Improving broadcast performance in multi-radio multi-channel multi-rate wireless mesh networks.

Author:
Qadir, Junaid

Publication Date:
2008

DOI:
https://doi.org/10.26190/unsworks/17710

Improving broadcast performance in multi-radio multi-channel multi-rate wireless mesh networks

by

Junaid Qadir

A thesis submitted in fulfillment of the requirements
for the degree of Doctor of Philosophy.

School of Computer Science and Engineering,
University of New South Wales.

May 11, 2008

This thesis entitled:

"*Improving broadcast performance in multi-radio multi-channel multi-rate wireless mesh networks*"

written by Junaid Qadir

has been approved for the School of Computer Science and Engineering

Supervisor: Assoc. Prof. Dr. Chun Tung Chou

Signed ....

Date ....

The final copy of this thesis has been examined by the signatory, and I find that both the content and the form meet acceptable presentation standards of scholarly work in the above mentioned discipline.

Declaration

I hereby declare that this submission is my own work and to the best of my knowledge it contains no materials previously published or written by another person, nor material which to a substantial extent has been accepted for the award of any other degree or diploma at UNSW or any other educational institution, except where due acknowledgement is made in the thesis. Any contribution made to the research by others, with whom I have worked at UNSW or elsewhere, is explicitly acknowledged in the thesis.

I also declare that the intellectual content of this thesis is the product of my own work, except to the extent that assistance from others in the project's design and conception or in style, presentation and linguistic expression is acknowledged.

Signed ....

Date ....

Copyright Statement

'I hereby grant the University of New South Wales or its agents the right to archive and to make available my thesis or dissertation in whole or part in the University libraries in all forms of media, now or here after known, subject to the provisions of the Copyright Act 1968. I retain all proprietary rights, such as patent rights. I also retain the right to use in future works (such as articles or books) all or part of this thesis or dissertation.
I also authorize University Microfilms to use the 350 word abstract of my thesis

in Dissertation Abstract International (this is applicable to doctoral theses only). I have either used no substantial portions of copyright material in my thesis or I have obtained permission to use copyright material; where permission has not been granted I have applied/will apply for a partial restriction of the digital copy of my thesis or dissertation.'

Signed ....

Date ....

Authenticity Statement

'I certify that the Library deposit digital copy is a direct equivalent of the final officially approved version of my thesis. No emendation of content has occurred and if there are any minor variations in formatting, they are the result of the conversion to digital format.'

Signed ....

Date ....

Publications from this Thesis

Conferences:

1. **J. Qadir**, C.T. Chou, A. Misra, and J.G. Lim, "Localized minimum-latency broadcasting in multi-radio multi-rate wireless mesh networks", submitted to IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2008.

2. **J. Qadir**, C.T. Chou, A. Misra, and J.G. Lim, "Localized minimum-latency broadcasting in multi-rate wireless mesh networks", In IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2007.

3. **J. Qadir**, C.T. Chou, and A. Misra, "Low latency broadcast in multi-radio multi-channel multi-rate wireless mesh networks", Proceedings of the 3rd Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON), 2006.

4. **J. Qadir**, C.T. Chou, and A. Misra, "Exploiting rate diversity for multicasting in multi-radio wireless mesh networks", Proceedings of the 31st IEEE Conference on Local Computer Networks (LCN), 2006.

Magazine:

1. C.T. Chou, A. Misra and **J.Qadir**, "Advances and challenges with data broadcasting in wireless mesh networks", To appear in IEEE Communications Magazine Featured Topics on Wireless Mesh Networks, 2007.

Journal:

1. C. T. Chou, A. Misra, and **J. Qadir**, "Low latency broadcast in multi-rate wireless mesh networks", IEEE Journal on Selected Areas in Communication (JSAC) Special Issue on Wireless Mesh Networks, 2006.

Workshop:

1. **J. Qadir**, C. T. Chou, and A. Misra, "Minimum Latency Broadcasting in Multi-rate Wireless Mesh Networks", ARC Communications Research Network (ACoRN) Early Career Researcher Workshop on Wireless Multihop Networking, Sydney, Australia, 2006.

Dedication

To

Allah (Most High),

and his last messenger Muhammad (Peace be upon him);

and then to my beloved Parents.

"And your Lord has decreed that you worship none but Him. And that you be dutiful to your parents" [Al-Quran: 17:23]

Calligraphy credit: Ted Thornton, Northfield Mount Hermon School

Acknowledgements

I begin by acknowledging and thanking Allah (Most High) for the numerous and unlimited bounties that He has bestowed upon me. Praise be to Allah (the All-Knowing), the Lord of the Worlds, who has made seeking knowledge praiseworthy. It is merely the grace of Allah (Most High) that I have managed to complete this dissertation. I begin by asking Allah (Most High) to accept my work, and make it a source of goodness for me in this world and in the hereafter.

I am greatly indebted to my supervisor, Dr. Chun Tung Chou, for his expert technical mentoring and model supervisory skills. Throughout the three and a half years of our association, he has been a great source of support. I recall in my starting years, how he was very kind and patient with me, and how he progressively got me thinking, firstly, how to precisely define a research problem, and subsequently on ways to solve this problem. Apart from his considerable technical and professional skills, I have learnt many things from Dr. Chun as a person. I have learnt from Dr. Chun to aspire for quality work only, and to work hard and honestly towards this goal. Without the continuous guidance of Dr. Chun, I would certainly never have been able to successfully complete this thesis.

I would also like to acknowledge and thank Dr. Archan Misra. He has been a valuable collaborator in my research, and a co-author of all my papers. He has also graciously hosted me for 2 weeks in IBM Research, Watson Research Center, New York in 2006. I would like to thank Dr. Archan for always providing valuable technical suggestions, and for his overall guidance.

I would like to thank all my colleagues at NRL, and especially: Adeel, Alex, Anjum, Assad, Baseem, Jerry, Joo Ghee, Malik, Nadeem, Nasir, Sarfraz, Usama, Xin. I would like to thank my friends Bilal, Usman, Waqar and Tanzeem for their valuable company, and constant encouragement. I would also like to thank the closely-knit Islamic community at UNSW. All of these people made my stay at UNSW extremely memorable.

I would like to extend thanks to the Higher Education Commission (HEC) of Pakistan, and to the National University of Science and Technology (NUST), Pakistan, for their joint scholarship. I would also like to thank University of New South Wales (UNSW) for awarding me the UPA scholarship that funded most of my candidature.

I cannot thank enough my parents for all their selfless love. I would like to acknowledge the numerous sacrifices they have made for me starting from the day I was born, and continuing to date. I would have been nothing without their love and compassionate support. I would also like to thank my siblings, my brothers and my dearest sister, for their love and affection. Last, but certainly not least, I am very grateful to my wife who has been very patient and supportive in the time I have been away from her. Needless to say without her sacrifices, I would not have been able to complete work on my thesis successfully.

Contents

List of Tables

List of Figures

List of Algorithms

List of Acronyms

Acronym	*Expanded form*	*Introduced in*:
WMN	Wireless Mesh Network	Section 1.1
WLAN	Wireless Local Area Network	Section 1.1
MANET	Mobile Ad-hoc Network	Section 1.1
MAC	Media Access Control	Section 1.1
SNR	Signal to Noise Ratio	Section 1.1
BER	Bit Error Rate	Section 1.1
WBA	Wireless Broadcast Advantage	Section 1.1
MR^2-MC	Multiple-Radio Multiple-Rate Multiple-Channel	Section 1.1
MLB	Minimum Latency Broadcasting	Section 1.2
MEB	Minimum Energy Broadcasting	Section 1.2
MR-MC	Multiple-Radio Multiple-Channel	Section 1.4
SR-SC	Single-Radio Single-Channel	Section 2
CDS	Connecting Dominating Set	Section 2.1.2
CCA	Common Channel Assignment	Section 2.4
VCA	Varying Channel Assignment	Section 2.4
INSTC	Interference Survivable Topology Control	Section 2.4
PER	Packet Error Rate	Section 4.2.1
SINR	Signal to Interference Noise Ratio	Section 4.2.1
MCDS	Minimum Connected Dominating Set	Section 4.3.1
WCDS	Weighted Connected Dominating Set	Section 4.3.1
CV	Cardinal Value	Section 4.3.2
SPT	Shortest Path Tree	Section 4.5
RAP	Rate-Area-Product	Section 4.7.1
Wu-Li	The algorithm of Wu and Li	Section 5.3
MDW	Multi-rate Delayed-pruning Wu-Li algorithm	Section 5.3
MEW	Multi-rate Expedited-pruning Wu-Li algorithm	Section 5.3
MRRA	Multi-point Rate-maximized Relaying Algorithm	Section 5.3
NG	Neighborhood Grouping	Section 5.3.2
RM	Rate Maximization	Section 5.3.2
FMM	Fully Multi-rate Multicast	Section 6.1
SBM	Single 'Best' rate multicast	Section 6.1
SCDS	Single-'best'-rate Connected Dominating Set	Section 6.4.2

Acronym	*Expanded form*	*Introduced in*:
MSPT	Multi-radio multi-channel Shortest Path Tree	Section 9.4.1
MWT	Multi-radio multi-channel WCDS Tree	Section 9.4.2
LMT	Locally-parallelized MWT	Section 9.4.3
PAMT	Parallelized Approximate-shortest MWT	Section 9.4.4
MRDT	Multi-radio Distributed Tree	Section 10.3
LRM	Local Rate Maximization	Section 10.3.3
ERM	External Rate Maximization	Section 10.3.3
PCDS	Parallelized Connected Dominating Set	Section 11.5

Abstract

This thesis addresses the problem of 'efficient' broadcast in a multi-radio multi-channel multi-rate wireless mesh network (MR^2-MC WMN). In such a MR^2-MC WMN, nodes are equipped with multiple radio network interface cards, each tuned to an orthogonal channel, which can dynamically adjust transmission rate by choosing a modulation scheme appropriate for the channel conditions. We choose 'broadcast latency', defined as the maximum delay between a packet's network-wide broadcast at the source and its eventual reception at all network nodes, as the 'efficiency' metric of broadcast performance. The problem of constructing a broadcast forwarding structure having minimal broadcast latency is referred to as the 'minimum-latency-broadcasting' (MLB) problem.

While previous research for broadcast in single-radio single-rate wireless networks has highlighted the wireless medium's '*wireless broadcast advantage*' (WBA) due to which a node's transmission can be received, assuming omnidirectional antennas, by all nodes within its communication range; little is known regarding how the new features of MR^2-MC WMN may be exploited. We study in this thesis how a MR^2-MC WMN's *rate-diversity* (WMN node's multi-rate transmission capability) and *radio-and-channel-diversity* (WMN nodes have multiple radio interfaces tuned to orthogonal channels) can be exploited, in addition to the WBA, to improve the 'broadcast latency' performance. It has been proved that the MLB problem for single-radio single-rate is NP-hard; clearly, MLB problem for MR^2-MC WMN is at least NP-hard. Designing a heuristic MLB solution for MR^2-MC WMN is also non-trivial as the network's rate-diversity, radio-and-channel diversity, and WBA must all be exploited and incorporated into design.

We divide the overall MLB problem for multi-rate WMNs into two subproblems, which we address in two separate parts of this thesis. *In the first part of this thesis*, the MLB problem is defined for the case of single-radio single-channel multi-rate WMNs where WMN nodes are equipped with a single radio tuned to a common channel. *In the second part of this thesis*, the MLB problem is defined for MR^2-MC WMNs where WMN nodes are equipped with multiple radios tuned to multiple orthogonal channels. We demonstrate that broadcasting in multi-rate WMNs is significantly different to broadcasting in single-rate WMNs, and that broadcast performance in multi-rate WMNs can be significantly improved if we exploit the available rate-diversity and radio-and-channel-diversity (which exists for MR^2-MC WMN only). We also present two alternative MLB broadcast frameworks and specific algorithms, centralized and distributed, for each framework that can exploit the multi-rate WMN's rate-diversity, WBA, and (if available) radio-and-channel-diversity to return improved 'broadcast latency' performance.

Chapter 1

Introduction

1.1 Overview

Wireless networks can be broadly classified as being either single-hop or multi-hop. Typically, single-hop wireless networks are called "*wireless local area networks*" (WLAN), while multi-hop wireless networks are referred to as "*wireless ad-hoc networks*". There are many types of wireless ad-hoc networks such as "*mobile ad-hoc networks*" (MANET), "*wireless mesh networks*" (WMN) [3], and "*wireless sensor networks*" (WSN) [4]. Wireless networks are an ongoing subject of extensive research, with standardization work including IEEE 802.11 [5] for WLAN and MANET, IEEE 802.11s [6] and 802.16 mesh-mode [7] for WMN, and IEEE 802.15.4 [8] for WSN.

WMN, the network setting this thesis considers, is a promising broadband access technology where mesh clients, that are potentially mobile, connect with a relatively stationary core of mesh routers using multi-hop wireless links [3]. The stationary mesh nodes[1] form a multi-hop wireless overlay such that an individual mesh node acts as both a forwarding relay between WMN nodes, and an access point to mobile and consumer devices in its vicinity. Since WMNs are dynamically self-organized and self-configured, with WMN nodes automatically establishing and maintaining connectivity, they promise easy network maintenance, robustness, and reliable service coverage [3]. WMNs, especially when built from commodity wireless cards that operate over unregulated spectrum, are also increasingly being recognized as a cost-effective, viable broadband solution for urban [9], rural [10], campus and office [11] environments.

The widespread adoption of multi-hop wireless networks in general, and WMNs in particular, is impeded by the relatively low spatial reuse of a single radio channel due to wireless interference. It has been shown that network capacity drops off as the number of nodes is increased in single-channel wireless networks [12]. While network

[1]The stationary mesh nodes can be mounted, for example, on residential rooftops or light poles.

capacity decreases with increasing nodes in single-channel wireless networks, the usage of multiple radio channels can promote simultaneous overlapped transmissions thereby improving the aggregate capacity. Current research indicates that equipping mesh nodes with multiple radio interfaces, tuned to distinct orthogonal channels, can significantly increase the capacity of the network [13–15] by exploiting concurrent spatial reuse of an individual channel. Fortunately, IEEE standards 802.11b and 802.11a offer a choice of 3 and 12 non-overlapping channels. The multiple interfaces on a WMN node can, therefore, tune to orthogonal channels, and increase capacity significantly.

Researchers are also beginning to move away from IEEE 802.11-based single-rate 'media access control' (MAC) protocols, and are studying the throughput and fairness issues that arise from multi-rate MAC protocols where adaptive modulation is used to dynamically modify the data rate on a particular link in response to the perceived signal-to-noise ratio (SNR) [16–19]. It is known that the received SNR in wireless networks is time-varying due to multi-path fading and interference [18]. For a given modulation scheme, this SNR variation also translates to variations in the bit error rate (BER). Since it is more difficult for the modulation scheme to decode the received signal for lower SNR, for a given modulation scheme, reduced SNR also translates to increased BER. It is known that higher rates are typically realized by using denser modulation encodings; a denser encoding scheme requires relatively higher SNR than a sparser encoding scheme to properly decode with a given BER [18]. If we assume constant transmitting power, it is observed that a tradeoff generally emerges between data rate and BER: the higher the data rate, the higher the BER. Thus, the encoded signals of higher-rate encodings—since they have higher BER and require higher SNR—can be decoded correctly within a smaller transmission range. *Therefore, when it is assumed that the same transmission power is used for all transmission rates, then, in general, the faster a transmission rate is, the smaller is its transmission range.* By employing rate-adaptation, WMN nodes can utilize the flexibility of multi-rate transmissions to make appropriate range and throughput/latency tradeoff choices across a wide range of channel conditions.

An important open question in such "*multi-radio multi-channel multi-rate WMN*" (MR^2-MC WMNs) that we address in this thesis, is how to perform efficient broadcast in such networks. Any routing protocol designed for broadcast must find a set of forwarding nodes that must relay the broadcasted packet of the source such that all network nodes receive. The problem of constructing "efficient" broadcasting forwarding structures has been an active area of research for both wired and wireless networks for quite some time now. Research has demonstrated that broadcasting paradigm of wireless networks is fundamentally different to that of wired networks due to wireless media's '*wireless broadcast advantage*' (WBA) [20] because of which a node's transmission can be received, assuming omnidirectional antennas, by all nodes lying within its communication range. Consequently for wireless communication, a single transmission suffices to reach multiple receivers if they are within transmitting node's communication range; this is unlike the wired communication case where multiple transmissions would have been needed. This is due to the shift in paradigm from the 'link-centric' nature of wired networks to the 'node-centric' nature of wireless communications [20].

Although, a lot of previous research work has focussed on broadcast in wireless multi-hop networks such as WSN and MANET, most of it is not relevant to multi-radio multi-channel multi-rate WMNs for reasons that follow. While WMN nodes are generally stationary and mains-powered, WSN and MANET nodes typically use disposable batteries. Therefore, typically, WSN and MANET designers have focussed on energy conservation when designing broadcast protocols [20–25]. WMN protocol designers, on the other hand, can focus more on performance based optimizations such as latency, throughput, capacity, etc. Multi-radio multi-channel multi-rate WMN also differ from traditional wireless multi-hop networks, which bulk of previous broadcast research address, in that nodes in such networks are equipped with multiple radio "*network interface cards*" (NIC) tuned to orthogonal channels that are able to adaptively change its link-layer transmission rate. Clearly, such MR^2-MC WMN cease to be an embodiment of the general MANET or WSN paradigm and introduce many fundamental challenges for broadcasting at both the protocol and architectural level.

Chapter Outline: We will now provide a general outline to the remainder of this chapter. We formally define our problem statement of 'minimum-latency-broadcasting' (MLB) in Section 1.2, and detail factors motivating our research in Section 1.2.1. We highlight the main research challenges and issues in MLB broadcasting in MR^2-MC WMNs in Section 1.3, and demonstrate that broadcast designers need to exploit the rate-diversity (Section 1.3.1) and radio-and-channel-diversity (Section 1.3.2) afforded by MR^2-MC WMNs. The research objectives of this thesis are outlined in Section 1.4. We summarize the contributions of this thesis in Section 1.5. This chapter is concluded in Section 1.6 with an outline of the rest of this thesis.

1.2 Minimum latency broadcast in MR^2-MC WMNs

As mentioned before, since WMNs are generally composed of stationary routers and nodes which are powered from mains, the performance of WMNs is generally benchmarked by high-performance metrics such as throughput and latency and not by metrics conventionally used for wireless networks such as energy-efficiency or the total number of transmissions. In our thesis, we have chosen ***"broadcast latency"*** as the metric with which we will evaluate the performance of broadcasting algorithms for MR^2-MC WMNs. We define the "broadcast latency" as the maximum delay between the transmission of a packet by the source node and its eventual reception by all the destinations. We define the problem of minimizing the broadcast latency as the 'minimum-latency-broadcasting' (MLB) problem.

The MLB problem in a multi-rate WMN (where a node's transmitting rate can take a range of discrete values) is significantly different from the minimum-energy-broadcasting (MEB) problem in a multi-power all-wireless network (where a node's transmitting power can take a range of values) that has been covered extensively in prior research [20–25]. Although both the MLB and the MEB problems, in general, attempt to exploit the WBA while varying the transmission-rate and transmission-power, respectively; the MEB problem tries to minimize the total consumption of energy (transmit power) whereas the MLB problem attempts to minimize the worst-case latency. Both problems are not the same, since non-interfering transmissions in an all-wireless networks (e.g., transmission of nodes further away than the interference range) can take place simultaneously in an overlapped manner. Thus, MLB becomes a more difficult problem since we have to take into account interference between wireless transmissions since a minimum spanning tree (MST with links' weight equal to their broadcast latency) may not necessarily be the MLB tree.

1.2.1 Motivation

The MLB problem in MR^2-MC WMNs is particularly challenging since such networks' multi-rate nature, WBA, and the availability of multiple radio interfaces (tuned to orthogonal channels) must be all exploited and incorporated into design. The MLB problem, apart from its theoretical significance, is an important practical problem in WMN. As many of the targeted broadcast-based applications of WMNs, e.g. IP-TV, audio conferencing, video-feeds and multi-player, multimedia games in community networks, are interactive and have strict latency requirements, we focus on how link-layer rate-diversity (WMN node's multi-rate transmission capability) and radio-and-channel-diversity (WMN nodes have multiple radio interfaces tuned to orthogonal channels) can be harnessed in multi-radio WMNs to improve the metric of broadcast latency.

Choosing latency as a performance measure also implicitly rewards approaches that use the WBA to reduce the number of distinct transmissions. This is because reduced transmissions also directly translate into lower contention induced delay. The upper-bounding of broadcast delay can enable the broadcast applications to provide QoS and possibly change service parameters when the worst-case broadcast latency varies. For example, VoIP and IPTV broadcast applications in community networks can dynamically adjust their service level that can be sustained by the worst-case broadcast latency.

Our study of the MLB problem is also motivated by the dearth of research in this important area. Gandhi et al. have studied the MLB problem for single-radio single-channel (SR-SC) single-rate wireless networks and proved that the problem is NP-hard [26]. Since the MLB problem in MR^2-MC WMNs is a more general case of the MLB problem in SR-SC single-rate WMNs, we can conclude that the MLB problem in MR^2-MC WMNs is at least NP-hard.

1.2.2 Research challenges of MLB problem in MR^2-MC WMNs

Traditionally, broadcasting approaches for wireless networks have been designed for single-rate networks and have not utilized two exploitable features, i.e., the rate-diversity, and the radio-and-channel-diversity, that MR^2-MC WMNs can offer. We will now summarize the problems that must be overcome for efficient broadcast in MR^2-MC WMNs.

1. Firstly, since different rates use different modulation schemes and have different transmission ranges, the neighbor sets of each rate are different; generally, the neighbor-set at a lower transmission rate includes all nodes in the neighbor-set of a higher transmission rate; therefore, the common implicit assumption in single-rate broadcasting algorithms that a node reaches all its neighbors in a single broadcast transmission (or that a node can only transmit once) returns sub-optimal results as we will show in the following examples. The extra degree-of-freedom of having a node perform multiple distinct-rate transmissions to reach different subset of neighboring nodes at different rates must be implemented in an intelligent manner to optimize our 'broadcast latency' metric. *Therefore, an important research challenge is to design algorithms that can decide how many transmissions, at distinct rates to different set of neighbors, must a node make for the same packet to ensure the best 'broadcast latency' performance.*

2. Since different rates have different transmission ranges, a network must be strongly-connected (to compensate for the asymmetry of multi-rate WMNs) when a single broadcast tree is used irrespective of the broadcast source (instead of source-based broadcast trees) to avoid partitioned networks in case of certain source nodes. We will expound on why this is a difficult issue in Chapter 5 of Part-I and Chapter 10 of Part-II; however, as a simple intuitive example, consider a linear IEEE 802.11b topology comprising of 3 nodes a, b and c as shown in Figure 1.1. Assume that the quickest rate supported on the link between $\{a, b\}$, and $\{b, c\}$ is 1 Mbps and 11 Mbps respectively. If a is the broadcast source, a possible connected dominating set (CDS) is a and b transmitting 1 and 11 Mbps respectively. However, this CDS cannot act as a global broadcast tree since it will not span node a if node b is the source (a is not a

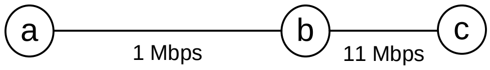


Figure 1.1: The requirement of strong connectivity when a single shared global tree for all broadcast sources

neighbor of b at 11 Mbps). *Accordingly, an important research challenge is to adapt to the routing challenges unique to multi-rate WMNs (and not present in single-rate WMNs) such as the need of strong connectivity when building shared global trees.*

3. Due to the availability of multiple radio interfaces tuned to orthogonal channels in MR^2-MC WMNs, the resulting radio-and-channel-diversity offers an opportunity of improved 'broadcast latency' performance through possible overlapped transmission on orthogonal channels. *An important research challenge is to design algorithms that can decide, depending on the channels assigned to different nodes' interfaces, the interface a transmitting node must use such that the 'broadcast latency' performance can be improved by employing increased parallelization of different transmissions in time.*

1.3 Novel features of MR^2-MC WMNs

Broadcast in MR^2-MC WMNs has not been extensively studied. Two new features of MR^2-MC WMNs offer extra degrees-of-freedom (DoF) that can be exploited by WMN protocol designers. These two features are the networks' rate-diversity and radio-and-channel-diversity. We will present in the next two subsections how these DoF can be exploited to realize minimum latency broadcast in MR^2-MC WMNs.

1.3.1 Rate-diversity

The ability to transmit at multiple distinct transmission rates by adapting the modulation scheme offers an interesting tradeoff to a broadcasting algorithm designer. With the assumption of uniform transmission power, a higher-rate transmission requires the use of a higher-rate encoding (which has higher BER and requires higher values of SNR for correct decoding). This implies that in general a higher-rate transmission has a smaller transmission range. For broadcast traffic, this generally implies a tradeoff between the transmitting rate and the number of neighbors a transmission can reach using WBA.

Also when multi-rate transmission capability is available, the implicit assumption in broadcasting algorithms that a transmitting node only needs to transmit once needs to be modified since a node might be required to transmit multiple times (at distinct transmission rates) to connect to all downstream neighbors. This possibility of multiple distinct-rate transmissions exists since while a node must connect to all its downstream neighbors, it is possible that the choice of a higher-rate transmission might connect to only a subset of downstream neighbors. In the extreme scenario, a node might be required to transmit L distinct times (where L is the number of distinct transmission rates supported by the underlying MAC) to connect to all the downstream neighbors if it chooses to group the distinct-rate transmissions in the sequence of a highest-rate transmission followed by a lower-rate transmission.[2]

We consider a simple example topology in Figure 1.2 with 5 nodes, labeled as Nodes 1 to 5, arranged in a straight line to gain the following two insights into multi-rate WMNs: firstly, broadcast tree formation and the MAC-layer scheduling are closely coupled, and secondly if a node in a distribution tree is limited to broadcasting a packet only once, it can lead to sub-optimal broadcast latency results. For simplicity, we will refer to Nodes 1 to 5 as N_1, N_2, N_3, N_4, and N_5 in the text. In Figure 1.2, the d value between 2 nodes indicates the physical distance in meters between them. We assume each node is equipped with an 802.11b radio tuned to the same channel. Using the Qualnet simulator [1] for reference and assuming a two-ray propagation model, we obtain the transmission range for different rates as shown in Table 1.1. The product of a transmission rate and its transmission area, called the rate-area-product (RAP), is shown in the last column of Table 1.1; the discussion of RAP is postponed till a future chapter (Chapter 4). Note also that the interference range in Qualnet is 520 m, i.e., the reception of a packet by a receiver will be unsuccessful if there are additional active transmitters within 520 m of the receiver. Thus, there are 4 links in the network configuration in Figure 1.2. Link (1,2) has a capacity of 11Mbps while the other three links have a capacity of 1 Mbps. Since our concern is packet delivery latency, we indicate the relative time required to send a packet for each link using the t value indicated in the Figure. Note that for simplification of our example, we make two ideal assumptions here: (1) The transmission time is computed based on the physical layer transmission rate; and (2) The Medium Access Control (MAC) layer is ideal (no collisions or backoff). The transmission time therefore ignores the overhead in packet headers, channel switching time and contention resolution. For the main results presented in this thesis, we shall demonstrate (via discrete-event simulation studies) that our fundamental insights hold even when we incorporate the overheads associated with a non-ideal MAC.

We assume that N_1 (i.e. Node 1) is the source node and it wants to send a packet to all the nodes in the network. Since the network is not fully connected, some nodes will need to act as a relay. We consider two different forwarding alternatives. In the

[2]This extreme scenario would ensue if for each of the L transmission rate, a higher-rate transmission covers only a subset of neighbors covered in the lower-rate (if any).

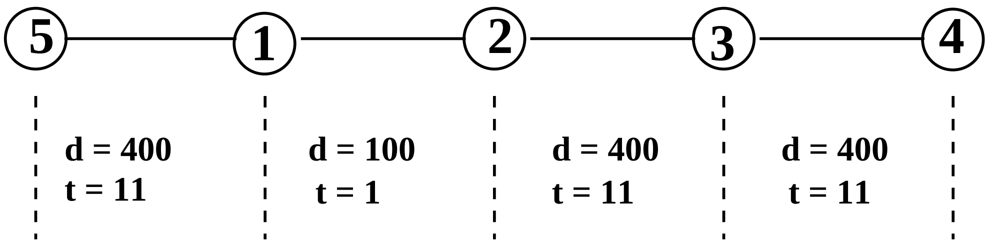


Figure 1.2: Motivating example for the multi-rate network-wide broadcast problem.

Transmission rate (Mbps)	Transmission range (m)	RAP (Mbps-km^2)
1.0	483	0.73
2.0	370	0.86
5.5	351	2.13
11.0	283	2.77

Table 1.1: This table shows the maximum transmission range and rate-area-product (RAP) for different IEEE 802.11b transmission rates obtained from Qualnet [1] assuming a two-ray model.

first approach, which we call Alt_1, each node is only allowed to broadcast the packet once. Due to this restriction, N_1 (the source node) must broadcast at the lower rate of 1Mbps to both N_2 and N_5, taking a time of 11 units to transmit the packet. Note that N_1 could not possibly use other transmission rates because N_5 will not receive the packet otherwise. This results in the transmission schedule depicted in Figure 1.3, and leads to a broadcast latency of **33** time units.

In the second approach, which we call Alt_2, we allow each node to broadcast the same packet more than once. Figure 1.4 depicts the transmission schedule. It shows the source N_1 transmitting the same packet two times. It first transmits to N_2 at 11Mbps (at time $t = 0$), taking 1 time unit. It then transmits the same packet again at time $t = 12$ to N_5 at a lower rate of 1Mbps. Note that the transmissions ($N_1 \rightarrow N_5$) and ($N_2 \rightarrow N_3$) cannot take place at the same time because of interference. In contrast to the first approach, the whole network-wide broadcast latency is now **23** time units. This examples illustrates the following important feature of broadcasting in multi-rate wireless meshes:

Property: *If a node is to multicast to a number of its neighboring nodes si-*

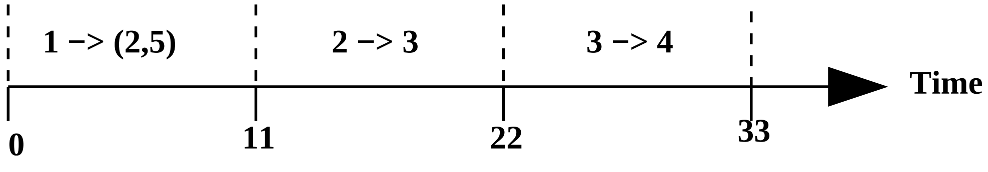


Figure 1.3: Alt_1*:* Transmission schedule if each node can only broadcast a packet at most once

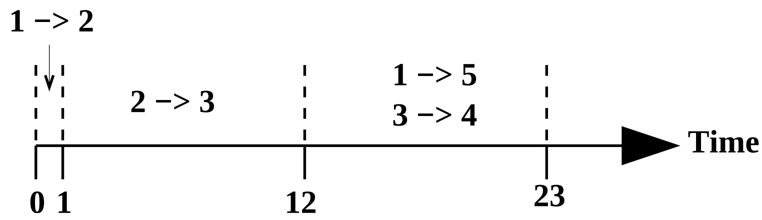


Figure 1.4: *Alt_2:* Transmission schedule if each node can broadcast a packet more than once

multaneously, the maximum broadcast rate that can be used is constrained by the lowest rate to reach all these nodes independently. Accordingly, if the objective is to improve the broadcast latency, a new DoF that can be used is to allow a node to transmit the same packet more than once, to different subsets of its immediate downstream neighbors.

By exploiting this DoF, an intermediate node can transmit the packet at a higher rate to children that lie along the "more critical" sub-trees (i.e., those that might take longer to forward the packet) to their leaf nodes, and subsequently use a lower-rate transmission to a subset of the "less critical" sub-trees. We point out that this DoF of allowing a node to transmit the same packet more than once have not been pointed out before our work. Note that this new DoF can be combined with others that have already been proposed, namely radio-and-channel-diversity [14] (discussed next) and network coding [27]. *It is instructive to point out that if the objective is to minimize the total energy consumption, then transmitting the same packet more than once will always result in worse performance.*

1.3.2 Radio-and-channel-diversity

When radio-and-channel-diversity is available, the multiple transceivers (alternatively, referred to as interfaces) of WMN nodes are tuned to orthogonal radio frequency channels. The usage of multiple radios and channels can be a double-edged sword for broadcast routing. Whereas on one hand, it reduces contention and interference between the different transmitters (broadcast source and relaying nodes) and increases capacity, it can possibly break down the WBA. Neighbors of a transmitting node that do not have an interface tuned to the same channel used by the transmitter would not be able to receive transmission even when we assume omnidirectional antennas and that SNR of received signal is above the reception threshold. Also, for a given number of interfaces per node, increasing the channel-diversity can lead to a disconnected network both for unicast and broadcast traffic. Kyasanur et al. have hinted about some of the potential problems that can be faced for broadcast routing in multi-radio meshes vis-a-vis channel assignment [28].

We first use a simple example topology to illustrate the potential degrees of

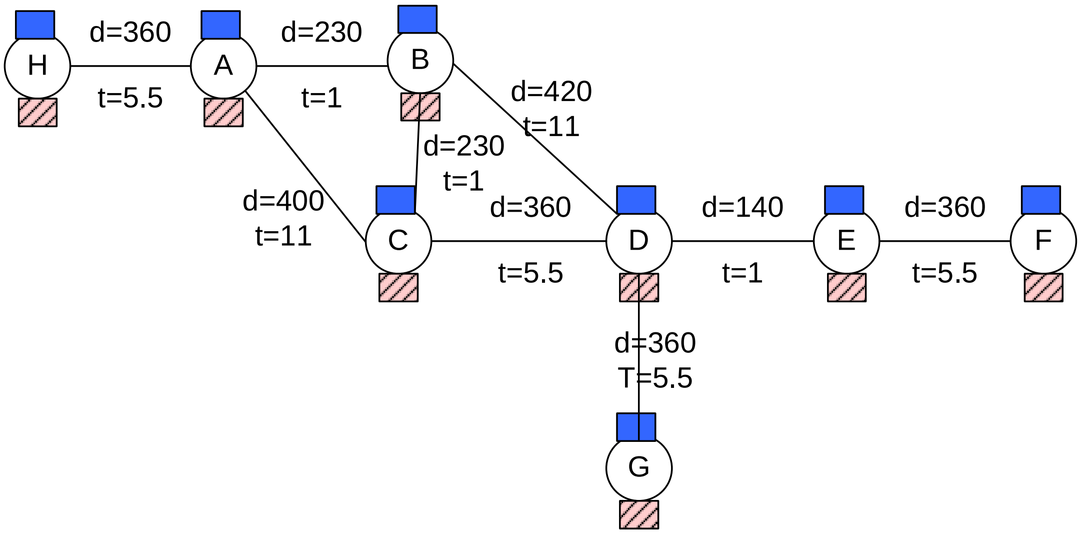


Figure 1.5: Sample WMN topology illustrating rate diversity and multi-channel mesh operation

freedom that may be available in a WMN due to the availability of multiple radios on each node. Figure 1.5 shows an IEEE 802.11b-based topology consisting of eight nodes $\{A, B, C, D, E, F, G, H\}$, where A is the source of a network wide broadcast. Assume that each node has two radios, with the blue (solid) and pink (striped) interface on each node denoting a radio tuned to channel C_1 and C_2 respectively. Each edge in Figure 1.5 includes the distance d between the neighbors and the assumed packet *transmission time* on that link. Note that the transmission times t in Figure 1.5 are normalized to the transmission time for the fastest rate (11 Mbps) and are inversely proportional to the link rate — thus, links with transmission rates 11, 2 and 1 Mbps have transmissions times $t = 1$, 5.5 and 11, respectively. The transmission rate used on a link in Figure 1.5 depends on the distance of the link. The quickest transmission rate that can cover the distance d can be observed from Table 1.1. The interference range, as mentioned before, remains 520 m. We have used the same assumptions as used in the previous section, which are: (1) The transmission time is computed based on the physical layer transmission rate; and (2) The Medium Access Control (MAC) layer is ideal (no collisions or backoff). The transmission time therefore ignores the overhead in packet headers, channel switching time and contention resolution for simplifying our example.

Returning back to our example topology of Figure 1.5, we initially consider the *case where only one radio* (the solid interface tuned to channel C_1) *is active* in each node. We assume that *each node is allowed to transmit the same packet more than once but at different rates.* Let us call this transmission scheme as Alt_3. In this case, if A first transmits the packet only to B at 11 Mbps ($t = 1$), this transmission would not be received by H as it can only decode transmissions at the lower 2 Mbps rate ($t = 5.5$). Moreover, the transmission $A \rightarrow H$ will not be able to proceed concurrently with the subsequent transmission $B \rightarrow C$ (rate of 11 Mbps with $t = 1$) as A's transmission would cause interference at C. However, the two transmissions $C \rightarrow D$ (rate of 2 Mbps with $t = 5.5$) and $A \rightarrow E$ (rate of 2 Mbps with $t = 5.5$) can proceed in parallel, if they both begin once C has received its packet. After D has received the packet, the remaining transmissions are $D \rightarrow \{E, G\}$ (rate of 2 Mbps with $t = 5.5$) followed by $E \rightarrow F$ (rate of 2 Mbps with $t = 5.5$). The broadcast latency $(1 + 1 + 5.5 + 5.5 + 5.5 =)$ **18.5** time units.

Continuing the example, we now assume that each node has both of its radio interfaces (channels C_1 and C_2) active. In this case, the nodes can exploit this additional level of concurrency to further reduce the number of interfering transmissions and the resulting latency. The transmission schedule Alt_4 is similar to Alt_3 above except that D would transmit on both radios but to different neighbors. The first four transmissions are (as in Alt_3): $A \rightarrow B$ $(t = 1)$, $B \rightarrow C$ (rate of 11 Mbps with $t = 1$), two simultaneous transmission by $A \rightarrow H$ and $C \rightarrow D$ (rate of 2 Mbps with $t = 5.5$). These transmission can take place in either of the channels. When D receives the packet, it transmits to E and G on two separate interfaces. Without loss of generality, we assume that $D \rightarrow E$ (rate of 11 Mbps with $t = 1$) is on channel C_1 and $D \rightarrow G$ (rate of 2 Mbps with $t = 5.5$) is on channel C_2. Once E has received the packet, it would transmit to F on channel C_1 using rate of 2 Mbps with $t = 5.5$. Note that the transmissions on C_1 and C_2 can take place simultaneously. Therefore, by exploiting multiple interfaces, the broadcast latency is reduced to $(1 + 1 + 5.5 + 1 + 5.5 =)$ **14** units.

As we have seen through this example, radio-and-channel-diversity offers a degree-of-freedom that can exploited to improve broadcast latency performance. It is also obvious that an approach that fails to exploit radio-and-channel-diversity will return suboptimal broadcast latency results.

1.4 Research objectives

In light of the discussion in the preceding sections, the objectives of the research presented in this thesis are given below.

1. Study the minimum-latency-broadcast (MLB) problem for multi-rate WMNs. Provide specific MLB algorithms that can improve the broadcast latency performance of single-rate broadcast algorithms proposed in literature.

2. Provide general insights and rules that can enable improved broadcast performance by helping a protocol designer exploit the inherent rate-diversity of multi-rate WMNs.

3. Provide general insight and rules that can enable improved broadcast perfor-

mance by helping a protocol designer exploit the inherent radio-and-channel-diversity of MR^2-MC WMN

1.5 Contributions

To fulfill the research objectives highlighted in the previous section, this dissertation makes multiple original contributions which are highlighted below:

1. This thesis points out the implications of multi-rate broadcast. It is shown that broadcast latency performance can be greatly improved by exploiting the rate-diversity of multi-rate WMNs. We study the effect of the extra degree-of-freedom of employing multiple distinct-rate transmissions at a node (for the same packet) on broadcast performance. This thesis also proposes a general rule-of-thumb (the usage of Rate-Area-Product (RAP) in Chapter 4) that can enable future WMN designers to predict the usefulness of particular MAC transmission rates for broadcast traffic. This rule-of-thumb can aid future WMN designers to decide, if it is decided as a design choice to utilize only a subset of possible transmission rates for broadcast traffic, which particular transmission rates are more useful.

2. This thesis provides heuristic algorithmic solutions to the NP-hard MLB problem for SR-SC multi-rate WMNs. Our work offers both centralized and decentralized (and localized) solutions. Our best performing centralized solution, called WCDS, returns near-optimal broadcast latency results (Chapter 4). We have also proposed a decentralized and localized algorithm requiring limited-topology information (called MDW) whose performance is not much worse than WCDS (Chapter 5). It is established through detailed simulations that our algorithms improve the performance of algorithms proposed in literature that do not exploit the available rate-diversity.

3. This thesis also provides multiple heuristic algorithmic solutions to the NP-hard MLB problem for the MR^2-MC WMN. Our work offers both centralized and decentralized (and localized) solutions. Our best-performing centralized solution, called PAMT, is an adaptive algorithm that adapts to the rate-diversity and radio-and-channel-diversity available in a MR^2-MC WMN (Chapter 9). We have also proposed MRDT, a distributed heuristic solution to the MLB problem for MR^2-MC WMN, which can approach the performance of the centralized PAMT algorithm especially when the number of radio interfaces are large (Chapter 10). It is established through detailed simulations that our algorithms improve the performance of existing algorithms by appropriately exploiting the rate-diversity and radio-and-channel-diversity of MR^2-MC WMNs.

4. This thesis highlights the significant effect of channel-assignment on broadcast latency performance in MR^2-MC WMNs. This stems from the somewhat conflicting requirements of broadcast flows of greater connectivity (to exploit WBA) and at the same lower channel contention (to avoid wireless interference). Perhaps a more important observation established by this thesis is that a channel assignment scheme designed for unicast flows may perform poorly for broadcast flows (Chapters 9 and 10).

5. This thesis proposed two alternative frameworks for broadcasting in MR^2-MC WMNs. The first alternative, called the FMM framework (Chapters 4, 5, 9 and 10), returns the best results but it is also more expensive to implement since it can require significant changes to existing MAC protocols. The second alternative, called the SBM framework (Chapters 6 and 11), can approach the performance of FMM framework with relatively smaller implementation costs.

1.6 Outline

In this section, we shall give a general outline of the rest of the dissertation. This dissertation consists of two parts as we divide the general problem of MLB broadcasting in multi-rate WMNs into two logical strands based on the number of radio interfaces WMN nodes are equipped with. Both these parts are comprised of multiple chapters. The chapter outline for a part is provided in the first chapter of that part.

1. The first part of our dissertation, called "*Improving broadcast performance of single-radio single-channel (SR-SC) multi-rate WMNs*", deals with MLB broadcasting in multi-rate WMNs where each WMN node is equipped with only one radio interface which is tuned a single common radio channel. This part of the thesis, therefore, only deals with networks in which rate-diversity is available but not radio-and-channel-diversity.

2. The second part of our dissertation, called "*Improving broadcast performance of multi-radio multi-channel, multi-rate (MR^2-MC) WMNs*", deals with MLB broadcasting in MR^2-MC WMNs where each WMN node is equipped with multiple radio interfaces (tuned to different orthogonal channels). This part of the thesis, therefore, deals with networks that not only offer rate-diversity but also radio-and-channel-diversity.

Chapter 2

Literature Review

In this chapter, we will review literature relevant to the problem considered in this thesis. Initially, we will survey existing broadcasting techniques for (single-rate) wireless networks in Section 2.1, and categorize them based on common features. These broadcasting techniques have generally been proposed for single-radio single-channel (SR-SC) single-rate wireless networks. We will follow this discussion with a brief survey of existing unicast and broadcast routing protocols that have been proposed for multi-radio multi-channel multi-rate (MR^2-MC) WMNs in Section 2.2 and 2.3, respectively. *Most existing routing protocols for MR^2-MC WMNs only cater for unicast traffic, and literature on multicast/ broadcast routing protocols for MR^2-MC WMNs is conspicuously sparse.* Finally, we will discuss the channel-assignment problem, which is unique to MR^2-MC WMNs, in Section 2.4.

2.1 Broadcasting in single-rate wireless networks

Various researchers have focussed their attention on wireless broadcasting. Broadcasting in single-rate multi-hop wireless networks has been an active area of research for quite some time now, and a diverse variety of protocols have been proposed [29] [30][1]. In what follows, a brief survey of existing broadcast routing protocols, categorized according to common characteristics, is presented.

2.1.1 Architectural paradigm based classification

1. ***Centralized operation:*** Centralized broadcast routing protocols, typically, delegate the responsibility of calculating the forwarding structure to a single node. This often leads to great processing overload on the centralized processing node which also, typically, requires global network information. This causes this approach to be expensive in terms of computation, and communication cost. Centralized algorithms, though not very useful practically for

[1] Although, this survey specifically addresses multicast, it is applicable to broadcast as well.

most wireless scenarios, are generally easier to design and simpler in operation. Some example centralized algorithms are the works of Wieselthier et al. [20] and of Guha et al. [31].

2. ***Decentralized operation:*** Decentralized or distributed routing protocols, typically, construct the routing forwarding structure in a distributed manner utilizing only localized information. Decentralized operation avoids the problems of overloaded central servers, and lends itself to scalable operation by generally requiring only limited neighborhood and topology information. Example distributed broadcasting algorithms include the works of Wu et al. [32], Qayyum et al. [33], Stojmenovic et al. [34], Al-zoubi et al. [34], Ni et al. [35], Lim et al. [36], and Lee et al. [37].

2.1.2 Routing paradigm based classification

1. ***Simple flooding:*** Flooding is a simple approach to broadcasting in which a broadcast packet is forwarded by every node in the network exactly once. Simple flooding, however, results in a high degree of redundancy and significant collisions at the MAC layer, leading to the so-called broadcast storm problem [35] [36]. Despite its drawbacks, many protocol designers resort to flooding (or, some approximation thereof) for broadcasting in highly mobile networks like MANET to ensure packet delivery.

2. ***Probability-based model:*** Ni et al. [35] proposed a probability-based approach to broadcast forwarding. It is very similar to flooding, in which each node forwards a non-duplicate packet with a probability of 1, in that each node forwards according to a pre-determined probability, however, this probability is not necessarily equal to unity. The setting of forwarding probability at the nodes should take into account the density of nodes in the network to ensure eventual delivery. Another work proposing a probability-based approach is the work of Haas et al. [38]. However, it is to be noted that the probabilistic approach cannot guarantee full coverage, with or without mobility and collision. In order to achieve a reasonably high delivery ratio, the forwarding-probability is usually conservative and yields a relatively large forward node set.

3. ***Backbone-based broadcasting:*** There are numerous algorithms that attempt reduction of the forwarding-node set required to reach each node in the network. These algorithms, alternatingly referred to as backbone-based routing [39], dominating-set-based routing [32, 40], and/or spine-based routing [41, 42], construct a small set of nodes that form a Connected Dominating Set (CDS) of all nodes. CDS of the nodes of the network, whose topology is represented by a graph $G = (V, E)$, is a connected subgraph of G spanned by

the nodes of $V' \subseteq V$ such that every node in the network is at most one hop distant from a node in V'. A good backbone, traditionally, is minimal in size; however, in case of multi-rate WMNs, there is also the additional question of which transmission rate should the transmitting nodes in the CDS use. For same-sized backbones, it is preferable that the CDS forwarding nodes utilize higher transmitting rates to ensure high broadcast performance. In the next few paragraphs, we will expand more on backbone-based broadcasting since we will be using these techniques in our thesis extensively.

The forwarding-node set or the CDS can be selected *statically* based on topology information only [32, 43], or *dynamically* based on both topology and broadcast state information [34, 36, 44]. We broadly classify backbone-based broadcasting (also called CDS-based techniques) into two further types: a) centralized CDS-based techniques, and; b) distributed and localized CDS-based techniques. These two types are discussed next.

The first ***centralized CDS construction*** strategy was proposed by Guha in his seminal work [31] which contains two greedy heuristic algorithms with bounded performance guarantees. In the first algorithm, the CDS is grown from one node outward. In the second algorithm, a weakly-connected CDS is constructed, and then intermediate nodes are selected to create a CDS. The distributed implementations of both algorithms were provided by Das et al. [40]. Many algorithms designed latter [45] are motivated by either of these two heuristics. Another example centralized CDS construction algorithm is the work of Butenko et al. who have proposed pruning-based algorithm [46].

Various ***distributed and localized CDS construction*** strategies have been proposed, including the works of Wu et al. [32], Qayyum et al. [33], Stojmenovic et al. [34], and Al-Zoubi et al [45]. These distributed protocols are essentially divided into two classes:

- *Distributed CDS algorithms of the first class* (e.g. Wu et al.'s algorithms [32] [47], and Adjih et al.'s algorithm [43]) initially compute a large CDS and then attempt to prune away redundant nodes by means of local optimizations.

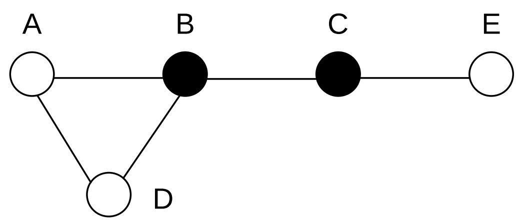


Figure 2.1: Example to explain Wu-Li algorithm's marking; dark and light nodes represent gateway and normal host respectively.

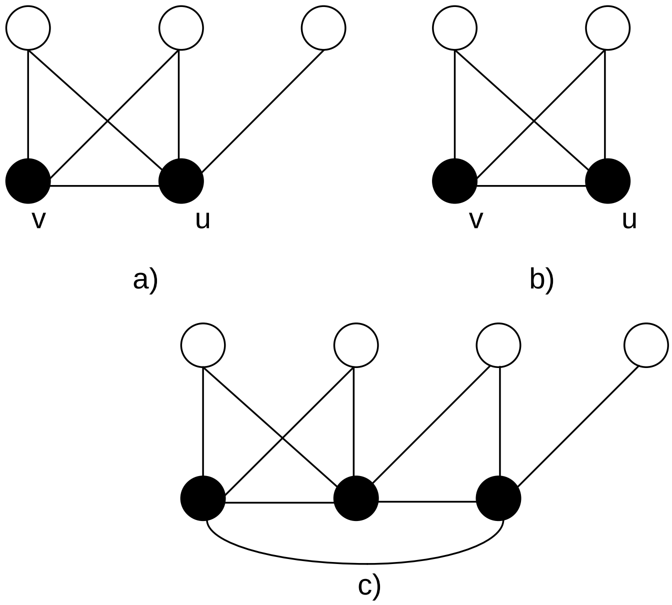


Figure 2.2: Example to explain Wu-Li algorithm's pruning rules; dark and light nodes represent gateway and normal host respectively.

(a) Wu et al.'s algorithm [32] is a simple localized technique that uses only 2-hop information to compute a CDS. It comprises of a marking phase in which a relatively large CDS is calculated, followed by a pruning phase in which redundancy in the CDS is reduced by pruning away some nodes from the CDS.

The marking process takes place as following: (1) Initially assign marker F to every v in V. (2) Every v exchanges its open neighbor set $N(v)$ with all its neighbors. (3) Every v assigns its marker $m(v)$ to T if there exist two unconnected neighbors. In the example of Figure 2.1, $N(A) = B, D$, $N(B) = A, C, D$, $N(C) = B, E$, $N(D) = A, B$, and $N(E) = C$. After the Step 2 of the marking process. Vertex A has $N(B)$ and $N(D)$, B has $N(A)$, $N(C)$, and $N(D)$, C has $N(B)$ and $N(E)$, D has $N(A)$ and $N(B)$, and E has $N(C)$. Based on Step 3, only vertices B and C are marked T. After the marking stage, a relatively large CDS V' results from all the vertices that have been marked T.

Two pruning techniques are then used to reduce the CDS size. A node u can be removed from V' if there exists a higher-id node $v \in V'$ such that the closed neighbor set[2] of u is a subset of the closed neighbor set of v. For the same reason, a node u will be deleted from V' when two of its connected neighbors in V' with higher IDs can cover all of u's neighbors. This pruning idea is generalized to

[2]Closed neighbor set is the union of the node itself and its neighbors.

the following rule [32]: a node u can be removed from S if there exist k connected neighbors with higher IDs in S that can cover all u's neighbors. In Figure 2.2 (a), since $N[v] \subset N[u]$, vertex v is removed from G' if $id(v) < id(u)$ and vertex u is the only dominating node in the graph. In Figure 2.2 (b), since $N[v] = N[u]$, either v or u can be removed from G'. To make sure one and only one is removed, we pick the one with a smaller id. Now we will consider the example in Figure 2.2 (c). Clearly, $N(v) \subset N(u) \cup N(w)$. If $id(v) = \min\{id(v), id(u), id(w)\}$, vertex v can be removed from G' based on Rule 2. If $id(u) = \min\{id(v), id(u), id(w)\}$, then vertex u can be removed based on Rule 1, since $N[u] \subset N[v]$. If $id(w) = min\{id(v), id(u), id(w)\}$, no vertex can be removed. Therefore, the id assignment also decides the final outcome of the dominating set. Note that Rule 2 can be easily extended to a more general case where the open neighbor set of vertex v is covered by the union of open neighbor sets of more than two neighbors of v in G'. However, the connectivity requirement for these neighbors is more difficult to specify at v.

(b) Adijh et al.'s technique to locally compute a CDS is called "multi-point relaying" (*MPR*) [43]. The MPR technique allows each node u to first elect a 'multi-point relay set" (*MRS*) [48] [33] from its one-hop neighbors to cover its two-hop neighbors. Finding a MRS with minimum size is NP-Complete [33]. The CDS is calculated as follows [43]: each node first computes a MRS, a subset of one-hop neighbors that can cover all its two-hop neighbors. After each node has determined its MRS, a node decides that it is in the connected dominating set or not by matching either *Rule 1:* the node is smaller than all its neighbors *or Rule 2:* it is the multipoint relay of its smallest neighbor.

- *Distributed CDS algorithms of the second class* (e.g. Al-zoubi's algorithm [45]), on the other hand, firstly calculate a small dominating set and then connect it up. The CDS calculated by the second class of algorithms is generally smaller than the CDS calculated by the first class of algorithms [45]; however, the smaller cardinality of the set of forwarding-nodes set comes at the expense of increased complexity and reduced locality.

CDS-based or backbone-based are also sometimes classified according to which node determines the forwarding status (forward or non-forward) of a node. In self-pruning algorithms [32, 34, 49], each node makes its local decision on forward status (i.e., whether it is a forward node or non-forward node). In neighbor-designating methods [36, 44], the forward status of each node is determined by its neighbors.

2.1.3 Forwarding structure based classification

1. ***Tree-based broadcasting:*** Tree-based broadcast is often used for building wired broadcast protocols. It is especially applicable for use in networks with low-loss links (e.g. wired networks) or in networks where nodes have minimal mobility. Using a tree-based approach can often reduce contention problems and reduce the overheads of more redundant forwarding structures. However, it is ill suited to mobile networks or networks with high-loss links (e.g. MANET). Examples of tree-based protocols are the following (early) protocols devised for operation in ad-hoc networks: AMRIS [50], MAODV [51] and LAM [52]. Obraczka et al. [53] reported that with increased node mobility and network load, these protocols do not perform well due to their fragile forwarding structure. This is, however, not as much of a problem in pre-dominantly static networks like Wireless Sensor Networks (WSN) and WMN.

2. ***Mesh-based broadcasting:*** Mesh-based broadcast offers a more redundant forwarding structure where multiple paths can exist between source-destination pair. Existing studies [37] have shown that mesh-based protocols seem to perform better for highly mobile networks than tree-based structures. ODMRP [37], CAMP [54] and FGMP [55] are examples of mesh-based protocols. Although, mesh-based protocols perform well in mobile networks and in the presence of high-loss links, the control overhead and maintenance cost can be unacceptably high for certain applications and topologies.

3. ***Hybrid-structure-based broadcasting:*** Hybrid structure protocols aim to have the resilience of mesh-based protocols with the simplicity of tree-based protocols. AMROUTE [56] is a an example hybrid-structure broadcast protocol that maintains virtual mesh links to ensure the broadcast forwarding tree remains unchanged when topology changes; its main disadvantage is that it may have temporary loops and may create non-optimal trees in case of mobile hosts. MCEDAR [57] is another hybrid protocol that attempts to have a robust mesh-based forwarding structure and also approximate the efficiency of tree-based forwarding.

2.1.4 Optimization metric based classification

1. ***Energy efficiency:*** A common metric, often used for broadcast protocols in power-constrained settings like WSN and MANET, is power-efficiency. Since nodes in such networks are battery powered, energy is an extremely valuable resource. Wieselthier et al. [20] presented three "energy-efficient" broad-

casting tree algorithms (BIP, BLU and BLiMST) for use in MANETs. The minimum-energy broadcasting problem explores the tradeoff between the energy consumption of a transmission and its reach; in general, the higher a transmission's transmit power, the more range it can cover and possibly the more neighbors it can connect to. Amongst the three algorithms proposed by Wieselthier et al., BIP is a variation of Prim's algorithm and adds nodes to the tree while reducing the incremental costs; BLU superimposes least-cost unicast paths on top of each other (same as the conventional SPT that does not take WBA into account) while BLiMST connects all the nodes while minimizing the accumulated sum of weights of all links (same as the conventional MST; does not take WBA into account). Of these three protocols, only BIP exploits the WBA, while none of these algorithms take possible rate-diversity or radio-and-channel-diversity into account (since these algorithms were designed for SR-SC single-rate wireless networks). In other work, Cartigny et al. proposed localized techniques to achieve energy-efficient broadcast [21], whereas Agarwal et al. [22] and Widmer et al. [23] proposed hitchhiking and network coding techniques, respectively, to achieve energy-efficient broadcast.

2. ***Number of transmissions:*** Traditionally, this is a commonly used metric to gauge the performance of a broadcast routing protocol[3]. The base-case (without any optimization) is the case of flooding where every node is a forwarding node. Protocols of this category attempt to minimize the number of transmission required for the broadcast message to disseminate to all nodes [32] [58].

3. ***Overhead in route-discovery and maintenance:*** A broadcast routing protocol's operation costs should be minimal for scalable operations; therefore, another metric used for performance evaluation is the overhead in route calculation and maintenance. Gui et al. [59], have tried to optimize the metric of "operational costs" by reducing the overhead in route calculations so that the broadcast algorithms can scale to large networks.

4. ***High-performance metrics (e.g., throughput, latency):*** Recently, researchers have started focussing more on high-performance metrics for evaluating broadcast performance using metrics such as broadcast latency [26] and high-throughput [60]. These metrics are suitable for networks such as WMNs whose nodes are largely static and powered from mains, since energy-efficiency for such networks is not a over-riding criteria, shifting the optimization focus on to high-performance metrics such as throughput and latency.

[3] Assuming fixed transmission power, minimizing the number of transmissions then translates to the case of minimization of transmitting-power/energy-consumption.

5. ***Multiple optimization metrics:*** Certain broadcast protocols take multiple optimization objectives into account. Examples of such protocols are the work of Roy et al. [61], that attempts to optimize a combination of different metrics like end-to-end latency and residual bandwidth utilization, and the work of Alba et al. which attempts to optimize both the reliability of broadcast and also its end-to-end latency.

2.2 Unicast routing in multi-rate WMNs

Unicast routing in SR-SC multi-rate wireless networks: Research has demonstrated that the hop-count routing metric does not perform particularly well in a multi-rate environment [16,62–64]. Since, a lower hop-count tends to result from using lower transmission rates (which generally have larger transmission ranges), the use of hop-count metric can result in reduced end-to-end performance. Instead of using the hop-count metric, Awerbuch et al. [16] have proposed the 'medium time metric' (MTM) which essentially measures the time taken to transmit a packet over a path composed of multi-rate links while taking into account the transmission delay, overheads of the RTS/CTS/ACK frames and channel contention. Since the transmission delay is inversely proportional to the transmission rate used by a link, MTM metric improves performance by choosing a path that can provide higher end-to-end throughput. Awerbuch et al. also showed that if we assume total interference (i.e., all transmissions interfere), then the path minimizing the MTM metric also maximizes the throughput between the source and destination. Seok et al. [63] proposed a multi-rate aware sub layer (MAS), which is independent of IP protocol and enables the full utilization of the multi-rate channel characteristics, which can improve throughput and resource utilization performance. Zhao et al. proposed the cross-layer PARMA routing metric that takes into account both physical layer link rate as well as estimated channel congestion, thus aiming to minimize end-to-end delay that includes both transmission and access times [64].

Unicast routing in MR^2-MC wireless networks: It has been observed that unicast routing performance can be significantly improved by exploiting the *radio-and-channel-diversity* that is available in MR^2-MC WMNs due to the availability of multiple radio interfaces tuned to orthogonal channels [14, 65]. The WCETT (weighted cumulative expected transmission time) routing metric, proposed by Draves et al. [14] specifically for MR^2-MC wireless networks, calculates the ETT (expected transmission time) of each hop and prefers paths based on the path's cumulative ETT (CETT) and its channel-diversity. The channel-diversity is characterized indirectly by the sum of ETTs of hops operating at the bottleneck frequency channel (the channel, which amongst all channels, has the maximum cumulative ETT). The

WCETT metric offers a tradeoff, through a tunable parameter β, between the path-length and channel-diversity of chosen routes. In another related work, a new routing metric called AETD (adjusted expected transfer delay) was proposed by Zhou et al. [65] which differs from the work of Draves et al. in the way the channel-diversity of the network is incorporated into design. The key idea of AETD is to make the routing decision based on the expected end-to-end transfer delay of a single packet as well as the expected delay jitter between consecutive packet transmissions, which serves as a good indicator of the channel-diversity level. There is also some recent work that has performed theoretical analysis on joint optimization of channel assignment and routing in multi-radio multi-channel wireless networks [66, 67].

2.3 Broadcast routing in multi-rate WMNs

Broadcast routing in SR-SC multi-rate wireless networks: There was a dearth of research in the area of broadcast routing in SR-SC multi-rate WMN before the work on this thesis started. Along with the work presented in this thesis, some other work in this area has recently been proposed. Traditionally, while the current IEEE 802.11a/b/g standards mandate the transmission of the control frames (e.g. RTS/CTS/ACK) at the lowest rate (e.g., 6 Mbps for IEEE 802.11a), transmission rates for broadcast data are typically implementation-specific. It has already been demonstrated (Chapter 1) that broadcasting schemes that do not exploit the multi-rate nature of WMNs can return sub-optimal performance.

In SR-SC multi-rate WMNs, Minimal Connected Dominating Set (MCDS) has been shown to be an effective technique. The use of MCDS to achieve optimal flooding in a single-rate multi-hop wireless networks has been explored in [36] where the authors prove that the size of the optimal flooding tree (measured by the number of nodes performing broadcasts, not by broadcast latency) differs from the size of the MCDS by at most one. However, MCDS performs poorly in multi-rate environments because it does not account for multi-rate links in the tree construction. Techniques that can be used to calculate MCDS have already been discussed in Section 2.1.2. In another work, Nguyen et al. have proposed a *multi-rate multicast* distributed algorithm for SR-SC multi-rate wireless networks called "*rate adaptive multicast*" (RAM) based on the "*on-demand multicast routing protocol*" (ODMRP) [37] algorithm. The RAM protocol does not exploit the WBA explicitly, and incurs large overhead for static WMNs since it does not attempt to minimize the 'forwarding group' size or to maximize the transmission rates at the forwarding nodes.

Broadcast routing in MR^2-MC wireless networks: There was limited research on broadcasting in MR^2-MC WMNs, like in SR-SC multi-rate WMNs, before work on this thesis started. Kyasanur et al. [28] had hinted about some of the potential problems that can be faced for broadcast routing in multi-radio meshes vis-a-vis channel assignment. We will elaborate more about the problem of channel-assignment in MR^2-MC WMNs in the next section. Kyasanur et al. also proposed to simply transmit a copy of the broadcast packet on every channel or use a separate broadcast channel at the expense of a dedicated interface.

Broadcasting in MR^2-MC WMNs has received some recent research attention in 2007 [68] [69] [70]. These recent work have appeared when considerable work on this thesis was already complete, and these works have cited some of the work presented in this thesis. Wang et al. have recently proposed a 'Distributed Rate-First' algorithm for use in SR-SC multi-rate WMNs [70]. This work, however, does not utilize the radio-and-channel-diversity available in a MR^2-MC WMN as it is designed specifically for SR-SC multi-rate WMNs. Song et al. have recently proposed a distributed broadcast algorithm for MR^2-MC WMN [68], this algorithm assumes that each channel can only use a particular rate. In other words, for the algorithm in [68], once a node has decided to use a particular channel, the link-layer transmission rate to be used by that node cannot vary. However, our MR^2-MC WMN setting (Chapters 9, 10 and 11) is completely general where any transmission rate can be used with any channel. In other work, Li et al. have proposed a self-pruning based protocol called Multi-Channel Self-Pruning (MCSP) [69] which reduces the broadcasting problem in multi-radio multi-channel WMNs into the minimal strong connected dominating set problem of the interface-extend graph (which is a graph they propose which extends the original network topology across interfaces). However, this work assumes single-rate WMNs and as such does not incorporate the rate-diversity of MR^2-MC WMNs into design.

2.4 Channel-assignment in MR^2-MC WMNs

Before discussing routing in MR^2-MC WMNs, we will discuss the intimately tied channel-assignment problem in MR^2-MC WMNs. The channel-assignment problem originates from the infeasibility of equipping WMN nodes with a dedicated radio interface for each supported radio channel; e.g., note the impracticality of equipping 802.11a networks, which support 12 orthogonal channels, with a dedicated radio. Since, the number of radio interfaces is generally less than the number of orthogonal channels, the channel-assignment defines the radio channel assigned to different interfaces of WMN nodes[4]. Generally there are two conflicting objectives for any channel assignment protocol: while nodes will usually benefit from increased 'connectivity' among themselves, the channel assignment protocol also tries to reduce some measure of 'interference'. Channel assignment strategies can be broadly

[4]There are single interface approaches to exploiting multiple radio channels; however, in such schemes, a WMN node can only operate in simplex mode [71] [72].

classified into *static*, *dynamic*, and *hybrid* approaches [28]. In the static channel assignment approaches, each interface is assigned a channel either permanently, or for a long interval of time where long interval is defined relative to the interface switching time [14] [73]. Amongst the static channel assignment strategies, the simplest approach is the '*common channel approach*' (CCA) (e.g. [14]), in which all nodes are assigned a common set of channels. The benefit of this approach is its simplicity and that the connectivity of the network is a multiple of the connectivity of a single channel mesh. In an alternative approach called '*varying channel approach*' (VCA), interfaces of different nodes may be assigned to a different set of channels (e.g. [73]). With this approach, there is a possibility of a network partition, unless the interface assignment is done carefully. In yet another approach called '*interference survivable topology control*' (INSTC) [74], the channel assignment is made such that the induced network topology is interference-minimum among all k-connected topologies. Dynamic assignment strategies allow any interface to be assigned to any channel, and interfaces can frequently switch from one channel to another. The benefit of dynamic assignment is the ability to switch an interface to any channel, thereby offering the potential to cover many channels with few interfaces. The key challenge with dynamic switching strategies, however, is to coordinate the decisions of when to switch interfaces as well as what channel to switch the interfaces to, among the nodes in the network. Lastly, hybrid assignment strategies combine static and dynamic assignment strategies by applying a static assignment for some interfaces and a dynamic assignment for other interfaces.

In this thesis, only the static channel-assignment schemes of CCA, VCA and INSTC are considered. We note that the focus of this thesis is not the channel-assignment problem, but the algorithmic routing aspects of the MLB problem. Also, since the channel-assignment scheme influences unicast traffic as well, it has been decided not to integrate channel-assignment into the MLB routing problem. Accordingly, we focus on how to efficiently perform efficient minimum-latency broadcasting for a given MR^2-MC WMN *with known channel-assignment*. Furthermore, we assume only static channel-assignment schemes in this thesis, and do not cater for dynamic reassignment of channels in MR^2-MC WMNs.

2.5 Summary

In this chapter, we have presented the related work to the problem of minimum-latency broadcast (MLB) in multi-radio multi-channel multi-rate (MR^2-MC) WMNs. We have shown that limited work is available for broadcasting in such networks, and existing work deals primarily with single-radio single-channel (SR-SC) single-rate wireless networks. We have presented a detailed survey of broadcasting protocols for SR-SC single-rate wireless networks. We show that routing research for multi-rate WMNs has primarily focused on unicast traffic, with very little work done on multi-rate broadcast. We discuss some of these unicast multi-rate routing protocols to gain general insights into routing for multi-rate WMNs. We also discuss the limited research that has been proposed for broadcasting in MR^2-MC WMNs.

Part I

Improving broadcast performance of single-radio single-channel (SR-SC) multi-rate WMNs

Chapter 3

Introduction to Minimum-Latency Broadcasting in SR-SC Multi-Rate WMNs

3.1 Overview

This chapter serves as an introduction to Part I, "*Improving broadcast performance of single-radio single-channel (SR-SC) multi-rate WMNs*". The specific broadcasting problem addressed in this thesis is the "minimum latency broadcast" (MLB) problem that was described in Section 1.2. Broadcasting in SR-SC single-rate wireless networks has earlier been discussed in Section 2.1. In this part of the thesis, the MLB problem is considered for the case of SR-SC multi-rate WMNs in which all the WMN nodes are equipped with only a single radio interface tuned to a common radio channel.

3.2 Outline

An outline of the remaining chapters of this part of the thesis is as follows:

Chapter 4: Centralized MLB Solution for SR-SC Multi-rate WMNs

In this chapter, after formally defining the MLB problem for SR-SC multi-rate WMNs and demonstrating its *NP-hardness*, it is shown that the multi-rate broadcast problem is significantly different from the single-rate case; thereafter, a centralized rate-aware heuristic called WCDS is proposed that exploits both the wireless broadcast advantage (WBA) and the multi-rate nature of the SR-SC multi-rate WMN. It is then shown through detailed performance evaluation that WCDS substantially improves ($\sim$ 3 to 6 times) the performance of those algorithms that do not exploit the rate-diversity offered in multi-rate WMNs. In addition, the significance of the product of a transmission's rate and its coverage area is demonstrated as a general rule-of-thumb that can predict the usefulness of a particular rate for broadcast in multi-rate WMNs.

Chapter 5: Distributed MLB Solution for SR-SC Multi-rate WMNs

In this chapter, three decentralized and localized rate-aware heuristics are proposed for the MLB problem in SR-SC multi-rate WMNs. We propose a connecting dominating set (CDS) based broadcast routing approach which calculates the set of forwarding nodes and the transmission rate at each forwarding node independent of the broadcast source; the forwarding tree is thereafter constructed from the CDS while taking into consideration the broadcast source. The performance comparisons of our centralized MLB algorithms (Chapter 4) and our distributed MLB algorithms (Chapter 5) point out a performance gap that is not large. It is demonstrated that the distributed algorithms (like the centralized algorithms of Chapter 4) also greatly improve performance by incorporating rate-awareness into its design.

Chapter 6: Alternative Framework for SR-SC Multi-rate WMN

In this chapter, an alternative framework for the MLB problem for SR-SC multi-rate WMNs is provided, along with a broadcasting heuristic for this new framework. Unlike the assumption made for the work presented in Chapters 4 and 5 of a "fully multi-rate multicast" (FMM) framework in which nodes can adjust link-layer multicast transmission rate for each link-layer frame, another framework called "single best-rate multicast" (SBM) is studied that exploits the link-layer rate-diversity by enabling each WMN to decide, depending on its topological properties, a single transmission rate for all its link-layer data multicasts. The findings of our research show that although FMM returns impressive performance, employing SBM is attractive since it can eliminate some undesirable features of practical multi-rate Media Access Control (MAC) protocols. In this chapter, methods to determine the "best" link-layer transmission rate for the SBM framework are also proposed. Two heuristic broadcast solutions that use the SBM framework are presented. Simulation results indicate that SBM broadcast heuristics give comparable performance to FMM broadcast heuristics especially in dense networks.

Chapter 7: Summary of Results

This chapter concludes Part I of this thesis by presenting its main results and findings.

Chapter 4

Centralized MLB Solution for SR-SC Multi-rate WMNs

4.1 Introduction

The minimum latency broadcast (MLB) problem, discussed previously in Section 1.2, is the problem of minimizing the *'broadcast latency'* which is defined as the maximum delay between the transmission of a packet by the source node and its eventual reception by all the destinations. The MLB problem for single-radio single-channel (SR-SC) single-rate wireless networks has been earlier studied by Gandhi et al. [26]. According to the best of our knowledge, our work is the first study of the MLB problem for multi-rate WMNs. Some of the unique challenges introduced by multi-rate broadcast have earlier been mentioned in Section 1.3.

The main contributions of this chapter are detailed below:

1. *We propose a novel broadcast framework called FMM (Fully Multi-rate Multicast) in which WMN nodes can adjust link-layer multicast transmission rate for each link-layer frame.*

2. *We propose a FMM centralized MLB heuristic, which we call WCDS, for SR-SC multi-rate WMNs.* WCDS incorporates the rate-diversity of SR-SC multi-rate WMNs into its design and improves the performance of the single-rate broadcast scheme that always uses the lowest rate $\sim$ 3 - 6 fold.

3. *We propose a general rule-of-thumb, which we call RAP, which can be used to predict the broadcast efficiency of a particular rate.*

Chapter Outline: The rest of the chapter is organized as follows. After introducing the network, the transmission, and the interference models in Section 4.2, we describe the framework of our heuristic solution to the *NP-hard* MLB problem for SR-SC multi-rate WMNs in Section 4.3. We evaluate the performance of the provided heuristic solution in Section 4.5 for an idealized MAC, which makes the assumptions listed in Section 4.2.2, and for the practical 802.11b MAC, as simulated in the Qualnet [1] simulator, in Section 4.6. In the end, we will propose some fundamental design principles for multi-rate broadcast in Section 4.7. This chapter is concluded thereafter in Section 4.8.

4.2 Network model

In this section, the network model used for the study presented in this chapter is discussed. The transmission and interference models are presented in Section 4.2.1. To tackle the complex MLB problem, some simplifying assumptions particularly about the network's Media Access Control (MAC) protocol, as discussed in Section 4.2.2, have been made.

4.2.1 Transmission and interference model

The IEEE 802.11 standards for multi-rate transmissions specify that a packet is received correctly if the packet error rate (PER) for a 1000-byte frame is less than 10%. This means that the signal-to-interference and noise (SINR) ratio at the receiver must be greater than a threshold in order for a packet to be received correctly at a particular transmission rate. The SINR threshold is different for different transmission rates; since higher-rate modulation schemes employ denser signal constellation, the SINR threshold for a higher transmission rate is higher, and vice versa.

We consider a multi-rate system with b different rates $r_1 > ... > r_b$. The SINR threshold for rate r_i is σ_i with the property that $\sigma_i > \sigma_j$ if $r_i > r_j$. We assume that a constant transmission power P_t is used for all transmission rates. Further, we assume that the receive power P_r at a distance d be given by the following propagation model:

$$P_r = P_t \frac{1}{d^\theta} \tag{4.1}$$

where θ is the path loss exponent which takes a value between 2 and 4. Following [75], we define the interference-free transmission range $\bar{d_i}$ of a transmission rate r_i as the maximum distance that a packet is received correctly in the absence of interference. Let N denote the thermal noise power in the system. It can readily be shown that

$$\bar{d_i} = \frac{1}{\sigma_i^{\frac{1}{\theta}}} \left(\frac{P_t}{N} \right)^{\frac{1}{\theta}}. \tag{4.2}$$

This shows that the interference-free transmission range is a decreasing function of transmission rate.

Consider a situation where we have a transmitter, a receiver and an interferer. Let d be the distance between the transmitter and receiver, and u be the distance between the receiver and the interferer. In order for the receiver to receive a packet correctly at rate r_i, the distances d and u must satisfy

$$\frac{\frac{P_t}{d^\theta}}{\frac{P_t}{u^\theta}+N} \geq \sigma_i \tag{4.3}$$

$$\Rightarrow \quad u \geq \left(\frac{\frac{P_t}{N}}{(\frac{\bar{d}_i}{d})^\theta - 1}\right)^{\frac{1}{\theta}} \tag{4.4}$$

This shows that the amount of interference a receiver can tolerate is dependent on distance between the transmitter and receiver. In particular, if the receiver is at the interference-free transmission range $\bar{d}_i$ and intends to receive at rate r_i, then it cannot tolerate any interference at all since u will be infinity. The above interference model, which is similar to the physical model used in [76], has good spatial reuse property but will not be easy to use to study the problem of minimum broadcast latency in a multi-rate network.

For our work, we will use the packet reception model where a packet at rate r_i is received correctly if *both* of the following conditions are satisfied:

1. The distance between the transmitter and receiver is less than s_i where s_i is the transmission range of transmission rate r_i.

2. No transmitter within a (finite) distance u_i from the receiver is transmitting concurrently where u_i is the interference range of transmission rate r_i.

This model is similar to the receiver based model used by Jain et al. [76]. Since our aim is to minimize the packet delivery latency from the source to all receivers, a receiver based model will be able to tell us the best possible achievable latency. Moreover, we expect that for a well designed broadcast scheme, the number of concurrent transmissions in the same local area will be low and this will be confirmed by simulation in Section 4.5. Thus, the one interferer model used earlier will be applicable. In this case, the relationship between s_i and u_i will be governed by Equation (4.4). In order to improve spatial reuse, we require s_i to be strictly less than $\bar{d}_i$ so that u_i is finite. A possible choice is to require that

$$s_i = \frac{1}{\xi^{\frac{1}{\theta}}}\bar{d}_i \tag{4.5}$$

where $\xi > 1$. The corresponding value of u_i will then be

$$u_i = \left(\frac{1}{\xi}\frac{P_t}{N}\right)^{\frac{1}{\theta}} \tag{4.6}$$

Note that for this particular choice, the interference radius is independent of the transmission range. It is interesting to note that for $\xi = 2$, the interference radius given above is equal to the maximum interference radius derived in [75] assuming that the thermal noise term is negligible compared with the interference. The above derivation therefore justifies the use of a constant interference radius in our reception model. This is consistent with the fact that for a high transmission rate link, the receiver is closer and therefore the signal strength is higher but at the same time a higher SINR is required for correct packet reception, thus the amount of interference that each transmission rate can tolerate is almost the same.

4.2.2 Modeling assumptions

The network model also assumes the following:

1. Each node in the network is equipped with one radio, with all radios tuned to a common channel.

2. By adjusting the modulation scheme, a node can multicast at different data rates, with the transmission range being a decreasing function of the data rate. Let $s_{\max}$ denote the maximum transmission range. Also, we use a disc model for the transmission range[1].

3. A node's neighbors are all the nodes that can be reachable using the lowest possible transmission rate.

4. Let $n_1, ..., n_m$ be the neighbors of a node n, the maximum rates node n can use to reach these nodes independently are $r_1, ..., r_k$ respectively. If node n wants to multicast to $n_1, ..., n_m$ in one go, this can only be performed at a rate of $\min(r_1, ..., r_k)$ or lower.

5. We assume a binary interference model, as follows: If while a node k is receiving a frame, a node j within a radius $\kappa s_{\max}$ from node k transmits a frame, then the frame that k is receiving is assumed to be corrupted and lost. We assume that the interference range $\kappa s_{\max}$ is a constant independent of the transmission rate. We call κ the normalized inference range.

6. We assume an ideal MAC layer, as follows: Two nodes i and j can multicast at the same time if and only if node i's multicast does not interfere with the intended recipients of node j's multicast and vice versa.

[1] In addition to evaluating our proposed algorithm's performance in Section 4.5 for the idealized MAC (the assumptions for which are presented in Section 4.2.2), we will also evaluate their performance for the more realistic 802.11 MAC model of the Qualnet simulator [1] in Section 4.6.

7. We assume a centralized entity which schedules these multicasts so that, under the ideal MAC layer assumption, no two multicasts will interfere with each other.

8. Each node can broadcast the same packet up to $m_{\max}$ times, clearly to different subsets of its neighbors. $m_{\max} = 1$ corresponds to the conventional use of broadcast trees, where each node reaches all its child nodes in a single transmission.

We assume the FMM broadcasting framework in which WMN nodes can adjust link-layer multicast transmission rate for each link-layer frame. This is in contrast to conventional technique of always using the lowest link-layer rate for broadcasting. We do not assume that all 1-hop neighbors of a transmitting node must be reached in a single transmission. Resultantly, a transmitting node can perform multiple link-layer multicasts, at distinct link-layer rates, to send the same packet to all its 1-hop neighbors (as was demonstrated in the motivating examples in Section 1.3). This assumption stems from the fact that the 1-hop neighbors reachable at a higher link-layer rate might be a subset of all the 1-hop neighbors. We note that all the 1-hop neighbors are necessarily reachable if we utilize lowest link-layer multicast rate. We use the term "link-layer multicast" to denote a transmission where the transmitting node can choose its transmission rate and intended recipients (subset or all of the 1-hop neighbors). Subsequent reference to link-layer multicast should be interpreted as the same as presented now.

We will now show that the MLB problem in a SR-SC multi-rate WMN is a non-trivial problem by demonstrating its NP-hardness.

Theorem 1 *The MLB problem with possibly multiple number of transmissions per node in a SR-SC multi-rate WMN is NP-hard.*

Proof 1 *Gandhi et al. showed that the MLB problem for single-rate wireless networks is NP-hard and provided a polynomial-time approximation algorithmic solution. If it is assumed that each node can relay a packet at most once in a SR-SC multi-rate WMN, then our considered problem becomes a generalization of the problem of Gandhi et al. [26] to the multi-rate case. The MLB problem for SR-SC multi-rate WMNs, therefore, is at least NP-hard; this follows from the fact that the MLB in a single-rate WMN where each node can transmit at most once, which is a special case of this problem, is NP-hard.*

The NP-hardness result for the single-rate case is given in the work of Gandhi et al. [26]. The single-rate broadcast problem has been well studied, for other NP-hardness results (e.g. inapproximability), refer to the work of Elkin et al. [77]. Since the single-rate case is a special case of the multi-rate case, these results also apply.

4.3 Heuristic MLB solution for SR-SC multi-rate WMNs

As discussed in the previous section, since the MLB problem is NP-hard, determining optimal MLB tree is expensive in terms of processing especially for large networks. Therefore, we resort to heuristics to determine broadcast trees that return close to optimal performance without requiring extensive calculations. Broadly speaking, any heuristic algorithm for the MLB problem in SR-SC multi-rate WMNs must make three important decisions. *Firstly*, it has to decide whether a node should multicast. *Secondly*, the algorithm must decide the number of transmissions at each transmitting node and determine the neighboring nodes covered in each of these transmissions. *Lastly*, the multicast transmissions of all nodes must be scheduled and their transmission time decided while taking radio interference into account. It should be noted that these decisions are closely coupled, since a multicasting node can only multicast after it has received the packet and radio interference dictates that the multicasts be scheduled so that interfering multicasts do not take place at the same time.

In this section, we will propose a heuristic solution to create 'efficient' delivery trees for broadcast packets in a multi-rate mesh network. With the hardness of the problem in mind, our heuristic solution is decomposed into three logically independent steps that are discussed next.

1. *Topology Construction*: In this step, the aim is to compute a broadcast tree (or a spanning tree) T. This step decides the make-up of the broadcast tree, i.e. it identifies all nodes that would transmit and the children/parent relation between different nodes. We assume that a node can transmit multiple times at different rates regardless of the number of radio resources available at this stage. The actual decision on the number of distinct-rate transmissions at each node is deferred to the next step. The 'topology construction' algorithm should take into account the multi-rate nature of our problem and must exploit the wireless broadcast advantage (WBA) [78] afforded by the wireless medium.

2. *Downstream Multicast Grouping*: The tree construction stage proceeds by assuming that each node can perform multiple distinct-rate transmissions to cover its children. The grouping algorithm can improve performance by suppressing these extra transmissions unless these extra transmissions can actually reduce delay while considering the resource constraints due to limited channel and transceivers. The aim of the 'multicast grouping' algorithm is to precisely determine the rates and number of distinct-rate transmissions that each node should make. Intuitively, the rationale behind multiple transmissions is to allow faster transmission to the more *'critical'* child nodes (i.e., those nodes

that have leaf nodes with larger delivery latencies) at the expense of larger transmission latency to the other child nodes.

3. *Transmission Scheduling*: While the number of transmissions at each node and the parent/child relationship amongst different nodes have been determined in the first two steps, the exact timing of the various multicasts (especially relative to different branches of the tree) still needs to be determined. The 'transmission scheduling' step schedules all transmissions taking into account our problem's precedence and interference constraints. These constraints mean that a node can only multicast after receiving the packet, and interfering multicast transmissions cannot occur concurrently. We are conceptually assuming a centralized scheduler in the description of our framework[2].

Clearly, this decomposition of the overall optimization problem is not optimal. For example, it is only after the 'multicast grouping' stage that the multicast transmission sets are obtained, as well as the transmission rate associated with each link-layer multicast. Similarly, the 'multicast grouping' choice would also depend on the scheduling strategy. Ideally, the 'topology construction' and 'multicast grouping' stages should take place simultaneously. However, a joint optimization, as already noted earlier, is computationally infeasible except for trivially small mesh topologies.

Section Outline: We will now outline the structure of the remainder of this section. In the next subsection, we review prior related work done for the 'topology construction' stage and then present WCDS, a centralized MLB heuristic for SR-SC multi-rate WMNs, for the construction phase (in Section 4.3.1). This is followed by the broad algorithmic approach for the grouping phase (in Section 4.3.2), which conceptually takes a tree as input and determines the partitioning of child nodes into different subsets, each corresponding to a separate link-layer multicast. In the last subsection (Section 4.3.3), the scheduling heuristic is presented which takes into account the conflict graph of the underlying tree topology irrespective of the choice of algorithm for 'topology construction'.

4.3.1 Topology Construction

The aim of the 'topology construction' stage is to compute a broadcast tree T spanning all nodes in the network. The children/parent relation between different nodes is determined in this stage after the transmitting nodes have been decided.

We will first introduce some mathematical notation. The entire WMN is represented as a graph (V, E), with the WMN nodes forming the vertices and the edges representing the direct link between any two nodes. Accordingly, $(i, j) \in E$ denotes

[2] The performance of our heuristics would also be examined using decentralized 802.11 MAC using the Qualnet [1] simulator.

the direct unicast link between nodes i and j. Based on the distance between such a node pair, each link (i, j) can be associated with a transmission rate r_{ij}. Assuming fixed transmission power, the transmission rate $r_{ij} = 0$ if i and j are not one-hop neighbors, i.e. j cannot correctly receive a packet from i even if i transmits at the slowest rate. There are a total of k link-layer transmission rates. The latency of a link $l(u, v)$ is the transmission latency of the quickest transmission rate that can be supported between nodes u and v. The set $\mathcal{L} = \{l_1, l_2, ..., l_k\}$ denotes the set of transmission latencies of all possible k transmission rates.

Weighted connected dominating set (WCDS)

In this section, a 'topology construction' algorithm based on the concept of weighted connected dominating set (WCDS) is proposed. In the proposed approach, the heuristics proposed in literature for calculating minimum connected dominating set (MCDS) are extended for a multi-rate setting. The CDS of the nodes of the network, whose topology is represented by a graph $G = (V, E)$, is a connected subgraph of G spanned by the nodes of $V' \subseteq V$ such that every node in the network is at most one hop distant from a node in V'. Among all CDSs of graph G, the one with minimum cardinality is called a minimum CDS (MCDS).

Computing an MCDS in a unit graph is NP-hard [31]. The use of MCDS to achieve optimal flooding in a single-rate multi-hop wireless networks has been explored by Lim et al. [36] who have shown that the size of the optimal flooding tree (measured by the number of nodes performing broadcasts, not by broadcast latency) differs from the size of the MCDS by at most one. However, MCDS performs poorly in multi-rate mesh environments because it does not account for multi-rate links in the tree construction. Ideally, the CDS nodes should decide their transmitting rates in a smart manner such that the goal of exploiting WBA while maintaining transmission rate as high as possible is achieved. Recall that assuming fixed transmission power, higher transmission rates translate to reduced coverage thereby limiting opportunities to exploit WBA.

To extend MCDS to our multi-rate setting, we assume there are k different rates whose transmission latencies are given by $l_1, l_2, ..., l_k$. Let $N(x, l_i)$ denote the nodes that are reachable from a node $x \in V$ when it transmits using the link-layer rate having latency l_i. We define the minimum WCDS problem whose aim is to find a subset $Y = \{y_1, y_2, ...\}$ in V and the broadcast latency l_i (which are chosen from $l_1, l_2, ..., l_k$) for node $y_i \in Y$ such that

1. Every element of $V \backslash Y$ is in $\cup_{y_i \in Y} N(y_i, l_i)$
2. The set Y is connected.
3. The weighted sum $\sum_{y_i \in Y} l_i$ is minimal.

Algorithm 1 The WCDS algorithm

Input: $G, s, \mathcal{L} = \{l_1, \cdots, l_k\}$
$C = \{s\}, T = \emptyset$
while $(V \backslash C \neq \emptyset)$ **do**
  for $(c \in C)$ **do**
    for $(l \in \mathcal{L})$ **do**
      $f(c,l) = |N(c,l) \backslash C| \times \frac{1}{l}$
    end for
  end for
  $(\hat{c}, \hat{l}) = \arg\max_{c \in C, l \in \mathcal{L}} f(c,l)$
  $A \leftarrow N(\hat{c}, \hat{l}) \backslash C$
  $C \leftarrow C \cup A$
  $T \leftarrow T \cup (\cup_{a \in A}\{(\hat{c}, a)\})$
end while

Note that when there is only one transmission rate, the minimum WCDS is equivalent to the MCDS. We expect the solution to the minimum WCDS problem to be similar to optimal broadcast tree for the multi-rate scenario. We use a greedy algorithm, depicted in Algorithm 1, to obtain an approximation of the minimum WCDS. The algorithm starts by making the source node s eligible to transmit. It does this by moving s to the set C which keeps track of the nodes which have received the message already and are eligible to transmit. We say that a node is covered if it has already received a packet and is in the set C. Also, the set R denotes the set of all possible k transmission rates. For an eligible node c and rate $l \in \mathcal{L}$, the quantity $|N(c,l) \backslash C|$ is the number of "not-yet-covered nodes" that are reachable by a broadcast by node c at latency l. Thus, in each round of the algorithm, we choose the (c,l) combination that maximizes the rate of increase of not-yet-covered nodes, as measured by $|N(c,l) \backslash C| \times \frac{1}{l}$. This metrics reflects our desire to both include as many nodes as possible in a single transmission, yet keep the transmission rate high (even though a higher transmission rate implies a smaller range, and thus, a smaller set of covered nodes). The algorithm returns T which is the set of directed links in the broadcast tree. At the end of 'topology construction' step, the tree constructed might have some nodes performing multiple transmissions at different rates.

It is to be noted that our WCDS algorithm does not place any restrictions in this stage (the 'topology construction' stage) to stem extra transmissions on any node, and we choose any number of transmissions at a node which our greedy algorithm deems fit. The limited resources available, i.e., the fact that we have only a single-channel and single-transceiver (since we are considering SR-SC multi-rate WMNs) is accounted for in the next two stages.

To demonstrate the working of WCDS, a simple example topology, shown in Fig. 4.1, is employed. The broadcast source is represented by a green square marker,

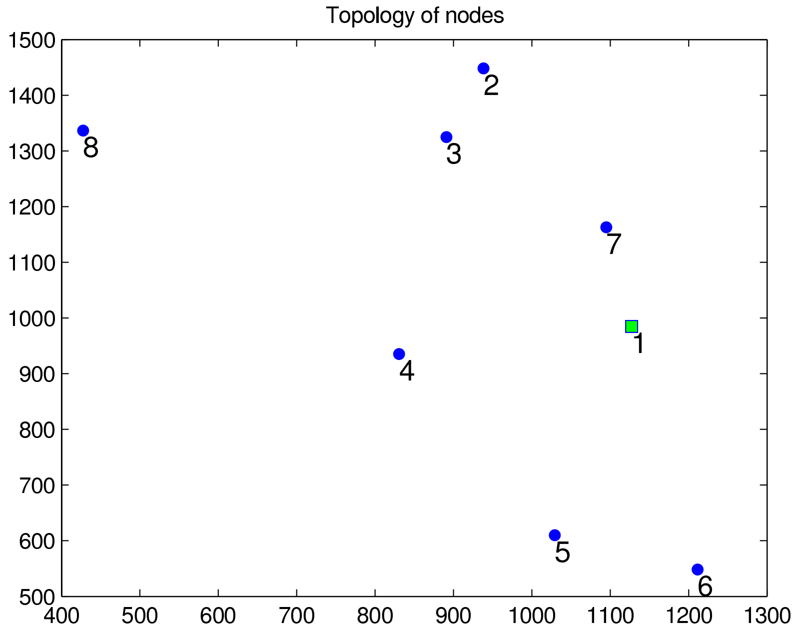


Figure 4.1: The sample topology

and the remaining nodes (the broadcast recipients) are represented by blue circular markers. The node number is written directly below the markers. The transmission latency for the transmission rates of 11, 5.5, 2 and 1 Mbps is normalized as 1, 2, 4, and 8, respectively. For our example topology, the choice of the (c, l) combination at the end of each successive round is (1,1), (7,1), (7,2), (1,8) and (3,8) in the order in which they are added. This illustrates that WCDS chooses, nodes 1, 7, 7, 1 and 8 and rates 11 Mbps, 11 Mbps, 5.5 Mbps, 1 Mbps and 1 Mbps in successive rounds in this particular order. These combinations (c, l) are chosen because the metric $f(c, l)$ for these combinations (i.e., 1, 1, 1, 0.25 and 0.125 respectively) is the maximum at the end of their respective round. The WCDS tree after the 'topology construction' is shown in Fig 4.2. The node numbers are written directly below the markers. The line between two nodes represent a transmission, with the link's latency written on the middle of the line.

4.3.2 Multicast Grouping

We recall from the discussion at the beginning of this section that to find the topology minimizing the broadcast latency, we must make a number of decisions including: 1) which node is to multicast, 2) how many times must the nodes multicast, 3) who are the recipients of these multicasts, and 4) what are the timing of these multicasts. The result of the 'topology construction' is a broadcast tree which specifies that the non-leaf nodes of the broadcast tree will multicast to its child nodes, in possibly multiple transmissions. However, the number of times a transmitting node (i.e. non-leaf node of the broadcast tree) will multicast and the recipients of each multicast still have not been decided. In case where a node multicasts only

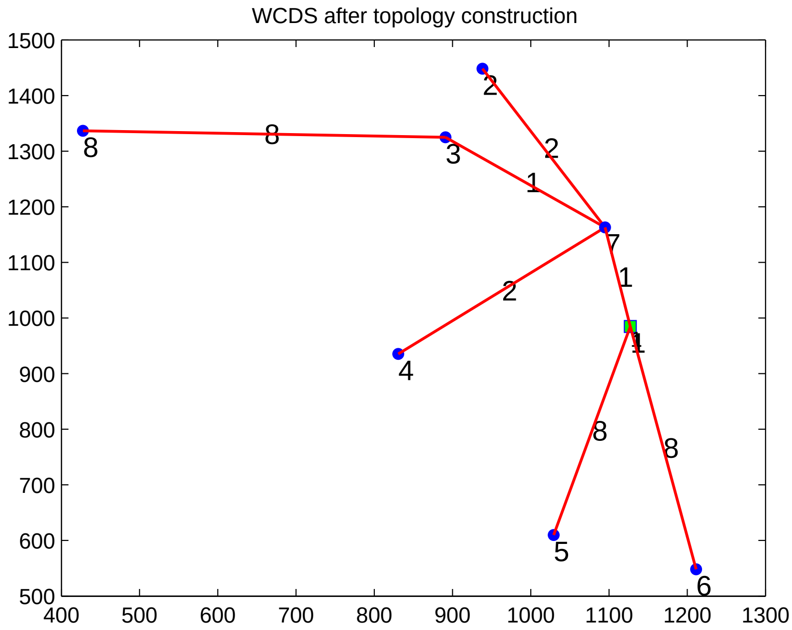


Figure 4.2: WCDS after 'topology construction'

once, then the recipients will be all its child nodes. For the case where a node is to multicast more than once, a different subset of child nodes will be reached in each multicast such that these subsets all together form the set of child nodes. The aim of the 'multicast grouping' stage is to determine the number of multicasts to be made and the recipients of each multicast.

We begin by defining the concept of valid transmission sequence at a transmitting (i.e. non-leaf) node of the broadcast tree. Consider for example a transmitting node n which has two child nodes c_1 and c_2, which can be reached using a minimum latency of $d_1(=1)$ and $d_2(=2)$ time units respectively. Node n can reach these nodes in a number of valid transmission sequences. For example, it can first multicast to c_1 (with latency 1) followed by another multicast to c_2 (with latency 2). We will denote this valid transmission sequence as (d_1, d_2). An alternative valid transmission sequence for node n is (d_2) which reaches both nodes in one multicast. These two are the only two valid transmission sequences for this example. The sequence (d_1) is invalid because it does not reach all the child nodes. In addition, (d_2, d_1) is invalid because the second transmission is unnecessary since both nodes are already reached by transmission d_2 whose coverage area is greater. In general, consider a transmitting node n which has m child nodes $c_1, ..., c_m$ that are reachable using minimum latency of $d_1, .., d_m$ respectively. Let k denote the number of distinct latencies in $d_1, .., d_m$ and let us denote these distinct latencies as $\mathcal{L} = \{l_1, ..., l_k\}$. Without loss of generality, we assume that $l_k \geq ... \geq l_1$. A valid transmission sequence is a r-tuple $(1 \leq r \leq k)$ whose entries are drawn from $\mathcal{L}$ such that:

1. Each latency in $\mathcal{L}$ appears in the r-tuple at most once.

2. The latencies in the r-tuple appear in a strictly increasing order.

3. The last entry of the r-tuple must be l_k.

Let $T_V(n)$ be the set of all valid transmission sequences for node n. Since node n uses k distinct rates to reach its child nodes, T_V contains 2^{k-1} valid transmission sequences.

Since our goal is to minimize the broadcast latency, we are interested to find the valid transmission sequence at all the transmitting nodes such that they together will minimize the broadcast latency. For ease of reference, we will refer to the optimal valid transmission sequence at a transmitting node as the *Cardinal Sequence* (CS). Also, if a transmitting node n and all its descendants use their cardinal sequences for transmission, the delay it takes a packet to reach all n's descendants will be called node n's *Cardinal Value* (CV). The aim of the 'multicast grouping' stage is to find the CS and CV at each transmitting node of the network.

Since the choice of CS and CV at a transmitting node n depends on the CS's and CV's of all the transmitting nodes who are descendants of n, the grouping algorithm should proceed from the leaf nodes of the broadcast tree back to the root. For the rest of the description, we will show how the CS and CV of an arbitrary transmitting node n can be determined. We assume that the CS's and CV's of all the transmitting nodes who are descendants of n are already known. Also, for initialization, we define the CV of all leaf nodes to be zero.

If we assume that node n uses k distinct transmission rates to reach its child nodes, then the set of all valid transmission sequences at node n, denoted by $T_V(n)$, has 2^{k-1} valid transmission sequences S_q ($1 \leq q \leq 2^{k-1}$). The CS at node n is determined by comparing the broadcast latency achieved by all possible $S_q \in T_V$ and then choosing the S_q with the least broadcast latency as the CS. The CV of the node is then the latency associated with the chosen CS. If node n uses the transmission sequence S_q, let $D(n)_{S_q}$ denote the resulting latency required to reach all the descendants of n, we can formally define CS and CV of node n as

$$CS(n) = \arg \min_{S_q \in T_V} (D(n)_{S_q}) \tag{4.7}$$

$$CV(n) = \min_{S_q \in T_V} (D(n)_{S_q}) \tag{4.8}$$

We will now explain in detail how $D(n)_{S_q}$ can be computed. Let S_q be the r-tuple $(S_{q,1}, ..., S_{q,x}, ..., S_{q,r})$. Since the coverage area of a higher latency transmission is larger, thus with the transmission sequence $S(q)$, some of the child nodes of n will receive the same packet multiple times. In particular, let $N(n)_{S_{q,x}}$ denote the child nodes of n that are reachable by a multicast of latency S_{qx} but are not reachable by

$S_{q,x-1}$. In other words, the nodes in $N(n)_{S_{q,x}}$ receive their packets from n for the first time via a multicast of latency $S_{q,x}$ and will receive the same packet a total of $(r-x+1)$ times. Note also that the sets $N(n)_{S_{q,x}}$ $(x = 1, ..., r)$ effectively partition the child nodes of n into r disjoint subsets. Let $D(n)_{S_{q,x}}$ denote the delay it takes n to reach all the nodes in the set $N(n)_{S_{q,x}}$ and their descendants. Assuming that the transmission of the descendants of $N(n)_{S_{q,x}}$ do not interfere with each other, we have

$$D(n)_{S_q} = \max_{1 \le x \le r} D(n)_{S_{q,x}} \tag{4.9}$$

As mentioned a number of times before, the decisions we need to make are highly coupled. Thus, by ignoring the inter-branch interference, we obtain an approximation which makes the problem tractable. The inter-branch interference will be taken into account in the scheduling stage in Section 4.3.3.

We propose to compute $D(n)_{S_{q,x}}$ using the following formula:

$$D(n)_{S_{q,x}} = \sum_{i=1}^{x} S_{q,i} + \max_{i \in N(n)_{S_{q,x}}} CV(i) + \sum_{i=1}^{x-1} SCDelay_{(S_{q,i})} \tag{4.10}$$

This equation is obtained by assuming the following *modus operandi*: Node n first transmits at latency $S_{q,1}$ reaching the nodes in $N(n)_{S_{q,1}}$. If some of the nodes in $N(n)_{S_{q,1}}$ are transmitting nodes, they will then begin their transmission to their respective downstream neighbors in parallel. (Note that we are again ignoring inter-branch interference). Note that node n does not begin transmitting at latency $S_{q,2}$ immediately after finishing transmitting at $S_{q,1}$. We assume that node n waits until all the transmissions from $N(n)_{S_{q,1}}$ and their descendants have proceeded sufficiently so that the $S_{q,2}$-transmission of node n does not interfere with those of $N(n)_{S_{q,1}}$ and their descendants. This operation then repeats itself until all transmissions in S_q have been made.

With this *modus operandi* in mind, we can now explain how Equation (4.10) comes about. We begin with the case for $x = 1$ where we have $D(n)_{S_{q,1}} = S_{q,1} + \max_{i \in N(n)_{S_{q,1}}} CV(i)$. Recall that $D(n)_{S_{q,1}}$ is the delay it takes to reach all the nodes in $N(n)_{S_{q,1}}$ and their descendants. The first term $S_{q,1}$ is simply the time it takes to reach the nodes in $N(n)_{S_{q,1}}$. After the packets have been received by the nodes in $N(n)_{S_{q,1}}$, we assume that the transmissions by the nodes in $N(n)_{S_{q,1}}$ will proceed in parallel, so the maximum time it takes all these transmissions to reach the end of their branches is given by the second term. Note that this follows from our definition of CV.

We now explain the derivation of Equation (4.10) for $x > 1$. The first two terms of the equation bear similar meaning to what is explained in the last paragraph, so we will focus on the third term only. Recall from our description of the *modus operandi* that the $S_{q,x}$-transmission of node n will only begin after the downstream transmissions caused by the $S_{q,x-1}$-transmission have proceeded sufficiently. The time gap between these two transmissions by node n is $SCDelay_{(S_{q,x-1})}$. Here the prefix "SC" stands for single-channel as this delay is caused by the fact that we have only one single-channel in the system.

Recall that the time $SCDelay_{(S_{q,x-1})}$ is needed so that the $S_{q,x}$-transmission is not interfered by the transmissions by the nodes in $N(n)_{S_{q,x-1}}$ and their descendants. In order to compute $SCDelay_{(S_{q,x-1})}$, we will first need to identify those transmissions which may interfere with the reception of the nodes in $N(n)_{S_{q,x}}$. Let $\mathcal{T}_{S_{q,x-1}}$ be all transmitting nodes in $N(n)_{S_{q,x-1}}$. Let $\tilde{t} \in \mathcal{T}_{S_{q,x-1}}$, the set $\mathcal{N}(\tilde{t})$ consists of all nodes $\tilde{n}$ with the following properties: 1) $\tilde{n}$ is a descendant of $\tilde{t}$, 2) the transmission of the parent of $\tilde{n}$ interferes with the reception of nodes in $S_{q,x}$, and 3) either $\tilde{n}$ is a leaf node or the transmission of $\tilde{n}$ and its descendants do not interfere with the reception of nodes $S_{q,x}$. In other words, the transmissions in $\mathcal{N}(\tilde{t})$ are the first ones that do not interfere with the $S_{q,x}$-reception. Thus, we have:

$$SCDelay_{(S_{q,x-1})} = \max_{\tilde{t} \in \mathcal{T}_{S_{q,x-1}}} \left(CV(\tilde{t}) - \min_{\tilde{n} \in \mathcal{N}(\tilde{t})} CV(\tilde{n})\right) \tag{4.11}$$

The term in parenthesis in the above equation essentially estimates the time it takes the transmissions due to $\tilde{t}$ and its descendants to clear the interference range of the nodes in $N(n)_{S_{q,x}}$.

Having examined how $D(n)_{S_{q,x}}$ (Equation (4.10)) was obtained, we can see how equations (4.7), (4.8), (4.9), and (4.10) can be used together to obtain the CV at a transmitting node. This process can be performed recursively starting from the leaf nodes back to the root of the broadcast tree.

In addition to deciding on the transmission sequence at each transmitting node, the results of the above computation will also be helpful in deciding the timing of the transmissions in the scheduling stage. Recall that the CV of a transmitting node n can be interpreted as the time required to reach all the descendants of node n. Thus, when it comes to scheduling all the multicast transmissions that are to be made, we can use the analogous concept of the CV of a transmission as a measure of the urgency of the transmission. If S_{q_x} is a transmission within the CS of node n (i.e. S_{q_x} is a chosen transmission), then the CV of S_{q_x} is in fact given by Equation (4.10).

Following on the example in Section 4.3.1 where we applied the WCDS 'topology construction' algorithm to the example network shown in Figure 4.1. The result of applying the 'multicast grouping' algorithm to the WCDS broadcast tree is showed in Figure 4.3. It shows that Node 1 uses multiple transmission rates to reach its child nodes.

We can compute the optimal 'multicast grouping' by using the well known branch and bound technique; i.e., given that the parent-children relationship between various nodes has been decided in the 'topology construction' stage, we can use the branch and bound technique to find out how many distinct-rate transmissions should

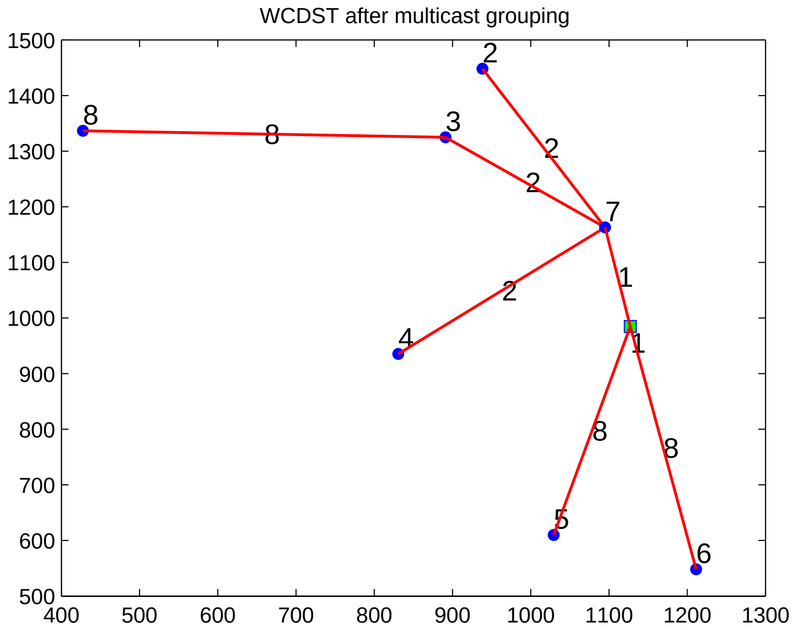


Figure 4.3: WCDS after 'multicast grouping'

a parent node use to reach all its children such that optimal broadcast latency results (for that particular parent-children relationship decided in 'topology construction'). Figures 4.4 and 4.5 show the optimal tree for the cases where each node can transmit the same packet at most, respectively, one and two times; these optimal trees have been calculated using the integer programming formulation of the MLB optimization problem for SR-SC multi-rate WMNs that can provide an optimal solution for very small sized networks [79]. Incidentally, by comparing Figures 4.3 and 4.5, we see that WCDS tree after 'multicast grouping' is the same as optimal tree (when multiple transmission are allowed) for this example.

4.3.3 Transmission Scheduling

After the completion of 'topology construction' and 'multicast grouping' stages, the transmitting nodes along with the number and rate of their distinct-rate transmissions have all been decided. However, the exact timings of these transmissions is decided in the 'transmission scheduling' stage. We approach the scheduling problem by formulating it with precedence constraints which enforces that a node can only multicast after it has received the packet. The interference between different transmissions is represented by a conflict graph. Let $V_b = \{b_1, b_2, ..., b_k\}$ be the set of all the multicast transmissions decided by the 'multicast grouping' algorithm in Section 4.3.2. Each multicast transmission b_i have four attributes: (1) A sender (which is a non-leaf node of the broadcast tree). (2) A group of recipients (which is a subset of the child nodes of the sender). (3) The latency required by the trans-

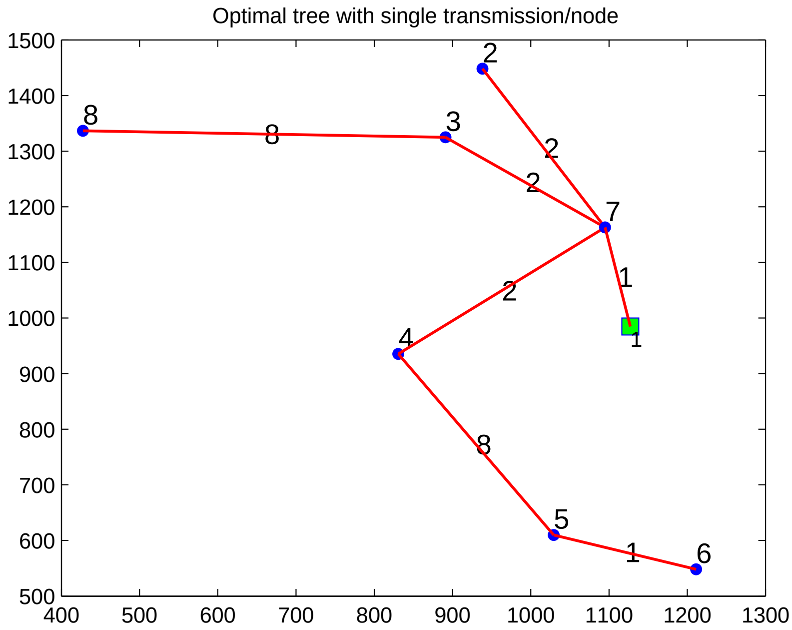


Figure 4.4: Optimal tree with single transmission/node

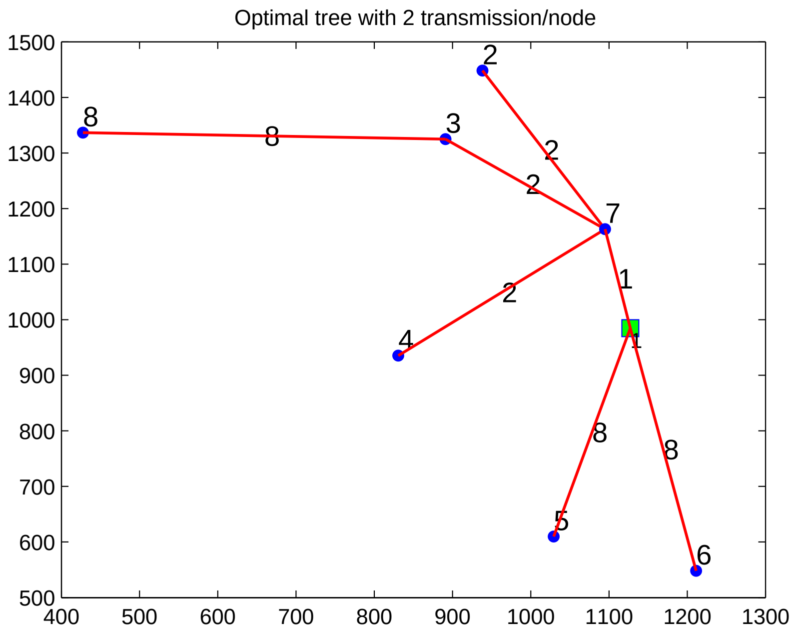


Figure 4.5: Optimal tree with multiple transmission/node

mission, denoted by $t(b_i)$, which is the minimum latency it takes the sender to reach all its designated recipients. (4) The CV value of a transmission as defined at the end of Section 4.3.2. Since the CV value of transmission measures the time it takes a packet to reach the end of the tree, it is viewed as an urgency measure by the scheduling algorithm.

In addition, we define an undirected conflict graph $G_c = (V_c, E_c)$ such that

$V_c = V_b$ and $(b_i, b_j) \in E_c$ if and only if (1) The multicast of b_i interferes with the reception of the recipients in b_j or vice versa; or, (2) Both multicasts b_i and b_j have the same sender.

Formally, a schedule can be defined as a mapping $\tau : V_b \rightarrow \mathbb{R}$ which gives the transmission starting time of $b_i \in V_b$. A valid schedule is one which meets the following constraints:

1. The source node multicasts at time zero.
2. A node can only multicast after it has received the packet: if the sender of b_j is a recipient of b_i, then $\tau(b_j) \geq \tau(b_i) + t(b_i)$
3. For any edge $(b_i, b_j) \in G_c$, we have $(\tau(b_i), \tau(b_i) + t(b_i)) \cap (\tau(b_j), \tau(b_j) + t(b_j)) = \phi$. Note that $(\cdot, \cdot)$ here also denotes an open interval in $\mathbb{R}$. Although the same notation is used to denote both an open interval and an edge of a graph, the usage should be clear from the context.

The scheduling algorithm is depicted in Algorithm 2. The input to the algorithm is transmissions information (TX) which contains the attributes discussed earlier. The aim of the scheduling algorithm is to find out the starting time (τ) and ending time (δ) of all transmissions at each transmitting node.

Initially, *time* depicting current running time is initialized to zero and E depicting *eligible transmissions* is initialized with all transmissions of the source node. A transmission is said to be eligible when the node performing this transmission receives the multicast from its parent, all transmissions of the source node are eligible at time 0. The scheduling process starts by scheduling the lowest latency transmission of the source node at time 0. This transmission is added to the set T which contains all transmissions currently being performed. The starting time (τ) and ending time (δ) of transmissions are decided as they are added to T or in other words as they start transmitting. The minimum of $\delta(\forall t \in T)$ is the earliest any transmission in T will finish and also the earliest a waiting eligible transmission can be scheduled and is called the next-stop time.

At the next-stop time, since the channel becomes available again due to completion of some transmission, a new transmission must be slotted for transmission. The transmission $t \in E$ having the maximum transmission CV is determined, and is assumed to be more *'critical'* as it connects to sub-trees of higher broadcast delay. Thereafter, it is checked that t does not interfere with any of the transmissions in T. In case of no interference, t is added onto T and deleted from E. The starting time $\tau(t)$ and ending time $\delta(t)$ for the transmission t is also decided at this time. However in case t interferes with any existing transmissions in T, it is held back until next-stop time. It is also ensured that a high-rate transmission does not follow a low-rate transmission at the same node.

After we have iterated through all eligible transmissions i.e. all t belonging to E, the next-stop time is found out by determining which transmission is going to finish the earliest. At the next-stop interval, the children nodes of the transmission finishing at next-stop interval receive the message and thus are eligible for transmitting. Thus at next-stop interval, the transmissions of these recently eligible nodes are added to the eligible transmissions E alongside those transmissions which were held back in the last round. We abide by the precedence constraint in this manner i.e. by allowing a transmission to be added to E only after the transmission has been enabled. A transmission is considered to be enabled when the node making the transmission has received from its parent. At the next-stop interval, all transmissions which are finishing are deleted from T. The algorithm runs in rounds and finishes when the starting time for all transmissions $\tau(\forall t \in V_b)$ and ending time for all transmissions $\delta(\forall t \in V_b)$ have been decided.

4.4 Maximum end-to-end throughput

The above discussion of the tree construction and scheduling algorithms focused on the case of a *single* packet, attempting to minimize the broadcast latency for a single packet. This approach is clearly directly applicable when the data rate of the broadcast stream is low enough (e.g., for control traffic), where one can safely assume the absence of interference/scheduling conflicts among successive packets of the same flow. For higher rate data flows, it is important to compute the maximum achievable throughput of a broadcast tree, defined as the maximum data rate that can be sustained without having any scheduling-related conflicts between packets of the same flow.

The maximum achievable throughput can be computed from the packet transmission schedule computed in Section 4.3.3. Using the same notation as in Section 4.3.3, the set of all multicast transmissions are $V_b = \{b_1, b_2, ..., b_k\}$ and the schedule says that transmission b_i will take place during the time interval $[\tau(b_i), \tau(b_i)+t(b_i)]$. Assuming that packets are generated by the source node at regular time at $(m-1)\Delta$ (for $m = 1, 2, ...$). Our goal is to maintain the same schedule computed earlier so node b_i is expected to multicast the m-th packet during $[(m-1)\Delta+\tau(b_i), (m-1)\Delta+\tau(b_i)+t(b_i)]$. The maximum throughput is achieved by the smallest possible Δ such that there is no conflict between the scheduling of all the packets. By defining

$$\mathcal{I}(m, b_i) \quad = \quad ((m-1)\Delta + \tau(b_i), (m-1)\Delta + \tau(b_i) + t(b_i)), \tag{4.12}$$

we can formally express the above problem as:

$$\min \Delta \;\; s.t. \;\; \mathcal{I}(m_1, b_i) \cap \mathcal{I}(m_2, b_j) = \phi \;\; \forall m_1, m_2 = 1, 2, ... \text{ if } (b_i, b_j) \in G_c \tag{4.13}$$

Algorithm 2 Algorithm for the 'Transmission Scheduling' stage

Input: TX { TX contains information about the to-be-scheduled transmissions}
Set $time = 0$
Initialize $E \leftarrow \cup_{TX.Node=n_s}\{TX\}$
Initialize $T = \emptyset$
while ($E \neq \emptyset$ or $T \neq \emptyset$) **do**
  while $E \neq \emptyset$ **do**
    $t = \arg\max_{TX \in E} TX.CV$
    $E = \{E \backslash t\}$
    if $|T| >= 1$ **then**
      if $TX(t).node$ and $TX(\cup_{t' \in T \backslash t}\ \{t'\}).node$ do not interfere **then**
        $T \leftarrow \{T \cup t\}$;
        Set $\tau(t) = time$
        Set $\delta((t) = time + TX(t).latency$
      else
        $E_{Next} \leftarrow t$
      end if
    else if $|T| < 1$ **then**
      $T \leftarrow \{T \cup t\}$;
      Set $\tau(t) = time$
      Set $\delta(t) = time + TX(t).latency$
    end if
  end while
  $NextStop$= min $(\delta(\cup_{t \in T}\{t\}))$
  $NextTrans$= $\{t\}$: $(\forall t)$ $(\delta(t)$= $NextStop)$
  $E \leftarrow E \cup t_{children}$ of $NextTrans$
  $T = T - NextTrans$
  $E = E - NextTrans$
  $E = E \cup EN$
  $time \leftarrow NextStop$
end while
Output: $\tau(t), \delta(t)\ \forall_{(1 \leq t \leq |TX|)}$

Note that G_c is the conflict graph defined in Section 4.3.3. Since the schedules repeat themselves periodically, it is sufficient to examine possible conflicts in $[0, T_{\max}]$ where $T_{\max}$ is the broadcast latency. Thus, we can simplify the problem to

$$\min_{\Delta \in [0, T_{\max}]} \Delta \ \ s.t. \ \ \mathcal{I}(1, b_i) \cap \mathcal{I}(m, b_j) = \phi \ \ \forall m = 1, 2, ... \text{ if } (b_i, b_j) \in G_c \tag{4.14}$$

Assuming two transmissions b_i and b_j do interfere with each other, the constraints in equation (4.14) can alternatively be expressed as:

$$(m-1)\Delta + \tau(b_j) \notin (\tau(b_i) - t(b_j), \tau(b_i) + t(b_i)). \tag{4.15}$$

The left-hand-side of the above expression is the start transmission time of the m-th packet by transmission b_j and it must not lie in the time interval given on the right-hand-side in order to avoid conflict. This expression also means that Δ cannot take certain values. Thus by identifying all the values that Δ cannot take within $[0, T_{\max}]$, we can easily find the optimal value of Δ. This computation method is similar to domain reduction in constraint logic programming [80]. This algorithm can find the optimal Δ in polynomial time.

4.5 Performance evaluation using idealized MAC

For all the results presented in this section, we are assuming an idealized MAC (Section 4.2.2) along with the scheduler described in Section 4.3.3. The rate-range relationship for 802.11b network from Qualnet simulator is assumed (Table 1.1). We will compare the performance of our algorithm WCDS with three other heuristics. All these four heuristics have the same structure, they initially perform 'topology construction', and follow it by 'multicast grouping' (Section 4.3.2) and 'transmission scheduling' (Section 4.3.3). These algorithms only differ in how the broadcast tree is computed (i.e., only in the 'topology construction' stage). The algorithms that we consider are:

1. Algorithm BIB: It uses an existing 'topology construction' heuristic for SR-SC multi-rate WMNs called BIB that was presented by Chou et al. [81].

2. Algorithm WCDS: It uses WCDS (Section 4.3.1) during the 'topology construction' stage to compute the broadcast tree.

3. Algorithm SPT: It uses the Dijkstra's algorithm to calculate the broadcast tree while assuming that there is only a single radio interface and channel.

4. Algorithm CDS: This heuristic assumes that all broadcasts are done at the lowest transmission rate. The broadcast tree can be computed by using WCDS in Section 4.3.1 with only the lowest rate allowed.

It has been observed that it is difficult to obtain an optimal solution for the MLB problem in SR-SC multi-rate WMNs even for a network of 20 nodes [79]. Therefore, to get a lower-bound of broadcast latency for a network, we use the broadcast latency of Dijkstra's tree, while assuming that the Dijkstra's tree can utilize unlimited number of radios and channels, and each node can perform unlimited number of distinct-rate transmissions. We normalize the results of our heuristics with this lower-bound. We note that the optimal tree for a SR-SC multi-rate WMN (that has the lowest possible broadcast latency for that WMN) might not be able to match the performance of Dijkstra's tree since we have (unrealistically) assumed unlimited radio resources for the Dijkstra's tree. Thus, the Dijkstra's tree latency is not a strict lower-bound, but an approximate of the optimal performance. We note that the minimum possible value of normalized delay is unity.

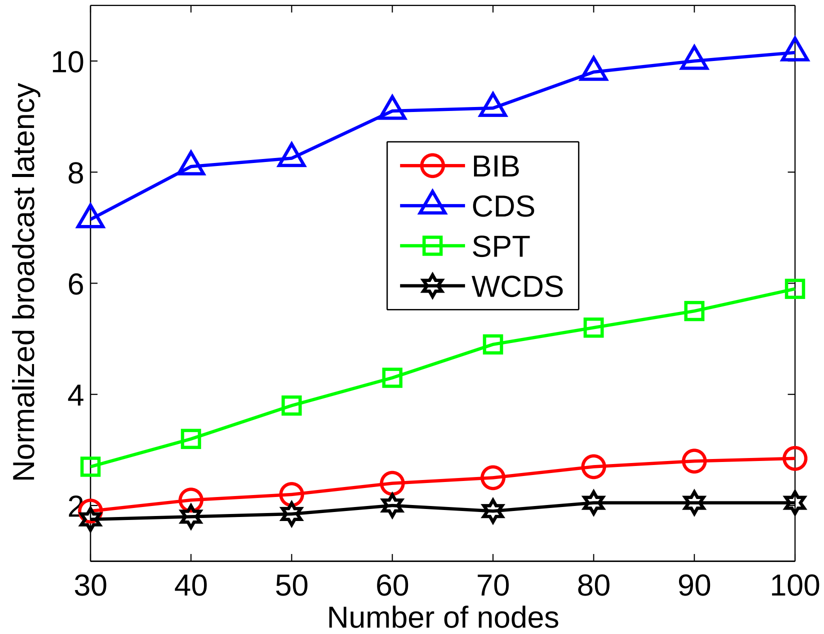
We compare the broadcast performance of our algorithms using 100 topologies of different network sizes (measured by the number of nodes) uniformly randomly distributed in an area of 1000^2 m^2. We will use the geometric mean, over 100 network instances of a fixed size, of the normalized delay and the throughput in our results. We have assumed that the interference range is κ times the lowest transmission rate's transmission range (Section 4.2.2). We refer to κ as the normalized interference range, and assume it to be 1.7 (as used by Xu et al. [75]) unless otherwise specified. We have assumed a maximum of one link-layer transmission per node for Sections 4.5.1, 4.5.2, 4.5.3, and 4.5.4. We demonstrate the possible improvement in performance by exploiting the degree-of-freedom of multiple distinct-rate transmissions per node in Section 4.5.5.

4.5.1 Broadcast latency and throughput performance

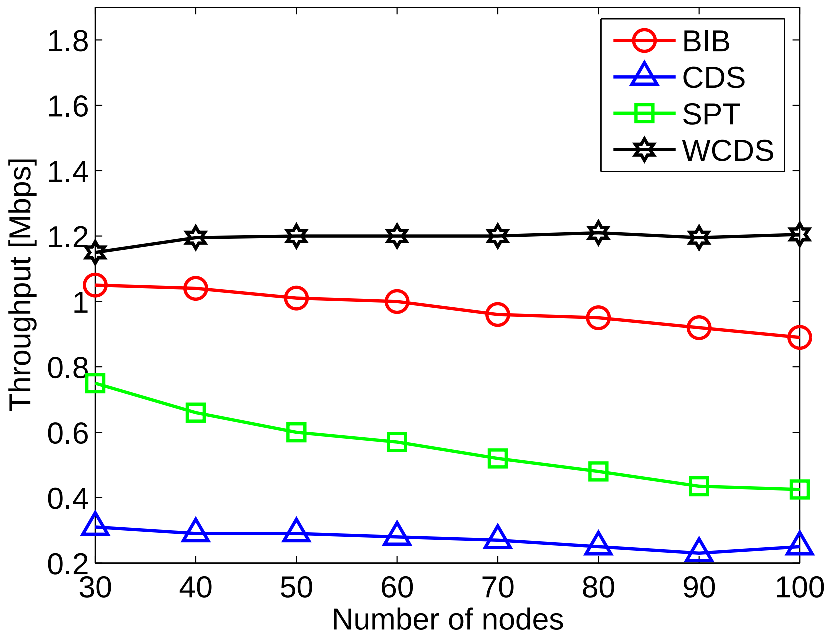
The broadcast latency and throughput performance results for our considered heuristics (algorithms BIB, CDS, SPT, and WCDS) are shown in Figures 4.6(a) and 4.6(b) for varying number of nodes. It turns out that good performance for delay also means good performance for throughput and vice versa. It is seen that Algorithm CDS, in which the all transmissions always take place at the lowest transmission rate, performs the worst of our considered algorithms for both the latency and throughput performance; this is due to the fact that although CDS exploits WBA, its performance is significantly impaired since it does not exploit the link rate-diversity. Interestingly, algorithm SPT, which exploits link rate-diversity but not the WBA, performs better than CDS for both latency and throughput results; this seems to indicate that although performance can benefit from exploiting both WBA and link rate-diversity, returns from exploiting link rate-diversity are more impressive compared to benefits of exploiting WBA. Not surprisingly, both algorithms BIB and WCDS outperform CDS and SPT since both BIB and WCDS exploit the link rate-diversity as well as the WBA. The latency and throughput performance of algorithm WCDS is better than that of algorithm BIB, as shown in Figures 4.6(a) and 4.6(b), which indicates that the WCDS algorithm is more adept at exploiting the link rate-diversity and WBA.

4.5.2 Effect of increasing network node densities

The effect of varying network's node density on the broadcast latency, the throughput, and the number of transmissions for our considered heuristics can be seen in Figures 4.6(a), 4.6(a), and 4.6(c) respectively. We have assumed a network area



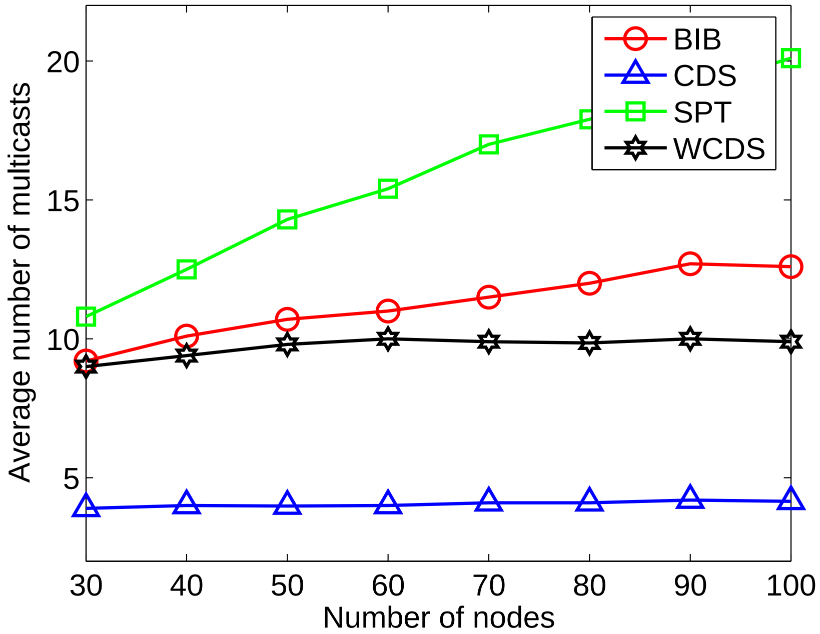
(a) Mean normalized broadcast latency



(b) Mean throughput

Figure 4.6: Performance of algorithms BIB, WCDS, SPT and CDS against varying number of nodes

of size 1000^2 m^2. It is seen that the performance of algorithms SPT, BIB is the most sensitive to increasing network density; with increasing node density, the mean broadcast latency and the number of total transmissions increase, while the mean throughput decreases. It is observed for the CDS and the WCDS algorithms, both of which are adept at exploiting WBA, that the mean number of transmissions remains



(c) Mean number of multicasts

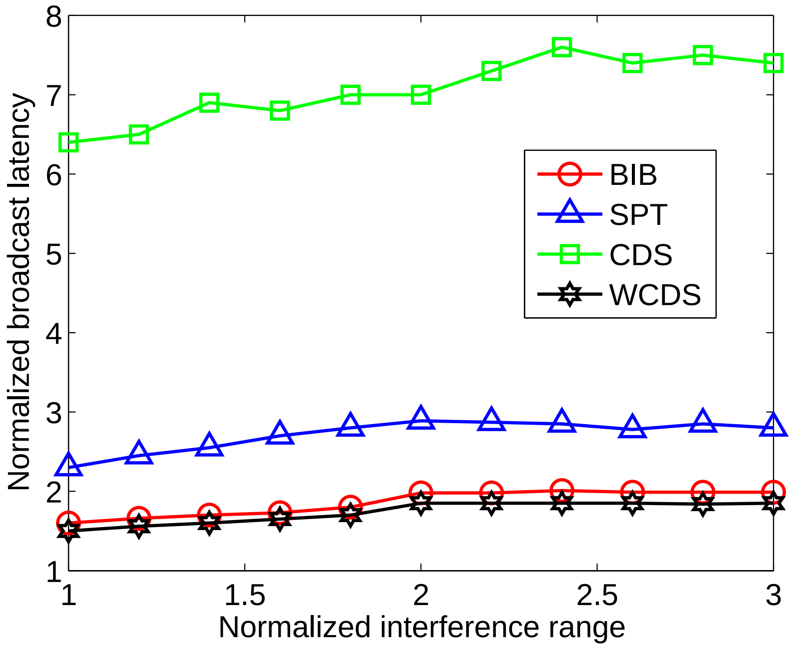
Figure 4.6: (cont.) Performance of algorithms BIB, WCDS, SPT and CDS against varying number of nodes

nearly constant for the whole of our considered range of 30 to 100 nodes per area of 1000^2 m^2. This demonstrates that the CDS and WCDS algorithms can maintain broadcast latency performance for increasing number of nodes in a fixed network area by exploiting the WBA and covering more nodes per transmission.

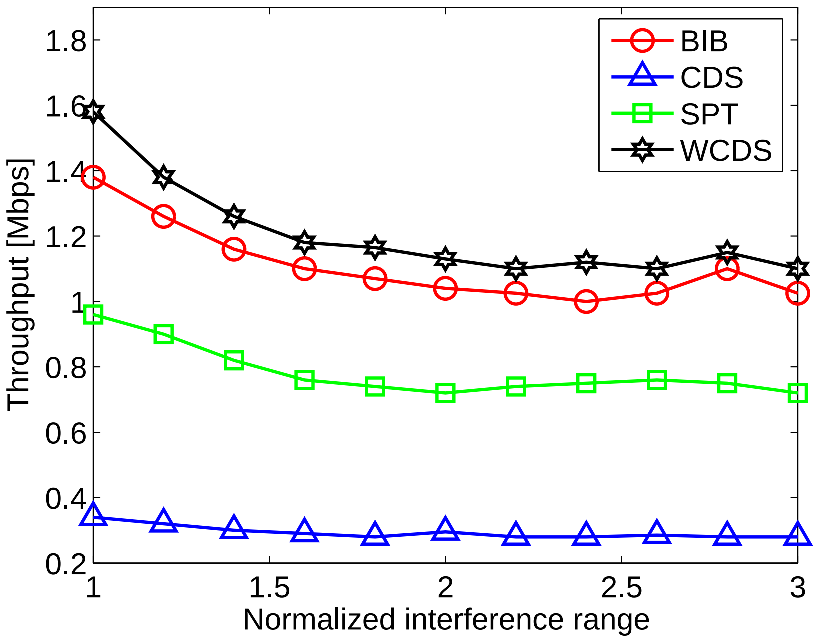
4.5.3 Total transmissions for different algorithms

It is observed that a trade-off exists between improved results for performance metrics like throughput and latency, and reduced number of transmissions. As can be observed from Figure 4.6(c), the algorithm requiring the least number of transmissions is CDS which also yields the worst broadcast latency and throughput performances (see Figures 4.6(a) and 4.6(b)). Since CDS always utilizes the lowest transmission rate, it employs lesser transmissions compared to algorithms that can employ higher rates[3]. On the other end of the spectrum, the SPT algorithm typically performs the most transmissions since it does not take the WBA into account. The algorithms BIB and WCDS manage to reconcile high performance with limited number of transmissions. The WCDS algorithm, not only returns the best broadcast latency and throughput performance of the compared algorithms, but also requires low number of transmissions (second to the CDS algorithm as shown in Figure 4.6(c)).

[3]Typically, the range of a lower rate transmission is greater than for higher rate transmissions



(a) Mean normalized broadcast latency



(b) Mean throughput

Figure 4.7: Impact of interference range on normalized broadcast latency (top) and broadcast throughput (bottom) for BIB, WCDS, SPT and CDS for a network of 30 nodes

4.5.4 Effect of interference range on broadcast performance

We also study the sensitivity of the results to the value of interference range. We vary the normalized interference range κ from 1 to 3 in steps of 0.2. Note that since the nodes are distributed within a square of 1000 ^{2} m^2 and the maximum transmission range is 483 m (see Table 1.1), a κ value of 3 corresponds to infinite interference range. Results for normalized broadcast latency and throughput for small networks (comprising of 30 nodes) are showed in Figures 4.7(a) and 4.7(b). Similar results have been obtained for larger networks (100 nodes) and are omitted. With increasing interference range, these figures unsurprisingly show that broadcast latency increases while throughout decreases. These figures also show that the interference range does not effect the relative performance of the algorithms.

4.5.5 Degree-of-freedom of performing multiple transmissions for same packet

We now consider the case when the nodes are allowed to transmit the same packet multiple times but at different rates. We run our simulation by assuming that the network consists of 30 nodes in varying network area[4]. We do not limit the number of times a node can transmit the same packet. We find that, over 100 random topologies of fixed number of nodes in a fixed area, multiple transmission do not significantly reduce the broadcast latency. In Figure 4.8, we plot the percentage of topologies that require multiple transmissions, and the percentage of reduction in delay for topologies that do require multiple transmissions. It appears that multiple transmission is only required by a fairly small number of topologies. If we consider the WCDS algorithm, only 2 out of 100 topologies for a network area of 1 km^2 require multiple transmissions, with multiple transmissions resulting in a 10% reduction in broadcast latency for these 2 topologies.

It appears that multiple transmission may not be required in the single-radio single-channel (SR-SC) WMN scenario. However, multiple transmissions is likely to be more useful in the multi-radio multi-channel environment since we know that for the infinite-radio infinite-channel case, the best broadcast latency is achieved by using the shortest path tree which requires multiple distinct-rate transmissions for the same packet. From Figure 4.6(a), we see that the WCDS algorithm (constrained to transmit the same packet only once) results in a normalized broadcast latency of about 2. *This means that the potential improvement offered by multi-radio multi-channel for latency reduction is still large, and should be investigated further.* This investigation will be taken up in Part-II of this thesis.

[4]We have also conducted simulations for a network of 100 nodes, and the results are quantitatively similar.

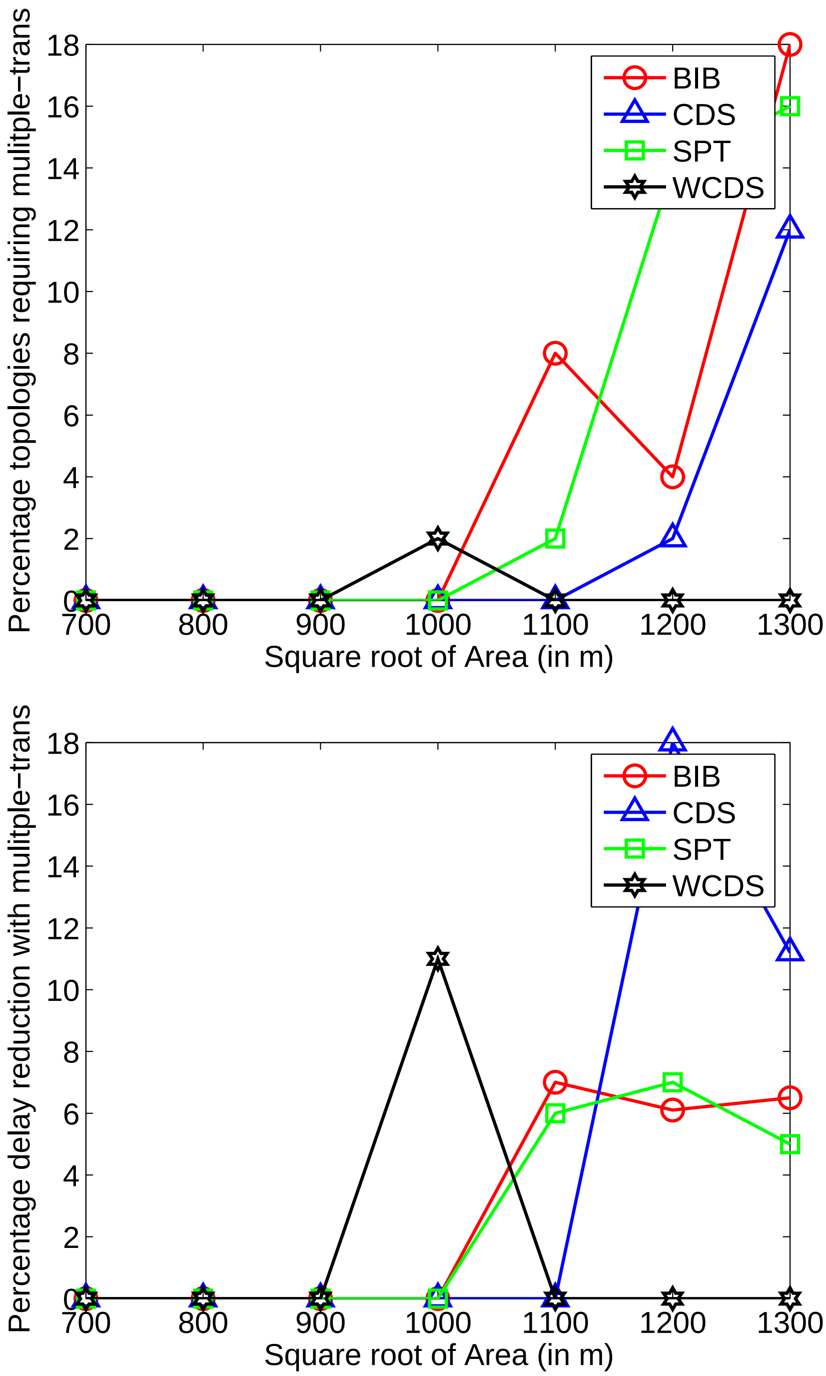


Figure 4.8: Percentage of topologies that require multiple transmission (top), and reduction in delay through multiple transmission (bottom) for a given network area and algorithm

4.6 Performance evaluation in Qualnet using 802.11 MAC

The results that have been presented up to now have used the idealized centralized scheduler described in Section 4.3.3, while assuming the simplifying assumptions presented in Section 4.2.2. *We will now present results of our simulations using the*

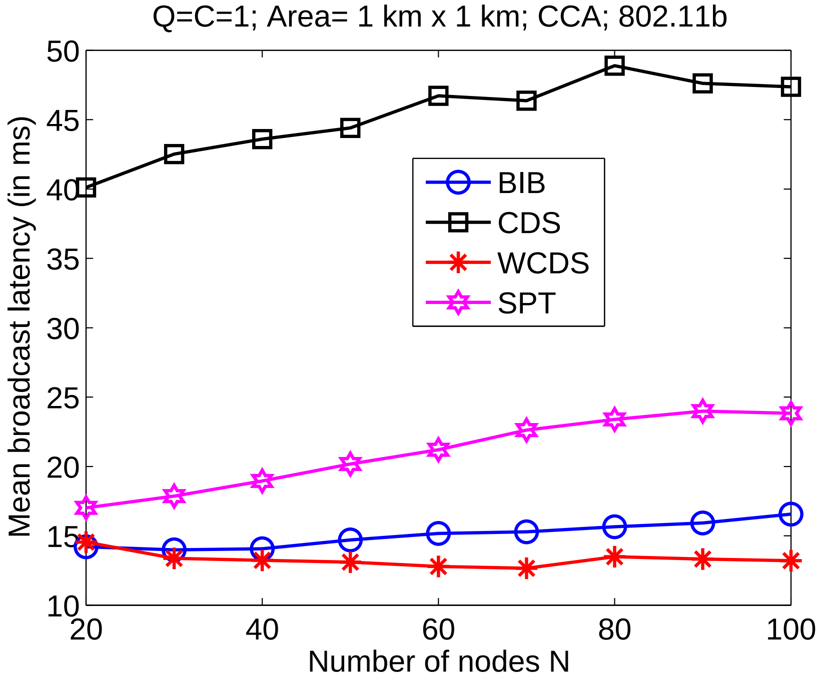


Figure 4.9: Performance comparison of WCDS, SPT and CDS when a practical MAC protocol (802.11b) is used in the Qualnet discrete-event simulator

Qualnet [1] simulator to analyze our heuristics' performance when the scheduling is done with decentralized practical MAC protocols. In our simulations, we assume PHY 802.11b at the physical layer, which uses a pre-configured BER-based packet reception model. The MAC 802.11 with Distributed Coordination Function (DCF) was chosen as the medium access control protocol. All default parameters are assumed unless stated otherwise. The simulation results are presented for average results of 100 topologies of varying node density in an area of 1000^2 m^2.

The broadcast latency performance of WCDS, BIB, SPT and CDS is presented in Figure 4.9 for varying number of nodes. We observe that the relative performance of our heuristics is qualitatively the same as the results that we have presented in the previous section (i.e., Section 4.5). It is seen that the best performing heuristic is WCDS, followed by BIB and SPT. CDS, which does not exploit the link rate-diversity, is the worst performing heuristic. *Therefore, the performance results for our broadcast heuristics when assuming idealized centralized MAC scheduler or alternatively practical decentralized scheduler return qualitatively similar results.*

4.7 Fundamental design principles for multi-rate broadcast

In Section 4.5, we have presented the performance evaluation of our heuristics using the transmission rate-transmission range characteristics (or rate-range curve for

Transmission rate (Mbps)	Max. transmission range (m)
r_0	d_0
ρr_0	γd_0
$\rho^2 r_0$	$\gamma^2 d_0$
$\rho^3 r_0$	$\gamma^3 d_0$

Table 4.1: This table shows the relationship between the transmission rates and transmission range for a hypothetical multi-rate system. The rate-range curves are parameterized by $\gamma \in (0, 1)$.

short) given in Table 1.1. The results show multi-rate multicast using BIB and WCDS give a lower worst case latency compared with the multicasting using the lowest rate only as given by CDS. In this section, we will study how sensitive this result is to the choice of rate-range curves. The result of this investigation can help us to answer a number of fundamental design questions for multi-rate systems, such as: (1) *Given a multi-rate given with n different rates, is it necessary to use all the n different rates?* (2) *If not, which of the n different rates should we use and what is an efficient method to decide that?*

4.7.1 Transmission's rate and coverage area product

In order to study the effect of rate-range curves on the broadcast performance, we use a family of hypothetical rate-range curves as given in Table 4.1. Our hypothetical system has a minimum transmission rate of r_0 Mbps whose transmission range is d_0 m. Each subsequent transmission rate is a factor of $\rho(> 1)$ greater but whose transmission range is a factor of $\gamma(< 1)$ smaller. Let us assume for the time being $\gamma = \frac{1}{2}$. Consider the transmission of a frame of size p bits. If the lowest rate is used, this packet will reach all nodes in the area of πd_0^2 in a time of $\frac{p}{r_0}$. However, if this is to be transmitted using the second lowest rate $r_1 = \rho r_0$, then each transmission will only cover an area of $\frac{1}{4}\pi d_0^2$ requiring a shorter time of $\frac{p}{\rho r_0}$ for each transmission. Therefore, four transmissions at rate r_1 can cover the same area as one transmission at rate r_0. Furthermore, in the worst case where these four transmissions at rate r_1 are within the interference range of each other, then they can only take place one after the other and this will take a total time of $4\frac{p}{\rho r_0}$ to complete. Thus, if $\rho > 4$, it will be always be more efficient to transmit at rate r_1.

Generalizing the argument used in the last paragraph, we propose to use the product of transmission rate and transmission coverage area (or rate-area product or RAP for short) as a measure of efficiency of a certain transmission rate. Thus, with the hypothetical system given in Table 4.1, it will be more efficient to use the higher rate if $\gamma^2\rho > 1$, otherwise the lowest rate should be used instead. Alternatively, a transmission rate with a higher RAP is more efficient for broadcast. In order to

	% of times each transmission rate is used			
γ	Rate $\rho^3 r_0$	Rate $\rho^2 r_0$	Rate ρr_0	Rate r_0
0.9	100	0	0	0
0.8	96	4	0	0
0.7	70	18	6	6
0.6	41	27	12	20
0.5	14	28	21	37

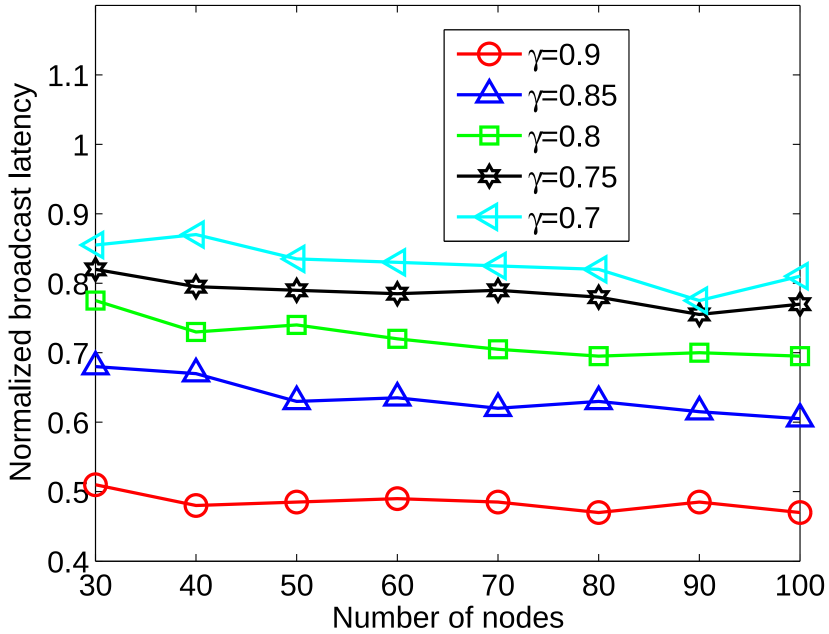
Table 4.2: The table shows the average percentage of times each transmission rate is used for different value of γ (the rates decrease from left to right since $\rho = 2$).

verify this conjecture, we perform a number of simulations using the same method as in Section 4.5 except that the rate-range curve in Table 4.1 is used.

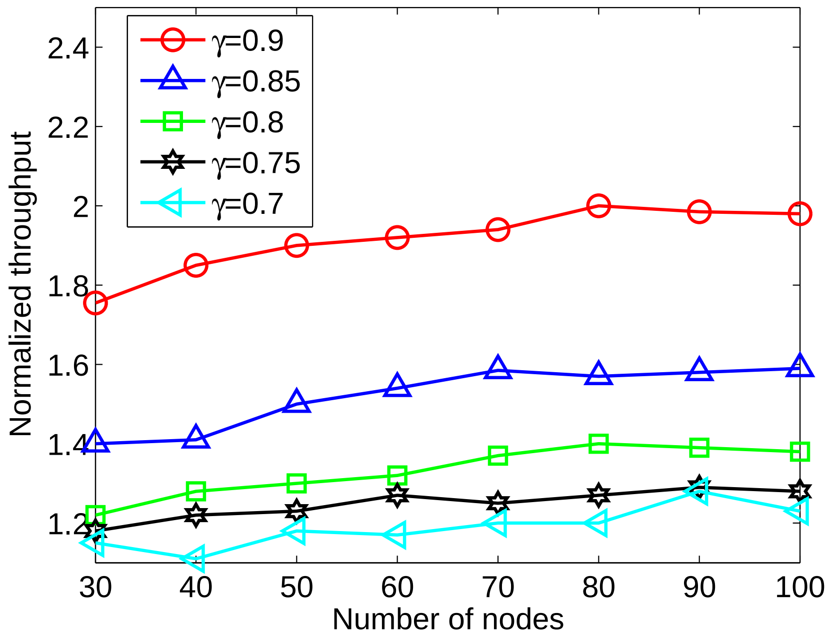
In the first set of simulations, we use $\rho = 1.5$, $r_0 = 1$, and $d = 500$. Thus, we expect that, if the above hypothesis holds, then it will be more efficient to use the higher transmission rates if $\gamma > \frac{1}{\sqrt{\rho}} = 0.82$. Five different values of $\gamma = 0.7$, 0.75, 0.8, 0.85 and 0.9 are used. Only heuristics BIB, WCDS and CDS are used. We normalized the delay and throughput by using those of CDS. The normalized delay and throughput of WCDS are given in Figures 4.10(a) and 4.10(b). It shows that WCDS gives a better latency and throughput than CDS for all values of γ. For $\gamma \geq 0.8$, WCDS exploits multi-rate and gives far better delay and throughput than CDS; for $\gamma < 0.8$, WCDS still performs better than CDS but the results are comparable. These observations therefore confirm our earlier conjecture.

We repeat the above experiment but this time we choose $\rho = 2$ which means that it is more efficient to use the higher rate for $\gamma > \frac{1}{\sqrt{\rho}} = 0.72$. The results for WCDS are plotted in Figures 4.11(a) and 4.11(b). These results again show that WCDS performs better than CDS for all values of γ though the performance gap diminishes for $\gamma \leq 0.7$. We can understand this by examining the average percentage of times that each transmission rate is used for each value of γ. The results are shown in Table 4.2. It shows that if the rate-range curve is favorable (i.e., when $\gamma > 0.7$), then the higher transmission rates are used most of the time. However, even when the rate-range curve is less favorable (i.e., when $\gamma < 0.7$), the higher rate transmissions are also used but less often.

We have also studied the sensitivity of the above results (regarding different rate-range curves) to interference range. All the above simulations are conducted with a normalized interference range κ, which is defined as the inference range to the maximum transmission range, of 1.7. We have simulated with six different normalized interference ranges: 1, 1.5, 2, 2.5, 3 and 5. Since the maximum transmission range is 500 m, a normalized interference range of 3 or 5 means an infinite interference range. Our results show that the interference range has negligible effect on the normalized delay and the normalized throughput of WCDS for different rate-range relationships.



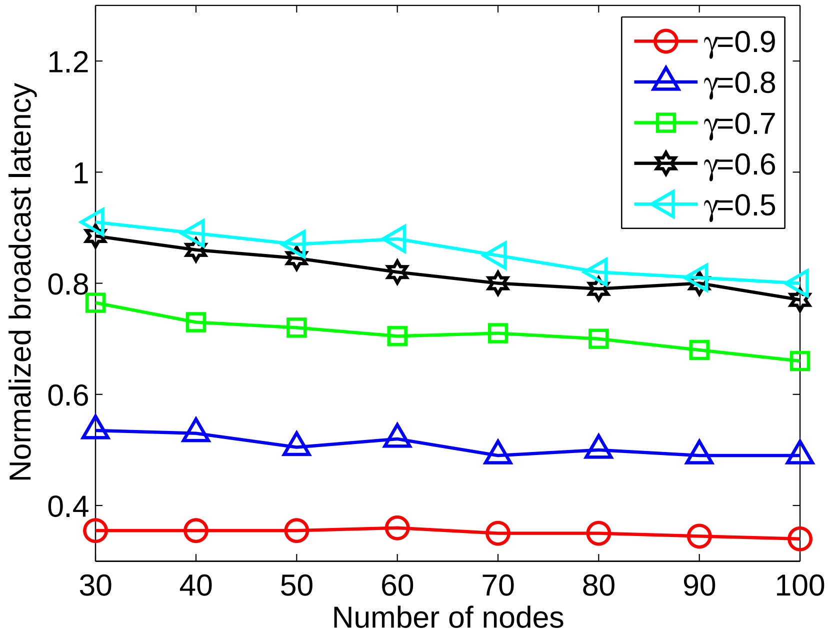
(a) Mean normalized broadcast latency



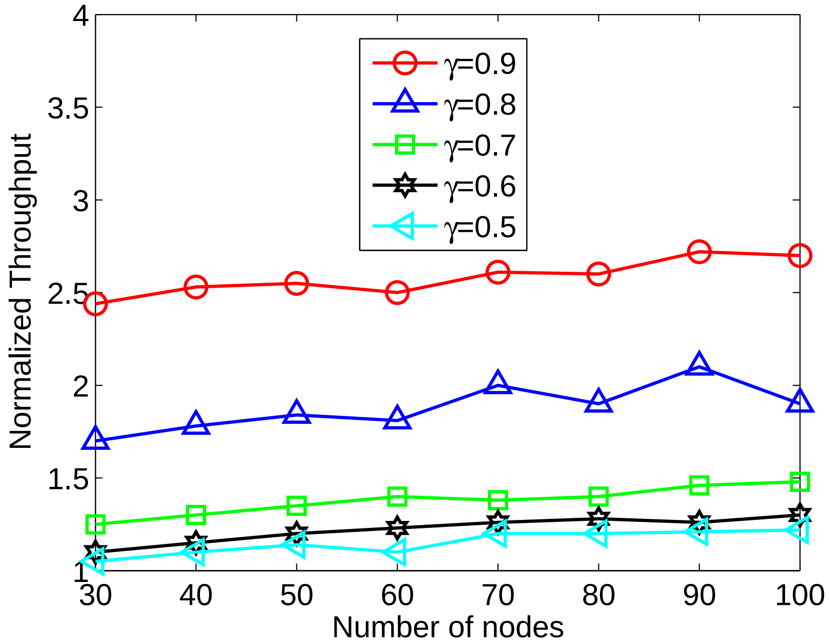
(b) Mean throughput

Figure 4.10: Performance of WCDS normalized to CDS over 100 randomly generated topologies of different network size for $\rho = 1.5$ and $\gamma = 0.7, 0.75, 0.8, 0.85$ and 0.9

The results in this section show that whether one should try to exploit multi-rate for network wide-broadcast depends on rate-area-product (RAP) of the transmission rates. If the higher transmission rates have a higher RAP compared with the lowest rate, then using multi-rate link layer broadcasts can result in significant reduction in broadcast latency. Applying this rule-of-thumb to the rate-range characteristics



(a) Mean normalized broadcast latency



(b) Mean throughput

Figure 4.11: Performance of WCDS normalized to CDS over 100 randomly generated topologies of different network size for $\rho = 2$ and $\gamma = 0.5, 0.6, 0.7, 0.8$ and 0.9

in Table 1.1, it can easily verified that the rate-area product is higher for higher transmission rates and this agrees with the results we have in Section 4.5.

4.7.2 Channel capacity and multi-rate networks

In Section 4.7.1, we demonstrate that transmission rates with large RAP are good for achieving low broadcast latency. With improvement in coding, wireless signal processing etc., the achievable wireless transmission rate is pushing closer to the Shannon capacity. An interesting question is to study the RAP if the transmission rate at a distance is given by the Shannon capacity. We consider a system where the bandwidth $B = 10MHz$, the SNR at distance $d_0 = 50$m is $30dB$[5]. Assuming that the rate R at distance d is given by the Shannon capacity formula, as follows:

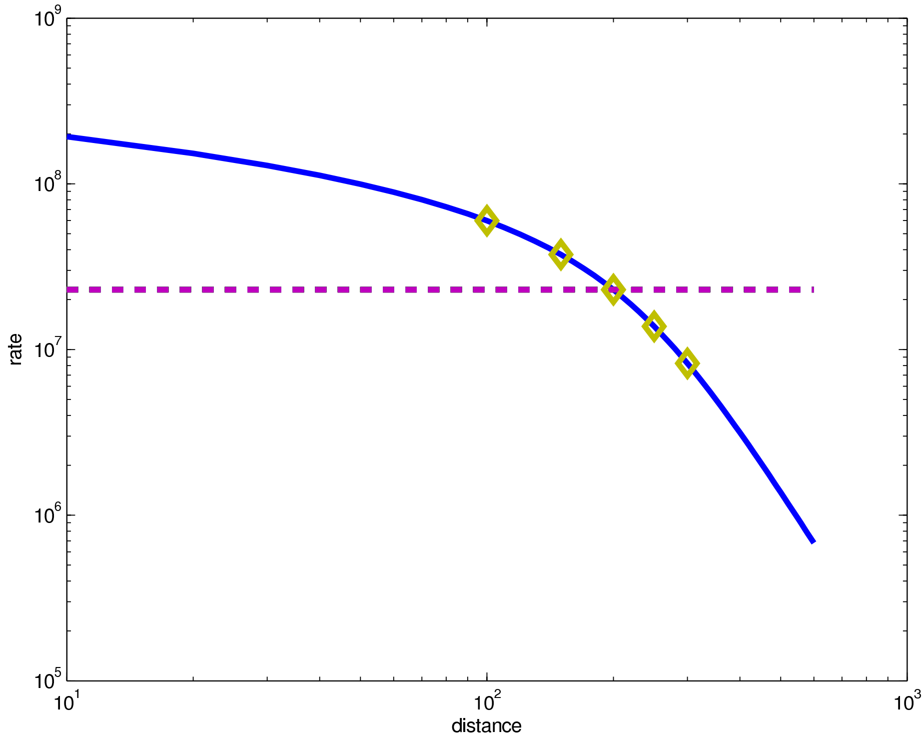
$$R \;=\; B\log_2(1 + SNR(\frac{d_0}{d})^{\theta}) \tag{4.16}$$

Let θ denote the path loss exponent. Assuming that $\theta = 4$, Figure 4.12 shows R and RAP as a function of d. It shows that the RAP increases for small values of d and decreases for large d. This is understandable since for small d, $R \sim log_2(\frac{1}{d})$ and for large d, $R \sim \frac{1}{d^{\theta}}$. It can be shown, via differentiating $R\pi d^2$, that the transmission rate (whose corresponding spectral efficiency is ψ) that maximizes the RAP is the solution to the equation $\psi - \frac{\theta\log_2 e}{2}(1 - 2^{-\psi})$ which says that the optimal ψ is a function of the path loss exponent θ only and not of other parameters. For $\theta = 4$, the maximum RAP (indicated by the dashed lines in Figure 4.12) occurs around a spectral efficiency of 2.3 bps/Hz. The lowest transmission rates for both 802.11a/b has a spectral efficiency far lower than this and therefore have poor RAP. By adding higher transmission rates with better RAP to 802.11b (see Table 1.1), the broadcast latency of 802.11b is improved as seen in Section 4.5. However, the Shannon RAP predicts that RAP will eventually fall for higher transmission rates. From the technical specifications of a commercial 802.11b/g product [2], we find that the outdoor transmission ranges for rates 1, 6, 11, 18 and 54 Mbps are respectively 610, 396, 304, 183 and 76m, giving RAP of 1.2, 3.0, 3.2, 1.9 and 1.0 Mbps-km^2 which eventually falls for high transmission rates.

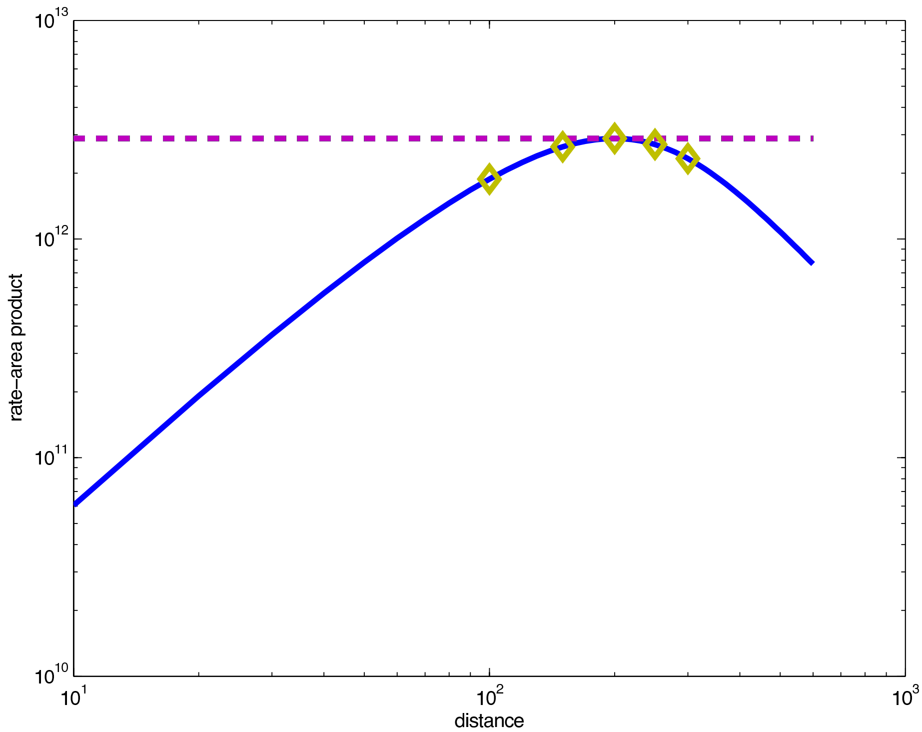
We assume a hypothetical multi-rate system by selecting five points from the Shannon rate-range curve indicated by the diamonds in Figures 4.12. Since it is likely that future wireless systems will have rates with efficiency above and below 2.3bps/Hz, the rate that gives the maximum RAP is selected as well as two points on each side of it. Note also that the Shannon transmission rate can only be used if no other nodes are transmitting, or in other words, the interference range is infinity. Since we find that in the last section that the interference range has little impact on the result, we keep the normalized interference range as 1.7 as before. We use the same simulation set up as in Section 4.7.1 except that we use the following five algorithms:

1. WCDS with all the five transmission rates

[5]We will see later that these parameter values will not affect the general discussion here.



(a) Distance and Shannon capacity

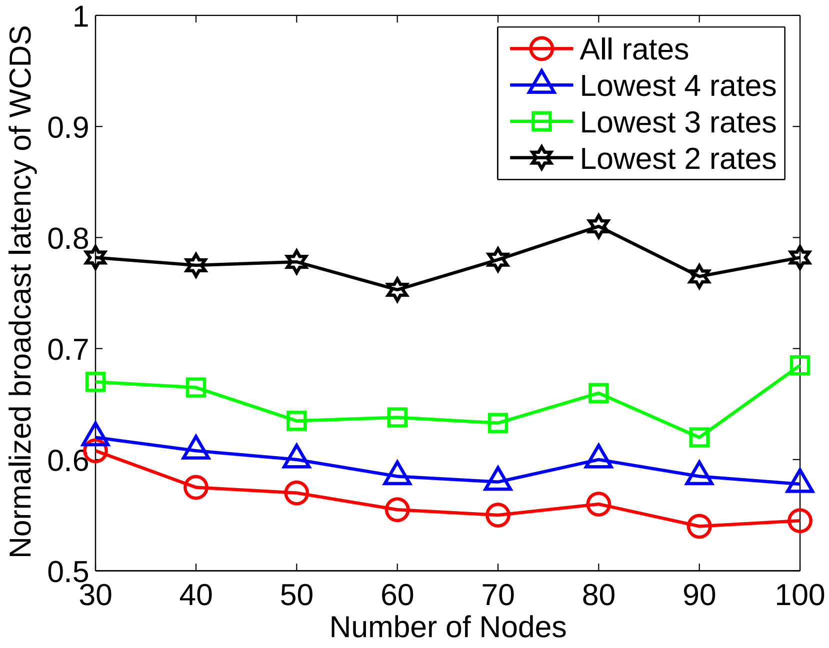


(b) Corresponding Rate-Area-Product (RAP)

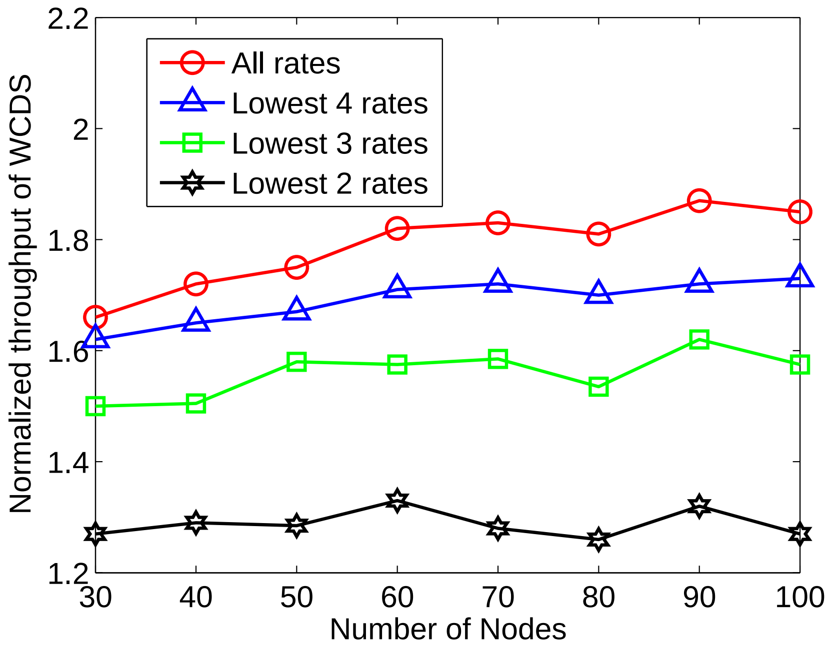
Figure 4.12: Relationship between distance and Shannon capacity, and corresponding RAP

2. WCDS with only the lowest four transmission rates

3. WCDS with only the lowest three transmission rates



(a) Mean normalized broadcast latency



(b) Mean throughput

Figure 4.13: Performance of WCDS relative to CDS using different number of distinct transmission rates

4. WCDS with only the lowest two transmission rates

5. CDS with the lowest transmission rate only

We normalize the results for the various WCDS algorithms with the results from CDS. These normalized results are showed in Figure 4.13. It can be seen that the best results are given by WCDS using all the five rates, thus again confirming that multi-rate is useful for reducing broadcast latency. Since the third rate has the highest RAP, note that there is sizeable performance gap between using the lowest 2 rates and the lowest 3 rates. These results also show that the efficiency of a particular rate for network-wide broadcast can be predicted by the rate's RAP value. The RAP values of the multiple supported rates can also inform a protocol designer regarding which rates should be employed if it is decided to use only a subset of all rates for broadcast. We will see in the next chapter (Chapter 5) that if it is decided that only one rate should be used for all broadcast (not necessarily the lowest rate), then as a general rule-of-thumb, a rate that has a higher RAP value performs better.

4.8 Conclusions

In this chapter, we have proposed a centralized heuristic called WCDS for the minimum-latency-broadcast problem in SR-SC multi-rate WMNs. Our proposed algorithm WCDS exploits the rate-diversity of a SR-SC multi-rate WMN by incorporating it in its design. It also exploits the network medium's WBA, and at the same time, it explores an extra degree-of-freedom of being able to perform multiple transmissions, at distinct transmission rates, for the same packet.

Our simulation results for WCDS, using typical IEEE 802.11 parameters, show $\sim$ 3 to 6 fold reduction in the broadcast latency compared to the scheme of always utilizing the lowest rate for broadcast. We have observed that broadcast latency performance can be improved if we use the degree-of-freedom of being able to perform multiple distinct-rate transmissions for the same packet; however, the performance gain was not significant and infrequent, therefore, dropping this feature is justifiable if its implementation is expensive.

We have also proposed a general rule-of-thumb, which we call the rate-area-product (RAP), that can be used to predict the efficiency of a particular transmission rate for broadcast. Investigations of theoretical Shannon limit suggest that the case for using at least a small subset of the available choice of rates for link-layer broadcasts will become even more compelling as better modulation and coding techniques are introduced.

Chapter 5

Distributed MLB Solution for SR-SC Multi-rate WMNs

5.1 Introduction

Although the WCDS algorithm, proposed in Chapter 4 for the MLB problem in SR-SC multi-rate WMNs, significantly improves performance of existing multi-rate unaware broadcast heuristics, its utility is hindered by its centralized operation. Since centralized operations require global network information, implementation can be quite expensive in terms of computation and communication costs. In addition, the centralized WCDS algorithm is not robust to topology reconfiguration since small changes in topology (for example, due to addition of new nodes) can result in recalculation of the entire forwarding tree. It is clearly desirable for scalable operations to develop decentralized algorithms that require only localized information. In this chapter, the focus is on building decentralized and localized MLB solutions, assuming the FMM framework (Chapter 4), that can exploit the rate-diversity and WBA offered by SR-SC multi-rate WMNs, and approach the performance of the centralized algorithms of Chapter 4. We recall that in a FMM broadcasting framework, WMN nodes can adjust link-layer multicast transmission rate for each link-layer frame.

The distributed algorithms that we shall present in this chapter are influenced by the WCDS centralized algorithm presented in Chapter 4. The WCDS algorithm utilized the concept of link-layer multi-rate multicast, in which a WMN node can adapt its link-layer transmission rate for multicast/broadcast traffic. The WCDS algorithm exploited two features that are present in multi-rate WMNs but not in a single-rate WMN. *Firstly*, if a node has to perform a link-layer multicast to reach a number of neighbors, then its transmission rate is limited by the smallest rate on each individual link, e.g., if a node n is to multicast to two neighboring nodes m_1 and m_2, and if the maximum unicast rates from n to m_1 and m_2 are, respectively, r_1

and r_2, then the maximum rate n can use is the minimum of r_1 and r_2. *Secondly*, for a multi-rate WMN, the broadcast latency can be minimized by having some nodes transmit the same packet more than once, but at a different rate to different subsets of neighbors (called as 'distinct-rate transmissions'). The WCDS algorithm utilized these insights to heuristically solve the MLB problem in SR-SC multi-rate WMNs. The WCDS algorithm considers the WBA and the multi-rate capability of the network, and also incorporates the possibility of multiple distinct-rate transmissions by a single node. In practice, however, multiple distinct-rate transmissions were rarely used by any node[1]; therefore, we do not consider the possibility of having a node perform multiple distinct-rate transmissions in this chapter.

Since the aim of this chapter is to build a distributed version of the WCDS centralized algorithm (which is essentially a CDS technique), it is now in place to study the various known methods of building a CDS distributively. We can utilize some aspects of the existing distributed CDS construction techniques (designed for single-rate networks) by supplementing them with techniques that can exploit the rate-diversity of multi-rate WMNs. A survey of existing research on distributed CDS-based broadcast algorithms was earlier presented in Section 2.1.2; we reproduce the relevant details next for completeness.

The CDS calculation techniques for single-rate networks can be broadly divided into two major classes. Algorithms of the first class (e.g. the algorithm of Wu and Li [32] [47] and that of Adjih et al. [43]) initially compute a large CDS and then attempt to prune away redundant nodes by means of local optimizations. The second class of algorithms (e.g. the algorithm proposed in [45]) firstly calculate a small dominating set and then connect it up. The CDS calculated by the second class of algorithms is generally smaller than the CDS calculated by the first class of algorithms; however, the smaller cardinality of the set of forwarding-nodes set comes at the expense of increased complexity and reduced locality. In our work, we shall see that the ability to exploit increased transmission rates is more important than reduced CDS size (this assertion is discussed in detail in Section 5.4). Accordingly, we only consider the first class of algorithms for modifications to be used in our research of designing distributed multi-rate CDS algorithms.

We will modify two underlying (rate-diversity unaware) techniques that both belong to the first class of algorithms discussed previously. The first technique, called the *Wu-Li* algorithm [32], is a simple localized technique that uses only 2-hop neighborhood information to compute a CDS as follows. Initially, all vertices (nodes) are unmarked. The marking process uses the following simple rule: any vertex having two unconnected neighbors (not connected directly) is marked as a dominator. The set of marked vertices form a rather large CDS V'. Two pruning

[1]Only a few ($\sim$ 20%) simulation topologies used multiple distinct-rate transmissions at an individual node.

techniques are then used to reduce the CDS size. A node u can be removed from V' if there exists a higher-id node $v \in V'$ such that the closed neighbor set[2] of u is a subset of the closed neighbor set of v. For the same reason, a node u will be deleted from V' when two of its connected neighbors in V' with higher IDs can cover all of u's neighbors. This pruning idea is generalized to the following rule [32]: a node u can be removed from S if there exist k connected neighbors with higher IDs in S that can cover all u's neighbors. The second technique to locally compute a CDS is called "multi-point relaying" (*MPR*) [43]. The MPR technique allows each node u to first elect a 'multi-point relay set" (*MRS*) [48] [33] from its one-hop neighbors to cover its two-hop neighbors. Finding a MRS with minimum size is NP-Complete [33]. The CDS is calculated as follows [43]: each node first compute a MRS, a subset of one-hop neighbors that can cover all its two-hop neighbors. After each node has determined its MRS, a node decides that it is in the connected dominating set by matching either Rule 1: the node is smaller than all its neighbors *or* Rule 2: it is the multipoint relay of its smallest neighbor.

Although neither of these two relatively simple algorithms necessarily form the smallest CDS, we shall use them in the 'initial marking' stage, since the subsequent stage of 'neighbor grouping' and 'rate maximization' (which we introduce) turn out to be much more important for multi-rate networks than the optimal computation of the initial CDS itself. In particular, for multi-rate WMNs, a desirable CDS, apart from a small sized forwarding set, has other characteristics such as high transmitting rates at the chosen nodes (in the backbone) to ensure low broadcast latency. We note that although maximization of transmission rates of the nodes in CDS is attempted initially, the WBA *is* accounted for by choosing those nodes that maximize the WBA, according to the RAP principle (Chapter 4), in the last phase (Section 5.3.3) of our framework.

The main contributions of this chapter are highlighted below:

1. *We propose three FMM localized and distributed MLB heuristics, called MDW, MEW and MRRA, for SR-SC multi-rate WMNs.* These heuristics improve the performance of rate unaware broadcast heuristics, and the performance gap between such distributed heuristics and the centralized heuristics of Chapter 4 is not large.

2. *We demonstrate that existing backbone-based protocols that do not exploit rate-diversity of SR-SC multi-rate WMNs perform poorly in such network.* We propose a heuristic solution composed of three stages, which we call '*initial marking*', '*neighbor grouping*', and '*rate maximization*', using which existing backbone-based non-rate-diversity-aware protocols can be enhanced.

[2]Closed neighbor set is the union of the node itself and its neighbors.

V	Set of vertices (or, nodes)
E	Set of edges (or, links)
Π	Set of transmission-rate of links in E
Λ	Set of channel of links in E
ρ_i	i^{th} highest transmission rate supported by MAC
N	Total number of nodes in network (=$\|V\|$)
$\rho(u)$	Current transmission-rate of u
ρ_0	Rate of a non-transmitting interface
$N(u)$	1-hop neighbors that u is currently covering
$N_{\rho_k}(u)$	1-hop neighbors of u (on rate ρ_k)
$r(u)$	Set of rates u having a "rate-limiting-node"
L	Number of distinct rates supported by MAC
$\pi(u,v)$	Highest transmission-rate link (u,v) can use
$\lambda(u,v)$	Channels link (u,v) can use
m	Number of marked-nodes
d	maximum number of neighbors of a marked-node

Table 5.1: Index of mathematical symbols used in this chapter

Chapter Outline: We introduce our network model next in Section 5.2. We present our three-staged distributed broadcast framework, and introduce three localized and distributed broadcast heuristics for this framework in Section 5.3. We present performance evaluation of our heuristics in Section 5.4. We conclude this chapter in Section 5.6.

5.2 Network model

We assume the FMM broadcasting framework (introduced in Chapter 4) in which WMN nodes can adjust link-layer multicast transmission rate for each link-layer frame. Our network and interference model is similar to that described in Chapter 4. We will reproduce the relevant details here for completeness. We use an undirected graph $G = (V, E, \Pi)$ to model the given mesh network topology, where V is the set of vertices, E is the set of edges and Π is the set of weights of edges in E. The vertex v in V corresponds to a wireless node in the network with a known location. An undirected edge (u, v), corresponding to a wireless link between u and v, is in the set E if and only if $d(u, v) \leq r$ where $d(u, v)$ is the Euclidean distance between u and v and r is the range of the lowest-rate transmission. The transmission rate of a link $\pi(e)$ ($e = (u, v) \in E$) is the quickest transmission rate that can be supported on link represented by e. The set Π contains the rates of all links in E. Let us assume that each node has a choice of L different rates: $\rho_1, \ldots, \rho_L$, with $\rho_1 > \rho_2 \ldots > \rho_L$. Also, let $\rho(u)$ denote the transmission rate of node u. Recall that $\pi(u, v)$ denotes the quickest-rate transmission supported between u and v. $N_k(u)$ denotes all nodes x such that $\pi(u, x) = \rho_k$; alternatively, $N_k(u) : k = 1, ..., L$ denotes the set of neighboring nodes that node u reaches at rate ρ_k (but cannot reach at any higher rate $\rho_j : j > k$). The mathematical symbols used in this chapter are tabulated in Table 5.1.

5.3 Distributed Broadcasting Algorithm

Our proposed distributed and localized broadcast algorithm for multi-rate WMN is composed of three stages. In the first stage named 'initial marking', we use any of the existing broadcast algorithms for single-rate wireless networks to calculate a sufficiently small-sized (which is unaware of rate-diversity, or 'rate-unaware') CDS; all transmissions at the end of the first stage of 'initial marking' are assumed to be taking place using the lowest possible rate. The second stage called the 'neighbor-grouping and rate-maximization' stage is itself composed of two substages: the decision of the neighboring nodes a particular node must cover is made during 'neighbor grouping' (NG) substage, whereas the Rate-Maximization substage attempts to maximize transmission rates across all the marked nodes (recall that nodes are marked during Stage 1). The third and last stage, called broadcast 'tree-construction' constructs a broadcast source-independent tree and eliminates redundant transmissions that were retained during the earlier two stages.

In this section, three new distributed and localized broadcast algorithms are presented. The first two of these algorithms are based on the Wu-Li algorithm and differ on how and when the pruning operation is performed; we name these two protocols: 'multi-rate expedited-pruning Wu-Li' (MEW) and 'multi-rate delayed-pruning Wu-Li' (MDW). The third algorithm is based on the concept of MPR and is called 'multi-point rate-maximized relaying algorithm' (MRRA). The mechanisms of these algorithms are next explained during the three different stages of our framework.

5.3.1 Stage 1-Initial Marking

During Stage 1, we determine a rough measure of the forwarding set (or CDS) by following a marking process using the lowest-rate transmission only. As different transmission rates have different transmission ranges (see Table 1.1), different rates have different neighbor sets. At the end of Stage 1, we have a forwarding set (or CDS) and the transmission rate at each of these forwarders is set to be the lowest-rate. The actual decision of rates (and attempts to increase them) is made in subsequent stage of 'neighbor grouping' and 'rate maximization'.

The MEW and MDW broadcast algorithms both employ the Wu-Li marking process (explained in Section 5.1 earlier) in which a node is marked if it has two neighbors that are not directly connected. A node u is considered a neighbor of v if distance between u and v is less than or equal to the range of the lowest-rate transmission i.e. $d(u, v) \leq r$ where r is the range of rate ρ_L. The MEW and MDW algorithms differ in their implementation of Wu-Li pruning rules as outlined in [47] and discussed in Section 5.1 earlier. Whereas MEW (multi-rate 'expedited-pruning' Wu-Li) prunes away the redundant marked nodes *expeditiously* (during Stage 1) by following Wu-Li pruning rules (Section 5.1), the MDW algorithm (multi-rate 'delayed-pruning' Wu-Li) does not perform the pruning as part of Stage 1. Thus, in MDW, the pruning process is *delayed* and performed later, during a substage of Stage 2 called 'rate maximization' (discussed later) and then again during Stage 3. We shall enumerate the potential benefits of such delayed pruning when we reach the discussion about 'rate maximization'.

The MRRA algorithm, on the other hand, follows the approach suggested by Adjih et al. [43] to determine the initial CDS. It employs the concept of multi-point relaying to calculate, at each node, all its one-hop neighbors that should forward to cover its two-hop neighborhood. We have adapted multipoint relaying to include rate-diversity available in WMN. This is done by using the WCDS algorithm (which is a rate-aware broadcast algorithm for SR-SC multi-rate WMNs presented in Chapter 4) to generate the multi-point relay set (MRS) of each node i.e. each node would execute the WCDS algorithm with itself as the source on its 2-hop subgraph to determine the set of its one-hop neighbors that should act as the MRS to cover all of its 2-hop neighbors. By utilizing rate-aware localized MRS decisions, we ensure that the choice of the relay set at each node takes into consideration the inherent rate-diversity available in the WMN. After each node has determined its MRS, a node decides that it is in the connected dominating set if and only if either of the following two rules is satisfied: *Rule 1:* the node is smaller than all its neighbors *or Rule 2:* it is multipoint relay of its smallest neighbor. Note that at the end of this marking process, only the initial forwarding set (or CDS) is calculated and all marked nodes are assumed to forward at the lowest-rate. The actual rates of transmission would be decided in the next stage.

Differences between our three algorithms are confined to those in Stage 1. Since, the next two stages (Stage 2 and Stage 3) are common to all three of our proposed algorithms (MEW, MDW and MRRA), we shall, therefore, give a general description of these two stages, which should be assumed to apply to all our algorithms.

5.3.2 Stage 2-'Neighbor Grouping' and 'Rate Maximization'

'Neighbor Grouping' (NG) substage

In the step of 'neighbor grouping', we decide the neighboring nodes a marked node has to cover. The logic employed is straight-forward: a marked node should not be reducing its rate to cover a node that can, alternatively, be 'better' covered by another node. This step ensures that transmission rate at marked nodes is not constrained to a lower-rate because it has to cover all its possible neighbors.

The 'neighbor grouping' algorithm is explained in Algorithm 3. In the algorithm, each node u searches to see if there exists a one-hop neighboring node v which can be 'better' covered by w (another 1-hop neighbor of u; i.e. $w \in N(u)$). v is said to be better covered by w if the aggregate throughput/rate of the path $u \rightarrow w \rightarrow v$ is better than the throughput of the path $u \rightarrow v$. At the end of the algorithm, the 1-hop neighborhood of each marked node has been decided. Each marked node is responsible for ensuring that its 1-hop neighborhood is covered (by itself, or through another marked node, as we shall later see).

Algorithm 3 Algorithm for 'Neighbor Grouping' at a node u

for each one-hop neighbors $v \in N(u)$ **do**
    for each node $w \in N(u)\backslash\{v\}$ **do**
        if $1/\pi(u,v) > 1/\pi(u,w) + 1/\pi(w,v)$ **then**
            remove v from neighbor-list of u at rate $\pi(u,v)$
        end if
    end for
end for

Message Complexity: Assuming that 2-hop neighborhood information has been established prior to the NG stage, no message needs to be exchanged during the NG stage. Let us represent the maximum number of neighbors of a marked node by d and the number of marked nodes by m. After the NG stage completes, each marked node will broadcast a packet for a total maximum of m packets. The maximum size of the sent packet is $(1 + (L)d)$ times the bytes required to represent a node-id since the packet sent by a marked node conveys the sending marked node's node-ID, its neighbors on different rates. We note that L is a small (constant) value since typically limited rates are supported; the total message-complexity of the NG stage, therefore, is $O(md)$.

'Rate Maximization (RM)' substage

Before discussing the RM stage, we introduce the concept of "*rate-limiting-nodes*". We note that a lower-rate transmission can cover all nodes reachable at a higher-rate but not vice-versa; this implies that the maximum rate a node u can use to reach all its 1-hop neighbors $N(u)$ collectively, is the minimum of the (maximum) rate u can use to reach each individual node in $N(u)$. To illustrate this concept, assuming a single radio interface, refer to Figure 5.1 for an example topology. Although, u can reach nodes v and w with rate of 54 Mbps, u is constrained to transmit at a lower rate of 11 Mbps to reach nodes x, v and w collectively. Node x, for this topology, is referred to as a rate-limiting-node of node u, for its presence limits u's rate to 11 Mbps, with its absence the rate of u can be increased to 54 Mbps.

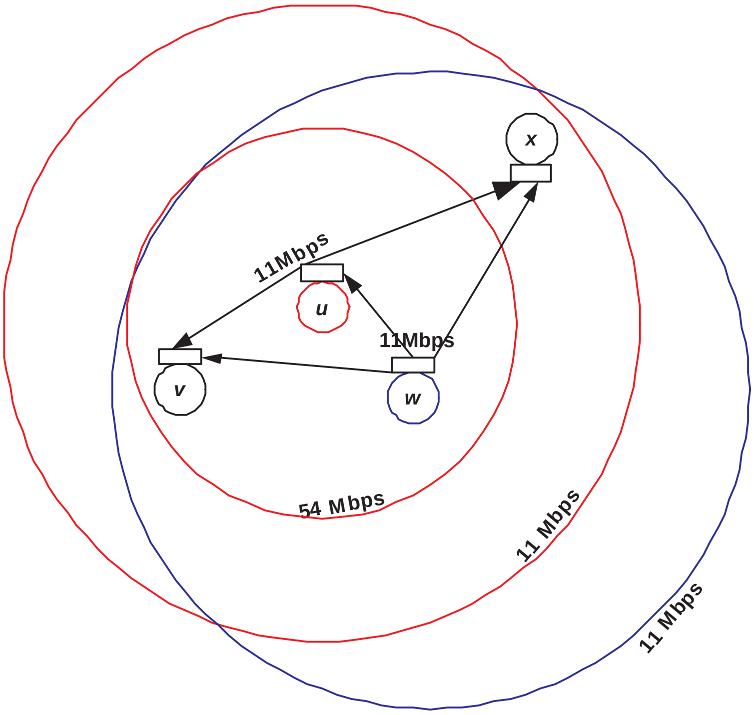


Figure 5.1: Before the 'rate maximization' stage at node u

The objective of the RM sub-stage is to find, for a node u, neighboring forwarding nodes to whom u's rate-limiting-nodes can be 'exported'. The utility of an export can be determined using, in particular, the "rate-area-product" (RAP) maximization principle described in Chapter 4. The export of rate-limiting-nodes, in general, will increase an interface's transmitting rate, with a node unmarking itself if all its neighbors have been exported. The challenge faced by RM, due to the potential danger of link asymmetry [3] that arises due to rate-diversity, is to maximize the rates at a node's interfaces while preserving the *strong* connectivity of the resulting dominating set. Since our framework determines forwarders and rates irrespective of the broadcast source (i.e., until Stage 3), it is important to ensure strong connectivity irrespective of the broadcast source.

To illustrate the concepts employed by RM, we refer to Figure 5.1 for an example topology comprising of three nodes. Node u can reach nodes $\{v, w\}$ and $\{x\}$ in a 54 Mbps and 11 Mbps transmission, respectively. Node w, however, can reach nodes $\{v$,$u\}$ and $\{x\}$ in a 54 and 11 Mbps transmission. We will study RM sub-stage at node u. Node u is constrained to use a lower rate (of 11 Mbps) if both neighbors of u (v and x) are to be covered in a single transmission. The rate-limiting-node of u is x. Node u will look for an higher-id marked node[4] that can cover u's rate-limiting-node using its current rate and be reachable from u after u increases its rate; also,

[3]For example, it is possible for node u to reach v but not vice-versa (where $\rho(u) < \rho(v)$) due to different ranges for different rates.

[4]The restrictive condition of only exporting to higher-ID neighbors is to avoid circular hand-offs.

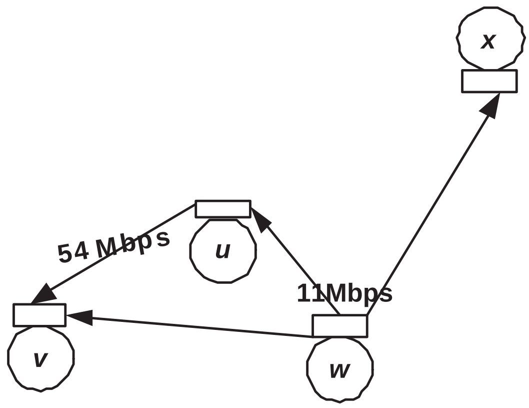


Figure 5.2: After the 'rate maximization' stage at node u

the sum of the uplink rates of u's neighbors should improve after an export. We check now if u's rate-limiting-node x can be exported to w. *Firstly*, x is reachable through w's current transmission; secondly, w is reachable from u even after u's rate is increased to 54 Mbps; *lastly*, the sum of rates of u's neighbor increases with this transfer (54+11=65 instead of 11+11=22 before). Since all conditions are satisfied, the export of x can take place increasing the transmitting rate of u to 54 Mbps as shown in Figure 5.2.

The RM algorithm, for any node u, is mathematically described in Algorithm 4. Node u will attempt to increase its rate if it is currently a transmitting node (i.e. it has some rate-limiting-nodes). The token *continue* is initially equal to 1 which indicates that rate-increase can be attempted; a token *continue* valued 0, on the other hand, implies that the rate-limiting-nodes of the current rate are non-exportable and further rate-increase must not be attempted. Initially, E (denoting the rate gain for the exported nodes) is set to zero. We denote the rates on which a node u has rate-limiting-nodes as $r(u)$. The total rates in $r(u)$ is not necessarily equal to the total number of rates L and is specific to the node u. The rates in $r(u)$ are arranged in a descending order, i.e., $r_1(u) > r_2(u)$ and so forth. For mathematical compactness, $r_0(u)$ denotes the fact that u would not be transmitting since a non-transmitting node has rate of zero.

The index of u's current transmission rate, $\rho(u)$, in $r(u)$ is represented as k in Algorithm 4. The rate-limiting-nodes (RLN) is calculated as the difference between the neighbors of u at the current rate ($N_{r_k(u)}(u)$) and the next higher rate in $r(u)$ (i.e., $N_{r_{k-1}(u)}(u)$, if $r_{k-1}(u) \neq r_0(u)$). For each node rln in RLN, it is checked for every node $h \in H$ where H comprises of higher-ID marked neighbors of u excluding RLN if, *firstly*, rln is a neighbor of h (i.e., $\pi(h, rln) \geq \rho(h)$ and, *secondly*, if u is a neighbor of h (to ensure strong-connectivity). The maximum uplink rate rln can receive from a node $h \in H$ fulfilling these conditions is stored in a variable called $rate\_new$ (that is initialized with -∞). The difference between the initial rate of rln

Algorithm 4 Algorithm for 'Rate Maximization' at a node u

$continue = 1$
while $continue$ and $\rho(u) \neq 0$ **do**
  $E = 0$; $continue = 0$;
  $r(u)$ = rates at u sorted in descending order
  k = index of $\rho(u)$ in $r(u)$
  if $k = 1$ **then**
    $RLN = N_{r_k(u)}(u) \cup u$
  else
    $RLN = N_{r_k(u)}(u) \backslash N_{r_{k-1}(u)}(u)$
  end if
  H= all higher-ID *marked* neighbors of $u \setminus \{RLN\}$
  ————————————————-
  {This part aims to find a neighbor to export nodes in RLN while satisfying RAP condition}
  ————————————————-
  for $m = 1$ to $|RLN|$ **do**
    rln= $RLN(m)$; $rate\_new = -\infty$;
    for $n = 1$ to $|H|$ **do**
      $h = H(n)$
      if $rln \in N(h)$ **and** $u \in N(h)$ **and** $\rho(h) > rate\_new$ **then**
        $rate\_new = \rho(h)$
      end if
    end for
    $rate\_diff = rate\_new - r_k(u)$
    $E = E + rate\_diff$
  end for
  ————————————————-
  if $E \geq 0$ **then**
    $continue = 1$; $\rho(u) = r_{k-1}(u)$
  end if
end while

and the $rate\_new$ is maintained in $rate\_diff$. The variable E contains the sum of $rate\_diff$ of all nodes in RLN. The nodes that cannot be exported have $rate\_diff$ of -∞. Thus, even for a single non exported rate-limiting-node at a particular rate, the value of E would be -∞. For each interface, if $E > 0$, its rate is increased and *continue* is set to 1; otherwise, if $E < 0$, *continue* is set to zero. The algorithm completes when increase in rate is not possible either due to export of all nodes, or due to *continue* token equal to zero.

Message Complexity: During the RM sub-stage, each time a marked node u is successful in increasing its rate, it would broadcast its new rate $\rho(u)$ to its neighbors in a message. The maximum number of these messages exchanged is $((m-1) \times L)$ with the size of a these messages being the sum of the bytes used to represent node-ID and rate-ID. Since L is a constant, total message-complexity of RM is $O(m)$.

Algorithm 5 Algorithm for the 'Tree Construction' stage, broadcast source is s

Initially, $label(v) = \infty, \forall v \in V$
$u = id(node)$
if $u = s$ **then**
  Send $RREQ$ with $RREQ.label = \frac{1}{\rho(s)}$
end if
————————————————————-
if $RREQRcvd.label < label(u)$ (non-duplicate) **then**
  $Parent(u) = RREQRcvd.sender$
  $RREP.nexthop = RREQRcvd.sender$
  $send(RREP)$ to $RREQRcvd.sender$
  $RREQ.sender = u$
  $RREQ.label = RREQRcvd.label + \frac{1}{\rho(u)}$
  $send(RREQ)$ to $N_{\rho(u)}(u)$
end if
if received $RREP$ and $RREP.nexthop = u$ **then**
  Activate $Forwarder$ flag
  $RREP.nexthop = Parent(u)$
  $send(RREP)$
end if

5.3.3 Stage 3-Tree-Construction

The forwarding set (CDS) and the transmission rates calculated are *independent* of the broadcast source, i.e., the same nodes (in the CDS) will forward at the same decided rate in all cases. However, the tree (i.e., the parent/children relationship among these nodes) will vary depending on the broadcast source. Redundant transmissions can be pruned (e.g. if a forwarding node can determine that all of its neighbors can also receive from another node of higher-priority, then this node can unmark itself). Thus, redundant transmission can be pruned away, based on the broadcast source, in Stage 3. We present our Stage 3 of Tree-Construction mathematically in Algorithm 3. Initially, the *label* of all nodes is equal to ∞. The source node, represented by s, starts by sending out a $RREQ$ message to its neighbors with $RREQ.label$ set to its transmission latency i.e. $\frac{1}{\rho(s)}$. Any node u that receives a $RREQ$ message will check if its label i.e. $RREQ.label$ is less than its current label; if so, then u will choose the sender of the $RREQ$ (represented by $RREQRcvd.sender$ in the algorithm) as its parent, send a $RREP$ back to it (setting $RREP.nexthop$ to $RREQRcvd.sender$) and modify its label to the received label. Furthermore, u would generate a new $RREQ$ message with itself in the $RREQ.sender$ field and increment its label with its transmission latency i.e. $\frac{1}{\rho(u)}$ and transmit it to its neighbors. When any node, represented by u again, receives a $RREP$ message and $RREP.nexthop$ is equal to u, it would activate the $Forwarder$ flag and set the $RREP.nexthop$ to its parent ($Parent(u)$) and re-send the $RREP$. In this manner, the Forwarding or Non-Forwarding status of each node is determined. During the

Transmission rate (Mbps)	Transmission range (m)	RAP (Mbps-km^2)
1	610	1.17
6	396	2.96
11	304	3.19
18	183	1.89
54	76	0.98

Table 5.2: The rate-range relationship and RAP of a commercial product [2]

actual data broadcast, each node that has its Forwarding flag activated will forward the message forward at its predetermined rate. In the next section, we shall see that most of the redundant transmissions (retained in CDS during Stage 2) are eliminated during this Tree-Construction stage.

Message Complexity: The maximum number of $RREQ$ messages sent in the network is contingent on the number of marked nodes chosen in earlier steps. The worst-case message complexity of the Tree-Construction stage is $O(md)$.

5.4 Simulation using idealized scheduler

We utilize an idealized scheduler (Section 4.3.3), along with ideal MAC assumptions (Section 4.2.2), for the results presented in this section. We compare the broadcast latency performance of our algorithms using 100 topologies of different network sizes (measured by the number of nodes) uniformly randomly distributed in an area of 1000^2 m^2. We normalize the broadcast latency results with the delay of Dijkstra's tree which is the shortest delay possible when there is no limit to the number of radios, channels and times a node can transmit a packet. Since determining the actual optimal is NP-hard, we use the Dijkstra's metric as a theoretical lower bound on the optimal achievable latency. Thus the minimum value of normalized delay is unity. The result that we will show is the average normalized broadcast latency over 100 network topologies. The transmission rate-range relationships depicted in Table 1.1 (obtained using Qualnet [1] for a 802.11b network) and Table 5.2 (obtained from a commercial product's [2] specifications) are assumed. The interference range is assumed to be 1.7 times the lowest transmission rate's range.

5.4.1 Rate-unaware vs. rate-aware distributed broadcast

We present the performance of our rate-aware distributed broadcast algorithm against the performance of rate-unaware distributed broadcast algorithm in Figures 5.3 and 5.4. The Wu-Li algorithm is an algorithm that does not take multi-rate capability into account during its operation, therefore, we would expect its performance to be poorer than MEW, MDW, with and without 'neighbor grouping', and MRRA algorithms, all of which are rate-aware algorithms. The performance results are shown

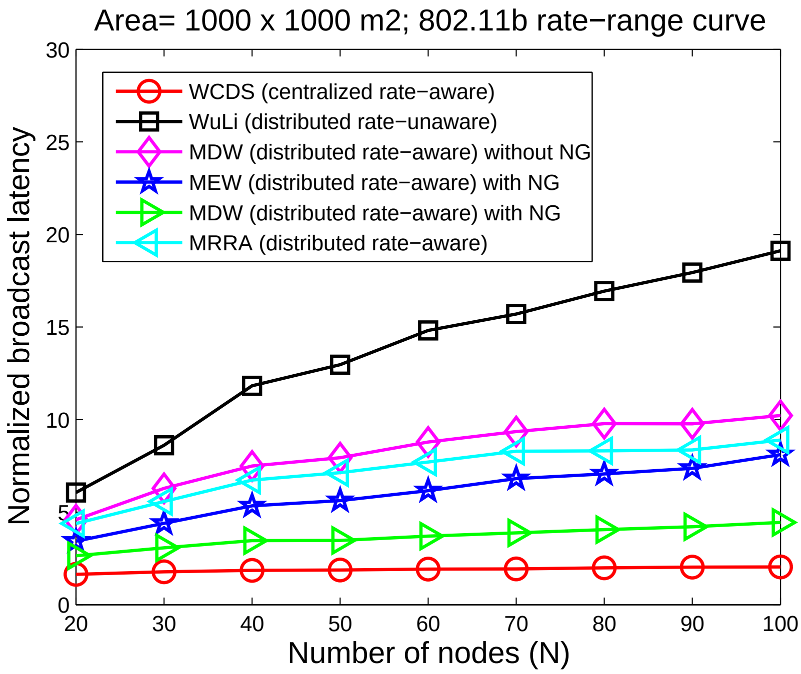


Figure 5.3: Normalized broadcast latency against varying number of nodes N (Area=1000 $\times$ 1000 m^2) for 802.11b rate-range curve [Table 1.1]

in Figures 5.3 and 5.4 for the rate-range curves in Table 1.1 and 5.2, respectively. It is observable that rate-aware broadcast algorithms have better performance than rate-unaware broadcast algorithms across the range of number of nodes (N) and for both rate-range curves. The performance of rate-unaware broadcasting is particularly poor for higher values of N. We can conclude therefore that Stage 2 of our broadcasting framework (i.e., the 'neighbor grouping' and 'rate maximization' stage) enables our algorithms to perform better than rate-unaware algorithms by maximizing transmission rates at the forwarding nodes, after grouping the neighboring nodes to minimize some redundancy.

5.4.2 Distributed versus centralized MLB algorithms

In this subsection, we compare the performance of our distributed MLB algorithms against the centralized MLB algorithm's (WCDS of Chapter 4) performance. The results of this comparison can be observed in Figures 5.3 and 5.4. We observe that the performance of WCDS, which is an example of a centralized multi-rate broadcast algorithm, is quite close to the 'optimal' value (Dijkstra tree assuming unlimited transmissions, and radio resources). As is to be expected, the performance of our distributed algorithm cannot match the performance of the centralized algorithm. The performance gap between WCDS and the MDW algorithm is, however, not large. The performance of MDW, in terms of broadcast latency, is better than MRRA's performance.

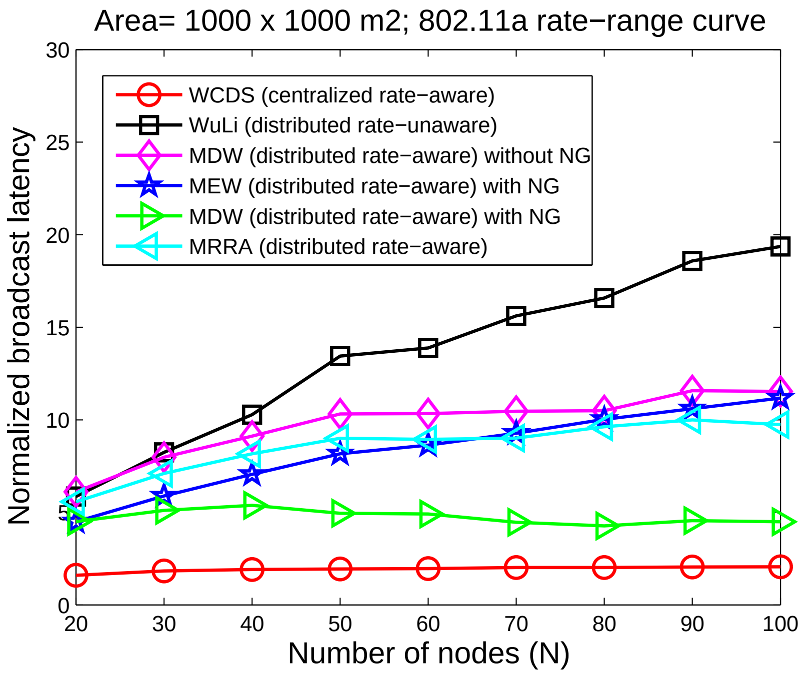


Figure 5.4: Normalized broadcast latency against varying number of nodes N (Area=1000 $\times$ 1000 m^2) for 802.11a rate-range curve [Table 5.2]

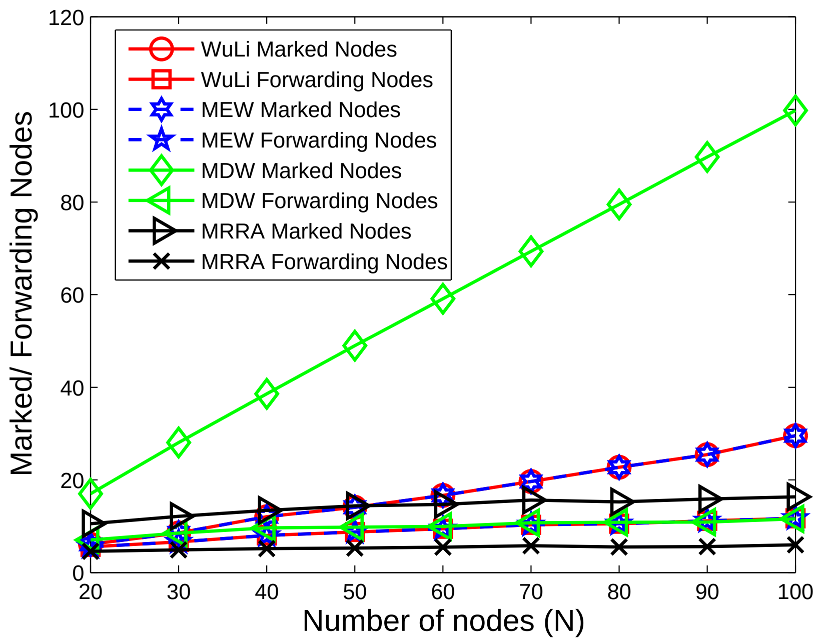


Figure 5.5: Number of marked-nodes/or forwarders against varying number of nodes N (Area=1000 $\times$ 1000 m^2) for our algorithms (assuming rate-range curve of Table 1.1)

5.4.3 Effects of 'delayed pruning' and 'Neighbor Grouping'

It should be observed in Figures 5.3 and 5.4 that delayed-pruning and 'neighbor grouping' substage improves the performance appreciably. Firstly, to see the effect of delayed pruning, we note that the performance of MDW (multi-rate *delayed-pruning* Wu-Li) with 'neighbor grouping' (NG) is better than the performance of MEW (multi-rate *expedited-pruning* Wu-Li) with NG, across the range of N for both the considered rate-range curves. Secondly, the effect of NG can be seen by seeing the improvement in MDW with NG over MDW without NG across the range of N for both the considered rate-range curves.

5.4.4 Number of marked nodes and forwarders

We make the distinction that marked nodes are the nodes marked for transmission before Stage 3, whereas, the nodes actually chosen to forward after Stage 3 are referred to as forwarders. The graph depicting number of marked nodes and forwarders for the different algorithms is depicted in Figure 5.5. It is interesting to note the effect of delayed-pruning on the number of marked nodes (or, the CDS set); although, the delayed pruning produces better broadcast latency results, it does this at the expense of a bigger CDS. Whereas MEW prunes away a substantial portion of the CDS before invoking the 'rate maximization' process, MDW does not have this explicit pruning step before 'rate maximization'. This implies that relatively few nodes are able to prune themselves completely during 'rate maximization' in Stage 2. More importantly with delayed pruning (and a larger CDS), there are more opportunities to increase transmission rates as a marked node has more neighboring marked nodes to export nodes to. Note that the actual nodes that would transmit for MDW are a lot lesser than the marked nodes (or, the size of CDS). This is because Stage 3 will eliminate the redundancy in the transmissions and ensure that the number of nodes that will actually forward is not large. The number of forwarders (after Stage 3) of MDW is comparable, though still slightly higher, to the number of forwarders for MEW.

5.5 Simulation in Qualnet using 802.11b MAC

In this section, we present results of our simulations using the Qualnet [1] simulator to evaluate our algorithms' performance when a practical MAC scheduler is being used. We have used IEEE 802.11b as our MAC scheduler, and PHY 802.11b, which uses a pre-configured BER-based packet reception model, at the physical layer. The IEEE 802.11 MAC with Distributed Coordination Function (DCF) was chosen as the medium access control protocol. All default parameters are assumed unless stated otherwise.

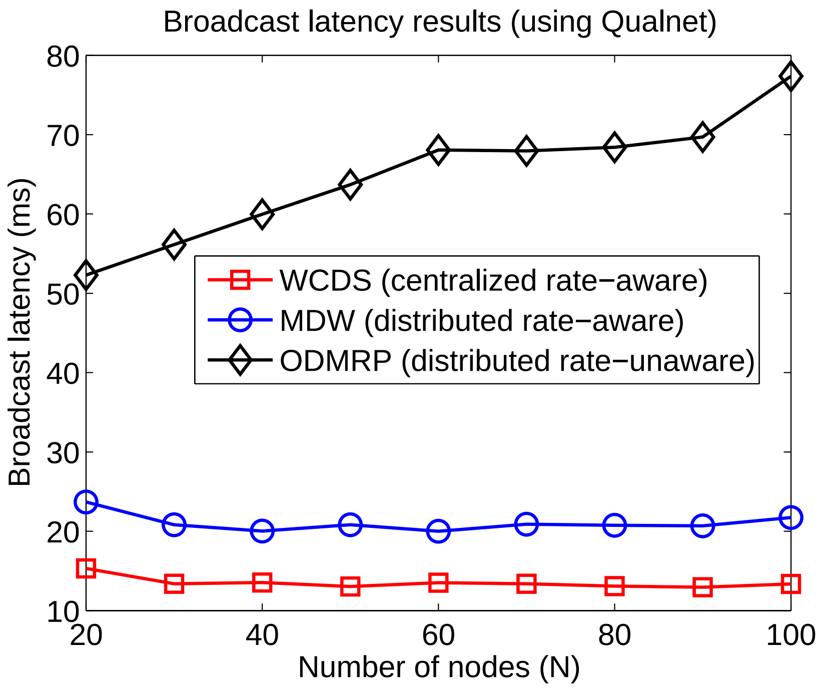


Figure 5.6: Normalized broadcast latency against varying number of nodes N (Area=1000 $\times$ 1000 m^2) using 802.11b simulation in Qualnet

5.5.1 Distributed versus centralized MLB algorithms

We have used MDW (with NG) as representative of our distributed MLB algorithm and compare it against WCDS (a centralized MLB algorithm) and ODMRP (a distributed rate-diversity unaware algorithm). Note that ODMRP is neither a MLB algorithm per se, nor is it a rate-diversity aware protocol (all its transmission are assumed to be at the lowest rate of 1 Mbps), we have included it for comparison nonetheless as a representative rate-diversity unaware distributed broadcast algorithm. The broadcast latency results (in milliseconds) of the simulations are shown in Figure 5.6. The results in Figure 5.6 are consistent with the results discussed earlier; MDW improves the performance of ODMRP across all values of N but does slightly worse than the centralized algorithm.

5.6 Conclusions

In this chapter, we have presented three localized algorithms to construct broadcast trees in SR-SC multi-rate WMNs distributedly. These algorithms incorporate techniques to exploit the rate-diversity of the underlying network, as well as the WBA of SR-SC multi-rate WMNs. More importantly, we have demonstrated that the performance gap between our distributed algorithms, which require limited topology information, and centralized algorithms of Chapter 4, which incur large operational overhead and require global topology information, is not large for practical purposes.

Chapter 6

Alternative MLB Framework for SR-SC Multi-rate WMNs

6.1 Introduction

In Chapters 4 and 5, we have presented centralized and distributed algorithms for the minimum-latency broadcast (MLB) problem in single-radio single-channel (SR-SC) multi-rate WMNs. While presenting these algorithms, we had assumed a framework where WMN nodes can adjust link-layer multicast transmission rate for each link-layer frame. We refer to this framework as a "*fully multi-rate multicast*" (FMM) framework. In this chapter, we proposed a new framework which we call the "*single best-rate multicast*" (SBM) framework that exploits the link-layer rate-diversity by enabling each WMN to decide, depending on its topological properties, a single transmission rate for all its link-layer data multicasts. Although, algorithms like WCDS and MDW (FMM broadcast heuristics proposed in Chapters 4 and 5 respectively) return impressive performance, employing the alternative SBM framework is attractive since it can eliminate some undesirable features of practical multi-rate media-access-control (MAC) protocols. We shall propose techniques in this chapter that a SBM framework can employ to determine the "best" link-layer transmission rate for all link-layer multicasts. We will also propose a heuristic broadcast solution which utilizes the SBM framework, and returns comparable performance to FMM broadcast heuristics, especially for dense networks, by exploiting the rate-diversity of SR-SC multi-rate WMNs.

The main contributions of this chapter are highlighted below:

1. *We propose a novel broadcasting framework called SBM (Single Best-Rate Multicast) that exploits the link-layer rate-diversity of SR-SC multi-rate WMNs by enabling each WMN to decide, based on its topological properties, a single transmission rate to use for all its link-layer data multicasts.*

2. *We propose a SBM centralized MLB heuristic, called SCDS, for SR-SC multi-*

rate WMNs. We show through analysis that the SBM heuristic SCDS can match the performance of the more expensive (implementation-wise) FMM heuristic WCDS especially for dense networks.

3. *We propose techniques that can be used to determine the 'best' rate that should be used in such a SBM framework.*

Chapter Outline: We provide motivations for providing a new broadcast framework for MLB broadcast in SR-SC multi-rate WMNs in Section 6.2. We detail our network and interference model in Section 6.3. We detail our three-staged SBM broadcasting framework and also propose a heuristic that uses this framework in Section 6.4. We provide a performance evaluation of our presented heuristic in Section 6.5 and Section 6.6 for the cases of idealized and practical decentralized MAC scheduler, respectively. We conclude this chapter in Section 6.7.

6.2 Motivation of a new framework

While using multi-rate multicast is desirable in an ideal case, it has been observed in practical MACs like 802.11 that the choice of a low transmission rate, even by an individual node, may substantially lower the total throughput achieved in that region. *This is due to the well-known paradigm of fairness in access attempts rather than bandwidth* [82] [83]. Hence, it is worth studying the impact of broadcasting, in an ideal setting, using a single *'best'* rate as opposed to the more powerful paradigm of broadcast transmission by different nodes at different rates. In particular, if it turns out that a single-rate broadcast strategy can provide latencies fairly close to those provided by the multi-rate case, then an approach based on adopting a single system-wide link-layer broadcast rate may become worthy of consideration. Using the SBM framework can also simplify our broadcasting algorithms—e.g., the *'multicast grouping'* stage of Chapter 4, which caters to the possibility of a transmitting node covering its neighbors in multiple transmissions (at different rates), is eliminated when SBM framework is used.

6.3 Network and interference model

We assume the SBM broadcasting framework in which each WMN decides, based on its topological properties, a single link-layer rate to use for all broadcasts. Our network model is similar to that described in Chapters 4 and 5 (except that we use the SBM framework). We reproduce the relevant details here for completeness. Each node in the network can transmit at multiple-rates. Using the Qualnet simulator [1] as a reference (assuming a two-ray propagation model), we obtain the transmission rate versus transmission range (*rate-range*) relationship (for 802.11b) shown in the first two columns of Table 1.1. We also employ an alterative rate-range relationship, shown in the first two columns of Table 6.1, of a rate-range relationship for 802.11a obtained from the Qualnet simulator to perform sensitivity analysis of the broadcast performance with different rate-range relationships.

We use an undirected graph $G = (V, E, L)$ to model the given mesh network topology, where V is the set of vertices, E is the set of edges and L is the set of weights of edges in E. The vertex v in V corresponds to a wireless node in the network with a known location. An undirected edge (u, v), corresponding to a wireless link between u and v, is in the set E *if and only if* $d(u, v) \leq r$ where $d(u, v)$ is the Euclidean distance between u and v and r is the range of the *lowest-rate* transmission. The latency of a link $l(u, v)$ is the transmission latency of the quickest transmission rate that can be supported between nodes u and v. The set L contains the latencies of all links in E. We use the same notation to refer to *vertices and nodes*, to *edges and links*, and to *weight of edges and latency of links* without confusion, the usage being clear from the context.

Interference Model: We use a generalized conflict graph based on transmissions to model the effects of wireless interference between different multicast transmissions in SR-SC multi-rate WMNs. The conflict graph indicates which transmissions mutually interfere and hence cannot be active simultaneously. A transmission b_i interferes with a transmission b_j if the receivers of transmission b_i are within the interference range of the transmitter of b_j or *vice-versa*. The transmissions b_i and b_j do not interfere otherwise.

6.4 Heuristic solution using the SBM Framework

In this section, we present heuristic algorithms that solve MLB problem, using the SBM framework, in SR-SC multi-rate WMNs. Broadly speaking, such heuristic algorithms must take *three* important decisions at each node. *Firstly*, it has to determine the 'best' transmission rate to use for *all* link-layer broadcasts (*this stems from our design choice to have only one broadcast rate*). *Secondly*, it has to decide whether a node should transmit (i.e., be a non-leaf node in the broadcast tree) or not. *Lastly*, each node's transmissions must be scheduled, while ensuring simultaneous transmissions (at different nodes) do not interfere, to minimize the broadcast delay. These *three* logically independent stages are discussed in more detail next.

1) ***The 'best' link-layer multicast rate selection***: Since we have taken the design decision to use the SBM framework to simplify the broadcast heuristics and their implementation, we need to determine the single 'best'link-layer multicast rate. This rate, to be used for all link-layer multicast, is determined by each WMN, during the first stage of our solution, according to its topological properties.

2) ***Topology Construction***: The aim of this stage is to compute a broad-

cast tree (or a spanning tree of the given topology) T that exploits the WBA, the multi-rate transmission capability and the plurality of radio interfaces and channels available. The transmitting nodes and the children/parent relationships between different nodes are all decided during this stage.

3) ***Transmission Scheduling***: While the non-leaf nodes (transmitting nodes) of the tree are determined during the 'topology construction' stage, *the 'transmission scheduling'* (or simply, scheduling)*, determines the exact timing of the various transmissions.* The scheduling of the transmissions is done according to the following constraints: firstly, a node must transmit only after receiving its parent's transmission and secondly, the interfering transmissions must not be scheduled simultaneously[1].

We note that the 'topology construction' and 'transmission scheduling' stages are essentially the same as those described in Chapter 4. The difference in the 'topology construction' stage is that now only the 'best' rate (decided in Stage 1) can be chosen as the transmitting rate unlike Chapter 4's 'topology construction' stage which could choose any rate. There is no difference between this chapter's 'transmission scheduling' stage and that of Chapter 4, and the same scheduling algorithm is assumed.

6.4.1 Determining the single 'best' link-layer multicast rate and "RAP" formulation

We point out a key finding of Chapter 4 that a transmission rate's broadcast efficiency (in reducing broadcast latency) can be predicted reasonably by the product of the transmission rate and its transmission coverage area (*rate-area product* or *RAP*). We propose using a similar approach for predicting a particular transmission rate's broadcast efficiency when using the SBM framework. The RAP values for different transmission rates of the rate-range relationship of 802.11b in Qualnet [1] are provided in Table 1.1. Similarly, the RAP values for transmission rates of our alternative rate-range relationship are provided in Table 6.1. As a general rule-of-thumb, for the FMM framework, a transmission rate that has a higher RAP is more broadcast-efficient (Chapter 4). We will now investigate if this conjecture still holds for the SBM framework.

We propose two methods of determining the 'best' link-layer multicast rate. For any given WMN, let R denote the set of transmission rates, which if used as the link-layer rate for all multicast, returns a connected network. In the first method, we use the highest link-layer multicast rate in R as the chosen 'best' rate. We call the transmission rate calculated by this method as the *"QC"* (*quickest-connected*) rate.

[1]We are conceptually assuming a centralized idealized scheduler, as was the case in Chapter 4, for the 'transmission scheduling' stage. We will also use 802.11 MAC of Qualnet simulator for comparison and validation of results (Section 6.6).

In the second method, we use the transmission rate in R that has the highest RAP value of all rates in R. We call the transmission rate calculated by this method as the "*HRC*" (*highest-RAP-valued connected*) rate. We will present results and analysis of these methods in Section 6.5.

6.4.2 Topology Construction

In this section, we present a heuristic algorithms for the 'topology construction'. Our presented algorithm is called "*single-best-rate connected dominating set*" (SCDS) and is presented next in Section 6.4.2. The inputs to the SCDS algorithms is the topology $G = (V, E, L)$, broadcast source s in V, the 'best' broadcast transmission rate $\hat{l}$ (chosen as described in the previous section).

Single-best-rate Connected Dominating Set (SCDS)

The 'topology construction' for the SBM framework is greatly simplified compared to FMM framework (Chapter 4, 5, 9 and 10). To illustrate this, we note that the Weighted Connected Dominating Set (WCDS) problem (Chapter 4)—which essentially is finding a connected dominating set that covers all nodes with minimum (latency) weighted sum—reduces to a problem of finding the Minimum Connecting Dominating Set (MCDS) when SBM framework is assumed. Unfortunately, MCDS in general graphs is also an NP-hard problem [31]. However, by assuming SBM framework, the extra processing of 'multicast grouping' stage as required by FMM framework heuristics is not required as shall be explained later.

Algorithm 6 The SCDS algorithm

Input: $[s, G = (V, E, L), \hat{l}]$
$\mathcal{R} \leftarrow \{s\}$
while $(V \backslash \mathcal{R} \neq \emptyset)$ **do**
  $(\hat{n}, \hat{l}) = \arg\max_{n \in \mathcal{R}} f(n, \hat{l})$
  (**where** $f(n, \hat{l}) = (|N(n, \hat{l}) \backslash \mathcal{R}|)$)
  let the transmission of $\hat{n}$ be represented by t
  $A \leftarrow N(\hat{n}, \hat{l}) \backslash \mathcal{R}$;
  $P_{SCDS}(A) = \hat{n}$; $L_{SCDS}(A) = \hat{l}$;
  $\mathcal{R} \leftarrow \mathcal{R} \cup A$
end while
Output: $[P_{SCDS}, L_{SCDS}]$

We use a simple greedy heuristic called SCDS for constructing the broadcast topology. The SCDS algorithm is the SBM framework adaptation of the WCDS algorithm (which assumes FMM framework). The main difference between SCDS and WCDS is that whereas WCDS could choose any transmission rate at a node (Algorithm 1 in Chapter 4), SCDS is limited to using only the 'best' rate ($\hat{l}$) decided in Stage 1 as described in Section 6.4.1. The SCDS algorithm is shown in Algorithm

6, The algorithm starts by making the source node s eligible to transmit. This is done by moving s to the set $\mathcal{R}$ which keeps track of the eligible nodes (nodes that have received the transmission already and are eligible to transmit). A transmission $(n, \hat{l})$ represents the transmission of node n (using the 'best' link-layer multicast rate $\hat{l}$). All eligible transmissions ($\forall n \in \mathcal{R}$) are given a 'priority' according to the number of new nodes (that have not yet received the transmission: $|N(n, \hat{l}) \backslash \mathcal{R}|$ *or* A) it covers. The algorithm works iteratively, and in each round finds the transmitting node $\hat{n}$ that can cover maximum number of nodes that have yet not received. The algorithm completes its execution when all the nodes have been covered, i.e. when $V \backslash \mathcal{R} = \emptyset$. The algorithm returns the sets P_{SCDS} and L_{SCDS}, where $P_{SCDS}(v_i)$ is the parent node of v_i, and $L_{SCDS}(v_i)$ is the latency of the link connecting v_i and $P_{SCDS}(v_i)$. The SCDS tree can be constructed from these.

6.4.3 Transmission Scheduling

In the SBM framework, the transmitting nodes are determined during the 'topology construction' stage (Stage 2) while the link-layer rate to use for all multicast is determined during Stage 1. Since, in the SBM framework, each node transmits only once using a pre-determined rate, there is no requirement for any transmitting node to perform multiple distinct-rate transmissions for the same packet. This eliminates the need of 'multicast grouping' stage of FMM framework heuristics and greatly simplifies the operation of the 'transmission scheduling' stage. The 'transmission scheduling' (Stage 3) of our SBM framework determines the exact timing of the various transmissions. The scheduling of the transmissions must respect the following constraints: firstly, a node must transmit only after receiving its parent's transmission and secondly, the interfering transmissions must not be scheduled simultaneously.

We have assumed the centralized scheduler proposed in Section 4.3.3 for 'transmission scheduling' for the results presented in Section 6.5. We have also utilized 802.11 MAC of the Qualnet simulator [1] for the performance evaluation of our algorithms which we present in Section 6.6.

6.5 Simulation using idealized scheduler

In this section, we present simulation results assuming the idealized scheduler of Section 4.3.3, along with ideal MAC assumption (Section 4.2.2), to evaluate the performance of our algorithms. We employ the rate-range relationships derived from Qualnet for 802.11b (Table 1.1) and 802.11a (Table 6.1) in our study. The broadcast latency of our heuristics are all shown by normalizing it against the broadcast latency

Transmission rate (Mbps)	Transmission range (m)	RAP (Mbps-km^2)
6	424.3	3.393
12	357.0	4.805
24	252.7	4.826
48	79.6	0.955

Table 6.1: The rate-range relationship and RAP values for a 802.11a network using the Qualnet simulator [1]

performance of the Dijkstra's tree[2] when the idealized scheduler of Section 4.3.3 is used. When we employ the 802.11 MAC of Qualnet simulator [1] for scheduling (see Section 6.6), we will provide absolute values of broadcast latency.

The results of broadcast latency are directly applicable to low throughput data flows (e.g., control traffic) as the metric applies to a *single* packet. However, for higher throughput data flows, an important metric is the maximum achievable end-to-end throughput. We essentially employ a generalization of the method used in Chapter 4 for throughput calculation in SR-SC multi-rate networks.

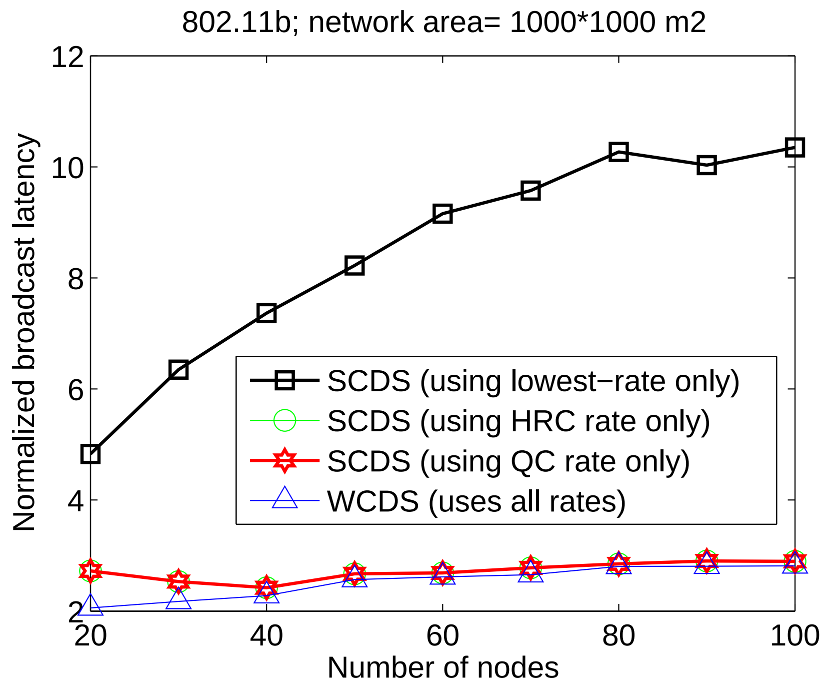
6.5.1 Performance of SCDS with increasing node density

Referring to Figures 6.1(a), 6.1(b) and 6.2(a), 6.2(b), which display latency and throughput results for rate-range relationship of Table 1.1 and 6.1 respectively, we note that our SBM heuristic, SCDS, performs comparably to our FMM heuristic which presented in Chapter 4, WCDS, as node density increases. It is to be noted that in sparse WMN, SCDS might be hindered by its design choice of SBM framework of only employing a single link-layer multicast rate as the network might have to decide a lower rate as the 'best' multicast rate to maintain connectivity. This can lead to worse performance for SBM heuristics compared to FMM heuristics in sparse networks. This is, however, not a problem for dense networks. Also, as we shall see in the next subsection, the performance of SCDS is also affected by the method use for calculation of the

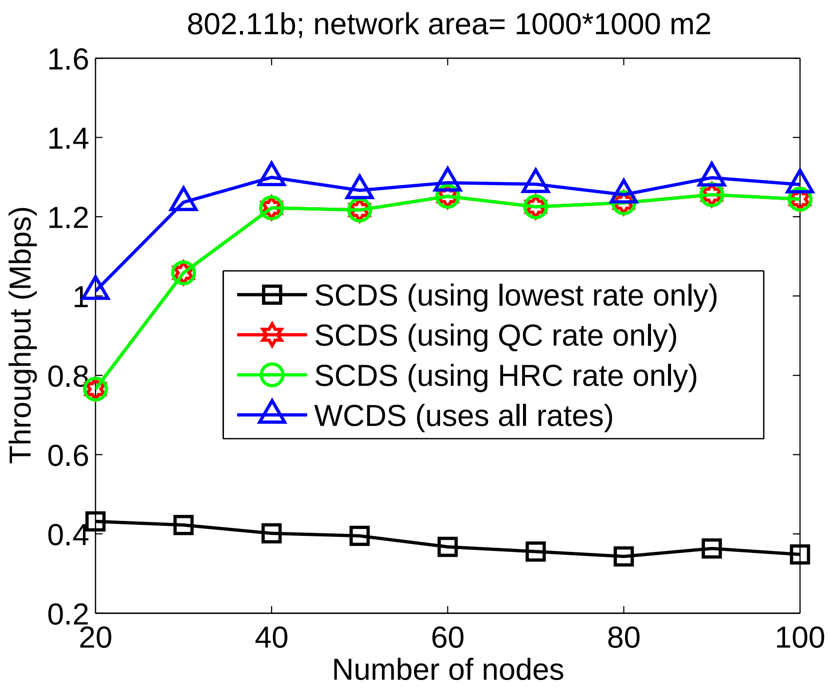
6.5.2 Methods for calculating the "best" multicast rate

We now evaluate the viability of the methods, proposed in Section 6.4.1, for determining the 'best' link-layer multicast rate . Considering the rate-range relationship shown in Table 1.1, since the RAP values are monotonically increasing with increasing rate, both the 'quickest' and *'HRC'* (highest RAP-valued connected) rate methods give the same rate. Thus SCDS using either *'HRC'* (highest RAP-valued connected) rate or *'QC'* (quickest connected) rate return identical performance.

[2]Since determining the actual optimal is NP-hard, we use Dijkstra tree's performance as a theoretical lower bound on the optimal achievable latency in a corresponding wired network.

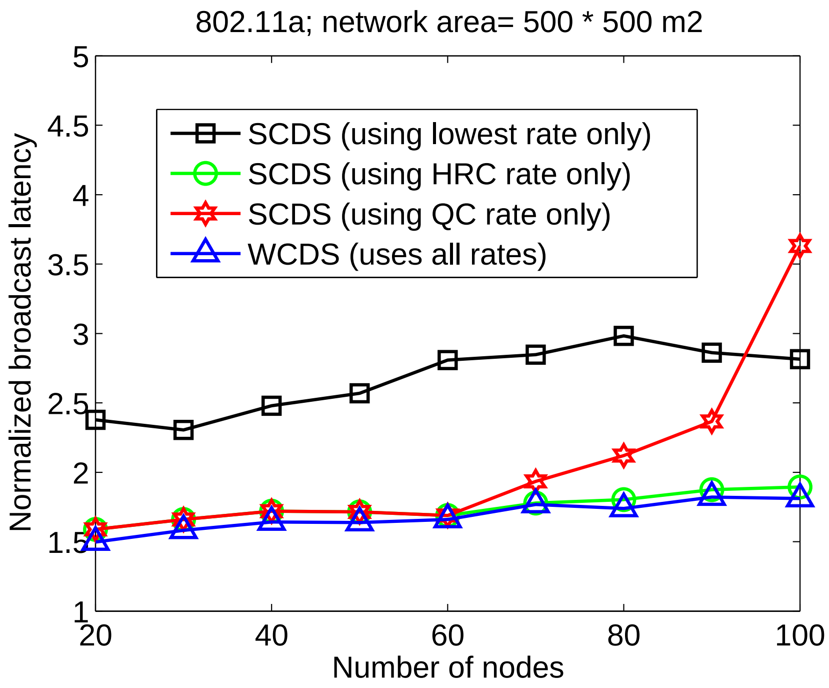


(a) Normalized broadcast latency

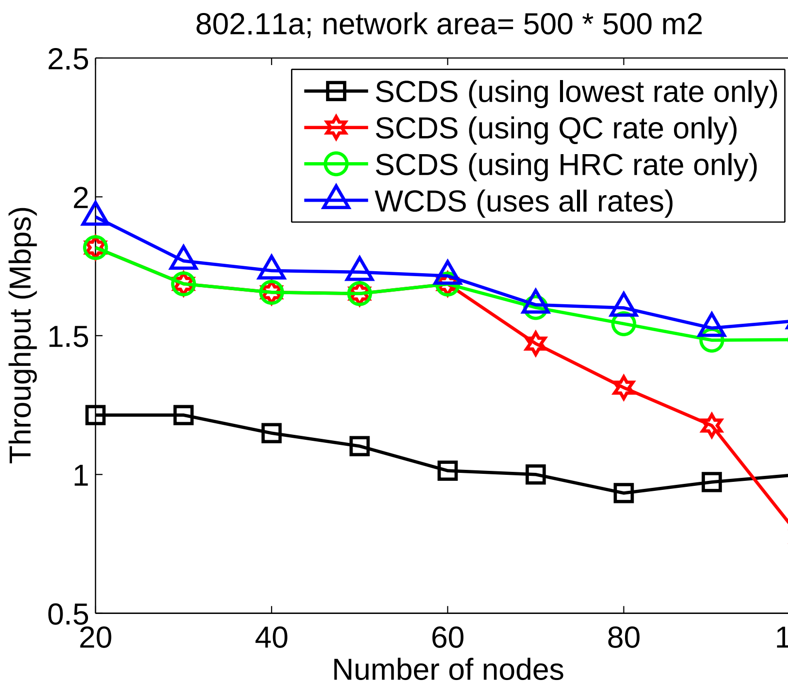


(b) Throughput (in Mbps)

Figure 6.1: Broadcast latency and throughput results using 802.11b networks (rate-range relationship of Table 1.1)



(a) Normalized broadcast latency



(b) Throughput (in Mbps)

Figure 6.2: Broadcast latency and throughput results using 802.11a networks (rate-range relationship of Table 6.1)

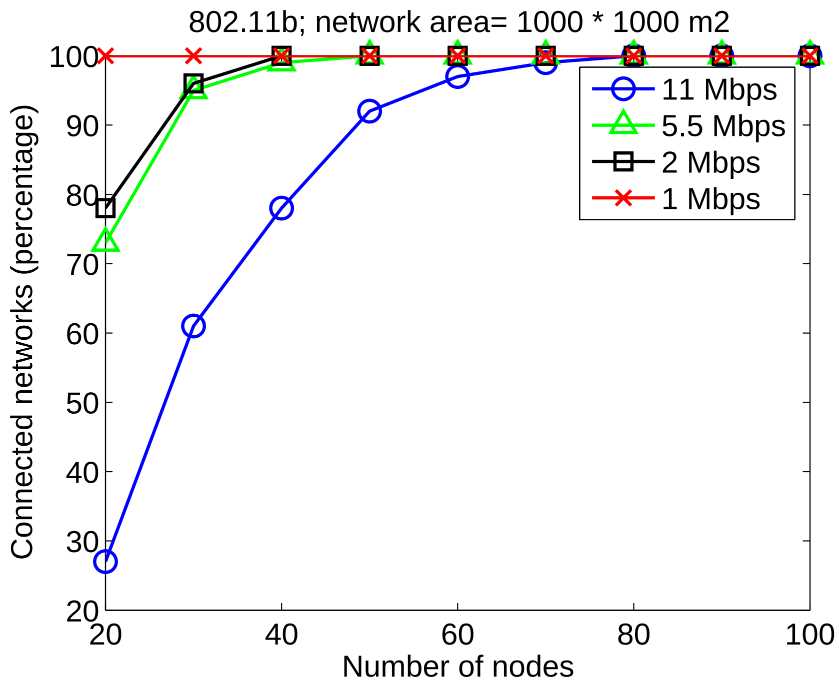
This can be observed in Figures 6.1(a) and 6.1(b) where the broadcast latency and throughput performance are compared for varying number of nodes in an area of 1000^2 m^2. When SCDS heuristics solely uses the lowest possible transmission rate (same as algorithm CDS of Chapter 4), both the latency and throughput results are significantly worse; this reinforces our assertion that rate-unaware broadcast is non-optimal in multi-rate WMNs.

We now consider the rate-range relationship, shown in Table 6.1, which is interesting since the RAP values do not monotonically increase with increasing rate; in fact, the quickest rate of 48 Mbps has, amongst all rates, the lowest RAP value. For our proposition of using RAP values of different rates for predicting broadcast performance to be satisfied, the *'HRC'* method should outperform the *'QC'* method. The broadcast latency and throughput results for 802.11a network (rate-range relationship of Table 6.1) are shown in Figures 6.2(a) and 6.2(b). The *'HRC'* method does indeed perform better than the *'QC'* method especially for dense networks. For low node density, *'HRC'* rate is likely to be the same as the *'QC'* rate according to the connectivity data in Figure 6.3(b). It is seen in Figure 6.3(b) that it is only for dense networks ($N > 60$ in an area of 500^2 m^2) that a connected network results while using a rate higher than the *'HRC'* rate. It can be seen in Figures 6.2(a) and 6.2(b) that in such cases the *'HRC'* method of selecting rate is superior. Interestingly, when the node density is sufficiently high, the *'QC'* method (using the quickest rate that returns a connected network) returns results that are even worse than the method of always using the lowest rate.

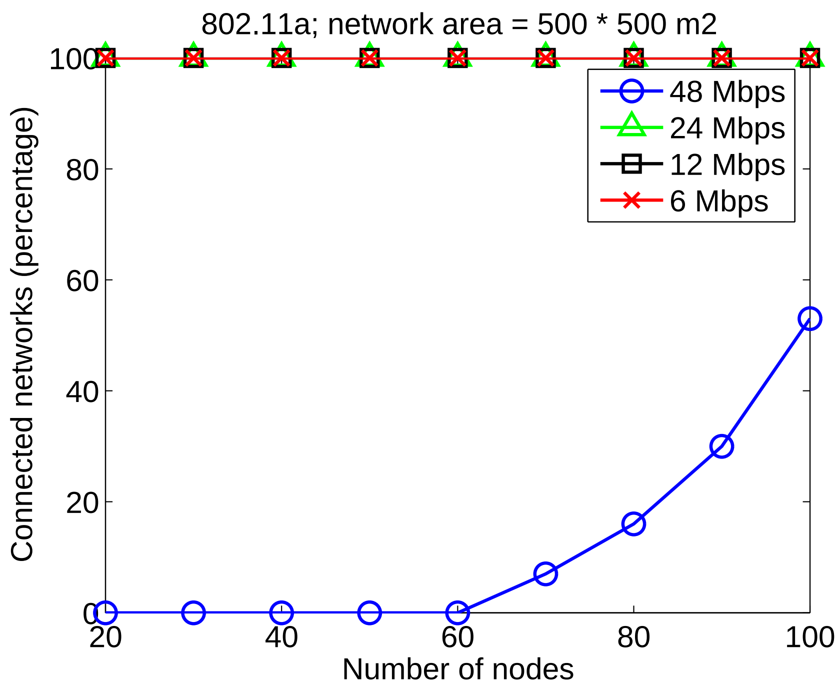
We conclude on the basis of these experimental results that using the *'HRC'* rate method is an efficient way of selecting the 'best' rate to be used by SBM framework broadcast heuristics.

6.5.3 Sensitivity of results to rate-range relationship

We present the sensitivity analysis of our broadcasting framework to the rate-range relationship of the WMN using the rate-range relationships shown in Tables 1.1 and 6.1. We have observed that, for both considered rate-range relationships, the performance of our SBM broadcast heuristic (SCDS) is comparable to the performance of our sample FMM broadcast heuristic (WCDS) especially for dense WMN. For the rate-range relationship as shown in Table 1.1, the broadcast latency, for varying node density, is depicted in Figure 6.1(a) and the throughput results in Figure 6.1(b). Similarly, for the rate-range relationship as shown in Table 6.1, the broadcast latency result is shown in Figure 6.2(a), whereas the throughput result is shown in Figure 6.2(b). Both the latency and throughput results, for both rate-range relationships, show a similar trend where the SBM broadcast heuristics perform comparably to FMM broadcast heuristics, especially at high node densities.



(a) to be used in conjunction with Figures 6.1(a) and 6.1(b)



(b) to be used in conjunction with Figures 6.2(a) and 6.2(b)

Figure 6.3: Probability of a connected network using 802.11b (Table 1.1) and 802.11a networks (Table 6.1)

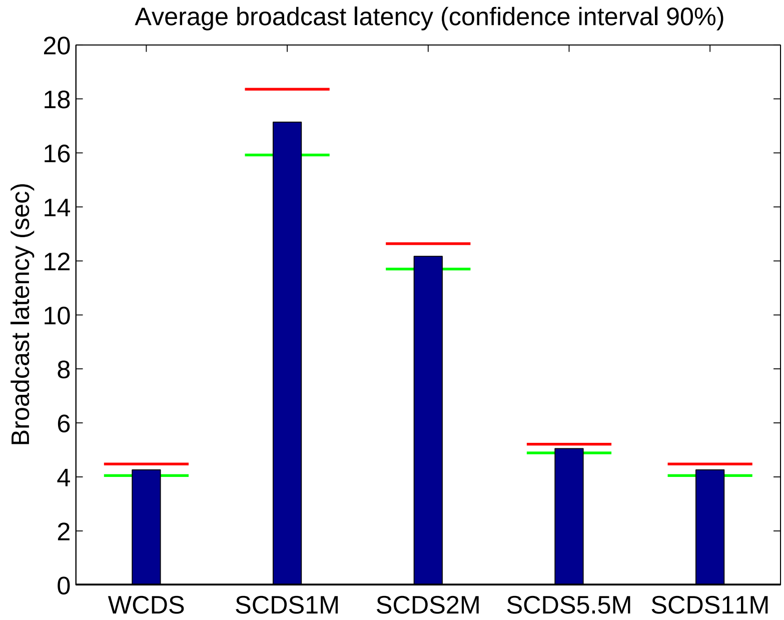


Figure 6.4: Performance of SCDS with different rates compared to WCDS

6.6 Simulation in Qualnet using 802.11b MAC

In this section, we have performed simulations on the Qualnet [1] simulator to see the performance of SCDS algorithm, as compared with WCDS (Chapter 4) algorithm, for different 'best' transmission rates assuming a practical decentralized MAC scheduler. We implemented PHY 802.11b at the physical layer, which uses a pre-configured BER-based packet reception model. The IEEE 802.11 MAC with Distributed Coordination Function (DCF) was chosen as the medium access control protocol. All default parameters are assumed unless stated otherwise.

To study the viability of using RAP as a rule-of-thumb for measuring the broadcast efficiency of different rates for the SBM framework, we simulate 100 random 70-node 802.11b network topologies in an area spanning 1000^2 m^2 where each node is equipped with a single radio interface. The rate-range relationship for a 802.11b network assuming default parameters of Qualnet simulator is depicted in Table 1.1. The results shown in Figure 6.4 must be observed together with Figure 6.3(a) which displays the probability of having a connected network for different rates of 802.11b when used (by themselves) for all broadcast traffic using a SBM framework. The lowest rate (1 Mbps) has the maximum connectivity probability of 1 since we only consider networks that are connected using the lowest rate. Figure 6.3(a) indicates that connectivity probability, using a particular rate, decreases with increasing rates. We note that the quicker rates (e.g., 11 Mbps in Figure 6.3(a)) have very low connectivity probability. We compare WCDS with SCDS which uses the rate 1, 2, 5.5 and 11 Mbps (which are in Figure 6.4 called SCDS1, SCDS2, SCDS5.5 and

SCDS11, respectively). To estimate the broadcast latency, we have used an interval estimate with the confidence interval of 90%. It is observed that the higher-RAP rates perform better with SCDS using the 'best' rate of 11 Mbps (SCDS11 in Figure 6.4) almost matching the broadcast latency performance of WCDS. The Qualnet simulation results corroborate the idealized scheduler results (that were presented in the previous section), and show that RAP is a good predictor of the efficiency of a particular rate for broadcast; the results also indicate that SBM heuristics can match the performance of FMM heuristics especially when the 'best' rate is chosen sensibly.

6.7 Conclusions

In this chapter, we have proposed a new broadcast/multicast framework called SBM that uses a single "best" link-layer rate for all multicast. We have proposed a broadcast heuristic called SCDS for this framework. We have also detailed specific techniques that can be used to select the "best" rate to be used for all multicast. Our analysis of SBM framework shows that its performance in dense network settings is comparable to the performance of the more powerful FMM framework that can adapt the link-layer multicast rate of each frame.

Although the single layer-rate multicast approach appears attractive, there are some important practical issues to be resolved: the SBM approach requires centralized knowledge of the entire topology; secondly, SBM framework requires improvement to make it less vulnerable to dynamic topologies. These directions require future work.

Chapter 7

Summary of Results

This chapter serves as a conclusion to Part I, "Improving broadcast performance of single-radio single-channel multi-rate WMNs" of this thesis. We will revisit now some of the research objectives discussed in Section 1.4 that are relevant to this part of the thesis.

- *Study the minimum-latency-broadcast (MLB) problem for multi-rate WMNs. Provide specific MLB algorithms that can improve the broadcast latency performance of single-rate broadcast algorithms proposed in literature.*

  In Chapter 4, it was shown that the MLB problem for multi-rate WMNs is NP-hard, thus precluding the design of an optimal algorithm that can scale to large networks. We proposed a centralized heuristic MLB solution called *'WCDS'* for SR-SC multi-rate WMNs that exploits the network's rate-diversity, the wireless broadcast advantage (WBA), and also the degree-of-freedom of performing multiple distinct-rate transmissions for the same broadcasted packet. It was shown that by exploiting both multiple transmission rates and WBA, significant reduction results in broadcast latency compared to the case of always using the lowest transmission rate. For example, based on simulations using typical 802.11-based values, the use of our rate-aware WCDS heuristic results in a $\sim$ 3-6 fold reduction in the broadcast latency compared to the CDS algorithm that always performs link-layer broadcasts at the lowest rate. It was observed, however, although performing multiple distinct-rate transmissions for the same broadcasted packet can improve performance, the performance gain was not frequent and significant; thus, dropping this feature is justified if its implementation is expensive.

  In Chapter 5, we proposed a distributed and localized MLB heuristic called *'MDW'* for SR-SC multi-rate WMNs that exploits the network's rate-diversity, as well as its WBA. We ignored the degree-of-freedom of multiple distinct-rate transmissions for the same broadcasted packet since its performance benefit did not merit the complexity of its implementation. We have also proposed two other algorithms called *'MEW'* and *'MRRA'* in this chapter. All these presented algorithms incorporate techniques to exploit the rate-diversity of the underlying network as well as the WBA of SR-SC multi-rate WMNs. We have demonstrated that the performance gap between our distributed algorithms, which require limited topology information, and Chapter 4's centralized algorithms, which requires global topology information, is not large for practical purposes.

In Chapter 6, we proposed a new broadcast framework called SBM that uses a single *'best'* link-layer rate for all broadcast unlike the more powerful FMM framework that was used in Chapter 4 and 5 which allows any link-layer rate to be used. We study the impact of broadcasting using a single *'best'* rate as opposed to the more powerful paradigm of broadcast transmission by different nodes at different rates. In particular, if a single-rate broadcast strategy can perform comparably with a fully multi-rate strategy, then adopting it becomes worthy of consideration. We have a proposed a heuristic 'SCDS' for the SBM framework in Chapter 5, along with simple techniques that can be used to select the "best" rate to be used for all multicast. Our analysis of SBM showed that, in dense settings, its performance is comparable to the more powerful FMM framework that can adapt the link-layer multicast rate of each frame. Although the SCDS approach assuming the SBM framework appears attractive, there are some important practical issues to be resolved: the heuristic requires centralized knowledge of the entire topology; secondly, SBM framework requires improvement to make it less vulnerable to dynamic topologies. These directions require future work.

- *Provide general insights and rules that can enable improved broadcast performance by helping a protocol designer exploit the inherent rate-diversity of multi-rate WMNs.*

  In *Chapter 4*, we answered a number of fundamental design questions for multi-rate systems, such as:

  1. *Given a multi-rate given with* n *different rates, is it necessary to use all the* n *different rates?*
  2. *If not, which of the* n *different rates should we use and what is an efficient method to decide that?*

We proposed the use of the product of transmission rate and transmission coverage area (or rate-area product or RAP for short) as a measure of efficiency of a certain transmission rate. An important conclusion was that a higher rate is not necessarily preferable for broadcast. It was shown that a rate, notwithstanding how high it is, is broadcast efficient only if its RAP is high. Consequently, high transmission rates that do not have high RAP values should not be used for broadcast (see Figure 6.2(a) in Chapter 6). Thus, the RAP values can serve as a general rule-of-thumb that can predict the efficiency of a transmission rate for broadcast.

In Chapter 4, we also showed that although exploiting rate-diversity returns impressive results, the benefit of the degree-of-freedom enabled by rate-diversity of WMNs, which enables a node to reach its downstream nodes in distinct rate transmissions, is reaped in very few topologies and can be discarded by protocol designers.

Part II

Improving broadcast performance of multi-radio multi-channel multi-rate (MR^2-MC) WMNs

Chapter 8

Introduction to Minimum-Latency Broadcasting in MR^2-MC WMNs

8.1 Overview

This chapter serves as introduction to Part II, "*Improving broadcast performance of multiple-radio multiple-channel multi-rate WMNs*". The specific broadcasting problem addressed is the "minimum latency broadcast" (MLB) problem that was described in Section 1.2. In this part of the thesis, we consider the MLB problem for the case of multiple-radio multiple-channel multi-rate (MR^2-MC) WMNs in which each WMN node is equipped with multiple radio interfaces each tuned to different channel.

8.2 Outline

An outline of the remaining chapters of this part of the thesis is as follows:

Chapter 9: Centralized MLB Solution for MR^2-MC WMNs

In this chapter, we address the problem of minimizing the worst-case broadcast delay in MR^2-MC WMN using centralized algorithms with the assumption that global network topology information is available. The problem of '*efficient*' broadcast in MR^2-MC WMNs is especially challenging due to the radio-and-channel-diversity, that was discussed in Section 1.3.2, offered by such networks. The multi-rate transmission capability of WMN nodes, interference between wireless transmissions, and the hardness of optimal channel assignment add complexity to our problem. We present four heuristic algorithms in this chapter to solve the MLB problem for such settings and show that the best performing algorithms usually adapt themselves to the available radio interfaces and channels. We also study the effect of channel assignment on broadcast performance and show that channel assignment can affect the broadcast performance substantially. More importantly, we show that a channel assignment that performs well for unicast does not necessarily perform well for broadcast/multicast. To the best of our knowledge, this work constitutes the first contribution in the area of broadcast routing for MR^2-MC WMN.

Chapter 10: Distributed MLB Solution for MR^2-MC WMNs

We address the problem of minimizing the worst-case broadcast delay in MR^2-MC WMN in a distributed and localized fashion. Efficient broadcasting in such networks is especially challenging due to the desirability of exploiting the wireless broadcast advantage, the radio-and-channel-diversity, and the rate-diversity offered by these networks. We propose a framework that calculates a set of forwarding nodes and transmission rate at these forwarding nodes irrespective of the broadcast source. Thereafter, a forwarding tree is constructed taking into consideration the source of broadcast. Our broadcasting algorithms are distributed and utilize locally available information. To the best of our knowledge, this works constitutes the first contribution in the area of distributed broadcast in multi-radio multi-rate wireless mesh networks. We present a detailed performance evaluation of our *distributed and localized algorithm* and demonstrate that our algorithm can greatly improve broadcast performance by exploiting the rate-diversity and radio-and-channel-diversity of MR^2-MC WMNs and match the performance of centralized algorithms proposed in literature while utilizing only limited two-hop neighborhood information.

Chapter 11: Alternative MLB Framework for MR^2-MC WMNs

In Chapters 4, 5, 9 and 10, we have assumed a "*fully multi-rate multicast*" (FMM) framework in which nodes can adjust link-layer multicast transmission rate for each link-layer frame. In this chapter, we utilize the "*single best-rate multicast*" (SBM) framework (which was used earlier in Chapter 6) in which the link-layer rate-diversity is exploited by enabling each WMN to decide, depending on its topological properties, a single transmission rate for all its link-layer data multicasts. As was discussed in Chapter 6, although FMM framework returns impressive performance, employing SBM is attractive since it can eliminate some undesirable features of practical multi-rate Media Access Control (MAC) protocols. In this chapter, we propose methods to determine the "best" link-layer transmission rate for the SBM framework. We also propose two heuristic broadcast solutions, using the SBM framework, that can realize low-latency broadcast by exploiting inherent radio-and-channel-diversity in a multi-radio multi-channel WMN. Simulation results indicate that SBM broadcast and FMM broadcast heuristics perform comparably for MR^2-MC WMNs, especially when the node density is high.

Chapter 12: Summary of Results

We conclude the Part II of our thesis in this chapter by presenting the main results and findings of the research presented in Chapter 9 to 11.

Chapter 9

Centralized MLB Solution for MR^2-MC WMNs

9.1 Introduction

The work in this chapter builds upon our previous work on minimizing broadcast latency in a SR-SC *multi-rate* WMN (Chapters 4, 5 and 6) in which the concept of link-layer *multi-rate* broadcast was introduced through which a node can adjust its link-layer broadcast transmission rate to its neighbors. It has been shown in Part I of this thesis (Chapter 4) that broadcast in a multi-rate WMN has two features not found in a single-rate WMN. *Firstly*, if a node has to perform a link-layer broadcast to reach a number of neighbors, then its transmission rate is limited by the smallest rate on each individual link, e.g., if a node n is to broadcast to two neighboring nodes m_1 and m_2, and if the maximum unicast rates from n to m_1 and m_2 are, respectively, r_1 and r_2, then the maximum rate n can use is the minimum of r_1 and r_2. *Secondly*, for a multi-rate WMN, the broadcast latency can be minimized by exploiting an extra degree-of-freedom where some nodes transmit the same packet more than once, but at a different rate to different subsets of neighbors (called as 'distinct-rate transmissions'). Based on these insights, the WCDS algorithm was presented in Chapter 4 as a centralized heuristic solution for the MLB problem in SR-SC multi-rate WMNs; similarly, MDW was proposed in Chapter 5 as a distributed heuristic solution for the MLB problem in SR-SC multi-rate WMNs using similar concepts. Both these algorithms consider the WBA and the multi-rate capability of the network. The WCDS algorithm, in addition, also incorporates the possibility of multiple distinct-rate transmissions by a single node for the same packet.

It must be noted that SR-SC multi-rate WMNs is a special case of MR^2-MC WMNs that has no radio-and-channel-diversity, as SR-SC WMN nodes are equipped with only a single radio interface. Clearly, the general MLB problem for MR^2-MC WMNs is more difficult due to the additional complexity of incorporating radio-and-channel-diversity (also called interface-diversity) into algorithm design. We have noted the *NP-hardness* of the MLB problem for SR-SC single-rate WMNs [26], and for SR-SC multi-rate WMNs (Chapter 4); therefore, by extension, the MLB problem for MR^2-MC is at least *NP-hard*. We will therefore focus on heuristics as we did for SR-SC multi-rate WMNs in Part-I of this thesis. We note that any well-designed MLB heuristic for MR^2-MC WMNs should exploit such networks' interface-diversity, rate-diversity and WBA.

The main contributions of this chapter are highlighted below:

1. *We present the implications of radio-and-channel-diversity on broadcast performance, and demonstrate that the best performing heuristics for MR^2-MC WMNs generally adapt to the radio resources available and exploit the radio-and-channel-diversity of such networks.*

2. *We propose four FMM centralized MLB heuristics for MR^2-MC WMNs. The best performing of our algorithms, PAMT, exploits radio-and-channel-diversity of MR^2-MC WMNs, and performs close to the approximation of the theoretical optimal, resulting in latencies that are on average $\sim$ 10-20% higher.*

Chapter Outline: The rest of the chapter is organized as follows. The network and interference models are discussed in Section 9.2. Our heuristic solution to the *NP-hard* MLB problem for MR^2-MC WMNs, composed of the 'topology construction', 'multicast grouping' and 'transmission scheduling' stages, is discussed briefly in Section 9.3. We present an example network in Section 9.3.1 that we shall use throughout this chapter to illustrate how different algorithms of our framework work. We present the 'topology construction' stage in detail in Section 9.4, and also present four heuristic algorithms for this stage. We discuss the 'multicast grouping' and the 'transmission scheduling' stages in detail in Sections 9.5 and 9.6. The performance evaluation of our heuristic algorithms assuming an idealized MAC scheduler is provided in Section 9.7; performance evaluation using practical 802.11 MAC, as simulated in the Qualnet [1] network simulator, is presented in Section 9.8. We will conclude this chapter in Section 9.10.

9.2 Network and interference model

We assume the FMM broadcasting framework (introduced in Chapter 4) in which WMN nodes can adjust link-layer multicast transmission rate for each link-layer frame. We follow the notation introduced by Tang et al. [74] to represent the channel assignment. We use a network model similar to that described by Raniwala et al. [73]. We assume that each node in the network can transmit at multiple-rates. There are totally C non-overlapping orthogonal frequency channels, and each node is equipped with Q radio interfaces where $Q \leq C$. The Q radio interfaces have omni-directional antennas. In order to efficiently utilize the network resources, two radio interfaces at the same node are not tuned to the same channel. Using the Qualnet simulator [1] as a reference, we obtain the transmission rate versus transmission range relationship in Table 1.1 (Chapter 1), assuming a two-ray propagation model. The interference range in Qualnet is assumed to be 520 m. Note that the transmission range is a decreasing function of transmission rate as illustrated in Table 1.1.

An undirected graph $G_T = (V, E_T, L_T)$ is used to model the given mesh network topology *before* channel assignment, where V is the set of vertices, E_T is the set of edges and L_T is the set of weights of edges in E_T. The vertex v in V corresponds to a wireless node in the network with a known location. An undirected edge (u, v), corresponding to a wireless link between u and v, is in the set E_T *if and only if* $d(u, v) \leq r$ where $d(u, v)$ is the Euclidean distance between u and v and r is the range of the *lowest-rate* transmission. The latency of a link $l(u, v)$ is the latency of the 'fastest' transmission rate that can be supported between nodes u and v. The set L_T contains the latencies of all links in E.

Channel assignment: A channel assignment $\mathcal{A}$ assigns each vertex v in V, Q different channels denoted by the set: $A(v) = \{a_1(v), a_2(v), \ldots, a_Q(v) : a_i(v) \neq a_j(v), \forall i \neq j;\ a_i(v) \in C, \forall i\}$ where $a_i(v)$ represents the channel assigned to i^{th} radio interface at node v. The topology defined by $\mathcal{A}$ is represented by $G = (V, E, L, \Lambda)$ in the following natural way: There is an edge $e = (u, v, k)$ on channel $\lambda(e) = k$ between nodes u and v in G *if and only if* $d(u, v) \leq r$ (i.e. edge$(u, v) \in E_T$) and $k \in \mathcal{A}(u) \bigcap \mathcal{A}(v)$. The latency of the edge e is the latency of the fastest transmission rate supported on e. The set L contains the latency of each edge in E; similarly the set Λ contains the channel used on each edge in E. Note that G may be a *multi-graph*, with multiple edges between the same pair of nodes, when the node pair shares two or more channels. We use the same notation to refer to *vertices and nodes*, to *edges and links*, and to *weight of edges and latency of links* without confusion, the usage being clear from the context.

It is assumed that the *channel assignment* is done independently from our broadcasting framework. This design decision reflects the practical reality that the channel assignment strategy will likely be dictated by other factors, including the presence of *unicast* traffic on the WMN. We have used the following three *static* channel assignment strategies in our current work: CCA [14], VCA [73] and INSTC [74]. All these algorithms have earlier been discussed in Section 2.4. For CCA, dedicated interfaces are allocated for the *same* Q channels at every node, therefore only Q channels are used in the network when using CCA. In VCA, an interface at all nodes is allocated the same channel to ensure a connected network; for the remaining $(Q - 1)$ interfaces, channels are chosen randomly from the remaining $C - 1$ channels. The last channel assignment scheme used is INSTC, which we use to construct at least a 1-connected topology (*i.e.* a connected topology).

Interference Model: A generalized conflict graph *based on transmissions* is used to model the effects of wireless interference between different broadcast transmissions in MR^2-MC meshes. The conflict graph indicates which transmissions mutually interfere and hence cannot be active simultaneously. A transmission b_i interferes with a transmission b_j, if both transmissions b_i and b_j are taking place on the *same* channel, *and* the receivers of the transmission b_i are within the interference range of the transmitting node of b_j or *vice-versa*. The transmissions b_i and b_j do not interfere otherwise.

9.3 Heuristic MLB solution for MR^2-MC WMN

In this section, we present heuristic algorithms to create minimized latency broadcast trees for MR^2-MC WMNs. Since the channel-assignment is performed independently of our framework, the topology defined by the channel-assignment process $\mathcal{A}$ is an input to our framework. Broadly speaking, any heuristic algorithm designed to solve the MLB tree in MR^2-MC meshes must make three important decisions at each node. *Firstly*, it has to decide whether a node should transmit (i.e., be a non-leaf node in the broadcast tree) or not, and if so, whether the transmission should occur over all or some of its radio interfaces. *Secondly*, the number of transmissions the node will actually make *must be determined according to the number of radio interfaces and channels available*, alongside the nodes covered in each of these transmissions. *Lastly*, the transmissions at each node must be scheduled to minimize the broadcast delay after due consideration of radio interference and the number of interfaces available.

The MLB problem is a combination of many closely inter-related hard subproblems e.g. minimum latency tree construction, interference free transmission scheduling and the choice of rate and interface to use for transmissions are all intertwined sub-problems of the overall MLB problem. With the complexity of the problem in mind, we have decomposed our solution into three logically independent steps:

1) *Topology Construction*: The aim of this step is to compute a broadcast tree (or a spanning tree) T of the given topology that exploits the WBA, the multi-rate transmission capability *and the plurality of radio interfaces and channels available*. The transmitting nodes, their interfaces used for transmissions and the children/parent relationships between different nodes are all decided in this stage. We have assumed that each node will transmit only once on a particular channel, i.e., it will not perform multiple distinct-rate transmissions on the same channel. This follows from the finding of Chapter 4 which showed that the performance improvement using multiple distinct-rate transmissions is not significant.

2) *Downstream Multicast Grouping*: The aim of the 'multicast grouping' algorithm is to take the spanning tree constructed during the 'topology construction'

stage and *determine both the rates and number of distinct-rate transmissions that each interface should perform.* Intuitively, the rationale behind multiple transmissions is to allow faster transmission to the more *critical* child nodes (those nodes that have leaf nodes with larger delivery latencies) at the expense of larger transmission latency to the other child nodes.

3) *Transmission Scheduling*: While the number of transmissions at each non-leaf node of the tree is determined after 'topology construction' and 'multicast grouping', the exact timing of the various transmissions especially relative to different branches of the tree still needs to be determined. The final step schedules all transmissions while taking into account that a node can only transmit after it has received the packet and interfering transmissions cannot occur concurrently. We are conceptually assuming a centralized scheduler in our current work. We will, however, also verify our results using the 802.11 MAC in Qualnet simulations [1].

This decomposition of the overall problem is not optimal as was noted earlier for a similar approach in Chapter 4. For example, we obtain the multicast transmission sets and the transmission rate associated with each link layer multicast only after the 'multicast grouping'. We note however that a joint optimization is computationally infeasible except for trivially small topologies.

The outline for the details of our heuristic MLB solution for MR^2-MC WMN is as follows. In the next subsection, we present an example network topology that we shall use throughout this chapter to explain our algorithms. We present four heuristic algorithms for the 'topology construction' stage next; the first (Section 9.4.1) does not exploit the WBA, the second (Section 9.4.2) exploits WBA but not the availability of multiple interfaces on the same node, while the other two (Sections 9.4.3 and 9.4.4) differ in how they exploit both WBA and the radio-and-channel-diversity on individual nodes. We follow the 'topology construction' stage with a broad algorithmic approach for the 'multicast grouping' stage in Section 9.5. We finally present the algorithm for the 'transmission scheduling' stage in Section 9.6. All four of our heuristic MLB algorithm (Sections 9.4.1, 9.4.2, 9.4.3, 9.4.4) share common 'multicast grouping' and 'transmission scheduling' stages, and differ only in the 'topology construction' stage in the manner the broadcast tree is built.

9.3.1 Example topology

To provide an intuitive understanding of our algorithms, we will use, throughout this chapter, a simple example MR^2-MC WMN of 10 nodes in an area of 800^2 m^2. It is assumed that Q (the number of interfaces) is equal to 2 and C (the number of channels) is equal to 4. The positioning of the nodes is as shown in the Figure 9.1(a). As mentioned earlier, the input topology to our algorithms depends on the channel-assignment scheme. The CCA, INSTC and VCA channel-assignment schemes are presented for our example MR^2-MC WMN in Figures 9.1(a), 9.1(b) and 9.1(c). The source-node of the broadcast is represented by a *green square* marker, and the receiver nodes are represented by *blue circular* markers. The node ID (or number) is represented below the node marker. We denote the channels assigned to the interfaces at a node (recall Q=2) in square brackets above the node marker; therefore, as an example [1 2] above the node marker would mean that the radio interfaces of this node are tuned to channels 1 and 2, respectively. As shown in Figures 9.1(a), 9.1(b) and 9.1(c), CCA scheme allocates the same set of channels at each node, whereas VCA scheme allocates *one* common-channel to all nodes, with the remaining channels allocated randomly. The INSTC scheme, in a bid to minimize interference, performs channel-assignment without enforcing a common-channel to be used amongst all nodes in the network while also maintaining a connected network.

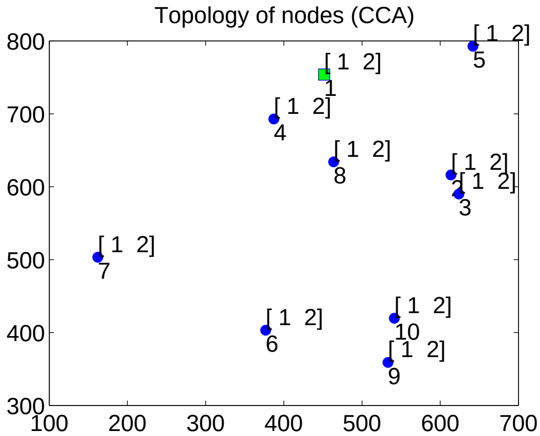
9.4 Topology Construction

The common input to each of our 'topology construction' heuristic algorithm is the channel-assignment defined input topology $G = (V, E, L, \Lambda)$, broadcast source s in V, the set $\mathcal{L} = \{l_1, l_2, ..., l_k\}$ denoting set of latencies of all possible k transmission rates, and the channel-assignments to all interfaces at each node $\mathcal{A}$. We will next discuss our four 'topology construction' algorithms in separate subsections.

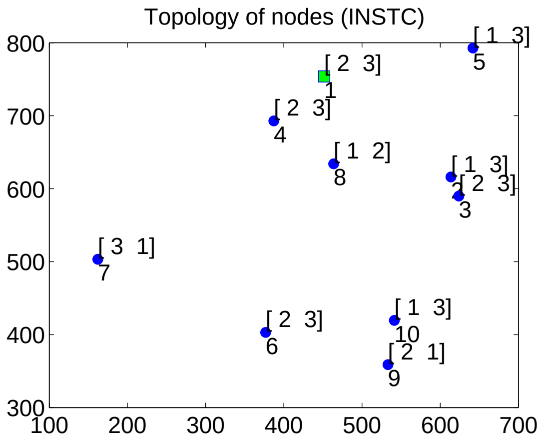
9.4.1 Multi-Radio, Multi-Channel, Shortest-Path Tree (MSPT)

The MSPT algorithm (see Algorithm 7) is used to construct the SPT for MR^2-MC WMNs. The MSPT algorithm, similarly to the greedy Dijkstra algorithm, works on the principle of edge relaxation. The MSPT algorithm differs from the general Dijkstra's algorithm in that it also has to choose appropriate channels for each link it chooses for the MSPT (since a node pair can have multiple links on distinct channels). The broadcast performance results (presented in Section 9.7) greatly depends on the channel selections made during the 'topology construction' stage. We note that channel selection if performed poorly (without due consideration of radio interference) can result in dramatically degraded performance even for the same spanning tree.

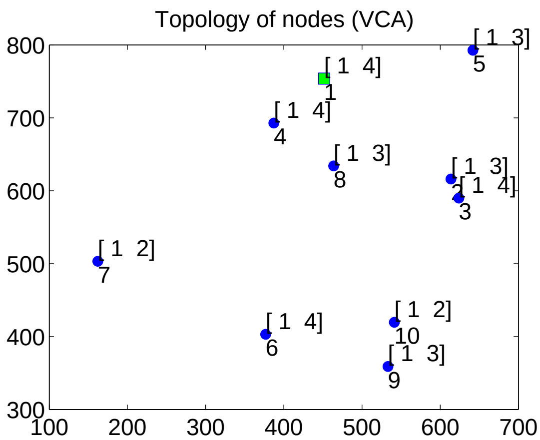
***Algorithm*:** The MSPT algorithm starts by initializing the '*labels*' of all nodes to ∞. The label of any node represents the '*cost*' of its *current* shortest path to the source s; with a label of ∞ indicating the absence of a path. The set $\mathcal{R}$ (representing the nodes, whose shortest paths to s have not been finalized yet) is initialized to contain all nodes in V. The algorithm starts by putting d (the node relaxed at the next iteration) equal to s for the initial round. The basic operation of MSPT algorithm is edge relaxation: if there is an edge from u to v, then the shortest known path from s to u (having cost $label(u)$) can be extended to a path from s to v by



(a) using CCA scheme



(b) using INSTC scheme



(c) using VCA scheme

Figure 9.1: Topology defined by different channel-assignment schemes

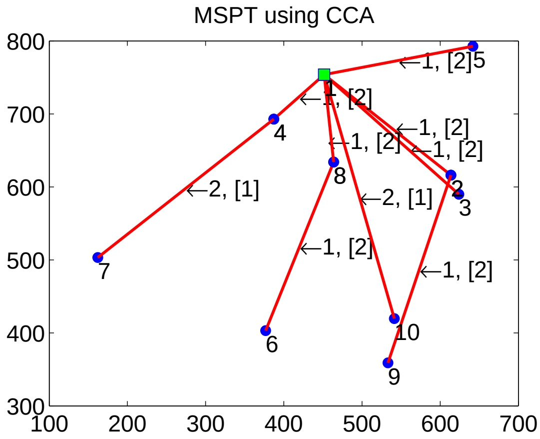
Algorithm 7 The MSPT algorithm

Input: $[s, G = (V, E, L, \lambda), \mathcal{L} = \{l_1 \cdots l_k\}]$
Initialize label$(v_i) = \infty$, $\forall v_i \in V$;
$\mathcal{R} = [1 \cdots |V|]$; $d = s$; $\mathcal{R} = \mathcal{R} \backslash \{s\}$;
while $(V \backslash \mathcal{R} \neq \emptyset)$ **do**
    N = connecting nodes of d;
    $label_{new} = label$;
    $label_{new}(N) =$
    $\min((label(d) + cost(d, N)), (label(N)))$;
    I $\leftarrow$ nodes s.t. $label_{new}(nodes) < label(nodes)$
    $P_{MSPT}(I) = d$;
    for $u = 1$ to |unique-latency-transmissions| at d **do**
        find all nodes I_{lu} **s.t.** $I_{lu} \in I$ and $l(d, I_{lu}) = l_u \in \mathcal{L}$
        $\Lambda_{MSPT}(I_{lu})) =$ least-used channel in the conflict graph of the transmission I_{lu}
    end for
    if d transmitting with latency $\tilde{l}$ on channel chosen **then**
        $L_{MSPT}(edge(d, I)) = max(l(d, I), \tilde{l}))$;
    else
        $L_{MSPT}(edge(d, I)) = l(d, I)$;
    end if
    $label = label_{new}$; $d = \arg\min(label(\mathcal{R}))$;
    $\mathcal{R} = \mathcal{R} \backslash \{d\}$
end while
Output: $[P_{MSPT}, L_{MSPT}, \Lambda_{MSPT}, label]$

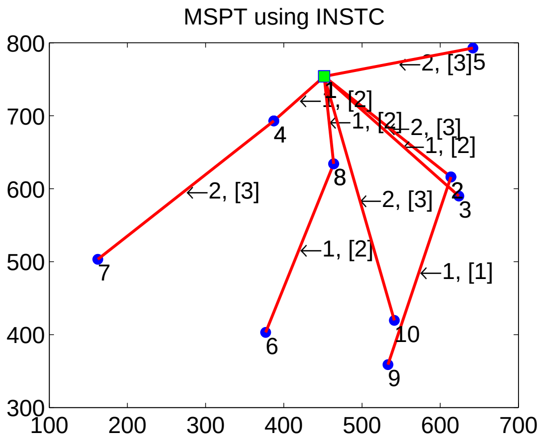
adding edge (u, v) at the end. This path will have length $label(u) + l(u, v)$ where $l(u, v)$ is the latency of link between vertices u and v. If this is less than the current $label(v)$, we can replace the current value of $label(v)$ with the new value.

After edge relaxation in each round, the set of nodes whose labels are reduced from their former values are referred to as I. Amongst the nodes in I, those connecting to d on the same latency transmission $l_u \in \mathcal{L}$ are denoted by I_{lu}, and are assigned a single channel if sharing a common channel. The channel chosen is the *'least-used'* in the conflict graph of this transmission. *Thus, MSPT is based on the Dijsktra algorithm and does not explicitly consider the WBA; it only considers using a less contended channel among available channels between a candidate node pair.* Edge relaxation is applied until all values $label(v)$ represent the cost of the shortest path from s to v. MSPT is mathematically described in Algorithm 7. After $|V| - 1$ rounds, the shortest path from each vertex $v \in V$ to s is determined.

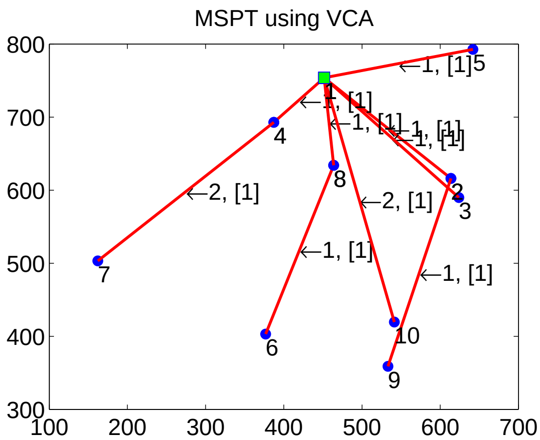
MSPT for the example network shown in Section 9.3.1: We now refer back to our example network, shown in Section 9.3.1, to explain the working of MSPT. The MSPT algorithm constructed trees are depicted in Figures 9.2(a), 9.2(b) and 9.2(c) for CCA, INSTC and VCA channel-assignment schemes, respectively. The broadcast source and receiver nodes are represented by *green square* marker, and *blue circular* markers, respectively, with the node's ID written below its marker.



(a)



(b)



(c)

Figure 9.2: The MSPT tree for different channel-assignment schemes

For every link decided for the tree, the latency and channel it uses is represented in the format: $l, [\lambda]$, where l represents the latency of the link, and λ the channel it uses. Therefore, the values: 1, [2] pointing to a link would indicate that the link's latency is 1 unit, and the link's used channel is 2.

The tree construction is similar to the tree construction of Dijkstra's algorithm. We will focus more on how appropriate channels are chosen for MSPT links. Recall that during the tree-construction, the channel used for a transmission is the least-used channel in the conflict-graph of that particular transmission. Initially, for all our considered channel-assignment schemes, the source node 1 has transmissions at latency 1 and 2. Since both these transmissions interfere with each other, they are assigned different channels. The nodes 2, 4 and 8 also transmit at latencies 1, 2 and 1, respectively. It is preferred that different channels be chosen for these three transmissions, as all of them interfere with each other. Since, CCA only utilizes $Q \leq C$ number of channels, in our example, we can only use channels 1 and 2, as Q=2. Although INSTC and VCA generally use more channels than CCA, their connectivity and WBA exploitation generally reduces due to their greater channel-diversity as the probability of two nodes sharing a common channel is minimized with increasing channel-diversity. The path from the source-node to each node has the lowest possible cost in the MSPT (i.e., without considering interference, MSPT is the best MLB tree). It shall be seen in Table 9.1 (page 125) that MSPT, despite being the shortest-path-tree, is not necessarily the best tree with respect to broadcast latency after accounting for wireless interference. The performance of MSPT with CCA, INSTC and VCA channel-assignment schemes, for our particular example, is 4, 4 and 7, respectively.

Complexity: The outer loop of the relaxation step takes $O(N)$ time to complete, since each node is extracted once. With linear storage, it would take time $O(N)$ to find the cheapest node, which results in a total cost of time $O(N^2)$. Improvements can be made on this time by using a Fibonacci heap to store the nodes, which allows extraction of the cheapest node in time $O(log(N))$. Choosing the latency and channel during each (of the $N-1$) rounds is of $O(|\mathcal{L}| + |C|)$. The complexity of MSPT, therefore, is $O(N(N + |\mathcal{L}| + |C|))$.

9.4.2 Multi-Radio, Multi-Channel, Weighted Connected Dominating Set Tree (MWT)

The MWT algorithm (see Algorithm 8) is an extension to the WCDS algorithm, which is designed for the MLB problem for SR-SC multi-rate networks (Chapter 4). In SR-SC multi-rate WMNs, WCDS performs creditably against other low-latency broadcast heuristics, because WCDS considers both: the multi-rate nature of the network and the WBA of the underlying wireless medium. The MWT, like WCDS, is

Algorithm 8 The MWT algorithm

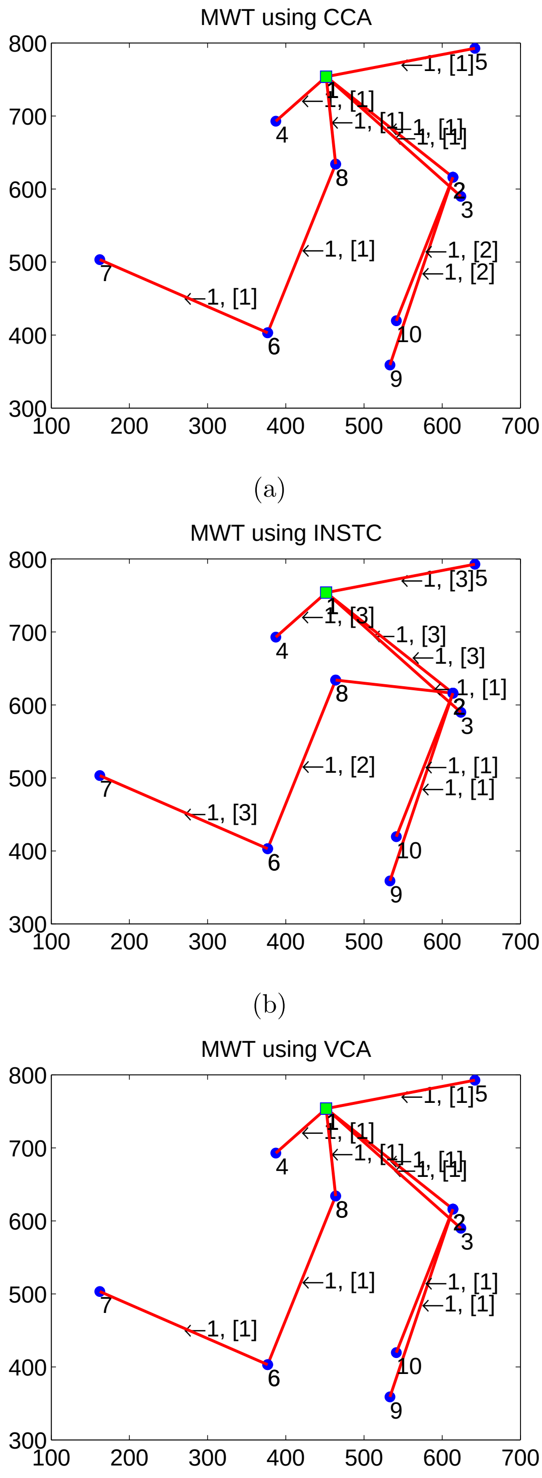
Input: $[s, \mathcal{A}, C, G = (V, E, L, \Lambda), \mathcal{L} = \{l_1 \cdots l_k\}]$
$\mathcal{R} \leftarrow \{s\}$
while $(V \backslash \mathcal{R} \neq \emptyset)$ **do**
    $(\hat{n}, \hat{l}, \hat{c}) = \arg\max_{n \in \mathcal{R}, l \in \mathcal{L}, c \in \mathcal{A}(n)} f(n, l, c)$
    (**where** $f(n, l, c) = (|N(n, l, c) \backslash \mathcal{R}| \div l)$)
    {if multiple $(\hat{n}, \hat{l}, \hat{c})$ with max f, choose whose
    $\hat{c}$ is least used in the conflict graph of $(\hat{n}, \hat{l}, \hat{c})$ }
    $A \leftarrow N(\hat{n}, \hat{l}, \hat{c}) \backslash \mathcal{R}$;
    $P_{MWT}(A) = \hat{n}$;
    $\Lambda_{MWT}(A) = \hat{c}$;
    if $\hat{n}$ already transmitting on $\hat{c}$ (with latency $\tilde{l}$) **then**
        $L_{MWT}(A) = max(\hat{l}, \tilde{l})$;
    else
        $L_{MWT}(A) = \hat{l}$;
    end if
    $\mathcal{R} \leftarrow \mathcal{R} \cup A$
end while
Output: $[P_{MWT}, L_{MWT}, \Lambda_{MWT}]$

a greedy heuristic algorithm that decides the '*best*' transmission in each round, from a set of eligible transmissions. However, as we shall see, *MWT does not consider the availability of multiple interfaces on each node, and thus fails to exploit the potential advantage of parallel transmissions at any intermediate node.*

Algorithm: The algorithm starts by making the source node s eligible to transmit. This is done by moving s to the set $\mathcal{R}$ which keeps track of the eligible-nodes (nodes that have received the transmission already and are eligible to transmit). We say that a node is covered and is eligible for transmission if it is in the set $\mathcal{R}$. We refer to (n, l, c) as a '*combination*' or as a '*transmission combination*', and define it as the transmission by an eligible node $n \in \mathcal{R}$, with latency $l \in \mathcal{L}$, on channel $c \in \mathcal{A}(n)$. We use the term $N(n, l, c)$ to refer to all neighbors of the n which are reachable by the transmission combination (n, l, c). For any transmission combination (n, l, c)—the quantity $|N(n, l, c) \backslash \mathcal{R}|$ (also represented as A in Algorithm 8) is the number of "not-yet-covered nodes" reachable by this transmission combination.

All eligible combinations $(\forall n \in \mathcal{R}, \forall l \in \mathcal{L}, \forall c \in \mathcal{A}(n))$ are given a 'priority' measure defined as the product of "not-yet-covered nodes" and the rate of transmission i.e. $\frac{1}{l}$, or as $|N(n, l, c) \backslash \mathcal{R}| \div l$. The priority is defined such to reflect the desire to both include as many nodes as possible in a single transmission, yet keep the transmission rate high (even though a higher transmission rate implies a smaller range, and thus, a smaller set of covered nodes).

In each round of the algorithm, the node with maximum 'priority' is selected. In case of multiple combinations (n, l, c) having the same priority, the combination transmitting on the channel $\hat{c}$, which is the least-loaded channel within the conflict graph of the transmission as explained in Section 9.4.1, is chosen. The algorithm



(a)

(b)

(c)

Figure 9.3: The MWT tree for different channel-assignment schemes

completes its execution when all the nodes have been covered, i.e. when $V \backslash \mathcal{R} = \emptyset$. The algorithm returns the sets P_{MWT}, L_{MWT} and Λ_{MWT}, where $P_{MWT}(v_i)$ is the parent node of v_i, $L_{MWT}(v_i)$ is the latency of the link connecting v_i and $P_{MWT}(v_i)$, and $\lambda_{MWT}(v_i)$ is the channel used on the link connecting v_i and $P_{MWT}(v_i)$, $\forall v_i \in V$.

The MWT is now readily constructed using these sets.

MWT for the example network shown in Section 9.3.1: We refer to the example network in Section 9.3.1 to illustrate the working of the MWT algorithm. The trees constructed by the MWT algorithm for the channel-assignment schemes of CCA, INSTC and VCA are depicted in Figures 9.3(a), 9.3(b), and 9.3(c), respectively.

Referring to the case of MWT using CCA (Figure 9.3(a)), the choice of the (n, l, c) combination at the end of each successive round is (1,1,1), (2,1,2), (8,1,1), and (6,1,1), respectively. These combinations (n,l,c) are drafted to the tree because their metric $f(n, l, c)$—i.e., 5, 2, 1 and 1 respectively—is the maximum during their respective rounds. The MWT for INSTC and VCA channel-assignment schemes is constructed similarly by adding the highest-priority transmission to the tree at the completion of each round. After the 'transmission scheduling' stage, discussed in Section 9.6, the results obtained for MWT using CCA, INSTC and VCA are 3, 4, and 4, respectively, as shown in Table 9.1 (page 125).

Complexity: Since the MWT algorithm operates in $N - 1$ rounds, in each of which finding $(\hat{n}, \hat{l}, \hat{c})$ requires computations of the order of $|\mathcal{R}| \times |\hat{L}| \times |C|$; $|\mathcal{R}|$ is equal to $N - 1$ in the first round, and in each subsequent round, $|\mathcal{R}|$ is one less than the preceding round. The total computational complexity of MWT is, therefore, calculated to be $O(\frac{N(N-1)}{2} \times |\hat{L}| \times |C|)$.

9.4.3 Locally Parallelized, Multi-Radio, Multi-Channel, WCDS Tree (LMT)

The development of LMT algorithm, which we discuss in this section, is motivated by the observation that MWT, while taking into account the WBA and multi-rate nature of the underlying medium, does not as readily exploit the radio-and-channel-diversity on individual nodes. This observation can be explained more intuitively by noting that MWT is inherently biased, by its priority metric, to include transmissions that cover greater number of uncovered nodes. This metric tends to work well when the number of radio interfaces and channels is small. However, it fails to exploit the increased opportunities for parallel 'faster' transmissions (on different orthogonal channels) when the number of interfaces are higher.

Accordingly, the LMT algorithm is based on the observation that a node m covered by a transmission combination (n, l, c) may also be covered by combination $(n, \hat{l}, \hat{c})$ where $l > \hat{l}$ and $c \neq \hat{c}$. Thus we may be able to cover node m for free on an orthogonal channel $\hat{c}$ without paying penalty on delay. This is done by considering node m as a covered node of $(n, \hat{l}, \hat{c})$ but not (n, l, c).

Algorithm: The LMT algorithm is identical to MWT, except in the calculation of the priorities of eligible transmissions at each round. In MWT, the '*best*'

Algorithm 9 The LMT algorithm

Input: $[s, \mathcal{A}, C, G = (V, E, L, \Lambda), \mathcal{L} = \{l_1 \cdots l_k\}]$
$\mathcal{R} = \{s\}$
while $(V \backslash \mathcal{R} \neq \emptyset)$ **do**
  $(\hat{n}, \hat{l}, \hat{c}) = \arg\max_{n \in \mathcal{R}, l \in \mathcal{L}, c \in C} f(n, l, c)$
  $\{$**where** $f(n, l, c) = (|N(n, l, c) \backslash \{\mathcal{R} \bigcup RN_{(n,l,c)}\}| \div l)$
  and $RN_{(n,l,c)} = \cup_{\forall (l_i \in \mathcal{L}) < l, \forall (c_i \in (\mathcal{A}(n) \backslash \{c\}))} N(n, l_i, c_i)\}$
  {if multiple $(\hat{n}, \hat{l}, \hat{c})$ with max f, choose whose
  $\hat{c}$ is least used in conflict graph of $(\hat{n}, \hat{l}, \hat{c})$}
  $N_{covered} = N(\hat{n}, \hat{l}, \hat{c}) \backslash \{\mathcal{R} \cup RN_{(\hat{n}, \hat{l}, \hat{c})}\}$
  $A \leftarrow N_{covered}$;
  $\mathcal{R} \leftarrow \mathcal{R} \cup A$
  $P_{LMT}(A) = \hat{n}$;
  $\Lambda_{LMT}(A) = \hat{c}$
  if $\hat{n}$ already transmitting on $\hat{c}$ (with latency $\tilde{l}$) **then**
    $L_{LMT}(A) = max(\hat{l}, \tilde{l})$;
  else
    $L_{LMT}(A) = \hat{l}$;
  end if
end while
Output: $[P_{LMT}, L_{LMT}, \Lambda_{LMT}]$

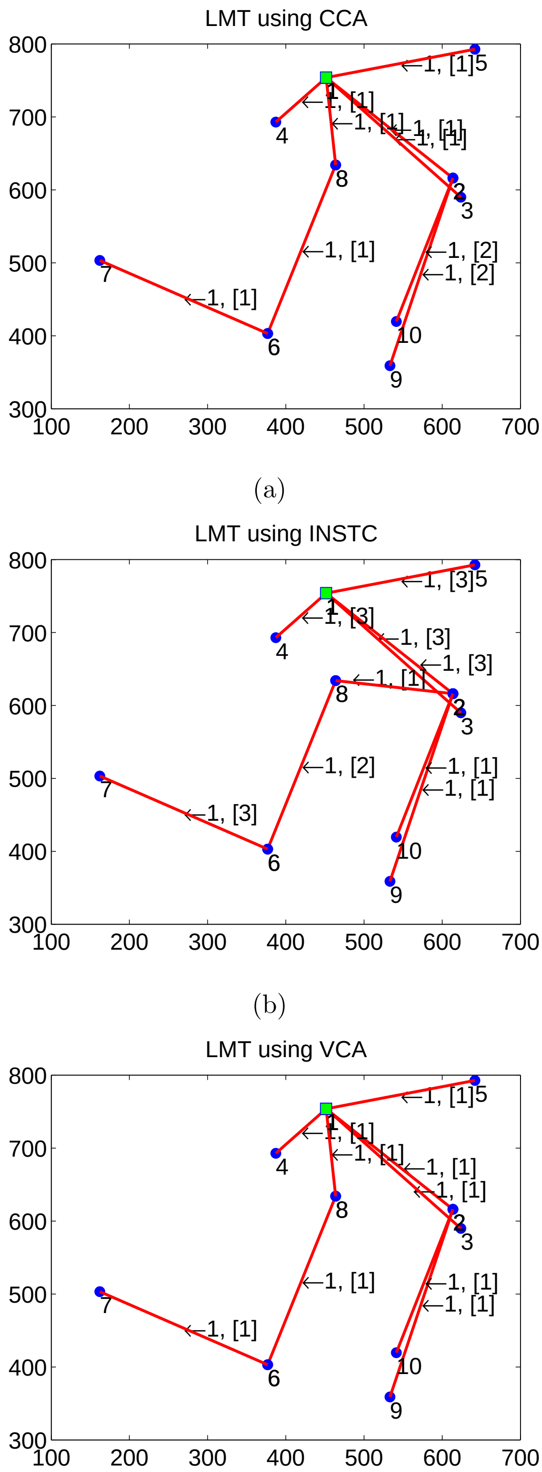
transmission in any particular round is the transmission (n, l, c) with maximum $f(n, l, c) = (|\text{neigh covered}| \div l)$ where 'neigh covered' is $(N(n, l, c) \backslash \mathcal{R})$. In LMT, the term 'neigh covered' is redefined to be $N(n, l, c) \backslash \{\mathcal{R} \cup RN_{(n,l,c)}\}$ where the set $RN_{(n,l,c)}$ contains all nodes that n can cover in *parallel*, at a lower latency than l, on a channel different than c of the (n, l, c) combination.

The nodes covered in each round are added to $\mathcal{R}$, which contains nodes eligible to transmit during the next round. Unlike MWT, where all non-covered neighboring nodes $N(\hat{n}, \hat{l}, \hat{c}) \backslash \mathcal{R}$ of the chosen transmission $(\hat{n}, \hat{l}, \hat{c})$ are added to $\mathcal{R}$; in LMT, only the nodes in $N(\hat{n}, \hat{l}, \hat{c}) \backslash \{\mathcal{R} \cup RN_{(\hat{n}, \hat{l}, \hat{c})}\}$ are added.

The algorithm completes its execution when all the nodes have been covered, i.e. when $V \backslash \mathcal{R} = \emptyset$. The algorithm returns the sets P_{LMT}, L_{LMT} and Λ_{LMT}, where $P_{LMT}(v_i)$ is the parent node of v_i, $L_{LMT}(v_i)$ is the latency of the link connecting v_i and $P_{LMT}(v_i)$, and $\Lambda_{LMT}(v_i)$ is the channel used on the link connecting v_i and $P_{LMT}(v_i)$, $\forall v_i \in V$. LMT can now be readily constructed from these sets.

LMT for the example network shown in Section 9.3.1: We refer to the example network in Section 9.3.1 to illustrate the working of the LMT algorithm. The trees constructed by the LMT algorithm for the channel-assignment schemes of CCA, INSTC and VCA are depicted in Figures 9.4(a), 9.4(b), and 9.4(c), respectively.

For ease of exposition, we have intentionally chosen a very small network i.e., a network of only 10 nodes. In a network of this size, the opportunities to *parallelize* transmissions are limited. The trees constructed using LMT algorithm for CCA, IN-



(a)

(b)

(c)

Figure 9.4: The LMT tree for different channel-assignment schemes

STC and VCA channel-assignment schemes are identical to those constructed using the MWT algorithm for these schemes; as in our example scenario, the transmissions (included in the MWT) are already at the 'quickest' rates, and *parallelizing* to '*quicker*' rates on alternative channel is not possible. After the 'transmission scheduling', discussed in Section 9.6, the results obtained for LMT using CCA, IN-STC and VCA are 3, 4, and 4, respectively, as shown in Table 9.1 (page 125).

Complexity: Since the LMT algorithm operates in $N-1$ rounds, in each of which finding $(\hat{n}, \hat{l}, \hat{c})$ requires computations of the order of $|\mathcal{R}| \times |\hat{L}| \times |C|$; $|\mathcal{R}|$ is equal to $N-1$ in the first round, and in each subsequent round, $|\mathcal{R}|$ is one less than the preceding round. The total computational complexity of LMT is, therefore, calculated to be $O(\frac{N(N-1)}{2} \times |\hat{L}| \times |C|)$.

9.4.4 Parallelized, Approximate-Shortest, Multi-Radio, Multi-Channel, WCDS Tree (PAMT)

The PAMT algorithm, like the LMT algorithm, is adapted from the MWT algorithm, and is designed to be *adaptive* to number of radio interfaces and channels available. The PAMT algorithm is intended as an improvement over the LMT algorithm. The LMT algorithm, during any particular round, might decide to cover some nodes with a transmission that has a longer latency path to s (the source node) compared to other eligible transmissions (*by currently unused interfaces on other intermediate nodes*) that can possibly take place on an alternative, non-interfering channel in parallel. Such a decision is possible despite the fact that in LMT, nodes always attempt to use 'fastest' possible transmitting rates to connect to its neighbors. The following simple example illustrates this idea.

First of all, let us define as the total cost (latency) of the path from a node n to source s as *label* of n. Let us assume that node n can reach a set of nodes Y by transmitting on channel c with latency l_1. The *labels* of all nodes in Y would then be $label(n) + l_1$. Let us assume further that $Y' \subset Y$ can also be covered by a transmission of some other node n' (assume $label(n') < label(n)$) on channel c', with same latency l_1. If covered by transmission of n', nodes in $Y' \subset Y$ have a label of $label(n') + l_1$. Since $Y' \subset Y$, LMT would prefer the transmission of n to that of n' (as it covers more nodes) and therefore would cover all the nodes in Y with n's transmission; this is despite the fact that nodes in $Y' \subset Y$ can be covered with a smaller path cost to s, if n' transmits in parallel on an alternative channel c'.

Algorithm: The PAMT algorithm is also adapted from the MWT algorithm, like the LMT algorithm. PAMT works in a greedy manner, similar to the method of MWT and LMT, to choose the 'best' transmission in each round. The priority metric $f(n, l, c)$ for each transmission (n, l, c), however is calculated differently for PAMT. The PAMT algorithm maintains an extra parameter called *label* for each

Algorithm 10 The PAMT algorithm

Input: $[s, \mathcal{A}, C, G = (V, E, L, \Lambda), \mathcal{L} = \{l_1 \cdots l_k\}]$
$\mathcal{R} = \{s\}$; $label(s) = 0$
while $(V \backslash \mathcal{R} \neq \emptyset)$ **do**
  $(\hat{n}, \hat{l}, \hat{c}) = \arg\max_{n \in \mathcal{R}, l \in \mathcal{L}, c \in \mathcal{A}(n)} f(n, l, c)$
  {if multiple $(\hat{n}, \hat{l}, \hat{c})$ with max f, choose whose
  $\hat{c}$ is least used in conflict graph of $(\hat{n}, \hat{l}, \hat{c})$**}**

  ---

  where $f(n, l, c)$ **is calculated as**:
  $X = Y_{(n,l,c)} = N(n, l, c) \backslash \mathcal{R}$
  $label_{trans} = label(n) + l$;
  if $X \neq \emptyset$ **then**
    $nodes_{tmp} = \cup_{(\forall c_{tmp} \in \mathcal{A}(n) \backslash \{c\}, \forall l \in \mathcal{L})} N(n, l, c_{tmp})$
    $nodes_p = nodes_{tmp} \cap \mathcal{R}$
    for $x = 1$ to $|X|$ **do**
      for $y = 1$ to $|nodes_p|$ **do**
        $latency_{node}(y) = l(nodes_p(y), X(x))$
        $label_{node}(y) = label(nodes_p(y))$
        $label_{round}(y) = latency_{node}(y) + label_{node}(y)$
        if $label_{round}(y) < label_{trans}$ **then**
          $Y_{(n,l,c)} = Y_{(n,l,c)} \backslash \{X(x)\}$; **break**
        end if
      end for
    end for
  end if
  $X = Y_{(n,l,c)}$
  $f(n, l, c) = |X| \div l$

  $A \leftarrow Y_{(\hat{n},\hat{l},\hat{c})}$
  $\mathcal{R} \leftarrow \mathcal{R} \cup A$
  $label(A) = label(\hat{n}) + \hat{l}$
  $P_{PAMT}(A) = \hat{n}$;
  $\Lambda_{PAMT}(A) = \hat{c}$
  if $\hat{n}$ already transmitting on $\hat{c}$ (with latency $\tilde{l}$) **then**
    $L_{PAMT}(A) = max(\hat{l}, \tilde{l})$;
  else
    $L_{PAMT}(A) = \hat{l}$;
  end if
end while
Output: $[P_{PAMT}, L_{PAMT}, \Lambda_{PAMT}]$

node, denoting the cost of its path to s (source node). The algorithm begins by adding node s to $\mathcal{R}$, which is the set of nodes that are eligible to transmit during the next-round. The *label* of s is set to 0, and the *label* for all other nodes is set to ∞. During the execution of each round, PAMT tries to find out which transmission (or edge(s)) should be added to the tree. The set $Y_{(n,l,c)} = N(n, l, c) \backslash \mathcal{R}$ contains all hitherto 'uncovered nodes' that can be covered by this transmission (n, l, c). The label of this transmission denoted by $label_{trans}$ is equal to $label(n) + l$.

During the calculation of priority for each transmission (n, l, c), X contains the neighboring nodes $Y_{(n,l,c)}$ of the transmission (n, l, c). For each node in X, neighboring nodes are searched ($nodes_p$ in Algorithm 10) to find out if they can offer a lower-cost path to s, on an *alternative* channel to c. If such a path is found, then this node *should* not be covered in the transmission (n, l, c). This node, therefore, is not considered a covered-node of (n, l, c) and is deleted from $Y_{(n,l,c)}$. After all nodes in X are checked in a similar manner, $Y_{(n,l,c)}$ contains the actual number of nodes that will be covered by the transmission (n, l, c). The priority of the transmission (n, l, c) is then calculated by dividing $Y_{(n,l,c)}$ by l. In case of multiple transmissions having the same priority, the transmission whose channel $\hat{c}$ is least-used in the conflict graph of that transmission, is chosen. After completion of each round, covered-nodes are added to $\mathcal{R}$. The algorithm completes its execution when all the nodes have been covered, i.e. when $V \backslash \mathcal{R} = \emptyset$. The algorithm returns the sets P_{PAMT}, L_{PAMT} and Λ_{PAMT}, where $P_{PAMT}(v_i)$ is the parent node of v_i, $L_{PAMT}(v_i)$ is the latency of the link connecting v_i and $P_{PAMT}(v_i)$, and $\Lambda_{PAMT}(v_i)$ is the channel used on the link connecting v_i and $P_{PAMT}(v_i)$, $\forall v_i \in V$. The PAMT is constructed from these sets.

It can be shown that the method of LMT of not considering a node as a covered-node of combination (n, l, c), if a higher-rate transmission (n, l', c') of n with $l' < l$ and $c' \in \mathcal{A}(n) \backslash \{c\}$ can cover it, is a special case of PAMT. In PAMT, a node is not considered a covered-node of combination (n, l, c), if there exists an *eligible* transmission (n', l', c') on an *alternative* channel c' with latency l', using which would result in a shorter *label* for the covered node. Due to the fact that higher-rate transmissions of the same node have lower-latency, another transmission on a higher-rate on an alternative channel would always result in a lower-label. Therefore, PAMT is more general than LMT.

PAMT for the example network shown in Section 9.3.1: We refer to the example network in Section 9.3.1 to illustrate the working of the PAMT algorithm. The trees constructed by the PAMT algorithm for the channel-assignment schemes of CCA, INSTC and VCA are depicted in Figures 9.5(a), 9.5(b), and 9.5(c), respectively.

The PAMT for CCA scheme is identical to MWT for CCA scheme, as the chosen transmissions in the tree already are the '*quickest*' and the reached nodes have the shortest paths to the source-node. However, the PAMT for INSTC scheme is different to the MWT and LMT for INSTC scheme; this is because node 8 can be reached by a transmission by the source-node on an alternate-channel (i.e., on channel 2, the channel used earlier by the source-node was 3). The choice of the (n, l, c) combination at the end of each successive round, for PAMT using INSTC, is (1,1,3), (2,1,1), (1,1,2), (8,1,2) and (6,1,3) in the order of their addition. These combinations (n, l, c) are drafted to the tree because the metric $f(n, l, c)$ for these combinations (i.e. 4, 2, 1, 1 and 1 respectively) is the maximum at the end of their respective round. This improves the broadcast latency performance of PAMT for INSTC—from 4 to 3, as shown in Table 9.1 (page 125). The results for PAMT using CCA and VCA (identical to MWT and LMT results) are 3, and 4, respectively, as shown in Table 9.1.

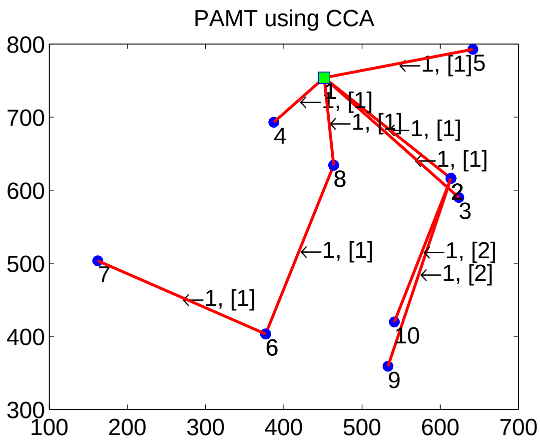
Complexity: Since the PAMT algorithm operates in $N-1$ rounds, in each of which finding $(\hat{n}, \hat{l}, \hat{c})$ requires computations of the order of $|\mathcal{R}| \times d^2 \times |\hat{L}| \times |C|$ where d is the maximum number of neighbors of a node on any latency and channel; $|\mathcal{R}|$ is equal to $N-1$ in the first round, and in each subsequent round, $|\mathcal{R}|$ is one less than the preceding round. The total computational complexity of PAMT is, therefore, calculated to be $O(\frac{N(N-1)}{2} \times d^2 \times |\hat{L}| \times |C|)$.

9.5 Multicast Grouping

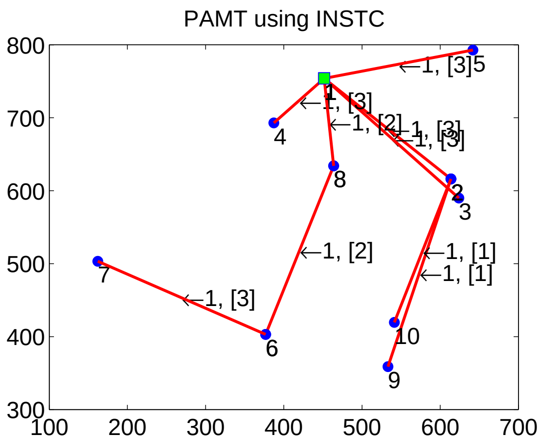
The output of the 'topology construction' stage is a directed broadcast tree with the non-leaf nodes representing the transmitting nodes. A non-leaf node can have possibly multiple outgoing edges with different weights. This translates in a 'physical' sense into multiple link layer multicasts. These link-layer multicasts with *different* transmission rates can take place simultaneously, *if and only if*, these transmissions take place on orthogonal channels (multiple outgoing edges at a node having the same latency weight correspond to a single transmission due to WBA). In the case of different-latency transmissions on the same channel, a decision has to be made to either retain or discard the lower latency transmission(s). The function of the 'multicast grouping' stage is to make this very decision.

The decision is made, while keeping in mind that a 'slower' transmission has a wider 'reach' and vice versa. This implies that the 'slowest' transmission can cover all neighboring nodes, albeit at the cost of increased latency. This trade-off has earlier been studied for the case of SR-SC multi-rate WMNs in Chapter 4. The case of SR-SC multi-rate WMNs, where all transmissions take place on the same channel, requires grouping decisions whenever there are multiple *different-latency* transmissions at a node. For the case of MR^2-MC WMN, a grouping decision needs to be made *only* when the different-latency transmissions are on the *same* channel; with no restriction on simultaneous transmissions on different channels. Our 'multicast grouping' algorithm for MR^2-MC is very similar to the grouping algorithm for the case of SR-SC multi-rate WMN described in Chapter 4, the only difference being that the grouping algorithm in MR^2-MC WMN is invoked only for the case of different-latency transmissions *on the same channel* (and *not on a different orthogonal channel*) at a node.

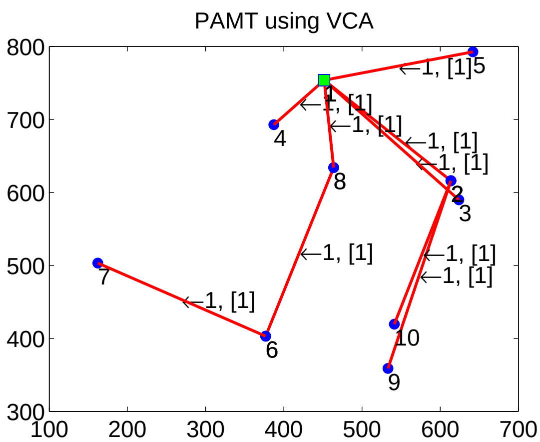
In order to find the topology which minimizes the broadcast latency, we must make a number of decisions, including which node is to multicast, and if so, how many times it is to multicast, whom the recipients are and its timing. As stated



(a)



(b)



(c)

Figure 9.5: The PAMT tree for different channel-assignment schemes

earlier, the result of the 'topology construction' stage is a broadcast tree which specifies that the non-leaf nodes of the broadcast tree will multicast to its child nodes, in possibly multiple transmissions. However, the number of times a transmitting node (i.e. a non-leaf node of the broadcast tree) will multicast and the recipients of each multicast still have not yet been decided. In case where a node multicasts only once, then the recipients will be all its child nodes. For the case where a node is to multicast more than once, a different subset of child nodes will be reached in each multicast such that these subsets all together form the set of child nodes. The aim of the 'multicast grouping' stage is to determine precisely the number of multicasts that must be made and the recipients of these multicasts.

The "multicast grouping" algorithm for our framework for MR^2-MC WMNs is similar to the grouping algorithm of our framework for SR-SC multi-rate WMNs that was described in Section 4.3.2. However, the grouping and the choice of transmission rates for the grouping algorithm of MR^2-MC WMNs, is performed at each node for each interface independently, and the downstream latency (called Cardinal Value in Chapter 4) of the node is subsequently determined by the maximum of the downstream latency across all interfaces.

9.6 Transmission Scheduling

The '*transmission scheduling*' algorithm tries to schedule the transmissions to minimize the broadcast delay whilst ensuring that interfering transmissions are not scheduled simultaneously. Our 'transmission scheduling' algorithm is very similar to the scheduling algorithm for the case of SR-SC multi-rate WMNs presented in Section 4.3.3. We modify the algorithm presented in Section 4.3.3 according to the interference model described in Section 9.2. These modifications are required to ensure that transmissions on orthogonal channels can be scheduled together.

The broadcast tree generated after the 'multicast grouping' stage can be modeled by a directed tree $T = (V, E, L, \Lambda)$. The transmitting nodes are represented by branching vertices (i.e. non-leaf nodes) in the tree T. Let us denote the number of transmitting nodes in the network by k, the set of transmitting nodes denoted by $V_b = \{b_1, b_2 \cdots b_k\}$. Let us denote the set of w different-latency transmissions at any arbitrary node b_i by $\hat{\mathcal{L}}_{bi} = \{b_{i1} \cdots b_{iw}\}$. The set B contains all the transmissions in the network, $B = \{b_{ij}\}, \forall b_i \in V_b, \forall j \in \hat{\mathcal{L}}_{b_i}$. We model the interference between transmissions in an MR^2-MC WMN by using a conflict graph for each channel. The conflict graph $G_{ci} = (B, E_{ci})$ models the interference, *on channel* $i \in C$, between the set of transmissions B. The set of conflicting edges E_{ci} contains an edge (b_{ij}, b_{kl}) *only and only if* both transmissions are on the same channel i, *and* the transmitter of b_{ij} interferes with the receivers of the transmission b_{kl} or vice versa.

Formally, the transmission schedule is the mapping $\tau : b_{ij} \rightarrow \mathbb{R}, \forall b_{ij} \in B$ which gives the starting time of b_{ij}. The transmission schedule must obey the following constraints:

1. The source node s must transmit at least once at time zero.
2. All nodes must follow precedence constraint, i.e a node can only transmit after receiving the packet from its parent.
3. Two arbitrary transmissions b_{ij} and b_{kl} can be scheduled together on an arbitrary channel i, *only and only if* the edge $(b_{ij}, b_{kl}) \notin G_{ci}$.
4. At most one transmission can take place at a time on one interface of any node.

Algorithm: The algorithm for 'transmission scheduling' stage is mathematically described in Algorithm 11. The set of transmissions $B = \{b_{ij}\}, \forall b_i \in V_b, \forall j \in \hat{\mathcal{L}}_{b_i}$, the channel used $\lambda(b_{ij})$ by $\forall b_{ij} \in B$, and the latency $l(b_{ij})$ of $\forall b_{ij} \in B$, is given as the input to our scheduling algorithm. The current time *time* is initialized to zero, and the set E, containing eligible-transmissions, is initialized with different-latency transmissions of the source-node s. The set T containing ongoing transmissions is initialized as an empty set.

The algorithm then finds the transmission with the maximum '*Cardinal Value*' (CV) amongst all eligible transmissions (depicted as $\forall b_E \in E$ in Algorithm 11). The CV of a node is defined as the worst-case '*latency distance*' to any of its downstream nodes. Our scheduling algorithm gives priority to transmissions which are more *'critical'* or have higher CV values. The transmission with the maximum CV (let us denote this transmission by b_{ij}) is then deleted from the set of eligible transmissions E. It is then confirmed that the selected transmission b_{ij} does not interfere with any ongoing transmission, represented as b_T, on the channel used by b_{ij} (represented by $\lambda(b_{ij})$). The number of ongoing transmissions p of the node transmitting b_{ij} (i.e. b_i) is then determined. If p is less than the number of radio interfaces Q, then b_{ij} is added to the set of transmissions taking place and its starting time $\tau(b_{ij})$ is decided as the current time *time*. The ending time of transmission b_{ij} is also decided as $time + l(b_{ij})$. However, if p is more than Q, it implies that node b_i has no free interface and all its interfaces are busy in transmitting. The transmission b_{ij}, therefore, has to be held-back until the next-round; the transmission b_{ij} is added to $E_N ext$ which is the set of eligible-transmissions for next-round.

Thereafter, $NextStop$ is calculated as the earliest finishing time of any transmission in T. The transmission(s) $NextTrans$ have the earliest finishing time of all transmissions in T. The transmissions enabled by the transmissions $NextTrans$ and the transmissions held-back during the current-round E_{Next}, are now added to E, as these transmissions are eligible for next-round. The transmission(s) $NextTrans$ are

Algorithm 11 Transmission scheduling with multiple radios and channels

Input: $\forall b_{ij} \in B$ **(all** j **trans. at transmitting node** $b_i (1 \leq i \leq k)$
Input: $\lambda(b_{ij})$ **(channel** b_{ij} **transmits at** ch_i**)**
Input: $l(b_{ij})$
Set $time = 0$
Initialize $E \leftarrow \cup_{\forall j} \{b_{sj}\}$
Initialize $T = \emptyset$
while $(E \neq \emptyset$ or $T \neq \emptyset)$ **do**
  while $E \neq \emptyset$ **do**
    $b_{ij} = \arg\max b_E.CV \ \ (\forall b_E \in E)$
    $E = \{E \backslash b_{ij}\}$
    $p = |\text{transmissions of } b_i \in T|$
    if $(b_{ij}, b_T) \notin E_c(\lambda(b_{ij}))$ in $G_c(\lambda(b_{ij}) \ \forall b_T \in T$ **then**
      if $p <$ Radios **then**
        $T \leftarrow \{T \cup b_{ij}\}$;
        Set $\tau(b_{ij}) = time$
        Set $\delta((b_{ij}) = time + l(b_{ij})$
      else
        $E_{Next} \leftarrow b_{ij}$
      end if
    end if
  end while
  $NextStop$= min $\delta(b_T) \ \forall b_T \in T$
  $NextTrans$= $\{b_N\} : (\forall b_N \ \delta(b_N) = NextStop)$
  $E \leftarrow E \cup b_{children}$ of $NextTrans$
  $T = T - NextTrans$
  $E = E - NextTrans$
  $E = E \cup E_{Next}$
  $time \leftarrow NextStop$
end while
Output: $\tau(b), \delta(b) \ \forall_{(1 \leq b \leq |b|)}$

then deleted from T and E. The round finishes by adjusting the $time$ to $NextStop$. The rounds continue until all transmissions have been scheduled and the start-time of each transmission has been calculated.

Transmission Scheduling for the example network shown in Section 9.3.1: We refer to the example network in Section 9.3.1 to illustrate the working of our 'transmission scheduling' algorithm. The output of 'transmission scheduling' for the CCA channel-assignment scheme is depicted in Figures 9.6(a), 9.6(b), 9.6(c) and 9.6(d) for the MSPT, MWT, LMT and PAMT algorithms, respectively. The node ID of the transmitting nodes is depicted on the vertical axis, while time is shown on the vertical axis. The red horizontal lines depict the time spent by a node transmitting, while the channel of transmission is depicted in blue *on* this horizontal line. The '*children nodes*' reached are shown *below* the line in black (or above the line in certain cases for readability).

Referring to the Figure 9.6(a), which contains the MSPT for CCA (Figure

Heuristic	CCA	INSTC	VCA
MSPT	4	4	7
MWT	3	4	4
LMT	3	4	4
PAMT	3	3	4

Table 9.1: Performance of the heuristics for the example topology in Sec. 9.3.1

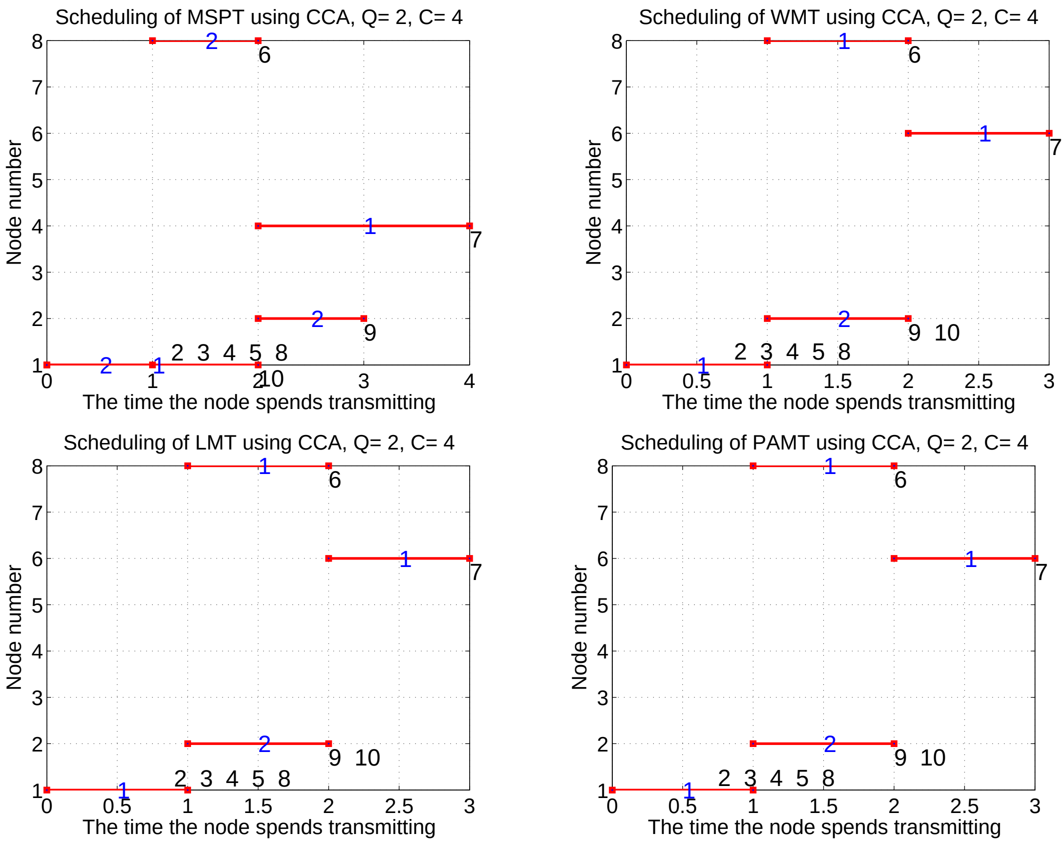


Figure 9.6: Transmission scheduling for CCA channel-assignment scheme

9.2(a)), we examine how the transmissions are scheduled. The source-node 1 starts with two transmissions, with latency 1 and 2, on channel 2 and 1, respectively. The nodes reached by the transmission (or the children nodes) with latency 1, are 2, 3, 4, 5 and 8. Node 8 then starts transmitting in parallel with the ongoing transmission (with latency 2) of node 1. It should be noted that, at any given time, interfering transmissions cannot coexist on the *same* channel. All transmissions interfere with each other for our example network due to its small size. Therefore, for any given channel, only a single transmission can take place at one time. The MSPT for CCA finishes transmitting to all nodes with a broadcast latency of 4.

We will now discuss the 'transmission scheduling' stage for the MWT, LMT and PAMT algorithms, with INSTC as the channel-assignment. These trees are shown in Figures 9.3(b), 9.4(b), and 9.5(b), and their transmission schedules are shown in Figures 9.7(b), 9.7(c), and 9.7(d), respectively. For the MWT algorithm, the source-

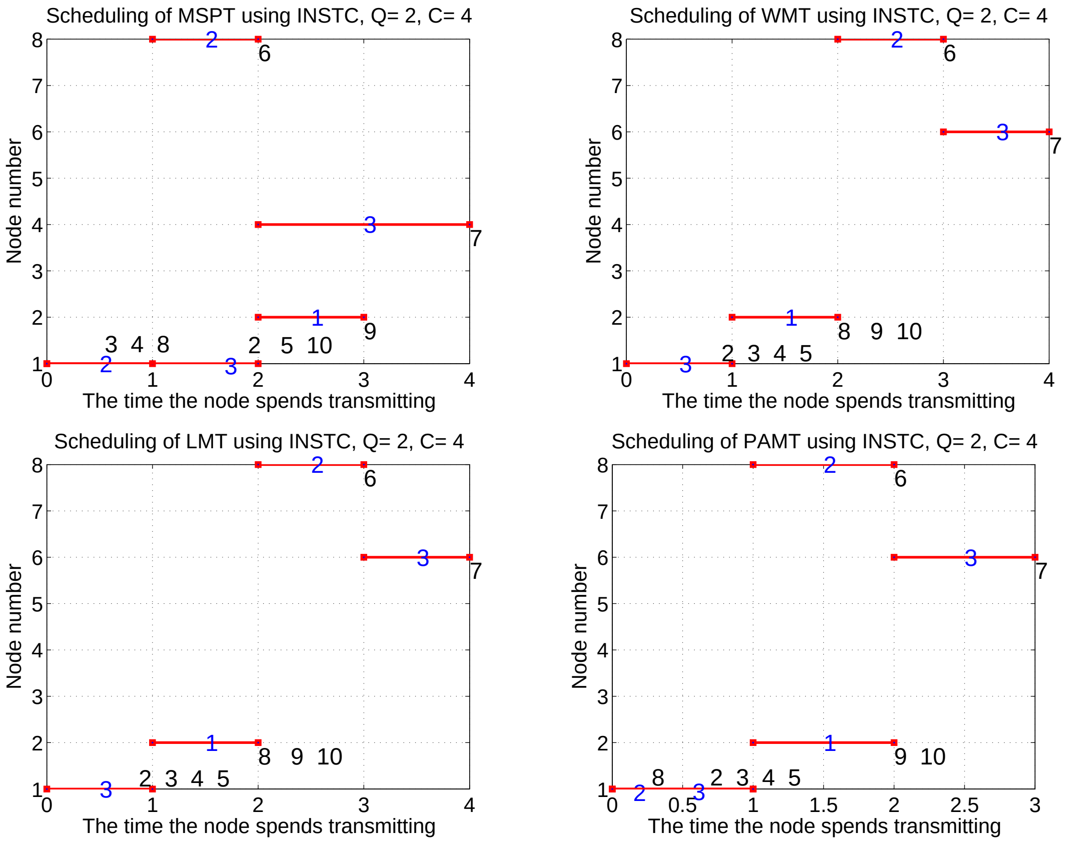


Figure 9.7: Transmission scheduling for INSTC channel-assignment scheme

node starts with a transmission with latency 1, on channel 3, and reaches nodes 2, 3, 4 and 5. The node 2 then transmits at time 1, on channel 1 with latency 1, to reach nodes 8, 9 and 10. The nodes 8 and 6 follow with transmissions of latency 1 on channels 2 and 3, at time 2 and 3, respectively. The broadcast latency of the MWT algorithm using INSTC is 4 units. The scheduling of the LMT algorithm is identical to MWT's scheduling, since both trees are identical. The PAMT for INSTC, however, improves performance by *parallelizing* the transmissions of node 1 with latency 1, on channels 2 and 3. The node 8, rather than being covered by the transmission of node 2 as was the case in MWT and LMT, is now covered by a transmission by the source-node. The node 8 can now start transmitting at time 1, and enable the transmission at node 6 to start and complete its transmission at time 2 and 3, respectively. This improves the performance of both MWT and LMT.

The scheduling for other trees and channel-assignments schemes is done similarly, and is shown in Figures 9.6(a) to 9.8(d). The broadcast latency of the trees and channel-assignment schemes are shown in Table 9.1.

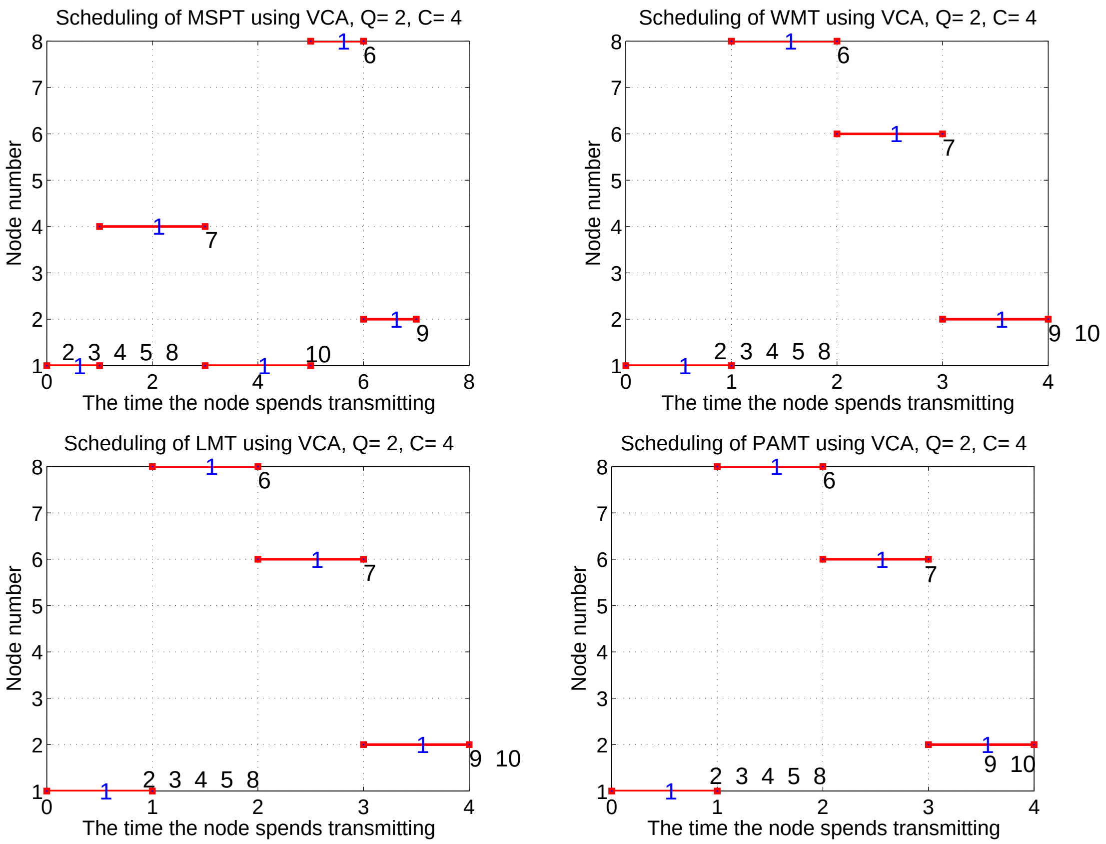


Figure 9.8: Transmission scheduling for VCA channel-assignment scheme

9.7 Simulation results using idealized scheduler

In this section, we present results that have assumed an idealized scheduler (Section 9.6), along with ideal MAC assumptions (Section 4.2.2). We have assumed static WMNs composed of N nodes randomly located in an area of 1200^2 m^2. The transmission rate-range relationship depicted in Table 1.1 is assumed. The interference range is assumed to be 821 m. We have considered three channel-assignment schemes in our current work: CCA, VCA and INSTC (discussed earlier in Section 9.2). We will observe the effect of the number of network nodes, the number of radio interfaces at each node, and the channel-assignment strategy on the broadcast latency performance of our algorithms. For the results presented, CCA channel-assignment scheme must be assumed unless stated otherwise.

The outline of the remainder of this section is: We study the effect of varying node density, and varying radio interfaces on the broadcast performance of our heuristics in Sections 9.7.1 and 9.7.2, respectively. The performance of our four proposed heuristics relative to each other is shown in Section 9.7.3. The effect of the choice of channel-assignment scheme on broadcast latency is then explored in Section 9.7.4.

9.7.1 Effect of node density

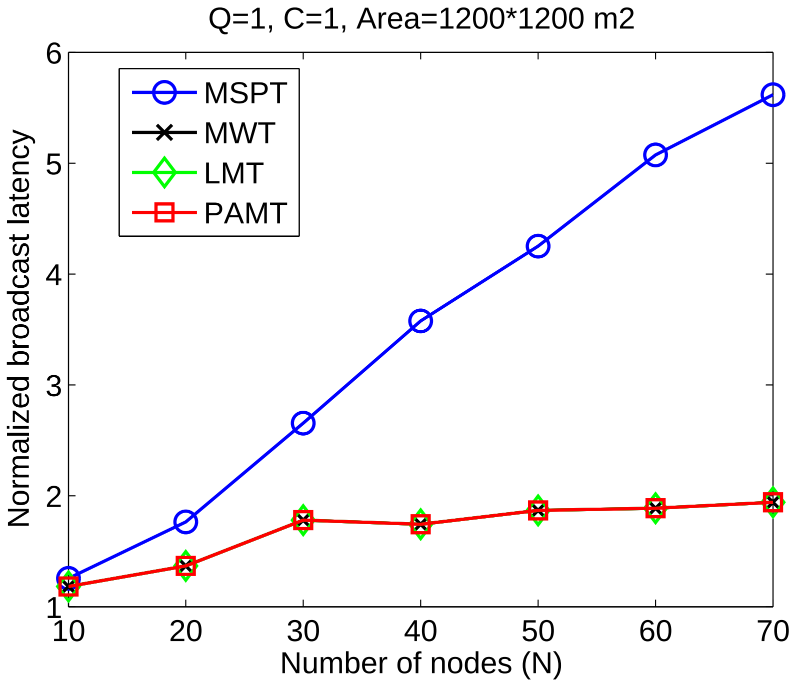
The effect of network's node density on the performance of our heuristics can be seen in Figures 9.9(a) and 9.9(b) for the case of Q and C being 1 and 8, respectively. The vertical axis shows the broadcast latency of our heuristics normalized against the broadcast latency of the Dijkstra's tree with infinite number of Q and C. Since determining the actual optimal is NP-hard, we will approximate the optimal performance by the broadcast latency of Dijkstra's tree, while assuming unlimited number of radios and channels, and that there is no limit to the number of distinct-rate transmissions a node can make. It is observed that the PAMT algorithm performs the best of all algorithms for the range of network node density (10 to 70 nodes in an area of 1200^2 m^2). The performance of LMT, although it uses parallelization like PAMT, is not as good as PAMT's (Figures 9.9(a) and 9.9(b)) but nonetheless is better than MWT and MSPT. The performance of MSPT, expectedly, is poor and worsens as the network node density is increased.

9.7.2 Effect of number of radio interfaces

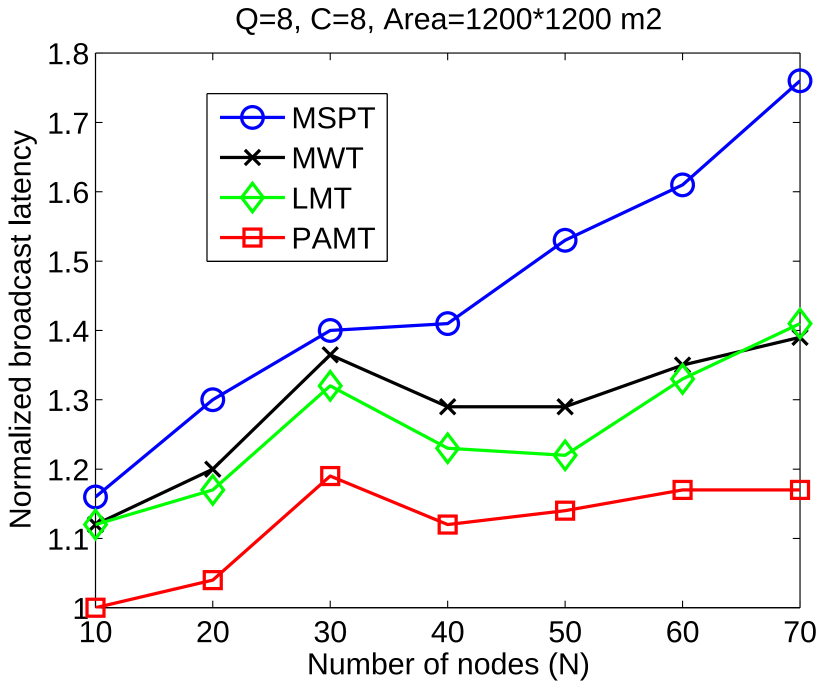
The performance of our heuristic algorithms for the case of a SR-SC multi-rate WMN is presented in Figure 9.9(a). MWT, LMT and PAMT perform identically for the specific case of a *SR-SC multi-rate WMN* (i.e., when $Q = C = 1$), with MSPT performing considerably worse (Figure 9.9(a)). These results are similar to those shown in Chapter 4. MWT performs better than MSPT since it considers both the WBA and the multi-rate nature of the mesh (Figure 9.9(a)). The LMT and PAMT algorithms, both adapted from MWT, can only *match* and *not improve* the performance of MWT (Figure 9.9(a)) in SR-SC multi-rate scenarios, since both cannot find *alternative* channel paths to '*parallelize*' transmissions on. *Thus for a SR-SC multi-rate WMN, the performance of LMT and PAMT is exactly the same as MWT.*

For the cases of MR^2-MC multi-rate meshes where $Q > 1$, all of our proposed heuristics improve their performance. This is true both for small networks (N=10, Figure 9.10(a)) and for large networks (N=70, Figure 9.10(b)). The Figures 9.10(a) and 9.10(b) display representative performance of different heuristics for MR^2-MC WMNs across the range of radio interfaces from Q= 2 to Q=8.

The improvement seen in MR^2-MC performance can be attributed to two main reasons: *Firstly*, the usage of MR^2-MC minimizes the interference in the network and allows interfering transmissions to be transmitted simultaneously using orthogonal channels. This improvement factor called '*interference reduction factor*' is general and applies to all our proposed heuristics. The '*interference reduction factor*' substantially improves performance when the heuristic constructed tree involves many transmissions (e.g. as in MSPT). *Secondly*, a heuristic broadcasting algorithm that

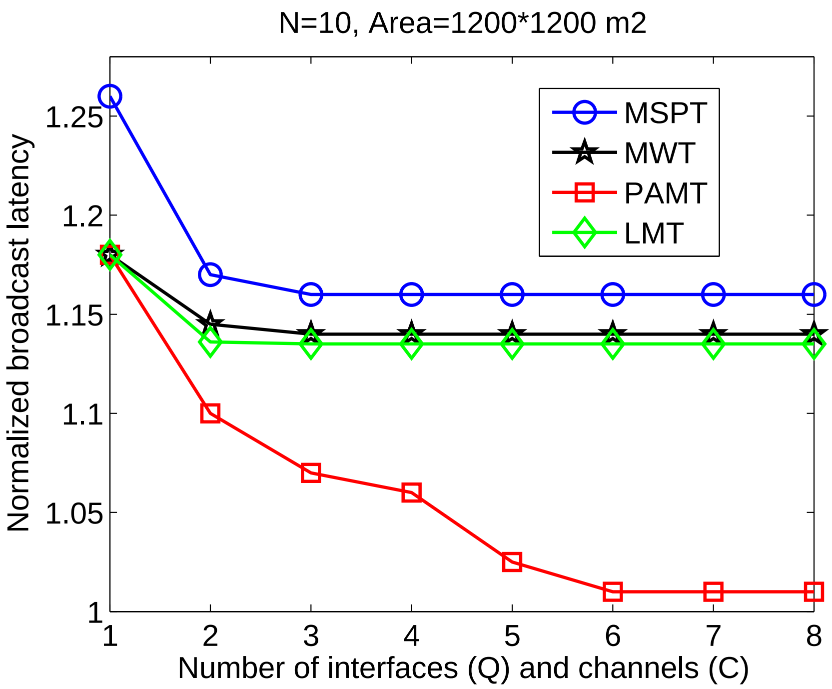


(a) $Q = C = 1$

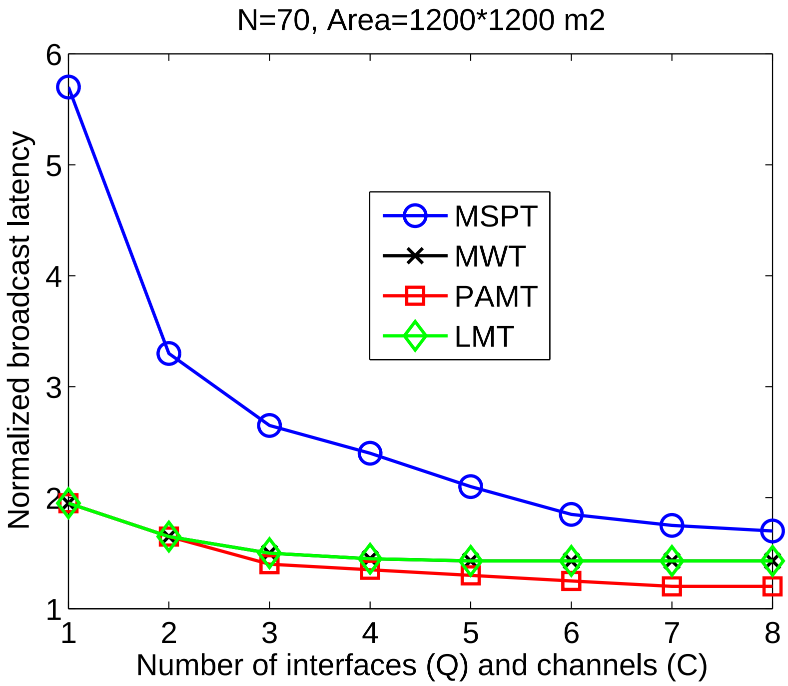


(b) $Q = C = 8$

Figure 9.9: Normalized broadcast latency against varying number of nodes

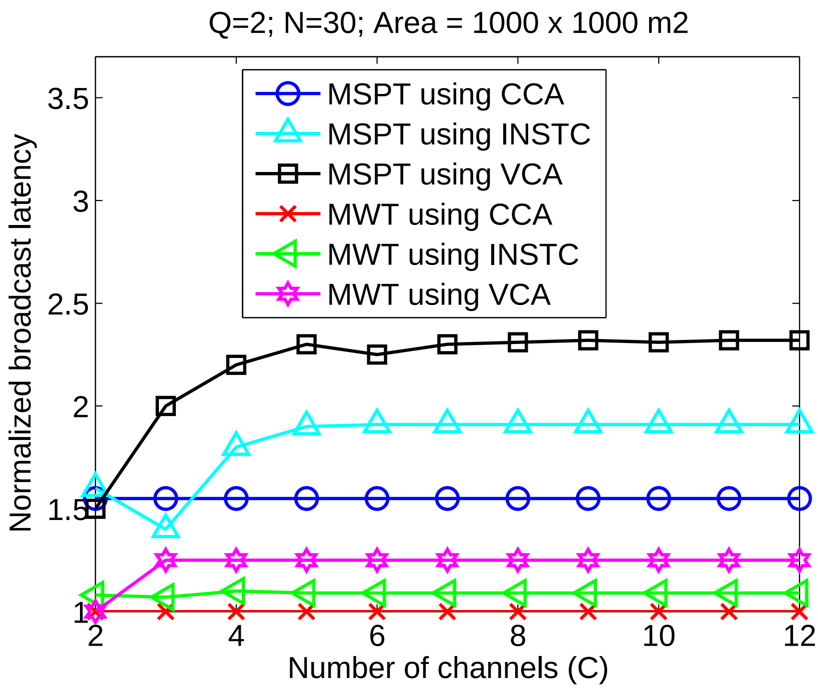


(a) $N = 10$

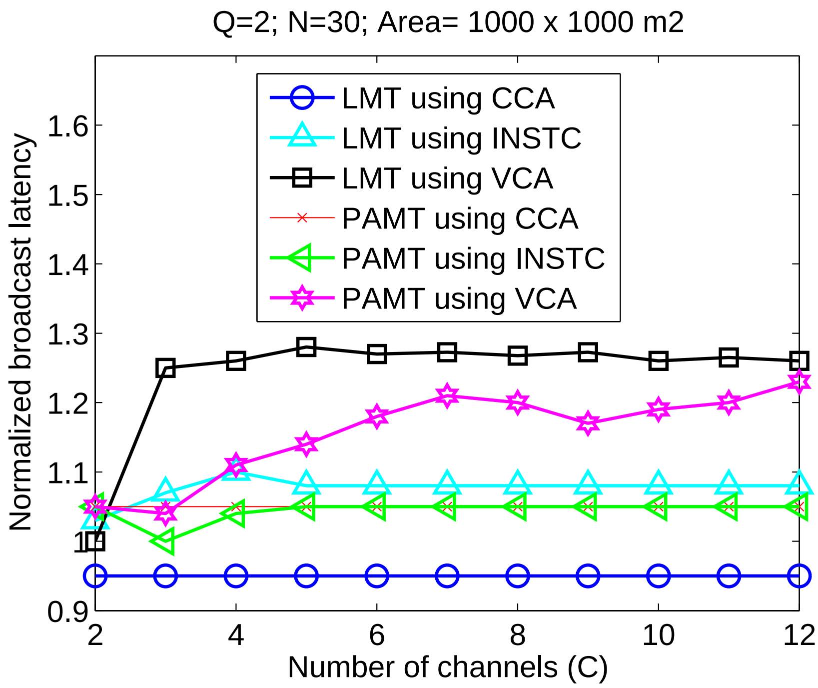


(b) $N = 70$

Figure 9.10: Normalized broadcast latency against varying number of radio interfaces ($Q = C$)



(a) MSPT and MWT



(b) LMT and PAMT

Figure 9.11: The impact of channel-assignment for Q= 2, N= 30 and Area=1000 $\times$ 1000 m^2)

parallelizes its transmission, according to the number of available interfaces and channels, reaps extra benefits by efficient usage of the resources available. This improvement factor called the *'radio adaptation factor'* is specific to broadcasting algorithms such as LMT and PAMT.

Finally, we point out the performance gain due to multiple radio interfaces in MR^2-MC meshes over SR-SC multi-rate meshes. Referring to Figures 9.10(a) and 9.10(b), we see that for Q as less as 3 or 4, the broadcast latency decreases by about 30 - 40% compared to the scenario where well-designed heuristics are used and by as much as 80% when poorly designed heuristics (e.g. MSPT) are used for Q=1.

9.7.3 Comparison of MSPT, MWT, LMT and PAMT

We will now discuss the performance of each of our heuristic in MR^{2}MC wireless meshes with increasing Q and C. The performance of MSPT improves with increasing radio resources due to '*interference reduction factor*'—however, in our considered range of nodes (10 to 70) and interfaces (1 to 8), its performance compared to other proposed heuristics is modest (Figures 9.10(a) and 9.10(b)). MSPT's poor performance is explained by its *lack of accounting for WBA* during its construction, which in turn implies that too many transmissions are involved in a MSPT. Another reason is its *lack of adaptation to the available radio resources.* It must be pointed out that although MSPT's performance is poor in the practical range of values of Q, its performance with the non-practical value of $Q = \infty$ corresponds to optimal achievable performance.

The performance of MWT can be seen in the Figures 9.10(a), 9.10(b), and 9.9(b). It is worth noting that the performance of MWT improves with increasing Q, till a point, beyond which increasing Q does not produce any noticeable gain. Note in the Figures 9.10(a) and 9.10(b), that although good gains are achieved when increasing the Q from 1 to 3, increasing Q further does not produce any gain. This is because MWT does not *parallelize* its transmission by *adapting* to increasing number of interfaces unlike LMT and PAMT. Thus for MWT, like MSPT, only '*interference reduction factor*' is relevant and the '*radio adaptation factor*' does not apply.

It is interesting that both LMT and PAMT improve upon MWT's performance when Q and C are increased, as depicted in the Figures 9.10(a), 9.10(b), and 9.9(b). This implies that both these algorithms are *adaptive* to the available radio resources, and can therefore benefit from both the '*interference reduction factor*' and the '*radio adaptation factor*'. The LMT algorithm is the best performing heuristic for $Q = 2$ and $N = 70$ (Figure 9.10(b)). In such large networks with limited resources (in the considered case, $Q = 2$), the effect of interference is dominant and the trees that transmit less generally perform better. Since LMT is more conservative than PAMT in adding *parallel* links, it performs slightly better than PAMT in this case.

PAMT is generally the most *adaptive* of our algorithms to the available Q and

C. The broadcast performance of PAMT is very close to optimal for small networks *and/or* large Q (Figure 9.10(a)). PAMT also performs consistently well across all ranges of Q and N. Interestingly, PAMT can approach the performance of MSPT with $Q = \infty$ with relatively few radio interfaces in most instances.

9.7.4 Effect of channel-assignment scheme

The graphs of the performance of different channel-assignment schemes (CCA, VCA and INSTC) are shown in Figures 9.11(a) and 9.11(b) for the cases of Q= 2. The results shown are representative of similar results seen across different values of Q. The vertical axis in the graphs show broadcast latency of the algorithm normalized against the MWT algorithm with channels assigned through CCA. All the channel-assignment schemes considered have different *connectivity* and *interference* characteristics. As noted earlier, the topology given as input to our heuristics greatly affects the broadcast performance; with the input topology being defined by the channel-assignment scheme, broadcast performance is closely affected by the channel-assignment scheme chosen.

In CCA, a set of common channels are shared amongst all nodes; hence both the connectivity and interference are maximum. In VCA, although connectivity is ensured by tuning one interface at all nodes to a common channel, the remaining interfaces are assigned channels randomly from the remaining channels in C, the connectivity, therefore, can suffer at the cost of reduced interference. In INSTC, much like VCA, network interference is reduced by increasing channel diversity; however, this is at the cost of reduced connectivity which can possibly mitigate the WBA. An ideal channel-assignment algorithm has to balance the two conflicting requirements of low interference and high connectivity. In the presence of low interference, more transmissions can be scheduled simultaneously resulting in reduced broadcast latency. Similarly, with large connectivity there are increased opportunities of availing the WBA.

From Figures 9.11(a) and 9.11(b), it can be seen that for values of C only slightly larger than Q, VCA and INSTC can sometimes outperform CCA. This is because in such a scenario, the effect of reduced interference outweighs any reduction in connectivity. However, with further increase in C, the reduced connectivity can adversely affect the broadcast latency of the heuristics by neutralizing the WBA. This leads to generally more transmissions (not availing the WBA), and higher broadcast latencies. The characteristic of reduced interference in VCA and INSTC schemes have a more pronounced effect on the performance of MSPT, LMT and PAMT than on MWT, since these algorithms generally involve more transmissions (on possibly interfering channels). As we can see from Figures 9.11(a) and 9.11(b), the best performing channel-assignment scheme for broadcast generally is CCA (which performs poorly for unicast flows [28]). Although the channel-assignment scheme INSTC gives improved performance for unicast traffic, it is not necessarily the best performing channel-assignment scheme for broadcast. Thus, we make an important observation that *a channel-assignment scheme designed for unicast flows may sometimes perform poorly for broadcast/multicast flows.*

9.8 Simulation results using Qualnet's 802.11 MAC

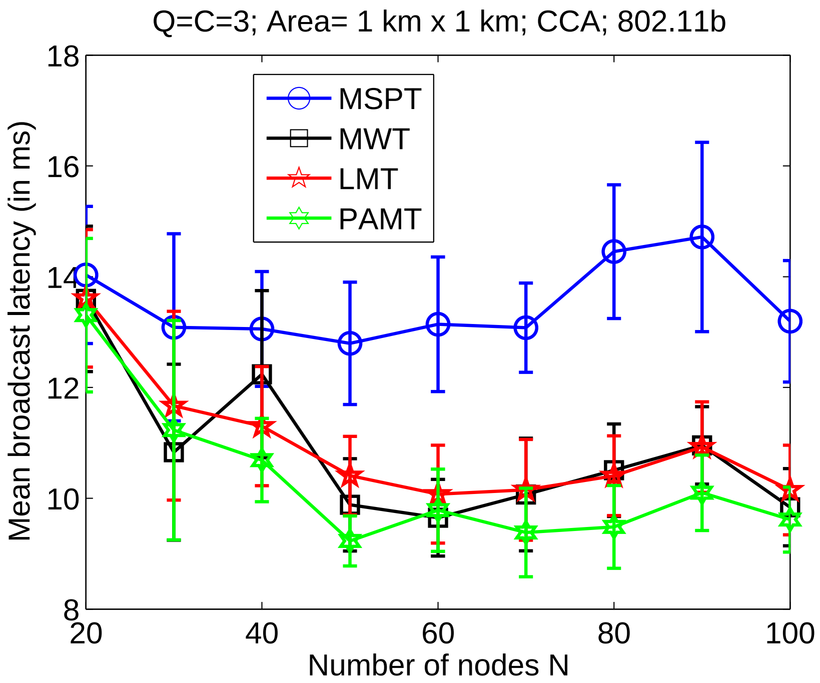
In this section, we present the results of our simulations using the Qualnet [1] simulator to evaluate our algorithms' performance with a decentralized MAC scheduler. We have used both 802.11a and 802.11b as our MAC scheduler with PHY 802.11a and PHY 802.11b, respectively, at the physical layer which use a pre-configured BER-based packet reception model. The IEEE 802.11 MAC with Distributed Coordination Function (DCF) was chosen as the medium access control protocol. All default parameters are assumed unless stated otherwise. For the presented results, an interval estimate with the confidence interval of 90 is used. The ticks in the graphs represent 5th and 95th percentiles over 100 uniformly distributed random topologies. We will now proceed to discuss the results in the next few subsections.

9.8.1 Effect of node density

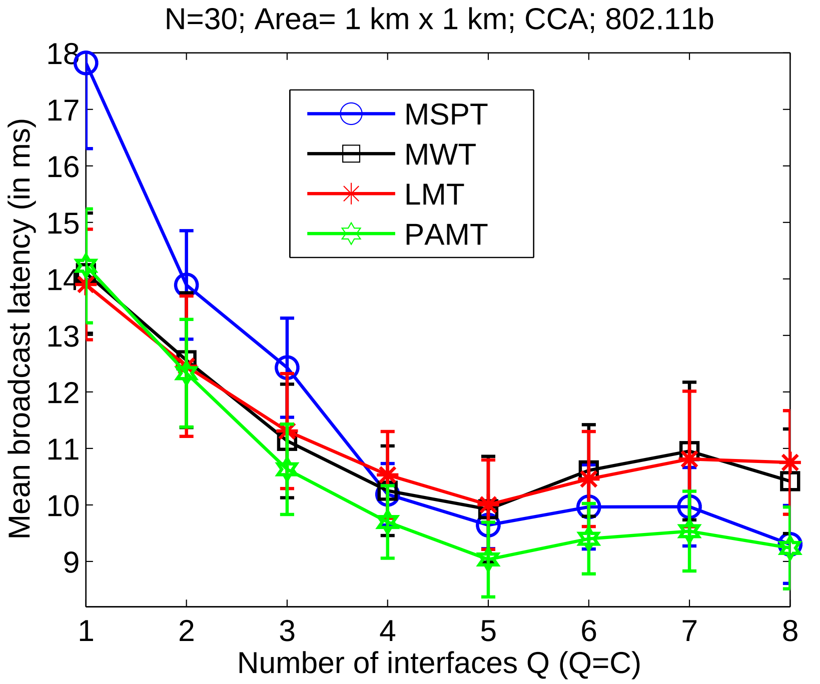
The effect of network's node density on the performance of our heuristics can be seen in Figures 9.12(a) and 9.13(a) for 802.11b and 802.11a networks, respectively. It is observed that the PAMT algorithm performs the best of all algorithms for the range of network node density (20 to 100 nodes in an area of 1000^2 m^2). The performance of LMT, although it uses parallelization like PAMT, is not as good as PAMT's (Figures 9.12(a) and 9.13(a)) as in the idealized MAC case where LMT performed better than MWT but worse than PAMT. The performance of MSPT, expectedly, is poor and worsens as the network node density is increased.

9.8.2 Effect of number of radio interfaces

The effect of varying number of radio interfaces Q on the performance of our algorithms can be seen in Figures 9.12(b) and 9.13(b) for 802.11b and 802.11a networks, respectively. It is observed that, for values of Q as low as 2 or 3, PAMT outperforms all other algorithms. It is also seen that MSPT's performance improves with increasing values of Q; MSPT performs comparably to MWT and LMT for values of Q as low as 4 and 3 (Figure 9.12(b) and 9.13(b)) for 802.11b and 802.11a networks, respectively. However, MSPT requires a very high value of Q ($Q = 8$ for 802.11b and $Q = 4$ for 802.11a) to match PAMT's performance. It must be pointed out here that MSPT is the ideal solution to the MLB problem if we can assume (impractically) that there are infinite number of Q.



(a) varying nodes while using 802.11b

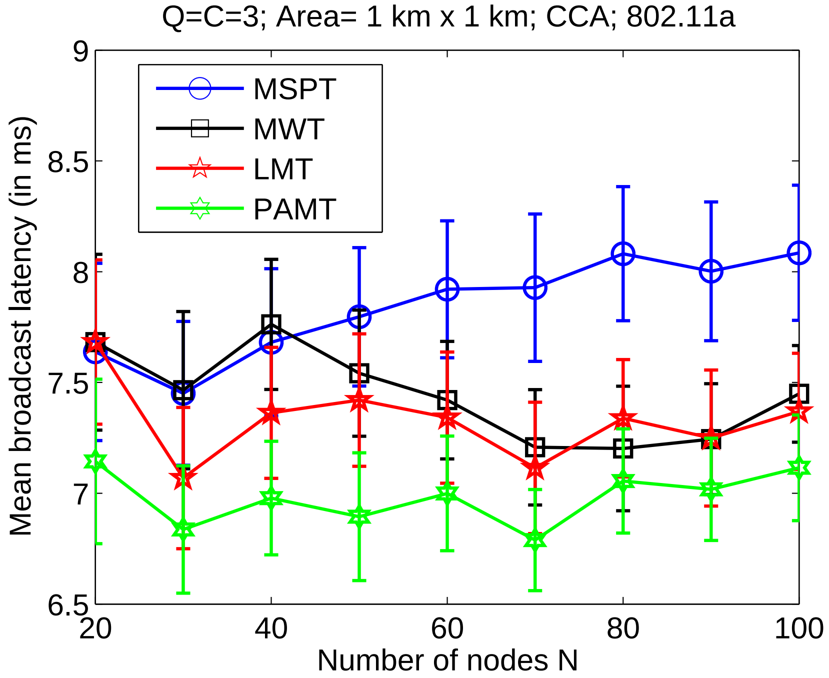


(b) varying radio resources while using 802.11b

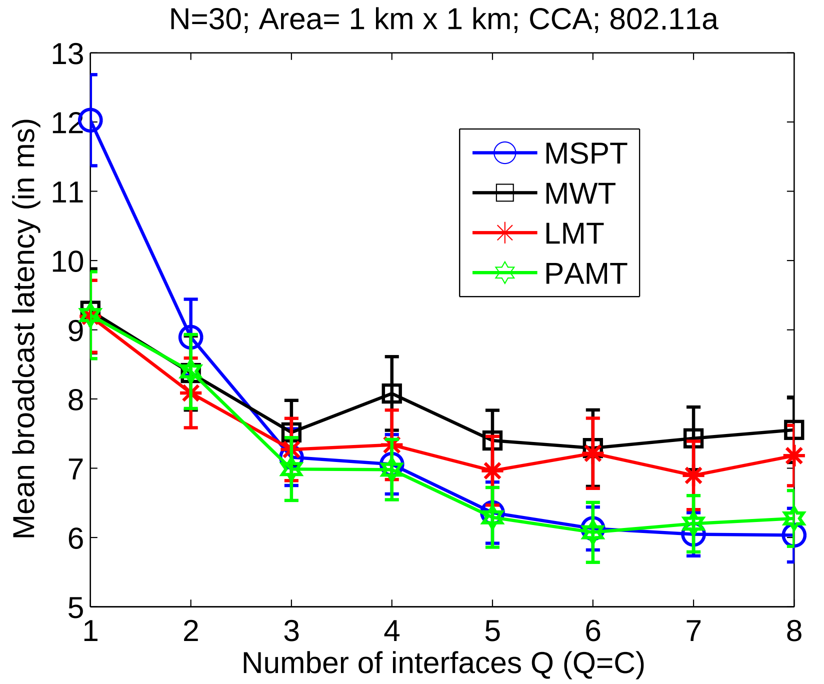
Figure 9.12: Normalized broadcast latency for varying nodes and radio resources in an area of 1000×1000 m^2 for 802.11a or 802.11b networks using Qualnet simulator

9.8.3 Effect of channel-assignment scheme

The performance of MSPT, MWT, LMT and PAMT for different channel-assignment schemes is shown in Figures 9.14(a), 9.14(b), 9.15(a) and 9.15(b), respectively. We note that the algorithms that incorporate WBA in their design (i.e., MWT, LMT



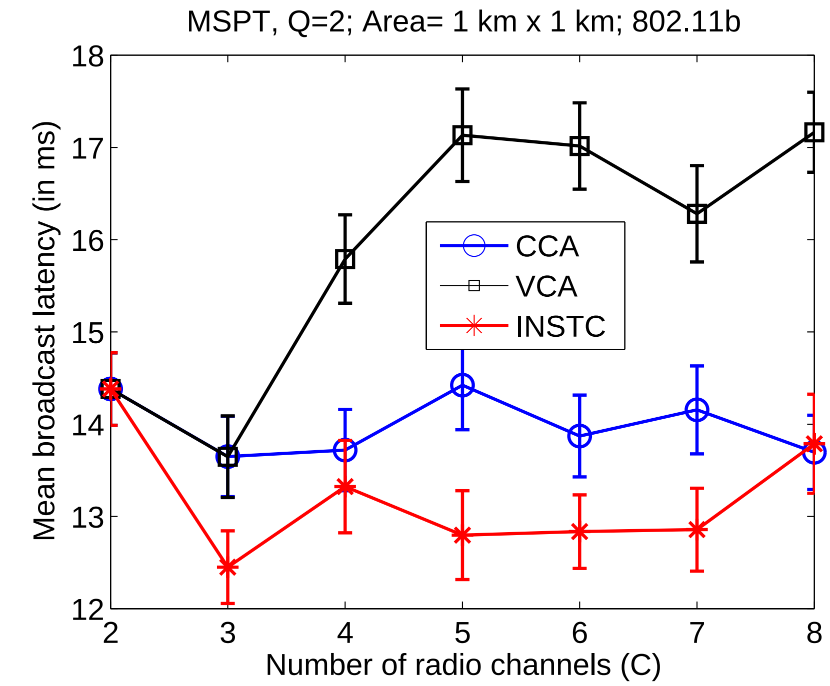
(a) varying nodes while using 802.11a



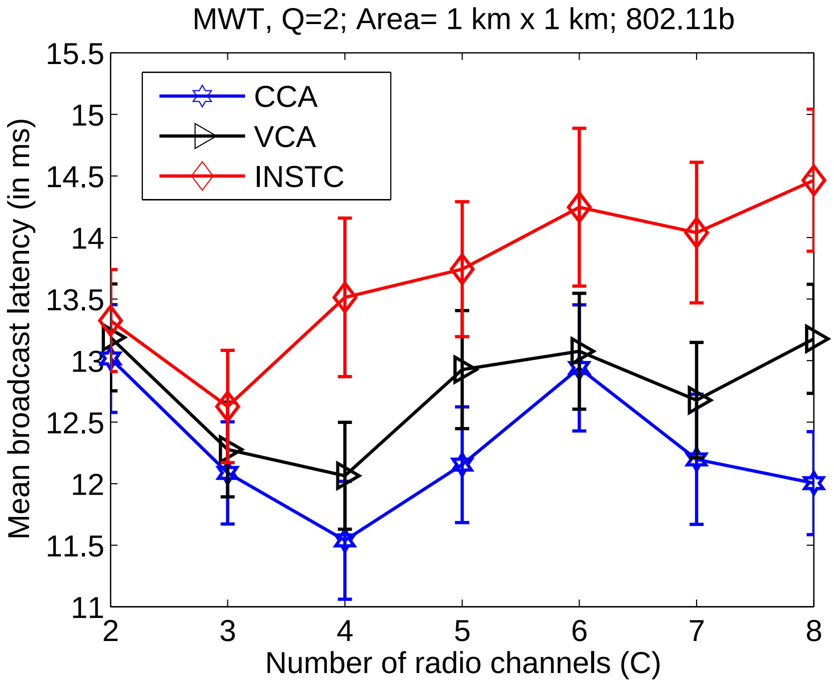
(b) varying radio resources while using 802.11a

Figure 9.13: Normalized broadcast latency for varying nodes and radio resources in an area of 1000×1000 m^2 for 802.11a or 802.11b networks using Qualnet simulator

and PAMT) can benefit from increased 'connectivity', present in schemes like CCA, which presents more opportunities for exploiting WBA. These results are similar to those presented in Section 9.7.4 where an idealized MAC scheduler was used. Interestingly, MSPT algorithm, which does not take WBA into account, behaves like a typical unicast protocol wherein INSTC presents better results than CCA or VCA.



(a) MSPT

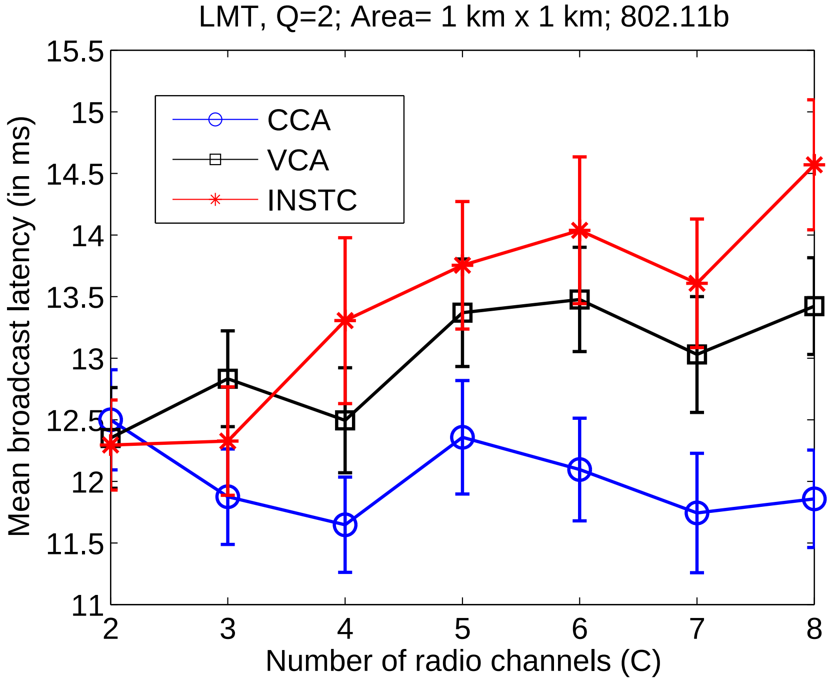


(b) MWT

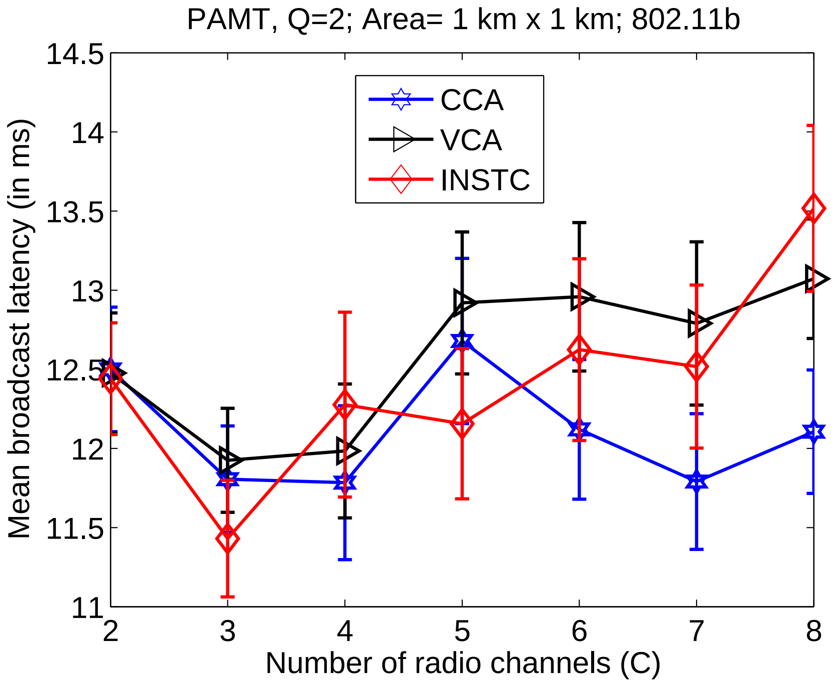
Figure 9.14: Normalized broadcast latency for different channel-assignment techniques for varying C (Q=2, Area= 1000 $\times$ 1000 m^2) for 802.11b networks using Qualnet simulation

9.9 Simulation results in Qualnet for a stream of broadcast packets

We have noted in Sections 9.7 and 9.8 that PAMT improves the performance of MWT and LMT through increased parallelization. These results are directly rel-



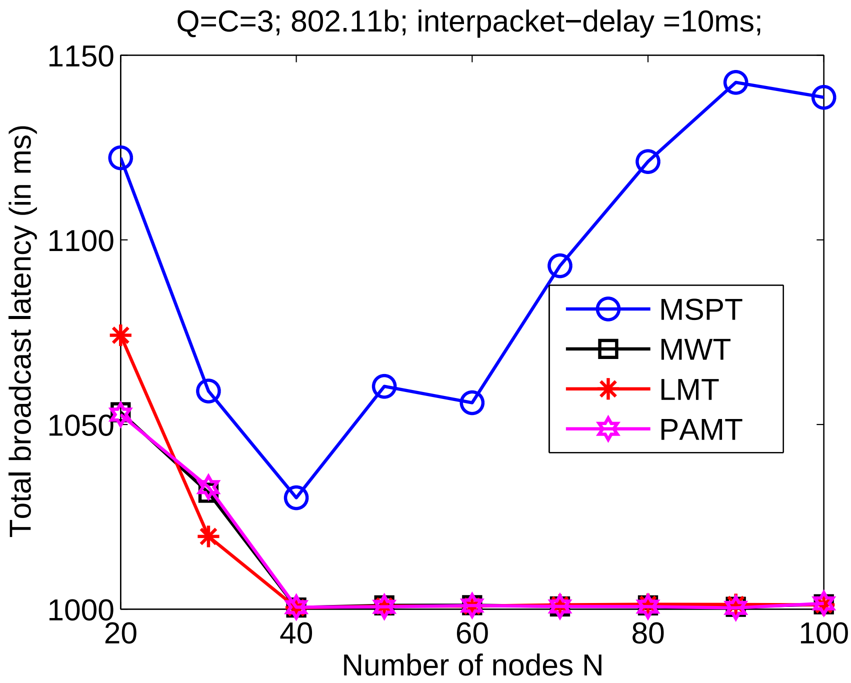
(a) LMT



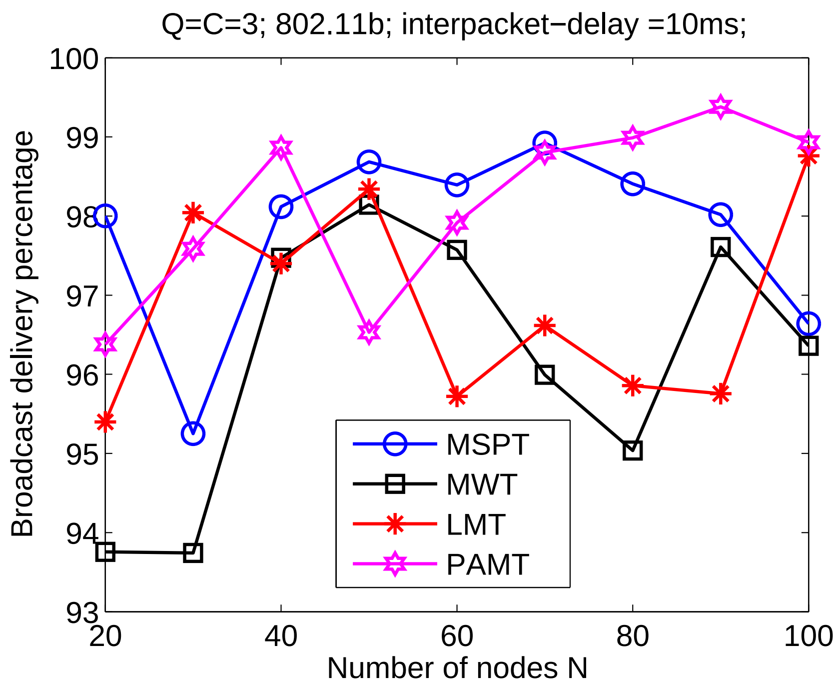
(b) PAMT

Figure 9.15: Normalized broadcast latency for different channel-assignment techniques for varying C (Q=2, Area= 1000 $\times$ 1000 m^2) for 802.11b networks using Qualnet simulation

evant to the case of a single broadcasted packet. Since most broadcast comprises of a stream of transmitted packets, we are also interested in knowing our protocol's broadcast performance for a stream of packets. Towards this end, we have programmed the Qualnet simulator to simulate the broadcast of a stream of 100



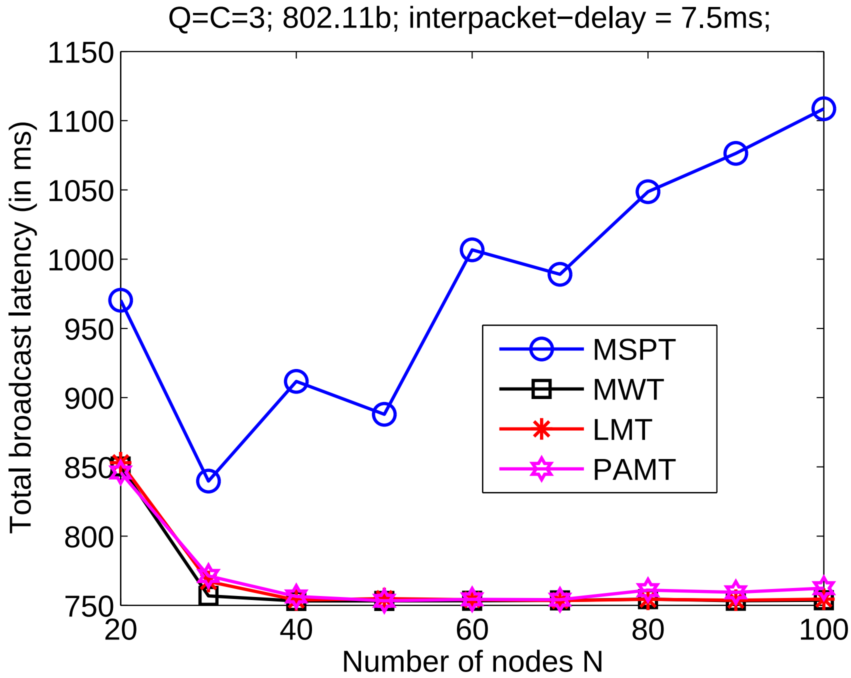
(a) Broadcast delay (secs); interpacket delay=10ms



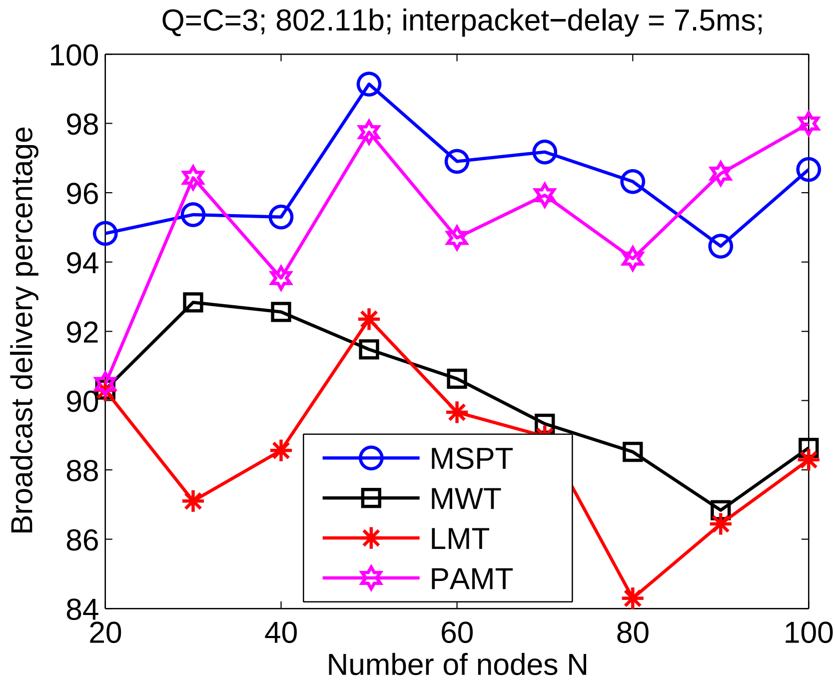
(b) Delivery percentage; interpacket delay=10ms

packets where each packet comprises of 1500 bytes. We assume that successive packets of the broadcast stream are separated in time by an interval called the *'interpacket delay interval'*.

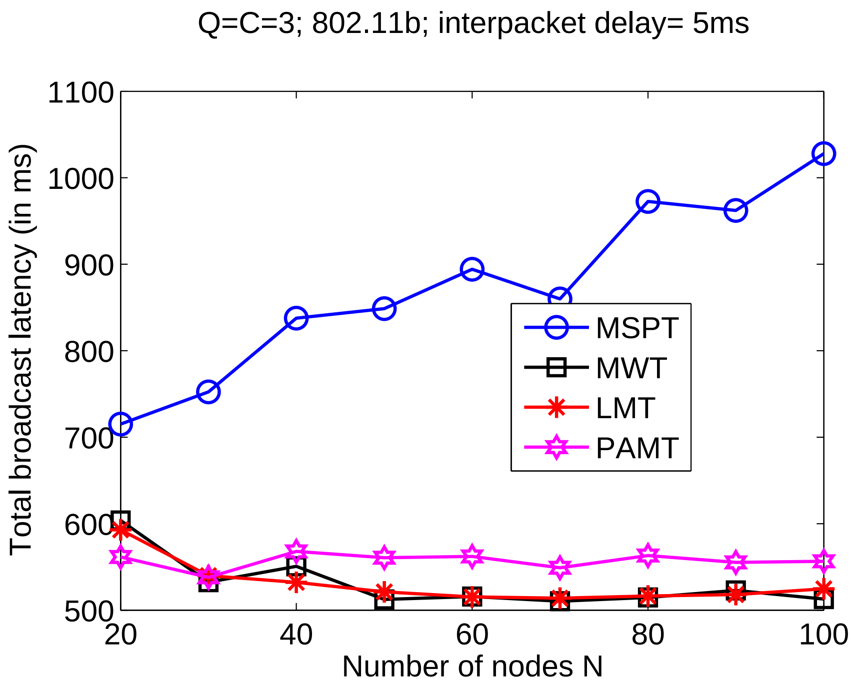
We will compare our algorithms for a stream of broadcast packets using two metrics: 1) the *'total broadcast latency'*, and 2) the *'broadcast delivery percentage'*. The *'total broadcast latency'* is defined as the time taken from the transmission of the first packet till the time the 100th packet of the broadcast stream is received



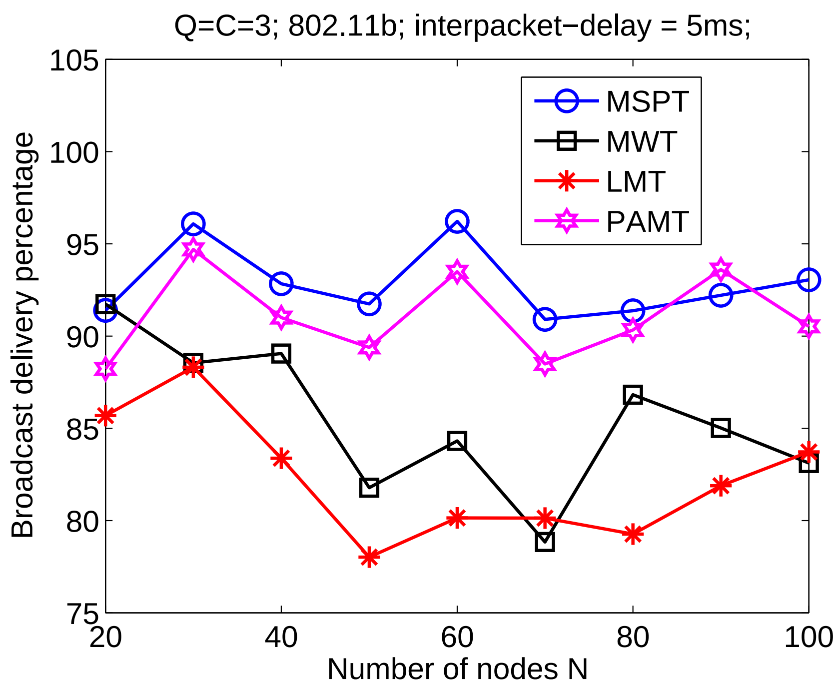
(c) Broadcast delay (secs); interpacket delay=7.5ms



(d) Delivery percentage; interpacket delay=7.5ms



(e) Broadcast delay (secs); interpacket delay=5ms



(f) Delivery percentage; interpacket delay=5ms

Figure 9.16: Broadcast delay and delivery percentage for varying interpacket delays

at *all* nodes. We note that successive packets of the broadcast stream is delayed by the interpacket delay interval. If we do not place the condition that the 100th packet be received at all recipient nodes, then comparing our algorithms by stamping the time all the nodes have received its last packet can provide skewed results. However, with the condition that 100th packet must be received, packets lost earlier do not make a big difference, since the 100th packet, which is received at all nodes, started at a fixed time (dictated by the interpacket delay interval); thus with the condition enforced that the 100th packet is received at *all* nodes, a more consistent comparison can be made between our different algorithms. The *'broadcast delivery percentage'*, on the other hand, is defined as the average (over all network nodes) of the average number of packets received at a network node amongst the 100 packets of broadcast stream. The 'total broadcast latency' results are displayed in Figures 9.16(a), 9.16(c) and 9.16(e) for different interpacket delay intervals of 10, 7.5, and 5 ms, respectively; similarly, 'broadcast delivery percentage' results for these respective interpacket delay intervals are displayed in Figures 9.16(b), 9.16(d) and 9.16(f). *These results assume the CCA channel-assignment.*

The simulation results present interesting results. Whereas for a single broadcasted packet, PAMT returned significant broadcast latency performance improvement over MWT and LMT (due to increased parallelization), we see that for a stream of broadcasted packets, the *'total broadcast latency'* performance of MWT, LMT and PAMT is similar (shown in Figures 9.16(a), 9.16(c) and 9.16(e)). In fact, with decreasing interpacket delay interval (e.g., with 5 ms, Figure 9.16(e)), we see that the *'total broadcast latency'* performance of MWT and LMT is better than that of PAMT. This results from the fact that PAMT's increased parallelization, as it utilizes more radio resources, is not helpful to pipelined transfer of broadcasted packets. This is unlike the case of MWT and LMT algorithms where the parallelization is more conservative, and pipelined transfer of broadcast packets is not impeded as much. The *'total broadcast latency'* performance of MSPT, on the other hand, is quite poor compared to the other three algorithms for all considered interpacket broadcast packet intervals.

The other results that we have studied measure the *'broadcast delivery percentage'*. This is calculated by taking the average percentage delivery of all the broadcast receiving nodes. For our presented results, we have placed the restriction that the last packet (the 100th packet) be received at all the broadcast receiving nodes; however, other packets might be lost and are duly recorded by the broadcast delivery percentage. The results show that the delivery percentage is inversely proportional to the broadcast latency results, i.e., for a given interpacket broadcast interval, the higher the broadcast latency, the better the delivery ratio. This, probably, results from the fact that a interpacket broadcast interval similar to the broadcast latency results in large queues at the transmitting nodes and causes a larger loss probability. On the other hand, when the broadcast latency is larger than the interpacket broadcast interval, the queues at the transmitting nodes are not as loaded resulting in fewer losses. We note, therefore, in our results that the delivery percentage performance of MSPT and PAMT is better than the performance of MWT and LMT, especially when the interpacket broadcast interval is reduced as seen in Figures 9.16(a), 9.16(c) and 9.16(e).

9.10 Conclusions

In this chapter, we have studied the problem of MLB problem for MR^2-MC WMNs. We have proposed four centralized MLB heuristics which exploit the rate-diversity of MR^2-MC WMNs. The first of our algorithm (MSPT) does not exploit the 'wireless broadcast advantage' (WBA), the second (MWT) exploits WBA but not the availability of multiple interfaces on the same node, while the other two (LMT and PAMT) differ in how they exploit both WBA and the radio-and-channel-diversity on individual nodes.

We study the performance of our algorithms through detailed simulations using both 1) an idealized scheduler (with idealized MAC assumptions) and 2) a practical IEEE 802.11 MAC based scheduler. We show that the PAMT outperforms all of our other algorithms showing the benefit of exploiting the rate-diversity, the radio-and-channel-diversity, the WBA of MR^2-MC WMNs. It also demonstrates the benefit of increased parallelization of transmissions in time, such that multiple transmissions simultaneously take place on different interfaces (of the same or different node).

The simulation results and performance studies (for both the idealized and 802.11-based scheduler) also show the impact of channel-assignment strategies on broadcast latency, due to the conflict between greater connectivity and lower channel contention. Perhaps a more important observation is that a channel-assignment scheme designed for unicast flows may sometimes perform poorly for broadcast/multicast flows. In our simulations for both the idealized and 802.11-based scheduler, the performance of CCA (which generally performs poorly for unicast flows) is generally better than both VCA and INSTC.

Chapter 10

Distributed MLB Solution for MR^2-MC WMNs

10.1 Introduction

In Chapter 9, centralized heuristics were presented for the MLB problem in MR^2-MC WMNs that exploit such networks' rate-diversity and radio-and-channel-diversity. Although these centralized algorithms do significantly lower the broadcast latency, they incur large communication overheads due to their requirement of global information. The focus of this chapter is on the design and performance evaluation of *localized* and *distributed* rate-diversity aware tree construction techniques that require only 2-hop topology information and can function even when the global network topology is unavailable. Our objective is to perform distributed broadcast such that the resulting broadcast latency is fairly close to that achieved by centralized heuristics.

We will now briefly survey the four centralized heuristic MLB algorithms which were proposed in Chapter 9 for MR^2-MC WMNs that exploit MR^2-MC WMN's rate-diversity and radio-and-channel-diversity. The MWT, 'multi-radio WCDS tree', is a direct extension of the WCDS algorithm. WCDS performs well in SR-SC WMN since it considers both the multi-rate nature of the network and the WBA of the underlying wireless medium. However, MWT, as WCDS, does not utilize multiple interfaces on each WMN node to exploit the advantage of parallel transmissions. This drawback of MWT motivated the development of LMT, 'locally parallelized, multi-radio WCDS tree', algorithm which parallelizes transmissions *locally* on a node's different interfaces (tuned to orthogonal channels) to have multiple overlapped transmissions. The $PAMT$, 'parallelized approximately-shortest multi-radio WCDS tree' algorithm, also adapted from the MWT algorithm like the LMT algorithm, improves LMT's performance by extending the parallelization scope to also include interfaces on *other* nodes. The PAMT tree also adapts to the number of radio resources available by resembling a WCDS tree for a SR-SC WMN, and a 'shortest path tree' (SPT) for a MR-MC WMNs with infinite radio interfaces and channels.

Since, our centralized MLB algorithms for MR^2-MC WMNs are essentially 'connected dominating set' (CDS) based algorithms, we will now examine the existing literature for building CDS backbones distributively. Although there is a variety of distributed data-broadcasting algorithms [32–34, 45] which compute a set of 'backbone' nodes, these algorithms are directly applicable only to the more primitive case of 'single-radio, single-channel' (SR-SC) single-rate WMNs. The existing CDS based algorithms fail to consider that data forwarding in MR^2-MC WMNs must make more effective use of the network's rate-diversity and radio-and-channel-diversity. These backbone-based CDS algorithms have already been discussed in Section 2.1.2. The heuristics that will be presented in this chapter are based on significant modifications to two underlying (rate-diversity unaware) techniques that both calculate a 'small' CDS by first computing a large CDS and then pruning away redundant transmissions. Both these algorithms (*Wu-Li* algorithm [32], and the 'multi-point relaying' (*MPR*) algorithm [43]) have been discussed in detail in Sections 2.1.2 and 5.1. We will briefly reintroduce the relevant details below for completeness.

The first of these algorithms, called the *Wu-Li* algorithm [32], is a simple localized technique (using only 2-hop neighborhood information) which computes a CDS in two phases: 1) the marking phase, and 2) the pruning phase. Initially, all vertices (nodes) are unmarked. The marking process uses the following simple rule: any vertex having two unconnected neighbors (not connected directly) is marked as a dominator. The set of marked vertices form a rather large CDS V'. Two pruning techniques are then used to reduce the CDS size. A node u can be removed from V' if there exists a node v with higher ID such that the closed neighbor set[1] of u is a subset of the closed neighbor set of v. For the same reason, a node u will be deleted from V' when two of its connected neighbors in V' with higher IDs can cover all of u's neighbors. This pruning idea is generalized to the following rule [32]: a node u can be removed from S if there exist k connected neighbors with higher IDs in S that can cover all u's neighbors. *The second algorithm that we consider*, the MPR technique [43], also computes a CDS using localized information. The MPR technique allows each node u to first elect a 'multi-point relay set' (MRS) [48] [33] from its one-hop neighbors to cover its two-hop neighbors. The CDS is calculated as follows [43]: each node first computes a MRS, a subset of one-hop neighbors that can cover all its two-hop neighbors. After each node has determined its MRS, a node decides that it is in the connected dominating set by matching either Rule 1: the node is smaller than all its neighbors *or* Rule 2: it is multipoint relay of its smallest neighbor. Although neither of these two relatively simple algorithms necessarily form the smallest CDS, we shall use them as candidates in our tree-formation process, as the subsequent steps of rate and channel diversity maximization (which we

[1] Closed neighbor set is the union of the node itself and its neighbors.

introduce in Section 10.3.3) turn out to be more important for MR^2-MC networks than the computation of a small sized initial CDS.

Not much research attention has focussed on building distributed multi-rate broadcasting protocols. From the little work done, most have focussed on SR-SC multi-rate WMNs. The "rate-adaptive multicast" (RAM) algorithm was proposed for SR-SC multi-rate MANETs in [84] based on the "on-demand multicast routing protocol" (ODMRP) [37]. We have proposed the 'Multi-radio delayed-pruning Wu-Li' (MDW) algorithm in Chapter 5 for SR-SC multi-rate WMNs. Lastly, a 'Distributed Rate-First' algorithm has recently been proposed for use in SR-SC multi-rate WMNs [70]. All these three algorithms do not utilize the radio-and-channel-diversity available in a MR^2-MC WMN. Although Song et al. have recently proposed a distributed broadcast algorithm for multi-channel multi-rate WMN [68], this algorithm assumes that each channel can only use a particular rate. In other words, for the work of Song et al. [68], once a node has decided to use a particular channel, the link-layer transmission rate to be used by that node cannot vary. However, our setting is completely general where any transmission rate can be used with any channel

The main contributions of this chapter are highlighted below:

1. *We propose a 4-staged FMM localized and distributed MLB heuristic solution, called MRDT, for MR^2-MC WMNs.* MRDT can match the performance of the centralized heuristics of Chapter 9 especially for large networks while utilizing only localized network information in a distributed manner.

2. It analytically determines the complexity of the presented heuristics and demonstrates that our heuristics can scale to large networks.

Chapter Outline: The rest of this chapter is organized as follows. We introduce our network model in Section 10.2. We describe MRDT, our distributed broadcasting framework comprising of four distinct stages, in Section 10.3. We present performance results of MRDT assuming a practical decentralized MAC scheduler (IEEE 802.11 MAC) in Section 10.4. We conclude our work in Section 10.5.

10.2 Network model

We assume the FMM broadcasting framework (introduced in Chapter 4) in which WMN nodes can adjust link-layer multicast transmission rate for each link-layer frame. Our network and interference model is similar to that described in Chapter 9. We will reproduce the relevant details here for completeness. We assume that there are C orthogonal channels in the system with each node equipped with Q

V	Set of vertices (or, nodes)
E	Set of edges (or, links)
Π	Set of transmission-rate of links in E
Λ	Set of channel of links in E
Q	Number of radio interfaces at each node
C	Total number of orthogonal radio channels
$\mathcal{C}(u)$	Set of channels node u's Q interfaces use
N	Total number of nodes in network (=$\|V\|$)
u_i	i^{th} interface of node u
ρ_i	i^{th} highest transmission rate supported by MAC
$\rho(u_i)$	Current transmission-rate of u_i
ρ_0	Rate of a non-transmitting interface
L	Number of distinct rates supported by MAC
$\mathcal{C}(u_i)$	Channel interface u_i is tuned to
$N(u_i)$	1-hop neighbors that u_i is currently covering
$N_{\rho_k}(u_i)$	1-hop neighbors of u_i (on rate ρ_k)
$r(u_i)$	Set of rates u_i having a 'rate-limiting-node'
$r_0(u_i)$	Rate of u_i when it is not transmitting
bin_i	Set of neighbors of a node's i^{th} interface
$\rho(bin_i)$	transmission-rate of a node's i^{th} interface
$\pi(u,v)$	Highest transmission-rate link (u,v) can use
$\lambda(u,v)$	Channels link (u,v) can use
m	Number of marked-nodes
d	maximum number of neighbors of a marked-node

Table 10.1: Index of mathematical symbols used in this chapter

interfaces where $Q \leq C$. We assume that two radio interfaces at the same node are not tuned to the same channel.

We represent the nodes in the network by V. The total number of nodes ($|V|$) in the network is represented by N. A channel assignment $\mathcal{C}$ assigns each vertex $v \in V$, Q different channels denoted by the set: $\mathcal{C}(v) = \{\mathcal{C}(v_1), \mathcal{C}(v_2), \ldots, \mathcal{C}(v_Q) : \mathcal{C}(v_i) \neq \mathcal{C}(v_j), \forall i \neq j\}$ where $\mathcal{C}(v_i)$ denotes the channel assigned to interface i of v. The topology defined by $\mathcal{C}$ is represented by $G = (V, E, \Pi, \Lambda)$ where V, E, Π, Λ are the set of nodes, links, rates of links and channel of links, respectively. The quickest-rate transmission supported between u and v is denoted by $\pi(u, v)$. The channels shared between two nodes u and v is denoted by $\lambda(u, v)$. The network topology is represented by G in the following natural way: an edge $e = (u, v)$ between two nodes u and v on channel k ($\lambda(e) = k$) is in G, *if and only if*, $d(u, v) \leq r$ and $\lambda(e) \in \mathcal{C}(u) \bigcap \mathcal{C}(v)$. The rate of the edge e is the fastest transmission rate supported on e. The set Π contains the rate of each edge in E; similarly the set Λ contains the channel used on each edge in E. Note that G may be a *multi-graph*, with multiple edges between the same pair of nodes, when the node pair shares two or more channels. We assume that there are L different transmission rates supported by the MAC layer which are represented by $\rho_1, \ldots, \rho_L$ where $\rho_1 > \rho_2 > \ldots > \rho_L$. For mathematical compactness, we denote the transmission rate of a non-transmitting interface as ρ_0. The interface i at node u is represented by u_i, and its transmitting rate by $\rho(u_i)$. $N_{\rho_k}(u_i)$ refers to the neighbors of u_i that share a channel with u_i and can use a maximum rate of ρ_k to connect to u_i.

We assume that channel assignment is performed independently of our broadcasting framework. Further, we assume that each node knows its neighbors as well as the interfaces and rates it can use to reach them. The rate-adaptation, for example, can be performed using any of the frame-error based adaptation [85], throughput-based adaptation [86], or the SNR-based adaptation [18] techniques. In order to maintain bidirectional connectivity, the rate that nodes u and v can use to reach each other is the minimum rate that can be used in the two directions. We assume that both broadcast and unicast traffic will coexist on the network; accordingly, the 'current rate' of a particular link between any two nodes can actually be inferred from the rates to which the unicast flows converge. The two-hop topology information can be built by having each node broadcast a packet containing information about sending node's node-ID, neighbors on different rates and neighbors on different channels.

10.3 The MRDT Framework

We will now propose a distributed and localized framework, called 'multi-radio distributed tree' (*MRDT*), that calculates low latency trees for broadcast in MR^2-MC WMNs in four logically independent stages. *Firstly*, the 'initial marking' stage, which is unaware of rate-diversity and radio-and-channel-diversity of the network, initially approximates the forwarding-node set and forms a CDS. *Secondly*, the 'neighbor grouping' *(NG)* stage, which is also rate-diversity and radio-and-channel-diversity unaware, decides the neighboring nodes a marked node has to cover. *Thirdly*, the 'rate maximization' *(RM)* stage, comprising of two sub-stages, maximizes the transmission rates at all the marked nodes; the first sub-stage, called 'local rate maximization' (*LRM*), attempts rate maximization locally at a marked node by parallelizing its transmissions over its interfaces, while the second sub-stage called 'external rate maximization' (*ERM*) attempts rate maximization at a node by 'exporting' its neighbors, that are limiting its rate, to other marked nodes. *Lastly*, the 'tree construction' stage constructs a source-specific broadcast tree that takes into account WBA and prunes redundant transmissions retained in earlier stages. The stages of our broadcasting framework, and specific algorithms used during them, are covered in more detail in the following subsections.

10.3.1 Stage 0—Initial Marking:

This stage can initially approximate the forwarding set (also called the CDS) by using three methods. *Firstly*, Wu-Li marking process (see Section 10.1) can be used in which a node is marked if it has two neighbors (at the lowest rate) that are not directly connected. *Secondly*, MPR based marking process (see Section 10.1) can be used in which a rate-diversity aware algorithm like WCDS (Chapter 4) is used to generate the MRS of each node i.e., each node executes WCDS algorithm with itself as the source on its 2-hop neighborhood subgraph to determine the set of its one-hop neighbors to cover its entire 2-hop neighborhood. *Lastly*, all the nodes can be marked as eligible forwarders; whereas such an approach results in a large CDS, we shall see in Section 10.4 that this approach returns good results as the rate-maximization steps (in following stages) can exploit the larger CDS.

10.3.2 Stage 1—Neighbor Grouping (NG)

We decide the neighboring nodes a marked node has to cover in the NG stage. The intuition is straight-forward, a marked node's transmission rate should not be constrained to a lower rate to cover a node that can be, alternatively, be 'better' covered by another marked node. This stage exploits the redundancy available in wireless networks where a node is potentially covered by many transmitters.

Algorithm 12 Algorithm for 'Neighborhood Grouping' at a marked node u

$N_{\rho_L}(u) = \cup_{i=1,\ldots,Q}\ N_{\rho_L}(u_i)$
for each one-hop neighbors $v \in N_{\rho_L}(u)$ **do**
  for each *marked* node $w \in N_{\rho_L}(u)\backslash\{v\}$ **do**
    if $1/\pi(u,v) > 1/\pi(u,w) + 1/\pi(w,v)$ **then**
      remove v from neighbor-list of u at rate $\pi(u,v)$
    end if
  end for
end for

In the NG stage, explained in Algorithm 12, at each node u, it is searched if there exists a 1-hop neighboring node v that can be 'better' covered by w (a 1-hop *marked* neighbor of u). The node v is said to be better covered by w if the aggregate throughput/rate of the path $u \to w \to v$ is better than the throughput of the path $u \to v$. The 1-hop neighborhood of each marked node is decided at the completion of this stage; each marked node is now responsible for ensuring coverage, by itself or through another *connected marked node*, of its 1-hop neighborhood. We illustrate the NG stage using a simple example. Let us assume node u can reach v and w using 1 Mbps and 11 Mbps link, respectively. Let us also assume that node v can be covered by w using a 11 Mbps link. Since the condition $1/\pi(u,v) > 1/\pi(u,w) + 1/\pi(w,v)$ of Algorithm 12 is satisfied (as $\frac{1}{1} > \frac{1}{11} + \frac{1}{11}$), v is removed from the neighbor-list of u at the rate of 1 Mbps.

Theorem 2 *The computational complexity of the NG stage at a marked node u is $O(d^2)$ where d represents the maximum degree of a marked node.*

Proof 2 *The computational complexity of the NG stage is $O(d(d-1)) = O(d^2)$ at the marked node with maximum neighbors d that has all its neighbors marked. The total computational complexity for the network is $O(md^2)$ where m is the number of marked nodes.*

Message Complexity: Assuming that 2-hop neighborhood information has been established prior to the NG stage, no message needs to be exchanged during the NG stage. After the NG stage finishes, each marked node will broadcast a *NEIGHBOR* packet for a total of m *NEIGHBOR* packets—the maximum size of which is $(1 + (Q + L)d)$ times the bytes required to represent a node-id—to convey the sending node's node-ID, neighbors on different channels and rates, to all its neighbors. We note that Q and L are small (constant) values since typically limited interfaces and rates are supported; the total message-complexity of the NG stage, therefore, is $O(md)$.

10.3.3 Stage 2—Rate Maximization (RM)

Before discussing the RM stage, we introduce the concept of '*rate-limiting-nodes*'. We note that a lower-rate transmission can cover all nodes reachable at a higher-rate but not vice-versa; this implies that the maximum rate, a node u can use to reach all its 1-hop neighbors $N(u)$ collectively, is the minimum of the (maximum) rate u can use to reach each individual node in $N(u)$. To illustrate this concept, assuming a single radio interface, refer to Figure 10.1 for an example topology. Although, u can reach nodes w, x and y with a rate of 5.5, 11 and 54 Mbps, respectively, u is constrained to transmit at a lowest-rate of 2 Mbps to reach node v. Node v, for this topology, is referred to as a rate-limiting-node since its presence limits the rate of u to 2 Mbps and its absence can increase the rate of u to 5.5 Mbps.

The two sub-stages of Stage 2, i.e. LRM and ERM, differ in how they deal with rate-limiting-nodes. LRM exploits interface-diversity at each marked node to 'export' the rate-limiting-node to a different interface *on the same node*, thereby increasing the original interface's rate; with this export, LRM also exploits the channel-diversity since different interfaces on a node are tuned to orthogonal channels. As the 'export' scope of rate-limiting-nodes in LRM is confined to local interfaces on the same node, LRM is similar to the LMT algorithm which parallelizes transmissions locally over the node's interfaces. ERM, on the other hand, extends the scope of exporting rate-limiting-nodes to include interfaces at other nodes (subject to a few conditions) like the PAMT algorithm in which an attempt is made to parallelize a node's transmissions in its neighborhood. The details of LRM and ERM follow next.

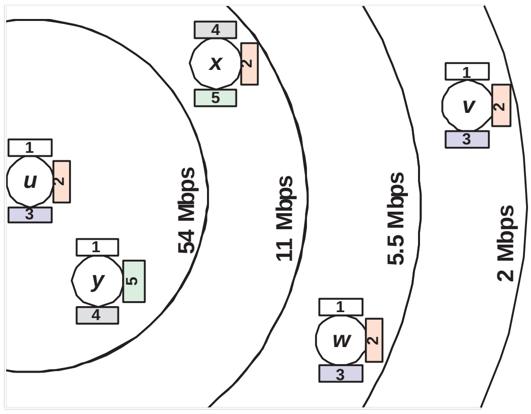


Figure 10.1: Before 'Local-Rate-Maximization' at node u

Step	bin_1	bin_2	bin_3	$\rho(u_1)$	$\rho(u_2)$	$\rho(u_3)$	$\delta(\texttt{sum})$
0	$\emptyset$	$\emptyset$	$\emptyset$	0	0	0	0
1	$\{y\}$	$\emptyset$	$\emptyset$	54	0	0	54
2	$\{y\}$	$\{x\}$	$\emptyset$	54	11	0	11
3	$\{y\}$	$\{x\}$	{w}	54	11	5.5	5.5
4	$\{y\}$	$\{x\}$	{w,v}	54	11	2	-1.5

Table 10.2: The steps of LRM at u of Figure 10.1

Local Rate Maximization (LRM)

We illustrate the working of LRM for a simple topology shown in Figure 10.1 in which each node has 3 interfaces. The channel to which u_i (the i^{th} interface of u) is tuned to is represented by $\mathcal{C}(u_i)$; also, $\mathcal{C}(u)$ represents the channel set of u, i.e., $\mathcal{C}(u) = \cup_{i=1}^{Q} \mathcal{C}(u_i)$. In Figure 10.1, $\mathcal{C}(u) = \{1, 2, 3\}$, $\mathcal{C}(v) = \{1, 2, 3\}$, $\mathcal{C}(w) = \{1, 2, 3\}$, $\mathcal{C}(x) = \{2, 4, 5\}$ and $\mathcal{C}(y) = \{1, 4, 5\}$. Initially, the value of $\texttt{sum}$ is equal to 0 as depicted in Table 10.2 since none of the interface is transmitting (i.e., $\forall i$, $\rho(u_i) = 0$). Node u connects to 4 neighbors; the interface neighbor set for first interface $N(u_1)$ $= \{v, w, y\}$, similarly, $N(u_2) = \{v, w, x\}$ and $N(u_3) = \{v, w\}$. Furthermore, u can connect to $\{y\}, \{x\}, \{w\}$ and $\{v\}$ on 54 Mbps, 11 Mbps, 5.5 Mbps and 2 Mbps, respectively. The current rate $\rho(u_i)$ at bin_i is then determined as the minimum of the rates to individual nodes in bin_i where bin_i denotes the set of nodes assigned to the i^{th} interface of u. During each step, the node and bin combination that maximizes $\delta(\texttt{sum})$ is chosen. During step 1, y is added to bin_1 which gives $\texttt{sum} = 54$ $(54 \times 1 + 0 + 0)$ at the end of step 1 for maximum $\delta(\texttt{sum})$ of 54. Similarly, for step 2, x is added to bin_2 with $\texttt{sum} = 65$ $(54 \times 1 + 11 \times 1 + 0)$ for maximum $\delta(\texttt{sum})$ of 11. During the following step (step 3 in Table 10.2), w can be added to bin_1, bin_2 or bin_3 since w belongs to each of $N(u_1)$, $N(u_2)$ and $N(u_3)$. However, adding w to bin_1 and bin_2 would cause $\delta(\texttt{sum})$ to be -43 and 0, whereas adding w to $\delta(\texttt{sum})$ would make $\delta(\texttt{sum})$ equal to 5.5. To maximize the incremental $\delta(\texttt{sum})$, we assign w to bin_3. Similarly, in step 4, v can be assigned to bin_1, bin_2 or bin_3 with $\delta(\texttt{sum})$ for

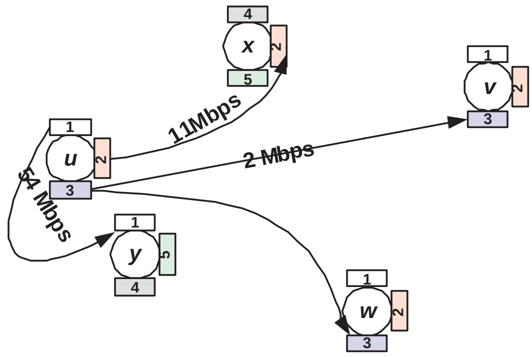


Figure 10.2: After 'Local-Rate-Maximization' at node u

each assignment being -50, -7 and -1.5 respectively. The assignment of v to bin_3 is chosen since it maximizes the step $\delta(\texttt{sum})$. The final value of bin_i and $\rho(u_i)$, for all values of i from 1 to Q, is shown in Table 10.2. As seen in the last step, $\delta(\texttt{sum})$ can be negative; since when a *bin* is non-empty, addition of new nodes can never increase the *bin* rate as increase of rate will imply disconnection of the nodes in the *bin*. On the other hand, when a node is added to an empty *bin* (i.e., when the interface's rate is zero), the $\delta(\texttt{sum})$ will always be positive.

The LRM stage distributes the neighbors of a node over its interfaces in a manner such that the rate-diversity and the radio-and-channel-diversity of the WMN is exploited. We define $\rho(u_i)$ as the current transmission rate of an interface u_i and $N(u_i)$ as the neighbors of u_i (that share a channel with $\mathcal{C}(u_i)$) that u_i is currently covering at its transmission rate $\rho(u_i)$. During the stage of LRM at any node u, we assign u's neighbors to u's interfaces such that u maximizes the sum, taken over all of u's interfaces, of the product of its interface's rate and neighbors on that interface (i.e., the metric $\sum_{i=1}^{Q} \rho(u_i) \times |N(u_i)|$ denoted by '$\texttt{sum}$' is maximized); if u's j^{th} interface is unused, its transmission rate is zero (i.e., if there is no transmission on u_j, $\rho(u_j) = 0$). By maximizing $\texttt{sum}$, we ensure that nodes are covered in maximum rate transmissions that are parallelized. The LRM sub-stage, analogously to the function performed by the centralized LMT algorithm (Chapter 4), performs *local* parallelization to utilize the radio-and-channel-diversity available at the node.

The LRM algorithm, for any node u, is mathematically described in Algorithm 13. Initially, the transmitting rate at all the interfaces of all nodes is 0; for this reason, all the interface *bins* at all nodes are also empty. U, denoting a set of unassigned (to any *bin*) neighbors, is initialized with all the neighbors of u. Our algorithm then, in each round, determines an unassigned node ($U(\widehat{j})$ in Algorithm 13) that can be placed in an interface *bin* to cause maximum increase in $\texttt{sum}$. We define as $f(j, i)$ (the $\delta(\texttt{sum})$ in Table 10.2) as the change in $\texttt{sum}$ after the j^{th} element of U has been added to the i^{th} interface *bin*. Since in the stage of LRM, our aim is to exploit interface-diversity and WBA simultaneously; a node should suppress a transmission on rate ρ_i on a non-transmitting interface to cover a node that is in range of an already transmitting interface whose rate is ρ_i. Accordingly, the metric $f(j,i)$ is set to -∞, if bin_i is currently empty and $U(j)$ can be placed in a non-empty interface *bin* of u whose current rate is exactly $\pi(U(j),u)$, to eliminate a redundant transmission. After placing a node $U(j)$ in bin_i, the rate for that *bin* (represented by $rate(j,i)$) is calculated as the minimum of the (max) rate to an individual node in the interface *bin*. In case of an empty *bin*, this is equal to maximum rate $U(j)$ can support, i.e. $\pi(U(j),u)$, while in the case of a non-empty *bin*, the rate is $\min(\pi(U(j),u),\rho(u_i))$. Since $rate(j,i)$ can be less than $\rho(u_i)$, nodes already assigned to bin_i can transfer to alternate *bin*s (represented by set O in Algorithm 13) whose current rate is more than $rate(j,i)$ in case of a shared common channel to increase rate. The set of these nodes is called E and the sum of the new rates for each node in E is represented by EG. The $f(j,i)$ metric, defining the desirability of assigning U_j to bin_i, is now calculated as $((1+|bin_i(u)|-|E|)\times rate(j,i)$ + EG - $(|bin_i(u)|\times\rho(u_i))$. An unassigned node $U(\widehat{j})$ and the interface $bin_{\widehat{i}}$ can be decided according to highest f value. If $rate(\widehat{j},\widehat{i})$ is lower than $\rho(u_{\widehat{i}})$, all nodes assigned to $bin_{\widehat{i}}$ check if they can migrate to an interface in set $O(\widehat{j},\widehat{i})$ which comprises of interfaces whose rate range from $rate(\widehat{j},\widehat{i})$ to $\rho(u_{\widehat{i}})$. N, denoting nodes that can migrate to an interface u_o in O, is then added to bin_o and subtracted from $bin_{\widehat{i}}$. $U(\widehat{j})$ can now be added to $bin_{\widehat{i}}$ and taken out from the set of unassigned nodes U. The algorithm completes when U becomes an $\emptyset$ when all neighbors of u have been assigned to an interface *bin*.

Theorem 3 *The LRM algorithm, at any marked node, is a polynomial-time algorithm of $O(d^2Q)$.*

Proof 3 *Assuming that node u has d neighbors, the LRM algorithm would, for $O(d)$ rounds, perform $O(Q)$ operations on $O(d)$ nodes to determine the f metric, in addition to another $O(Q)$ operations for $O(d)$ nodes in the decided bin to decide if they should 'migrate'. The LRM algorithm at any marked node is, therefore, $O(2d^2Q)=O(d^2Q)$. Since the value of Q is a small constant; therefore, the LRM algorithm at any marked node is also $O(d^2)$. The total computational complexity of LRM at all nodes is $O(md^2)$.*

Message Complexity: No message needs to be exchanged during the LRM sub-stage. However, after the LRM sub-stage completes, each marked node broadcasts a *NEIGHBOR* packet—the maximum size of which is $(1+dQ)$ times the bytes required to represent a node-ID—to convey the sender's node-ID and the neighbors it can cover on each of its interface. Each marked node also broadcasts a *RATE* packet which advertises the transmission rate of each of its Q interfaces; the size of *RATE* packet is sum of the bytes required to represent a node-ID and a rate-ID. The total number of message exchanged by LRM is $2m$ and thus the message-complexity of LRM is $O(m)$.

Algorithm 13 Algorithm for 'Local Rate Maximization' at node u

$\rho(u_q) = 0$ and $bin_q = \emptyset$, $\forall q = 1$ to Q
for $q = 1$ to Q **do**
  while $U \neq \emptyset$ **do**
    $f(j,q)$ and $rate(j,q)$=$-\infty$, $\forall j = 1$ to $|U|$, $\forall q = 1$ to Q {Finds the assignment of a node in U to an interface on u that maximizes $\delta(\texttt{sum})$}
    for $j = 1$ to $|U|$ **do**
      $\tilde{\mathcal{C}}= \mathcal{C}(U(j)) \cap \mathcal{C}(u)$
      for $k = 1$ to $|\tilde{\mathcal{C}}|$ **do**
        $i =$ interface of u tuned to $\tilde{\mathcal{C}}(k)$;
        $\widehat{\mathcal{C}} = \mathcal{C} \backslash \tilde{\mathcal{C}}(k)$; $I = \bigcup$ u's interfaces tuned to a channel in $\widehat{\mathcal{C}}$
        if $\exists\ \rho_{I(x)}(u) = \pi(U(j), i)$, $\forall\ x = 1$ to $|I|$ **then**
          $f(j,i) = -\infty$; $bin_{I(x)} = bin_{I(x)} \cup U(j)$
        else
          if $\rho(u_i) \neq 0$ **then**
            $rate(j,i) = \min(\pi(U(j), u), \rho(u_i))$
          else
            $rate(j,i) = \pi(U(j), u)$
          end if
          $O = \bigcup_{\forall u_o \neq u_i} u_o$ **s.t.** $\rho(u_i) > \rho(u_o) > rate(j,i)$
          $E = \cup n \in bin_i$ **s.t.** for any $u_o \in O$, $\pi(n, u_o) > rate(j,i)$
          $EG = \sum_{\forall n \in E} \max_{\forall u_o \in O} \pi(n, u_o)$
          $f(j,i) = (1 + |bin_i(u)| - |E|) \times rate(j,i) + EG$ - $(|bin_i(u)| \times \rho(u_i))$;
          $O(j,i) = O$
        end if
      end for
    end for
    $[\widehat{j}, \widehat{i}] = \arg\max(f)$
    if $\max(f) \neq -\infty$ **then**
      {if assigned nodes should change bin to increase $\delta(\texttt{sum})$}
      if $rate(\widehat{j}, \widehat{i}) < \rho(u_{\widehat{i}})$ **then**
        $O = O(\widehat{j}, \widehat{i})$
        while $O \neq \emptyset$ **do**
          u_m= arg $\max_{\forall u_o \in O} \rho(u_o)$
          $N = \bigcup n \in bin_{\widehat{i}}$ **s.t.** $C(n) \cap C(u_m) \neq \emptyset$
          $bin_o = bin_o \cup N$; $bin_{\widehat{i}} = bin_{\widehat{i}} \setminus N$; $O = O \backslash \{u_m\}$
        end while
      end if
      $\rho(u_{\widehat{i}}) = rate(\widehat{j}, \widehat{i})$; $bin_{\widehat{i}} = bin_{\widehat{i}} \cup U(\widehat{j})$;
    end if
    $U = U \backslash U(\widehat{j})$
  end while
end for

Theorem 4 *The LRM algorithm returns an optimal solution to the problem of allocating neighbors to node's interfaces to maximize* $\sum_{i=1}^{Q} \rho(u_i) \times |N(u_i)|$.

Proof 4 *We prove the algorithm's optimality by contradiction. Let us assume that a node* n *is assigned to a wrong bin (*bin_1*) by the LRM algorithm, i.e., if* n *was placed in another bin (let us say* bin_2*), the value of* $\delta($**sum**$)$ *would increase. This could be true in two cases:* firstly*,* n *was added to* bin_1 *while* bin_2*'s rate was higher, or* secondly*, the rate of* bin_1 *was reduced below* bin_2*'s rate after* n*'s assignment to* bin_1 *by a latter addition of a lower-rate node. The first case is impossible since if* $\rho(bin_2) > \rho(bin_1)$*, the metric* $f(n, 2)$ *is greater than* $f(n, 1)$*, and* bin_1 *can not be chosen. The second case is impossible since each time a bin's rate is reduced, nodes assigned to that bin that can go into multiple bins migrate to the bin that is at the higher rate. Since both conditions are false, the algorithm returns an optimal interface neighbor allocation that maximizes* **sum***.*

External Rate Maximization (ERM)

The objective of the ERM sub-stage is to find, for an interface u_i, neighboring forwarders to whom u_i's rate-limiting-nodes can be 'exported'. The utility of an export can be determined using, in particular, the 'rate-area-product' (RAP) maximization principle described in Chapter 4. The export of rate-limiting-nodes, in general, will increase an interface's transmitting rate, with a node unmarking itself if all its neighbors have been exported and the rate of *all* its interfaces has become ρ_0. Whereas, in the LRM sub-stage, rate-limiting-nodes were exported to another interface on the same node; the export, in ERM, is to an interface on a neighboring marked node. The challenge faced by ERM, due to the potential danger of rate-diversity making links asymmetric[2], is maximizing rates at node's interfaces while preserving *strong* connectivity of the resulting dominating set. Since our framework determines forwarders and rates irrespective of the broadcast source (i.e., until Stage 4), it is important to ensure strong connectivity irrespective of the broadcast source.

To illustrate the concepts employed by ERM, we refer to Figure 10.3 for an example topology consisting of three nodes, each fitted with 3 interfaces. Node u can reach nodes $\{v, w\}$ and $\{x\}$ in a 54 Mbps and 11 Mbps transmission, respectively. However, since u shares channel 1 only with node v and x, only nodes $\{v, x\}$ are reached with the transmission of u_1 (u's interface 1 which is tuned to channel 1); for u to reach w, it must transmit on its interface 2 (tuned to the channel shared with node w). Node w, however, can reach nodes $\{v,u\}$ and $\{x\}$ on a single channel (channel 2) in a 54 and 11 Mbps transmission. We will study ERM sub-stage at node u. Interface u_1 is constrained to use a lower rate (of 11 Mbps) if both neighbors of u_1 (v and x) are to be covered in a single transmission. The rate-limiting-node of u_1 is x. Interface u_1 will look for an interface on a higher-id marked node[3]

[2]For example, it is possible for node u to reach v but not vice-versa (where $\rho(u) < \rho(v)$) due to different ranges for different rates.

[3]The restrictive condition of only exporting to higher-ID neighbors is to avoid circular hand-offs.

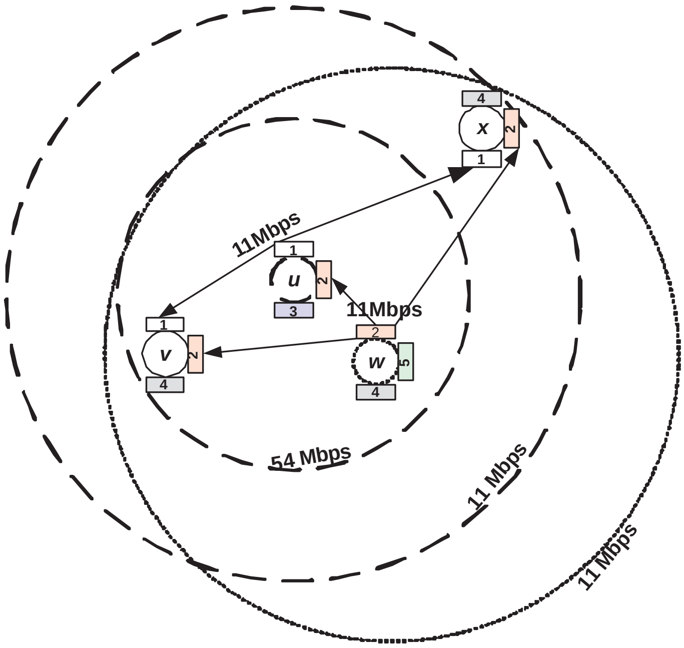


Figure 10.3: Before 'External-Rate-Maximization' at node u

that can cover u_1's rate-limiting-node using its current rate and be reachable from u_1 after u_1 increases its rate; also, the sum of the uplink rates of u_1's neighbors should improve after an export. We check now if u_1's rate-limiting-node x can be exported to w. Firstly, x is reachable through w_1's current transmission; secondly, w is reachable from u even after u_1's rate is increased to 54 Mbps through u_2 (which shares a channel with w_1); lastly, the sum of rates of u's neighbor increases with this transfer (54+11=65 instead of 11+11=22 before). Since all conditions are satisfied, the export of x can take place increasing the rate of u_1 to 54 Mbps as shown in Figure 10.4.

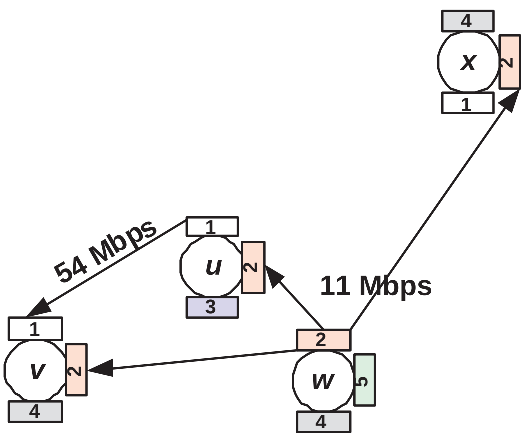


Figure 10.4: After 'External-Rate-Maximization' at node u

The ERM algorithm, for any node u, is mathematically described in Algorithm 14. Node u will attempt to increase the rate of its transmitting interfaces if it is

Algorithm 14 Algorithm for 'External Rate Maximization' at node u

for $i = 1$ to Q **do**
  $continue = 1$
  while $continue$ and $\rho(u_i) \neq 0$ **do**
    $E = 0$; $continue = 0$;
    $r(u_i)$ = rates at u_i sorted in descending order
    k = index of $\rho(u_i)$ in $r(u_i)$
    if $k = 1$ **then**
      $RLN = N_{r_k(u_i)}(u_i) \cup u$
    else
      $RLN = N_{r_k(u_i)}(u_i) \backslash N_{r_{k-1}(u_i)}(u_i)$
    end if
    H= all higher-ID *marked* neighbors of u (on any interface i= 1 to Q) \ $\{RLN\}$
    ——————————————————-
    {This part aims to find a neighbor's interface to export nodes in RLN while satisfying RAP condition}
    ——————————————————-
    for $m = 1$ to $|RLN|$ **do**
      rln= $RLN(m)$; $rate\_new = -\infty$;
      for $n = 1$ to $|H|$ **do**
        $h = H(n)$
        for $q = 1$ to Q **do**
          if $rln \in N(h_q)$ **and** $u \in \cup_{i=1,\ldots,Q} N(h_i)$ **and** $\rho(h_q) > rate\_new$ **then**
            $rate\_new = \rho(h_q)$
          end if
        end for
      end for
      $rate\_diff = rate\_new - r_k(u_i)$
      $E = E + rate\_diff$
    end for
    ——————————————————-
    if $E \geq 0$ **then**
      $continue = 1$; $\rho(u_i) = r_{k-1}(u_i)$
    end if
  end while
end for

currently a transmitting node (i.e. it has some rate-limiting-nodes). The token *continue* is initially equal to 1 which indicates that rate-increase can be attempted; a token *continue* valued 0, on the other hand, implies that the rate-limiting-nodes of the current rate are non-exportable and further rate-increase must not be attempted. Initially, E (denoting the rate gain for the exported nodes) is set to zero. We denote the rates on which an interface u_i has rate-limiting-nodes as $r(u_i)$. The total rates in $r(u_i)$ is not necessarily equal to the total number of rates L and is specific to each u_i. The rates in $r(u_i)$ are arranged in a descending order, i.e., $r_1(u_i) > r_2(u_i)$ and so forth. For mathematical compactness, $r_0(u_i)$ denotes the fact that u_i would not be transmitting since a non-transmitting interface has rate of zero. The index of u_i's current transmission rate, $\rho(u_i)$, in $r(u_i)$ is represented as k in Algorithm 14. The rate-limiting-nodes (RLN) is calculated as the difference between the neighbors of u_i at the current rate ($N_{r_k(u_i)}(u_i)$) and the next higher rate in $r(u_i)$ (i.e., $N_{r_{k-1}(u_i)}(u_i)$, if $r_{k-1}(u_i) \neq r_0(u_i)$). For each node rln in RLN, it is checked for every node $h \in H$ where H comprises of higher-ID marked neighbors of u excluding RLN if, *firstly*, rln is a neighbor of h (i.e., $\pi(h, rln) \geq \rho_q(h)$ for some interface h_q where h_q and rln share a common channel) and, *secondly*, if u is a neighbor of h (to ensure strong-connectivity). The maximum uplink rate rln can receive from a node $h \in H$ fulfilling these conditions is stored in a variable called $rate\_new$ (that is initialized with -∞). The difference between the initial rate of rln and the $rate\_new$ is maintained in $rate\_diff$. The variable E contains the sum of $rate\_diff$ of all nodes in RLN. The nodes that cannot be exported have $rate\_diff$ of -∞. Thus, even for a single non exported rate-limiting-node at a particular rate, the value of E would be -∞. For each interface, if $E > 0$, its rate is increased and *continue* is set to 1; otherwise, if $E < 0$, *continue* is set to zero. The algorithm completes when increase in rate is not possible either due to export of all nodes, or due to *continue* token equal to zero.

Message Complexity: During the ERM sub-stage, each time an interface u_i of a marked node u is successful in increasing its rate, it would broadcast its new rate $\rho(u_i)$ to its neighbors in a $RATE$ message. The maximum number of $RATE$ messages exchanged is $((m-1) \times Q \times L)$ and the size of a $RATE$ packet is the sum of the bytes used to represent node-ID and rate-ID. Since Q and L are constants, total message-complexity of ERM is $O(m)$.

Theorem 5 *The ERM algorithm at a marked node is a polynomial-time algorithm of $O(d^2Q)$*

Proof 5 *The number of 'rounds' in ERM stage is equal to the number of distinct-rate rate-limiting nodes (RLN) on u_i. Since, distinct-rates corresponding to RLN at u_i is represented by $r(u_i)$, and $r(u_i) \leq L$, we conclude that this is a small constant value. An attempt is made, for each rate in $r(u_i)$, to export the RLN nodes for this rate to any interface of a marked-node in H (subject to RAP improvement). As both RLN and H depend on the size of u_i's neighbors, the algorithm is of $O(d^2Q)$ at any marked-node. Also, since ERM is run after receiving a* RATE *packet, total computation complexity of the algorithm in the network is* $(m-1) \times Q \times L \times O(d^2 \times Q)$ $= O(md^2)$.

10.3.4 Stage 3—Tree Construction:

The calculation of the forwarding interfaces (CDS) and their rates, till Stage 3 of our framework, is performed independent of the broadcast source. During this stage, tree relationship spanning all nodes is built and many redundant transmissions (retained during earlier stages) are eliminated. The decisions in this stage is restricted to the choice of the interfaces amongst the 'candidate' forwarding interfaces chosen earlier (along with their rates). The explicit aim of Stage 3 is to calculate a high performance tree that minimizes broadcast latency. Topology construction algorithms, for this objective, must incorporate WBA and rate-diversity since the use of WBA reduces the number of transmissions to mitigate the adverse effects of interference in a wireless network, while the usage of rate-diversity can improve broadcast performance by employing higher rate links.

Algorithm 15 Algorithm for distributed 'Tree construction' for MR^2-MC WMNs

Let $label$ for all nodes in $V = \infty$ and $u = id(node)$
if u is the broadcast source s **then**
  for $i = 1$ to Q **and** $\rho(s_i) \neq 0$ **do**
    $RREQ.source = s$;
    $RREQ.sender = s$; $RREQ.interface = i$
    $RREQ.label = 1/\rho(s_i) \times 1/N(s_i)$
    $RREQ.neighbors = N(s_i)$
    $send(RREQ)$ on interface i
  end for
end if
if non-duplicate $RREQ$ received on an interface $\widehat{i}$ **then**
  Let $\tilde{RREQ}$ = the received $RREQ$
  if $\tilde{RREQ}.label < label(u)$ **then**
    $P(u) = \tilde{RREQ}.sender$
    $PI(u) = \tilde{RREQ}.interface$
    $S(u) = \tilde{RREQ}.neighbors$
    if $RREPACK$ from $RREQ.sender$ not received yet **then**
      $RREP.nexthop = \tilde{RREQ}.sender$
      $send$ $(RREP)$ on interface $\widehat{i}$ to $\tilde{RREQ}.sender$
    end if
  end if
  if u is a marked-node that has not forwarded $RREQ$ for $\tilde{RREQ}.source$ before **then**
    for $i = 1$ to Q **and** $\rho(u_i) \neq 0$ **do**
      $RREQ.sender = u$;
      $RREQ.interface = i$; $RREQ.neighbors = N(u_i)$
      $\hat{l}(u_i) = 1/\rho(u_i) \times 1/(N(u_i) \backslash S(u))$
      $RREQ.label = \tilde{RREQ}.label + \hat{l}(u_i)$
      $send$ $(RREQ)$ on interface i
    end for
  end if
  if received $RREP$ and $RREP.nexthop = u$ **then**
    Activate $Forwarder$ flag for u
    $RREP.nexthop = P(u)$
    $send$ $(RREP)$ on interface $PI(u)$
  end if
end if

The tree construction (Stage 3) is mathematically described in Algorithm 15.

Initially, the *label* of all nodes is equal to ∞. The source node s initially broadcasts a $RREQ$ message on each of its (transmitting) interface after setting $RREQ.source$ and $RREQ.sender$, $RREQ.interface$ and $RREQ.neighbors$ to s, the interface's id, and the neighbor set of s on that interface, respectively. The $RREQ.label$ for an interface s_i is set to its 'weighted latency', $\hat{l}(s_i)$, which is calculated as $1/\rho(s_i) \times 1/N(s_i)$. The weighted-latency metric $\hat{l}$ is based on the RAP concept presented in Chapter 4 which states that the efficiency of a rate (for broadcast latency) can be reasonably predicted from the product of the rate and its transmission range. As a special case of the RAP principle, the efficiency of a rate is proportional to the product of that rate's transmission latency and the number of receivers (that have not received previously) in this rate's transmission area. *Firstly*, any node u that receives a non-duplicate $R\tilde{R}EQ$ message on its interface $\widehat{i}$ will determine if $R\tilde{R}EQ.label$ is less than $label(u)$; if so, u will choose the sender of the $R\tilde{R}EQ$ as its parent ($P(u)$) and $\widehat{i}$ as the interface to connect to its parent ($PI(u)$). The neighbors of $P(u)$ (contained in $R\tilde{R}EQ.neighbors$) are referred to as sibling nodes of u and are denoted by $S(u)$. Node u, if it knows that its parent $R\tilde{R}EQ.sender$ is currently not a forwarder (we shall see later how), will send a $RREP$ to it on interface $\widehat{i}$ after setting $RREP.nexthop$ to $R\tilde{R}EQ.sender$. *Secondly*, if node u receives a $RREP$ message with the $RREP.nexthop$ set to u's ID, it will activate its $Forwarder$ flag and broadcast a $RREPACK$ message to announce to its neighbors that it is a forwarding node as well as send a $RREP$ back to its parent. After a node u has broadcasted a $RREPACK$, u's neighbors would not send a $RREP$ to it knowing that u is already a forwarder. *Thirdly*, a marked-node u will forward a $RREQ$ message for a particular broadcast source $RREQ.source$ only once. Note that only marked-nodes can forward $RREQ$s. If after forwarding a $RREQ$ once, a $RREQ$ with a better *label* (than the node's current *label*) is received, the node will modify its P, PI and S; the node will, however, not rebroadcast $RREQ$ another time after broadcasting $RREQ$ for a particular $RREQ.source$ before[4]. *Lastly*, when marked-node u does broadcast a $RREQ$, it will generate a $RREQ$ message for each of its interface with u's ID in $RREQ.sender$ and $RREQ.label$ modified to the sum of $label(u)$ and the weighted latency $\hat{l}$ of the interface. Similarly, $RREQ.interface$ and $RREQ.neighbors$ is set to i and $N(u_i)$, respectively, for the case when interface u_i is used. The weighted-latency $\hat{l}$ for interface u_i is calculated as the product of u_i's rate and the new receivers on u_i with the new receivers on u_i approximated by subtracting $S(u)$ from $N(u_i)$. The tree construction is complete when each marked-node has relayed the RREQ once. The nodes that have the Forwarder flag activated will forward the broadcast data at pre-determined rates (decided prior to this stage).

Theorem 6 *The message complexity of our tree construction algorithm is $O(mQ)$.*

[4]This enables our algorithm to minimize its message complexity.

Protocol	Type	Radio-and-channel-diversity and	Rate-diversity
MRDT	distributed	✓	✓
PAMT (Chapter 9)	centralized	✓	✓
MDW (Chapter 5)	distributed	×	✓
ODMRP [37]	distributed	×	×

Table 10.3: Comparison of different broadcast protocols

Proof 6 *The maximum number of RREQ messages sent in the network can be* $(m+1)Q$ *if the marked nodes and the source node send RREQ on all interfaces. Similarly, the maximum number of* $RREPACK$ *and* $RREP$ *packets that can be exchanged is* $m + 1(Q)$ *and* m*, respectively. Therefore, the worst-case message-complexity of the tree construction algorithm is* $O(2(m+1)Q + mQ) = O(mQ)$.

10.4 Simulation results using Qualnet simulator

In this section, we will present performance evaluation results for our algorithms using the Qualnet [1] simulator with a decentralized MAC scheduler (we have used 802.11 as our MAC scheduler). We implemented PHY 802.11a and 802.11b at the physical layer, which uses a pre-configured BER-based packet reception model. The IEEE 802.11 MAC with Distributed Coordination Function (DCF) was chosen as the medium access control protocol. All default parameters are assumed unless stated otherwise. The simulations are performed using random topologies assuming a network of area 1×1 km^2.

10.4.1 Analysis of MRDT's performance

To evaluate the performance of our distributed framework MRDT, we compare it against the performance the centralized PAMT (Chapter 9), the distributed MDW (Chapter 5), and the distributed ODMRP [37] algorithms. The characteristics of these protocols are tabulated in Table 10.3. We use three *static* channel assignment strategies in our experiment. *Firstly*, the 'common channel approach' *(CCA)* ([14]), in which all nodes are assigned a common set of channels. *Secondly*, 'varying channel approach' *(VCA)*, interfaces of different nodes may be assigned to a different set of channels ([73]). *Thirdly*, we have used the 'interference survivable topology control' *(INSTC)* [74] in which channels are assigned to reduce the interference in the induced network topology. Unless stated otherwise, CCA scheme with Q and $C = 3$ should be assumed for our results. For the presented results, an interval estimate with the confidence interval of 90 is used. The ticks in the graphs represent 5th and 95th percentiles over 100 uniformly distributed random topologies. We will now proceed to discuss the results in the next few subsections.

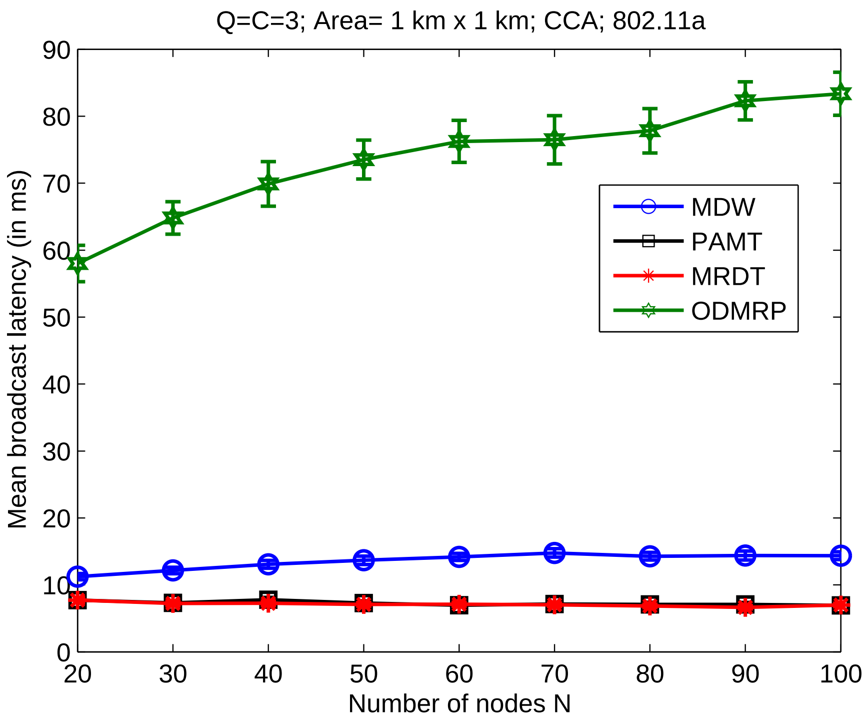


Figure 10.5: Broadcast latency for varying number of nodes (N) in a 802.11a network

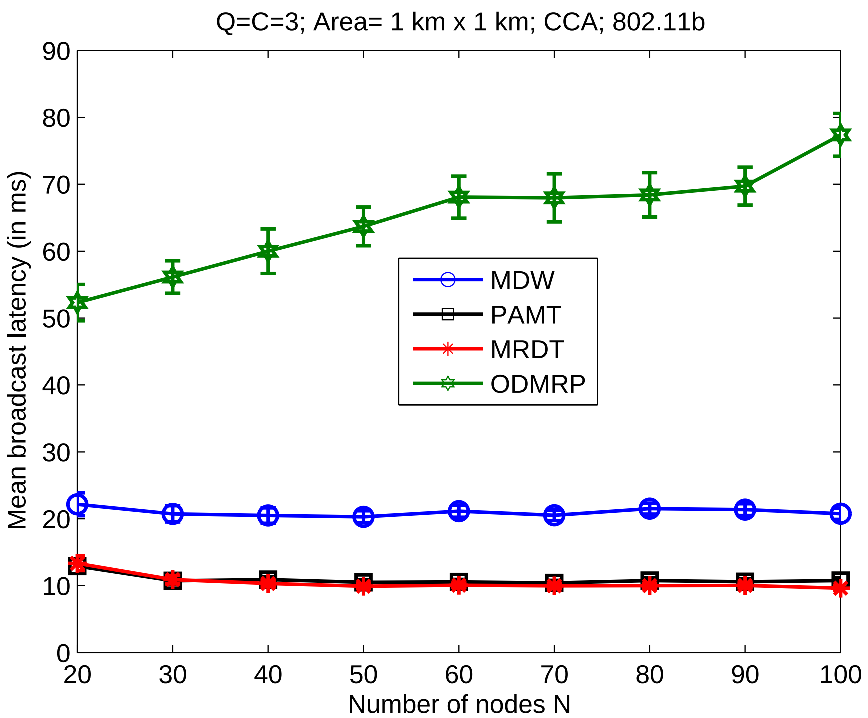


Figure 10.6: Broadcast latency for varying number of nodes (N) in a 802.11b network

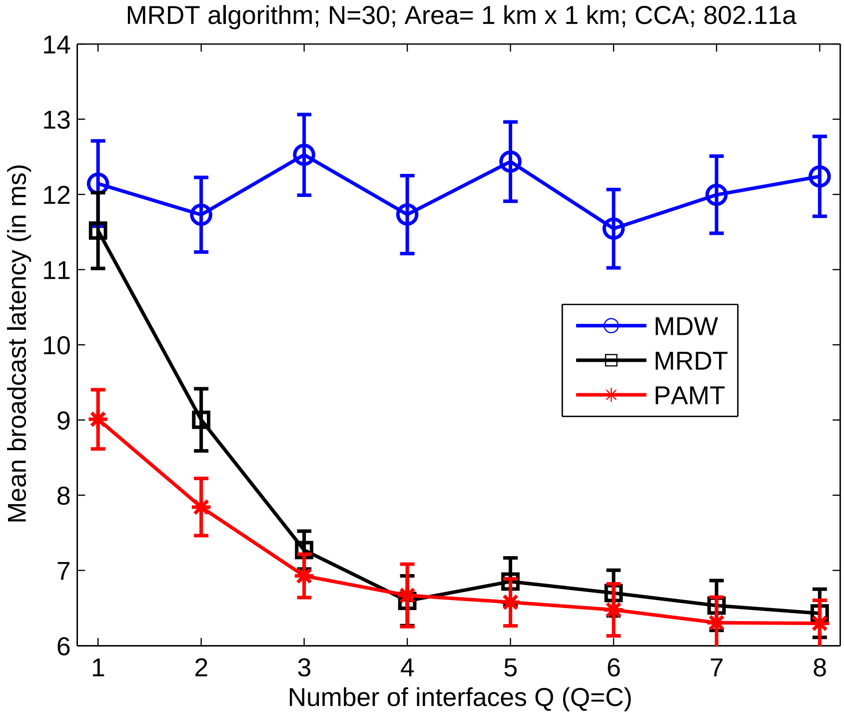


Figure 10.7: Broadcast latency for varying number of radio interfaces (Q) in a 802.11a network

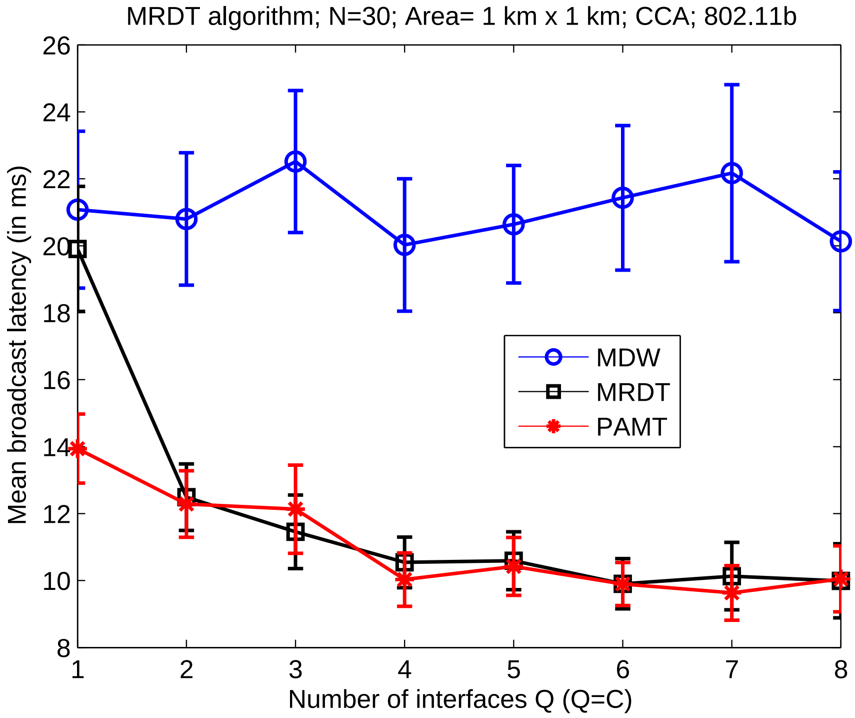


Figure 10.8: Broadcast latency for varying number of radio interfaces (Q) in a 802.11b network

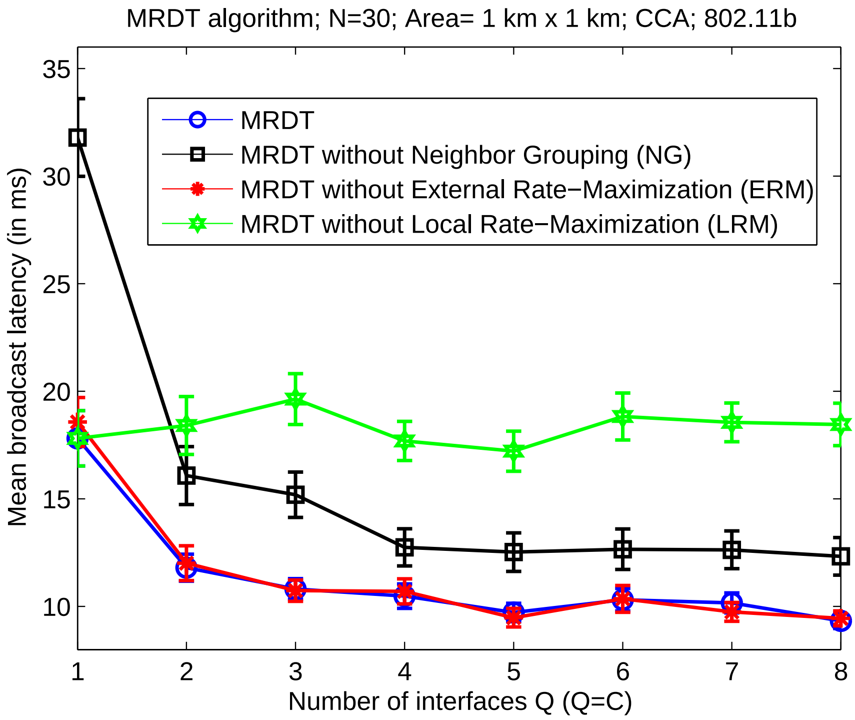


Figure 10.9: The effects of MRDT's different stages on broadcast latency performance

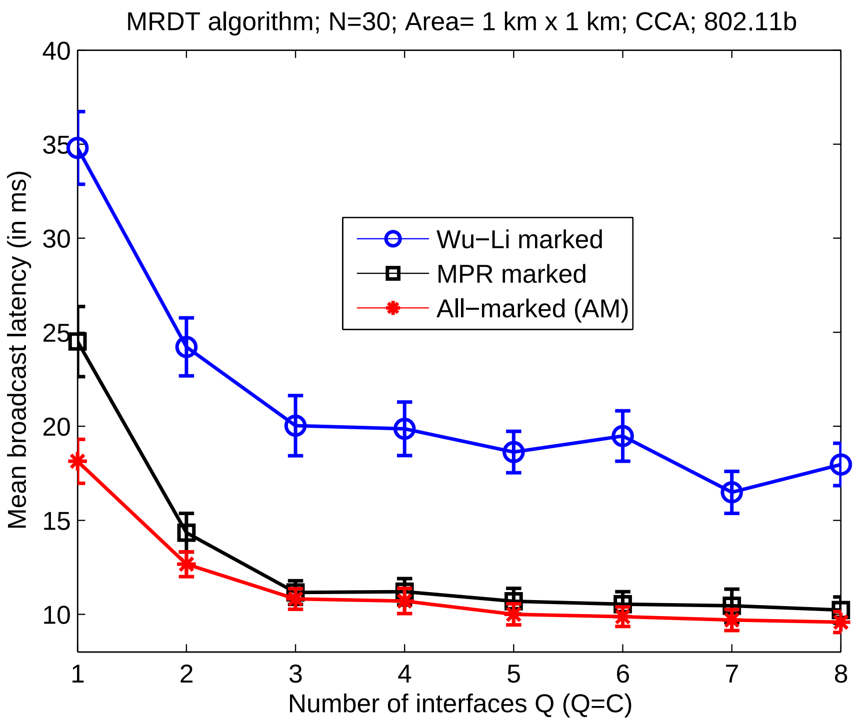


Figure 10.10: The effects of different Stage 0 marking techniques on broadcast latency performance

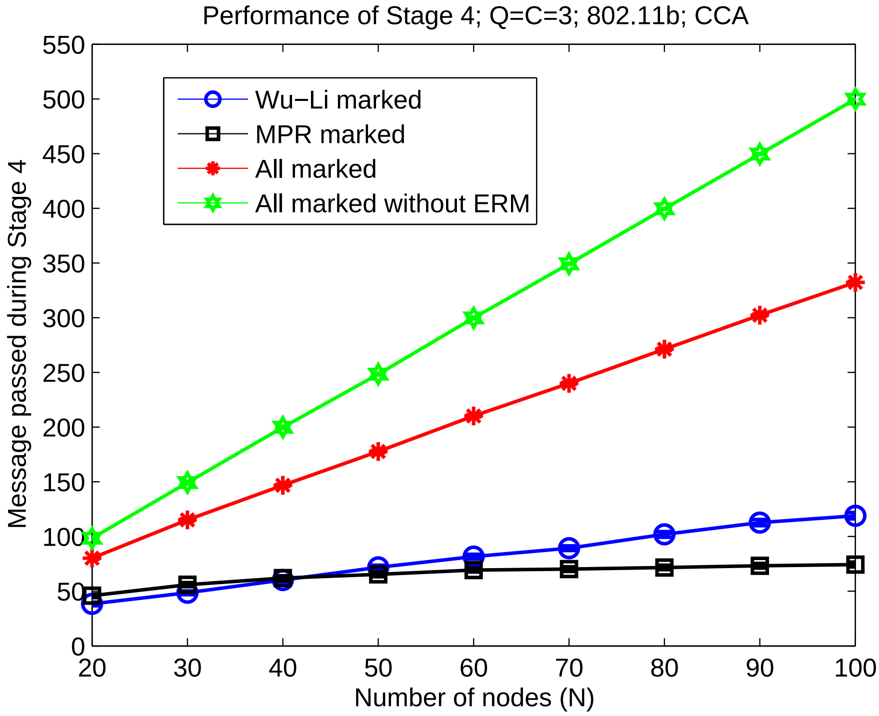


Figure 10.11: Messages exchanged (in Stage 3) for Stage 0 techniques for varying N

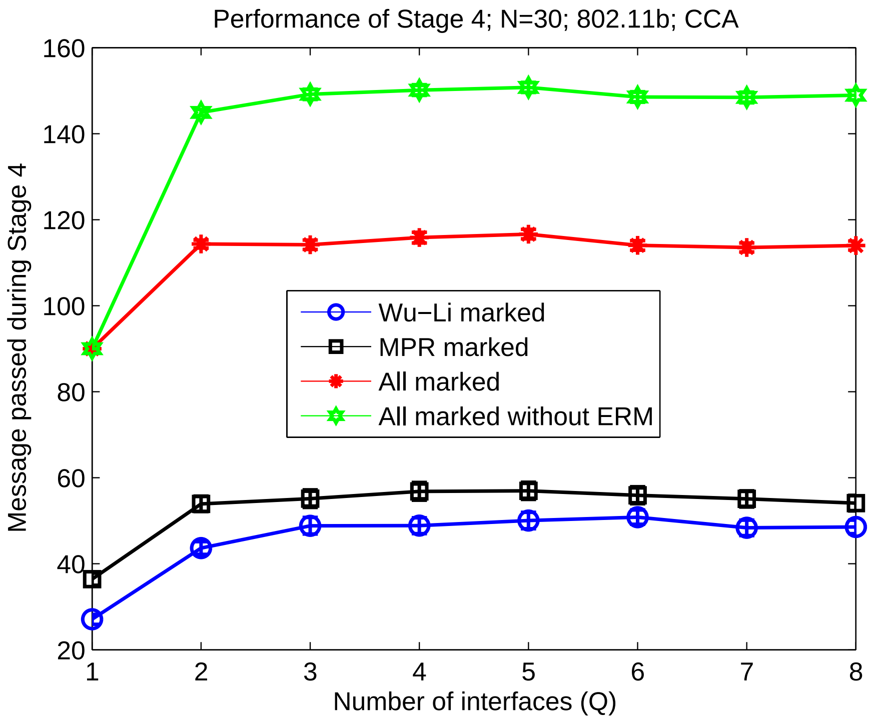


Figure 10.12: Messages exchanged (in Stage 3) for Stage 0 techniques for varying Q

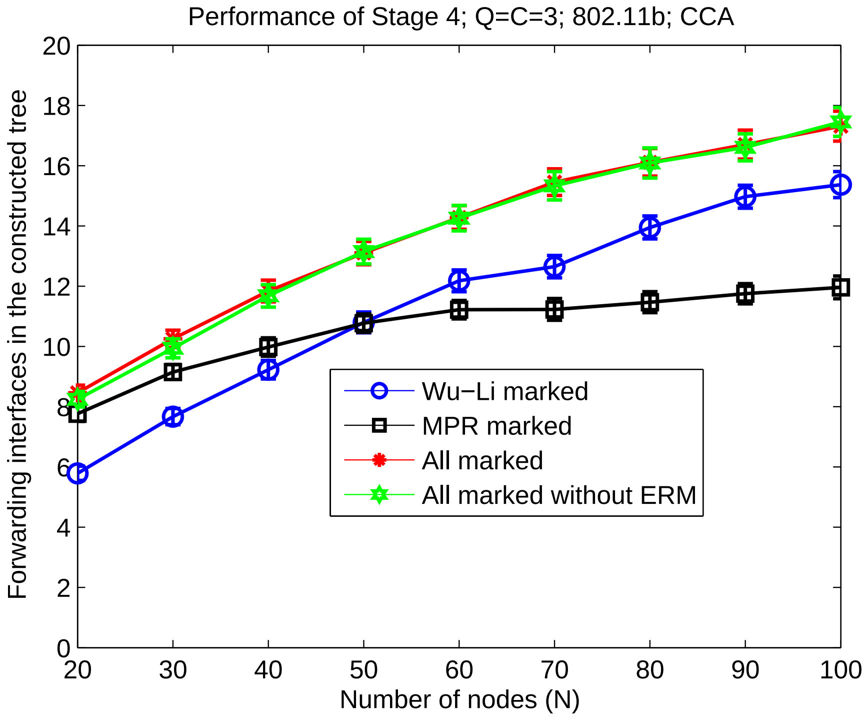


Figure 10.13: Forwarding interfaces in tree constructed by Stage 3

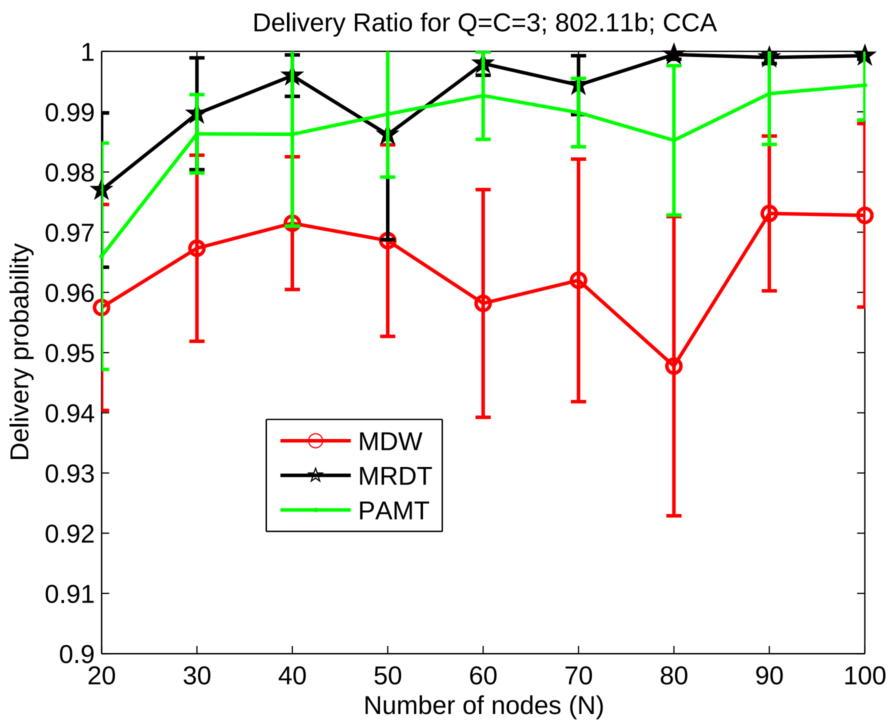


Figure 10.14: Probability of successful delivery to all nodes

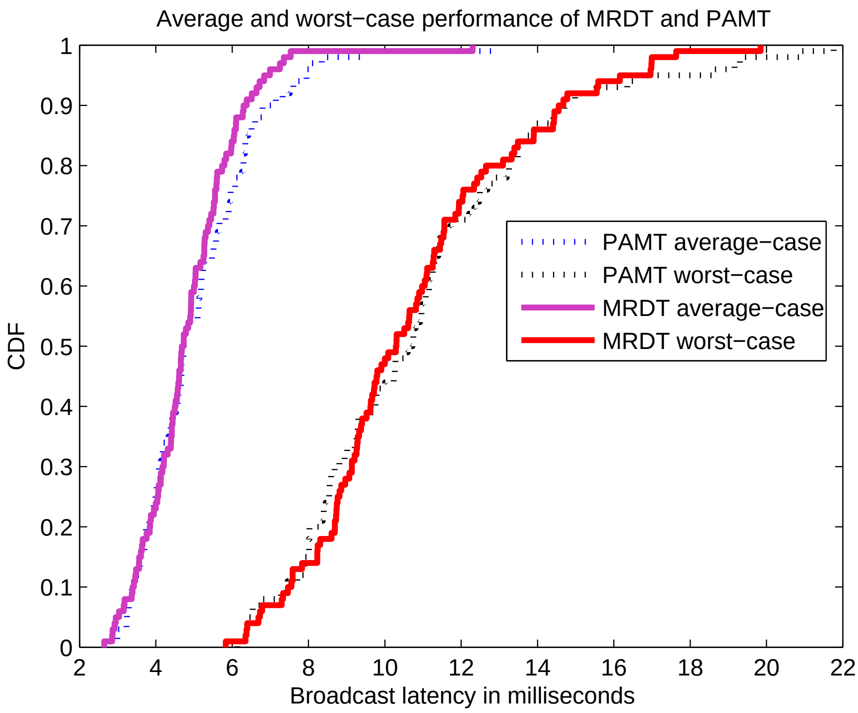


Figure 10.15: Forwarding interfaces in tree constructed by Stage 3

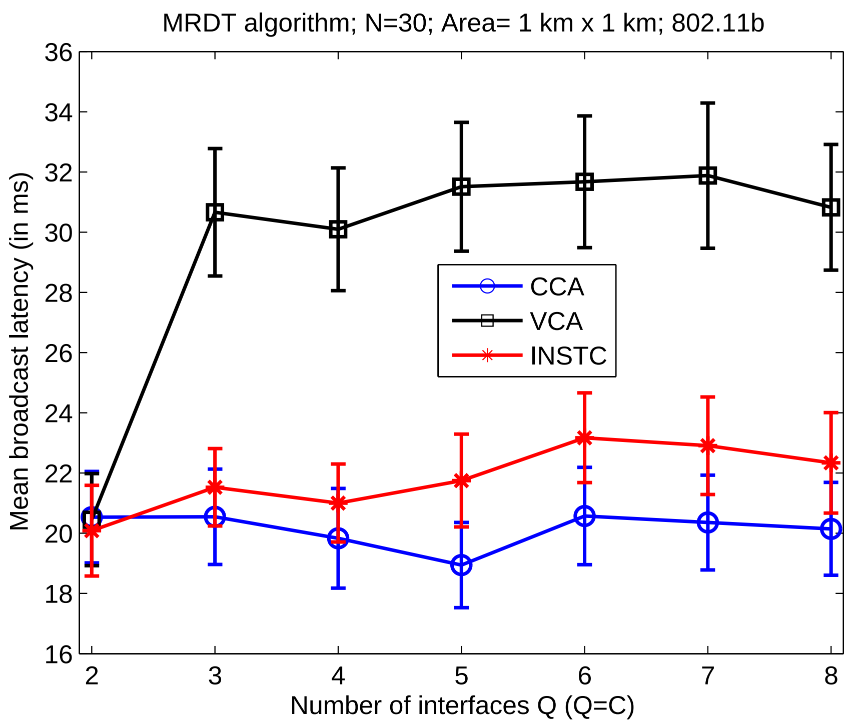


Figure 10.16: CDF of average and worst-case performance

10.4.2 Comparative performance and variation with node density

The effect of network's node density on MRDT's performance can be seen in Figures 10.5 and 10.6 for 802.11*a* and 802.11*b* networks, respectively. It is observed that distributed MRDT algorithm performs as well as the centralized PAMT algorithm as seen in Figures 10.5 and 10.6 (where the results of MRDT sit on top of PAMT's results for the range of N considered). It is also observed that MRDT improves the performance of MDW (another distributed multi-rate algorithm) by incorporating radio-and-channel-diversity in its calculations. The ODMRP algorithm is included in our results as an example protocol that does not utilize the rate-diversity or the radio-and-channel-diversity. The performance gain of MRDT over ODMRP and MDW algorithms is ~ 10 times and ~ 2 times, respectively (Figure 10.5).

10.4.3 The effect of number of radio interfaces

The effect of varying number of radio interfaces Q on MRDT's performance can be seen in Figures 10.7 and 10.8 for 802.11*a* and 802.11*b* networks, respectively. It is observed that, for values of Q as low as 2 or 3, MRDT matches the performance of PAMT algorithm. This improvement is possible since MRDT exploits the radio-and-channel-diversity by employing parallel transmissions on non-interfering interfaces. The difference in the performance of MRDT and MDW in Figure 10.8 demonstrates that MRDT, like PAMT and unlike MDW, is an *adaptive* algorithm that adapts to the number of radio resources available. The performance gap, between MRDT and MDW, widens as the number of radio interfaces is increased clearly demonstrating the benefit obtained by exploiting radio-and-channel-diversity.

10.4.4 Evaluation of different stages of MRDT

Among the different stages of MRDT, the LRM sub-stage offers the most improvement in broadcast-latency performance across the range of Q excepting the case when $Q = 1$ (since LRM clearly cannot parallelize transmissions when no alternative interface is available). Without the LRM substage, MRDT does not remain adaptive to the available radio resources and cannot exploit the radio-and-channel-diversity. The improvement due to LRM is clearly evident in Figure 10.9. The Neighbor-Grouping also improves the performance markedly across the range of Q and especially for $Q = 1$, as shown in Figure 10.9. The use of ERM substage (of Stage 2), however, has limited benefits vis-a-vis broadcast latency reduction; its main impact, as we shall see later, is in reducing overhead during the stage of Tree-Construction (Stage 3). The different marking schemes (of Stage 0) also affect the broadcast latency performance as shown in Figure 10.10 which shows that marking by MPR, or marking all nodes in Stage 1 gives better results than marking by the

Wu-Li method.

10.4.5 The overhead of Tree Construction (Stage 3)

The message overhead during Tree Construction (Stage 3) depends upon the marking scheme used in Stage 0. Recall that the message-complexity of Stage 3 is $O(m)$; therefore, the message overhead for marking schemes that return a smaller set of marked nodes m is lower. We show the total number of messages exchanged during Tree Construction (Stage 3) in Figures 10.11 and 10.12 for varying number of N and Q. We see that fewer messages need to be exchanged when Wu-Li or MPR marking technique is used in Stage 0 as compared to when all nodes are marked. The benefit in overhead reduction due to ERM is evident in both Figures 10.11 and 10.12. We note that the overhead using all-node-marking scheme (especially without ERM) is particularly large. However, after the tree has been constructed by Stage 3, the forwarding interfaces are nearly the same regardless of the marking used in Stage 0 as shown in Figure 10.13.

10.4.6 Average latency vs. the 'broadcast latency'

We have compared the average-case and worst-case latency performance for MRDT and PAMT algorithms. This comparison is depicted in Figure 10.15 which displays the probability of nodes receiving the packet in given broadcast latency (in milliseconds). We see that the average-latency performance can also benefit from the increased parallelization of MRDT, as MRDT's performance is marginally better than the performance of PAMT.

10.4.7 Successful delivery probability results

The probability of successful delivery for our evaluated algorithms is shown in Figure 10.14. We see that MRDT's performance is marginally better than the other algorithms. We note here that the delivery probability shown is for a single packet, i.e. for a single broadcasted packet for 100 topologies of network with N nodes, it shows the probability that all N nodes in the network receive this packet.

10.4.8 The effect of channel-assignment

The performance of MRDT with different channel-assignment schemes is shown in Figure 10.16. We note that MRDT can benefit from increased 'connectivity' present in schemes like CCA which presents more opportunities of exploiting WBA. These results, however, are not specific to MRDT as PAMT presented similar features when different channels-assignment schemes were used (Chapter 9).

10.4.9 The effect of transmission rate-range curve

For sensitivity analysis, we have performed our experiments in Qualnet for both 802.11*a* and 802.11*b* networks, both of which have different rate-range characteristics. Our results, Figures 10.5, 10.6, 10.7 and 10.8, seem to indicate that MRDT is relatively insensitive to the rate-range curve as it performs well in both 802.11*a* and 802.1*b* networks.

10.5 Conclusions

In this chapter, we have proposed a four-stage *completely distributed and localized* broadcast heuristic that exploits the 'wireless broadcast advantage' the rate-diversity, and the radio-and-channel-diversity of a multi-radio multi-rate multi-channel (MR^2-MC) WMN. These algorithms all operate by initially constructing a basic source-independent CDS and pruning it to increase the rate of individual transmissions, and subsequently by constructing a source-specific tree over this 'pruned CDS".

Simulation-based experimental studies show that the MRDT decentralized algorithm can provide low broadcast latencies close to that of the best-performing, centralized PAMT heuristic, especially for practical environments where the number of interfaces (Q) is relatively low ($2-3$). Moreover, the studies show that the performance gains of MRDT are relatively insensitive to the choice of the initial CDS computation (Wu-Li vs. MPR), but benefit significantly from the 'rate maximization' stages (LRM and ERM) which exploit the radio-and-channel-diversity of WMN nodes and the redundancy of wireless broadcast. Moreover, the distributed nature of MRDT implies that the signaling overhead remains fairly low, especially when initial CDS is computed through Wu-Li or MPR, even as the size of the WMN increases.

Chapter 11

Alternative MLB Framework for MR^2-MC WMNs

11.1 Introduction

In Chapters 4, 5, 9 and 10, we have assumed a "*fully multi-rate multicast*" (FMM) framework in which nodes can adjust link-layer multicast transmission rate for each link-layer frame. In this chapter, a new framework called SBM, that was described earlier in Chapter 6, is utilized. The SBM framework exploits link-layer rate-diversity by enabling each WMN to decide, depending on its topological properties, a single transmission rate for all its link-layer data multicasts. Although, FMM framework returns impressive performance, employing SBM is attractive (as explained in Chapter 6) since it can eliminate some undesirable features of practical multi-rate Media Access Control (MAC) protocols. In this chapter, we propose a heuristic SBM solution that can realize low-latency broadcast by exploiting a multi-radio multi-channel multi-rate (MR^2-MC) WMNs' rate and radio-and-channel-diversity. We also propose methods to determine the "best" link-layer transmission rate for the SBM framework. Simulation results indicate that SBM broadcast heuristics perform comparably to FMM broadcast heuristics especially for dense networks.

While using multi-rate broadcast is desirable in an ideal case, it has been observed in practical MACs (such as 802.11) that the choice of a low transmission rate, even by an individual node, may substantially lower the total throughput achieved in that region (*due to the well-known paradigm of fairness in access attempts rather than bandwidth*) [82] [83]. Hence, it is worth studying the impact of broadcasting, in an ideal setting, using a single *'best'* rate, as opposed to the more powerful paradigm of broadcast transmission by different nodes at different rates. In particular, if it turns out that a single-rate broadcast strategy can provide latencies fairly close to those provided by the multi-rate case, then an approach based on adopting a single system-wide link-layer broadcast rate may become worthy of consideration. Using the SBM framework can simplify the broadcasting algorithms—e.g., the *'multicast grouping'* stage in broadcasting heuristics of Chapter 9 and 10, that cater to the possibility of a transmitting node covering its neighbors in multiple transmissions (at different rates), is eliminated when SBM framework is used.

The main contributions of this chapter are detailed below:

1. *We present a heuristic SBM broadcast solution to the MLB problem in MR^2-MC WMNs.*

2. *We demonstrate that for MR^2-MC WMNs (like for SR-SC multi-rate WMNs), the transmission rates having higher RAP values are more efficient for broadcast as a general rule-of-thumb.*

3. *We propose two techniques to calculate the 'best' link-layer transmission rate to use, given a MR^2-MC WMN that supports n different rates, for the SBM framework'. We show that the highest RAP-valued transmission rate, which ensures a connected network when that rate is exclusively used, is a good choice for use as the 'best' rate for the SBM framework when the CCA channel-assignment is used.*

Chapter Outline: The rest of the chapter is organized as follows. We detail our network and interference model in Section 11.2. Our MLB broadcasting solution for the SBM framework in MR^2-MC WMNs, composed of the "*calculation of 'best' rate*", "*topology construction*", and "*interface grouping and transmission scheduling*" stages, is discussed briefly in Section 11.3. The "calculation of 'best' rate" stage is discussed in Section 11.4 where we propose two techniques to calculate the 'best' rate. We present the "topology construction" stage in detail in Section 11.5, and also present a heuristic algorithm called PCDS for this stage. We discuss the "interface grouping and transmission scheduling" stages in detail in Section 11.6. The performance evaluation of our heuristic algorithm is provided for an idealized scheduler and a practical decentralized 802.11 MAC based scheduler in Sections 11.7 and 11.8, respectively. We will conclude this chapter in Section 11.9.

11.2 Network and interference model

We assume the SBM broadcasting framework in which each WMN decides, based on its topological properties, a single link-layer rate to use for all broadcasts. Our network and interference model is similar to that described in Chapters 9 and 10 (except that we use the SBM framework). We will reproduce the relevant details here for completeness. We use a network model similar to that described by Raniwala et al. [73]. We will use the notation introduced by Tang et al. [74] to represent our channel-assignment. Each node in the network can transmit at multiple-rates. We assume the SBM broadcasting framework in which each WMN decides, based on its topological properties, a single link-layer rate to use for all broadcasts. There are totally C non-overlapping orthogonal frequency channels in the system and each node is equipped with Q radio interfaces where $Q \leq C$. The Q radio interfaces have omni-directional antennas. In order to efficiently utilize the network resources, two radio interfaces, at the same node, are not tuned to the same channel. Using the Qualnet simulator [1] as a reference (assuming a two-ray propagation model), we obtain the transmission rate versus transmission range (*rate-range*) relationship (for 802.11b) shown in the first two columns of Table 11.1. We also employ an alterative rate-range relationship, shown in the first two columns of Table 11.2, of a commercial IEEE 802.11a/b/g product [2] to perform sensitivity analysis of the broadcast performance with different rate-range relationships.

We use an undirected graph $G_T = (V, E_T, L_T)$ to model the given mesh network topology *before channel assignment*, where V is the set of vertices, E_T is the set of edges and L_T is the set of weights of edges in E_T. The vertex v in V corresponds to a wireless node in the network with a known location. An undirected edge (u, v), corresponding to a wireless link between u and v, is in the set E_T *if and only if* $d(u, v) \leq r$ where $d(u, v)$ is the Euclidean distance between u and v and r is the range of the *lowest-rate* transmission. The latency of a link $l(u, v)$ is the latency of the '*quickest*' transmission rate that can be supported between nodes u and v. The set L_T contains the latencies of all links in E.

Channel assignment: A channel assignment $\mathcal{A}$ assigns each vertex v in V, Q different channels denoted by the set: $A(v) = \{a_1(v), a_2(v), \ldots, a_Q(v) : a_i(v) \neq a_j(v), \forall i \neq j;\ a_i(v) \in C, \forall i\}$ where $a_i(v)$ represents the channel assigned to i^{th} radio interface at node v. The topology defined by $\mathcal{A}$ is represented by $G = (V, E, L, \Lambda)$ in the following natural way: There is an edge $e = (u, v, k)$ on channel $\lambda(e) = k$ between nodes u and v in G *if and only if* $d(u, v) \leq r$ (i.e. edge$(u, v) \in E_T$) and $\lambda(e) \in \mathcal{A}(u) \bigcap \mathcal{A}(v)$. The latency of the edge e is the latency of the fastest transmission rate supported on e. The set L contains the latency of each edge in E; similarly the set Λ contains the channel used on each edge in E. Note that G may be a *multi-graph*, with multiple edges between the same pair of nodes, when the node pair shares two or more channels. We use the same notation to refer to *vertices and nodes*, to *edges and links*, and to *weight of edges and latency of links* without confusion, the usage being clear from the context. For the work in this chapter, we have only used the '*common channel approach*' (CCA) channel-assignment scheme [14] [28] (Section 2.4).

Interference Model: We use a generalized conflict graph *based on transmissions* to model the effects of wireless interference between different multicast transmissions in MR^2-MC meshes. The conflict graph indicates which transmissions mutually interfere and hence cannot be active simultaneously. A transmission b_i interferes with a transmission b_j, if both transmissions b_i and b_j are taking place on the *same* channel, *and* the receivers of the transmission b_i are within the interference range of the transmitting node of b_j or *vice-versa*. The transmissions b_i and b_j do not interfere otherwise.

11.3 Heuristic solution to MLB problem

In this section, we present heuristic algorithms that solve MLB problem, using the SBM framework, in MR^2-MC WMNs. Broadly speaking, such heuristic algorithms must take *four* important decisions at each node. *Firstly*, it has to determine the 'best' transmission rate to use for *all* link-layer broadcasts (*this stems from our design choice to have only one broadcast rate*). *Secondly*, it has to decide whether a node should transmit (i.e., be a non-leaf node in the broadcast tree) or not. *Thirdly*, the 'interface grouping' decision must be made to decide the interface (*or alternatively the channel, since each interface is tuned to a distinct channel*) a transmitting node will use for its transmission. *Lastly*, each node's transmissions must be scheduled, while ensuring simultaneous transmissions (at different nodes) do not interfere, to minimize the broadcast delay.

The MLB problem for MR^2-MC WMNs, as discussed in Chapter 9, is at least NP-hard and is composed of many inter-related hard subproblems. With the complexity of the problem in mind, we have decomposed our solution into *three* logically independent stages:

1) ***The 'best' link-layer multicast rate selection***: Since we have adopted the design choice of using the SBM framework so that the implementation of broadcast heuristics can be simplified, we need to determine the single 'best' link-layer multicast rate. This rate, to be used for all link-layer multicast, is determined by each WMN, during the first stage of our solution, according to its topological properties.

2) ***Topology Construction***: The aim of this stage is to compute a broadcast tree (or a spanning tree of the given topology) T that exploits the WBA, the multi-rate transmission capability and the plurality of radio interfaces and channels available. The transmitting nodes and the children/parent relationships between different nodes are all decided during this stage.

3) ***Interface Grouping and Transmission Scheduling***: While the non-leaf nodes (transmitting nodes) of the tree are determined during the 'topology construction' stage, *the interface used for transmission at these non-leaf nodes is only decided during the interface grouping substage of the joint 'interface grouping and transmission scheduling' stage.* The interface grouping (or simply, grouping) substage must ensure that the interface chosen, at any transmitting node, shares its channel with the children node(s) of the transmitting node. *The transmission scheduling* (or simply, scheduling) *substage, on the other hand, determines the exact timing of the various transmissions.* The scheduling of the transmissions is done

according to the following constraints: firstly, a node must transmit only after receiving its parent's transmission and secondly, the interfering transmissions must not be scheduled simultaneously. By having a joint '*grouping and scheduling*' stage, the radio-and-channel-diversity of the network can be utilized more efficiently.

11.4 Determining the single 'best' link-layer multicast rate and "RAP" formulation

We point out a key finding of Chapter 4 that a transmission rate's broadcast efficiency in reducing broadcast latency, for SR-SC multi-rate WMN, can be predicted reasonably by the product of the transmission rate and its transmission coverage area (*rate-area product* or *RAP*). We propose using a similar approach for predicting a particular transmission rate's broadcast efficiency in MR^2-MC WMN. The RAP values for different transmission rates of the rate-range relationship of 802.11b in Qualnet [1] are provided in Table 11.1. Similarly, the RAP values for transmission rates of our alternative rate-range relationship, as obtained from a 802.11a commercial product [2], are provided in Table 11.2. As a general rule-of-thumb, a transmission rate that has a higher RAP is more broadcast-efficient for SR-SC multi-rate WMNs (Chapter 6). We investigate if this conjecture still holds for MR^2-MC WMNs.

We propose two methods of determining the 'best' link-layer multicast rate. For any given WMN, let R denote the set of transmission rates, which if used as the link-layer rate for all multicast, returns a connected network. *In the first method, we use the highest link-layer multicast rate in R as the chosen 'best' rate.* We call the transmission rate calculated by this method as the "*quickest*" rate. *In the second method, we use the transmission rate in R that has the highest RAP value of all rates in R.* We call the transmission rate calculated by this method as the "*HRC*" (*highest-RAP-valued connected*) rate.

We will present results and analysis of these methods in Section 11.7.

11.5 Topology Construction in MR^2-MC WMN using a single link-layer broadcast rate:

In this section, we will present a heuristic algorithm, called Parallelized Connected Dominating Set (PCDS), for the 'topology construction' stage which exploits both the WBA and the interface (or, the radio-and-channel) diversity offered by MR^2-MC WMNs. The input to our algorithm is the channel assignment defined input topology $G = (V, E, L, \Lambda)$, broadcast source s in V, the 'best' broadcast transmission rate $\hat{l}$ (chosen as described in the previous section), and the channel assignments to all interfaces at each node $\mathcal{A}$.

We have observed in Chapter 9 that the best performing trees in MR^2-MC WMNs, generally, adapt to the radio resources (i.e., radio interfaces (Q) and channels (C)) available. It is desirable to construct similar adaptive trees when we use the SBM framework. The PCDS algorithm (depicted in Algorithm 16) is adapted from the PAMT algorithm (Chapter 9) that assumed a FMM framework. Like the PAMT algorithm that improves WCDS algorithm (Chapter 4) for FMM broadcast in SR-SC multi-rate WMNs, the PCDS algorithm improves the SCDS algorithm (Chapter 6) for SBM broadcast by parallelizing transmissions through exploitation of MR^2-MC WMNs' radio-and-channel-diversity.

The PCDS algorithm differs from the SCDS algorithm in the calculation of each transmission's priority f. In an attempt to modify SCDS to contain more parallel transmissions (when multiple radio interfaces are available), PCDS does not include downstream neighbors in a transmission that can be reached by *any other* eligible node, *on an alternative channel*, with a better path. The PCDS algorithm uses an extra parameter called *label* for each node, found using Dijkstra's algorithm, to represent the distance of this node to the source node (alternatively, instead of using *label*, the depth of the node can be used, since all nodes use the same rate).

PCDS begins by adding node s to the set $\mathcal{R}$ which contains nodes that are eligible to transmit during the next-round. The set $Y_{(n,\hat{l},c)} = N(n,\hat{l},c)\backslash\mathcal{R}$ contains all hitherto 'uncovered nodes' that can be covered by a transmission $(n,\hat{l},c)$. The label of this transmission ($label_{trans}$) is *label(n) + l*. During the calculation of priority of each transmission $(n,\hat{l},c)$, all the yet not-reached neighboring nodes $Y_{(n,\hat{l},c)}$ of transmitting node n are contained in X. We search the neighborhood of each node in X to find if there are other eligible (to transmit) nodes in its vicinity that can cover this node on a channel different to that used by n. Such nodes are referred to as $nodes_p$ in the algorithm. If the depth of any node in $nodes_p$ is less than the depth of node n, then the considered node in X should be covered by this node instead of n. We, therefore, do not count this node as a covered-node for n. This node is deleted from the nodes that n shall cover: $Y_{(n,\hat{l},c)}$. After all nodes in X are checked in a similar manner, $Y_{(n,\hat{l},c)}$ contains the priority of the transmission $(n,\hat{l},c)$. In case the transmitting node n can choose multiple interfaces to reach all downstream nodes $\mathcal{A}$, then the channels of these possible interfaces are stored in B. We represent the eligible channels for a transmission t by $c(t)$.

After completion of each round, covered-nodes A are added to $\mathcal{R}$. The algorithm completes its execution when all the nodes have been covered, i.e. when $V\backslash\mathcal{R} = \emptyset$. The algorithm returns the sets P_{PCDS}, L_{PCDS} and Λ_{PCDS}, where $P_{PCDS}(v_i)$ is the parent node of v_i, $L_{PCDS}(v_i)$ is the latency of the link connecting v_i and $P_{PCDS}(v_i)$, and $\Lambda_{PCDS}(v_i)$ is the channel used on the link connecting v_i and $P_{PCDS}(v_i)$, $\forall v_i \in V$. The PCDS tree can be constructed from these.

Algorithm 16 The PCDS algorithm

Input: $[s, \mathcal{A}, C, G = (V, E, L, \Lambda), \hat{l}]$
$\{label\}$ = Dijkstra's algorithm (G)
$\mathcal{R} = \{s\}$
while $(V \backslash \mathcal{R} \neq \emptyset)$ **do**
  $(\hat{n}, \hat{l}, \hat{c}) = \arg\max_{n \in \mathcal{R}, c \in \mathcal{A}(n)} f(n, \hat{l}, c)$
  where $f(n, \hat{l}, c)$ **is calculated as:**
  $X = Y_{(n,\hat{l},c)} = N(n, \hat{l}, c) \backslash \mathcal{R}$
  $label_{trans} = label(n) + \hat{l};$
  if $X \neq \emptyset$ **then**
    for $x = 1$ to $|X|$ **do**
      $nodes_{tmp} = \cup_{(\forall c_{tmp} \in \mathcal{A}(n) \backslash \{c\})} N(X(x), \hat{l}, c_{tmp})$
      $nodes_p = nodes_{tmp} \cap \mathcal{R}$
      for $y = 1$ to $|nodes_p|$ **do**
        $label_{round}(y) = label(nodes_p(y)) + \hat{l}$
        if $label_{round}(y) < label_{trans}$ **then**
          $Y_{(n,\hat{l},c)} = Y_{(n,\hat{l},c)} \backslash \{X(x)\}$; **break**
        end if
      end for
    end for
  end if
  $f(n, \hat{l}, c) = |Y_{(n,\hat{l},c)}|$
  $A \leftarrow Y_{(\hat{n},\hat{l},\hat{c})}$ (Y computed in f)
  let the transmission of $\hat{n}$ be represented by t
  if $\hat{n}$ can connect to A, using latency $\hat{l}$, on any of
  multiple channels (contained in set B)
  then $c(t) = B$
  $\mathcal{R} \leftarrow \mathcal{R} \cup A$
  $P_{PCDS}(A) = \hat{n}$; $L_{PCDS}(A) = \hat{l}$; $\Lambda_{PCDS}(A) = \hat{c}$
end while
Output: $[P_{PCDS}, L_{PCDS}, \Lambda_{PCDS}]$

11.6 'Interface Grouping and Transmission Scheduling' joint stage

The broadcast performance of MR^2-MC WMN can be improved by combining interface grouping and transmission scheduling sub-stages into a joint stage. By delaying the choice of the interface to use till the scheduling stage, the WMN can maximally exploit the radio-and-channel-diversity in the system. This is done by either choosing an interface (for a transmission) that is tuned to a currently unused channel *or* alternatively to a channel on which the transmission can take place alongside existing transmissions due to lack of interference. Clearly, having disjoint stages of grouping and scheduling can easily lead to a non-optimal choice of the interface to use causing broadcast performance deterioration.

For joint *grouping and scheduling*, the 'topology construction' stage instead of deciding the single interface it will use (alternatively, the channel it will use) picks all candidate interfaces and postpones the actual decision about the interface to use until scheduling. During scheduling, an appropriate choice of the interface to use, depending on the other transmissions at that time, is made to fully exploit the radio-and-channel-diversity available. This substantially improves performance especially for large number of radio interfaces Q (and channels C).

Algorithm 17 Algorithm for 'Interface grouping and Transmission Scheduling'

$time = 0$;
E= all transmissions of s;
while $E = \emptyset$ **or** $U = \emptyset$ **do**
  E= descending-sort ($height(E)$)
  $T\_sel_{round} = \emptyset$;
  for $x = 1$ to $|E|$ **do**
    $t = E(x)$;
    $c(t)$ = The channels t **can** use to transmit to its neighbors
    $C = c(t)$
    for $j = 1$ to $|C|$ **do**
      $c = j^{th}$ element of C;
      if $c \notin U$ **or** ($c \in U$ **and** if (t and T_c) do not interfere) **then**
        $\lambda(t) = c$; $T_c = T_c \cup t$; $U = U \cup c$;
        $\tau(t) = time$; $\delta(t) = time + l$;
        $T\_sel\_round = T\_sel\_round \cup t$;
      end if
    end for
  end for
  $T\_decided = T\_decided \cup T\_sel\_round$;
  $E = E \backslash \{T\_sel\_round\}$;
  $T\_not\_finished = \{t\} : \delta(t) > time$;
  $NextStop = \min(\delta(T\_not\_finished))$;
  $NextTrans = \{t\} : \delta(t) = NextStop, \forall t$;
  for $v = 1$ to $|NextTrans|$ **do**
    $\hat{c} = \lambda(NextTrans(v))$
    $T_{\hat{c}} = T_{\hat{c}} \backslash \{NextTrans(v)\}$
    if $T_{\hat{c}} = \emptyset$ **then**
      $U = U \backslash \{\hat{c}\}$;
    end if
  end for
  E_{next} = transmissions enabled by $NextTrans$;
  $E = E \cup E_{next}$;
  $E = E \backslash \{T\_decided\}$;
  $time = NextStop$;
end while

The joint grouping and scheduling algorithm is shown in Algorithm 17. The algorithm's aim is to find the start-time (τ) and end-time (δ) of each transmitting node. The algorithm works iteratively in rounds. The algorithm begins by initializing *time*, depicting current time, to zero and E, depicting eligible transmissions, to contain source node's transmission. In each round, the eligible transmissions (E) are descending-order sorted according to the *height* of respective transmitting nodes. The height of a node n is the length of the path from the node n to its furthest leaf. For all eligible transmissions $t \in E$, the function $c(t)$ represents the possible channels (channels on eligible interfaces) for this transmission. These values are available from the 'topology construction' stage. The set C contains eligible channels for the current eligible-transmission being checked (t). The choice of channel (*or interface*) to be used for a particular transmission is dictated by our desire to include as many *parallel* transmissions as possible. A channel c is found amongst the eligible channels (C) that is either not currently being used (i.e., it is not in the set of used channels U) or is being used by some transmission(s), but these transmission(s) do not interfere with t. The channel decided for a transmission t is denoted by $\lambda(t)$. If it is decided that transmission t is to use channel c, t is added to the set T_c that contains ongoing transmissions on c. The channel c is then added to U containing channels currently being used. As soon as the channel (and interface) of the transmission has been decided, we can schedule the transmission at the current time *time*. The start-time of t, denoted by $\tau(t)$, is set to *time*; while the end-time of t, denoted by $\delta(t)$, is set to *time* + l. *T_sel_round* contains the transmissions selected in the current round; while *T_decided* contains all transmissions whose start-time and end-time have already been determined.

After each round, the transmissions selected during that round (*T_sel_round*) are added to decided transmissions (*T_decided*) and deleted from eligible transmissions (E). *T_not_finished* contains all yet unfinished transmissions. The earliest finishing time for transmissions in *T_not_finished* is called *NextStop*; all transmissions completing at *NextStop* are called *NextTrans*. Let $\hat{c}$ be the channel used by a transmission in *NextTrans*. This transmission (in *NextTrans*) must be removed from $T_{\hat{c}}$ as the transmission is going to finish. If $T_{\hat{c}}$ becomes empty after removing this transmission, then $\hat{c}$ can be removed from U. We represent the transmissions of nodes reached by the finishing transmissions as E_{next}. These transmissions are eligible now, and thus they are added to E. The *time* is then set equal to *NextStop*. The algorithm schedules all transmissions similarly, and completes execution when $E = \emptyset$ (i.e., there is no eligible transmission) and $U = \emptyset$ (i.e, there is no ongoing transmission).

11.7 Simulation using an idealized scheduler

In this section, we present simulation results to evaluate the performance of PCDS assuming the idealized scheduler described in Section 11.6 and an ideal MAC as described in Section 4.2.2. We consider a static WMN having nodes located in a square of 1000^2 m^2 using a uniform random distribution. We employ the rate-range relationships derived from Qualnet for a 802.11b network (Table 11.1) and from the specifications of a commercial product (Table 11.2) in our study. We have assumed the CCA channel-assignment scheme. We will compare the PCDS algorithm with the WCDS algorithm (Chapter 4; FMM framework), the SCDS algorithm (Chapter 6;

SBM framework), and the PAMT algorithm (Chapter 9; FMM framework). We note that the WCDS and SCDS algorithms are designed for SR-SC multi-rate WMNs; therefore, all transmissions of these algorithms always use the same channel (say Channel 1). The other algorithms, i.e., PCDS and PAMT, can both use the available radio-and-channel-diversity of MR^2-MC WMNs and use any of the available radio interface and channel.

11.7.1 Performance of topology construction algorithms

In this subsection, we will discuss the performance of the PCDS heuristic. We will initially discuss the broadcast latency results, and will follow it up with broadcast throughput results. We will consider both the effect of changing node density and changing radio resources on broadcast latency and throughput performance.

Broadcast latency performance

The broadcast latency of our heuristics are all shown by normalizing it against the broadcast latency performance of the Dijkstra's tree while assuming that there is no bounds on the number of radio interfaces and channels, or the number of transmissions a node can make for the same packet at different rates. Since determining the actual optimal is NP-hard, we have used Dijkstra tree's performance (as in earlier chapters) as the theoretical lower bound or as the approximation of the optimal achievable latency.

a) Effect of changing node density:

Referring to Figure 11.1, the performance of PCDS, our SBM heuristic for MR^2-MC WMNs, becomes comparable to the performance of PAMT, our FMM heuristic for MR^2-MC WMNs, with increasing node density. We are presenting the results for the case of $Q = C = 4$. We note that for sparse WMN, the SBM heuristics might be hindered by its design choice of only employing a single link-layer broadcast rate (for all broadcast) as the network might have to decide a lower rate as the 'best' broadcast rate to maintain connectivity. This can lead to lower performance for SBM heuristics compared to FMM heuristics in sparse networks. This is, however, not a problem for dense networks. The improvement of PCDS, and PAMT over SR-SC multi-rate WMN heuristics such as SCDS and WCDS due to exploitation of radio-and-channel-diversity through parallelization of transmissions over transmitting node's multiple interfaces is also visible in Figure 11.1.

b) Effect of changing radio resources:

We observe that the performance of PCDS and PAMT is identical to the performance of SCDS and WCDS, respectively, for SR-SC WMNs, since PCDS and PAMT can improve performance of CDS and WCDS, respectively, only when multiple radio

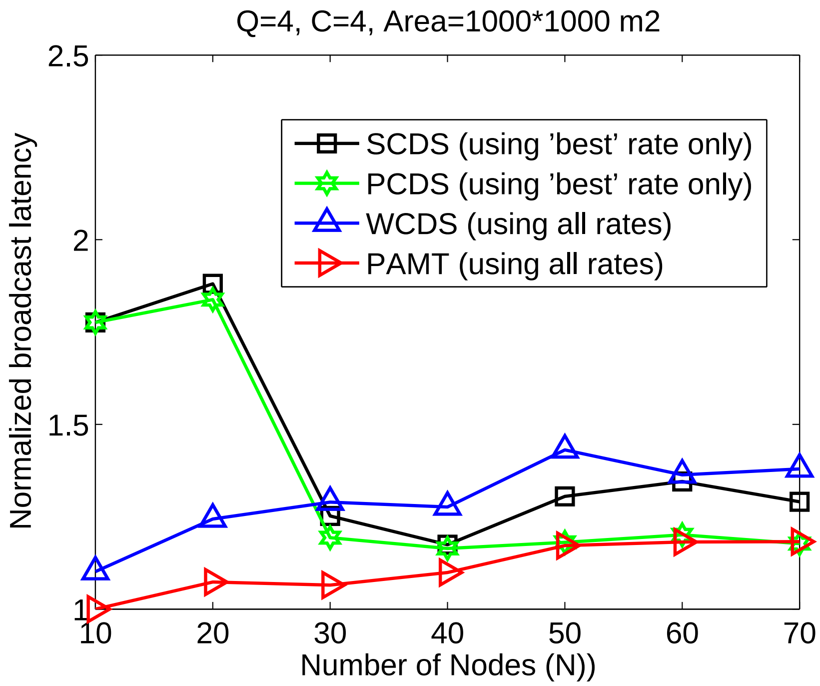
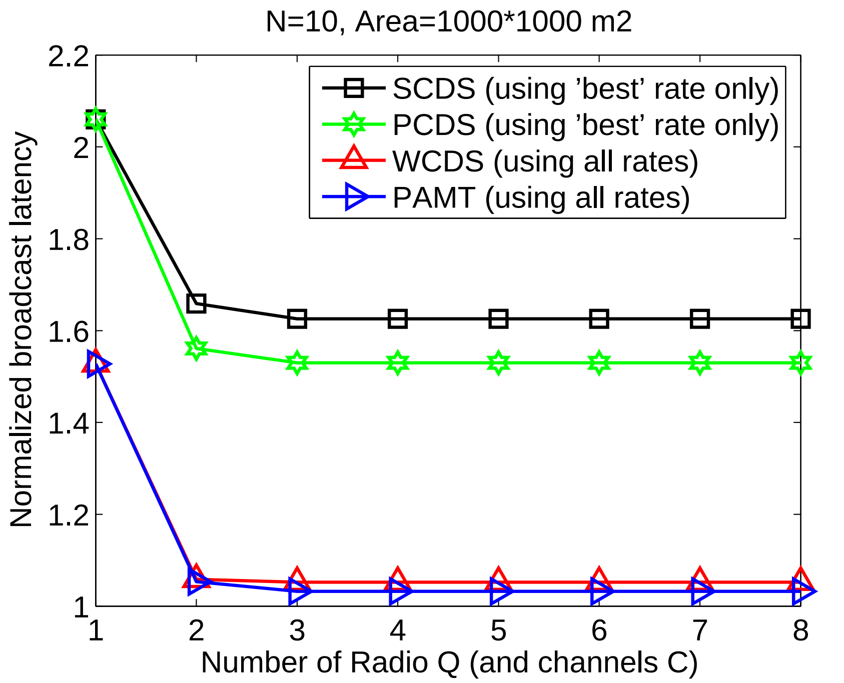


Figure 11.1: The normalized broadcast latency (with $Q = C = 4$) for varying nodes in area= 1000×1000 m^2 assuming 802.11b having the rate-range relationship of Table 11.1

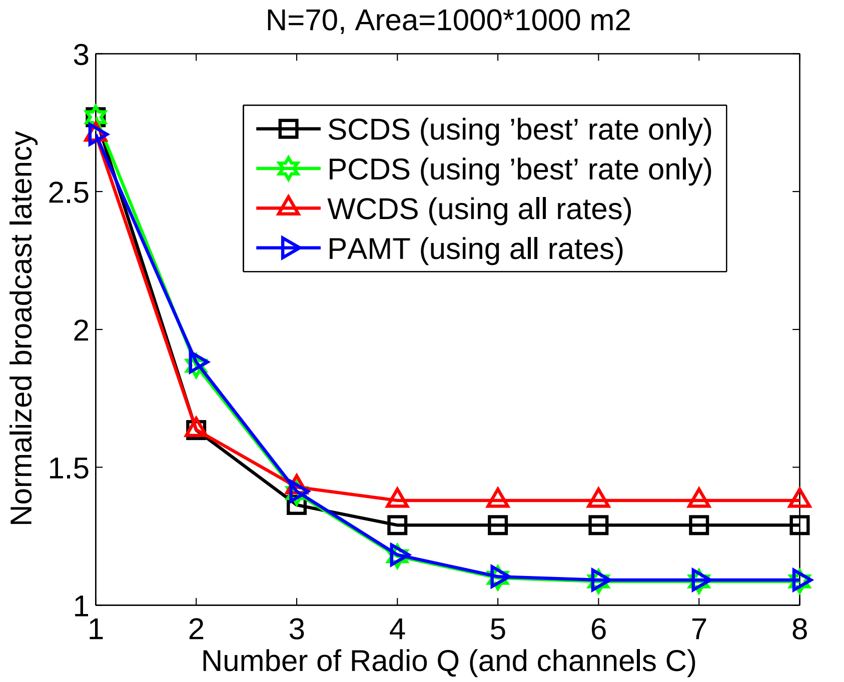
interfaces are available by *parallelizing* transmissions. This can be observed (for $Q = 1$) in Figures 11.2(a) and 11.2(b) for both sparse networks (N=10) and for dense networks (N=70). For the MR^2-MC WMN scenario (i.e., for $Q > 1$ in Figures 11.2(a) and 11.2(b)), PCDS and PAMT improve the performance of SCDS and WCDS, respectively, by *parallelizing* their transmissions. It is seen that for dense networks (Figure 11.2(b)), PCDS exactly matches the performance of PAMT.

Maximum end-to-end throughput

The results of broadcast latency are directly applicable to low throughput data flows (e.g., control traffic) as the metric applies to a *single* packet. However, for higher throughput data flows, an important metric is the maximum achievable end-to-end throughput. We essentially employ a generalization of the method used in Chapter 4, for throughput calculation in SR-SC multi-rate networks, to determine the throughput of MR^2-MC wireless mesh. An important distinction between SR-SC multi-rate WMN and MR^2-MC WMN is that interfering transmissions can take place simultaneously (on orthogonal channels) in MR^2-MC WMN. The algorithm for throughput calculation in Chapter 4 is modified to adapt to our interference model (presented in Section 11.2). The throughput results for our heuristic algorithms are provided next both for varying node density and for varying radio resources.



(a) Varying radio interfaces and channels; 802.11b network (Table 11.1)

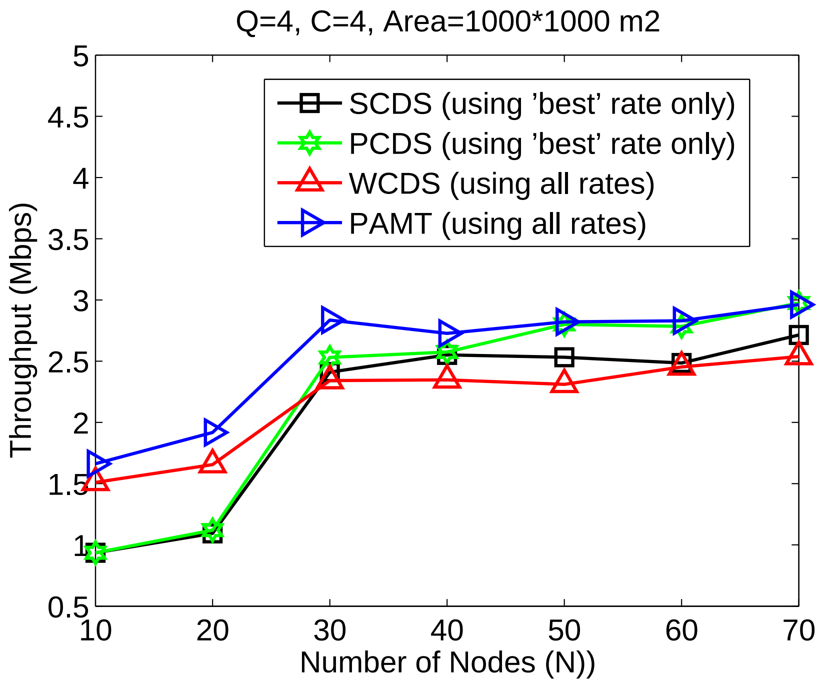


(b) Varying radio interfaces and channels; 802.11b network (Table 11.1)

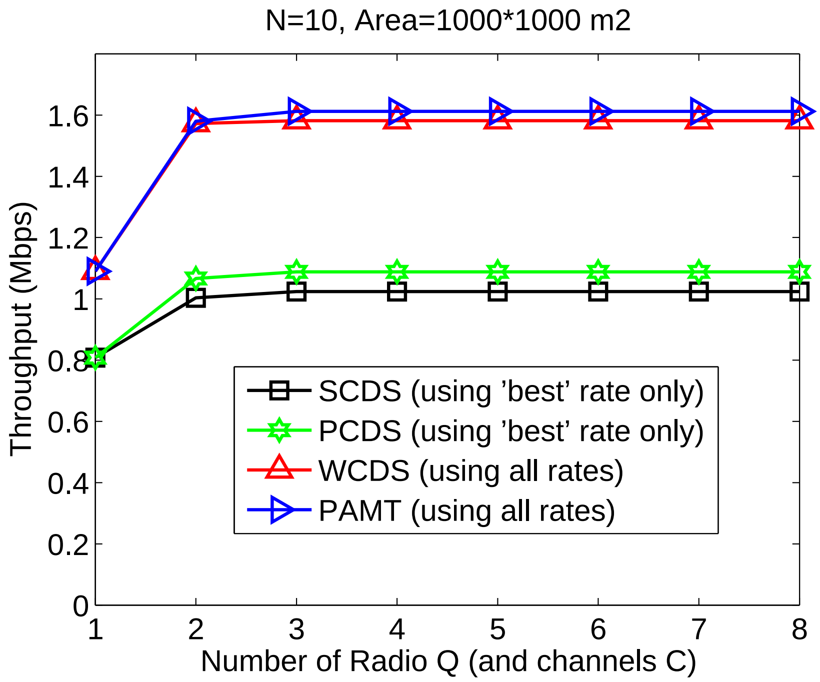
Figure 11.2: Normalized broadcast latency against varying Q (and C) for $N = 10$ and 70, respectively, in network area= 1000×1000 m^2 assuming a 802.11b network having the rate-range relationship of Table 11.1

a) Effect of changing node density:

We see the broadcast throughput result for the varying number of nodes in an area of 1000×1000 m^2 (assuming $Q = C = 4$) in Figure 11.3(a). It can be seen that as the network node density increases, the SBM heuristics, PCDS and SCDS, approach the



(a) Varying nodes; 802.11b network (Table 11.1)

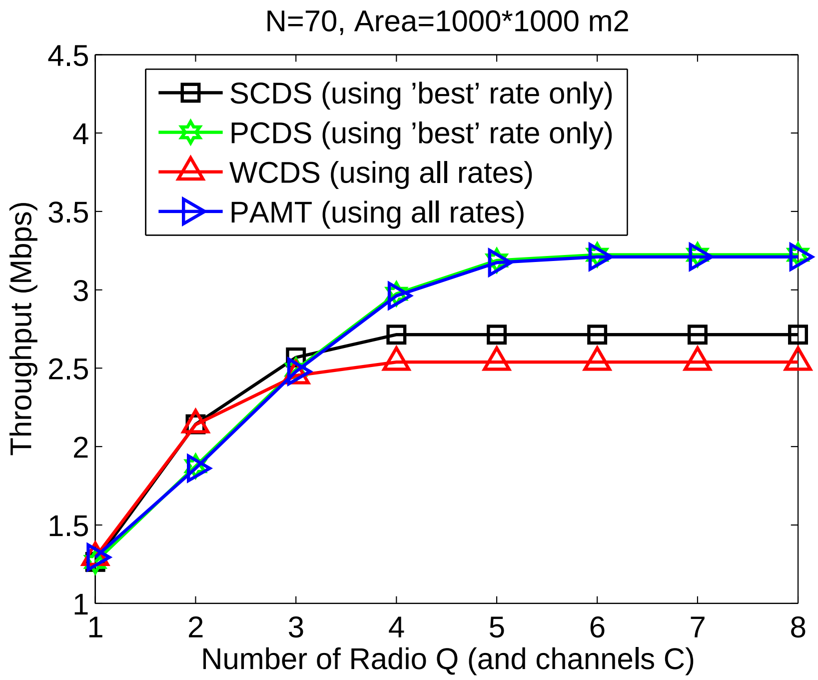


(b) Varying radio interfaces and channels; 802.11b networks (Table 11.1)

performance of the FMM heuristics, PAMT and WCDS, respectively, for the reasons stated earlier. We see that the throughput and broadcast latency results are similar, and there does not seem to be a tradeoff between them when the idealized scheduler of Section 11.6 is assumed.

b) Effect of changing radio resources:

We see broadcast throughput results for the varying radio resources (assuming N



(c) Varying radio interfaces and channels; 802.11b networks (Table 11.1)

Figure 11.3: Throughput (in Mbps) for varying nodes in area= 1000×1000 m^2 assuming a 802.11b network having the rate-range relationship as in Table 11.1

= 10 and 70, respectively) in Figures 11.3(b) and 11.3(c). It is seen that the SBM heuristics, PCDS and SCDS, can match the performance of the FMM heuristics, PAMT and WCDS, especially for dense networks.

11.7.2 Methods for calculating the "best" multicast rate

To study the viability of using RAP as a rule-of-thumb for measuring the broadcast efficiency of different rates for the SBM framework, we have performed simulations to calculate the PCDS algorithm's broadcast latency—for varying number of nodes in an area of 1000^2 m^2 using the rate-range relationship shown in Table 11.1 and Table 11.2. The data in Table 11.1 and 11.2 is to be observed together with Figure 11.5(a) and 11.5(b), respectively, which displays the probability of having a connected network (calculated using 1500 sample topologies), for shown rates. The lowest rate (1 Mbps in Table 11.1 and 11.2) has the maximum connectivity probability of 1—since we only consider networks that are connected using the lowest rate. Figures 11.5(a) and 11.5(b) indicate that connectivity probability, using a particular rate, decreases with increasing rates. We note that the quicker rates (e.g., 18 Mbps and 54 Mbps in Figure 11.5(b)) have very low connectivity probability. The average broadcast latency (geometric mean of the normalized broadcast latency of 50 random topologies) of PCDS, using the CCA channel-assignment and SBM framework (Q and $C = 2$), is displayed for different rates and node densities in Table 11.1 and Table 11.2 for different rate-range relationships. A N/A value describes the case

Trans. rate (Mbps)	Trans. range (m)	20 nodes	30 nodes	40 nodes	50 nodes	60 nodes	70 nodes	80 nodes	RAP (Mbps-km^2)
1	483	3.7527	4.9654	6.0404	6.0581	6.8685	7.2944	7.6121	0.73
2	370	3.1943	3.8988	4.2859	4.4506	4.9467	5.0476	5.0275	0.86
5.5	351	1.7585	2.146	2.2224	2.4239	2.6947	2.8111	2.8019	2.13
11	283	1.3572	1.4739	1.5991	1.6652	1.8205	1.8615	2.0036	2.77

Table 11.1: Broadcast latency of PCDS, (for Q=2, C=2, Area= 1000×1000 m^2), using SBM framework, CCA channel-assignment and the rate-range relationship of 802.11b from Qualnet [1], also see probability of a connected network for this table in Fig 11.5(a)

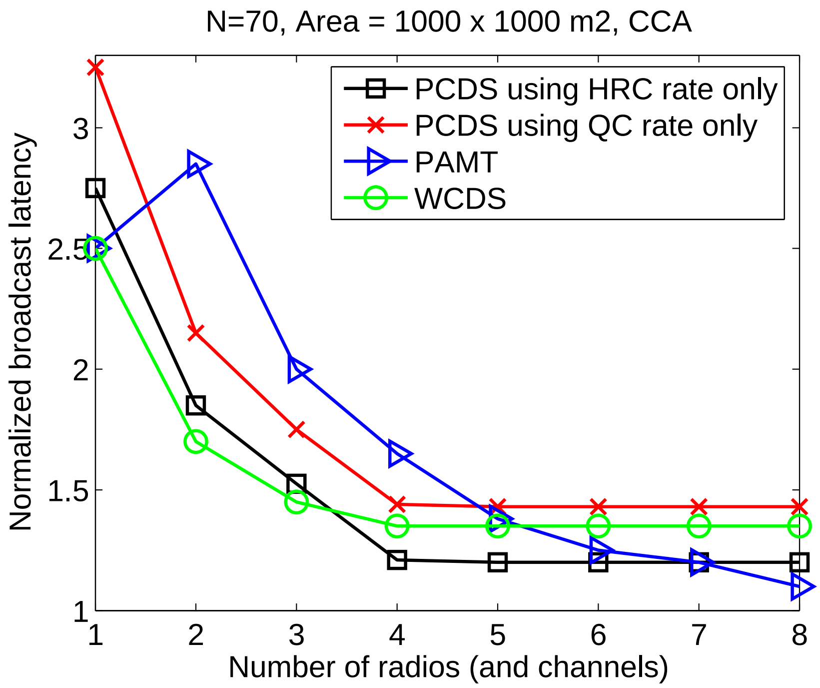
Trans. rate (Mbps)	Trans. range (m)	20 nodes	30 nodes	40 nodes	50 nodes	60 nodes	70 nodes	80 nodes	RAP (Mbps-km^2)
1	610	4.559	6.591	7.5292	7.3821	7.8563	8.0737	8.2701	1.17
6	396	1.7714	2.0259	2.1964	2.2626	2.3385	2.3496	2.4012	2.96
11	304	1.4436	1.6392	1.7638	1.9037	1.9563	2.0248	2.0773	3.19
18	183	N/A	N/A	N/A	2.5746	2.6788	2.7648	2.7033	1.89
54	76	N/A	N/A	N/A	N/A	N/A	N/A	N/A	0.98

Table 11.2: Broadcast latency of PCDS, (Q=2, C=2, Area= 1000×1000 m^2), using SBM framework, CCA channel-assignment, and the rate-range relationship of a commercial 802.11a product [2], also see probability of a connected network in Fig 11.5(b)

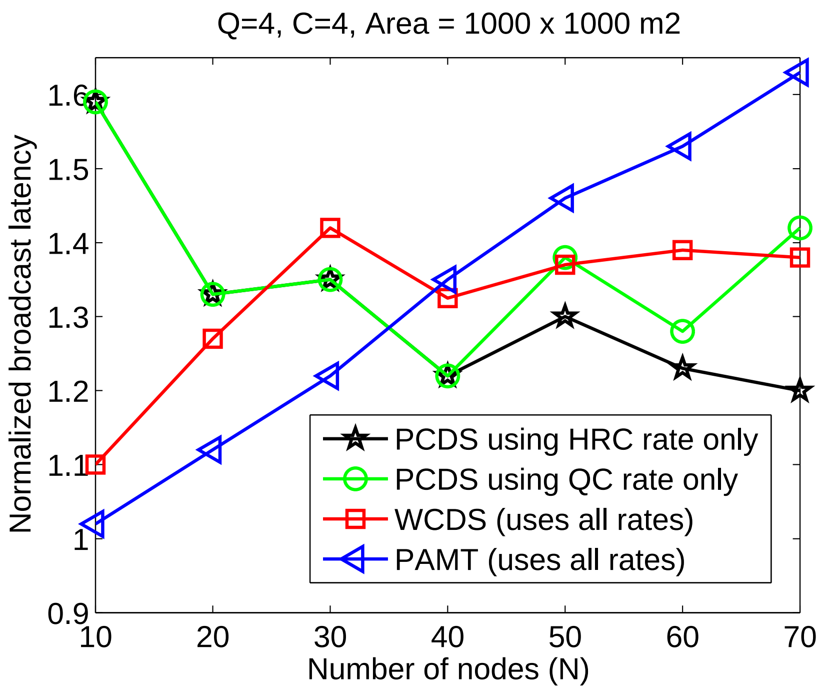
where a rate does not return a connected network. It is seen from that the highest RAP-value transmission rate (11 Mbps for both Table 11.1 and 11.2) has the best average broadcast latency.

We now evaluate the viability of the methods, proposed in Section 11.4, for determining the 'best' link-layer multicast rate . Considering the range-rate relationship shown in Table 11.1, since the RAP values are monotonically increasing with increasing rate, both the 'quickest' and 'HRC' (highest RAP-valued connected) rate methods give the same rate. We refer to this rate as the 'best' rate in Figures 11.1, 11.2(b), and 11.3(c). The normalized broadcast latency and throughput for varying number of Q (and C) with $N = 70$ are shown in Figures 11.2(b) and 11.3(c), respectively. Similarly, for varying number of nodes (using Q and $C = 4$), results are displayed in Figure 11.1. For the range-rate relationship shown in Table 11.2, Figure 11.4(b)—displaying the normalized broadcast latency for varying number of nodes (using Q and $C = 4$) in 1000^2 m^2 area—shows that the 'quickest' rate and the 'HRC' rate methods perform comparably for low node densities. For low node density, 'HRC' rate is *likely* to be the same as the 'quickest' rate according to the connectivity data in Table 11.1 (for $N \leq 40$).

For higher node densities, however, 'HRC' rate method performs much better

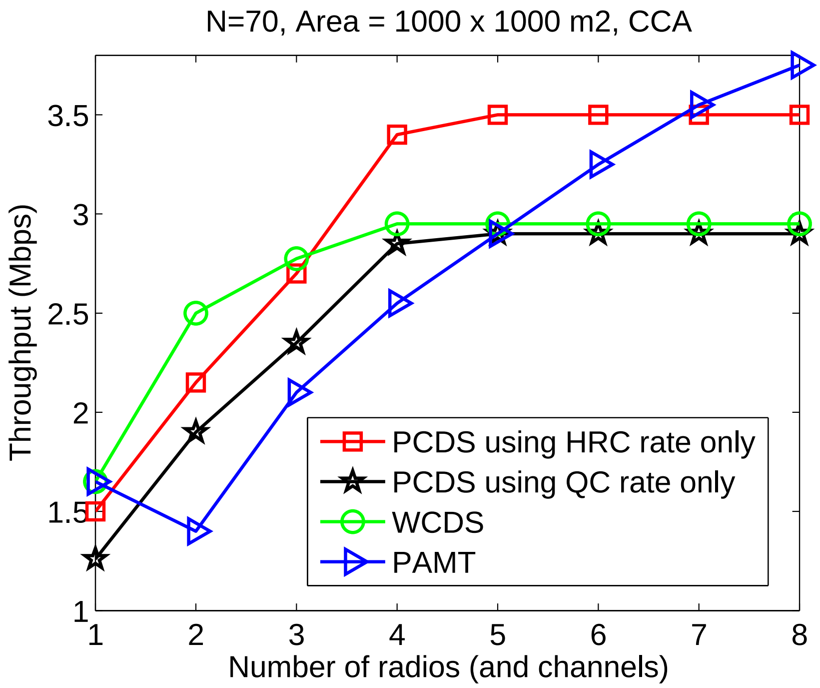


(a) Varying radio interfaces and channels; 802.11a (Table 11.2)



(b) Varying nodes; 802.11a (Table 11.2)

than the 'quickest' rate method as shown in Figure 11.4(b) (for N more than 40). Figure 11.4(a) and Figure 11.4(c) also show that latency and throughput performance of our heuristics—for $N = 70$ in 1000^2 m^2 area—using 'HRC' rate method is much better than 'quickest' rate method across the range of Q and C. This is also illustrated in Table 11.2 for Q and $C = 2$ for varying number of nodes. Hence on the basis of these experimental results, we can conclude that, assuming CCA channel-assignment, the 'HRC' rate method is an efficient way of selecting the 'best'



(c) Varying radio interfaces and channels; 802.11a (Table 11.2)

Figure 11.4: Broadcast throughput, assuming 802.11a network having the rate-range relationship of Table 11.2, against varying Q (and C) for N=70 and Area= 1000 × 1000 m^2

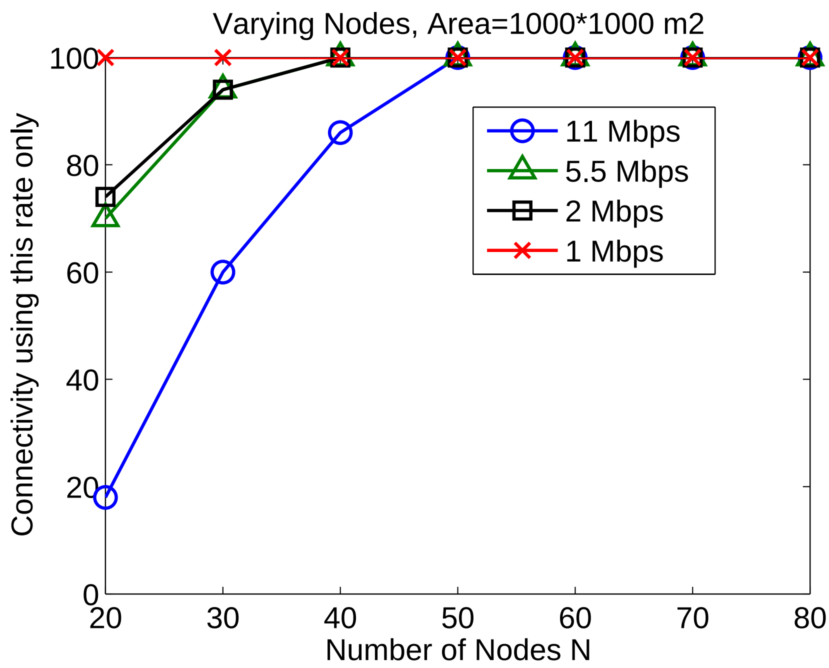
rate to be used in a SBM framework network.

11.7.3 Sensitivity of results to rate-range relationship

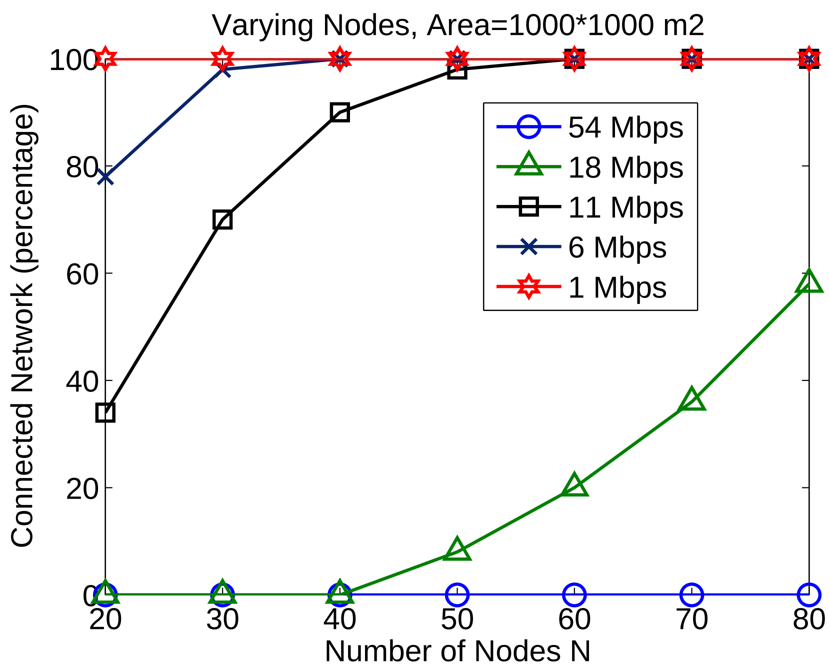
We present the sensitivity analysis of our broadcasting framework to the rate-range relationship of the WMN using the rate-range relationships shown in Table 11.1 and 11.2. We have observed that, for both considered rate-range relationships, the performance of SBM broadcast heuristics (CDS and PCDS) is comparable to performance of FMM broadcast heuristics (WCDS and PAMT), *especially for dense WMNs*. For the rate-range relationship as shown in Table 11.1, the broadcast latency, for varying Q (and C), is depicted in Figure 11.2(b) and the throughput results in Figure 11.3(c). Similarly, for the rate-range relationship as shown in Table 11.2, the broadcast latency result is shown in Figure 11.4(a), whereas the throughput result is shown in Figure 11.4(c). Both the latency and throughput results, for both rate-range relationships, show a similar trend where the SBM broadcast heuristics perform comparably to FMM broadcast heuristics, especially at high node densities.

11.8 Simulation in Qualnet using 802.11b MAC

In this section, we present results of Qualnet [1] simulations that we performed to compare the PCDS algorithm with PAMT (Chapter 9) algorithm for different 'best'



(a) Connectivity percentage for the 802.11b rate-range relationship (Table 11.1)



(b) Connectivity percentage for the 802.11a rate-range relationship (Table 11.2)

Figure 11.5: The probability of having a connected network using different rate-range relationships

transmission rates assuming a practical decentralized MAC scheduler. We assume IEEE PHY 802.11b at the physical layer, which uses a pre-configured BER-based packet reception model. The IEEE 802.11 MAC with Distributed Coordination Function (DCF) was chosen as the medium access control protocol. All default parameters are assumed unless stated otherwise.

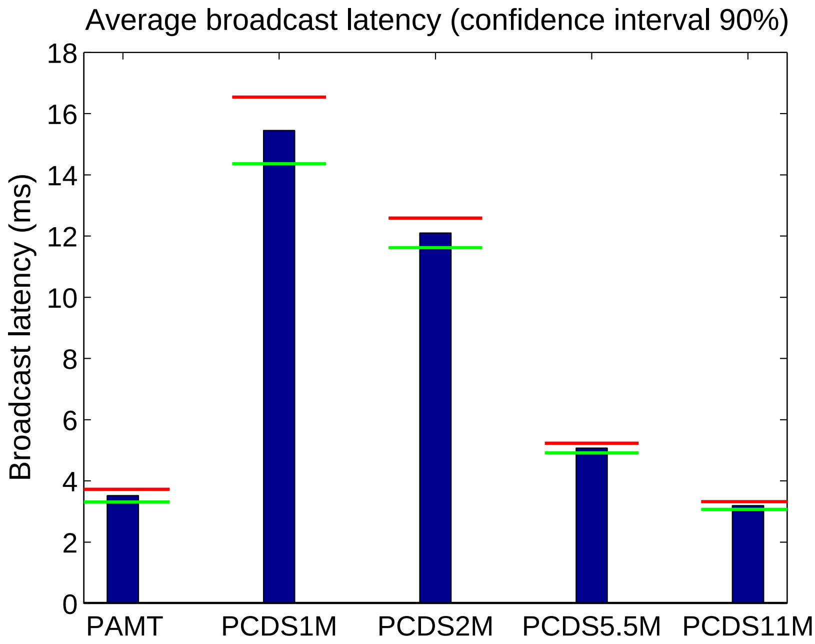


Figure 11.6: Performance of PCDS with different rates compared to PAMT

To study the viability of using RAP as a rule-of-thumb for measuring the broadcast efficiency of different rates for the SBM framework, we simulate 100 random 70-node 802.11b network topologies in an area spanning 500^2 m^2 where each node is equipped with 4 radio interfaces. We assume that $Q = C = 4$ and assume that the CCA channel-assignment scheme is used. The rate-range relationship for a 802.11b network assuming default parameters of Qualnet simulator is depicted in Table 11.1. Our results are displayed in Figure 11.6. We have only assumed those topologies for our simulation for which the network is connected for all possible rates, i.e., we only assume networks that are connected even when the quickest rate is used as the 'best' rate for the SBM framework. We compare PAMT with PCDS which uses the rate 1,2,5.5 and 11 Mbps (which are in Figure 11.6 called PCDS1, PCDS2, PCDS5.5 and PCDS11, respectively). It is observed that the higher-RAP rates perform better with PCDS using the 'best' rate of 11 Mbps (PCDS11 in Figure 11.6) almost matches the performance of PAMT. The Qualnet simulation results corroborate the idealized scheduler results (presented in the previous section), and show that RAP is a good predictor of the efficiency of a particular rate for broadcast; the results also indicate that SBM heuristics can match the performance of FMM heuristics especially when the 'best' rate is chosen sensibly.

11.9 Conclusions

In this chapter, we proposed a centralized heuristic called PCDS that utilizes the SBM broadcast framework which dictates that a single “best” link-layer rate be used for all multicast. We have also proposed techniques that can be used to select the “best” rate to be used with our SBM framework. Our analysis of SBM showed that, in dense settings, its performance is comparable to the more powerful FMM framework that can adapt the link-layer multicast rate of each frame. We have seen that the SBM heuristic PCDS matches the performance of the the more powerful FMM heuristic PAMT. We have also observed that PCDS, like PAMT, adapts to the radio resources available by exploiting the radio-and-channel-diversity by parallelizing its transmission over the various interfaces on transmitting nodes.

Chapter 12

Summary of Results

This chapter serves as a conclusion to Part II, "Improving broadcast performance of multi-radio multi-channel multi-rate WMNs" of this thesis. We will enlist some of the research objectives discussed in Section 1.4 that are relevant to this part of the thesis.

- *Study the MLB problem for MR^2-MC WMNs and provide algorithms that improve the performance of single-radio single-rate broadcast techniques. Quantify the extent of improvement by comparing with radio-and-channel-diversity and rate-diversity unaware broadcast techniques.*

  In Chapter 9, we proposed four centralized MLB heuristics for MR^2-MC WMNs; the first (MSPT) does not exploit the 'wireless broadcast advantage' (WBA) offered in WMNs, the second (MWT) exploits WBA but not the availability of multiple interfaces on the same node, while the other two (LMT and PAMT) differ in how they exploit both WBA and the radio-and-channel-diversity on individual nodes. *Interestingly, both PAMT and LMT perform fairly close to the theoretical optimal, resulting in latencies that are on average only* $\sim 10 - 20\%$ *higher.*

  In Chapter 10, we have proposed a *distributed and localized* broadcast heuristic that exploits the "wireless broadcast advantage", the rate-diversity, and the radio-and-channel diversity of a MR^2-MC WMN. These algorithms all operate by initially constructing a basic source-independent CDS and pruning it to increase the rate of individual transmissions, and subsequently by constructing a source-specific tree over this "pruned CDS". Simulation-based experimental studies show that the MRDT decentralized algorithm can provide low broadcast latencies close to that of the best-performing, centralized PAMT heuristic, especially for practical environments where the number of interfaces (Q) is relatively low $(2 - 3)$.

  In Chapter 11, we proposed a centralized heuristic called PCDS that utilizes the SBM broadcast framework which dictates that a single "best" link-layer rate be used for all multicast. We have also proposed techniques that can be used to select the "best" rate to be used with our SBM framework. Our analysis of SBM showed that, in dense settings, its performance is comparable to the more powerful FMM framework that can adapt the link-layer multicast rate of each frame. We have seen that PCDS matches the performance of the PAMT algorithm that utilizes the more powerful FMM framework. We have also observed that PCDS, like PAMT, adapts to the radio resources available by exploiting the radio-and-channel-diversity by parallelizing its transmission over the various interfaces on transmitting nodes.

- *Provide general insight and identify rules that can enable multi-rate WMN protocol designers for MR^2-MC WMNs to decide how to exploit radio-and-channel and rate-diversity for improving broadcast performance.*

  In Chapters 4,5, and 6 of Part I of this thesis, we had shown the utility of the product of transmission rate and transmission coverage area (or rate-area product or RAP for short) as a measure of efficiency of a certain transmission rate for SR-SC multi-rate WMNs. An important conclusion was that a higher rate is not necessarily preferable for broadcast. It was shown that a rate, notwithstanding how high it is, is broadcast efficient only if its RAP is high. We have corroborated this proposition for MR^2-MC WMNs in this part of the thesis, and have shown in Chapter 11 that RAP is a good predictor of broadcast efficiency of a particular rate even for MR^2-MC WMNs. Consequently, high transmission rates that do not have high RAP values can be disregarded for broadcast (see Figure 11.4(a) in Chapter 11).

  In Chapters 9, 10 and 11, we have identified that a well-designed broadcast algorithm for MR^2-MC WMNs must exploit the interface-diversity (or, the radio-and-channel-diversity) such network offers. We have observed that the best performing algorithms, such as PAMT (Chapter 9), MRDT (Chapter 10), and PCDS (Chapter 11), generally adapt to the radio resources available and parallelizes transmission over multiple interfaces when available.

  In Chapter 9, we made an important observation: we showed that broadcast performance is impacted by the channel-assignment strategy used, due to the inherent conflict for a channel-assignment scheme between greater connectivity and lower channel contention. We observed that a channel-assignment scheme designed for unicast flows may sometimes perform poorly for broadcast/multicast flows. In our simulations, the performance of CCA (which generally performs poorly for unicast flows) is generally better than both VCA and INSTC.

Part III

Future work

Chapter 13

Open Research Questions

Our goal in this thesis has not been the development of *practical protocols*, but rather the use of algorithmic techniques to establish some principles. In particular, we wanted to demonstrate the potential benefits that broadcast traffic can reap from rate diversity and multi-channel, multi-radio WMN architectures. A lot of additional research questions will need to be solved to provide effective practical support for point-to-multipoint traffic in WMNs. Some of these are:

- Variable link quality (i.e., the link loss rate) is a challenge in real WMN environments. In this thesis, we have assumed that if a particular link can use a particular transmission rate at one time, then that rate can be utilized by that link at all times. In practice, however, the link quality in multihop wireless networks is time variant. The impact of lossy links in wireless networks is particularly important, since wireless standards (e.g., IEEE 802.11) usually do not provide retransmission-based link-layer reliability for broadcast traffic. Thus, we need to analyze: *How do we incorporate reliability into the design of our broadcast algorithms keeping in view the potentially time-varying quality of links to different neighbors?*

- Our studies have demonstrated the fact that existing unicast-oriented channel-allocation schemes do not work uniformly well for both broadcast and unicast traffic. Since real WMNs will have a mixture of both traffic types, another important question is: *How do we design channel-allocation mechanisms for multi-radio multi-channel WMNs that perform equally well for both unicast and broadcast traffic, and are relatively insensitive to variations in the relative proportion of broadcast and unicast flows?*

- While in this thesis we have utilized the broadcast latency of Dijkstra's tree (assuming that the Dijkstra's tree can utilize unlimited number of radios and channels, and each node can perform unlimited number of distinct-rate transmissions) as the approximate bound on broadcast latency performance (since it is non-trivial to calculate the optimal performance bound for multi-radio multi-channel multi-rate WMNs), an interesting future work direction is to develop theoretical bounds on the optimal broadcast latency performance for a general multi-radio multi-rate WMN.

- There are open questions for the SBM framework. More specifically, a question requiring further attention is: "*How does the connecting and disconnecting of new nodes affect the choice of the 'best' rate to be used in a SBM framework*". For the SBM framework to be practical, it is important that the 'best' rate determined does not continuously change with every addition and deletion of nodes in the WMN. Also, the method used to determine the 'best' rate must also appropriately consider the variable nature of the link quality in multi-rate WMNs in its choice.

- While multi-radio WMN architecture is promising in principle, there are practical concerns in building a multi-radio WMN [87]. For one, limitations arise due to increased interference between radio interfaces co-existing on the same node, even when they are tuned to mutually orthogonal channels. Since, we have analyzed the performance of our broadcast algorithms using simulations assuming idealized and non-idealized MAC, an important pending work is to study: *How this increased crosstalk and interference between radio interfaces, co-located on the same mesh node, affects the performance of our broadcast algorithms in practical multi-radio WMNs?.*

Bibliography

[1] Scalable Networks Inc. http://www.scalable-networks.com/.

[2] Cisco Systems. Cisco Aironet 802.11a/b/g Wireless LAN Client Adapters (CB21AG and PI21AG) Installation and Configuration Guide, 2004.

[3] I.F. Akyildiz, X. Wang, and W. Wang. Wireless mesh networks: a survey. *Computer Networks, Elsevier*, 47(4):445–487, 2005.

[4] IF Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor networks: a survey. *Computer Networks, Elsevier*, 38(4):393–422, 2002.

[5] The IEEE 802.11 standards. http://standards.ieee.org/getieee802/802.11.html.

[6] The status of the IEEE 802.11s standard project. http://grouper.ieee.org/groups/802/11/Reports/tgs_update.htm.

[7] IEEE 802.16-2004. "IEEE Standard for Local and Metropolitan Area Networks - Part 16: Air Interface for Fixed Broadband Wireless Access Systems", October 2004.

[8] IEEE Std. 802.15.4-2003. Institute of Electrical and Electronics Engineers, Inc., "Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low Rate Wireless Personal Area Networks (LR-WPANs)", New York, IEEE Press. October 1, 2003. .

[9] Implementation of Wireless Mesh Network in Taiwan. http://www.nortel.com/corporate/pressroom/feature_article/2005b/collateral/feature_taiwan_wireless_meshv2.pdf.

[10] B. Raman and K. Chebrolu. Design and evaluation of a new MAC protocol for long-distance 802.11 mesh networks. *Proceedings of the 11th Annual International Conference on Mobile Computing and Networking (MobiCom '05)*, pages 156–169, 2005.

[11] J. Eriksson, S. Agarwal, P. Bahl, and J. Padhye. Feasibility study of mesh networks for all-wireless offices. *Proceedings of the 4th ACM International*

Conference On Mobile Systems, Applications And Services (MOBISYS '06) Conference, pages 69–82, 2006.

[12] P. Gupta and P. R. Kumar. The capacity of wireless networks. *IEEE Transactions on Information Theory*, IT-46(2):388–404, March 2000.

[13] P. Kyasanur and N. H. Vaidya. Capacity of multi-channel wireless networks: impact of number of channels and interfaces. In *Proceedings of the 11th Annual International Conference on Mobile Computing and Networking (MobiCom '05)*, pages 43–57, New York, NY, USA, 2005. ACM Press.

[14] R. Draves, J. Padhye, and B. Zill. Routing in multi-radio, multi-hop wireless mesh networks. In *Proceedings of the 10th Annual International Conference on Mobile Computing and Networking (MobiCom '04)*, pages 114–128, New York, NY, USA, 2004. ACM Press.

[15] A. Raniwala and T. Chiueh. Architecture and algorithms for an IEEE 802.11-based multi-channel wireless mesh network. *Proceedings of the 24th Annual Conference of the IEEE Computer and Communications Societies (Infocom '05)*, 3, 2005.

[16] H Rubens B Awerbuch, D Holmer. High throughput route selection in multi-rate ad hoc wireless networks. *Lecture Notes in Computer Science*, 2928:253 – 270, Jan 2004.

[17] S. Bansal, R. Shorey, and A. A. Kherani. Performance of TCP and UDP protocols in multi-hop multi-rate wireless networks. In *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '04)*, volume 1, page 231, 2004.

[18] G. Holland, N. Vaidya, and P. Bahl. A rate-adaptive MAC protocol for multi-Hop wireless networks. In *Proceedings of the 7th ACM Annual International Conference on Mobile Computing and Networking (MobiCom '01)*, pages 236–251, New York, NY, USA, 2001. ACM Press.

[19] D. Qiao, S. Choi, and K.G. Shin. Goodput analysis and link adaptation for IEEE 802.11a wireless LANs. *IEEE Transactions on Mobile Computing*, 1(4):278–292, 2002.

[20] J. E. Wieselthier, G. D. Nguyen, and A. Ephremides. Energy-efficient broadcast and multicast trees in wireless networks. *Mobile Networks Application, Springer*, 7(6):481–492, 2002.

[21] J. Cartigny, D. Simplot, and I. Stojmenovic. Localized minimum-energy broadcasting in ad-hoc networks. In *Proceedings of the 22nd Annual Conference of*

the IEEE Computer and Communications Societies (Infocom '03), page 2210, 2003.

[22] M. Agarwal, L. Gao, J.H. Cho, and J. Wu. Energy Efficient Broadcast in Wireless Ad hoc Networks with Hitch-hiking. *Mobile Networks and Applications, Springer*, 10(6):897–910, 2005.

[23] J. Widmer, C. Fragouli, and J.Y. Le Boudec. Low-complexity energy-efficient broadcasting in wireless ad-hoc networks using network coding. *Proceedings of the Workshop on Network Coding, Theory, and Applications (NetCod '05)*, 2005.

[24] A. Durresi, VK Paruchuri, SS Iyengar, and R. Kannan. Optimized broadcast protocol for sensor networks. *Computers, IEEE Transactions on*, 54(8):1013–1024, 2005.

[25] I. Kang and R. Poovendran. Maximizing static network lifetime of wireless broadcast ad hoc networks. *IEEE International Conference on Communications (ICC'03), 2003.*, 3, 2003.

[26] R. Gandhi, S. Parthasarathy, and A. Mishra. Minimizing broadcast latency and redundancy in ad hoc networks. In *Proceedings of the 4th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc '03)*, pages 222–232, New York, NY, USA, 2003. ACM Press.

[27] Y. Wu, P. A. Chou, Q. Zhang, K. Jain, W. Zhu, and S.Y. Kung. Network planning in wireless ad hoc networks: a cross-layer approach. *IEEE Journal on Selected Areas in Communications*, 23(1):136, 2005.

[28] P. Kyasanur and N.H. Vaidya. Routing and interface assignment in multi-channel multi-interface wireless networks. *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '05), Volume 4, Page(s): 2051 - 2056*, 2005.

[29] B. Williams and T. Camp. Comparison of broadcasting techniques for mobile ad hoc networks. *Proceedings of the 3rd ACM International Symposium on Mobile Ad hoc Networking and Computing (MobiHoc '02)*, pages 194–205, 2002.

[30] C. Cordeiro, H. Gossain, and D.P. Agrawal. Multicast over wireless mobile ad hoc networks: present and future directions. *Network, IEEE*, 17(1):52–59, 2003.

[31] S. Guha. Approximation Algorithms for Connected Dominating Sets. *Algorithmica*, 20(4):374–387, 1998.

[32] J. Wu and H. Li. On calculating connected dominating set for efficient routing in ad-hoc wireless networks. *Proceedings of the 3rd International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications (ACM DIALM '99)*, pages 7–14, 1999.

[33] A. Qayyum, L. Viennot, and A. Laouiti. Multipoint relaying for flooding broadcast messages in mobile wireless networks. *Proceedings of the 35th Annual Hawaii International Conference on System Sciences (HICSS '02)*, pages 3866–3875, 2002.

[34] I. Stojmenovic, M. Seddigh, and J. Zunic. Dominating sets and neighbor elimination-based broadcasting algorithms in wireless networks. *IEEE Transactions on Parallel and Distributed Systems*, 13(1):14–25, 2002.

[35] S. Ni, Y. Tseng, Y. Chen, and J. Sheu. The broadcast storm problem in a mobile ad hoc network. In *Proceedings of the 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom '99)*, pages 151–162, New York, NY, USA, 1999. ACM Press.

[36] H. Lim and C. Kim. Flooding in wireless ad hoc networks. *Computer Communications, Elsevier*, 24(3-4):353, 2001.

[37] S. J. Lee, W. Su, and M. Gerla. On-demand multicast routing protocol in multihop wireless mobile networks. *Mobile Networks and Applications, Springer*, 7(6):441–453, 2002.

[38] Z.J. Haas, J.Y. Halpern, and L. Li. Gossip-based ad hoc routing. *IEEE/ACM Transactions on Networking (TON)*, 14(3):479–491, 2006.

[39] B. Das, R. Sivakumar, and V. Bharghavan. Routing in ad hoc networks using a virtual backbone. *Proceedings of the IEEE International Conference on Computer Communications and Networks (IC3N)*, 21, 1997.

[40] B. Das and V. Bharghavan. Routing in ad-hoc networks using minimum connected dominating sets. *Proceedings of the IEEE International Conference on Communications (ICC '97), Montreal*, 1, 1997.

[41] B. Das, R. Sivakumar, and V. Bharghavan. Routing in ad hoc networks using a spine. *Proceedings of the 6th International Conference on Computer Communications and Networks (ICCCN '97)*, pages 34–39, 1997.

[42] R. Sivakumar, B. Das, and V. Bharghavan. An improved spine-based infrastructure for routing in ad hoc networks. *Proceedings of the IEEE Symposium on Computers and Communications (ISCC '98)*, 1998.

[43] C. Adjih, P. Jacquet, and L. Viennot. Computing connected dominated sets with multipoint relays. *Journal of Ad Hoc and Sensor Wireless Networks*, 1, 2005.

[44] W. Lou and J. Wu. On reducing broadcast redundancy in ad hoc wireless networks. *IEEE Transactions on Mobile Computing*, 1:111, 2002.

[45] P.J. Wan, K.M. Alzoubi, and O. Frieder. Distributed Construction of Connected Dominating Set in Wireless Ad Hoc Networks. *Mobile Networks and Applications, Springer*, 9(2):141–149, 2004.

[46] S. Butenko, X. Cheng, C. Oliveira, and P. Pardalos. A new heuristic for the minimum connected dominating set problem on ad hoc wireless networks. *Proceedings of the 4th International Conference on Cooperative Control and Optimization (CCO '04)*, pages 61–73, 2004.

[47] F. Dai and J. Wu. An extended localized algorithm for connected dominating set formation in ad hoc wireless networks. *IEEE Transactions on Parallel and Distributed Systems*, 15(10):908–920, 2004.

[48] G. Călinescu, I.I. Măndoiu, P.J. Wan, and A.Z. Zelikovsky. Selecting Forwarding Neighbors in Wireless Ad Hoc Networks. *Mobile Networks and Applications, Kluwer Academic Publishers*, 9(2):101–111, 2004.

[49] F. Dai and J. Wu. Distributed dominant pruning in ad hoc networks. *IEEE International Conference on Communications (ICC'03), 2003.*, 1, 2003.

[50] C.W. Wu and Y.C. Tay. AMRIS: a multicast protocol for ad hoc wireless networks. *Proceedings of the IEEE Military Communications Conference Proceedings (MilCom '99), volume 1*, 1999.

[51] E.M. Royer and C.E. Perkins. Multicast Ad hoc On-Demand Distance Vector (MAODV) Routing. *draft-ietf-manet-maodv-00, July*, 700, 2000.

[52] L. Ji and MS Corson. A lightweight adaptive multicast algorithm. *Proceedings of the 41st IEEE Global Telecommunications Conference (Globecom '98)*, 2, 1998.

[53] K. Obraczka, G. Tsudik, and K. Viswanath. Pushing the limits of multicast in ad hoc networks. *Proceedings of the 21st IEEE International Conference on Distributed Computing Systems (ICDCS '01)*, 1, 2001.

[54] J.J. Garcia-Luna-Aceves and E.L. Madruga. A Multicast Routing Protocol for Ad-Hoc Networks. *Proceedings of the 23rd Annual Conference of the IEEE Computer and Communications Societies (Infocom '99)*, 99:784–792, 1999.

[55] C.C. Chiang, M. Gerla, and L. Zhang. Forwarding Group Multicast Protocol (FGMP) for multihop, mobile wireless networks. *Cluster Computing, Springer*, 1(2):187–196, 1998.

[56] J. Xie, R. Talpade, A. Mcauley, and M. Liu. AMRoute: ad hoc multicast routing protocol. *Mobile Networks and Applications, Springer*, 7(6):429–439, 2002.

[57] P. Sinha, R. Sivakumar, and V. Bharghavan. MCEDAR: multicast core-extraction distributed ad hoc routing. *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '99)*, pages 1313–1317, 1999.

[58] C. Adjih, P. Jacquet, and L. Viennot. Computing connected dominated sets with multipoint relays. *Journal of Ad Hoc and Sensor Wireless Networks*, 1, 2005.

[59] C. Gui and P. Mohapatra. Scalable multicasting in mobile ad hoc networks. In *Proceedings of the 23rd Annual Conference of the IEEE Computer and Communications Societies (Infocom '04)*, page 2119, 2004.

[60] A. Chen, D. Lee, G. Chandrasekaran, and P. Sinha. HIMAC: High Throughput MAC Layer Multicasting in Wireless Networks. *Proceedings of the 3rd IEEE International Conference on Mobile Adhoc and Sensor Sysetems (MASS '06)*, pages 41–50, 2006.

[61] A. Roy and S.K. Das. QM 2 RP: A QoS-Based Mobile Multicast Routing Protocol Using Multi-Objective Genetic Algorithm. *Wireless Networks*, 10(3):271–286, 2004.

[62] D. De Couto, D. Aguayo, J. Bicket, and R. Morris. A high-throughput path metric for multi-hop wireless routing. In *Proceedings of the 9th ACM Annual International Conference on Mobile Computing and Networking (MobiCom '03)*, pages 134–146, New York, NY, USA, 2003. ACM Press.

[63] Y. Seok, J. Park, and Y. Choi. Multi-rate aware routing protocol for mobile ad hoc networks. *The 57th IEEE Vehicular Technology Conference (VTC 2003-Spring), 2003.*, 3, 2003.

[64] S. Zhao, Z. Wu, A. Acharya, and D. Raychaudhuri. PARMA: A PHY/MAC Aware Routing Metric for Ad-Hoc Wireless Networks with Multi-Rate Radios. *IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM'05), June*, 2005.

[65] W. Zhou, D. Zhang, and D. Qiao. Comparative study of routing metrics for multi-radio multi-channel wireless networks. *IEEE Wireless Communications and Networking Conference (WCNC'06), 2006.*, 1, 2006.

[66] M. Alicherry, R. Bhatia, and L. Li. Joint channel assignment and routing for throughput optimization in multiradio wireless mesh networks. In *Proceedings of the 11th ACM Annual International Conference on Mobile Computing and Networking (MobiCom '05)*, 2005.

[67] M. Kodialam and T. Nandagopal. Characterizing the capacity region in multi-radio multi-channel wireless mesh networks. In *Proceedings of the 11th Annual International Conference on Mobile computing and Networking (MobiCom '05)*, pages 73–87, New York, NY, USA, 2005. ACM Press.

[68] M. Song, J. Wang, and Q. Hao. Broadcasting Protocols for Multi-Radio Multi-Channel and Multi-Rate Mesh Networks. *IEEE International Conference on Communications (ICC'07), 2007.*, pages 3604–3609, 2007.

[69] L. Li, B. Qin, C. Zhang, and H. Li. Efficient Broadcasting in Multi-radio Multi-channel and Multi-hop Wireless Networks Based on Self-pruning.

[70] Z. K. Yang W. Q. Cheng T. Wang, B. Li. A minimized latency broadcast in multi-rate wireless mesh networks: distributed formulation and rate first algorithm. *IEEE ICME, Beijing, 2007.*

[71] J. So and N. Vaidya. Multi-Channel MAC for Ad Hoc Networks: Handling Multi-Channel Hidden Terminals Using A Single Transceiver. In *Proceedings of the 5th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc '04)*, 2004.

[72] P. Bahl, R. Chandra, and J. Dunagan. SSCH: slotted seeded channel hopping for capacity improvement in IEEE 802.11 ad-hoc wireless networks. *Proceedings of the 10th ACM Annual International Conference on Mobile Computing and Networking (MobiCom '04)*, pages 216–230, 2004.

[73] A. Raniwala, K. Gopalan, and T. Chiueh. Centralized channel assignment and routing algorithms for multi-channel wireless mesh networks. *ACM SIGMOBILE Mobile Computing and Communications Review*, 8(2):50–65, 2004.

[74] J. Tang, G. Xue, and W. Zhang. Interference-aware topology control and qos routing in multi-channel wireless mesh networks. In *Proceedings of the 6th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc '05)*, pages 68–77, New York, NY, USA, 2005. ACM Press.

[75] K. Xu, M. Gerla, and S. Bae. Effectiveness of RTS/CTS handshake in IEEE 802.11 based ad hoc networks. *Ad Hoc Network Journal*, 1(1):107–123, 2003.

[76] K. Jain, J. Padhye, V. N. Padmanabhan, and L. Qiu. Impact of interference on multi-hop wireless network performance. In *Proceedings of the 9th Annual International Conference on Mobile Computing and Networking (MobiCom '03)*, pages 66–80, New York, NY, USA, 2003. ACM Press.

[77] M. Elkin and G. Kortsarz. Logarithmic inapproximability of the radio broadcast problem. *Journal of Algorithms, Elsevier*, 52(1):8–25, 2004.

[78] J.E. Wieselthier, G.D. Nguyen, and A. Ephremides. On the construction of energy-efficient broadcast and multicast trees in wireless networks. *Proceedings of the 10th Annual Conference of the IEEE Computer and Communications Societies (Infocom '00), volume 2*, 2000.

[79] C.T. Chou, A. Misra, and J. Qadir. Low latency broadcast in multi-rate wireless mesh networks. Technical Report 0514, School of CSE, UNSW, `ftp://ftp.cse.unsw.edu.au/pub/doc/papers/UNSW/0514.pdf`, 2005.

[80] K. R. Apt. The essence of constraint propagation. *Theoretical Computer Science*, 221(1-2):179, 1999.

[81] C.T. Chou and A. Misra. Low latency multimedia broadcast in multi-rate wireless meshes. In *Proceedings of the First IEEE Workshop on Wireless Mesh Networks (WiMesh '05), in conjunction with IEEE SECON '05, Santa Clara, CA*, 2005.

[82] G.R. Cantieni, Q. Ni, C. Barakat, and T. Turletti. Performance analysis under finite load and improvements for multi-rate 802.11. *Computer Communications, Elsevier*, 28(10):1095–1109, 2005.

[83] M. Heusse, F. Rousseau, G. Berger-Sabbatel, and A. Duda. Performance anomaly of 802.11 b. *Proceedings of the 22nd Annual Conference of the IEEE Computer and Communications Societies (Infocom '03), Volume 2*, 2003.

[84] U.T. Nguyen and X. Xiong. Rate-adaptive multicast in mobile ad-hoc networks. *Proceedings of the 1st IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WIMOB '05)*, pp. 352- 360 Vol. 3, 2005.

[85] S.H.Y. Wong, S. Lu, H. Yang, and V. Bharghavan. Robust rate adaptation for 802.11 wireless networks. *Proceedings of the 12th ACM Annual International Conference on Mobile Computing and Networking (MobiCom '06)*, 2006.

[86] J.C. Bicket. Bit-rate selection in wireless networks. *Master's thesis, Massachusetts Institute of Technology*, 2005.

[87] J. Robinson, K. Papagiannaki, C. Diot, X. Guo, and L. Krishnamurthy. Experimenting with a multi-radio mesh networking testbed. *1st workshop on Wireless Network Measurements (WiNMee'05), 2005.*