

CHANNEL ASSIGNMENT AND ROUTING FOR MULTI-RADIO WIRELESS MESH NETWORKS

A THESIS

SUBMITTED TO THE DEPARTMENT OF COMPUTER ENGINEERING
AND THE INSTITUTE OF ENGINEERING AND SCIENCE
OF BILKENT UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
MASTER OF SCIENCE

By
Ahmet Murat Özdemiray
September, 2008

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Assist. Prof. Dr. İbrahim Körpeođlu(Advisor)

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Dr. Cengiz elik

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Prof. Dr. Adnan Yazıcı

Approved for the Institute of Engineering and Science:

Prof. Dr. Mehmet B. Baray
Director of the Institute

ABSTRACT

CHANNEL ASSIGNMENT AND ROUTING FOR MULTI-RADIO WIRELESS MESH NETWORKS

Ahmet Murat Özdemiray

M.S. in Computer Engineering

Supervisor: Assist. Prof. Dr. İbrahim Körpeoğlu

September, 2008

Wireless Mesh Network is a promising technology since it extends the range of wireless coverage by multi-hop transmission between routers. However, in multi-hop networks the total throughput decreases with increasing number of nodes and hops. To increase the total throughput, some mesh routers are equipped with multiple radios to use the available bandwidth of multiple non-overlapping channels. However, channel assignment should be done carefully to effectively use this available bandwidth. Moreover, the optimal channel assignment algorithm is NP-hard. In this thesis, we propose a joint channel assignment and routing solution to effectively use the available bandwidth for multi-radio wireless mesh networks with given network topology and traffic profile. Initially, we predict the final routes of the flows and estimate the loads on the links using these path predictions and given traffic profile. Then three different heuristics determine the assignment order of the links. Then the least busy channel among the available channels is assigned to the link. Finally, our routing algorithm routes the flows such that the selected path is the least busy path among the alternatives. We evaluated our channel assignment and routing algorithm using ns-2 simulator which supports multiple channels and multiple radios per node and we compared our results with single channel WMNs, and different algorithms for multi-radio multi-channel WMNs. The results show that our joint algorithm successfully achieves up to 5 times more throughput than single channel WMN with using just 2 radios and 3 channels. Our algorithms also out-performs other compared channel assignment algorithms for multi-radio multi-channel WMNs.

Keywords: Wireless Mesh Network, Channel Assignment, Multi-channel.

ÖZET

ÇOK RADYOLU KABLOSUZ ÖRGÜ AĞLARDA KANAL TAHSİSİ VE YÖNLENDİRME

Ahmet Murat Özdemiray

Bilgisayar Mühendisliği, Yüksek Lisans

Tez Yöneticisi: Yrd. Doç. Dr. İbrahim Körpeoğlu

Eylül, 2008

Kablosuz Örgü Ağlar yönlendiriciler arası çoklu atlamalı haberleşme yaparak kablosuz kapsama alanını genişlettikleri için geleceği parlak olan bir teknolojidir. Ama, çoklu atlamalı ağlarda atlama ve düğüm sayısı artarken ağın toplam iş gücü düşmektedir. Toplam iş gücünü artırmak için, bazı örgü yönlendiricileri birden fazla radyoyla donatılarak çok sayıdaki örtüşmeyen kanalların kullanılabilir bant genişliğinden faydalanabilmektedir. Fakat, bu bant genişliğini verimli kullanabilmek için de kanal tahsisi dikkatli bir şekilde yapılmalıdır. Üstelik, en iyi kanal tahsisi algoritması da NP-Zordur. Bu tezde, topolojisi ve trafik profili verilen bir çok radyolu kablosuz örgü ağının kullanılabilir bant genişliğinin verimli şekilde kullanması için birleşik bir kanal tahsisi ve yönlendirme çözümü öneriyoruz. İlk başta, akışların son yollarını öngörüyor ve verilen trafik profili ve öngörülen yolları kullanarak bağların yükünü tahmin ediyoruz. Daha sonra üç değişik buluşsal ile bağların tahsis sırasını belirliyoruz. Daha sonra, sırası gelen bağa kullanılabilir kanallar arasında en az meşgul olanını tahsis ediyoruz. Son olarak, yönlendirme algoritmamız akışları alternatifler arasında en az yoğun olan yoldan yönlendiriyor. Çok radyolu ve çok kanallı kablosuz ağları destekleyen bir Ns-2 simülatörünü kullanarak kanal tahsis ve yönlendirme algoritmamızı değerlendirdik ve sonuçlarımızı tek kanallı ve çok radyolu çok kanallı kablosuz örgü ağlarında kanal tahsisi için kullanılan çeşitli algoritmaların sonuçlarıyla karşılaştırdık. Sonuçlar gösterdi ki, birleşik kanal tahsisi ve yönlendirme algoritmamız sadece 2 radyo ve 3 kanal kullanarak tek kanallı ağdan yaklaşık 5 kat fazla iş gücü yarattı. Ayrıca algoritmamız diğer çok radyolu çok kanallı algoritmalarından daha başarılı sonuçlar elde etti.

Anahtar sözcükler: Kablosuz Örgü Ağlar, Kanal Tahsisi, Çok Kanallı.

To my parents and wife...

Acknowledgement

I would like to express my gratitude to my supervisor Assist. Prof. Dr. İbrahim Körpeoğlu for his realistic, encouraging and constructive comments throughout my masters study and his efforts and patience during supervision of the thesis.

I would like to thank Dr. Cengiz Çelik, and Prof. Dr. Adnan Yazıcı for kindly accepting to spend their valuable time to evaluate my thesis.

I thank to the members of Networking and Systems Group at Bilkent University during the years 2005 to 2008 for their comments on my studies and for the sparks they caused.

I also thank to the Scientific and Technical Research Council of Turkey (TÜBİTAK) for its support to the project with grant number 104E028.

I would like to express my appreciation to TÜBİTAK UEKAE / ILTAREN for the understanding and support during my academic studies. I also want to thank to people I work with for the joy they bring to my life making my business life together with academic life a pleasurable experience.

I would like to express my thanks to my parents, making me who I am now with their love, trust, freedom, understanding and every kind of support throughout my life.

Finally, I would like to express my appreciation to my wife, whose love, patience, support, and encouragement helped me to finish this study.

Contents

- 1 Introduction** **1**

- 2 Background** **6**
 - 2.1 Wireless Mesh Networks 6
 - 2.1.1 MAC protocols for Wireless Mesh Networks 8
 - 2.1.2 Routing Protocols for Wireless Mesh Networks 9

- 3 Related Work** **11**
 - 3.1 Single-Radio Multi-Channel WMNs 11
 - 3.2 Multi-Radio Multi-Channel WMNs 12

- 4 Our Solution** **17**
 - 4.1 System Model and Assumptions 17
 - 4.2 Algorithm Overview 19
 - 4.3 Initial Load Estimation 20
 - 4.3.1 Finding all min-cost paths 21

4.3.2	Distributing Estimated Traffic Among Paths	24
4.4	Channel Assignments	25
4.4.1	Selection order of Links	25
4.4.2	Assign Channel For a Link	26
4.5	Final Routing	28
4.6	Discussions	29
5	Evaluation	31
5.1	Simulation Scenarios	31
5.1.1	WMN with one gateway to Internet Access	33
5.1.2	WMN with 5 servers	35
5.1.3	WMNs with random peer to peer traffic profiles	36
5.2	Effect of Routing Algorithms	47
5.3	Discussion on Comparison Results	48
6	Conclusion and Future Work	49
6.1	Conclusion	49
6.2	Future Work	50
	Bibliography	51
A	Table of Acronyms	57

List of Figures

1.1	A single-radio multi-channel wireless mesh network.	2
1.2	A multi-radio multi-channel wireless mesh network.	3
4.1	System architecture for proposed multi channel wireless mesh network.	18
5.1	WMN with one gateway. Total number of received packets.	34
5.2	WMN with one gateway. Reliability of the paths.	35
5.3	WMN with 5 servers.	37
5.4	WMN with 5 servers.	38
5.5	WMN with random traffic profile 1.	39
5.6	WMN with random traffic profile 1.	40
5.7	WMN with random traffic profile 2.	42
5.8	WMN with random traffic profile 2.	43
5.9	WMN with random traffic profile 2.	45
5.10	WMN with random traffic profile 3.	46

5.11 Effect of Routing Algorithms. 48

List of Tables

5.1	Simulation Results for WMN with 1 gateway	33
5.2	Simulation Results for WMN with 5 servers	35
5.3	Simulation Results for Random Traffic Profile #1	41
5.4	Simulation Results for Random Traffic Profile #2	41
5.5	Simulation Results for Random Traffic Profile #3	44

List of Algorithms

4.1	Finding all min-cost paths for a source.	23
4.2	Distributing traffic among initial paths.	24
4.3	Evaluate channel.	27
4.4	Assign channel for link.	28
4.5	Set channel for link	29

Chapter 1

Introduction

Wireless networks become part of daily life. Day by day, every hotel, mall, cafe, restaurant etc. are covered by wireless internet zones. Not only laptop computers and PDAs are used to connect to wireless networks, but also some of the new mobile phones are also equipped with wireless network cards to connect to the wireless access points to get high-speed Internet access. In order to extend the limited range of wireless access points, multiple access points are deployed to the zones to compose wireless mesh networks.

A wireless mesh network is a type of *wireless ad hoc network* which consists of mesh routers and mesh clients. Mesh routers are usually stationary and they form the backbone of the WMNs, while serving as access points for mesh clients, which can be stationary or mobile. Some of the routers act as Internet gateways to provide Internet access to the clients in the WMN. Other routers contain additional multi-hop routing capabilities to forward packets to reach their destinations. WMNs are used as wireless backbones in campuses, enterprises and by ISPs to provide last-mile Internet access to users.

In order to communicate with other mesh routers, routers should use common channels. However, traditional wireless routers are equipped with single wireless network interface card (NIC), therefore whole network uses only one channel.

Furthermore, in multi-hop settings throughput decreases significantly with increasing number of nodes and hops [19]. Therefore improving throughput is a major concern in WMN research.

IEEE 802.11 standards [20] define 12 and 3 non-overlapping frequency channels for 802.11a and 802.11b/g respectively which can be used simultaneously to increase the aggregate bandwidth. However, this available bandwidth cannot be used by wireless mesh networks without any hardware or software modification on the routers. There are several approaches to use multiple channels to increase the throughput of the WMN. These approaches can be categorized by the number of radios used by each router.

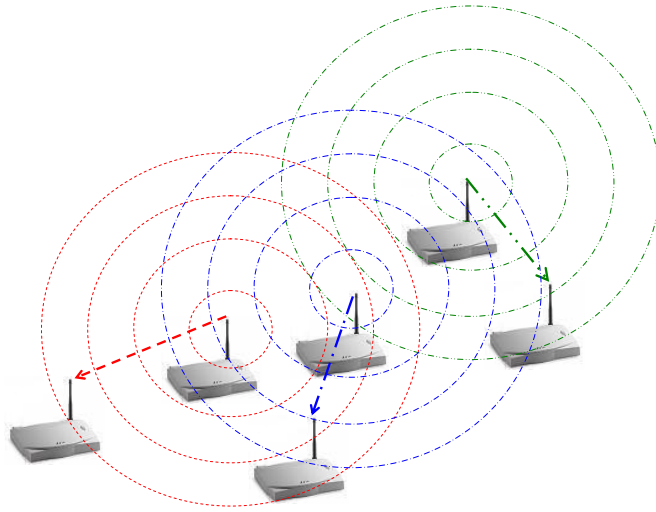


Figure 1.1: A single-radio multi-channel wireless mesh network.

Single radio approaches for using multiple non-overlapping channels to increase the available bandwidth for WMNs generally require MAC layer modification and per packet channel switching capability for wireless radios. As seen in Figure 1.1 simultaneous communication can be done by using different channels. Before every data transmission, a channel selection mechanism evaluates the available channels and selects a channel to transmit. There are also some problems

introduced with channel switching mechanism. These problems include multi-channel hidden terminal problem, deafness problem, channel deadlock problem and broadcast problem [39].

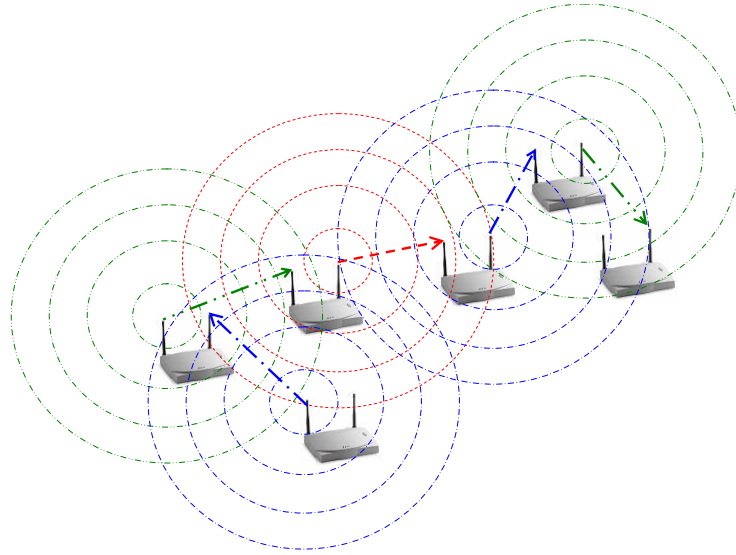


Figure 1.2: A multi-radio multi-channel wireless mesh network.

There are also approaches which introduce multiple NICs to wireless routers to utilize the available bandwidth resulting from non-overlapping frequency channels. Each radio in a multi-radio router is tuned to a different channel to allow simultaneous communication by these NICs. Each NIC has its own physical and MAC layers and a layer on top of these NICs coordinates them. Therefore multi-radio approaches do not need a coordination mechanism embedded into MAC layer to communicate simultaneously, so they can work with commodity 802.11 routers. As seen in Figure 1.2 a router can receive data from one NIC while it is transmitting traffic to other NIC.

Crichigno et al. [14] categorizes the multi-radio channel assignment approaches as *dynamic channel assignments*, *semi-dynamic channel assignments* and *static channel assignments* by the frequency of changes in the channel assignments. Like single-radio approaches, *dynamic channel assignments* switch channels per packet basis. Since channel switching cost is still present, they cannot fully utilize

the available bandwidth. *Semi-dynamic channel assignment* approaches change assignments at a relatively slow time scale, by minutes, hours or days. Either a central assignment server [9], a central administrator [31] or a distributed mechanism [24], [30] changes channel assignments according to external interference changes or load changes. In static channel assignment approaches [6], [17], channel assignments do not change after the network setup. Generally a uniform assignment is made which assigns channel 1 to NIC 1, channel 2 to NIC 2, etc.

In this thesis we propose a load aware channel assignment and routing algorithm for multi-radio multi-channel wireless mesh networks. Given network topology and traffic profile between nodes, our channel assignment algorithm makes an estimation of the final loads of the links. Then using this estimation, it selects a channel depending on the estimated loads of other links in the interference range and assigned channels to its incident routers' NICs. After channel assignment algorithm assigns a channel to each link, the routing algorithm routes each flow such that none of the links will be bottleneck for the network.

Since our approach takes the traffic profile into account, our algorithm assigns channels to links considering their expected loads and interference effects of other links in the neighborhood which are assigned to the same channel. Therefore, a fairness among the loads of channels is established which increases the utilization of the channel bandwidths. Furthermore, our routing algorithm considers the expected loads of links while routing flows, so it tries to balance the loads on links. Although this approach is similar to LACA [31], their routing algorithms do not consider the traffic profile while estimating loads and routing flows. And also, in their approach, if a pair of routers is within the communication range, then there is one bi-directional link between this pair, which use only one channel and can be used to send or transmit at a time. However, in our approach there are two directional links between a pair, which may use different channels. Therefore, we allow simultaneous 2-way communication between a pair of routers.

We implemented our heuristic algorithms to create channel assignments and routing for given network topology and traffic profile. Then we evaluated the assignments and routing in a modified ns-2 simulator to support multiple NICs

and multiple non-overlapping channels to compare our results with single channel WMNs, a random channel assignment with shortest path routing and LACA's [31] results for the same topology and 5 different traffic profiles. The simulations show that we have improved the total network throughput, increased the number of reliable paths (i.e. the paths which have no or little packet loss) and decreased the average delay and delay variation over the flows by significant amounts.

The remaining of the thesis is organized as follows. In the next chapter we provide some background information about wireless mesh networks. In Chapter 3, we describe previous studies related to our work. Then in Chapter 4, we give detailed description of our solution. And in Chapter 5 we evaluate our solution and provide some results with comparisons. Finally we provide our conclusion and discuss some future work in Chapter 6

Chapter 2

Background

This chapter gives information about Wireless Mesh Networks to provide some background information before we describe our solution in detail.

2.1 Wireless Mesh Networks

A Wireless Mesh Network is a type of wireless ad hoc network which consists of wireless mesh routers and wireless mesh clients. Each router in the network operates as a host to its clients, serving as a gateway/bridge to forward their data like conventional access points. Moreover, routers are capable of forwarding other routers' traffic to support multi-hop communication and they construct the backbone of the wireless mesh network.

An application area of WMNs is providing last-mile broadband Internet access by ISPs, in which some of the nodes are connected to the Internet via physical wires (i.e. gateways) and other nodes relay traffic to/from these nodes to provide Internet connectivity to their clients [30]. Other application areas of WMNs include but not limited to wireless enterprise backbones, building automation, broadband home networking, community and neighborhood networks [10], video

surveillance and perimeter security, mines and industrial sites, military communication, sports events, emergency and hastily formed networks, rail and highway corridors, VOIP phone applications [2], etc.

It is easy to construct a WMN because all of the required components are easy to purchase and most of these components are self-organized and self-configurable to construct the WMN [1]. Also, there are several commercial solutions to deploy WMNs [2], [3], [4], [5]. However, scalability is still a problem since throughput drops significantly as the number of nodes or hops in a wireless multi-hop network increases [19]. There are also other critical factors [8], [36] which influence the design and performance of WMNs. These factors are:

- *Radio techniques.* With the rapid development of the semiconductor technology, wireless radios are also evolving. Many approaches including directional antennas and multi-radio chipsets require design changes in some layers since they introduce new problems while solving some of the existing problems.
- *Scalability.* Routers in a WMN forward their clients' packets using multi-hop communication. It is known that multi-hop communication suffers from scalability. Routing protocols may fail to find a reliable path, transport protocols may lose connections and MAC protocols may suffer from throughput decrease because of contention.
- *Broadband and QoS.* The applications of WMNs are generally broadband services with QoS requirements. Therefore, performance metrics such as delay, jitter, total throughput and packet loss ratio should also be considered by new protocols.
- *Security.* Security is a huge concern which discourages subscribers to use the mesh network. Current security solutions of ad hoc networks are immature and the architecture of WMNs are different from conventional ad hoc networks. Therefore, new security schemes need to be developed for WMNs.

- *Routing.* Wireless links are affected by interference and multi-path fading. Therefore, routing solutions should consider radio-related metrics. Furthermore, routing protocols for ad hoc networks introduce overhead to handle link failures. However, in mesh networks link failures rarely occur, therefore, the overhead is intolerable.
- *Seamless Roaming.* Mobility of the mesh clients should also be considered to provide seamless connectivity for them.

2.1.1 MAC protocols for Wireless Mesh Networks

There are two different approaches to consider unique features and critical factors affecting the performance of WMNs in MAC layer. First way is improving existing 802.11 MAC protocols to increase throughput of the network when there is only one channel is available. Second way is to allow communications on different channels.

2.1.1.1 Single channel MAC for WMNs

CSMA/CA protocol is designed for single-hop WLAN environment and it cannot meet the requirements of WMNs such as capacity, throughput, latency etc [40]. Therefore it needs to be modified to guarantee end-to-end delay, increase throughput, etc. Some MAC protocols are proposed for multi-hop ad hoc networks by modifying existing CSMA/CA protocol. By changing backoff procedure and contention window sizes of CSMA/CA, they improved the throughput of one-hop communication. However, multi-hop communication and interference of nodes within two or more hops is not considered, resulting contentions among neighboring nodes. There are also some approaches which introduce directional antennas to decrease the number of exposed nodes by a transmission. However, more hidden nodes are produced by this approach. Moreover, cost and complexity of the system increases by introducing directional antennas [8].

2.1.1.2 Multi channel MAC for WMNs

There are also other approaches which try to improve the WMN's performance by using multiple channels. These approaches can further be categorized as *single-radio multi-channel* and *multi-radio multi-channel* MACs. The approaches to use multiple channels with single radio ([37], [38], [11], [33]) require synchronization among the nodes and they also depend on radios with fast channel switching capability. They also need to deal with the new problems introduced by the channel switching capability. These problems include deafness, broadcast, deadlock and multi-channel hidden terminal problems [39]. The MAC layer approaches which use multiple radios to utilize the available bandwidth of multiple channels work as a *virtual layer* to coordinate the multiple radios which have their own MAC layer. If the number of available radios is less than number of available channels, a channel assignment strategy is needed to assign a channel to a radio which may introduce network partitioning, channel dependency and topology alteration problems [14].

2.1.2 Routing Protocols for Wireless Mesh Networks

Since WMN is a type of wireless ad hoc network, routing protocols designed for ad hoc networks can be used by WMNs. As a matter of fact, some of the commercial wireless mesh solutions use these routing protocols [8]. However, because of the different requirements of WMNs, new routing protocols are needed to improve performance. For example, scalability is a major concern in wireless mesh networks, while existing routing protocols provide limited scalability [22], [27]. Furthermore, nodes in WMN are generally stationary and they do not have power constraints. However, routing protocols for ad hoc networks have extra cost to deal with mobility [7], [12], [18] and energy constraints [23], [35].

There are various methods for routing in WMNs. In LQSR [16], three different link quality metrics is considered to find the paths using DSR like protocol. The metric with expected transmission count, ETX [13], performed the best, however, the changes in link quality cannot be reflected to the routing quickly [8]. In

[17], same authors proposed a new routing algorithm for multi radio routing and they introduced a new metric called *weighted cumulative expected transmission time* (WCETT), which considers both ETX and min hop-count. However, their proposed channel assignment uses as many channels as the number of their radios, so they cannot use the available bandwidth of other channels. A hierarchical routing algorithm for multi radio WMNs is proposed in [30]. The algorithm converts the mesh topology into a multi-tree topology with gateway routers as the roots of the trees. However, since the topology is changed by assigning different channels to neighbor routers, the support for peer to peer communication between routers is ignored.

Chapter 3

Related Work

In this chapter we briefly describe some of the previous studies which use multiple channels in wireless networks to increase available bandwidth. Firstly, channel assignment algorithms for single-radio multi-channel wireless networks are mentioned and then channel assignment approaches for multi-radio WMNs are presented.

3.1 Single-Radio Multi-Channel WMNs

Single radio approaches for using multiple non-overlapping channels to increase the available bandwidth for wireless networks require MAC layer modification (Section 2.1.1.2) and per packet channel switching capability for wireless radios. Before every data exchange process, a channel selection mechanism evaluates the available channels and a channel is selected to communicate.

One of the approaches to use multiple channels with one radio is *channel hopping* method. In [37], [38] Tzamaloukas et al. proposed common hopping technique in which all of the nodes listen to the same channel at the same time. A node who wants to communicate with another node starts a handshake procedure

and they remain in the same channel to exchange data, while other nodes continue hopping in the common sequence. In [11], a slotted seeded channel hopping method is proposed in which two nodes exchange a schedule and meet periodically on a common channel to communicate. If hopping schedules are disjoint, the nodes do not interfere with each other and communicate simultaneously on different channels. Both approaches require a strict synchronization of the nodes which is hard to achieve. Furthermore, commercial off-the-shelf 802.11b radios require about 150 - 200 μ sec to switch channel [26] which is not negligible.

Another approach to utilize available bandwidth using multiple channels is *split phase* protocols. In these protocols, time is divided into control and data exchange phases. In control phase, all nodes use one channel and send data exchange requests on this channel. Then when control phase ends, the pairs which are agreed to exchange data meet in another channel and exchange data. In [33] the duration of control and data exchange phases are fixed, while in [21] nodes make an agreement about the duration of the data exchange phase.

3.2 Multi-Radio Multi-Channel WMNs

There are also some approaches which introduce multiple radios to wireless routers in order to utilize the available bandwidth resulting from non-overlapping frequency channels.

Identical channel assignment scheme is used in [17] and [6]. In this scheme the channel assignments of all nodes are the same where 1st channel is assigned to 1st NIC, 2nd channel assigned to 2nd NIC, etc. When a node wants to transmit some data, it looks for the channel with best condition and transmits data on that channel. This scheme is easy to implement and deploy, however, the resources are wasted because number of channels used is equal to the number of NICs. In our proposed method, on the other hand, a maximum number of $2q - 1$ channels can be used where q is the number of NICs.

In [31], a centralized load-aware channel assignment and routing algorithm

is proposed, which uses the traffic profile of the network to allocate required bandwidth to links. They assume that there exists an undirectional *virtual link* between two routers if the distance between them is less than or equal to the communication range of their radios. The method has three phases. Initially, the expected loads on *virtual links* are estimated by finding min-hop shortest paths between each pair which has expected flow and sending traffic on these paths. In the second phase, channel assignment algorithm starts its first iteration using the estimated loads calculated in the first phase and visits all of the links in the decreasing order of expected load on link. For each link, a channel which can be used by the incidents of the link is selected such that the sum of expected loads of the links in the interference range which use the same channel is minimum. If the incidents of the link does not share a common channel, then a recursive channel replacement procedure starts to alter the channel assignments of previously visited links. After a channel is assigned to each link, shortest path or randomized multi-path routing is used to route flows and the resulting capacities of the links are calculated. If the capacities of all links are greater than the amount of flow passing on them, or there is no significant improvement on the last iteration then algorithm is terminated. If any link's capacity is less than the amount of flow passing on it, or there is some improvement on the last iteration, then the channel assignment process restarts using the expected loads of previous iteration. Our proposed method has also same three phases. We estimate initial loads, we assign channels and we route the flows. However, we do not use min-hop shortest path algorithms for initial load estimation and final routing. Our routing algorithms consider the expected traffic on paths, and try to route flows on less busy paths. And also our proposed method is deterministic (i.e. finds same channel assignment and routing for the same input).

Raniwala et al. proposed a multi-channel WMN architecture called *Hyacinth* and defined a distributed channel assignment algorithm for this network in [30]. The topology of *Hyacinth* has multiple spanning trees which have Internet gateways as the roots. They defined a protocol for topology discovery and each node starts a route discovery procedure to connect to a routing tree. The channel assignment procedure starts from Internet gateways and nodes are visited in the

order of their level on the trees. Each node is responsible for assigning channels to its children considering the expected load on channels. Since topology of the network is altered, peer to peer communications may suffer from longer delays. Unlike *Hyacinth*, our proposed method does not modify the topology and takes the traffic between wireless routers into consideration and assigns channels and finds efficient routes for multi-purpose WMNs (i.e. WMNs which are not only used for Internet access).

In [9] a joint approach is used to solve channel assignment, routing and scheduling problems jointly. It assumes that the traffic demands of the nodes are known and system operates synchronously in a time slotted mode. The solution is composed of 5 phases. An LP is formulated to maximize system throughput while routing given traffic demands subject to constraints of on channel fairness, number of NICs, expected traffic of the links in the interference range. Since the problem becomes NP-hard, a relaxation of LP is solved. Then channel assignment algorithm is used to adjust the flow on the flow graph to ensure a feasible channel assignment. In post-processing phase and flow scaling phase the flows are re-adjusted to ensure that the maximum interference over all channels is minimized. In the final phase the flows are scheduled to create interference free flows. Our proposed method does not need any synchronization between routers, so it can work successfully with commercial wireless routers.

In [25], Kyasanur et al. proposed a hybrid channel assignment scheme where some radios use fixed channels and other radios change their channels dynamically. The fixed radio is used for control messages or data reception. Other radios switch channels to static channels of neighbors when the node has a packet to transmit. As we said in Section 3.1, channel switching requires an extra time which is much higher than the transmission delays. Therefore throughput is affected by channel switches. Our proposed method uses only static channel assignments which will change rarely.

Tang et al. proposed a centralized interference-aware channel assignment in [34], which visits the links in decreasing order of the number of links in the interference region and selects the least used channel in that region. In channel

assignment phase some links cannot be assigned a channel because no common channel is found between incidents of the link. In routing phase, both an optimal algorithm based on solving an LP and a simple heuristic is proposed to route the traffic. Since topology is changed, this may lead to inefficient routes for peer to peer connections. Our proposed method does not alter the network topology.

In [10], a centralized channel assignment and routing algorithm (MCAR) is proposed to maximize the throughput. In order to prevent channel assignment to propagate through previously assigned channels which is caused by not having a common channel between incidents of a link as in [31], their algorithm considers the links outside the interference range while selecting a channel to assign to the link but they do not consider any traffic profile. They assign a group to each link based on flows they carry. Then they assign a channel to each group to maximize throughput. Since they do not consider the traffic profile of the network, there may be bottleneck links if the paths containing that link has excessive flow demands. Our proposed method routes flows according to the flow demands on the paths and tries to balance the traffic passing on the links using probably longer paths which have less loaded links.

In [28], [24], [32], a default channel is used to preserve connectivity. Although this approach simplifies the channel assignment process, it can be wasteful when only a few interfaces are available.

In [28] a multi-radio conflict graph is created to model interference. The WMN has one Internet gateway which is also used as the Channel Assignment Server (CAS). A protocol is defined to collect estimated interferences on the channels resulting from co-located wireless activity and CAS uses this estimation to assign a default channel to the network. However, interference collection depends on successfully received packets from co-located wireless activity. Therefore, this interference data may not be accurate. Default channel is the least interfered channel and connectivity is established by this default channel. After default channel is assigned CAS uses neighbor information collected by the nodes to construct multi-radio conflict graph. This graph is used to calculate the distances of each router to the gateway. After distances to router are computed, CAS

uses BFS-CA (Breadth First Search - Channel Assignment) algorithm to assign channels to mesh radios.

[24] presents a distributed algorithm based on a localized greedy heuristic. In their model, each node is equipped with a 802.11a NIC and a 802.11g NIC. Initially, every node assigns a random channel to 802.11g NIC and a network-wide default channel to 802.11a NIC. Then each node updates its channel on 802.11g band in order to minimize its local objective function based on the channel assignments of neighbor nodes. In order to prevent nodes from updating their channels simultaneously, a three-way handshake protocol is used which guarantees the stabilization of the algorithm. After channel assignment is stabilized, they use MR-LQSR [17] protocol to find routes.

In [32] spanning sub-graphs which maintain connectivity with the original graph are found through linear programming. Then the edges which connect these sub-graphs are assigned a default channel to preserve connectivity. Other edges' channels are assigned randomly. This approach alters the network topology. Moreover, the channel assignment depends on randomization which may assign same channels to nearly located routers.

Chapter 4

Our Solution

In this chapter, we describe in detail our proposed channel assignment and routing algorithm for multi radio mesh networks. We will first give an overview of the algorithm, then we will describe our solution in detail.

4.1 System Model and Assumptions

In this section, our proposed multi-channel wireless mesh network architecture and our assumptions will be described. As shown in Figure 4.1, our proposed network consists of stationary wireless routers which form the backbone of the wireless mesh network. Some of the routers are connected to the Internet via wired links and some of them are connected to servers such as email server, FTP server, etc., which wireless clients need to access. All of the routers can act as wireless access points to wireless clients and provide network connectivity for them by relaying their traffic to other routers.

Each wireless router in the wireless mesh network is equipped with multiple 802.11 compatible network interface cards (NICs) each of which is statically bound to different non-overlapping frequency channels defined by 802.11 standards [20]. In order to communicate, two nodes need to be in the *communication range* of

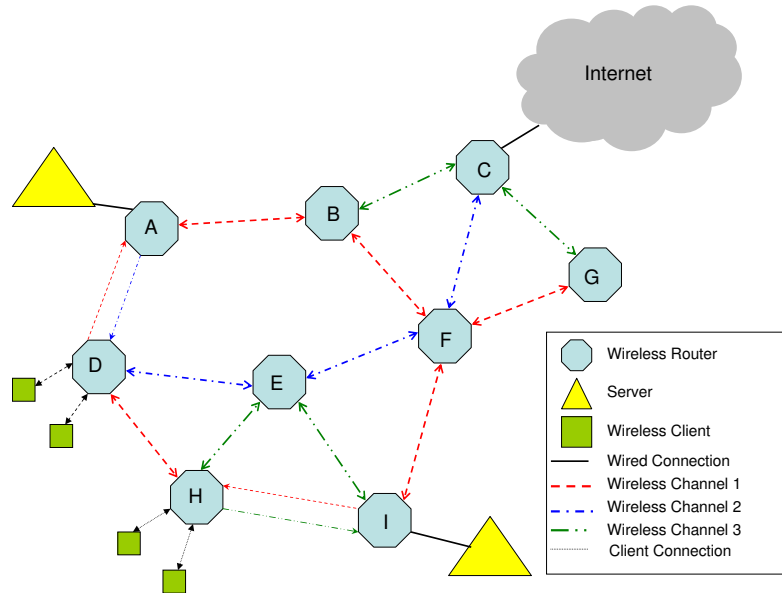


Figure 4.1: System architecture for proposed multi channel wireless mesh network.

each other and have at least one common channel assigned to their NICs. We assume that there exists virtually two *links* between such a pair (i.e. from A to B and from B to A) in the reverse directions. If pairs have two common channels (as between routers A and D in Figure 4.1), they can use different channels for these *links*. So different from previous approaches, pairs can communicate on two different channels for different directions which allows simultaneous two-way communication between these routers. If two nodes are within *interference range*, then their communications on the same channel interfere each other, while node pairs using different channels can communicate simultaneously even though they are close to each other.

We assume that the topology of the network is static and does not change frequently, and also expected traffic profile is known a priori. We also assume that, network supports peer to peer communication between clients even though they are connected to different routers in the same network. Network can be setup by setting assigned channels and constructing routing tables programmatically.

4.2 Algorithm Overview

IEEE 802.11 standards define 3 and 12 non-overlapping frequency channels for 802.11b/g and 802.11a respectively [20], which can be used simultaneously to increase the available bandwidth. However, each wireless router on traditional wireless mesh networks has one wireless NIC to communicate with their clients and other routers. Since two routers should be in the same channel to communicate, either all routers should use the same channel or a per-packet channel switching mechanism should be supported by the radios to communicate with other routers.

Another option is to introduce multiple NICs to each router, which allows routers to communicate simultaneously on different non-overlapping frequency channels. In that case, the available bandwidth on these channels can be used effectively to increase the throughput. However, a careless channel assignment may assign same channels to routers in the same area, sparing the available bandwidth of the unused channels. And also a careless routing may route flows such that some links become bottleneck, while other links rarely used for transmission. Therefore, in order to use the available bandwidth effectively, two problems must be solved:

- Which link gets which channels?
- Which paths flows are routed?

For two routers to communicate, the medium should be idle on both sides of the link, because of the RTS - CTS exchange. So, the available bandwidth is affected by the number of links in the interference range which are using the same channel.

The traffic flow on each link also affects the available bandwidth of the links using same channel in the interference range. So, the paths of the flows also affect the overall bandwidth.

Therefore, our channel assignment and routing algorithm solves these problems using the following 3 phases:

- Initial load estimation,
- Channel assignment for links,
- Routing.

In the first phase, we estimate initial loads on links and routers. For a flow between a pair of routers, we find some paths which have possibility to carry traffic and then increase the loads of links on that path by the expected amount of traffic divided by number of paths found.

After estimated loads on links and routers are found, we assign channels to links in some order considering the channels assigned to communicating routers (i.e. the ends of the links) and channels assigned to links in its interference range.

Finally we find paths between each pair of routers such that a flow between a pair is routed through the least busy path among the alternative paths. Therefore we avoid contention which results packet drops and delays.

4.3 Initial Load Estimation

In order to increase the total throughput, our channel assignment algorithm tries to assign the channel with maximum available bandwidth to each link. Therefore we need an estimation of available channel bandwidths for each link, before channel assignment. A naive approach is equally splitting the total bandwidth of the channel to all of the links using that channel in an interference region. However, that would not be accurate, because the traffic demands of the routers is not the same. Therefore, in order to find available channel bandwidths for each link, we need an estimation of traffic passing through that link.

We apply a probabilistic approach for estimating loads on links. Lets call all

min-cost paths between a source s and a destination d as $P(s, d)$. Then for each link l , l 's probability of carrying traffic between s and d , $p_l(s, d)$, is:

$$p_l(s, d) = \frac{|P_l(s, d)|}{|P(s, d)|} \quad (4.1)$$

where $P_l(s, d)$ is the set of min-cost paths containing link l . Lets call the expected traffic between s and d as $T(s, d)$, then estimated load on link l is:

$$ET_l = \sum_{s,d} p_l(s, d) * T(s, d) \quad (4.2)$$

which says that the estimated load on a link is sum of link's probability of carrying traffic multiplied by the expected amount of traffic between all router pairs. This estimation is not accurate also, since we assume multi-path routing and perfect load balancing between paths, however, it gives some idea about how the loads will be distributed in the network.

In order to calculate estimated loads of links, firstly we find all min-cost paths between each pair in the network. Then for each pair, we traverse the links in each min-cost path and add expected amount of traffic divided by number of total paths to link's estimated load.

Now let's describe the algorithms we apply for finding min-cost paths between each pair.

4.3.1 Finding all min-cost paths

In order to find all min-cost paths between all pairs, for each router we run a shortest paths algorithm similar to Dijkstra's shortest path algorithm [15], however, in our version all minimum cost paths between a pair is found. Execution order of the routers is not important because we use static information to find all shortest paths.

Algorithm starts with an initialization phase. In this phase all routers' costs are set to zero and an empty parent list is created for each router. After this loop, source router's cost is set to zero, and an empty heap is created and source router is added to this heap to complete initialization phase. Then the main loop starts. In the loop, the router with minimum cost is extracted from heap. Let's call this router r . Then all neighbors of r is processed. Sum of r 's cost and the link's cost is *proposedCost*. If *proposedCost* is equal to neighbor's cost, then r is added to neighbor's parents list. If *proposedCost* is less than neighbor's cost, then neighbor's parent list is cleared, r is added to the parent list, and neighbor's cost is set to *proposedCost*. Neighbor is also added to the heap, if it is not already in it. After updating of all neighbors is finished for r , path extraction phase starts. In this phase, r 's paths to the source node is found. For each parent of r p , we append the link between r and p to the paths from source to p and add resulting paths to paths of r . Main loop runs until heap becomes empty. The pseudocode is shown in Algorithm 4.1.

We have two different link cost functions which create two different methods for finding all min-cost paths between pairs:

4.3.1.1 Min-hop paths routing

In this method, each link's cost is 1 and we find traditional shortest paths between each pair.

$$\phi_l = 1 \tag{4.3}$$

4.3.1.2 Min-interfering paths routing

The success of load estimation depends on the accuracy of the initial routing. If the flows are routed in the initial routing such that they will never follow that path in the final configuration, then the estimation becomes false. So the more initial routing resembles to final routing, the better the load estimation.

```

for all  $r \in routers$  do
   $r.Key \leftarrow \infty$ 
   $r.Parents \leftarrow \emptyset$ 
end for
 $source.Key \leftarrow 0$ 
 $heap \leftarrow \emptyset$ 
 $heap.Insert(source)$ 
while  $heap$  is not empty do
   $r \leftarrow heap.GetMin()$ 
  for all  $nbr \in r.Neighbors$  do
     $proposedCost \leftarrow r.Cost + GetLink(r, nbr).Cost$ 
    if  $proposedCost = nbr.Cost$  then
       $nbr.Parents.Add(r)$ 
    else if  $proposedcost < nbr.Cost$  then
       $nbr.Parents.Clear()$ 
       $nbr.Parents.Add(r)$ 
       $nbr.Cost \leftarrow proposedCost$ 
    end if
  end for
   $r.Paths \leftarrow \emptyset$ 
  for all  $parent \in r.Parents$  do
     $link \leftarrow GetLink(r, parent)$ 
    for all  $path \in parent.Paths$  do
       $r.Paths.Add(path.Insert(link))$ 
    end for
  end for
end while

```

Algorithm 4.1: Finding all min-cost paths for a source.

In order to estimate the final routing, in which flows are routed to paths with less activity, we need to find paths such that the expected activity is less. Since a link's available bandwidth is affected by other links in the interference range, a link with less number of interfering links is expected to have more bandwidth.

Therefore, in this method, each link's cost is equal to the number of links in the interference range:

$$\phi_l = |intf(l)| \quad (4.4)$$

4.3.2 Distributing Estimated Traffic Among Paths

After all min-cost paths between each pair is found, the estimated traffic passing through each link and each router is calculated by sending portions of the flows through found paths.

For each router pair src and dst , the expected amount of $data$ from src to dst is gathered from traffic profile and added to src as outgoing data and dst as incoming data. Then $dataPerPath$ is calculated by dividing $data$ to the number of paths in $pathList$ from src to dst . Then for each $path$ in $pathList$, links and intermediate routers are traversed and $dataPerPath$ is added to their estimated traffic. Algorithm 4.2 shows the pseudocode of this process.

```

for all  $src \in routers$  do
  for  $i = 0$  to  $routers.Count$  do
     $dst \leftarrow routers[i]$ 
     $data \leftarrow GetExpectedDataForPair(src, dst)$ 
     $src.AddOutGoingData(data)$ 
     $dst.AddIncomingData(data)$ 
     $pathList \leftarrow src.GetPathsTo(dst)$ 
     $dataPerPath \leftarrow data/pathList.Count$ 
    for all  $path \in pathList$  do
      for all  $link \in path$  do
         $link.EstimatedTraffic \leftarrow link.EstimatedTraffic + dataPerPath$ 
      end for
      for all  $router \in path$  do
         $router.AddPassingData(dataPerPath)$ 
      end for
    end for
  end for
end for

```

Algorithm 4.2: Distributing traffic among initial paths.

4.4 Channel Assignments

After estimated loads for links and routers are found, our channel assignment selects links in some order and assigns a channel for each selected link. In this phase our goal is to maximize total available bandwidth by assigning channels to links such that the activity on that channel which is interfering the link is minimum.

The channel assignment algorithm is *NP-hard* [31]. Therefore we present 3 greedy algorithms to select a link to assign channel. For all 3 of the algorithms, we initially select a link to assign channel, and then we assign a channel to the link using dynamic interference information resulting from estimated traffic of previously assigned links.

In the next section we present our greedy link selection algorithms, and in Section 4.4.2 we describe how we assign a channel to the selected link.

4.4.1 Selection order of Links

Since our channel assignment algorithm is a greedy algorithm and we use dynamic interference data resulting from previously assigned links' estimated traffic as an input to assign a link's channel, the visiting order of links has impact on overall success of the algorithm. In order to determine which link to select, we use initial loads estimated in Section 4.3. We have 3 approaches for order of selecting links. First approach uses the loads of links and the remaining approaches use the loads of routers.

4.4.1.1 Max Link Selection (ML)

In this approach, we select links in the decreasing order of estimated traffic passing through the links. In other words, each time, we select the link with maximum expected traffic among the unprocessed links (i.e. Max Link). Similar approach

is also used in [31]. Since most active links are visited earlier, they are likely to be assigned less busy channels.

4.4.1.2 Max Router - Max Neighbor Selection (MR-MN)

In this approach, first we select routers in the decreasing order of total estimated traffic passing through their links (i.e. estimated traffic of router). Then for each router, we visit its links in the decreasing order of estimated traffic of the neighbor router. The advantage of this approach is, the channels of most active routers are assigned earlier, so other routers will more likely select other channels.

4.4.1.3 Max Router - Max Link Selection (MR-ML)

Similar to MR-MN, we first select the router among unprocessed routers which has the maximum estimated traffic. Then we visit its links in the decreasing order of estimated traffic passing through that link. This approach assigns channels of most active routers and their most active links earlier, so they are more likely to get less busy channels.

4.4.2 Assign Channel For a Link

Our channel assignment algorithm for a link is similar to the one described in LACA [31]. However, when there are no common channels between the incidents of a link, their algorithm suffers from ripple effects caused by altering the channels of previously assigned links. This alteration process may result infeasible assignments, since the altered links are forced to use a channel, that may have more interference than the others. Therefore, we simplified their model and assume that if the number of available network interfaces is q , then a maximum number of c channels will be used by the system, such that:

$$c \leq 2 * q - 1 \tag{4.5}$$

Like LACA, a channel is assigned to a link based on current channels of incident routers. There are 3 possible cases:

- Both *router1* and *router2* has less than q members in their channel list. Then any channel with the minimum *evaluation point* is assigned to the link.
- One of the routers, say *router1*, already has q members in its channel list and *router2* has less than q members in its channel list. Then the channel which has the minimum *evaluation point* in the *router1*'s channel list is assigned to the link.
- Both *router1* and *router2* has q members in their channel list. In this case they should have at least 1 common channel, since number of available channels is less than $2 * q$. Then the channel with the minimum *evaluation point* in their common channels list is assigned to the link.

Evaluation point is the sum of currently expected traffic (i.e. set by the algorithm up to now) on the links which use same channel and are in the interference range of this link. *Evaluation point* of a channel for given link is calculated by Algorithm 4.3. In this algorithm, the total amount of affected traffic on incidents of the link which is set by Algorithm 4.5 is calculated.

```

points ← 0
for all traffic ∈ router1.AffectedTraffic do
  if traffic.Channel = channel then
    points ← points + traffic.Amount
  end if
end for
for all traffic ∈ router2.AffectedTraffic do
  if traffic.Channel = channel then
    points ← points + traffic.Amount
  end if
end for
return points

```

Algorithm 4.3: Evaluate channel.

Initially, channel assignment algorithm gets the estimated traffic which is calculated in Section 4.3. Then for each channel ch , both incident routers of the link is checked whether ch can be used by this routers or not. If ch is eligible to be assigned to the link, then Algorithm 4.3 evaluates the channel and determines whether this channel is the one with minimum *evaluation points* or not. After processing all channels, Algorithm 4.5 sets the link's channel, adds the channel to the channel lists of both incidents of the link and adds the estimated traffic to *AffectedTraffic* lists of the routers which are affected by the communication on this link (i.e. routers in the interference range of this link).

```

traffic ← GetLink(router1, router2).GetEstimatedTraffic()
minPoints ← 0
minPointChannel ← -1
for all channel ch do
  if router1 does not have  $q$  channels and  $ch \in router1.Channels$  then
    if router2 does not have  $q$  channels and  $ch \in router2.Channels$  then
      points ← EvaluateChannel(router1, router2, ch)
      if minPoints < points then
        minPoints ← points
        minPointChannel ← ch
      end if
    end if
  end if
end for
SetChannelForLink(router1, router2, minPointChannel, traffic)

```

Algorithm 4.4: Assign channel for link.

4.5 Final Routing

After a channel is assigned to each link, the set of links which are interfering with this link (i.e. *interference set*) is also formed. In the final phase, we try to route each flow on the path which has less interference than the other paths. In order to find such paths, we use the links in the *interference set* and their load estimations from Section 4.3.

In this phase, we also run Algorithm 4.1 for each pair to find all min-cost


```

router1.SetChannel(router2, ch)
GetLink(router1, router2).Channel ← ch
if router1.Channels does not contain ch then
    router1.Channels.Add(ch)
end if
if router2.Channels does not contain ch then
    router2.Channels.Add(ch)
end if
for all r ∈ intf(router1) do
    r.AffectedTraffic.Add(traffic)
end for
for all r ∈ intf(router2) do
    r.AffectedTraffic.Add(traffic)
end for

```

Algorithm 4.5: Set channel for link

paths between this pair with the link cost:

$$\phi_l = ET_l + \sum_{i \in \text{intf}(l)} ET_i \quad (4.6)$$

After min-cost paths are found, we reset the interference data of routers and links. Then we calculate the costs of flows by Equation 4.7 and traverse the pairs in the decreasing order of their costs. For each pair, we select the path among the paths found by Algorithm 4.1, whose total current link interference is minimum. After path for a flow is selected, its interference is added to interference lists of interfering nodes and links.

$$\Phi_{s,d} = T(s, d) * \text{len}(P(s, d)) \quad (4.7)$$

4.6 Discussions

Although we have implemented a complete solution for channel assignment and routing in multi-radio multi-channel WMNs, some revisions may be considered to achieve better performance in real life conditions.

First of all, our algorithms can only utilize $2q - 1$ channels where q is the number of available radios for a router. Our purpose on such a limitation is to ensure that there is at least one common channel between two neighbor routers. Some new mechanisms can be embedded into our algorithm which allow using more channels while preserving the connectivity between neighbor routers. And also in some cases, having more channels may be more favorable than preserving connectivity between each pair of neighbors, where low delay is sacrificed to high throughput.

Secondly, our algorithm and to the best of our knowledge other algorithms dealing with load aware channel assignment in WMNs do not consider the interference introduced by the communication between wireless routers and their wireless clients. Considering these communications may lead to more effective channel assignments. And also, we do not consider the external wireless activity while computing the expected interferences on links.

Our algorithms are tested by a custom-designed program to find channel assignments in different topologies and different traffic profiles. We also verified that our algorithm can effectively use multiple radios to increase the bandwidth of the network using a simulator. The simulation scenarios and evaluation results are described in the next section.

Chapter 5

Evaluation

We implemented and tested our heuristics in a custom-designed program and also evaluated our algorithms using ns-2 simulator which is modified by Raniwala et al. [29] to support multi-radio multi-channel wireless mesh networks and evaluate LACA [31]. The implementation of the heuristics proved that our methods can find channel assignments and routing for different types of network topologies and traffic profiles. Furthermore, simulations on ns-2 simulator proved that the channel assignment and routing which our heuristics find has better throughput and delay compared to traditional single-radio single-channel wireless mesh networks and another alternative algorithm called LACA [31].

5.1 Simulation Scenarios

We evaluated our heuristics by measuring several metrics from the resulting network which have different channel assignments. Then, we compared our heuristics with following channel assignments:

- *Traditional single-radio single-channel channel assignment with min-hop shortest path routing.* To show the increase in throughput by introducing one NIC and using 3 channels.

- *A random channel assignment with min-hop shortest path routing.* To show the effect of a careless channel assignment
- *Load-Aware Channel Assignment (LACA) [31].* To compare our solution with a previous work which also uses traffic profiles.

We assumed that $R_i = 2R_c$ where R_i is the interference range and R_c is the communication range and we used a 5x5 uniform grid topology where width of a cell is R_c . Each node is located at the center of a cell, allowing a router to communicate with the routers in its neighbor cells and hear communications of routers which are 2 hop distant. We used 3 non-overlapping channels (as in 802.11b/g) in our simulations to show the effect of our algorithm on few number of available channels, and each router is equipped with 2 NICs.

As we noted before, we used a modified version of ns-2 [29], which supports multiple non-overlapping channels and multiple radios for each node. The default bandwidth of the simulator was 11 Mbps with a maximum packet size of 210 bytes. We send CBR flows between the pairs with the rate defined in the traffic profile file. The simulations lasted for 25 seconds and the following metrics are measured during this period:

- *Total number of successfully received packets,* shows the total throughput of the network in 25 secs.
- *Number of successfully received packets for each second,* shows the average throughput per second.
- *Average delay of all flows,* shows the average delay introduced by multi-hop transmissions.
- *Average standard deviation of delays of all flows,* shows the average jitter.
- *Percent of packets successfully received for each flow,* show the reliability of the paths.

We represented the traffic profile as a matrix, containing a number between 0 to 9 in each cell, which corresponds to the traffic rate coefficients. We calculated

the exact amount of traffic by multiplying these coefficients with actual data rates of 1 KBps, 2 Kbps etc. For small data rates, the amount of traffic for each traffic profile was so little that most of the packets reached to the destinations. Then we incremented data rates until the total number of dropped packets exceed 15% of the total number of sent packets for all of the algorithms. We analyzed the metrics and displayed the results for last three data rates.

We call *optimal data rate* to the maximum data rate which results less than 15% of packet drops for some of the algorithms. For this *optimal* cases, we analyzed different parameters of the networks and displayed these results also.

Unlike our algorithms (which takes coefficients as traffic profile.), LACA depends on the exact rate of traffic in Kbps, so for each data rate we ran LACA to get channel assignment for that amount of traffic. And also, since there is some randomness in LACA, we ran LACA with 4 different seeds to get 4 different channel assignments for each scenario and traffic profile. When comparing LACA to our results, we get the best of these 4 results.

5.1.1 WMN with one gateway to Internet Access

Table 5.1: Simulation Results for WMN with 1 gateway

Min. Data Rate	5 Kbps	6 Kbps	7 Kbps
Total # of sent packets	18920	22706	26490
Single Channel	6004	5711	5362
Random Assignment	11367	14309	14933
LACA	17877	18614	19340
ML	17013	18001	19920
MR-MN	18824	18780	19277
MR-ML	17967	19503	19993
% of Max Packets Received	99.5	85.9	75.4

In this scenario, the network has an Internet gateway which is located at the center of the 5x5 grid. The requests of routers are modeled as flows with a coefficient of 2 and the reply of the Internet gateway is modeled as flows with a coefficient of 9 (i.e. the ratio of request packets to reply packets is 2/9). Table 5.1

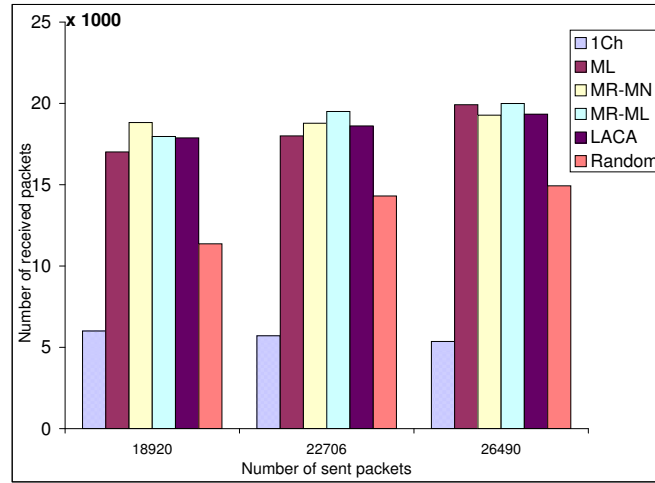


Figure 5.1: WMN with one gateway. Total number of received packets.

shows the total number of packets sent at the end of the simulations for the last three data rates.

Figure 5.1 shows the total number of received packets for last three data rates. In the *optimal* case where the minimum data rate is 6 Kbps, the gain by using 3 channels and 2 NICs is nearly 3 times. And also, we see all of the *intelligent* channel assignment schemes using 2 NICs, perform similar results for this scenario.

We further investigated the *optimal* case to find the reliability of the paths. Figure 5.2 also shows that the reliabilities of the paths are almost similar. There are a few pairs whose packets are successfully delivered for most of the algorithms, and nearly half of the flows delivered more than 90% of their packets in MR-MN, MR-ML and LACA. Also, ML algorithm has more flows which successfully delivered all of their packets, however, ML has also more flows which delivered only around 20% of their packets.

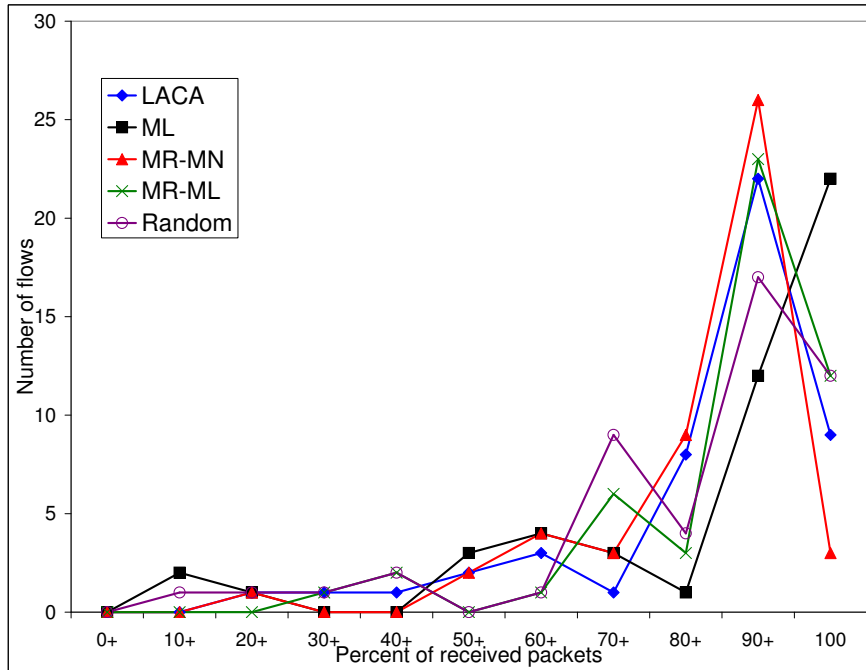


Figure 5.2: WMN with one gateway. Reliability of the paths.

5.1.2 WMN with 5 servers

In this scenario, there are 5 servers in the WMN which may serve as Internet gateway, Intranet server, email server, FTP server, etc. The servers are located in the network such that there is 1 server in each row in the grid. Similar to Section 5.1.1, each node sends some data to each server which represents request and get some data from each server which represents requested data. Table 5.2 shows the total number of packets sent and received by compared algorithms

Table 5.2: Simulation Results for WMN with 5 servers

Min. Data Rate	1 Kbps	2 Kbps	3 Kbps
Total # of sent packets	14060	28037	42018
Single Channel	5781	5419	6606
Random Assignment	11996	14854	16932
LACA	14031	19847	20644
ML	14011	24255	24986
MR-MN	14031	23310	23832
MR-ML	14013	22881	24533
% of Max Packets Received	99.8	86.5	59.5

for different minimum data rates. And also Figure 5.3 and Figure 5.4 show the results of the simulations.

From Table 5.2 and Figure 5.3 (a) we can see that in the *optimal* case where minimum data rate is 2 Kbps, our assignments can successfully deliver more than 4 times more packets than single-radio single-channel assignment. And also we can see that ML performs more than 20% better than LACA and more than 60% than random assignment. Figures (b) and (c) also show that the average delay and average standard deviation of all flows in ML assignment is minimum which makes our assignment algorithm more suitable for applications with delay and jitter constraints like multimedia streams.

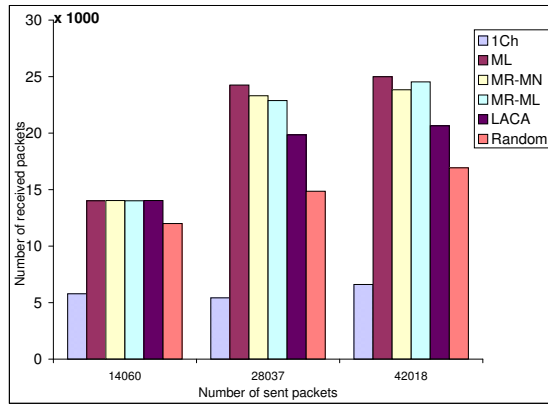
Figure 5.4 (a) shows the total throughput per second. While ML reaches nearly 1.7 Mbps, LACA has 1.3 Mbps, random assignment has 1 Mbps and single channel WMN has 400 Kbps average throughput. Figure (b) shows that more than 70 out of 220 flows delivered packets without a loss in the assignment by ML, while nearly 40 flows are lossless in the random assignment and LACA.

5.1.3 WMNs with random peer to peer traffic profiles

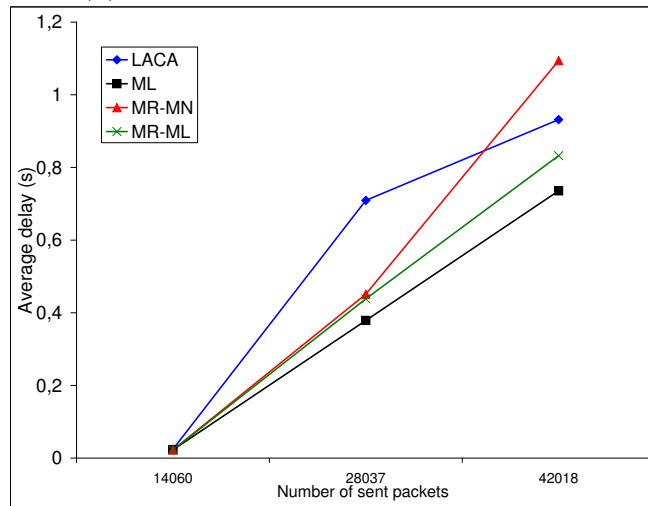
In these scenarios, 3 different random traffic profiles which contain peer to peer communications instead of client - server communications are examined.

5.1.3.1 Random Traffic Profile #1

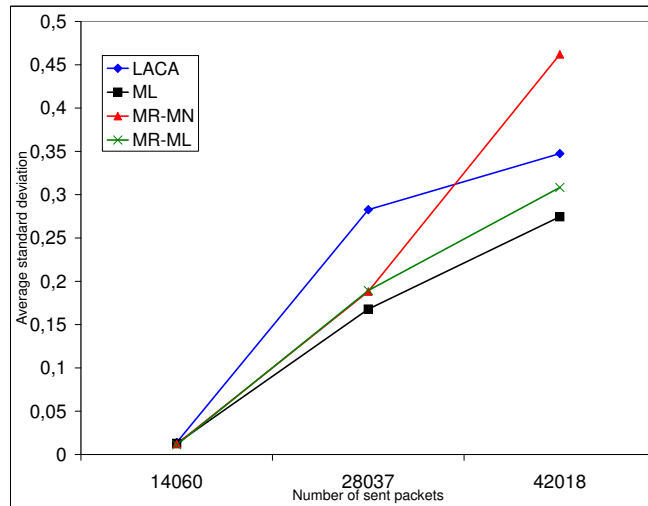
Table 5.3 and Figure 5.5 shows the results of 3 simulations run for 1 Kbps, 2Kbps and 3Kbps minimum data rates for random traffic profile 1. In the *optimal* case, where min. data rate is 2 Kbps, Figure (a) shows that ML algorithm delivers nearly all of the packets, which is nearly 5 times more than the single channel case. Also ML algorithm performs more than 30% better than LACA, and nearly 60% better than random assignment. Figures (b) and (c) also show that the average delay and average standard deviation of all flows in ML and MR-MN assignments are minimum.



(a) Total number of received packets.

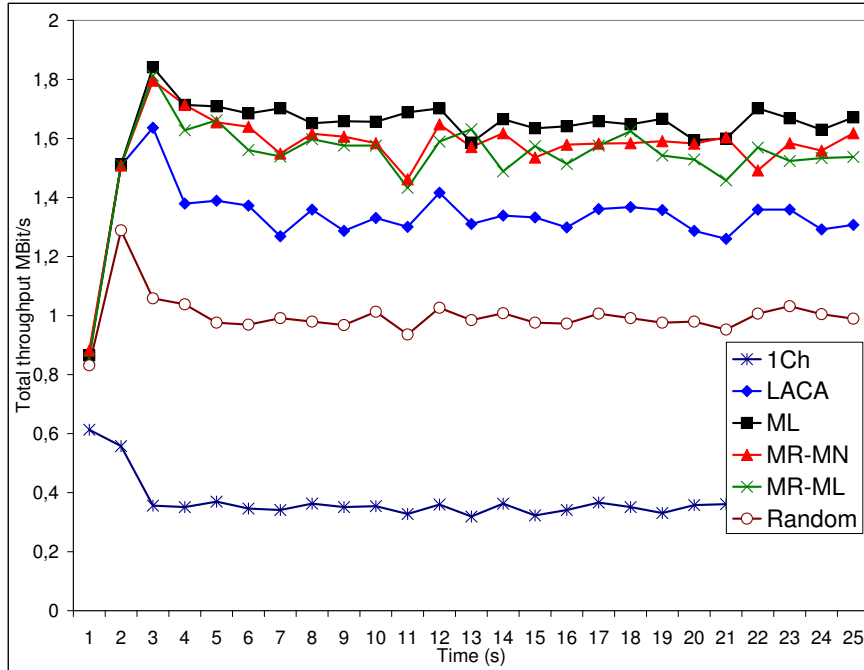


(b) Average delay for all flows

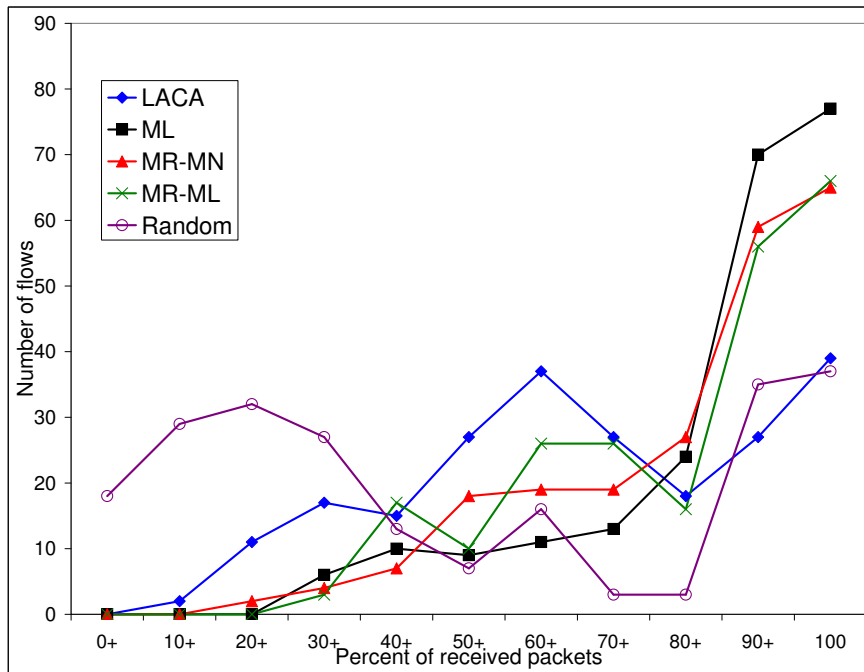


(c) Average of standard deviations of all flows

Figure 5.3: WMN with 5 servers.

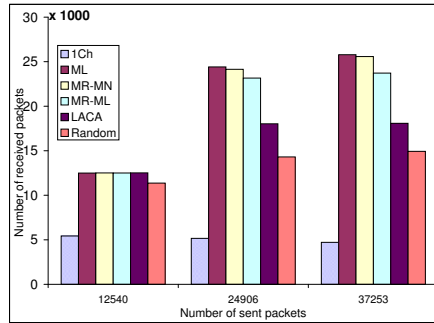


(a) Throughput per second.

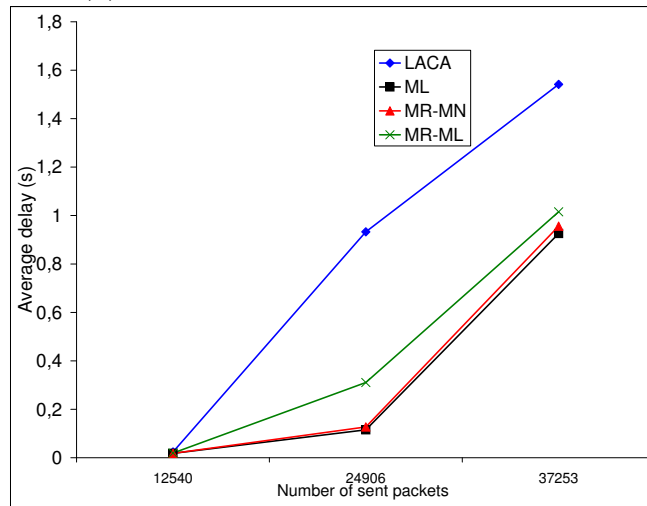


(b) Reliability of the paths.

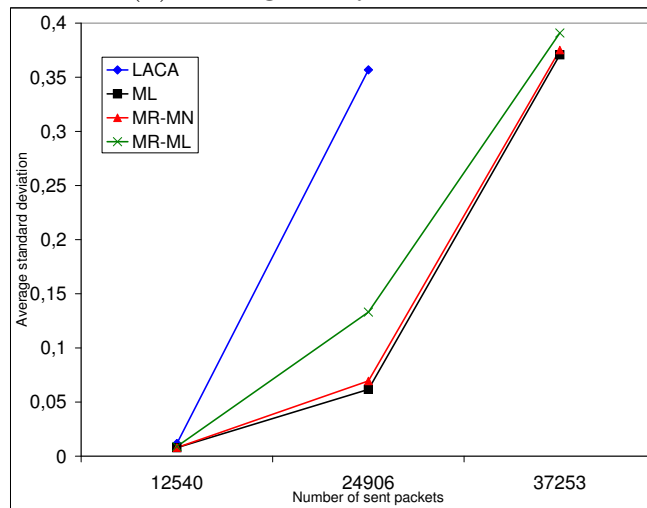
Figure 5.4: WMN with 5 servers.



(a) Total number of received packets.

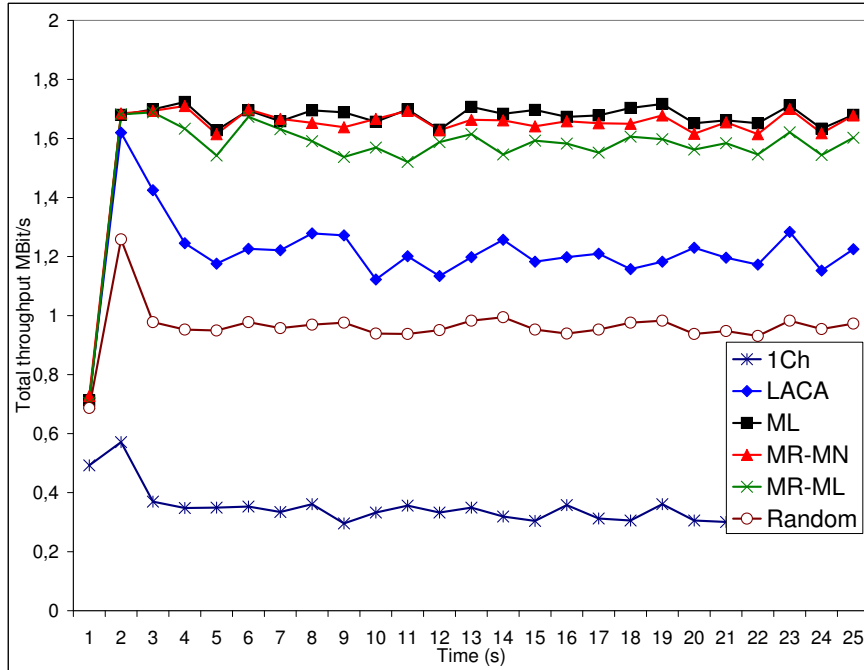


(b) Average delay for all flows

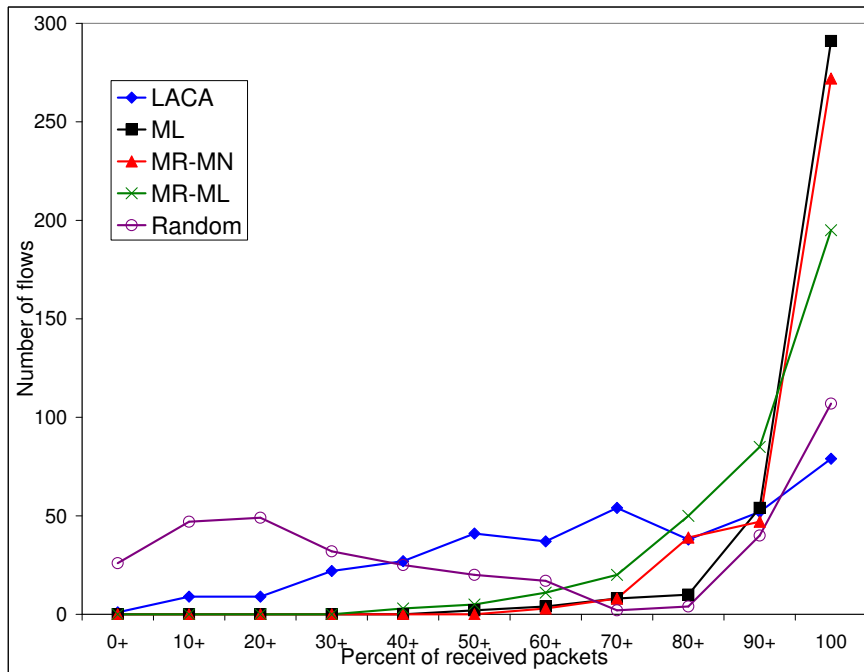


(c) Average of standard deviations of all flows

Figure 5.5: WMN with random traffic profile 1.



(a) Throughput per second.



(b) Reliability of the paths.

Figure 5.6: WMN with random traffic profile 1.

Table 5.3: Simulation Results for Random Traffic Profile #1

Min. Data Rate	1 Kbps	2 Kbps	3 Kbps
Total # of sent packets	12540	24906	37253
Single Channel	5430	5160	4715
Random Assignment	11367	14309	14933
LACA	12518	18026	18083
ML	12495	24412	25782
MR-MN	12509	24143	25576
MR-ML	12502	23161	23720
% of Max Packets Received	99.8	98	69.2

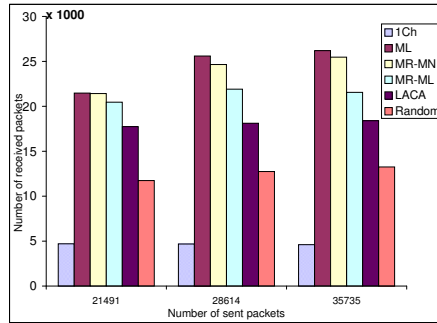
Figure 5.6 (a) shows the total throughput per second, where our algorithms deliver has nearly 1.7 Mbps throughput, while LACA has around 1.3 Mbps, random assignment has 1 Mbps and single channel 400 Kbps throughput. Figure (b) shows that 290 out of 370 flows delivered packets without a loss in the assignment by ML, while nearly 80 and 100 flows are lossless in the LACA and random assignment respectively.

5.1.3.2 Random Traffic Profile #2

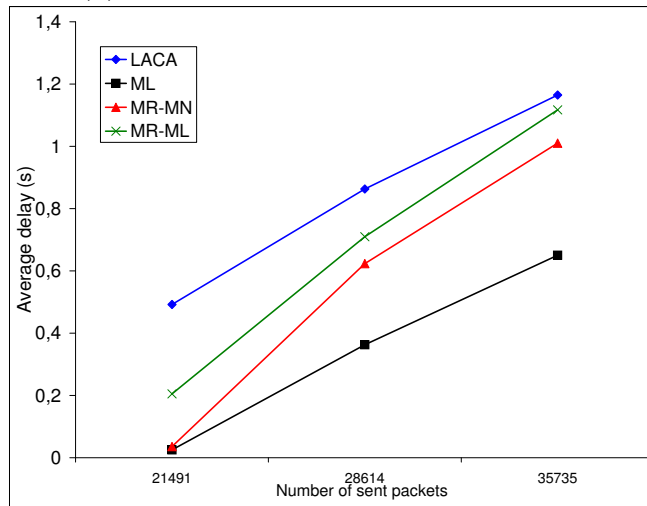
Table 5.4: Simulation Results for Random Traffic Profile #2

Min. Data Rate	3 Kbps	4 Kbps	5 Kbps
Total # of sent packets	21491	28614	35735
Single Channel	4700	4680	4605
Random Assignment	11734	12743	13251
LACA	17735	18112	18399
ML	21472	25597	26191
MR-MN	21416	24649	25466
MR-ML	20459	21910	21555
% of Max Packets Received	99.9	89.4	73.3

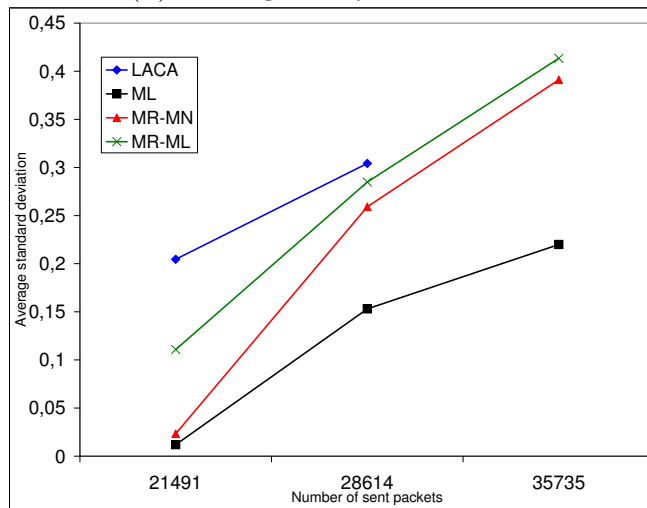
Table 5.4 and Figure 5.7 shows the results of 3 simulations run for 3 Kbps, 4 Kbps and 5 Kbps minimum data rates for random traffic profile 2. In the *optimal* case, where minimum data rate is 4 Kbps, Figure (a) shows that ML algorithm delivers more than 5 times the number of packets single channel network delivered. Also ML algorithm performs nearly 30% better than LACA and 2 times better than random assignment case. Figures (b) and (c) also show that



(a) Total number of received packets.

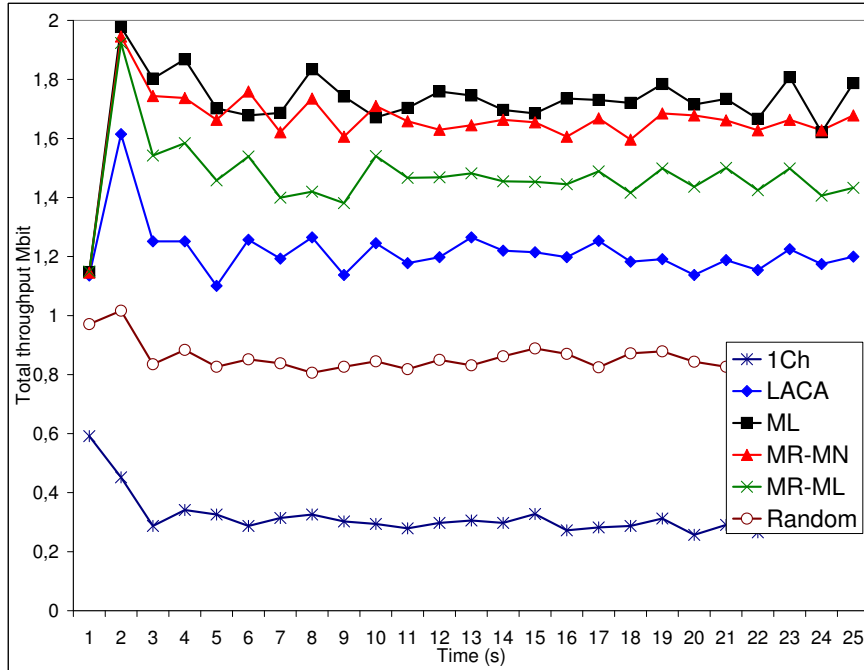


(b) Average delay for all flows

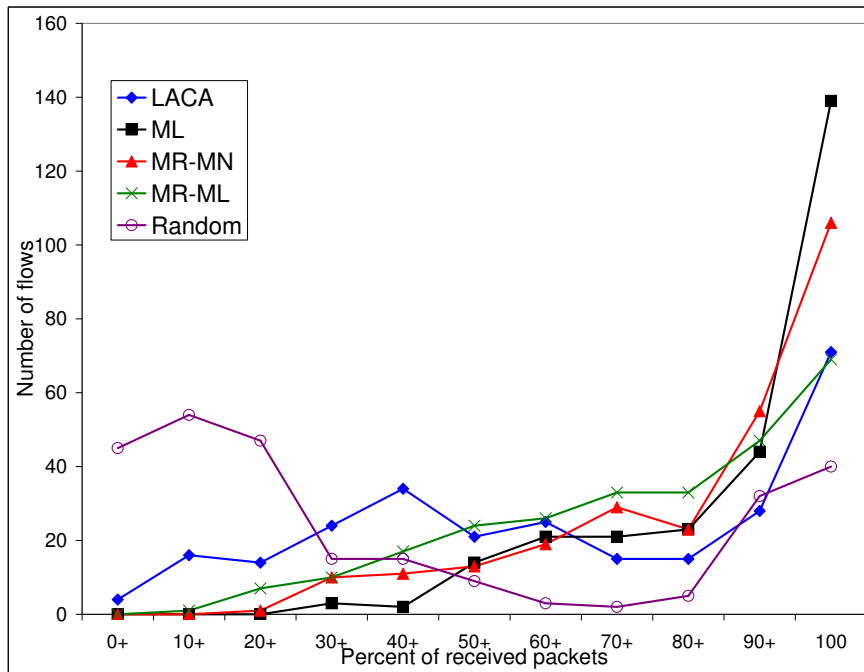


(c) Average of standard deviations of all flows

Figure 5.7: WMN with random traffic profile 2.



(a) Throughput per second.



(b) Reliability of the paths.

Figure 5.8: WMN with random traffic profile 2.

the average delay and average standard deviation of all flows in ML assignment is also minimum.

Figure 5.7 (a) shows the throughput per second, where ML and MR-MN has nearly 1.7 Mbps, while LACA has nearly 1.3 Mbps, random assignment has 0.8 Mbps and single channel network has 350 Kbps throughput. Figure (b) shows that 140 out of 270 flows delivered packets without a loss in the assignment by ML, while nearly 70 and 40 flows are lossless in the LACA and random assignment respectively.

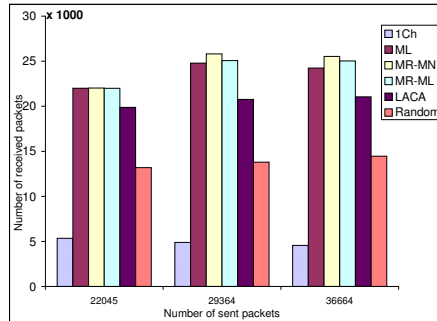
5.1.3.3 Random Traffic Profile #3

Table 5.5: Simulation Results for Random Traffic Profile #3

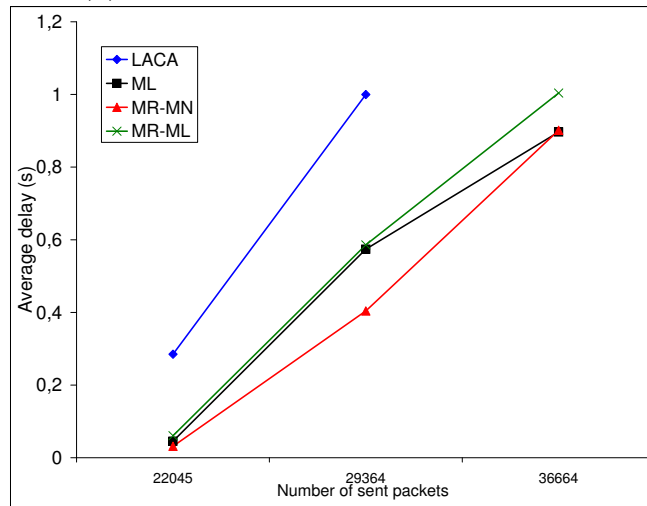
Min. Data Rate	3 Kbps	4 Kbps	5 Kbps
Total # of sent packets	22045	29364	36664
Single Channel	5344	4890	4553
Random Assignment	13191	13794	14457
LACA	19880	20759	21049
ML	21997	24786	24242
MR-MN	22019	25812	25532
MR-ML	21991	25067	25028
% of Max Packets Received	99.9	87.9	69.6

Table 5.5 and Figure 5.9 shows the results of 3 simulations run for 3 Kbps, 4 Kbps and 5 Kbps minimum data rates for random traffic profile 3. In the *optimal* case where data rate is 4 Kbps, Figure (a) shows that MR-MN algorithm delivers more than 5 times the number of packets single channel network delivered. Also MR-MN algorithm performs nearly 20% better than LACA and 70% better than random assignment case. Figures (a) and (b) also show that the average delay and average standard deviation of all flows in MR-MN assignment is minimum.

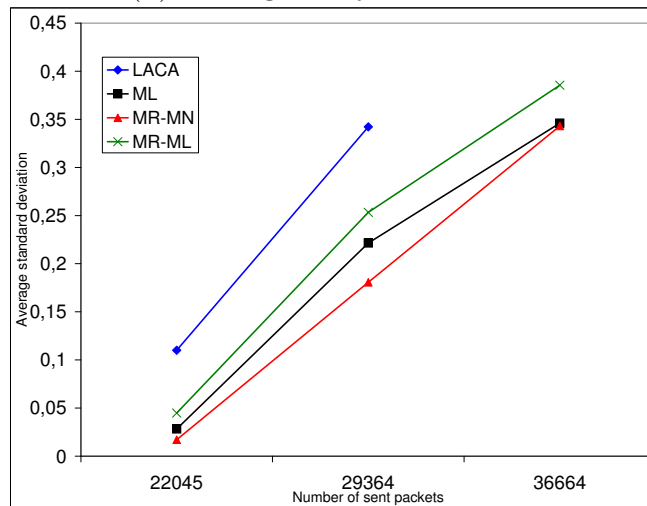
Figure 5.10 (a) also shows the throughput per second, where all of our algorithms has nearly 1.7 Mbps, while LACA has nearly 1.4 Mbps, random assignment delivers has 0.9 Mbps and single channel network delivers has 0.4 Kbps throughput. Figure (b) shows that 140 out of 270 flows delivered packets without a loss



(a) Total number of received packets.

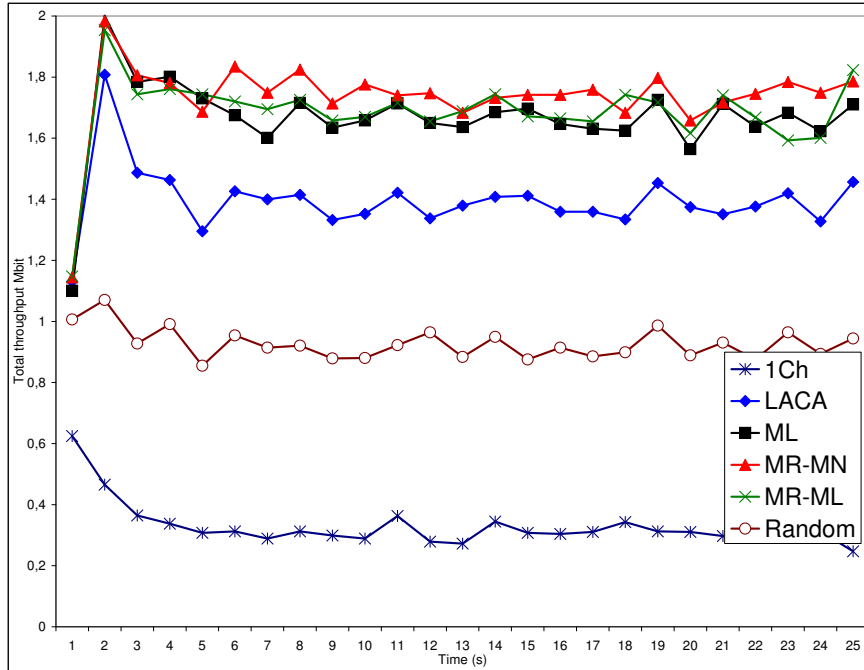


(b) Average delay for all flows

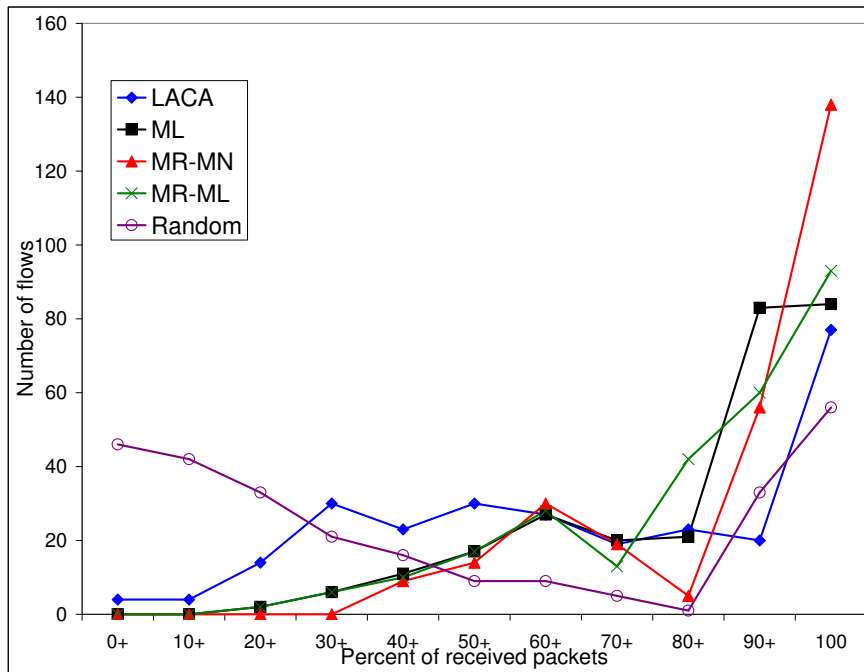


(c) Average of standard deviations of all flows

Figure 5.9: WMN with random traffic profile 2.



(a) Throughput per second.



(b) Reliability of the paths.

Figure 5.10: WMN with random traffic profile 3.

in the assignment by MR-MN, while nearly 80 and 60 flows are lossless in the LACA and random assignment respectively.

5.2 Effect of Routing Algorithms

Instead of our proposed routing algorithms for initial load estimation (IR) and final routing (FR) defined in Section 4, we used shortest path routing (SP) to observe the effects of routing algorithm on our channel assignment heuristics. For this purpose, the *optimal* cases of scenarios in Section 5.1 are reconstructed with following algorithm combinations:

- Initial Routing, Max Link Selection, Final Routing (IR-ML-FR)
- Shortest Path Routing, Max Link Selection , Final Routing (SP-ML-FR)
- Shortest Path Routing, Max Link Selection, Shortest Path Routing (SP-ML-SP)
- Initial Routing, Max Link Selection, Shortest Path Routing (IR-ML-SP)

Figure 5.11 shows the results of these simulations. As the graph shows, while our combination has the best average throughput for all cases, the combination which uses shortest path routing for initial estimation and our algorithm as the final routing has better average throughput than other modified combinations. Therefore, we can claim that final routing is very important to effectively use the available bandwidth. Even though the estimation is not accurate, final routing adjusts paths to effectively use the available bandwidth. We can also see that, the combination with shortest path routing for both initial estimation and final routing performs better than the one with shortest path routing for the final routing only. As we claim before, the more the initial routing resembles the final routing, the better the channel assignments are, because in that case the load estimation will use the actual routes and estimations become real loads.

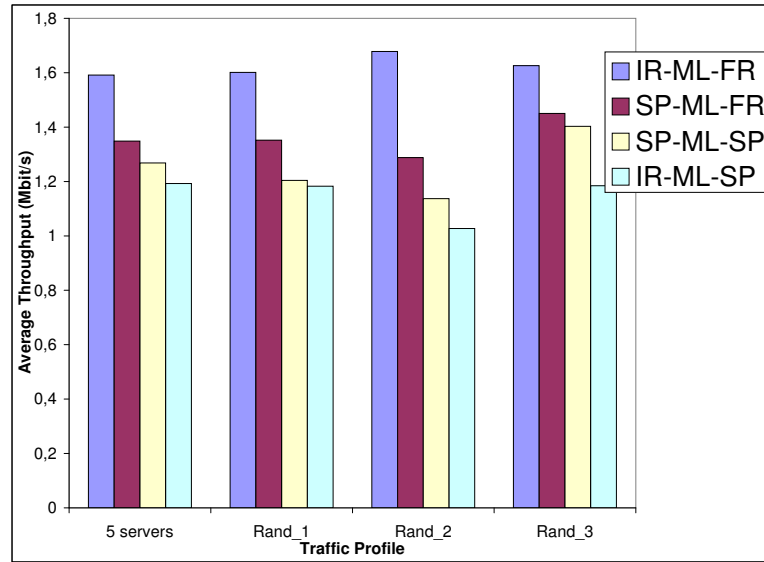


Figure 5.11: Effect of Routing Algorithms.

5.3 Discussion on Comparison Results

The simulations show that, our algorithms out-perform the random channel assignment and LACA [31] in the optimal cases of all of the scenarios. And also our algorithms help multi-radio multi-channel WMN successfully deliver up to 5 times more packets than single-radio single-channel WMNs.

Furthermore, in most of the cases, ML (4.4.1.1) method perform better than rest of the methods. And also, in some cases MR-MN (4.4.1.2) method perform better than other methods. However, MR-ML (4.4.1.3) method failed to achieve better performance results than ML or MR-MN, although it performed better than LACA [31] and random channel assignment algorithm in multi-radio multi-channel environment. Therefore we can claim that our channel assignment and routing algorithm with ML method can be used to as a channel assignment and routing solution in multi-radio multi-channel WMNs, if the traffic profile is known.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

In this thesis, we propose a channel assignment and routing algorithm for multi-radio multi-channel wireless mesh networks, which aims to increase the total throughput of the network while decreasing the average delay of the flows. The proposed solution tries to use the available bandwidth resulting from the non-overlapping frequency channels defined by IEEE 802.11 standards [20] by assigning less busy channel among the available channels to the links between neighbor routers. The routing algorithm, tries to find paths which have less end to end delays by choosing the paths with less traffic and interference.

Given the network topology and traffic profile between the nodes, our channel assignment algorithm tries to estimate the loads on links and routers by guessing the final routing of the flows. After link and router loads are estimated, three different link selection mechanisms, ML (Max Loaded Link), MR-MN (Max Loaded Router's Max Loaded Neighbor) and MR-ML (Max Loaded Router's Max Loaded Link) select the order of link assignments, and the channel with minimum activity is assigned to the channel considering the assignments of previously processed links. Finally our routing algorithm finds paths between routers such that the load and interference on the path is minimum among the alternatives.

We implemented our algorithms and proved that they can successfully assign channels to all links and routers with given channel and NIC constraints, and find paths between all pairs. Then we evaluated our channel assignment algorithms in ns-2 simulator, which supports multiple radios per node and multiple non-overlapping channels [29], running extensive simulations to compare our algorithms with a classic single radio single channel WMNs and random channel assignments and LACA [31] for multi-radio multi-channel WMNs.

Simulation results show that in a mesh network with 1 Internet gateway, all of the algorithms using multiple radios perform similar results, although our algorithms have better results. However, in networks with multiple source-destination pairs, our algorithms perform better than LACA and random channel assignment algorithms with respect to total throughput, number of reliable paths, average end-to-end delays and delay variations. Therefore, our algorithms can be used as the channel assignment and routing algorithm in wireless mesh networks.

6.2 Future Work

Our channel assignment algorithms can only utilize $2xq - 1$ channels where q is the number of available radios to preserve connectivity between all neighbors. A new mechanism can be embedded to our algorithms which can use more channels while preserving the connectivity between routers.

To the best of our knowledge, most of the channel assignment algorithms for multi-radio multi-channel WMNs do not consider the traffic between mesh routers and mesh clients. However, if the mesh router uses same NIC for both communication with its clients and other mesh routers, then the load of its NICs and the interference resulting from this communication should also be considered for load aware channel assignments and routing.

Since our proposed algorithm is a centralized off-line algorithm, it needs a network setup for assigning channels of the NICs, and constructing routing tables. And also, current distributed approaches either use a default channel [24]

or modify the network topology [30] to use multiple channels. A new distributed approach can be proposed which collects dynamic traffic and interference information and assigns channels to links without using a default channel or modifying the existing topology.

Bibliography

- [1] Airties wireless networks. <http://www.airties.com>.
- [2] Mesh dynamics. <http://www.meshdynamics.com>.
- [3] Motorola mesh networks. <http://www.meshnetworks.com>.
- [4] Muni wifi mesh networks. <http://www.belairnetworks.com>.
- [5] Radiant networks. <http://www.radiantnetworks.com>.
- [6] A. Adya, P. Bahl, J. Padhye, A. Wolman, and L. Zhou. A multi-radio unification protocol for ieee 802.11 wireless networks. In *BROADNETS '04: Proceedings of the First International Conference on Broadband Networks*, pages 344–354, Washington, DC, USA, 2004. IEEE Computer Society.
- [7] S. Agarwal, A. Ahuja, J. Singh, and R. Shorey. Route-lifetime assessment based routing (RABR) protocol for mobile ad-hoc networks. *Communications, 2000. ICC 2000. 2000 IEEE International Conference on*, 3:1697–1701, 2000.
- [8] I. F. Akyildiz, X. Wang, and W. Wang. Wireless mesh networks: a survey. *Computer Networks*, 47(4):445–487, March 2005.
- [9] M. Alicherry, R. Bhatia, and L. E. Li. Joint channel assignment and routing for throughput optimization in multi-radio wireless mesh networks. In *Mobi-Com '05: Proceedings of the 11th annual international conference on Mobile computing and networking*, pages 58–72, New York, NY, USA, 2005. ACM.

- [10] S. Avallone and I. F. Akyildiz. A channel assignment algorithm for multi-radio wireless mesh networks. *Comput. Commun.*, 31(7):1343–1353, 2008.
- [11] P. Bahl, R. Chandra, and J. Dunagan. Ssch: slotted seeded channel hopping for capacity improvement in ieee 802.11 ad-hoc wireless networks. In *MobiCom '04: Proceedings of the 10th annual international conference on Mobile computing and networking*, pages 216–230, New York, NY, USA, 2004. ACM.
- [12] C. Chang and S. Tu. Active route-maintenance protocol for signal-based communication path in ad hoc networks. *Journal of Network and Computer Applications*, 25(3):161–177, 2002.
- [13] D. S. J. D. Couto, D. Aguayo, J. Bicket, and R. Morris. A high-throughput path metric for multi-hop wireless routing. In *MobiCom '03: Proceedings of the 9th annual international conference on Mobile computing and networking*, pages 134–146, New York, NY, USA, 2003. ACM.
- [14] J. Crichigno, M.-Y. Wu, and W. Shu. Protocols and architectures for channel assignment in wireless mesh networks. *Ad Hoc Netw.*, 6(7):1051–1077, 2008.
- [15] E. W. Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1:269–271, 1959.
- [16] R. Draves, J. Padhye, and B. Zill. Comparison of routing metrics for static multi-hop wireless networks. In *SIGCOMM '04: Proceedings of the 2004 conference on Applications, technologies, architectures, and protocols for computer communications*, pages 133–144, New York, NY, USA, 2004. ACM.
- [17] R. Draves, J. Padhye, and B. Zill. Routing in multi-radio, multi-hop wireless mesh networks. In *MobiCom '04: Proceedings of the 10th annual international conference on Mobile computing and networking*, pages 114–128, New York, NY, USA, 2004. ACM.
- [18] M. Gerharz, C. de Waal, M. Frank, and P. Martini. Link stability in mobile wireless ad hoc networks. *Local Computer Networks, 2002. Proceedings. LCN 2002. 27th Annual IEEE Conference on*, pages 30–39, 2002.

- [19] P. Gupta and P. R. Kumar. The capacity of wireless networks. *IEEE Transactions on Information Theory*, IT-46(2):388–404, March 2000.
- [20] IEEE Std. 802.11. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications.*, 1999. Available at <http://standards.ieee.org/getieee802/802.11.html>.
- [21] C.-A. Y. Jenhui Chen, Shiann-Tsong Sheu. A new multichannel access protocol for ieee 802.11 ad hoc wireless lans. In *Personal, Indoor and Mobile Radio Communications, 2003. PIMRC 2003*, pages 2291–2296, 2003.
- [22] D. Johnson, D. Maltz, and J. Broch. DSR: The Dynamic Source Routing Protocol for Multi-Hop Wireless Ad Hoc Networks. *Ad Hoc Networking*, pages 139–172, 2001.
- [23] D. Kim, J. Garcia-Luna-Aceves, K. Obraczka, J. Cano, and P. Manzoni. Routing Mechanisms for Mobile Ad Hoc Networks Based on the Energy Drain Rate. *IEEE Transactions on Mobile Computing*, 2(2):161–173, 2003.
- [24] B.-J. Ko, V. Misra, J. Padhye, and D. Rubenstein. Distributed channel assignment in multi-radio 802.11 mesh networks. pages 3978–3983, 2007.
- [25] P. Kyasanur and N. H. Vaidya. Routing and interface assignment in multi-channel multi-interface wireless networks. volume 4, pages 2051–2056 Vol. 4, 2005.
- [26] Maxim. 2.4ghz 802.11b zero-if transceivers. Available at <http://pdfserv.maxim-ic.com/en/ds/MAX2820-MAX2821.pdf>, 2005.
- [27] C. Perkins and E. Royer. Ad-hoc on-demand distance vector routing. *Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications*, 2, 1999.
- [28] K. N. Ramachandran, E. M. Belding, K. C. Almeroth, and M. M. Budhikot. Interference-aware channel assignment in multi-radio wireless mesh networks. *infocom*.

- [29] A. Raniwala. Modified ns-2 simulator which supports multi-radio and multi-channels. Available at <http://www.ecsl.cs.sunysb.edu/multichannel/code/ns-2.1b9a-hyacinth.tgz>.
- [30] A. Raniwala and T. cker Chiueh. Architecture and algorithms for an IEEE 802.11-based multi-channel wireless mesh network. In *INFOCOM*, pages 2223–2234. IEEE, 2005.
- [31] A. Raniwala, K. Gopalan, and T. cker Chiueh. Centralized channel assignment and routing algorithms for multi-channel wireless mesh networks. *SIGMOBILE Mob. Comput. Commun. Rev.*, 8(2):50–65, 2004.
- [32] M. Shin, S. Lee, and Y.-A. Kim. Distributed channel assignment for multi-radio wireless networks. pages 417–426, 2006.
- [33] J. So and N. H. Vaidya. Multi-channel mac for ad hoc networks: handling multi-channel hidden terminals using a single transceiver. In *MobiHoc '04: Proceedings of the 5th ACM international symposium on Mobile ad hoc networking and computing*, pages 222–233, New York, NY, USA, 2004. ACM.
- [34] J. Tang, G. Xue, and W. Zhang. Interference-aware topology control and qos routing in multi-channel wireless mesh networks. In *MobiHoc '05: Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing*, pages 68–77, New York, NY, USA, 2005. ACM.
- [35] C. Toh. Maximum battery life routing to support ubiquitous mobile computing in wireless ad hoc networks. *Communications Magazine, IEEE*, 39(6):138–147, 2001.
- [36] T.-J. Tsai and J.-W. Chen. Ieee 802.11 mac protocol over wireless mesh networks: Problems and perspectives. In *AINA '05: Proceedings of the 19th International Conference on Advanced Information Networking and Applications*, pages 60–63, Washington, DC, USA, 2005. IEEE Computer Society.
- [37] A. Tzamaloukas and J. J. G. luna aceves. Channel-hopping multiple access. In *Proc. IEEE ICC 2000*, pages 415–419, 2000.

- [38] A. Tzamaloukas and J. J. G. luna aceves. Channel-hopping multiple access with packet trains for ad hoc networks. In *In In Proc. IEEE Mobile Multimedia Communications (MoMuC 00, 2000*.
- [39] S.-L. Wu, C.-Y. Lin, Y.-C. Tseng, and J.-P. Sheu. A new multi-channel mac protocol with on-demand channel assignment for multi-hop mobile ad hoc networks. In *ISpan '00: Proceedings of the 2000 International Symposium on Parallel Architectures, Algorithms and Networks (ISpan '00)*, page 232, Washington, DC, USA, 2000. IEEE Computer Society.
- [40] S. Xu and T. Saadawi. Does the ieee 802.11 mac protocol work well in multi-hop wireless ad hoc networks? *Communications Magazine, IEEE*, 39(6):130–137, 2001.

Appendix A

Table of Acronyms

CBR	Constant Bit Rate
IEEE	Institute of Electrical and Electronics Engineers
IP	Internet Protocol
LAN	Local Area Network
Kbps	Kilobits per second
MAC	Medium Access Control
Mbps	Megabits per second
MHz	Megahertz
NS	Network Simulator
TCL	Tool Command Language
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
WLAN	Wireless Local Area Network
WMN	Wireless Mesh Network
NIC	Network Interface Card
LACA	Load Aware Channel Assignment
MR-MC	Multi Radio Multi Channel
SR-MC	Single Radio Multi Channel
ML	Max Link Selection Heuristics
MR-MN	Max Router - Max Neighbor Selection Heuristics
MR-ML	Max Router - Max Link Selection Heuristics
FTP	File Transfer Protocol