ROUTING IN DISTRIBUTED WIRELESS MESH NETWORK

by

Subrata Saha

A dissertation submitted in partial fulfillment of the
requirements for the degree of

Master of Science

Department of Electrical and Computer Engineering
Faculty of Graduate Studies
University of Manitoba

THE UNIVERSITY OF MANITOBA

FACULTY OF GRADUATE STUDIES

*****

COPYRIGHT PERMISSION

ROUTING IN DISTRIBUTED WIRELESS MESH NETWORK

BY

Subrata Saha

A Thesis/Practicum submitted to the Faculty of Graduate Studies of The University of Manitoba in partial fulfillment of the requirement of the degree

MASTER OF SCIENCE

Abstract

As the Internet continues to gain in popularity, demand for broadband access has outpaced the wired infrastructure in many areas. To meet the needs of wireless broadband access, the IEEE 802.16 protocol for wireless metropolitan area networks has been recently standardized. The medium access control (MAC) layer of this protocol has point-to-multipoint (PMP) mode and multipoint-to-multipoint or mesh mode.

Wireless mesh network consists of mesh routers and mesh clients, where mesh routers form the backbone of the mesh network. Wireless mesh networks are anticipated to resolve the limitations and to significantly improve the performance of ad-hoc networks, wireless local area networks, wireless metropolitan area networks and wireless personal area networks.

Wireless mesh network can work in distributed system, where there is no central controller to manage the nodes in the network. Thus scheduling the nodes for packet transmission and routing packets in the network are two big challenges to the researchers.

In this thesis we have introduced a new routing method that suggests how a path can be selected to ensure packet transmission in minimum time, when multiple paths are available to a same destination.

Acknowledgements

First, I would like to express my deepest gratitude to my thesis supervisor Dr. Attahiru Sule Alfa for his guidance, encouragement, patience and inspiration. Dr. Alfa introduced me to the area of research in routing in wireless mesh network. Without his guidance this research could not be accomplished. I would like to thank also Dr. Jun Cai for his suggestions and guidance during my research.

I would like to thank Dr. Alfa again for supporting me with Research Assistantship throughout my studies.

I am also thankful to the thesis committee members, Dr. Jun Cai and Dr. Vojislav Misic, for being in my thesis committee.

I convey my cordial thanks to my friends and colleagues Arash Abadpour, Femi Adelani, Haitham Abu Ghazaleh, Dr. Jinting Wang, Md. Mostafizur Rahman and Sangho Lee for their suggestions during my research work.

Finally I express my heartfelt gratitude to my parents and my friend Md. Mostafizur Rahman, who encouraged me to come here and gave me support during my research.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Wireless networks play a major role in communication systems in present days. One of the main objectives of next generation wireless networks is to provide broadband internet service to end-users. There are several IEEE (Institute of Electrical and Electronics Engineers) standards for wireless communications. IEEE 802.11, IEEE 802.15, IEEE 802.16 and IEEE 802.20 are different wireless protocols used in different levels. According to the coverage area and deployment level the standards can be ordered as follows:

IEEE 802.15, which is used to implement wireless personal area network (WPAN),

IEEE 802.11, used in wireless local area network (WLAN),

IEEE 802.16, used in wireless metropolitan area network (WMAN) and

IEEE 802.20, which is used in wireless wide area network (WWAN) [29].

1.1 WiFi and Wimax

IEEE 802.11 standard is used as wireless local area network. It is also known as wireless fidelity (WiFi) [1, 12]. WiFi is the wireless way to handle networking. The main advantage of WiFi is its simplicity. WiFi enabled computers anywhere in a home or office can be connected to internet without the need of wires. Computers connect to the

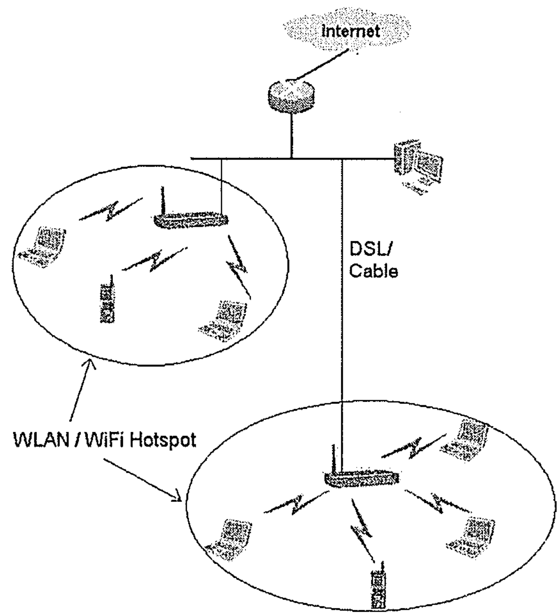

Figure 1.1: How WiFi Works

network using radio signals. The coverage area of this system is small (only 100 meters radius). These small WiFi enabled areas are called WiFi hotspots. In each hotspot there is a wireless router, which acts as the base station (BS) of the area, and provides wireless connections to the computers. Digital subscriber line (DSL), cable modem are used to connect these routers to internet service provider (ISP). Thus, a wired system is used to connect the WiFi hotspots. In other words, the backbone network of WiFi hotspots is a wired system. Such a scenario is shown in Figure 1.1.

IEEE 802.16 focusses on the last mile applications of wireless technology for broadband access. This standard is called worldwide interoperability for microwave access (WiMax). WiMax is actually wireless MAN technology that can connect WiFi hotspots to each other and to other parts of the internet. Such a scenario is shown in Figure 1.2 (figure modified from [21]). WiMax provides a wireless alternative to cable and DSL for last mile broadband access. This technology is less expensive and easier to deploy compared to DSL / cable. In another sense WiMax operates in a fashion similar to WiFi, but

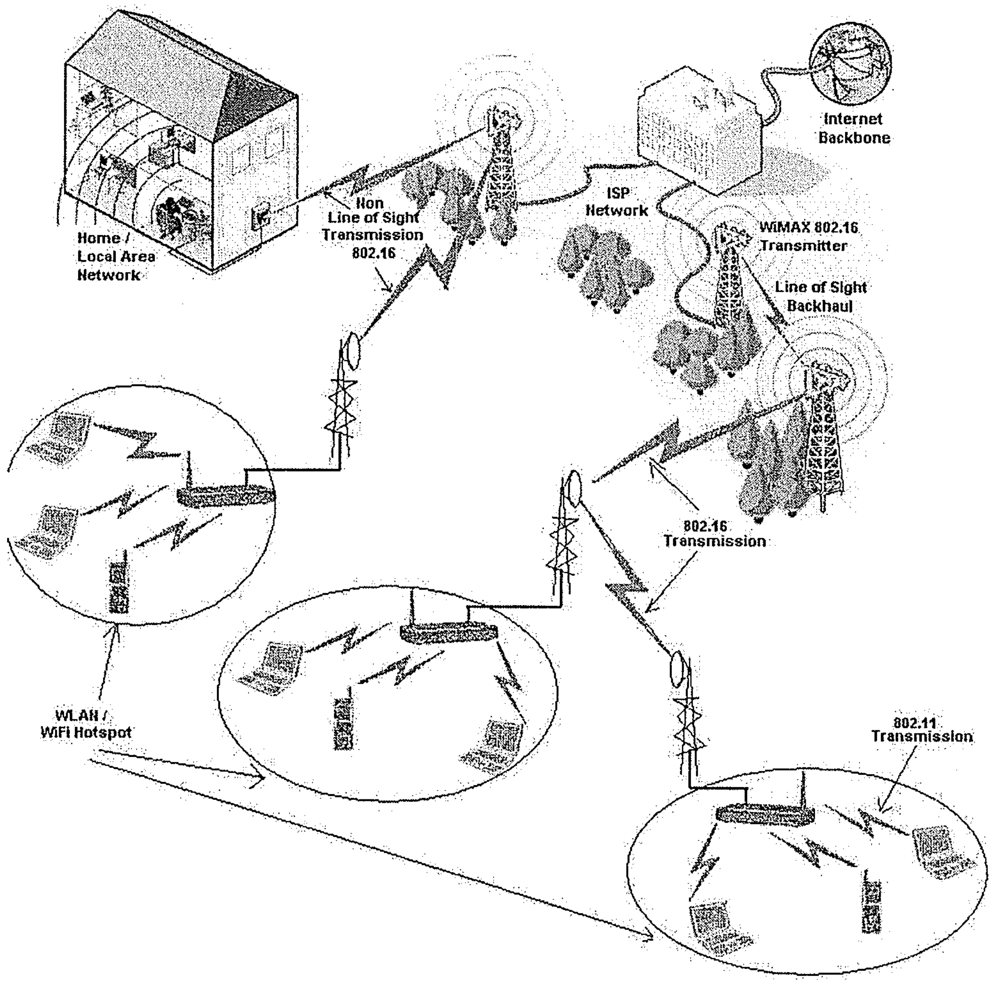

Figure 1.2: How WiMAX Works

at higher speeds, over greater distances and for a greater number of users [12, 23]. In WiMax, BS provides wireless connection to the subscriber stations (SS's). BS can be connected directly to the internet using a high-bandwidth wired connection. It can also be connected to another BS using direct line-of-sight microwave link.

1.2 Point-to-Multipoint and Mesh Networks

The IEEE 802.16 MAC protocol has two modes of operation: point-to-multipoint (PMP) mode and multipoint-to-multipoint (mesh) mode. In PMP mode, the nodes are organized into a cellular like structure, consisting of a BS and some SS's. The channels are divided into uplink (from SS to BS) and downlink from (BS to SS), both shared among the SS's. This type of network requires all SS's to be within the transmission range of BS. IEEE 802.16 operates under the frequency band of 10 GHz to 66 GHz. With this high frequency the BS can cover a large distance up to 50 km. Due to the high frequency all the SS's need to be on the direct line-of-sight to the BS. But buildings, trees, hills and other obstacles often make line-of-sight difficulty in many neighborhoods. To overcome this problem IEEE 802.16a was evolved which focuses on the spectrum of 2 GHz to 11 GHz [29]. With this lower frequency WiMax can provide connection to users within 8 km without direct line-of-sight. Thus in PMP mode if a BS has to cover a large distance, then all the SS's should be on direct line-of-sight with the BS. And line-of-sight problem can be overcome by using lower frequency band, but then the BS can cover only small area.

On the other hand, in the mesh mode, the nodes are organized in an ad-hoc fashion. All stations are peers and each node can act as routers to relay packets to its neighbors. There are still some nodes which provide the functions of BS for connecting the mesh network to backhaul links. However, there is no need to have direct link from the BS to each of the SS's in a mesh network. In mesh network traffic can be routed around obstacles over multiple hops and thus can avoid the requirement of multiple BS's. This makes the coverage of a residential area less expensive. Mesh network is more flexible and cost effective compared to PMP network when extending broadband services to a mass residential market. New SS's are allowed to join the network even if they are out of range, or have no direct connectivity with the BS. Each SS works as a router and becomes a part of the infrastructure [24].

1.3 Route Selection in Mesh Network

IEEE 802.16 mesh network can operate either in centralized scheduling or in distributed scheduling. In centralized scheduling the BS schedules the transmission of all the nodes within the mesh. In distributed mesh BS does not control the scheduling and routing of any SS. Each SS makes its own routing when it sends packet to another node. When there is only one route from a source node to a destination node then the source has to follow that route. But if there are multiple routes from a source to a destination then the source has multiple options to choose a route to send its packet. When a source finds more than one route to a destination, it can also distribute the packets over multiple routes, i.e., a source can distribute its load over the available routes. An intelligent technique for balancing load over multiple routes can minimize the total transmission time.

1.4 Thesis Organization

This thesis addresses a load balancing technique for distributed wireless mesh network. The thesis is organized as follows.

- In this chapter we have discussed the basic wireless networking system for end users.
- We describe some basic routing mechanism and some related works on routing in wireless networks in chapter two.
- In chapter three we have elaborated our load balancing technique. In order to do this first we have considered a small network.
- Then we have extended the work to larger networks in chapter four.

- We present our conclusion and future work in chapter five.
- We have used some mathematical notations and formulas to generate some combinations. Those formulas have been clarified in appendix section.

Chapter 2

Related Works on Routing

2.1 Routing

Routing is the process of finding a path from a source to a destination in a network. This is accomplished by means of routing protocols, which are established by mutually consistent routing tables in every router in the network. Routing is categorized in many ways. Some commonly available choices of routing algorithms for different types of networks are discussed in [17, 22, 26].

Routing protocols in conventional wired networks generally use either distance vector routing algorithm or link state routing algorithm. In both of them all the routers periodically broadcast some information, called routing advertisement, to their neighbors. In distance vector routing each router broadcasts its view of the distance to all the hosts and each router computes the shortest path towards all the hosts based on the information advertised by their neighbors. In link state routing each router broadcasts to all other routers in the network its own view of the status of its adjacent network links. And then each router computes its own routing table. In addition to its use in wired networks, basic distance vector routing is used in wireless networks [15, 25, 28].

2.1.1 Different Types of Routing

Centralized Routing and Distributed Routing

In centralized routing a central processor is responsible for routing of all the nodes in the network. The central processor collects information about all the links and processes the information to compute a routing table for every node. Then it distributes the routing tables to the routers. Centralized routing is reasonable in a centrally administrated network.

In distributed routing the routers exchange some message among them and based on these information they build mutually consistent routing tables. If the network is too large then distributed routing is necessary.

Source-based Routing and Hop-by-hop Routing

In source-based routing the source decides the complete path (that is, the sequential list of routers on the path from source to destination) of the packet to the destination and set the path in the packet header. This method allows the sender to specify a packet's path precisely. But the sender needs to be aware of the entire network. Again, if a link goes down after the packet is dispatched from the sender then the packet cannot reach to its destination. In this method the packet header becomes larger.

In hop-by-hop routing the packet header contains only the address of its final destination. Each router along the path can choose the next hop. All the nodes do not need to be aware of the whole network. This method is necessary if the network state changes over time.

Stochastic Routing and Deterministic Routing

In deterministic routing each router tries to forwards packets towards a destination along a fixed path. In stochastic routing [18] routers maintain more than one next hop for each possible destination. It picks up a path randomly before it forwards a packet. Stochastic routing cannot guarantee that a series of packets will reach the destination in order.

Single-path Routing and Multiple-path Routing

In single-path routing each router maintains exactly one next hop for each destination. In multiple-path routing routers maintain more than one next hop for a destination. The paths might be sorted according to some order (such as number of hops, propagation delay, etc.). If the path on the top of the list is unavailable then the router forwards the packet along the next available path. If such order is not maintained in multiple-path routing and the next hop is picked up randomly, then it becomes stochastic routing.

State-dependent Routing and State-independent Routing

State-dependent routing is a dynamic method. In this type of routing the router chooses the next hop depending on the present state of the network. For example, if some links on a path are heavily loaded, then router may try to send packet through another route. On the other hand state-independent routing is a static method. For example, shortest path routing is state-independent routing. State-dependent routing usually finds better routes to a destination. But it requires more overhead for monitoring the network.

2.1.2 Load Balancing in Routing: Our Research Goal

In our model (discussed in Chapter 3 and onward), when a number of packets are to be sent to a destination, the sender tries to complete the total transmission in minimum time. Sending a series of packets through the shortest path cannot guarantee to make the transmission in possible shortest time. Thus the sender distributes the packets among the available routes to the destination. According to the routing techniques mentioned above, our model can be applied in distributed, hop-by-hop, multiple-path, stochastic, state-dependent routing.

2.2 Related Works on Routing in Wireless Mesh Network

In recent years the widespread availability of wireless communication and handheld devices has stimulated research on self-organizing networks, which do not require a preestablished infrastructure. These ad-hoc networks [20] consist of autonomous nodes that collaborate in order to transport information. Usually these nodes act as end systems (individual users / subscribers) and routers simultaneously. There are two types of ad-hoc networks: static ad-hoc networks and mobile ad-hoc networks. In static ad-hoc networks the position of a node usually does not change once it has become a part of the network. Rooftop network [5], community wireless networks [3, 16, 27, 31] are examples of static ad-hoc network. As a relatively new standard, IEEE-802.16 has been studied much less than other standards like IEEE 802.11. Routing methods applied in static ad-hoc networks are mostly used in wireless mesh network.
The shortcomings of shortest-path routing have been discussed by many researchers in [32, 4, 13, 7, 10, 11]. In [4] Awerbuch *et al.* have proposed an algorithm that selects route with the highest throughput in multi-rate ad-hoc network. Dube *et al.* [10] have proposed a method that selects a route in ad-hoc network with stable signal level on the wireless links.

Interference-Aware Routing

Wei *et al.* proposed an interference-aware routing algorithm for IEEE 802.16 centralized mesh network in [30]. They proposed an interference-aware research framework to improve spectral utilization. Using the framework they introduced an interference-aware route construction algorithm to improve the network throughput by selecting routes with minimal interference to existing nodes.

Routing in Multi-Radio Multi-Hop Network

Couto *et al.* [8] proposed a new metric called ETX (Expected Transmission Count) for routing in multi-radio multi-hop wireless networks with stationary nodes. ETX measures the expected number of transmissions (transmission and retransmissions) to send a packet over a link. ETX is a function which estimates the probability of packet transmission failure on a link.

Draves *et al.* [9] proposed another metric WCETT (Weighted Cumulative Expected Transmission Time). Their method assigns weights on each link based on the expected transmission time over that link. They have shown that when nodes are equipped with multiple heterogeneous radios, then selecting channel diverse paths provides high throughput.

Multiple-path Routing

Lee *et al.* [19] proposed an algorithm that utilizes a mesh structure to provide multiple alternate paths from a source to a destination in ad-hoc networks without producing additional control message. They have shown that having multiple alternate paths in ad-hoc networks is beneficial since wireless networks are prone to router breaks because of fading environment, packet collisions, signal interference and high error rate. Maintaining multiple paths and distributing traffic can minimize the total number of required transmissions.

Routing in Mobile Ad-hoc Networks

Routing in mobile ad-hoc networks is more complex compared to routing in static ad-hoc network because in mobile ad-hoc network the routers are moving. Johnson *et al.* [14]

presented a protocol for routing in wireless mobile hosts. Instead of using distant vector routing, their protocol uses dynamic source routing of packets between hosts that want to communicate.

Chapter 3

Load Balancing

3.1 Background of IEEE 802.16 Mesh Mode

The IEEE 802.16 standard is designed to evolve as a set of air interfaces based on a common MAC protocol with physical layer specifications dependent on the spectrum used [6]. In centralized scheduling the BS schedules the transmission of all the nodes within the mesh and the mesh BS is responsible for collecting bandwidth requests from the SS's and for managing resource allocations. In this scheme each SS estimates and sends its resource request to the BS. BS determines the amount of granted resource and sends the grant message to the SS. In this procedure transmissions are coordinated to ensure collision-free scheduling typically in a more optimal manner than distributed scheduling. Centralized scheduling procedure is relatively simple compared to distributed scheduling. However, the connection setup delay is long in centralized scheduling. Centralized scheduling is not suitable for occasional traffic needs.

In distributed scheduling every node computes its transmission time without any global information. This technique is more complex than centralized scheduling. The IEEE 802.16 mesh frame in distributed scheduling is divided into control and data subframes. Data subframe follows control subframe in a frame. There are two types of control subframes: network control and schedule control. By transmitting control subframe, nodes maintain their schedule and data subframe allocation in the neighborhood. Data subframes are allocated based on a request-grant-confirm three-way hand shaking among the nodes.

The distributed scheduling of mesh mode operation can be of two types: coordinated distributed scheduling and uncoordinated distributed scheduling. The uncoordinated distributed scheduling adopts a simple contention approach where collisions may occur if multiple nodes try to transmit at the same control transmission opportunity. This scheduling scheme is only suitable for links with occasional or brief traffic needs.

On the other hand, the coordinated distributed scheduling scheme is contention free. In this scheme nodes exchange 2-hop neighboring schedule information with each other. In coordinated distributed scheduling all nodes compete for channel access using a pseudo random election algorithm. Each node knows about the two hop neighbors' scheduling. Since nodes run the election algorithm independently, a common algorithm is used by each node in a neighborhood [2, 6, 33]. The algorithm is random but predictable. That is why this is called pseudo random election algorithm. The randomness and predictability are achieved by using a common rule for all the nodes for construction of seeds for random number.

3.2 Problem Definition

Let us consider the network shown in Figure 3.1. We are considering the problem where node-1 wants to send some packets to node-2. It can select two possible routes (route 1-3-2 and route 1-4-2). First let us consider the case where only node-1 transmits packets to node-2. Node-3 and node-4 only forward packets of node-1 to node-2. Node-2 does not transmit any packet.

We assume that after successful transmission of each packet the receiver sends an ac-

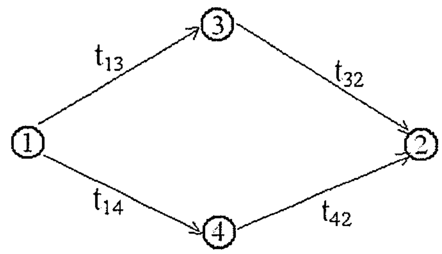

Figure 3.1: A 4-node network

knowledgement to the sender. Then the sender transmits the next packet. Let t_{ab} be the expected required time for the successful transmission of a packet from node-a to node-b. A successful transmission from node-a to node-b includes sending a packet from node-a to node-b and then sending a positive acknowledgement from node-b to node-a. Due to transmission errors any packet may need to be retransmitted. Let us assume that node-a is transmitting x packets to node-b. Let total y number of retransmissions take place due to transmission error. Then $t_{ab} =$ (time for x transmissions + time for y retransmissions + time for all acknowledgements $)/x$.

For simplicity we assume that due to interference only one node can transmit at a time. We are not considering contention during packet transmission in the system. If we consider the situation that no node is transmitting / forwarding any packet and there is no packet queued for transmission at any node, i.e., all the buffers of all the nodes are empty, in that case if node-1 sends a packet to node-2 through node-3, the time associated with this path $T_1' = t_{13} + t_{32}$. Similarly if the packet is sent through node-4, the expected required time $T_2' = t_{14} + t_{42}$. Let $T_1' < T_2'$. Let us further assume that node-1 wants to transmit n packets to node-2. If it sends the packets using route 1-3-2, then the expected time for transmission is nT_1' and if it selects route 1-4-2, then the expected time is nT_2'. If n_1 packets are sent through route 1-3-2 and n_2 packets are sent through route 1-4-2 $(n_1+n_2 = n)$, since node-3 and node-4 cannot transmit packets simultaneously, expected time for transmission of all these n packets, $T_3' = n_1T_1' + n_2T_2'$. Clearly $T_3' > T_1'$. That is,

$T_1' < T_2', T_3'$. Thus, in this case routing packet through the shortest path takes minimum time.

Now let us add some more nodes to the network in Figure 3.1, resulting in a new network, as shown in Figure 3.2. Like the previous case, here node-1 sends packets to node-2. In addition to that, node-5 and node-6 send packets to each other. These packets are forwarded by node-3. Also node-7 and node-8 send packets to each other. These packets are forwarded by node-4. Node-2 also sends packets to node-1 and these packets can be forwarded by node-3 and node-4. Thus node-3 forwards the following 4 types of packets:

1. packet of node-1 to node-2
2. packet of node-2 to node-1
3. packet of node-5 to node-6
4. packet of node-6 to node-5

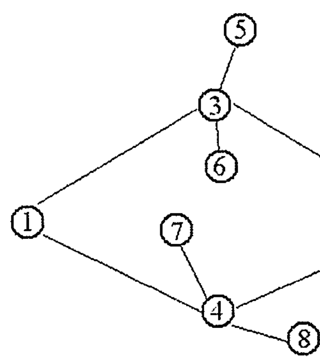

Figure 3.2: A 8-node network

Similarly, node-4 also forwards 4 types of packets.

As in the previous case, here $T_1' = t_{13} + t_{32}$ and $T_2' = t_{14} + t_{42}$. We also assume that $T_1' < T_2'$.

For simplicity we assume that due to interference only one node of node-1, node-2, node-3 and node-4 can transmit at a time. i.e., when node-1 transmits, node-2, 3 and 4 cannot transmit. Media is shared among these 4 nodes (1,2,3 and 4). The nodes compete for media and one of them gets the access. Let us call the time duration (time slot) TM_a^i when node-a uses the media for the ith time (Figure 3.3). TM_a^i may be different for different values a and i. Consider a long finite number L and let TS_a^L be the total time

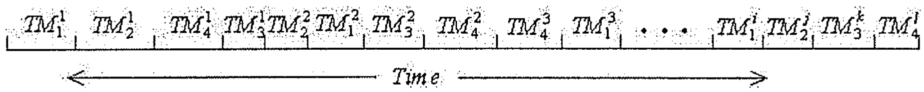

Figure 3.3: Transmission opportunity

that node-a has used the media. Then $TS_a^L = \sum_{i=1}^{L} TM_a^i$. Let m_a be the fraction of time that node-a gets access over media. Then $m_{a(1 \leq a \leq 4)} = TS_a^L / \sum_{j=1}^{4} TS_j^L | L \rightarrow \propto$. $m_1 + m_2 + m_3 + m_4 = 1$.

If all the packets of node-1 are transmitted through route 1-3-2 then the packets of node-1 are forwarded by node-3 only. In that case the packets get only some of the TM_3 slots of time to be forwarded (these TM_3 time slots are used by node-3 to forward packets, some of the slots are used to forward the packets of node-1 to node-2, other slots are used to forward other packets). If some packets are transmitted through route 1-4-2, then the packets get some of TM_4 time slots in addition to some of TM_3 time slots. Thus, transmitting some packets through route 1-3-2 and transmitting the other packets through route 1-4-2 will require less time compared to transmitting all the packets through only the shortest path (route 1-3-2). In the previous case (as shown in Figure 3.1) if node-1 sends all the packets through the shortest path (route 1-3-2) then it takes minimum time for transmission. However, in this situation (as shown in Figure 3.2), if the packets are divided between the two routes then it takes minimum time for transmission. The reason for this difference is that, in previous case if no packet is transmitted through the path 1-4-2, then media is shared by only node-1 and node-3. However, in this case, since node-4 has to forward other packets, media is shared among all the 4 nodes regardless whether node-4 forwards packets of node-1 or not.

3.3 Problem Modeling

Let us consider the network as shown in Figure 3.2. We already know that node-3 forwards 4 types of packets: packet of node-1 to node-2, packet of node-2 to node-1, packet of node-5 to node-6 and packet of node-6 to node-5. We consider the situation that node-1 got some packets to send to node-2. Let us assume that node-1 is not transmitting any packet and it is going to start transmission. Assume that meanwhile other nodes (node-2, node-5 and node-6) are transmitting their packets. This situation as seen by node-1 is represented in Figure 3.4. From the view point of node-1 we now analyze the situation.

Let λ_{ab} be the arrival rate of the packets from node-a to node-b. Thus λ_{23} is the arrival rate of packets from node-2 to node-3. Let λ_3 be the combined arrival rate of packets at node-3 from node 5, 6 and 2. Then, $\lambda_3 = \lambda_{53} + \lambda_{63} + \lambda_{23}$.

Let μ_{ab} be the transmission rate of packet at node-a, which is destined to node-b. The

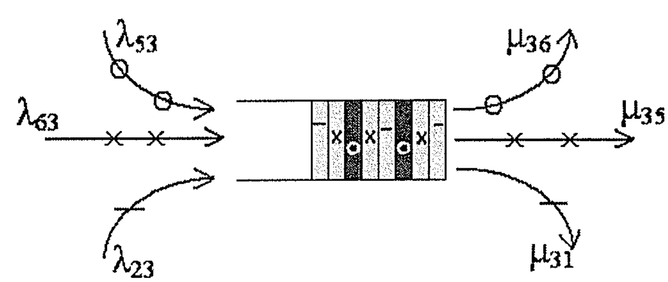

Figure 3.4: Buffer of node-3 of the network shown in Figure 3.2

packet that comes from node-2 is forwarded to node-1. The arrival rate of these packets at node-3 is λ_{23} and the transmission rate of these packets from node-3 is μ_{31}. The same scenario stands for (λ_{53}, μ_{36}) pair and (λ_{63}, μ_{35}) pair (Figure 3.4). Let p_2, p_5 and p_6 be the probabilities that a packet at node-3 came from node-2, node-5 and node-6 respectively.

Then, $p_2 = \lambda_{23}/(\lambda_{23} + \lambda_{53} + \lambda_{63}) = \lambda_{23}/\lambda_3$.

Similarly, $p_5 = \lambda_{53}/\lambda_3$ and $p_6 = \lambda_{63}/\lambda_3$.

$1/\mu_{ab}$ is the average transmission time of a packet from node-a to node-b. *[Note that we have already introduced t_{ab} as the average transmission time of a packet from node-a to node-b. That t_{ab} and this $1/\mu_{ab}$ are not same. t_{ab} indicates the average time taken by a packet to be transmitted from node-a to node-b when node-a got the access over media. t_{ab} does not depend on which fraction of time node-a gets access over media. But the transmission rate of packets (such as μ_{ab}) from a node depends on which fraction of time the node gets access over media.]* Then the average transmission time of a packet from node-3,

$$
\begin{aligned}
t_3' &= \frac{p_2}{\mu_{31}} + \frac{p_5}{\mu_{36}} + \frac{p_6}{\mu_{35}} \\
&= \frac{\lambda_{63}/\mu_{35} + \lambda_{53}/\mu_{36} + \lambda_{23}/\mu_{31}}{\lambda_{53} + \lambda_{63} + \lambda_{23}}
\end{aligned}
$$

If μ_3 is the combined transmission rate of the packets (destined to node 1, 5 and 6 at next step) at node-3, then

$$
\begin{aligned}
\mu_3 &= \frac{1}{t_3'} \\
&= \frac{\lambda_{53} + \lambda_{63} + \lambda_{23}}{\lambda_{63}/\mu_{35} + \lambda_{53}/\mu_{36} + \lambda_{23}/\mu_{31}}
\end{aligned}
$$

Thus λ_3 and μ_3 are the combined arrival rate and combined transmission rate of the following 3 types of packets at node-3: packet of node-2 to node-1, packet of node-5 to node-6 and packet of node-6 to node-5, all these packets are forwarded by node-3. Let α_3 and β_3 be the corresponding combined discrete parameters of packet arrival and transmission probabilities at node-3 for the above 3 types of packets in interval τ. Then $\alpha_3 = 1 - e^{-\tau\lambda_3}$ and $\beta_3 = 1 - e^{-\tau\mu_3}$. We can get the queue length distribution at node-3 for the above 3 types of packets. Let P be the transition matrix for queue length distribution at node-3. Then

$$P = \begin{bmatrix} \alpha_3' & \alpha_3 & & & & \\ \alpha_3'\beta_3 & \alpha_3\beta_3 + \alpha_3'\beta_3' & \alpha_3\beta_3' & & & \\ & \alpha_3'\beta_3 & \alpha_3\beta_3 + \alpha_3'\beta_3' & \alpha_3\beta_3' & & \\ & & \cdot & \cdot & \cdot & \\ & & & \cdot & \cdot & \cdot \\ & & & & \cdot & \cdot \quad \cdot \\ & & & & \alpha_3'\beta_3 & \alpha_3 + \alpha_3'\beta_3' \end{bmatrix},$$

where $\alpha_3' = 1 - \alpha_3$ and $\beta_3' = 1 - \beta_3$. Let P_{3k} be the probability that at any instant there are total k packets at node-3 from node-5, node-6 and node-2 (they are waiting to be forwarded to node-6, node-5 and node-1 respectively).

Now let us assume that node-1 wants to transmit a packet to node-2. Let r be the probability that node-1 selects route 1-3-2 for transmission, then $r' = 1 - r$ is the probability that node-1 selects route 1-4-2.

Transmission of 1 packet from node-1 to node-2 will take $t_{13} + t_{32}$ time if the buffer of node-3 is empty, node-1 gets access over media, node-1 selects path 1-3-2 for this transmission and then node-3 gets access over media. i.e., the probability that required transmission time is $t_{13} + t_{32}$,

$$Pr\{t_{13} + t_{32}\} = m_1 P_{3,0} m_3 r.$$

Similarly,

$$Pr\{t_{14} + t_{42}\} = m_1 P_{4,0} m_4 r'.$$

The required transmission time will be $t_{13} + t_{32} + t_{23}$, if node-1 transmits a packet to node-3, then node-2 transmits a packet to node-3 and finally node-3 forwards the packet of node-1 to node-2. Let g_a be the probability that there is at least 1 packet in the buffer of node-a. We assume that we can calculate the packet length distribution at all the nodes and let $P_{a,k}$ be the probability that at any instant there are k packets in the buffer of node-a. Then $g_a = 1 - P_{a,0}$. Now,

$$Pr\{t_{13} + t_{32} + t_{23}\} = m_1 P_{3,0} m_3 r m_2 r_2 g_2$$

where r_2 is the probability that node-2 selects route 2-3-1 for transmitting a packet to node-1. $r_2^{'} = 1 - r_2$ is the probability that node-2 selects route 2-4-1 for transmitting a packet to node-1.

The required transmission time will be $t_{13} + t_{32} + xt_{23}$, if node-1 transmits a packet to node-3, then node-2 transmits x packets to node-3 and finally node-3 forwards the packet of node-1 to node-3. Thus,

$$Pr\{t_{13} + t_{32} + xt_{23}\} = m_1 P_{3,0} m_3 r (m_2 r_2 g_2)^x.$$

Similarly,

$$Pr\{t_{14} + t_{42} + yt_{24}\} = m_1 P_{4,0} m_4 r^{'} (m_2 r_2^{'} g_2)^y.$$

The required transmission time will be $t_{13} + t_{32} + xt_{23} + yt_{24}$, if node-1 transmits a packet to node-3, then node-2 transmits x packets to node-3 and y packets to node-4 and finally node-3 forwards the packet of node-1 to node-3. Transmission of these $x + y + 1$ packets (1 packet from node-1 to node-3 plus x packets from node-2 to node-3 and y packets from node-2 to node-4) can be arranged in $(x + y + 1)!/((x + 1)!y!)$ ways. Thus,

$$Pr\{t_{13} + t_{32} + xt_{23} + yt_{24}\} = m_1 P_{3,0} m_3 r (m_2 r_2 g_2)^x (m_2 r_2^{'} g_2)^y \frac{(x + y + 1)!}{(x + 1)!y!}.$$

Let t_3 be the average transmission time for these 3 types of packets, then t_3 is the weighted sum of individual transmission times. i.e. $t_3 = (\lambda_{63} t_{35} + \lambda_{53} t_{36} + \lambda_{23} t_{31})/(\lambda_{53} + \lambda_{63} + \lambda_{23})$. When node-1 wants to start transmission, at that moment if there are u packets in the buffer of node-3, then additional time ut_3 will be required to transmit these u packets before the packet of node-1 is forwarded by node-3. Thus the required transmission time will be $t_{13}+t_{32}+ut_3$, if there are u packets in the buffer of node-3, node-1 gets access over media, node-3 gets access over media for u times and finally node-3 gets access over media again to transmit the packet of node-1 to node-2. Transmission of 1 packet from node-1 to node-3 can take place within any transmission of u packets from node-3. Thus, transmission of 1 packet from node-1 to node-3 and transmission of u packets from node-3 can be arranged in $(u+1)$ ways. Thus,

$$Pr\{t_{13}+t_{32}+ut_3\} = m_1 P_{3,u}(m_3)^{u+1} r(u+1).$$

Now consider the situation that, node-1 wants to start transmission of a packet and there are u_1 packets in the buffer of node-3. Let us assume that x_1 packets are transmitted from node-2 to node-3 before the transmission of the packet from node-1 to node-3 and x_2 packets are transmitted from node-2 to node-3 after that transmission (the transmission of the packet from node-1 to node-3). Thus, $u = u_1+x_1$ packets must be transmitted from node-3 before the packet of node-1 is forwarded by node-3 to node-2 and within this total period node-2 transmits $x = x_1+x_2$ packets to node-3. In that case the required time is $t_{13}+t_{32}+ut_3+xt_{23}$. Let the transmission of these u packets by node-3, x packets by node-2 and the transfer of the packet of node-1 can occur in $\phi(u_1, x_1, x_2)$ ways. $\phi(u_1, x_1, x_2)$ has been defined in appendix A.1. if $u_1 = 0$, then $x_1 = u$, $x = x_1+x_2 = u+x_2$, i.e. $x \geq u$. In other words, if $u \leq x$, then u_1 can vary from 0 to u, x_1 can vary from u to 0. But if $u > x$, then x_1 can vary form 0 to x and thus u_1 can vary form $(u-x)$ to u. i.e., u_1 varies from $u - min(u,x)$ to u. Thus,

$$\begin{aligned} Pr\{t_{13}+t_{32}+ut_3+xt_{23}\} &= \sum_{u_1=u-min(u,x)}^{u} m_1 P_{3,u_1}(m_2r_2g_2)^{x_1}(m_2r_2g_2)^{x_2}{m_3}^{(u+1)} r\phi(u_1,x_1,x_2) \\ &= m_1(m_2r_2g_2)^{x}{m_3}^{(u+1)} r \sum_{u_1=u-min(u,x)}^{u} P_{3,u_1}\phi(u_1, u-u_1, x-u+u_1). \end{aligned}$$

Now, assume that node-1 wants to send n packets to node-2. If all the packets go through node-3, then these packets can be arranged in $\psi(n)$ ways. $\psi(n)$ has been defined in appendix A.2. Transmission of n packets from node-1 to node-3 and again these n packets from node-3 to node-2 can occur in $\psi(n)$ ways. Transmission of x packets from node-2 to node-3 and u packets from node-3 can occur in $\sum_{u_1=u-min(u,x)}^{u} P_{3,u_1}\phi(u_1, u-u_1, x-u+u_1)$ ways. Thus, transmission of all the packets can occur in

$$\frac{(n+n-1+u+x)!}{(n+n-1)!(u+x)!}\psi(n) \sum_{u_1=u-min(u,x)}^{u} P_{3,u_1}\phi(u_1, u-u_1, x-u+u_1) \text{ ways.}$$

And the required transmission time will be $nt_{13}+nt_{32}+ut_3+xt_{23}$. Thus,

$$Pr\{nt_{13}+nt_{32}+ut_3+xt_{23}\} =$$

$$m_1^n m_3^{(u+n)} r^{(n)} (m_2 r_2 g_2)^x \tfrac{(n+n-1+u+x)!}{(n+n-1)!(u+x)!}\psi(n) \textstyle\sum_{u_1=u-min(u,x)}^{u} P_{3,u_1}\phi(u_1, u-u_1, x-u+u_1)$$

Similarly,

$$Pr\{nt_{14}+nt_{42}+vt_4+yt_{24}\} =$$

$$m_1^n m_4^{(v+n)} r'^{(n)} (m_2 r_2' g_2')^y \tfrac{(n+n-1+v+y)!}{(n+n-1)!(v+y)!}\psi(n) \textstyle\sum_{v_1=v-min(v,y)}^{v} P_{4,v_1}\phi(v_1, v-v_1, y-v+v_1)$$

Now assume that node-1 wants to transmit n packets to node-2 using both the routes. Let r be the probability that node-1 selects route 1-3-2 for transmission, then $r' = 1-r$ is the probability that node-1 selects route 1-4-2. Thus, on the average nr packets are routed through path 1-3-2 and nr' packets are routed through path 1-4-2.

$$Pr\{nrt_{13}+nrt_{32}+nr't_{14}+nr't_{42}+ut_3+vt_4+xt_{23}+yt_{24}\} =$$
$$\begin{pmatrix} n \\ nr \end{pmatrix} r^{nr}(1-r)^{n(1-r)} Pr_1 + \begin{pmatrix} n \\ n-nr \end{pmatrix} r^{nr}(1-r)^{n(1-r)} Pr_2$$

where

$$Pr_1 =$$

$$m_1^{nr} m_3^{(u+nr)} (m_2 r_2 g_2)^x \tfrac{(2nr-1+u+x)!}{(2nr-1)!(u+x)!}\psi(nr) \textstyle\sum_{u_1=u-min(u,x)}^{u} P_{3,u_1}\phi(u_1, u-u_1, x-u+u_1)$$

and

$$Pr_2 =$$

$$m_1^{nr'} m_3^{(v+nr')} (m_2 r_2' g_2')^y \tfrac{(2nr'-1+v+y)!}{(2nr'-1)!(v+y)!}\psi(nr') \textstyle\sum_{v_1=v-min(v,y)}^{v} P_{4,v_1}\phi(v_1, v-v_1, y-v+v_1).$$

Given t_{13}, t_{32}, t_{14}, t_{42}, t_{23}, t_{24}, t_3, t_4, for a particular value of n and r we can say that

$$\gamma(u,v,x,y) = nrt_{13} + nrt_{32} + nr't_{14} + nr't_{42} + ut_3 + vt_4 + xt_{23} + yt_{24}$$

and $$\theta(u,v,x,y) = \begin{pmatrix} n \\ nr \end{pmatrix} r^{nr}(1-r)^{n(1-r)}Pr_1 + \begin{pmatrix} n \\ n-nr \end{pmatrix} r^{nr}(1-r)^{n(1-r)}Pr_2$$

Then

$$Pr\{t = \gamma(u,v,x,y)\} = \theta(u,v,x,y)$$

Finally, the average transmission time

$$T = \sum_{x=0}^{\infty}\sum_{y=0}^{\infty}\sum_{u=0}^{\infty}\sum_{v=0}^{\infty}\gamma(u,v,x,y)\theta(u,v,x,y) \tag{3.1}$$

3.4 Experimental Results

We solved Equation 3.1 for different values of r with some fixed values of other parameters. We have changed the values of r from 0 to 1 with a step of 0.05 keeping $m_i(i = 1,2,3,4) = 0.25$, $n = 10$, $t_{13} = t_{32} = t_{31} = t_{23} = 2$, $t_{53} = t_{63} = t_{35} = t_{36} = 3$, $\lambda_{53} = 0.042$, $\lambda_{63} = 0.042$, $\lambda_{23} = 0.06$, $\mu_{31} = 0.125$, $\mu_{35} = 0.08$, $\mu_{36} = 0.08$, $t_{14} = t_{42} = t_{41} = t_{24} = 5$, $t_{74} = t_{84} = t_{47} = t_{48} = 5$, $\lambda_{74} = 0.025$, $\lambda_{84} = 0.025$, $\lambda_{24} = 0.024$, $\mu_{41} = 0.05$, $\mu_{47} = 0.05$, $\mu_{48} = 0.05$, $r_2 = 0.5$, $g_2 = 0.95$. For different values of r we got different values of T. We plotted the result in Figure 3.5. The result shows that, transmission time T becomes minimum when $r = 0.75$. That is, if 75% packets are transmitted through path 1-3-2 and the rest 25% packets are transmitted through path 1-4-2, then the total packets can be transmitted in minimum time.

We also made a network model for the graph as shown in Figure 3.2 and ran simulation on that model setting the same values for the parameters and measured the time for transferring 10 packets from node-1 to node-2. Figure 3.5 shows the result as well.

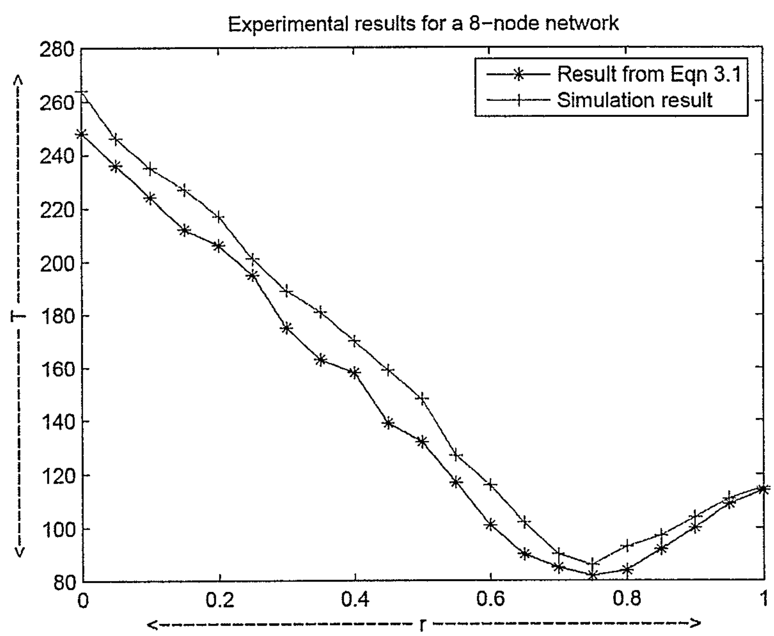

Figure 3.5: Experimental results for the network shown in Figure 3.2

Chapter 4

Load Balancing in Large Network

The complexity of the calculation of Equation 3.1 is very high, hence there is a stability issue. The approach described in Section 3.3 may not be available in larger networks. In this chapter we present another approach more suitable to large networks.

4.1 Approach for Larger Networks

We consider a network shown in Figure 4.1. There are two distinct routes from node-1 to

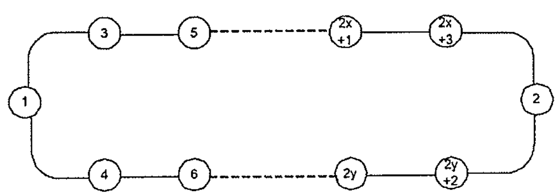

Figure 4.1: A larger network

node-2: route $1-3-5---(2x+1)-(2x+3)-2$ and route $1-4-6---(2y)-(2y+2)-2$. Node-1 sends packets to node-2 and node-2 sends packets to node-1. The other nodes (node-3, 4, 5, ...) forward packets of node-1 and node-2. They also forward packets of other nodes present in the network (not shown in the figure). When node-1 transmits, node-3 and node-4 cannot transmit at the same time due to interference. But when node-3 transmits a packet to node-5, node-4 can also transmit a packet to node-6 and so on. That is, there is no interference among the nodes of $3-5---(2x+1)-(2x+3)$ path and the nodes of $4-6---(2y)-(2y+2)$ path. Now suppose, node-1 wants to send some packets to node-2. The total time required for a packet to be transmitted from node-1 to node-2 through path $1-3-5---(2x+1)-(2x+3)-2$ depends on many parameters like how busy the nodes on the path (i.e. node $3,5,...$) are, the channel conditions between the nodes on the path, etc. Also in this model, like the previous case, we assume that transmission of every packet is followed by sending back an acknowledgement. When node-1 transmits a packet to node-3, we can see that node-3, node-5 cannot transmit due to interference. i.e., when any node transmits, then the nodes within next two hops cannot transmit. In other words, on a path only one node among three consecutive nodes can transmit at any moment. Let T'_{11} be the average time for a packet to be transmitted from node-1 to node-2 through path $1-3-5---(2x+1)-(2x+3)-2$. If we assume that a packet takes equal time to pass each hop, then we can say that $T'_{11}/h_{1-3-5---(2x+1)-(2x+3)-2}$ amount of time is taken by a packet in each hop, where $h_{1-3-5---(2x+1)-(2x+3)-2}$ is the number of hops in the path. Since only one node out of three consecutive nodes on a path can transmit at a time, if node-1 transmits a series of packets through route $1-3-5---(2x+1)-(2x+3)-2$, on average, the inter-arrival time of the packets at node-2 will be $3T'_{11}/h_{1-3-5---(2x+1)-(2x+3)-2}$. If T_1 indicates the average time for transmission of n packets from node-1 to node-2 through this path, then

$$T_1 = T'_{11} + \frac{(n-1)3T'_{11}}{h_{1-3-5---(2x+1)-(2x+3)-2}}.$$

Similarly, let T'_{22} be the average time for a packet to be transmitted from node-1 to node-2 through path $1-4-6---(2y)-(2y+2)-2$. If node-1 sends these n packets using both the routes and let r be the probability that node-1 selects route $1-3-5---(2x+1)-(2x+3)-2$ for transmission. Then $1-r$ is the probability that node-1 selects route $1-4-6---(2y)-(2y+2)-2$. Then the expected time for transmission of these n packets,

$$T = max(T_1, T_2) \tag{4.1}$$

where

$$\begin{aligned} T_1 &= T'_{11} + \frac{(nr-1)3T'_{11}}{h_{1-3-5---(2x+1)-(2x+3)-2}} \\ T_2 &= T'_{22} + \frac{(n(1-r)-1)3T'_{22}}{h_{1-4-6---(2y)-(2y+2)-2}} \end{aligned}$$

4.2 Experimental Results

We consider a network of 13 nodes as shown in Figure 4.2. Here $h_{1-3-5-7-9-11-2} = 6$ and $h_{1-4-6-8-10-12-14-2} = 7$. At first, node-1 does not transmit any packet. Other nodes

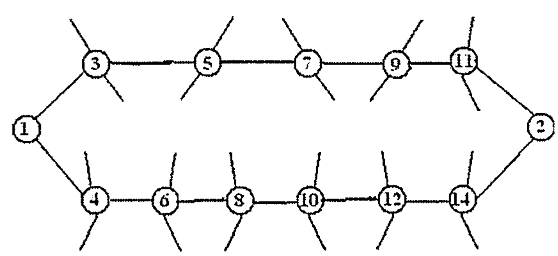

Figure 4.2: A 13-node network

in the network are transmitting / forwarding packets. Let λ_i be the packet arrival rate to node-i and μ_i be the service rate at node-i. We set $\lambda_3 = \lambda_5 = \lambda_7 = \lambda_9 = \lambda_{11} = 0.3$, $\lambda_4 = \lambda_6 = \lambda_8 = \lambda_{10} = \lambda_{12} = \lambda_{14} = 0.4$, $\mu_3 = \mu_5 = \mu_7 = \mu_9 = \mu_{11} = 0.6$, $\mu_4 = \mu_6 = \mu_8 = \mu_{10} = \mu_{12} = \mu_{14} = 0.5$. Then we made node-1 transmitting some packets through both the routes (route 1-3-5-7-9-11-2 and route 1-4-6-8-10-12-14-2) and measured the average time for transmitting 1 packet from node-1 to node-2 independently through both the routes. We got $T'_{11} = 20.4$ and $T'_{22} = 34.6$. We plugged the values of T'_{11}, T'_{22}, $h_{1-3-5-7-9-11-2}$ and $h_{1-4-6-8-10-12-14-2}$ in Equation 4.1 and set $n = 10$.

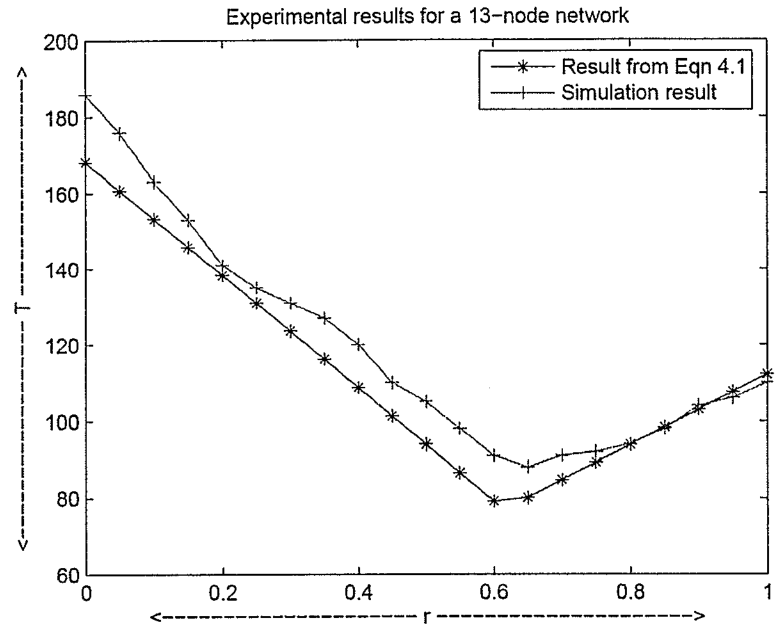

Figure 4.3: Experimental results for graph shown in Figure 4.2

We solved Equation 4.1 for different values of r. We varied the values of r from 0 to 1 with a step of 0.05. For different values of r we got different values of T. We plotted the result in Figure 4.3. The result shows that, transmission time T becomes minimum when $r = 0.6$. That is, if 60% packets are transmitted through path 1-3-5-7-9-11-2 and the rest 40% packets are transmitted through path 1-4-6-8-10-12-14-2, then the total packets can be transmitted in minimum time.

On the network model we also ran simulation for different values of r and obtained the times required for transmitting 10 packets from node-1 to node-2. The results are shown in Figure 4.3.

4.3 Modifying this Model for the IEEE 802.16

In Section 4.1 we have assumed that all the nodes use the same channel in the media. Thus interference plays a significant role in the modeling. In IEEE 802.16 mesh the nodes use OFDMA to share the media [2]. Thus when one node transmits a packet, the neighbors of this node can also transmit / receive packet to / from any other node. Even multiple radios can be used in a single node and then that node can keep transmitting and receiving packets simultaneously [9]. As an example, in Figure 4.2, if 2 radios are installed in node-3, then node-3 can receive packet from node-1 while transmitting a packet to node-5. Sometimes multiple sub-carriers can be assigned to any link between 2 nodes. Then those nodes can transmit 2 packets parallelly between them. As an example, if 2 sub-carriers are assigned to the link between node-3 and node-5, then node-3 can transmit 2 packets simultaneously to node-5.

Now, in our model we are not considering all these scenarios. We are assuming that nodes use OFDMA to share the media and thus no node has to wait due to interference. But we are assuming that all nodes are equipped with single radio and there is no multiple sub-carrier in any link, i.e., only 1 OFDMA channel is assigned to a link. In that case, any node can participate in only one transmission at any moment. But when any node is transmitting or receiving any packet, its neighbors can also transmit or receive at that moment. i.e., when node-1 is transmitting a packet to node-3, node-5 can also transmit a packet to node-7. In this case, only 1 node among 2 consecutive nodes on a path can transmit at any moment. Then Equation 4.1 becomes

$$T = max(T_1, T_2) \tag{4.2}$$

where

$$\begin{aligned} T_1 &= T_{11}^{'} + \frac{(nr-1)2T_{11}^{'}}{h_{1-3-5----7-9-11-2}} \\ T_2 &= T_{22}^{'} + \frac{(n(1-r)-1)2T_{22}^{'}}{h_{1-4-6----8-10-12-14-2}} \end{aligned}$$

In our simulation model we have kept all arrival rates (λ_i) and service rates (μ_i) same as stated in Section 4.2. We changed the model to support OFDMA (now only 1 node among 2 consecutive nodes on a path can transmit at any moment, where as in previous case only 1 node among 3 consecutive nodes on a path could transmit at any moment). We got $T'_{11} = 16.2$ and $T'_{22} = 24.4$.

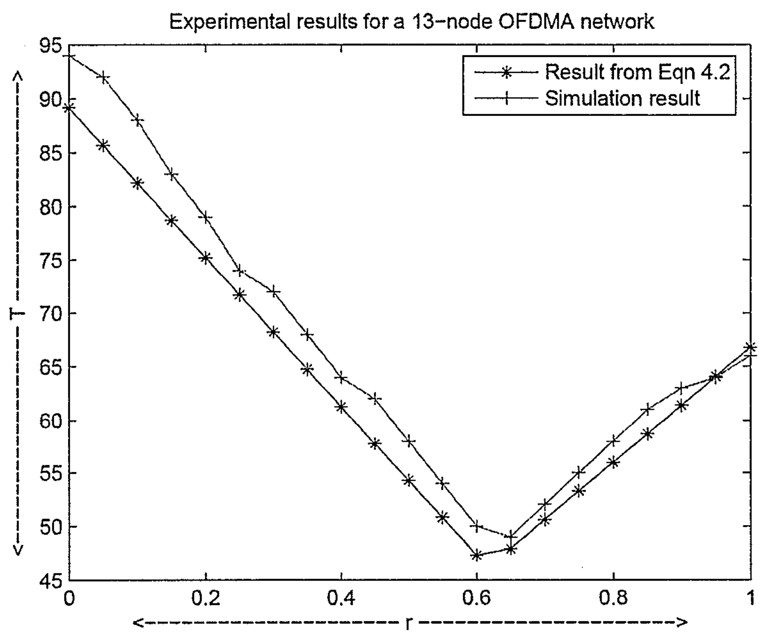

Figure 4.4: Experimental results for graph shown in Figure 4.2

We solved Equation 4.2 with these parameters for different values of r. We varied the values of r from 0 to 1 with a step of 0.05. For different values of r we got different values of T. Figure 4.4 shows the result. The result shows that, transmission time T becomes minimum when $r = 0.6$. That is, if 60% packets are transmitted through path 1-3-5-7-9-11-2 and the rest 40% packets are transmitted through path 1-4-6-8-10-12-14-2, then the total packets can be transmitted in minimum time.

On the network model we also ran simulation for different values of r and obtained the times required for transmitting 10 packets from node-1 to node-2. The results are shown in Figure 4.4.

4.4 Adaptive Load Balancing for a more Real Scenario

So far we have assumed that we know the time required for a packet to be transmitted from a source to a destination. We have built our model that calculates which fraction of traffic should be routed in which path when 2 paths are available. The calculation requires the total time associated to each path. But in real life these time-values vary over time. Because in wireless media channel conditions change over time, load of any intermediate node on a path can change while transmitting its own packets or forwarding others packets. Thus the total packet transmission time associated to a path can change. In that case if we can measure the transmission time on a path from time to time and thus adjust the ratio parameter (r), we can improve the system.

We changed our simulation model (for the network shown in Figure 4.2, that supports OFDMA) such that the total time associated to the paths change over time. For this we have changed λ_i within the range 0.25 to 0.4 and μ_i within the range 0.45 to 0.6 randomly over time and measured the total time for transmitting 50, 100 and 150 packets for the following 3 cases:

1. Always transmit packets through the shortest path
2. Calculate r at the beginning and follow the same r for the transmission
3. Adjust r after the transmission of each 10 packets by newly measured transmission time (T'_{11} and T'_{22}).

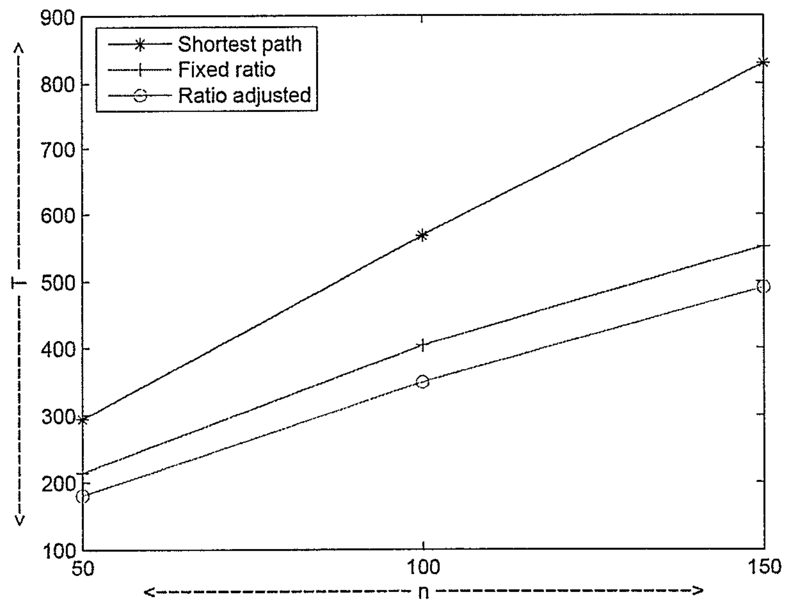

Figure 4.5: Experimental results for varying channel condition

The result is shown in Figure 4.5. From the figure we see that if we recalculate the fraction parameter r and distribute packets according to the newly calculated value, then we can reduce the time required for transmission of a series of packets from a source to a destination. Now the major challenge is to measure the transmission time associated to each route, which changes over time in wireless media.

4.4.1 Estimating Transmission Time

In the method stated above, when node-1 wants to send some packets to node-2 (Figure 4.2), in order to calculate the fraction parameter r node-1 needs to know the values of T'_{11} and T'_{22}. Let us assume that node-1 transmits a packet to node-3 at time t_1 and node-2 receives the packet (from node-11) at time t_2. Then $T'_{11} = t_2 - t_1$. Similarly let node-1 transmits a packet to node-4 at time t_3 and node-2 receives the packet (from node-14) at time t_4. Then $T'_{22} = t_4 - t_3$. The problem is that, node-1 can only know the values of t_1 and t_3. It does not know t_2 or t_4. Thus it is a problem for node-1 to estimate T'_{11} and T'_{22} in regular interval. Node-2 can know the values of t_2 and t_4. If a time-stamp is set to each packet by the sender then node-2 can also know the values of t_1 and t_3. Thus node-2 can calculate T'_{11} and T'_{22}, which are actually needed by node-1. Similarly, if node-2 sends packets to node-1 through different routes, then node-1 can easily calculate the transmission times of a packet from node-2 to node-1 over different routes. Let T''_{11} be the transmission time of a packet from node-2 to node-1 through the path 2-11-9-7-5-3-1 and T''_{22} be the transmission time of a packet from node-2 to node-1 through the path 2-14-12-10-8-6-4-1.

Thus, node-1 knows T''_{11} and T''_{22} which are needed by node-2. Similarly node-2 knows T'_{11} and T'_{22} which are needed by node-1. If we assume that the transmission times on both direction in a route are same, i.e., $T'_{11} = T''_{11}$ and $T'_{22} = T''_{22}$, then this problem can be solved.

4.4.2 Implementing Load Balancing in Multiple-path Routing

In traditional routing each node maintains a routing table, which contains entries for each destination nodes. Each entry has two fields, "Destination" and "Next hop". Table 4.1 shows a single-path routing table for node-1 of Figure 4.2.

Destination	Next hop
1	-
2	3
3	3
4	4
5	3
6	4
7	3

Destination	Next hop
8	4
9	3
10	4
11	3
12	4
14	4

Table 4.1: Single-path routing table at node-1 of Figure 4.2

In single-path routing a router maintains only one path to each destination. Thus only one value is kept in the field "Next hop". In multiple-path routing a router maintains multiple paths to a destination [17] and thus multiple values are kept under the "Next hop" field. So, the entry for destination 2 of Table 4.1 can be modified as Table 4.2 for multiple-path routing.

Destination	Next hop
.	.
2	3, 4
.	.
.	.

Table 4.2: Multiple-path routing table at node-1 of Figure 4.2

Now, this load balancing technique needs to keep track of transmission times over each path for a destination. The routing table can be modified to keep record for the transmission times. And this values shuold be updated from time to time. As an example, Table 4.2 shows that there are two routes for the destination node-2 from node-1. Along with the "Next hop" values we also need to store the times associated with this routes. Thus we also need to store T_{1-3-2} and T_{1-4-2} in the table (which will be used as $T_{11}^{'}$ and $T_{22}^{'}$ respectively for the calculation of fraction parameter r as stated in Section 4.3). The routing table shown in Table 4.2 can be modified as shown in Table 4.3.

Destination	Next hop
.	.
2	$3\{T_{1-3-2}\}$, $4\{T_{1-4-2}\}$
.	.
.	.

Table 4.3: Multiple-path routing table at node-1 of Figure 4.2

When a packet from node-2 is received by node-1 from node-3, then node-1 can easily calculate $T_{11}^{''}$ and use this value to modify T_{1-3-2}. And when a packet from node-2 is received by node-1 from node-4, then node-1 can easily calculate $T_{22}^{''}$ and use this value to modify T_{1-4-2}. Actually T_{1-3-2} can be the average of last n number of T''_{11} values, where n is to be chosen carefully depending on how rapidly the channel condition, the traffic load, etc. change.

Chapter 5

Conclusion and Future Works

5.1 Conclusion

In distributed wireless mesh networks all nodes make their own scheduling. There is no centralized controller for guiding the nodes. The nodes also decide their own route for sending packets to a destination. In this thesis our goal was to develop a method for selecting a path among multiple paths during sending a packet to a destination. We have seen that if two paths are available for a particular destination and if packets can be distributed properly among the two paths, then all the packets can be transmitted within minimum time.

We have developed a mathematical model for selecting a path among multiple routes. We have also carried out network simulation that supports our method.

5.2 Future Works

We can extend our work in the following directions:

Extension to Multiple Paths

In this thesis we have distributed the packets between two routes to a destination. This work can be extended to distribute traffic when more than two paths are available to a destination. If packets can be distributed intelligently among multiple paths, total transmission time can be minimized.

Consideration of Contention

In this thesis we have not considered contention of packets when multiple nodes are trying to send their packets at the same time. We have developed the system for IEEE-802.16 coordinated distributed mesh network, which is actually contention free. Contention takes place in uncoordinated systems such as IEEE-802.11 distributed mesh network, IEEE-802.16 uncoordinated distributed mesh network, etc. By considering contention this load balancing method can be extended to uncoordinated systems as well.

Load Balancing in Whole Network

In this model each node calculates its own fraction parameter r. This parameter depends on how busy the other nodes on a route are. When a node distributes its packets over the available routes for transmission, then the nodes on the route have to forward those packets. This affects the fraction parameter r of the neighboring nodes. Thus when a node adjusts its fraction parameter r, then its neighbors may have to adjust their fraction parameter again. Thus the nodes have to adjust their own parameter again and again until the network condition becomes stable. Further calculation and simulation can be done to analyze the situation.

Time-stamp by All Nodes on The Path

In Section 4.4.2 we have proposed that sender may set a time-stamp on each packet when it sends the packet. Reading that time-stamp the receiver can calculate the required transmission time on that path. Setting this time-stamp will make the packet larger, which ultimately increases load on the network. The resultant complexity is needed to be measured.

This can be done in many ways. As an example, in Figure 4.2, if node-2 sets a time-stamp on a packet sending to node-1, then node-1 can calculate the required transmission time for this path from node-1 to node-2 and update its routing table (Table 4.3) for the entry "Destination: 2, Next hop: 3". All the nodes on the path (those forward the packet towards node-1) can also read this time-stamp to update their routing tables for the same destination. Thus node-3, node-5, ..., node-11 can update their routing tables for the entry "Destination: 2". Again all the nodes on the path can also add their own time-stamps on the packet they are forwarding. In that case with the single packet node-1 will be able to update the required transmission time to all the nodes either they are forwarding / sending. That is, just a single packet transmission from node-2 to node-1 through the path 2-11-9-7-5-3-1 can update the routing table of node-1 for destinations 2, 11, 9, 7, 5, 3, the routing table of node-3 for destinations 2, 11, 9, 7, 5, routing table of node-5 for destinations 2, 11, 9, 7 and so on. Certainly it will increase the workloads on the routers and the amount of traffic over the network. Further calculations and simulations can be done to analyze the scenario.

Development of New Protocols

To implement the load balancing technique further extended protocols might be needed to be developed.

Appendix A

A.1 Explanation of ϕ

Let us consider the graph shown in figure A.1.

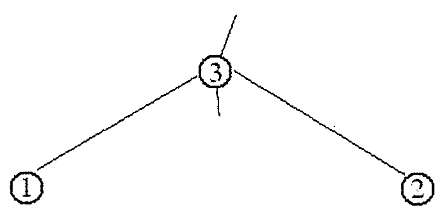

Figure A.1: A 3-node network

To make this scenario clear we look at the possible sequence of transmissions:

- Node-1 wants to transmit a packet.
- There are u_1 packets in the buffer of node-3.
- Either node-2 or node-3 transmits their packets. If node-2 transmits its packets to node-3, it will increase the number of packets in the buffer of node-3. If node-3 transmits its packets, it will decrease the number of the packets in the buffer of node-3.
- In this way x_1 packets are transferred from node-2 to node-3.

- After that, node-1 transmits its packet to node-3.
- When node-1 transmits to node-3, let there are k' packets in node-3. $0 \leq k' \leq u, u = u_1 + x_1$. $k' = u$ if node-3 does not transmit any packet during the transmission of x_1 packets from node-2 to node-3. Similarly, $k' = 0$ if node-3 transmits all its u_1 packets and the x_1 packets it got from node-2.
- Now the packet of node-1 is in the buffer of node-3. These $k'(0 \leq k' \leq u)$ packets (which are ahead of the packet of node-1 in the buffer of node-3) needs to be transferred by node-3 before the packet of node-1 is forwarded by node-3.
- Within the transmission of k' packets by node-3, another x_2 packets are transmitted to node-3 by node-2 (clearly in the buffer of node-3, these k' packets will stay ahead of the packet of node-1 and x_2 packets will stay behind the packet of node-1).

In this section we going to explain $\phi(u_1, x_1, x_2)$, where $\phi(u_1, x_1, x_2)$ indicates in how many ways the above transmissions can take place.
At the moment when node-1 transfers its packet to node-3, there are k' packets in node-3 means node-2 has transmitted x_1 packets to node-3 and node-3 has transmitted $k = u - k'$ packets. k (as well as k') can have any value from $0, 1, 2, ..., u_1, u_1 + 1, u_1 + 2, ..., u_1 + x_1$.
Let $\phi_1(k)$ indicates the number of all possible combinations in which node-3 can transmit k packets. $k = 0$ means node-3 has transmitted no packet and node-2 has transmitted x_1 packets. This can happen only in 1 way. Thus
$\phi_1(0) = 1$.
$k = 1$ means node-3 has transmitted one packet and node-2 has transmitted x_1 packets. This can happen only in $(x_1 + 1)!/(x_1!1!)$ way. Thus
$\phi_1(1) = (x_1 + 1)!/(x_1!1!)$.
Similarly we get,
$\phi_1(k) = (x_1 + k)!/(x_1!k!)$ for $0 \leq k \leq u_1$.

Calculation of $\phi_1(k)$ for $u_1 < k \leq u_1 + x_1$ is a little bit complex. We try to explain this with two examples.

A.1.1 Example-1

Let us consider $u_1 = 2$, $x_1 = 2$ and $k = 3$. This means that node-2 has transmitted 2 packets to node-3 and node-3 has transmitted 3 packets (previously it had $u_1 = 2$ packets and it got $x_1 = 2$ new packets from node-2; it has transmitted its previous 2 packets and the first 1 packet it got from node-2). Here, we introduce some symbols like b_i, c_i and d_i, where b_i indicates the transmission of ith packet from those u_1 packets from node-3, c_i indicates the transmission of ith packet from those x_1 packets from node-2 and d_i indicates the transmission of ith packet from those x_1 packets from node-3, which was transmitted from node-2 to node-3. Thus, in this example there can be two b_i, two c_i and one d_i. Two c_i implies $x_1 = 2$, and two b_i one d_i implies $k = 3$. If we consider only

<table>
<tr><th>Combinations of $b_1b_2c_1c_2$</th><th>Combinations of $b_1b_2c_1c_2d_1$</th></tr>
<tr><td rowspan="2">$b_1b_2c_1c_2$</td><td>$b_1b_2c_1d_1c_2$</td></tr>
<tr><td>$b_1b_2c_1c_2d_1$</td></tr>
<tr><td rowspan="2">$b_1c_1b_2c_2$</td><td>$b_1c_2b_2d_1c_2$</td></tr>
<tr><td>$b_1c_2b_2c_2d_1$</td></tr>
<tr><td>$b_1c_1c_2b_2$</td><td>$b_1c_1c_2b_2d_1$</td></tr>
<tr><td>$c_1c_2b_1b_2$</td><td>$c_1c_2b_1b_2d_1$</td></tr>
<tr><td>$c_1b_1c_2b_2$</td><td>$c_1b_1c_2b_2d_1$</td></tr>
<tr><td rowspan="2">$c_1b_1b_2c_2$</td><td>$c_1b_1b_2d_1c_2$</td></tr>
<tr><td>$c_1b_1b_2c_2d_1$</td></tr>
</table>

Table A.1: Possible transfer combinations for example-1

two b_is and two c_i, we get total $(2+2)!/(2!2!) = 6$ combinations. These combinations are shown in the first column of table A.1. Now we want to add d_1 in the each of the above 6 combinations. It is clear that d_1 cannot be placed before any b_i. Again this d_1 must be placed anywhere after c_1 in any combination. Thus we get 9 new combinations from the above 6 combinations. The combinations are shown in the second column of table A.1. We can see that insertion of d_i and thus generation of new combinations does not depend on how many b_is are there in the combination, rather it depends on how many consequent c_is are there at the end of each combination.

A.1.2 Example-2

Let us consider $u_1 = 1$ and $x_1 = 3$. Thus in these combinations there will be one b_i and three c_is. For one b_i and three c_is we get total $(1+3)!/(1!3!) = 4$ combinations. The combinations are shown in the first column of table A.2. If we set $k = 2$ then one d_i (i.e. d_1) will be inserted and from the 4 combinations we get 9 combinations. These combinations are shown in the second column of table A.2. If we set $k = 3$ then there will be two d_is (i.e. d_1 and d_2) and from the 9 combinations we get 14 combinations. These combinations are shown in the third column of table A.2. if we set $k = 4$ then there will be three d_is (i.e. d_1, d_2 and d_3) and from the 14 combinations we get 14 combinations. These combinations are shown in the fourth column of table A.2. With this example we are trying to explain $\phi_1(k)$ for $u_1 < k \leq u_1 + x_1$. In this example $u_1 = 1$. If we set $k = 2$, that means $k - u_1 = 1$ d_i (i.e. d_1) will be inserted and total 9 combinations (the 9 combinations of second column) can be generated. i.e. $\phi_1(k = 2) = 9$, (for $u_1 = 1, x_1 = 3$). If we set $k = 3$, that means $k - u_1 = 2$ d_is (i.e. d_1 and d_2) will be inserted and total 14 combinations (the 14 combinations of third column) can be generated. i.e. $\phi_1(k = 3) = 14$, (for $u_1 = 1, x_1 = 3$). Similarly $\phi_1(k = 4) = 14$ (there are 14 combinations in fourth column).

If we consider the combination $b_1c_1c_2c_3$ (shown in first row first column) and want to

Combinations of $b_1c_1c_2c_3$	Combinations of $b_1c_1c_2c_3d_1$	Combinations of $b_1c_1c_2c_3d_1d_2$	Combinations of $b_1c_1c_2c_3d_1d_2d_3$
$b_1c_1c_2c_3$	$b_1c_1d_1c_2c_3$	$b_1c_1d_1c_2d_2c_3$	$b_1c_1d_1c_2d_2c_3d_3$
		$b_1c_1d_1c_2c_3d_2$	$b_1c_1d_1c_2c_3d_2d_3$
	$b_1c_1c_2d_1c_3$	$b_1c_1c_2d_1d_2c_3$	$b_1c_1c_2d_1d_2c_3d_3$
		$b_1c_1c_2d_1c_3d_2$	$b_1c_1c_2d_1c_3d_2d_3$
	$b_1c_1c_2c_3d_1$	$b_1c_1c_2c_3d_1d_2$	$b_1c_1c_2c_3d_1d_2d_3$
$c_1b_1c_2c_3$	$c_1b_1d_1c_2c_3$	$c_1b_1d_1c_2d_2c_3$	$c_1b_1d_1c_2d_2c_3d_3$
		$c_1b_1d_1c_2c_3d_2$	$c_1b_1d_1c_2c_3d_2d_3$
	$c_1b_1c_2d_1c_3$	$c_1b_1c_2d_1d_2c_3$	$c_1b_1c_2d_1d_2c_3d_3$
		$c_1b_1c_2d_1c_3d_2$	$c_1b_1c_2d_1c_3d_2d_3$
	$c_1b_1c_2c_3d_1$	$c_1b_1c_2c_3d_1d_2$	$c_1b_1c_2c_3d_1d_2d_3$
$c_1c_2b_1c_3$	$c_1c_2b_1d_1c_3$	$c_1c_2b_1d_1d_2c_3$	$c_1c_2b_1d_1d_2c_3d_3$
		$c_1c_2b_1d_1c_3d_2$	$c_1c_2b_1d_1c_3d_2d_3$
	$c_1c_2b_1c_3d_1$	$c_1c_2b_1c_3d_1d_2$	$c_1c_2b_1c_3d_1d_2d_3$
$c_1c_2c_3b_1$	$c_1c_2c_3b_1d_1$	$c_1c_2c_3b_1d_1d_2$	$c_1c_2c_3b_1d_1d_2d_3$

Table A.2: Possible transfer combinations for example-2

insert d_1 in it, we see that d_1 must be inserted after b_1 and c_1. i.e., d_i must be inserted after b_j (for all j) and c_l (for $l \leq i$). d_i can be inserted anywhere within c_l ($l > i$) to make new combinations. Thus when we want to insert a d_i in any combination, we need to know that how many consecutive c_ls (for $l > i$) are there at the end of this combination. If there are 'n' number of such c_ls ($l > i$) at the end of a combination, insertion of d_i will make $n + 1$ new combinations, each of them ends with m consecutive c_ls ($0 \leq m \leq n$). On the next step d_{i+1} will be inserted in each of these newly generated combinations and again other combinations will be generated following the same rule. i is increased by 1 on each step (first we insert d_1, then d_2 and so on) and thus number of consecutive c_ls at end of new combinations will decrease on each step. As in the example if we consider the combination $b_1c_1c_2c_3$, we see that this ends with three consecutive c_ls ($c_1c_2c_3$). When we want to insert d_1, n becomes 2 (n = number of consecutive c_l at the end for $l > 1$). When we insert d_1, three new combinations are generated, one ends with 2 c_ls at the end ($b_1c_1d_1c_2c_3$), one ends with 1 c_l at the end ($b_1c_1c_2d_1c_3$) and one ends with no c_l at the end ($b_1c_1c_2c_3d_1$). The cell indicated by first row and second column of table A.2 is divided into three sub-cells and these three sub-cells show those three combinations.

		number of c_ls where $l > i$				number of combinations
		0	1	2	3	
i	0	$a_{0,0} =$ $\frac{(u_1-1+x_1)!}{(u_1-1)!x_1!}$ $= 1$	$a_{0,1} =$ $\frac{(u_1-1+x_1-1)!}{(u_1-1)!(x_1-1)!}$ $= 1$	$a_{0,2} =$ $\frac{(u_1-1+x_1-2)!}{(u_1-1)!(x_1-2)!}$ $= 1$	$a_{0,3} =$ $\frac{(u_1-1+x_1-3)!}{(u_1-1)!(x_1-3)!}$ $= 1$	
	1	$a_{1,0} =$ $a_{0,0} + a_{0,1} + a_{0,2} + a_{0,3}$ $= 4$	$a_{1,1} =$ $a_{0,1} + a_{0,2} + a_{0,3}$ $= 3$	$a_{1,2} =$ $a_{0,2} + a_{0,3}$ $= 2$		$\phi_1(k = u_1 + i)$ $= \sum a_{1,j}$ $\Rightarrow \phi_1(2) = 9$
	2	$a_{2,0} =$ $a_{1,0} + a_{1,1} + a_{1,2}$ $= 9$	$a_{2,1} =$ $a_{1,1} + a_{1,2}$ $= 5$			$\phi_1(k = u_1 + i)$ $= \sum a_{2,j}$ $\Rightarrow \phi_1(3) = 14$
	3	$a_{3,0} =$ $a_{2,0} + a_{21}$ $= 14$				$\phi_1(k = u_1 + i)$ $= \sum a_{3,j}$ $\Rightarrow \phi_1(4) = 14$

Table A.3: Calculation of $\phi_1(k)$ for example-2

The computation of $\phi_1(k)$ is defined in table A.3. In the first row $a_{0,j}$s are calculated. $a_{0,j}$ indicates the number of combinations with 0 d_i (i.e. no d_i), which ends with exactly

j consecutive c_ls at the end. A combination ends with exactly j consecutive c_ls means a b_i is immediately followed by those consecutive c_ls. If there is no d_i, i.e. if we consider only the combinations with u_1 b_is and x_1 c_ls, then there will be $(u_1 - 1 + x_1 - j)!/((u_1 - 1)!(x_1 - j)!)$ combinations which ends with exactly j consecutive c_ls. Now we will insert d_1 in these combinations. The combination that ends with three c_ls, after inserting d_1 will generate one combination ending with no c_l, one combination ending with one c_l and one combination ending with two c_ls. And the combination with no c_l will generate only one combination with no c_l after inserting d_1. The same rule will be applied for inserting d_2 and so on. Thus, we get $a_{i,j} = \sum_{p=i}^{x1-i+1} a_{i-1,p}$, for $i > 0$.

There will be a little modification in the above calculation shown in table A.3 if we consider the situation where $u_1 = 0$, i.e., if no b_j is there. In that case all the c_ls can make only one combination. i.e., we get $a_{0,0} = a_{0,1} = a_{0,2} = 0$ and $a_{0,3} = 1$. This modified calculation is shown in table A.4.

Thus, we get,

$$\begin{aligned}
\phi_1(k) &= \frac{(x_1+k)!}{x_1!k!}, \text{ for } 0 \leq k \leq u_1 \\
&= \sum_{j=0}^{u_1+x_1-k} a_{k-u_1,j}, \text{ for } u_1 < k \leq u_1 + x_1 \\
&\text{where } a_{i,j} = \sum_{p=j}^{x_1-i+1} a_{i-1,p}, \text{ for } i > 0 \\
&\text{and } a_{0,j} = \frac{(u_1-1+x_1-j)!}{(u_1-1)!(x_1-j)!}, \text{ for } u_1 > 0 \\
&\text{and } a_{0,j} = \delta_{jx_1}, \text{ for } u_1 = 0, \text{ where } \delta_{ij} \text{ is Kronecker delta function.}
\end{aligned}$$

Now we get $\phi_1(k)$, which indicates the number in how many ways x_1 packets can be transmitted from node-2 to node-3 before the packet of node-1 is transferred to node-3. In the mean time node-3 has transmitted k packets and $k' = u_1 + x_1 - k$ packets are there in the buffer of node-3 just ahead of the packet of node-1 in node-3. Transmission

		number of c_ls where $l > i$				number of
		0	1	2	3	combinations
i	0	if $(u_1 = 0)$ then $a_{0,0} = 0$ else $a_{0,0} = \frac{(u_1-1+x_1)!}{(u_1-1)!x_1!} = 1$	if $(u_1 = 0)$ then $a_{0,1} = 0$ else $a_{0,1} = \frac{(u_1-1+x_1-1)!}{(u_1-1)!(x_1-1)!} = 1$	if $(u_1 = 0)$ then $a_{0,2} = 0$ else $a_{0,2} = \frac{(u_1-1+x_1-2)!}{(u_1-1)!(x_1-2)!} = 1$	if $(u_1 = 0)$ then $a_{0,3} = 1$ else $a_{0,3} = \frac{(u_1-1+x_1-3)!}{(u_1-1)!(x_1-3)!} = 1$	
	1	$a_{1,0} = a_{0,0} + a_{0,1} + a_{0,2} + a_{0,3} = 4$	$a_{1,1} = a_{0,1} + a_{0,2} + a_{0,3} = 3$	$a_{1,2} = a_{0,2} + a_{0,3} = 2$		$\phi_1(k = u_1 + i) = \sum a_{1,j} \Rightarrow \phi_1(2) = 9$
	2	$a_{2,0} = a_{1,0} + a_{1,1} + a_{1,2} = 9$	$a_{2,1} = a_{1,1} + a_{1,2} = 5$			$\phi_1(k = u_1 + i) = \sum a_{2,j} \Rightarrow \phi_1(3) = 14$
	3	$a_{3,0} = a_{2,0} + a_{21} = 14$				$\phi_1(k = u_1 + i) = \sum a_{3,j} \Rightarrow \phi_1(4) = 14$

Table A.4: Modified calculation of $\phi_1(k)$ for example-2

of these k' packets from node-3 and transmission of x_2 packets from node-2 to node-3 can occur in $(k' + x_2)!/(k'!x_2!)$ ways. Thus we get,

$$\begin{aligned}\phi(u_1, x_1, x_2) &= \sum_{k=0}^{u_1+x_1} \phi_1(k)\frac{(k' + x_2)!}{k'!x_2!} \\ &= \sum_{k=0}^{u_1+x_1} \phi_1(k)\frac{(u_1 + x_1 - k + x_2)!}{(u - k)!x_2!}\end{aligned}$$

A.2 Explanation of ψ

Let us consider the graph shown in figure A.2.

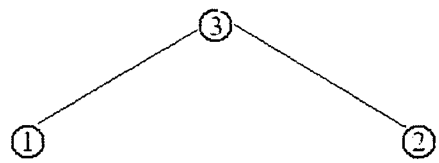

Figure A.2: A 3-node network

Node-1 wants to transmit k packets to node-2. To do this, node-1 first sends its packet to node-3 and then node-3 forwards the packet to node-2. Node-2 does not transmit any packet. Node-3 does not transmit any packet of its own. It only forwards the packet of node-1 to node-2. Only one node can transmit a packet at a time. In this section we are going to explain $\psi(k)$, where $\psi(k)$ indicates in how many ways node-1 can transmit k packets to node-2.

Here we introduce some symbols like a_i and b_i, where a_i indicates the transmission of ith packet from node-1 to node-3 and b_i indicates the transmission of ith packet from node-3 to node-2. For any possible transmission combination, it is clear that a_j must be placed after a_i and similarly b_j must be placed after b_i , for $j > i; i, j \leq k$. Again b_i must be placed after a_i.

A.2.1 Example-3

In this example, if we set $k = 1$, then we can see that the transmissions can happen only in one way. The only possible transmission combination is shown in the first column of table A.5. Thus $\psi(1) = 1$. If we set $k = 2$, then the transmissions can occur in the following two ways. The possible transmission combinations are shown in the second column of table A.5. Thus $\psi(2) = 2$. Similarly the possible five combinations for $k = 3$ are shown in the third column of table A.5 and the possible fourteen combinations for $k = 4$ are shown in the fourth column of table A.5. Thus we get $\psi(3) = 5$ and $\psi(4) = 14$.

$k=1$	$k=2$	$k=3$	$k=4$
a_1b_1	$a_1a_2b_1b_2$	$a_1a_2a_3b_1b_2b_3$	$a_1a_2a_3a_4b_1b_2b_3b_4$
			$a_1a_2a_3b_1a_4b_2b_3b_4$
			$a_1a_2a_3b_1b_2a_4b_3b_4$
			$a_1a_2a_3b_1b_2b_3a_4b_4$
		$a_1a_2b_1a_3b_2b_3$	$a_1a_2b_1a_3a_4b_2b_3b_4$
			$a_1a_2b_1a_3b_2a_4b_3b_4$
			$a_1a_2b_1a_3b_2b_3a_4b_4$
		$a_1a_2b_1b_2a_3b_3$	$a_1a_2b_1b_2a_3a_4b_3b_4$
			$a_1a_2b_1b_2a_3b_3a_4b_4$
	$a_1b_1a_2b_2$	$a_1b_1a_2a_3b_2b_3$	$a_1b_1a_2a_3a_4b_2b_3b_4$
			$a_1b_1a_2a_3b_2a_4b_3b_4$
			$a_1b_1a_2a_3b_2b_3a_4b_4$
		$a_1b_1a_2b_2a_3b_3$	$a_1b_1a_2b_2a_3a_4b_3b_4$
			$a_1b_1a_2b_2a_3b_3a_4b_4$

Table A.5: Possible transfer combinations for example-3

If we look at table A.5, when we generate the combinations for $k = i + 1$ from the combinations for $k = i$, it is clear that b_{i+1} must be placed at the end of each of the new combinations. And a_{i+1} can be placed anywhere after a_i. Thus, if any combination for $k = i$ ends with consecutive n b_ls (that means a_i is placed just before those consecutive b_ls), then a_{i+1} can be placed anywhere within those consecutive n b_ls and thus can make $n+1$ new combinations. b_{i+1} is placed at the end of each of the new combinations. Thus, out of these $n + i$ new combinations, one ends with exactly 1 b_l, one ends with exactly 2 b_ls, one ends with exactly 3 b_ls, . . . , one ends with exactly $n + 1$ b_ls. Each of these $n + 1$ combinations can then generate new combinations for $k = i + 2$ following the same rule.

In this example, for $k = 1$ the only possible combination is a_1b_1. This combination ends with 1 b_l (i.e. b_1). Now when we want to generate the combinations for $k = 2$, a_2 can be placed on either side of b_1. b_2 must be placed at the end. Thus two new combinations ($a_1a_2b_1b_2$ and $a_1b_1a_2b_2$) are generated. Out of these two combinations, the first one ends with 2 consecutive b_ls and the last one ends with 1 b_l.

Thus, at any level k if there are total n possible combinations, and out of those n combinations if n_j number of combinations end with exactly j number of b_ls ($1 \leq j \leq k$, $\sum n_j = n$), then on next level each of these n_j combinations will generate $n_j + 1$ combinations. And out of these $n_j + 1$ combinations 1 will end with exactly one b_l, 1 will end with exactly two consecutive b_ls, . . . , 1 will end with exactly $n_j + 1$ consecutive b_ls. As in the example for $k = 3$ there are total 5 possible combinations. Out of these 5 combinations, $n_1 = 2$ combinations end with exactly one b_l, $n_2 = 2$ combinations end with exactly two consecutive b_ls, $n_3 = 1$ combination ends with exactly three consecutive b_ls. Each of the 2 combinations which ends with exactly two consecutive b_ls will generate 1 combination with exactly one b_l at the end, 1 combination with exactly two b_ls at the end and 1 combination with exactly three b_l at the end.

Now if we want to know that how many combinations will be there for $k = i + 1$ those end with exactly j consecutive b_ls, we have to count the number of combinations for $k = i$ those can generate such combinations on next level. The combinations for $k = i$ that ends with at least $j - 1$ consecutive b_ls, will generate the combinations for $k = i + 1$ that ends with exactly j consecutive b_ls. This calculation is defined in table A.6.

Thus, we get,

$$\psi(k) = \sum_{j=1}^{k} a_{k,j} \tag{A.1}$$

$$\text{where } a_{i,j} = \sum_{p=max(1,j-1)}^{i-1} a_{i-1,p} \text{ and } a_{1,1} = 1$$

		number of b_ls at end				number of combinations
		1	2	3	4	
k	1	$a_{1,1}$ $= 1$				$\psi(1) =$ $\sum a_{1,j} = 1$
	2	$a_{2,1} =$ $a_{1,1}$ $= 1$	$a_{2,2} =$ $a_{1,1}$ $= 1$			$\psi(2) =$ $\sum a_{2,j}$ $= 2$
	3	$a_{3,1} =$ $a_{2,1} + a_{2,2}$ $= 2$	$a_{3,2} =$ $a_{2,1} + a_{2,2}$ $= 2$	$a_{3,3} =$ $a_{2,2}$ $= 1$		$\psi(3) =$ $\sum a_{3,j}$ $= 5$
	4	$a_{4,1} =$ $a_{3,1} + a_{3,2} + a_{3,3}$ $= 5$	$a_{4,2} =$ $a_{3,1} + a_{3,2} + a_{3,3}$ $= 5$	$a_{4,3} =$ $a_{3,2} + a_{3,3}$ $= 3$	$a_{4,4} =$ $a_{3,3}$ $= 1$	$\psi(4) =$ $\sum a_{4,j}$ $= 14$

Table A.6: Calculation of $\psi(k)$ for example-3

We can calculate $\psi(k)$ in another way also. The transfer scenario mentioned in this section can be compared with the transfer scenario when we calculated $\phi_1(k)$ in Section A.1. In the calculation of $\phi_1(k)$ if we set $u_1 = 0$, $x_2 = 0$ and $k = x_1$, then $\phi_1(k)$ becomes $\psi(k)$. Thus we can say,

$$\psi(k) = a_{k,0} \tag{A.2}$$

$$\text{where } a_{i,j} = \sum_{p=j}^{k-i+1} a_{i-1,p}, \text{ for } i > 0$$

and $a_{0,j} = \delta_{jk}$, where δ_{ij} is Kronecker delta function.

Finally, $\psi(k)$ can be calculated using either Equation A.1 or Equation A.2.

Bibliography

[1] IEEE 802.11. IEEE Standard 802.11 - Wireless LAN Medium Access Control (MAC) and Physical Layer Specifications, 1997.

[2] IEEE 802.16a. IEEE Standard for Local and metropolitan area networks-Part 16: Air Interface for Fixed Broadband Wireless Access Systems- Amendment 2: Medium Access Control Modifications and Additional Physical Layer Specifications for 2-11 GHz, 2003.

[3] Bay area wireless users group. `http://www.bawug.org/`.

[4] B. Awerbuch, D. Holmer, and H. Rubens. High Throughput Route Selection in Multi-rate Ad-hoc Wireless Networks. In *Technical Report, John Hopkins University*, 2003.

[5] D. Beyer, M. Vestrich, and J. Aceves. The Rooftop Community Network: Free, High-speed Network Access for Communities. In *The First 100: New Options for Internet and Broadband Access*, pages 75–91, 1999.

[6] M. Cao, W. Ma, Q. Zhang, X. Wang, and W. Zhu. Modeling and Performance Analysis of the Distributed Scheduler in IEEE 802.16 Mesh Mode. In *MobiHoc 2005: 6th ACM international symposium on Mobile ad hoc networking and computing*, pages 78–89, May 2005.

[7] K. Chin, J. Judge, A. Williams, and R. Kermode. Implementation Experience with MANET Routing Protocols. In *ACM CCR*, November 2002.

[8] D. Couto, D. Aguayo, J. Bicket, and R. Morris. High Throughput Path Metric for Multi-hop Wireless Routing. In *MOBICOM*, 2003.

[9] R. Draves, J. Padhye, and B. Zill. Routing in Multi-Radio, Multi-Hop Wireless Mesh Networks. In *MobiCom 2004: ACM international symposium on Mobile Computing*, pages 114–128, September 2004.

[10] R. Dube, C. Rais, K. Wang, and S. Tripathi. Signal Stability Based Adaptive Routing for Ad-hoc Mobile Networks. In *IEEE Personal Comm*, February 1997.

[11] T. Goff, N. Abu-Aahazaleh, D. Phatak, and R. Kahvecioglu. Preemptive Routing in Ad-hoc Networks. In *MOBICOM*, 2001.

[12] E. Grabianowski and M. Brain. How WiMax Works. HowStuffWorks Inc. `http://computer.howstuffworks.com/wimax.htm/printable`, 2006.

[13] Y. Hu and D. Johnson. Design and Demonstration of Live Audio and Video over Multi-hop Wireless Networks. In *MILCOM*, 2002.

[14] D. Johnson and D. Maltaz. *Mobile Computing*, chapter Dynamic Source Routing in Ad-hoc Wireless Networks. Kluwer Academic Publishers, 1996.

[15] J. Jubin and J. Tornow. The DARPA Packet Radio Network Protocols. In *Proceedings of The IEEE, 75(1)*, pages 21–32, January 1987.

[16] R. Karrer, A. Sabharwal, and E. Knightly. Enabling Large-scale Wireless Broadband: The Case for TAPs. In *HotNets)*, 2003.

[17] S. Keshav. *An Engineering Approach to Computer Networking.* Addison-Wesley Professional Computing Series, 1997.

[18] L. Kleinrock. *Communication Nets: Stochastic Message Flow and Delay.* Dover Publications Inc., New York, 1964.

[19] S. Lee and M. Gerla. AODV-BR: Backup Routing in Ad-hoc Network. In *IEEE WCNC'00*, pages 1311–1316, 2000.

[20] M. Mauve, J. Widmer, and H. Hartenstein. A Survey on Position-Based Routing in Mobile Ad-Hoc Networks. In *IEEE Network*, pages 30–39, November 2001.

[21] E. Meyer. WiMAX vs WiFi. TechwareLabs. `http://www.techwarelabs.com/articles/other/wimax_wifi`, 2006.

[22] M. Norris. *Mobile IP Technology for M-Business.* Mobile Communication Series, 2001.

[23] Intel White Paper. Understanding Wi-Fi and WiMAX as Metro-Access Solutions. Intel White Paper, Wi-Fi and WiMAX Solutions. `http://www.intel.com/netcomms/technologies/wimax/304471.pdf`.

[24] Nokia White Paper. Nokia RoofTop Wireless Routing. Nokia Wireless Broadband, Nokia Networks. `http://www.americasnetwork.com/americasnetwork/data/articlebrief/americasnetwork/412002/34898/article.pdf`.

[25] C. Perkins and P. Bhagwat. Highly Dynamic Destination-Sequenced Distance-Vector Routing for Mobile Computers. In *Proceedings of The SIGCOMM'94 Conference on Communications, Architectures, Protocols and Applications*, pages 234–244, August 1994.

[26] M. Pioro and D. Medhi. *Routing, Flow, and Capacity Design in Communication and Computer Networks.* Elsevier, 2004.

[27] MIT Roofnet. `http://www.pods.lcs.mit.edu/roofnet/`.

[28] N. Shacham and J. Westcott. Future Directions in Packet Radio Architectures and Protocols. In *Proceedings of The IEEE, 751*, pages 83–99, January 1987.

[29] Smith and Clint. *3G Wireless with WiMax and WiFi: 802.16 and 802.11.* McGraw-Hill, New York, 2005.

[30] H. Wei, S. Ganguly, R. Izmailov, and Z. Haas. Interference-Aware IEEE 802.16 WiMax Mesh Networks. In *61st IEEE Vehicular Technology Conference (VTC 2005 Sprint)*, Stockholm, Sweden, May 2005.

[31] Seattle Wireless. `http://www.seattlewireless.net/`.

[32] A. Woo, T. Tong, and D. Culler. Taming The Underlying Challenges of Reliable Multihop Routing in Sensor Networks. In *SenSys*, 2003.

[33] H. Zhu and K. Lu. On The Interference Modeling Issues for Coordinated Distributed Scheduling in IEEE 802.16 Mesh Networks. In *BROADNETS 2006: 3rd International Conference on Broadband Communications, Networks and Systems*, San Jose, California, USA, October 2006.