] Ning Li, J.C. Hou, and L. Sha. Design and analysis of an MST-based topology control algorithm. *IEEE Transactions on Wireless Communications*, 4(3):1195–1206, 2005.

[35] Lefteris M. Kirousis, Evangelos Kranakis, Danny Krizanc, and Andrzej Pelc. Power consumption in packet radio networks. *Theoretical Computer Science*, 243:289–305, 2000.

[36] A. Clementi, P. Penna, and R. Silvestri. The power range assignment problem in radio networks on the plane. In *Proceedings of the 17th Symposium on Theoretical Computer Science*, pages 125–140, 2000.

[37] J.E. Wieselthier, G.D. Nguyen, and A. Ephremides. On the construction of energy-efficient broadcast and multicast trees in wireless networks. In *Proceedings of the IEEE 19th Annual Joint Conference of the IEEE Computer and Communications Societies*, volume 2, pages 585–594, 2000.

[38] J. Cartigny, D. Simplot, and I. Stojmenovic. Localized minimum-energy broadcasting in ad-hoc networks. In *Proceedings of the IEEE 22nd Annual Joint Conference of the IEEE Computer and Communications*, volume 3, pages 2210–2217, 2003.

[39] Young-Bae Ko and Nitin H. Vaidya. Location-aided routing (LAR) in mobile ad hoc networks. *Wireless network*, 6:307–321, 2000.

[40] Hyojun Lim and Chongkwon Kim. Multicast tree construction and flooding in wireless ad hoc networks. In *Proceedings of the 3rd ACM international workshop on Modeling, analysis and simulation of wireless and mobile systems*, MSWIM '00, pages 61–68, 2000.

[41] Wei Peng and Xicheng Lu. AHBP: An efficient broadcast protocol for mobile ad hoc networks. *Journal of Computer Science and Technology*, 16:114–125, 2001.

[42] T. Camp, J. Boleng, B. Williams, L. Wilcox, and W. Navidi. Performance comparison of two location based routing protocols for ad hoc networks. In *Proceedings of the 21st Annual Joint Conference of the IEEE Computer and Communications Societies*, volume 3, pages 1678–1687, 2002.

[43] Wei Peng and Xi-Cheng Lu. On the reduction of broadcast redundancy in mobile ad hoc networks. In *Proceedings of the 1st Annual Workshop on Mobile and Ad Hoc Networking and Computing*, pages 129–130, 2000.

[44] A. Qayyum, L. Viennot, and A. Laouiti. Multipoint relaying for flooding broadcast messages in mobile wireless networks. In *Proceedings of the 35th Annual Hawaii International Conference on System Sciences*, pages 3866–3875, 2002.

[45] Jian Li and P. Mohapatra. LAKER: location aided knowledge extraction routing for mobile ad hoc networks. In *Proceedings of the IEEE Wireless Communications and Networking*, volume 2, pages 1180–1184, 2003.

[46] Q. Jiang, R. Finkel, D. Manivannan, and M. Singhal. RPSF: A routing protocol with selective forwarding for mobile ad-hoc networks. *Wireless Personal Communications*, 43:411–436, 2007.

[47] G. Bianchi. Performance analysis of the IEEE 802.11 distributed coordination function". *IEEE Journal on Selected Areas in Communications*, 18(3):535–547, 2000.

[48] P. Chatzimisios, A.C. Boucouvalas, and V. Vitsas. Performance analysis of IEEE 802.11 DCF in presence of transmission errors. In *Proceedings of the IEEE International Conference on Communications*, volume 7, pages 3854–3858, 2004.

[49] Haitao Wu, Yong Peng, Keping Long, Shiduan Cheng, and Jian Ma. Performance of reliable transport protocol over ieee 802.11 wireless lan: analysis and enhancement. In *Proceedings of the 21st Annual Joint Conference of the IEEE Computer and Communications Societies*, volume 2, pages 599–607, 2002.

[50] G. Bianchi and I. Tinnirello. Remarks on IEEE 802.11 DCF performance analysis. *IEEE Communications Letters*, 9(8):765–767, 2005.

[51] A. Kumar, E. Altman, D. Miorandi, and M. Goyal. New insights from a fixed-point analysis of single cell IEEE 802.11 WLANs. *IEEE/ACM Transactions on Networking*, 15(3):588–601, 2007.

[52] Yun Han Bae. Analysis of IEEE 802.11 non-saturated DCF by matrix analytic methods. *Annals of Operations Research*, 2008.

[53] K. Ghaboosi, B.H. Khalaj, Yang Xiao, and M. Latva-aho. Modeling IEEE 802.11 DCF using parallel space time markov chain. *IEEE Transactions on Vehicular Technology*, 57(4):2404–2413, 2008.

[54] Ren Ping Liu, G.J. Sutton, and I.B. Collings. A new queueing model for qos analysis of IEEE 802.11 DCF with finite buffer and load. *IEEE Transactions on Wireless Communications*, 9(8):2664–2675, 2010.

[55] David Malone, Ken Duffy, and Doug Leith. Modeling the 802.11 distributed coordination function in non-saturated heterogeneous conditions. *IEEE/ACM Transactions on Networking*, 15(1):159–172, 2007.

[56] O. Tickoo and B. Sikdar. Modeling queueing and channel access delay in unsaturated IEEE 802.11 random access MAC based wireless networks. *IEEE/ACM Transactions on Networking,*, 16(4):878–891, 2008.

[57] A.N. Zaki and M.T. El-Hadidi. Throughput analysis of IEEE 802.11 DCF under finite load traffic. In *1st International Symposium on Control, Communications and Signal Processing*, pages 535–538, 2004.

[58] Jd.P. Pavon and Sunghyun Choi. Link adaptation strategy for IEEE 802.11 WLAN via received signal strength measurement. In *IEEE International Conference on Communications*, volume 2, pages 1108–1113, 2003.

[59] The Network Simulator NS-2. http://www.isi.edu/nsnam/ns/.

[60] Gion Reto Cantieni, Qiang Ni, Chadi Barakat, Thierry Turletti, Plante Group, and Inria Sophia Antipolis. Performance analysis under finite load and improvements for multirate 802.11. *Elsevier Computer Communications*, pages 1095–1109.

[61] K. Duffy, D. Malone, and D.J. Leith. Modeling the 802.11 distributed coordination function in non-saturated conditions. *IEEE Communications Letters*, 9(8):715–717, 2005.

[62] O. Tickoo and B. Sikdar. Modeling queueing and channel access delay in unsaturated IEEE 802.11 random access mac based wireless networks. *IEEE/ACM Transactions on Networking*, 16(4):878–891, 2008.

[63] Ashikur Rahman University, Ashikur Rahman, and Pawel Gburzynski. Hidden problems with the hidden node problem. In *Proceedings of the 23rd Biennial Symposium on Communications*, pages 270–273, 2006.

[64] Fangqin Liu, Chuang Lin, Hao Wen, and Peter Ungsunan. Throughput analysis of wireless multi-hop chain networks. In *Proceedings of the IEEE/ACIS International Conference on Computer and Information Science*, pages 834–839, 2009.

[65] K. Medepalli and F.A. Tobagi. Throughput analysis of IEEE 802.11 wireless LANs using an average cycle time approach. In *Proceedings of the IEEE Global Telecommunications Conference*, volume 5, pages 3007–3011, 2005.

[66] R. Khalaf, I. Rubin, and Julan Hsu. Throughput and delay analysis of multihop IEEE 802.11 networks with capture. In *Proceedings of the IEEE International Conference on Communications*, pages 3787–3792, 2007.

[67] Guoqiang Mao, Lixiang Xiong, and Xiaoyuan Ta. On the maximum throughput of a single chain wireless multi-hop path. In *Proceedings of the 3rd International Conference on Communications and Networking*, pages 398–402, 2008.

[68] J.H. Li, Song Luo, Wei Tang, R. Levy, and Kihong Park. Heavy-tailed workload aware ad hoc routing. In *Proceeding of the IEEE International Conference on Communications*, pages 2437–2442, 2008.

[69] Jinyang Li, Charles Blake, Douglas S.J. De Couto, Hu Imm Lee, and Robert Morris. Capacity of ad hoc wireless networks. In *Proceedings of the 7th annual international conference on Mobile computing and networking*, MobiCom '01, pages 61–69, 2001.

[70] Ping Chung Ng and Soung Chang Liew. Throughput analysis of IEEE802.11 multi-hop ad hoc networks. *IEEE/ACM Transactions on Networking*, 15(2):309–322, 2007.

[71] Sungoh Kwon and Ness B. Shroff. Analysis of shortest path routing for large multi-hop wireless networks. *IEEE/ACM Transactions on Networking*, 17:857–869, 2009.

[72] R. Asokan. A review of quality of service (QoS) routing protocols for mobile ad hoc networks. In *Proceedings of the International Conference on Wireless Communication and Sensor Computing*, pages 1–6, 2010.

[73] T.A. Alghamdi and M.E. Woodward. QoS algorithm for localised routing based on bandwidth as the dominant metric for candidate path selection. In *Proceedings of the IEEE 10th International Conference on Computer and Information Technology*, pages 321–328, 2010.

[74] S. Ziane and A. Mellouk. Inductive routing based on dynamic end-to-end delay for mobile networks. In *Proceedings of the IEEE Global Telecommunications Conference*, pages 1–5, 2010.

[75] Qi Xue and Aura Ganz. Ad hoc QoS on-demand routing (AQOR) in mobile ad hoc networks. *Parallel and Distributed Computing*, 63:154–165, 2003.

[76] Satyabrata Chakrabarti and Amitabh Mishra. Quality of service challenges for wireless mobile ad hoc networks: Research articles. *Wireless Communication and Mobile Computing*, 4:129–153, 2004.

[77] Baoxian Zhang and H.T. Mouftah. QoS routing for wireless ad hoc networks: problems, algorithms, and protocols. *Communications Magazine, IEEE*, 43(10):110–117, 2005.

[78] Jae-Hwan Chang and L. Tassiulas. Energy conserving routing in wireless ad-hoc networks. In *Proceedings of the IEEE 19th Annual Joint Conference of the IEEE Computer and Communications Societies*, volume 1, pages 22–31, 2000.

[79] Sheetalkumar Doshi, Shweta Bhandare, and Timothy X Brown. An on-demand minimum energy routing protocol for a wireless ad hoc network. *Mobile Computing and Communications Review*, 6:50–66, 2002.

[80] Chung wei Lee. Energy efficiency of QoS routing in multi-hop wireless networks. In *Proceedings of the IEEE International Conference on Electro/Information Technology*, pages 1–6, 2010.

[81] C. Toh. Maximum Battery Life Routing to Support Ubiquitous Mobile Computing in Wireless Ad Hoc Networks. *IEEE Communications Magazine*, 39(6):138–147, 2001.

[82] Chansu Yu, Ben Lee, and Hee Y. Youn. Energy efficient routing protocols for mobile ad hoc networks. *Wireless Communications and Mobile Computing*, 3(8):959–973, 2003.

[83] N. Wisitpongphan, G. Ferrari, S. Panichpapiboon, J.S. Parikh, and O.K. Tonguz. QoS provisioning using BER-based routing in ad hoc wireless networks. In *Pro-*

ceedings of the IEEE 61st Vehicular Technology Conference, volume 4, pages 2483–2487, 2005.

[84] D. Djenouri and I. Balasingham. Power-aware QoS geographical routing for wireless sensor networks. In *Proceedings of the 6th IEEE International Conference on Distributed Computing in Sensor Systems Workshops*, pages 1–5, 2010.