

RICE UNIVERSITY

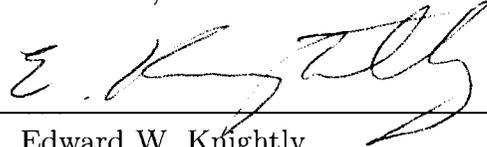
**Distance-1 Constrained Channel Assignment in Single Radio  
Wireless Mesh Networks**

by

**Ehsan Aryafar**

A THESIS SUBMITTED  
IN PARTIAL FULFILLMENT OF THE  
REQUIREMENTS FOR THE DEGREE  
**MASTER OF SCIENCE**

APPROVED, THESIS COMMITTEE:



---

Dr. Edward W. Knightly  
Professor  
Electrical and Computer Engineering



---

Dr. Rudolf H. Riedi  
Associate Professor  
Statistics



---

Dr. Ashutosh Sabharwal  
Assistant Professor  
Electrical and Computer Engineering

Houston, Texas

September, 2007

UMI Number: 1455216

### INFORMATION TO USERS

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleed-through, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

**UMI**<sup>®</sup>

---

UMI Microform 1455216

Copyright 2008 by ProQuest LLC.

All rights reserved. This microform edition is protected against unauthorized copying under Title 17, United States Code.

ProQuest LLC  
789 E. Eisenhower Parkway  
PO Box 1346  
Ann Arbor, MI 48106-1346

## ABSTRACT

### **Distance-1 Constrained Channel Assignment in Single Radio Wireless Mesh Networks**

by

Ehsan Aryafar

This thesis addresses channel assignment and random medium access design for single-radio multi-channel mesh networks. Two prior approaches include: (i) designing MAC protocols that dynamically select channels based on local information and (ii) partitioning the mesh into subnetworks with different channels and using IEEE 802.11 as the medium access protocol. Both of these approaches suffer from limited throughput improvement; the first approach due to wrong or incomplete channel state information that inherently arises in a multi-hop wireless environment, while the second approach due to high interference within each subnetwork. In this thesis, I first introduce D1C-CA, Distance-1 Constrained Channel Assignment. D1C-CA statically assigns channels to a set of links as a function of physical connectivity, contention, and the unique gateway functionality of mesh networks, i.e, all internet (non-local) traffic has a gateway node as its source or destination. To design D1C-CA, I model

the channel assignment problem as a new form of graph edge coloring in which edges at distance one are constrained. I prove that the problem is NP-complete and design an efficient heuristic solution for mesh networks. Second, I design an asynchronous control-channel-based MAC protocol that solves multi-channel coordination problems and employs the proposed channel assignment algorithm. Finally, I investigate the performance of my approach through extensive simulations and show considerable performance improvements compared to alternate schemes.

## Acknowledgments

I would like to start by thanking my parents, Afssaneh and Majid, for being helpful throughout all these years, and their support in all aspects of my life. I am very much thankful to my advisor, Prof. Edward W. Knightly for his thoughtful comments and guidance during these two years of graduate study at Rice University. He has been a model researcher and a great example to learn from. I would also like to express my sincere appreciation to my other committee members, Prof. Rudolf H. Riedi and Prof. Ashutosh Sabharwal, for their insights and comments on my work. I am indebted greatly to both Alireza K. Haddad and Omer Gurewitz for their help throughout my entire thesis and for all their valuable input as my work developed. I am also thankful to my other friends that I had the chance of interacting and collaborating for the past two years. I can not name all of them; however, I have to mention a few: Jingpu, Tasos, Ahmed, Ahmad, Vincenzo, ....

# Contents

Abstract	ii
Acknowledgments	iv
List of Figures	viii
List of Tables	x
<b>1 Introduction</b>	<b>1</b>
<b>2 Background</b>	<b>7</b>
2.1 System Model . . . . .	7
2.2 Channel Assignment Problem Formulation . . . . .	9
2.2.1 Distance-1 Edge Coloring Problem . . . . .	10
2.2.2 Computational Complexity . . . . .	12
2.2.3 Upper Bounds on Distance-1 Chromatic index . . . . .	16
2.2.4 The case with Insufficient Number of Channels . . . . .	23
<b>3 Channel Assignment Protocol</b>	<b>26</b>
3.1 Overview . . . . .	26
3.2 Algorithm Description . . . . .	27
3.3 Protocol Performance Analysis . . . . .	28
<b>4 Common Channel Reference MAC (CCR-MAC)</b>	<b>35</b>

4.1	Multi-Channel Coordination Problems . . . . .	35
4.2	CCR-MAC . . . . .	42
4.2.1	Overview . . . . .	42
4.2.2	Protocol Rules . . . . .	44
4.3	Addressing Multi-Channel Coordination Problems . . . . .	48
<b>5</b>	<b>Performance Evaluation</b>	<b>51</b>
5.1	Simulation Setup . . . . .	51
5.2	Simulation Results . . . . .	53
5.2.1	Aggregate channel utilization . . . . .	54
5.2.2	Effect of traffic load . . . . .	56
5.2.3	Fairness among flows . . . . .	58
5.2.4	Effect of channel switching delay . . . . .	59
5.2.5	Topology Factors . . . . .	60
<b>6</b>	<b>Related Work</b>	<b>65</b>
6.1	Graph Theoretic Techniques . . . . .	65
6.2	Protocols to Exploit Frequency Diversity . . . . .	66
6.2.1	Single-Radio Protocols . . . . .	66
6.2.2	Multi-Radio Protocols . . . . .	67
<b>7</b>	<b>Conclusion</b>	<b>69</b>

References

## List of Figures

2.1	Two-tier mesh topology. . . . .	8
2.2	D1EC of an example mesh topology. . . . .	11
2.3	The graph $T$ . . . . .	13
2.4	Example Graph $H$ . . . . .	14
2.5	Construction of $G$ from $H$ . . . . .	14
2.6	$K_{D1EC}$ as a function with square degree of $\Delta$ . . . . .	17
2.7	D1EC of a geometric graph. . . . .	19
2.8	Channel assignment procedure for a tree. . . . .	21
2.9	Basic cell coloring of (a) hexagonal $G_3$ ; (b) square $G_4$ ; (c) triangular $G_6$ ; (d) octagonal $G_8$ grid topologies . . . . .	25
3.1	Packing of 39 circles of radius $\frac{R}{7}$ in a circle of radius $R$ . . . . .	31
4.1	Two flow topology . . . . .	37
4.2	The Multi-Channel Hidden Terminal problem . . . . .	37
4.3	The Missing Receiver problem . . . . .	38
4.4	Single three hop flow. . . . .	41
5.1	Topology . . . . .	52
5.2	Aggregate throughput comparison for TFA topology . . . . .	54

5.3	Aggregate throughput comparison for grid topology . . . . .	55
5.4	Total throughput variation with offered load . . . . .	57
5.5	Per flow throughput in TFA topology . . . . .	58
5.6	Impact of channel switching delay for grid topology . . . . .	60
5.7	Ternary tree: Tree with 3 levels and connectivity degree $m + 1 = 4$ . . . . .	61
5.8	Tree topology . . . . .	62
5.9	Relative gain of D1C-CA to AMCP . . . . .	63
5.10	Relative gain of D1C-CA to 802.11 . . . . .	64

## List of Tables

4.1	Mutual deafness and contaminated channel data base problems investigation in a 3 hop chain topology. Simulations are done for 1000 byte packets on a 2Mbps channel. All results are in terms of pkt/s. . . . .	41
5.1	MAC layer parameters . . . . .	51

# Chapter 1

## Introduction

Wireless mesh networks are a cost-effective last-mile access network, providing wireless Internet service to a large coverage area with low infrastructure cost [17]. As a result, many cities and ISPs plan to deploy two-tier mesh networks for city-wide Internet access [9]. A two-tier mesh network consists of an access tier, which provides connectivity to client devices, and a backhaul tier, which forwards traffic among mesh nodes to a wired Internet gateway. Use of multiple channels has the potential to increase aggregate throughput in such congested mesh networks. Existing approaches that utilize multiple available channels can be partitioned into two classes. The first class equips each Mesh Point (MP) with multiple transceivers and assigns each transceiver a different channel based on interference and traffic load [28, 29, 35, 26, 23, 34]. This approach has the advantage that it can be implemented without any modification to the IEEE 802.11 MAC; however, it requires multiple transceivers at each MP to utilize multiple channels and hence increases the cost of deployment.

The second class of solutions, which is considered in the 802.11s mesh standard [15], dynamically switches channels on the same transceiver [31, 33, 4, 15]. This approach utilizes multiple channels even with a single transceiver, thereby having lower deployment cost than the multi-transceiver approach. This class of solutions

relies on fast channel switching and modification of the 802.11 MAC.

In this thesis, I focus on the single-transceiver approach and investigate the following challenging problem: *Which of the available channels should a node use to transmit at any point in time?* The common approach to channel assignment is to let each transceiver dynamically select the channel for its own data transmissions based on local inference and control packets exchanged with neighbors: I refer to this class of schemes as Transceiver Based Channel Selection (TBCS). There are many variations of TBCS protocols. For example, TBCS protocols can be classified by the time period a channel is assigned to a transceiver, i.e., according to whether the channel selected is used for multiple-packet time scale, e.g., MMAC [33], or for single-packet time scale, e.g., AMCP [31] and DCA [38].

Unfortunately, TBCS protocols suffer from two inherent limitations that hinder their performance. First, they are vulnerable to inaccurate channel availability state: Since transceivers choose channels for transmission based on local inference of channel availability state, the performance of any TBCS protocol is dependent on the accuracy of the channel availability state. There are multiple reasons that cause transceivers under TBCS protocols to have inaccurate information regarding channel availability including (i) loss of reception of control packets when a transceiver is tuned to a different channel and (ii) corrupted reception of control messages due to low reception power, channel switching and collisions.

Second, TBCS protocols suffer from inconsistent neighborhood topology seen by neighboring nodes: Even accurate channel availability state does not guarantee high performance in TBCS protocols. In order to communicate, both the sender and receiver must agree on the availability of a channel based on their individual channel-availability state. Thus channel utilization will be high only if both transceivers (sender and receiver) agree on the availability of a specific channel when they want to communicate. However, since different transceivers have different views of the network topology and channel-availability state, coordination between them is imperfect. Note that the problem of different transceivers seeing different topologies is unique to multihop networks: in a single hop topology, each transceiver is in transmission range of all other transceivers.

In this thesis, I introduce a novel approach to channel assignment and medium access in single-radio multi-channel mesh networks. The key technique that I use is to employ static\* channel assignment that minimizes the number of interfering links with each active link, thereby significantly reducing contention. I exploit two properties of mesh networks to achieve this: First, the gateway node acts as a centralization point to compute the channel assignment based on the gateway's knowledge of topology.† Second, because all traffic originated from or destined for the Internet traverses a

---

\*In practice, my proposed approach is more precisely quasi-static, with the assignment recomputed under major network changes such as node failure.

†In fact, in Cisco's Aironet architecture, *all* network management functions are performed at gateway nodes.

gateway, links closest to the gateway are the most congested. Consequently, my channel assignment algorithm favors these links. In addition to channel assignment, I present a MAC protocol that transmits data on channels determined by the channel assignment scheme while employing a single transceiver, asynchronous random access, and a common control channel for medium arbitration.

In particular, I make the following contributions in this thesis. First, I present Distance-1 Edge Coloring (D1EC), a new form of graph coloring. If a graph has a D1EC, it has a sufficient number of channels such that the assignment is contention free, i.e., all data links that have the potential to be active simultaneously (because they do not share a common node) can be active simultaneously. I show that for arbitrary graphs, the D1EC problem is NP complete when the number of channels is three or more. Moreover, for structured graphs (geometric and grid), I bound the distance-1 chromatic index as a function of the node degree, thereby providing the minimum number of channels required to have a D1EC in these graphs. I show that many such structures require a small number of channels for maximum throughput that is feasible in standards such as IEEE 802.11a.

Second, I develop the Distance-1 Constrained Channel Assignment (D1C-CA) algorithm with the objective of minimizing interference among active links in order to increase the network's aggregate throughput. Thus, if a valid D1EC exists, the outcome of the algorithm is a heuristic mechanism to find it. If a valid D1EC does not

exist, the algorithm minimizes the number of interfering links per channel. In other words, with D1C-CA, some links will be assigned an interference free channel which will allow them to transmit whenever the transmitter-receiver pair is available. The remaining links will be given information regarding the interference of the channel assigned to them. This information will help them contend for the channel more efficiently.

Third, I design an asynchronous single-radio multi-channel MAC protocol, termed Common Channel Reference MAC (CCR-MAC). The objective is not MAC protocol design itself, but rather to demonstrate an example protocol that employs the aforementioned channel assignment. The key technique is to use a common control channel to arbitrate contention and coordinate sender-receiver pairs, and to transmit data on the channels assigned by D1C-CA. By employing a separate control channel, CCR-MAC solves the multi-channel coordination and multi-channel hidden terminal problems.

Finally, I corroborate my theoretical results via an extensive set of ns-2 simulations. I show that my proposed scheme yields significant improvement in both aggregate and per-flow throughput as compared to both IEEE 802.11 and a recently proposed multiple channel MAC protocol with TBCS approach, irrespective of the number of channels.

The remainder of this thesis is organized as follows. In chapter 2, I present the net-

work model and formulate the channel assignment problem as a new form of graph edge coloring. Several interesting properties of this problem, including complexity analysis and upper bounds on the needed number of channels are discussed in this chapter. An efficient channel assignment algorithm that achieves the channel assignment objectives and its performance analysis are described in chapter 3. I introduce the multi-channel coordination problems and an example random access MAC protocol that works in collaboration with the channel assignment algorithm in chapter 4. Ns-2 simulation results are presented in chapter 5. I contrast my work with existing literature in chapter 6, and finally conclude the thesis in chapter 7.

## Chapter 2

# Background

In this chapter, I first define two-tier wireless mesh networks and describe the system model and assumptions used throughout this thesis. I next introduce the objectives for the channel assignment problem and define the distance-1 edge coloring problem. Several theoretical results for this problem are provided in this chapter.

### 2.1 System Model

I assume a two-tier wireless mesh network, which consists of a backhaul tier for connection between mesh nodes and an access tier for connection between mesh nodes and client devices such as wireless bridges or laptops. An example of such architecture is depicted in figure 2.1. The mesh nodes are stationary wireless devices which use a *single half-duplex transceiver* for back-haul access and an independent wireless access technology for client access. A subset of mesh nodes feature a wired connection to the Internet and are referred to as wired gateway nodes. I assume that there are  $K$  non-overlapping frequency channels in the system and each node can listen, receive or transmit to only one of these channels at a time. All nodes in the network use the same fixed transmission power, i.e, there is a fixed transmission range ( $r \geq 0$ ) and a fixed interference range ( $R \geq r$ ) associated with every node. The physical graph of the network is modeled as an *undirected* graph  $G = (V, E)$ . (This assumption is

used to ensure that both data and acknowledgement packets are feasible on the same link.) Here,  $V$  is a set of vertices denoting the transceivers comprising the wireless network and  $E$  is a set of undirected edges between vertices representing inter-node link characteristics. There is an undirected edge  $(v_i, v_j) \in E$  connecting vertices  $v_i$  and  $v_j$  iff  $\|v_i - v_j\| \leq R$ .

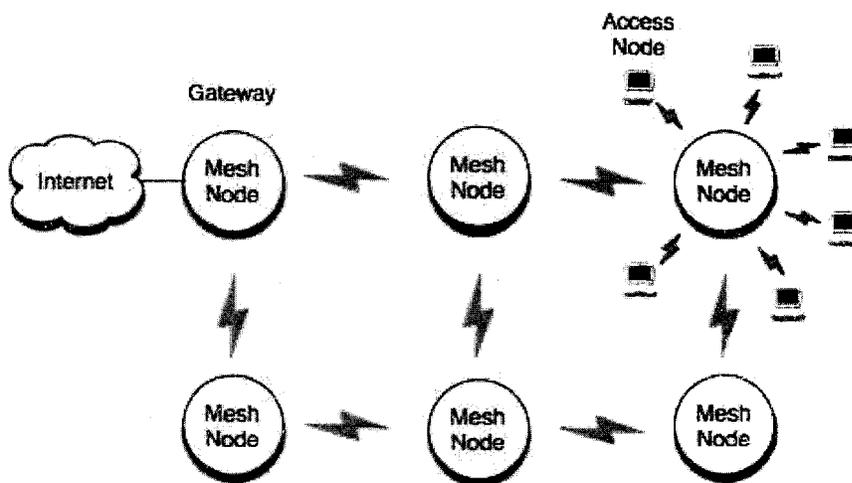


Figure 2.1 Two-tier mesh topology.

At least one of the routers within the mesh is designated as a *gateway* and has information about the mesh network's physical connectivity.

I also assume a separate routing protocol that determines the routes to and from the gateways. I refer to the links that are selected by the routing protocol to forward traffic as *active* links and denote the set of all active links by  $A$ . Note that an active link *does not always have a packet to forward, but that it is present in the routing table of a node for packet forwarding.*

## 2.2 Channel Assignment Problem Formulation

The objective of channel assignment, is to increase network capacity by allowing concurrent transmission of links that would not be able to be active at the same time with the same channel. In other words, assigning different channels to any two active links that can be active at the same time *iff* their transmissions occur on two different channels. I denote a channel assignment that can realize this objective as a valid “distance one channel assignment.” This means that a distance one channel assignment should assign different channels to any two active links that do not share a mutual transceiver but do interfere with each other. In other words, if at least one transceiver is within interference range of a transceiver from other link, the two links should be assigned different channels. Note that two cases need not be considered in D1EC channel assignment. First, non-interfering links need not be considered because they can *always* be active simultaneously even if assigned the same channel. Second, links that share a mutual transceiver should also not be considered because they can *never* transmit simultaneously due to the single transceiver and half duplex system.

On the other hand, due to the limited number of channels, there might not exist a channel assignment that can realize a valid distance one channel assignment. In this case, the objective of my channel assignment is to minimize the number of interfering links with each active link, i.e., to spread the load between channels in a way that

each active link will have as few interfering links on the same channel as possible. With this background on channel assignment objectives, I next formally define them.

### 2.2.1 Distance-1 Edge Coloring Problem

Let the distance between two nodes  $u_1$  and  $u_2$  in a graph  $G$ , denoted by  $d(u_1, u_2)_G$ , be the minimum number of hops in  $G$  from  $u_1$  to  $u_2$ . Accordingly, I define the distance between two links  $l_{u_1u_2}$  and  $l_{v_1v_2}$  as:

$$d(l_{u_1u_2}, l_{v_1v_2})_G = \min (d(u_i, v_j)_G) \quad i, j \in \{1, 2\} \quad (2.1)$$

Note that distance zero between two links defines two links that share a mutual node; distance one between two links defines two links that are within interference range of each other and do not share a mutual node; and distance greater than one between two links defines two links that are out of interference range of each other. Note that according to the channel allocation objective defined above, only links that are at distance one should be assigned different channels. By equating channels to colors, I model this channel assignment problem as a graph edge coloring problem.

*Definition 1 : (D1EC problem) Given a physical graph  $G$  and a selected subgraph  $A \in G$ , the **Distance-1 Edge Coloring (D1EC)** problem seeks a mapping of colors to links in  $A$  such that any two links in  $A$  that are at distance one with respect to  $G$  are assigned different colors.*

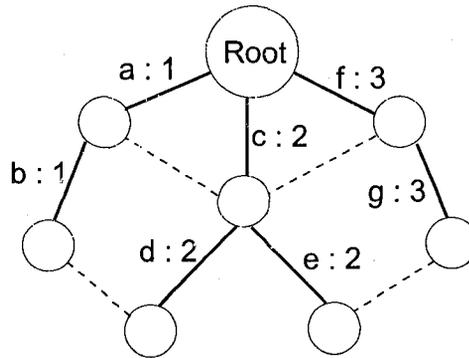
Note that the D1EC problem is a variation of the classical edge coloring problem

where edges at distance zero have to be assigned different channels, a problem that is known to be NP-complete [7].

Observe that if a valid D1EC is realized, any selected set of links in  $A$  that do not share a node in common can be active at the same time and hence *maximum throughput* is achieved.

Next I define the *Distance-1 chromatic index* that describes the number of colors needed to have a valid D1EC.

*Definition 2:* The **distance-1 chromatic index**,  $k_{D1EC}$ , of a subgraph  $A \in G$ , is the minimum number of colors to have a valid D1EC of links in  $A$ .



**Figure 2.2** D1EC of an example mesh topology.

Figure 2.2 shows an example of D1EC. Here, thick edges represent the set of active links, i.e.,  $A \in G$ . The colors assigned to links:  $\{a, b, c, d, e, f, g\}$  are  $\{1, 1, 2, 2, 2, 3, 3\}$ , respectively. Note that any set of links that do not share a node in common can be active at the same time *iff* at least three channels are available, e.g.,  $\{a, e, g\}$ . On

the other hand, if the number of available channels is smaller than three the D1EC problem cannot be solved, which means that the Distance-1 chromatic index is 3.

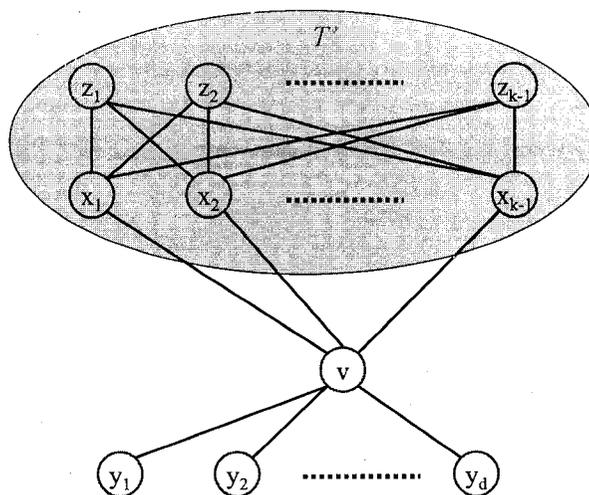
### 2.2.2 Computational Complexity

A key problem in channel assignment is to find the Distance-1 chromatic index, i.e., what is the minimum number of channels needed for a valid distance-1 channel assignment of a given topology. However, in the next theorem I prove that finding the Distance-1 chromatic index is an NP-complete problem. Later I will provide upper bounds to the *Distance-1 chromatic index* problem.

**Theorem 1** The decision problem whether  $k$  colors are sufficient to have a valid D1EC is NP-complete for  $k \geq 3$ .

**Proof of Theorem 1:** It is clear that the problem is in NP. I prove that it is NP-complete by showing a reduction from *GRAPH K-Colorability* to this problem. Without loss of generality I assume  $A$  is equal to  $G$ . It is proved [12] that *GRAPH K-Colorability* is NP-complete for every fixed  $k \geq 3$ . For every graph  $H$ , we construct another graph  $G$  such that  $H$  is  $K$ -colorable if and only if  $G$  has a distance-1 edge coloring using  $K$  colors.

Before starting to construct  $G$ , consider the graph  $T_{K,d}$  shown in figure 2.3. This graph consists of a vertex  $v$  adjacent to a set of  $K + d - 1$  other vertices. The vertices  $y_1, \dots, y_d$  connected to  $v$  are called the heads of  $T_{K,d}$ . The rest of vertices denoted by



**Figure 2.3** The graph T

$x_i$  are respectively connected to another set of  $K - 1$  vertices denoted by  $z_j$ . The important property of this graph, which will be used throughout the proof, is based on the following lemma:

**Lemma 1** *In any distance-1 edge coloring of  $T_{K,d}$  (figure 2.3) with  $k$  colors, the color of all  $vy_j$  edges is the same.*

**Proof:** First, I prove that this graph is  $K$  colorable. In order to do this, I color the whole graph by giving  $K$  different colors to all  $x_i$  and  $v$ . Next I color all the edges connected to each  $x_i$  by giving them the color given to  $x_i$ . The remaining edges are  $vy_j$  edges which can be colored by giving them the color used for  $v$ . Now I prove that in any other distance one coloring of this graph with  $K$  colors, the colors used by all  $vy_j$  edges should be the same. First, consider the subgraph of  $T_{K,d}$  ( $T'$ ) composed of

nodes  $x_i$  and  $z_j$ , and all the edges between them. It is easy to see that this graph contains a set of  $K - 1$  edges which all are mutually at distance one from each other. Hence in any distance one coloring of  $T_{K,d}$  using  $k$  colors, this subgraph should be colored with  $K - 1$  different colors. Now, since  $T_{K,d}$  is  $K$  colorable and each  $vy_j$  edge is at distance one from all edges in  $T'$ , only one color remains for all  $vy_j$  edges.  $\square$

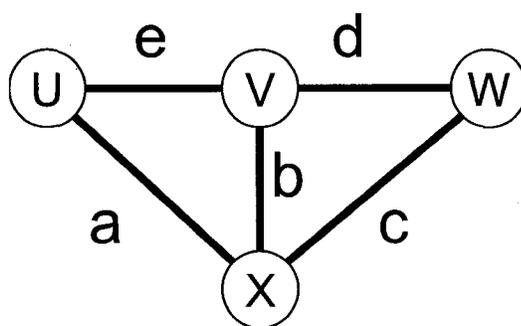


Figure 2.4 Example Graph  $H$ .

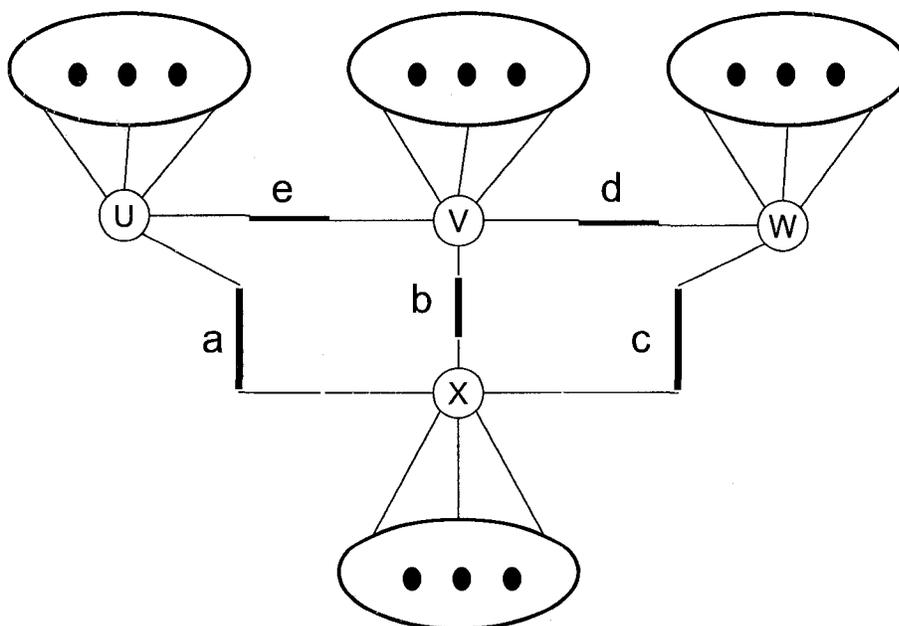


Figure 2.5 Construction of  $G$  from  $H$ .

Now, we are ready to construct the graph  $G$  from the graph  $H$ . Corresponding to each vertex  $v$  of degree  $d$  in  $H$ , we put a copy  $C_v$  of  $T_{K,d}$  in  $G$ . Each head of  $C_v$  corresponds to one of the edges incident to  $v$ . If two vertices  $u$  and  $v$  in  $H$  are joined by an edge  $e$ , their corresponding heads in  $C_u$  and  $C_v$  are connected through  $e$ , in the resulting graph  $G$ . This procedure is illustrated in figures 2.4 and 2.5. We claim that  $G$  has distance-1 edge coloring with  $K$  colors if and only if  $H$  is  $K$  colorable.

Assume that  $G$  has a distance-1 edge coloring with  $K$  colors. By the property of  $T_{K,d}$ , we know that for every vertex  $v \in V(H)$  the color of all  $vy_j$  edges in  $C_v$  is the same. Color the vertex  $v$  in  $H$  with the color of  $vy_1$ . Since for any two adjacent vertices  $u$  and  $v$  in  $H$ , exists edges  $uy_i$  and  $vy_j$  that are at distance one from each other (through an edge  $e$  in  $H$ ), the color of  $uy_1$  and  $vy_1$  in  $C_u$  and  $C_v$  is not the same and therefore the color of the vertices  $u$  and  $v$ , can not be the same. Thus, the coloring is a proper vertex coloring of  $H$ .

Conversely, assume that  $H$  has a proper vertex coloring using  $K$  colors. I construct a distance-1 edge coloring of  $G$  using  $K$  colors. For every  $v$ , I color all the edges connected to  $v$  of  $C_v$  in  $G$  with the color of  $v$  in  $H$ . These set of edges include  $vy_j$  and  $vx_i$ . The remaining edges of graph  $G$  are divided to two sets. The first set includes The edges in  $T'$  associated with each  $C_v$ . These edges can be colored with  $k - 1$  colors as I showed in lemma 1. The remaining set of edges includes original edges in graph  $H$  between any two vertices  $u$  and  $v \in H$  that are still present in  $G$  through

corresponding heads in  $C_u$  and  $C_v$ . It is easy to check that for any such sample edge  $e$  between two heads  $C_u$  and  $C_v$ , the set of edges that are at distance-1 is a subset of  $uy_i$  and  $vy_j$ . Since all these edges are colored with 2 colors, there are  $k - 2$  colors yet available which I randomly assign one to each such  $e$ .  $\square$

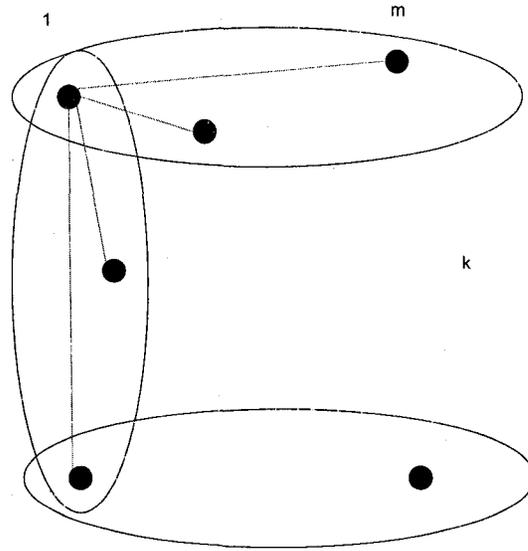
### 2.2.3 Upper Bounds on Distance-1 Chromatic index

I now provide upper bounds on  $K_{D1EC}$  to determine the needed number of channels to guarantee a solution for D1EC problem for specific topology structures. I assume the selected subgraph  $A$  is equal to  $G$ . In a real network, only a subset of links are used for traffic forwarding and need to be assigned channels such that  $A$  is not equal to  $G$  and the needed number of channels can only be smaller. Using the Brook and Vizing theorem [7], the following bounds on  $k_{D1EC}$  can be derived for any graph with *maximum degree*  $\Delta$ :

$$k_{D1EC} \leq \min\{|V|, 2 \times (\Delta - 1)^2 + 1\} \quad (2.2)$$

The above theorem gives a bound on the chromatic number that is of square degree of  $\Delta$ . Before moving on bounds that are appropriate for ad hoc network graphs (e.g., geometric graphs), I give the following theorem that shows the chromatic number can grow as a square function of  $\Delta$  for general network topologies.

**Theorem 2** *The distance-1 chromatic index of arbitrary graphs can be lower bounded*



**Figure 2.6**  $K_{D1EC}$  as a function with square degree of  $\Delta$ .

by a function with square degree of  $\Delta$ .

**Proof of Theorem 2:** I prove the above theorem by giving an example of a graph where the chromatic number is lower bounded by a function which is square degree of  $\Delta$ . This graph is shown in figure 2.6.

The graph is constructed as  $k$  parallel cliques with  $m$  nodes, where all nodes  $i$  in each of these cliques are also connected together and form  $m$  parallel vertical cliques with  $k$  nodes. In order to find a lower bound on the chromatic number, we check the maximum number that a color can be used. For simplicity, let's assume that  $m \geq k$ . Consider the link  $l_{ij}$  between nodes  $i$  and  $j$  in clique 1, and assume it's given color  $c$ . Since in a clique, every two distinct sender receiver are at distance one, hence color  $c$  can not be reused to color a distinct pair of nodes in clique 1. By considering different scenarios in which a color can be reused, it's easy to check that the maximum number

of links that can be colored with  $c$  happens when all the edges connected to a node have the same color. So  $c$  can be used to color at most  $m + k - 2$  distinct edges. A lower bound on the minimum number of needed colors is then given by dividing total number of edges to  $m+k-2$ , which is:

$$\chi \geq \frac{\binom{k}{2} \times m + \binom{m}{2} \times k}{m + k - 2} \quad (2.3)$$

By assuming  $m=k$ , this gives the following lower bound which is indeed of square degree of  $\Delta$ .

$$\chi \geq \frac{(\Delta + 2)^2}{4} \quad (2.4)$$

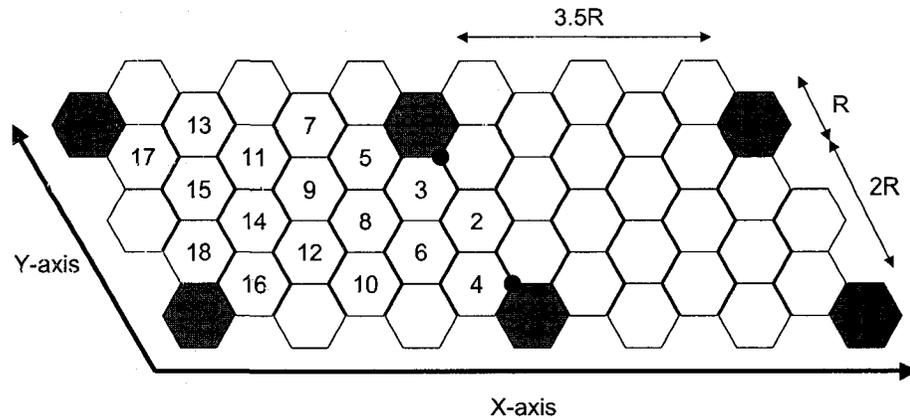
This ends the proof  $\square$ .

The above inequality gives a bound on  $k_{D1EC}$  that is of square degree of  $\Delta$ . However, geometric properties of wireless networks can be used to provide linear bounds.

Our next theorem provides an upper bound on  $k_{D1EC}$  for the Unit Disk Graph (UDG) model [22] and random placement of nodes. In the UDG model, all nodes have the same transmission and interference range and this value is the same for all nodes.

**Theorem 3**  $K_{D1EC}$  for a geometric graph with maximum degree  $\Delta$  is upper bounded by  $18 \times (\Delta + 1)$ .

**Proof of Theorem 3:** My proof is based on division of the physical graph into hexagonal cells of diameter  $R$  (figure 2.7), where  $R$  is the transmission range of nodes in the network.



**Figure 2.7** D1EC of a geometric graph.

Within each cell of diameter  $R$ , at most  $\Delta + 1$  nodes can exist. This is because of the fact that all nodes inside a cell are connected and maximum degree of graph is  $\Delta$ . Now I propose an algorithm for an arbitrary node placement, which uses number of channels equal to the upper bound. My algorithm is based on a coloring of *nodes* in the physical graph. Assume a coloring to each node is given. I color the edges among different nodes based on the following rules: a) For an edge inside a cell, randomly assign the color of one of its end point nodes. b) For edges among nodes in different cells, give the color of the node with higher y-axis coordinate value. For the resulting edge coloring to be valid, the original node assignment should have the following properties: a) For each cell, a pool of  $\Delta + 1$  different colors is available where nodes

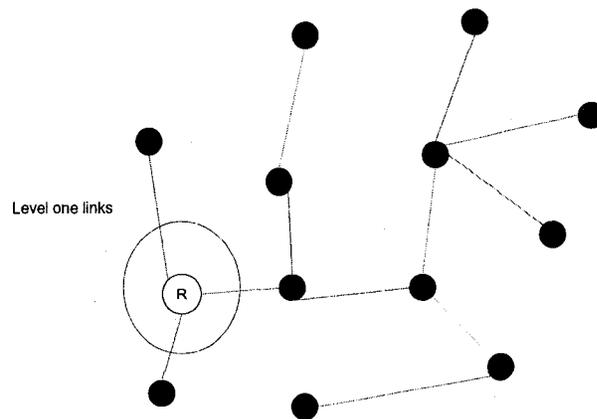
inside a cell are assigned one randomly. b) For each cell, its pool of colors is reused at cells which are apart for the distance of  $3.5 \times R$  and their centers are parallel to x-axis or are at the distance of  $2R$  and their centers are parallel to y-axis. c) For two cells which have the same channel reuse and are apart for distance of  $2R$ , if there exists nodes exactly at the corner of cells (figure 2.7 shows such example), these nodes should be assigned different colors from the pool of  $\Delta + 1$  colors (I assume  $\Delta \geq 2$  for this argument be valid. Otherwise, equation 2.2 gives a smaller upper bound). This channel reuse is depicted through colored cells in figure 2.7. It is easy to check that with such node assignment, the resultant coloring is a valid D1EC. I have shown the number of cells in figure 2.7 that will use different channel pools. From this figure it is evident that such coloring algorithm needs at most  $18 \times (\Delta + 1)$  colors for a valid distance-1 edge coloring.  $\square$

Since this thesis is mainly concerned with mesh networks where nodes are not placed randomly over space, I next consider structured node placements and upper bounds on  $k_{D1EC}$  for these topologies. I start by deriving the following upper bound on  $K_{D1EC}$  for tree topologies:

**Theorem 4** *The distance-1 chromatic index of a tree is less than or equal to three.*

**Proof of Theorem 4** In order to find a bound on trees, all we need to do is to give a coloring algorithm and show its correctness. I start by choosing a random node in the tree as the root (parent) and call all the edges connected to it as level one links

and give them color one. Next I consider the nodes connected to the other end of these links (children) and call the links connected to them as level two links and give them all color 2. I then consider the links connected to level two links and call them level 3 links and give them color 3. I proceed with the same procedure to give levels to links and start reusing the colors. More precisely, I give color  $i = \{1, 2, 3\}$  to links belonging to level  $j$ , if  $(j \bmod 3) = i$ . This procedure is shown in figure 2.8.



**Figure 2.8** Channel assignment procedure for a tree.

To check the correctness of the proof, I use the following results based on the definition of edge distances I gave before:

if  $e_u, e_v \in level_i$  then  $d(e_u, e_v) = 2k$  for  $k = 0, \dots$

if  $e_u \in level_i$  and  $e_v \in level_{i+1}$  then  $d(e_u, e_v) = 0$  or  $2k + 1$  for  $k = 0, \dots$

if  $e_u \in level_i$  and  $e_v \in level_{i+2}$  then  $d(e_u, e_v) = 1$  or  $2(k + 1)$  for  $k = 0, \dots$

if  $e_u \in level_i$  and  $e_v \in level_j$  and  $j \geq i + 3$ , then  $d(e_u, e_v) \geq 2$

From equation 1, the edges belonging to the same level can be assigned the same

color and from equations 2 and 3 the edges belonging to levels  $i, j$  such that  $1 \geq |i - j| \leq 2$  can be colored with two additional colors. From equation 4, we see that color given to level  $i$ , can be reused at level  $i + 3$  and this finishes the correctness of the algorithm.  $\square$

I next consider regular grid topologies. Let  $G_\Delta, \Delta = 3, 4, 6, 8$  denote the hexagonal, squared, triangular and octagonal grids respectively. Portions of these grids are shown in Figure 2.2.3. The next theorem provides an upper bound on  $k_{D1EC}$ , for these regular grid topologies:

**Theorem 5**  $K_{D1EC}$  of regular grid topologies  $G_\Delta, \Delta = 3, 4, 6, 8$ , is sequentially upper bounded by 3,4,7,10.

**Proof of Theorem 5:** The proof is based on two steps: First, I construct a basic cell topology, such that its replication generates the infinite topology. Second, I find an edge coloring for the cell with the following properties: a) the edge coloring is a valid D1EC, b) permits replication by matching the boundary colors, and c) the resultant edge coloring after replication is a valid D1EC. The basic cells with the above properties for hexagonal, square, triangular and octagonal grids are plotted in figure 2.2.3. From this figure the upper bounds are achieved.  $\square$

#### 2.2.4 The case with Insufficient Number of Channels

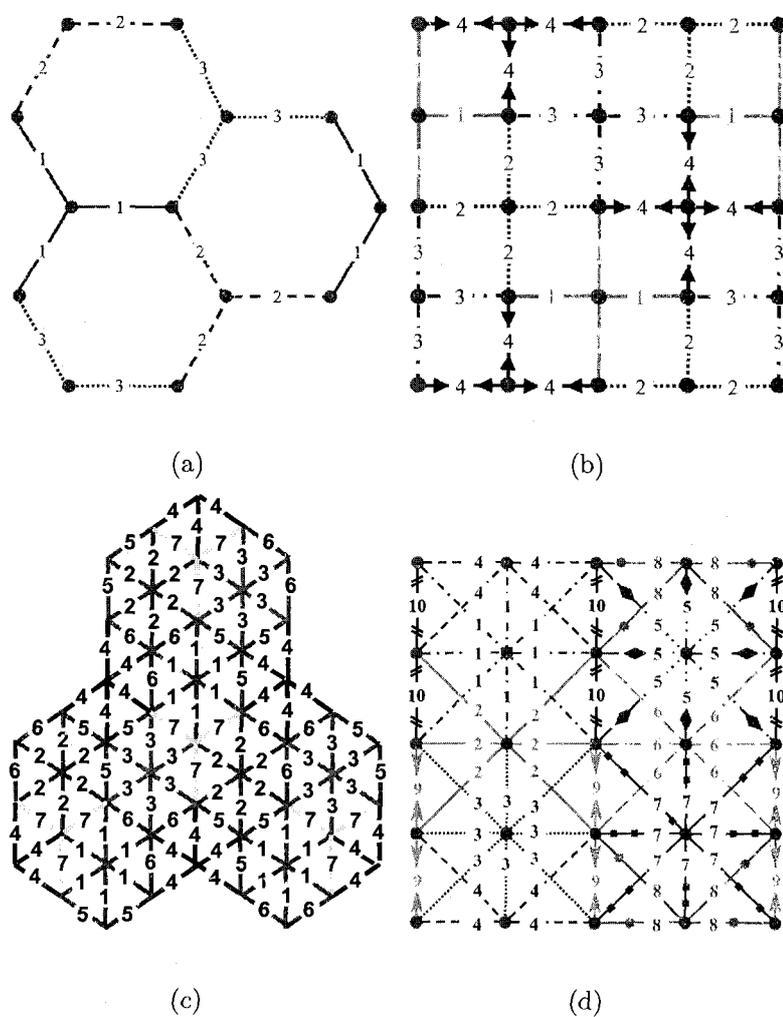
While Theorem 5 guarantees that the number of channels available in standards such as IEEE 802.11a is sufficient to have a valid D1EC for many grid topologies, Theorem 3 shows that in a random deployment of nodes, the needed number of channels can become very large and hence there may not be enough channels for the D1EC problem to have a solution. Only in this case, the channel assignment algorithm should allow two links at distance one to be assigned the same channel. For a colored link, the set of links that have the same channel and are at distance one will contend for channel access and hence will have to share the bandwidth. We define the set of links that will contend for channel access with a specific link  $e$  as follows.

*Definition 3: Suppose  $A$  is a subset of the network graph  $G$ , and a channel assignment  $C'$  to the links of  $A$  is given. The contention degree of a link  $e \in A$ ,  $Co(e)$ , is the **maximum cardinality matching** of a set  $M$  with the following properties:  $M$  is a subgraph of  $A$  containing  $e$  and the following set  $\{l \in A \mid Color(l) = Color(e), d(l, e)_G = 1\}$ .*

$Co(e)$  is the maximum number of contenders a link can have at a given time and on the same channel. Hence the bandwidth provided for a link is dependent on the objective of channel assignment. For example a channel assignment scheme that minimizes the sum of contention degrees can result in a high total throughput

but starvation of some links, while a channel assignment scheme that minimizes the maximum contention degree has better fairness properties at the cost of lower total throughput. Thus, I target the following channel assignment objectives:

- *Solving the D1EC problem:* If a valid D1EC is achieved, each link will have a channel without a need to share its bandwidth such that the maximum possible throughput is achieved. On the other hand as I proved in Theorem 1, the D1EC problem is NP-complete so that we must rely on a heuristic solution.
- *Providing low maximum contention degree after channel assignment:* If the number of channels is not sufficient to solve the D1EC problem, some links which are at distance one have to be assigned the same channel and hence the bandwidth is shared among them. Note that because mesh network traffic aggregates at the gateway, links closer to the gateway should have higher channel assignment priority. Hence it is desirable for channel assignment to have zero contention degree for links connected to the gateway and low maximum contention for other links.



**Figure 2.9** Basic cell coloring of (a) hexagonal  $G_3$ ; (b) square  $G_4$ ; (c) triangular  $G_6$ ; (d) octagonal  $G_8$  grid topologies

## Chapter 3

# Channel Assignment Protocol

In this chapter, I first introduce my proposed architecture that is implemented by all nodes in the network. I next describe the details of my proposed channel assignment algorithm.

### 3.1 Overview

The complete channel assignment protocol comprises three separate mechanisms. In the first mechanism the Network Control Center (NCC), which is co-located with one of the gateways and is responsible for channel assignment, constructs the physical graph  $G$  and the set  $A$  comprise of the active links which are expected to forward data. *This procedure is run only once during network setup and is updated based on deployment of new nodes or node failures.*

The second mechanism is the D1C-CA algorithm, described in this chapter. In particular, based on the physical topology and the forwarding topology, the NCC runs the D1C-CA algorithm which is the key component of the protocol. According to D1C-CA, the NCC allocates channels to links in  $A$  in a way that satisfies the objectives described in chapter 2.2. After completing the algorithm, the NCC distributes to each node a vector of entries, one entry for each active link connected to it. Each entry is comprised from two elements: the channels assigned to the link and the number

of links in its interference neighborhood that are assigned the same channel, i.e., the contention degree of the link.

The third mechanism is medium access: As the channel assignment algorithm assigns different channels to links connected to a node, a mechanism is needed to coordinate between each sender and receiver to schedule their transmission. This is the function of the MAC protocol described in chapter 4.

### 3.2 Algorithm Description

Once the NCC gathers information about the physical graph  $G$  and its selected forwarding subgraph  $A$ , it uses the DIC-CA algorithm to assign channels to links in  $A$ . This procedure is summarized in the psuedocode of Algorithm 2.

The algorithm assumes that there are  $N$  gateways present in the mesh (Line 1) and uses hop count metric between nodes and nearest gateway to visit the nodes and assign colors to links connected to them. In each iteration, nodes that are at a specific distance are selected (Line 10), assigned labels (Line 11) and finally edges connected to them are colored (Line 12).

In my coloring procedure, I select the node with highest label (Line 2) and first try to assign the same color to all of the links connected to it (Line 3). If such color does not exist, I randomly select one of uncolored edges connected to it,  $l$  (Line 7). If a valid color is found, then the color with lowest index is assigned to  $l$ . If a valid color is not found, I *reserve the colors of links connected to gateways* by eliminating

channels selected by them from the set of available channels for  $l$  (Line 10-12). Next, a color with minimum contention degree is greedily assigned to  $l$  (Line 13-15).

As this assignment has impact on the contention degree of links that are at distance one from  $l$  and have the same color as  $l$ , their contention degree is updated by using Edmond's algorithm which finds maximum cardinality matching for each affected link [7].

### 3.3 Protocol Performance Analysis

I now capture the performance of my proposed heuristic in achieving the objectives defined in section 2.2, i.e., worst cast ratio between the required number of channels to have a valid D1EC and the number of channels used by our heuristic and minimizing the maximum contention degree if the number of channels is not sufficient to have such assignment. I assume the UDG model and that the forwarding subgraph  $A$ , is equal to  $G$ .

**Theorem 6** *Algorithm D1C-CA needs at most  $C_1 \cdot OPT(K_{D1EC})$  channels to have a valid D1EC. Furthermore, if  $k$  is the number of available channels, the ratio between maximum contention degree after the channel assignment and the optimum value is bounded by  $24k$  as  $\Delta \rightarrow \infty$ .*

**Proof of Theorem 6:** The proof is based on the following lemma on the distance-1 chromatic index of a clique of size  $\Delta$ :

**Lemma 2** For a clique of maximum degree  $\Delta$  ( $\Delta \geq 2$ ),  $K_{D1EC}$  is equal to  $\Delta - 1$ .

**Proof** For  $\Delta = 2$  one color is sufficient. Hence I assume  $\Delta \geq 3$ . For an uncolored clique,  $G$ , a color can be *most* used if all edges connected to a random node are colored the same. Choose a random node and color all edges connected to it with the least available color. Now, eliminate this node and all edges connected to it. For the resulting graph, if  $\Delta = 2$ , all remained edges can be colored with another color, otherwise repeat the procedure. Since at each step the maximum possible number of edges are colored, the algorithm colors the clique optimally. Since this algorithm repeats until a triangle is reached,  $\Delta - 1$  colors are used.  $\square$

Now, as a UDG of degree  $\Delta$  contains a clique of size  $\lceil \frac{\Delta}{6} + 1 \rceil$ , [22], we have the following lower bound on  $OPT(K_{D1EC})$ :

$$OPT(K_{D1EC}) \geq \lceil \frac{\Delta}{6} \rceil \tag{3.1}$$

On the other hand, there are at most  $2 \times (\Delta - 1)^2 + 1$  links at distance one of a specific link. Hence the ratio of any assignment algorithm in finding a valid distance-1 channel assignment is  $O(\Delta)$ .

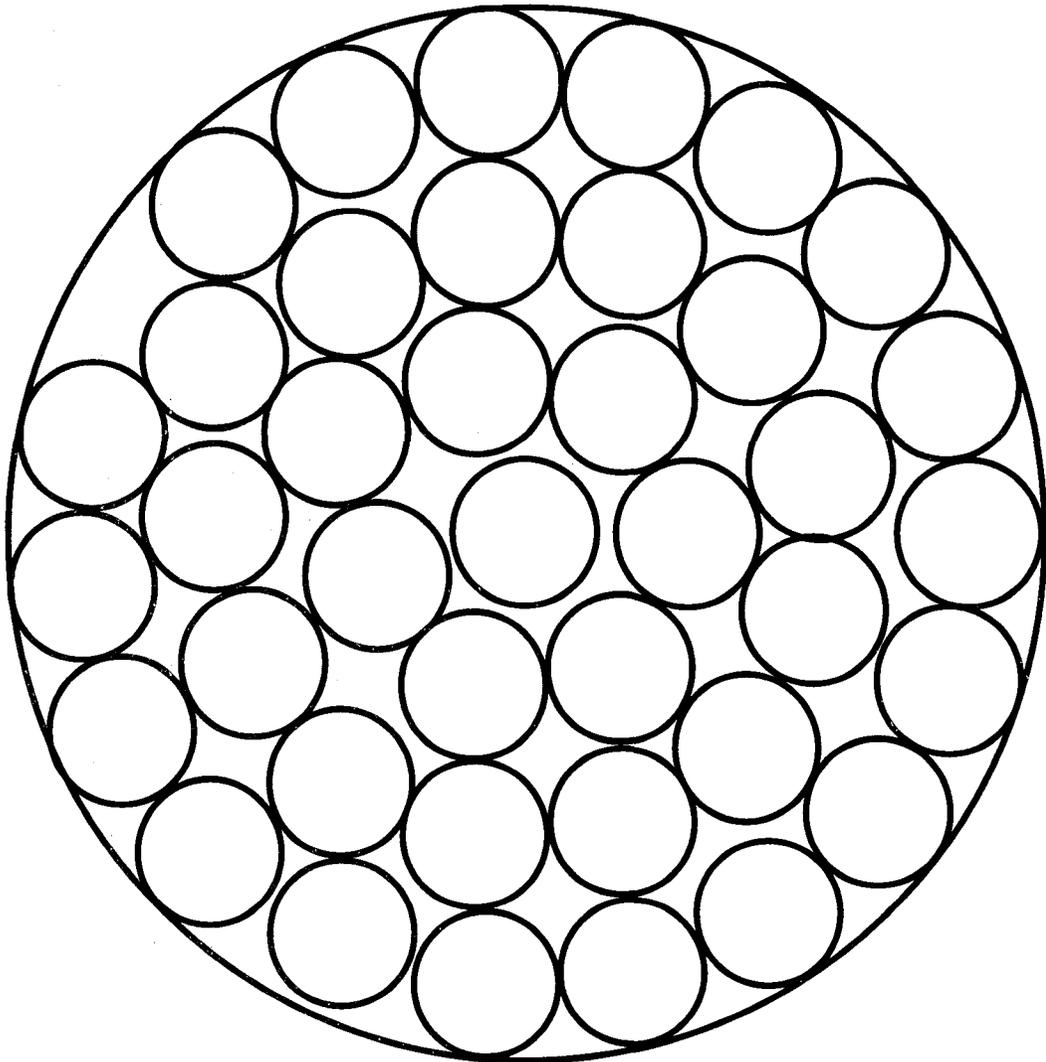
However, certain properties of the D1C-CA algorithm provides better upper bounds. The main property that I use in the proof of performance approximation, is based on visiting the nodes instead of edges. More precisely, since D1C-CA algorithm visits nodes in a given order and *greedily assigns first color that is possible to*

*assign to all edges connect to it*, we have the following property:

Suppose that a node  $u$  is randomly selected and we want to color all uncolored edges connected to it. In order to have a valid distance-1 channel assignment, no other edge at distance-1 should have the same color. This proposition will be true *if* no other node at distance  $3R$  from  $u$  has used the same color. So my greedy algorithm will not use more colors than the maximum number of nodes that can be present in a circle of radius  $3R$ .

So, we need to find an upper bound on the maximum number of nodes that can be present in a circle of radius  $3R$  with the constraint that the model is a Unit Disk Graph and maximum node degree is  $\Delta$ . To find a tight upper bound for this problem, I use the maximal independent sets. An independent set is a set of vertices in a graph  $G$ , such that no two vertices of which are adjacent. A maximal independent set is an independent set such that adding any other node to the set forces the set to contain an edge. As a result the maximum number of nodes is equal to the size of maximal independent set times  $\Delta + 1$ . Since in the geometric graph any two node in the maximal independent set are more than  $R$  apart from each other, the problem is equivalent to packing circles of radius  $\frac{R}{2}$  in the big circle of radius  $3R$ . however since the center of these small circles can be on the boundary of our big circle, an upper bound can be achieved by increasing the radius of the original circle to  $3.5R$  and packing circles of radius  $\frac{R}{2}$  in it. This is a well known problem and is addressed

in [13]. I have drawn this result in figure 3.1. From [13], we know that this ratio is equal to 39. Hence the number of colors used by my algorithm is upper bounded by  $39 \times (\Delta + 1)$ . With this linear bound we get the constant approximation ratio.



**Figure 3.1** Packing of 39 circles of radius  $\frac{R}{7}$  in a circle of radius  $R$ .

Next, I proceed to calculate  $\max (Co(e)) \forall e \in G$  after the coloring algorithm. For any given link  $e \in G$ , there are at most  $2 \times \Delta - 2$  neighbor vertices connected

to either end of the link  $e$ .  $Co(e)$  is defined as a matching  $M$  containing  $e$ , where all members of  $M$  are at distance one from  $e$  and have the same color as  $e$ . This set is a subset of links connected to the above  $2 \times \Delta - 2$  vertices. Since at most one link connected to the above nodes can be member of  $M$  (because of matching property), hence the worse case scenario is that if all these members have the same color as  $e$ . Consequently:

$$\max(Co(e)) \leq 2 \times \Delta - 2 \quad (3.2)$$

On the other hand, since for a clique of size  $\omega$  there exists a matching of size  $\lfloor \frac{\omega}{2} \rfloor$ , we have at least a matching of size  $\lfloor \frac{\lceil \frac{\Delta+1}{2} \rceil}{2} \rfloor$ . As a result, with  $k$  channels the contention degree is lower bounded by:

$$OPT(\max(Co)) \geq \lceil \frac{\lfloor \frac{\lceil \frac{\Delta+1}{2} \rceil}{2} \rfloor}{k} \rceil - 1 \quad (3.3)$$

This proves that with sufficiently large  $\Delta$ , the ratio is upper bounded by a constant factor equal to  $24k$ .  $\square$

---

**Algorithm 1** D1C-CA: Distance-1 Constrained Channel Assignment Algorithm
 

---

**Input:**

$G = (V, E)$  : Physical Graph Model.

$A = (V, E_A)$  : Subgraph of  $G$  selected by the routing protocol.

**Output:**

a) Channels assigned to links present in  $A$ .

b) Contention degree of each colored link.

```

1: Let  $r_i = i$ 'th root of the mesh for  $i = 1$  to  $N$ 
2: Let  $h = \max(\min(d(v_j, r_i)_A)) \forall v_j \in V_i$  and  $\forall i \in 1$  to  $N$ 
3: Let  $AvailChan =$  List of available channels
4: for all edges  $e \in E_A$  do
5:    $Color(e) \leftarrow 0$ 
6:    $Co(e) \leftarrow 0$ 
7: while  $counter \neq h + 1$  do
8:   for  $i = 1$  to  $N$  do
9:      $Q = \{v_j \in V \mid d(v_j, r_i)_A = counter\}$  for some  $i \in 1$  to  $N$ 
10:   AssignLabel ( $A, Q$ )
11:   AssignColor ( $A, Q, G$ )

```

---

---

**Algorithm 2** Label and Color Assigning Procedures
 

---

**Procedure** AssignLabel ( $G1 = (V1, E1), F$ )

- 1: delete all colored edges and labels in  $G1$  and let  $l = 1$
- 2: *WhilenotallVerticesLabeled(F)* **do**
- 3:     pick unlabeled vertex  $u \in F$  of minimum neighbors in  $G1$
- 4:     **if**  $\text{degree}(u) = 0$
- 5:          $\text{label}(u) \leftarrow 0$
- 6:         **continue**
- 7:      $\text{label}(u) \leftarrow l$
- 8:     increase( $l$ )
- 9:     delete all edges incident on  $u$  from  $G1$

**Procedure** AssignColor ( $G1 = (V1, E1), F, G2 = (V2, E2)$ )

- 1: **for**  $j$  from largest label of  $\{v \in F\}$  to 1 **do**
  - 2:     let  $u \leftarrow$  vertex with label  $j$
  - 3:     **if**  $\exists$  valid  $c \in \text{AvailChan}$  for AllUncoloredEdgesOf( $u$ )
  - 4:         ColorAllEdgesOf( $u$ )
  - 5:         **continue**
  - 6:     **for**  $i = 1$  to  $\|\text{uncolored edges connected to } u \text{ in } G1\|$  **do**
  - 7:         Let  $l =$  random uncolored edge connected to  $u$
  - 8:          $c_l =$  the least indexed color not used by links at distance-1 with respect to  $G2$
  - 9:         **if** such color does not exist
  - 10:             Let  $\text{Conflict}(l) = \{\text{channels taken by root links at distance-1 from } l \text{ in } G2\}$
  - 11:             **if**  $\|\text{AvailChan} - \text{ConflictChan}(l)\| > 1$
  - 12:                  $\text{AvailChan}(l) = \{c \in \text{AvailChan} - \text{ConflictChan}(l)\}$
  - 13:             Let  $\text{AffectedLinks}(l) = \{e \in G2 \mid d(e, l)_{G2} = 1 \text{ and } \text{Color}(e) \in \text{AvailChan}(l)\}$
  - 14:              $\forall c \in \text{AvailChan}(l) \Rightarrow \text{Contention}(c) = \text{Max}(\text{Co}(e)) \{ \forall e \in \text{AffectedLinks}(l), \text{Color}(e) = c \}$
  - 15:             Let  $\text{LeastLoaded} = \{c \in \text{AvailChan}(l), \text{Contention}(c) \text{ is minimum}\}$
  - 16:             Assign the highest indexed channel from  $\text{LeastLoaded}$
  - 17:             update( $\text{Co}(e)$ ) for  $l$  and  $\{e \in \text{AffectedLinks}(l), \text{Color}(l) = \text{Color}(e)\}$
-

## Chapter 4

# Common Channel Reference MAC (CCR-MAC)

Once the DIC-CA is completed, an access protocol is required to ensure that both transmitter and receiver are available, notify the receiver about the transmitter's intention to transmit a packet, etc. In this chapter, I provide the details of the medium access protocol that is implemented by all nodes in the network. I first study four generic coordination problems inherent in multi-channel systems, namely: the Multi-Channel Hidden Terminal problem identified in [33], the Missing Receiver problem identified in [31], Contaminated Channel Availability Data Base and the Mutual Deafness problem identified in this thesis. All these problems can manifest in different single radio solutions and may cause performance degradation if not addressed properly. I then illustrate the basic principles of the proposed medium access protocol and present its complete implementation. Finally, I show how it addresses the multi-channel coordination problems.

### 4.1 Multi-Channel Coordination Problems

Regardless of the solution approach, it is challenging to coordinate transmissions over different channels where each node has a single radio transceiver. Transmissions occurring on different channels can still be misaligned. When a node communicates on a channel, it is not aware of the status on other channels and hence can attempt

to exchange information with its neighbors while they are on different channels. To design an efficient medium access protocol, we must be able to accurately characterize this lack of coordination. In this section I investigate four of such generic coordination problems.

**Multi-Channel Hidden Terminal Problem:** In this problem, which was first identified in [33], control packets sent on a certain channel fail to inform neighboring nodes currently communicating on a different channel.

For the sake of illustration, I start with a simple multi-channel MAC protocol that does not address this problem. Suppose there are  $N$  channels available. One channel is dedicated for exchanging control messages and all the other for data. When a node is neither transmitting nor receiving, it is listening on the control channel. When a node A wants to transmit a packet to node B, A and B exchange RTS and CTS messages to reserve the channel as in IEEE 802.11 DCF. RTS and CTS messages are sent on the control channel. When sending an RTS, node A includes a list of channels it is willing to use. Upon receiving the RTS, B selects a channel and includes the selected channel in the CTS. After that, node A and B switch their channels to the agreed data channel and exchange the data and ACK packets. When this handshake is done, nodes A and B immediately switch to the control channel.

Now, consider the two-flow topology of figure 4.1. In this example I assume the protocol operates with two data channels only. As shown in figure 4.2, a control

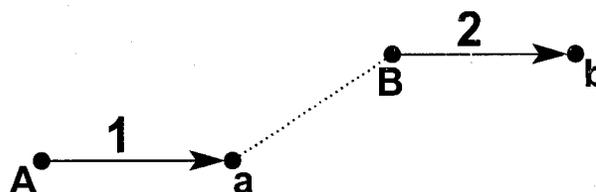


Figure 4.1 Two flow topology

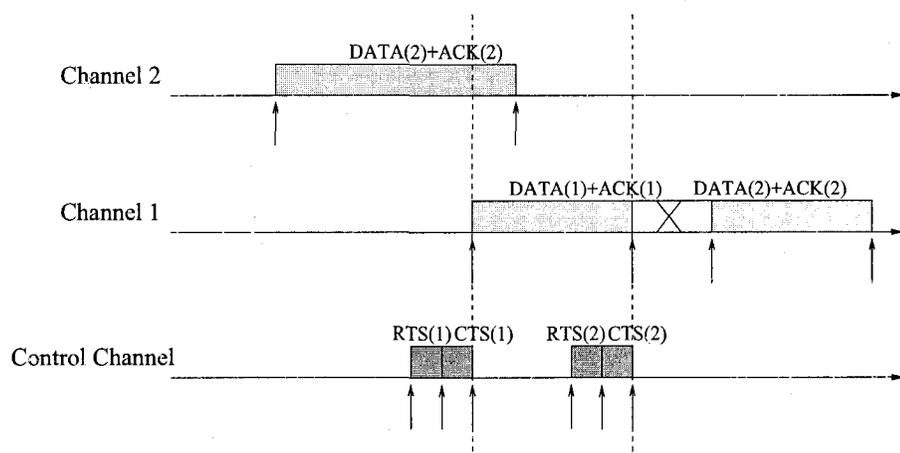
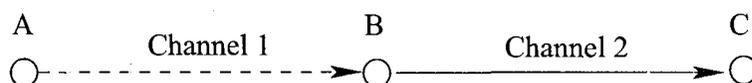


Figure 4.2 The Multi-Channel Hidden Terminal problem

packet exchange of flow  $Aa$  may occur when the flow  $Bb$  transmits on data channel 2. Suppose  $Aa$  selects data channel 1 and initiates a transmission. When flow  $Aa$  transmits, flow  $Bb$  will return to the control channel. Since it has not heard the reservation of flow  $Aa$ , it may select data channel 1. In this case, flow  $Aa$  will experience a collision, while the transmission of  $Bb$  succeeds.

The above problem occurs due to the fact that nodes may listen to different channels, which makes it difficult to use virtual carrier sensing to avoid the hidden terminal problem. If there was only one channel that every node listens to, C would



**Figure 4.3** The Missing Receiver problem

have heard the CTS and thus deferred its transmission.

**Missing Receiver Problem:** This problem, which was first identified in [31], arises when control packets sent on a certain channel to access an intended receiver, fail because this node is currently on a different channel.

To illustrate this problem, I consider the simple three-node scenario of figure 4.3, where node *A* transmits to node *B* and node *B* transmits to node *C*. First, consider the naive protocol version where all control messages are transmitted on different channels. In figure 4.3, an access attempt of *A* for *B* on channel 1 will fail if *B* is on channel 2. Then node *A* will perform random back-off and retry on channel 1. Unless proper measures are taken, this problem will cause large packet delays for flow *AB* and decreases its throughput.

The problem also persists with a protocol that separates the control channel from data channels. Suppose *A* starts contending for *B* and *B* starts contending for *C* on the control channel. As long as one of them wins the contention, the other node will be able to synchronize and resume contention at the end of the data transmission. Unfortunately, synchronization is lost when the nodes count-down simultaneously. In this case, both nodes will not be able to hear each others RTS while they transmit.

Therefore the RTS from  $B$  to  $C$  succeeds, while the RTS from  $A$  to  $B$  fails. After this point, node  $A$  will try to discover node  $B$  using random back-off. This is difficult to occur since  $A$  will need to find a short interval where  $B$  returns for its own back-off on the control channel. It is more likely for  $B$  to contact  $A$  when it contends in the control channel for the next packet for  $C$ . In this case,  $A$  synchronizes with the end of transmission of  $B$  but it will already have a large back-off interval and will not be able to compete fairly for  $B$ . It is evident that similar inefficiencies arise in the other version of the Missing Receiver Problem, where node  $B$  acts as receiver on link  $BC$ . Note that the Missing Receiver Problem does not exist in a single channel system because  $A$  can carrier sense the data transmissions of  $B$  and immediately defer until the end of  $BC$  transmission.

**Mutual Deafness Problem:** This problem arises in a node based channel assignment approach, where each transceiver is assigned a “home channel” different from its neighbors’ channels. If two such nodes intend to transmit packets to each other at the same time, they will switch to two different channels and can remain there for a long time interval.

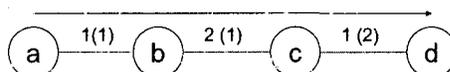
An example of this problem occurs in the protocol introduced in [20]. In this protocol, whenever a transceiver wants to transmit a DATA packet to any of its neighbors, it switches to its neighbor’s home channel and negotiates for the transmission using common contention resolution techniques similar to IEEE 802.11 DCF.

However, such an approach can result in severe throughput degradation because of the Mutual Deafness problem. This problem specially manifests itself in scenarios such as: mixed upstream-downstream traffic and TCP traffic as nodes wait for each other on their neighbors' "home channels." Hence, they are mutually deaf to packets that are received on their own home channel. I refer to this problem as *Mutual Deafness problem*. Note that this deadlock occurs irrespective of the number of available channels and due to the fact that each radio can listen or transmit to only one channel at a time. I have provided TCP simulation results of a single chain of 3 hops to show this problem. Here original DATA packets are sent in one direction and ACK packets come from the reverse direction and hence very often mutual deafness problem occurs. Simulation results show that TCP throughput decreases to 25% of its original value in single channel 802.11 irrespective of the number of channels.

**Contaminated Channel Availability Data Base:** This problem is a generic problem of TBCS protocols (e.g., AMCP [31], MMAC [33], DCA [38],...). All these protocols, divide the spectrum to control and data channels and use control packet message exchange prior to any data packet transmission. As a result the performance of all these protocols depends on the accuracy of the channel availability data base. As a result, whenever a node receives collisions on the control channel or receives packets it can not decode (e.g., being in interference range but not in transmission range), it loses the information available in the control packets:

**Table 4.1** Mutual deafness and contaminated channel data base problems investigation in a 3 hop chain topology. Simulations are done for 1000 byte packets on a 2Mbps channel. All results are in terms of pkt/s.

Topology	Traffic	Protocol Name	Protocol Throughput	802.11 Throughput
Chain	TCP	xRDT	16	43
Chain	CBR	AMCP	52	55



**Figure 4.4** Single three hop flow.

Corrupted information in the channel availability data base results in wrong channel selection and collisions of data packets. As a result, the throughput degradation can be very high.

Figure 4.4, shows an example of such problem in AMCP [31] for a 3 hop chain topology and CBR traffic. In this protocol, spectrum is divided to control channel (for channel reservation packet exchange) and data channels (for DATA/ACK packet exchange). Here whenever node *b* receives a collision on control channel it loses the channel node *c* has chosen for its transmission and might select this channel for its later DATA reception from *a*. In case of AMCP the channel node *b* chooses is always the one that *c* selects irrespective of number of channels and is related to the specific mechanism that channel selection is done in AMCP. Compared to 802.11, the total throughput is decreased though the low throughput in 802.11 is due to high average

contention window of node  $a$  and single channel nature of 802.11 which does not allow any spatial reuse.

## 4.2 CCR-MAC

In this section, I first provide basic principles of the random access MAC protocol introduced in this thesis. I next provide a detailed description of the MAC rules that collaborates with the channel assignment algorithm to achieve high channel utilization.

### 4.2.1 Overview

In the MAC protocol, I use a separate dedicated control channel as a common reference for all transceivers in the network. Whenever a transceiver wants to send a DATA packet, it contends for the control channel, and exchanges reservation packets similar to the RTS/CTS defined in 802.11 distributed coordination function. After a successful control message exchange in the control channel, the transceivers switch to the data channel and exchange their DATA and ACK packets.

In contrast to TBCS protocols, CCR-MAC selects the channel according to the D1C-CA assignment rather than making a local and dynamic assignment decision. A second difference between CCR-MAC and TBCS MACs that utilize control channels [31, 38, 33] is the behavior in case of contaminated channel availability database, i.e., channel availability uncertainties. In TBCS protocols, each node can transmit

data on each of its outgoing links on all data channels. Hence whenever a transceiver misses a control packet due to listening on a different channel or receives a corrupted control packet due to collisions, its channel availability database is inaccurate and all database entries should be marked as corrupted.

The behavior in the case of a corrupted database varies between different approaches. Some force a transceiver to wait until it is certain that a channel is available before contending for this channel. Others may allow users to contend for a channel even if they are not certain that the data channel is available thereby risking data-channel collision. In all cases, in heavy traffic scenarios, when channel utilization is crucial due to an increasing number of control message collisions, the data-channel availability database can become increasingly contaminated as I illustrate later with simulations.

In CCR-MAC, the channel allocation algorithm provides the MAC with a channel to use and the contention degree for each of its outgoing links. In case the channel assigned to a link has a contention degree equal to zero, it is always safe to transmit on this channel without any risk of data-channel collision. Hence, whenever the sender and receiver are available, they coordinate an immediate data channel switching and data transmission based on the control channel contention rules described above.

In case of a contention degree greater than zero, the transceiver must still take precautions before scheduling a data transmission on the pre-assigned channel. For

example, if a node was previously on a different channel, it must track the availability of the assigned data channel and wait until it ensures the channel is idle. In the special case of control packet collision, transceivers do not reset their channel availability entry and risk that the missed control packet has reserved the relevant data channel. However, since the channel allocation algorithm minimizes the contention degree on each data channel, i.e., minimizes the number of contending transceivers per channel, the risk of data channel collision is small.

Observe that for a mesh topology, the channel allocation algorithm gives high priority to links connected to the gateway, and the contention degree of all first hop links is typically zero. Hence whenever a first hop transceiver wants to forward traffic to the gateway (or vice versa) and both transceivers are available, they can coordinate an immediate data transmission. Furthermore, the first hop links, which are expected to be the mesh network bottleneck, are free of collisions on the data channel.

#### 4.2.2 Protocol Rules

Initially all nodes are listening on the control channel. The procedure for sending the control and data packets is as follows:

***Data Message Send (MSG<sup>i</sup>(j)):*** In order for node  $i$ , to send a packet to node  $j$ , it first contends on the control channel, but before an RTX message is sent the channel state is checked. If the link that node  $i$  wants to communicate to is not conflict free (has a contention degree greater than 0) and the CAV (Channel

Availability Vector as defined in Algorithm 4) is greater than current time, node  $i$  has to wait on the control channel long enough to ensure the availability of the data channel. This is done by setting its NAV for the time remained for the channel to become available and resetting its current contention window. Otherwise node  $i$  will send an RTX with the assigned channel in its index.

**RTX Receive ( $RTX^{s \rightarrow r}(k)$ ):** After node  $j$  receives an RTX, it first checks to see if it is the intended receiver of the packet. If so, the node will check its CAV and if it is greater than current time and the link it wants to communicate is not conflict free (has a contention degree greater than 0), then the node has to wait on the control channel so that the assigned channel becomes available. Hence the node sends a rejecting CTX, putting 0 in the header of the packet and also sending the amount of time until the channel will become available for it. To avoid any kind of deadlock, next the node  $j$  sets its NAV for the time until the channel will become available. Otherwise the node simply sends a CTX with the same channel in its header and switches to the data channel. If  $j$  is not the intended receiver, it simply updates its CAV by adding the duration in the header of RTS to the current time for the channel inside the header of RTX.

**CTX Receive ( $CTX^{s \rightarrow r}(k,n)$ ):** If node  $i$  is the intended receiver of the CTX, the packet is checked. If it is a rejecting CTX,  $i$  will set its NAV for the amount of time the channel will become available for its intended receiver and resets its

---

**Algorithm 3** Multi-Channel Mac Protocol
 

---

Variables:

- $CH$  -The set of available data channels  
 $Time^i$  -The current time according to node  $i$ 's clock  
 $CAV^i$  -Channel Availability Vector, containing the list of channels and the exact time that any channel will become available according to node  $i$   
 $NCL^i$  -Neighbor Channel List, containing the list of node  $i$ 's neighbors and the assigned channels for communication  
 $CFL^i$  -Conflict Free List, containing the list of node  $i$ 's neighbors and a 0 or 1 bit indicating if the channel assigned to this link is conflict free or not, i.e., a 1 indicates that there no link with the same channel at distance one

Messages:

- $RTX^{i \rightarrow j}(k)$ - The message sent by node  $i$  to node  $j$  requesting it to switch to data channel  $k \in CH$   
 $CTX^{j \rightarrow i}(k,n)$ - The message sent ny node  $j$  to node  $i$  to notify it is clear to switch to data channel  $k \in CH$ . if  $k$  is not available for receiver  $k = 0$  and  $n =$  remaining time until the channel is available for receiver  
 $MSG^i(j)$  -Data message sent by node  $i$  to node  $j$   
 $ACK^j(i)$  -Acknowledgement sent by node  $j$  to node  $i$

Initialization:

- 1  $CAV^i(h) \leftarrow Time^i + T \quad \forall h \in CH$
  - 2  $NCL^i \leftarrow$  Provided by the NCC
  - 3  $CFL^i \leftarrow$  Provided by the NCC
  - 4 LongNav = 0
-

---

**Algorithm 4** Multi-Channel Mac Handshaking Rules
 

---

DATA Message Send(  $MSG^i(j)$  )

```

1 Contendi
2 if  $CAV^i(NCL^i(j)) \geq Time^i$  and  $NCL^i(j) \neq ConflictFree$ 
3   SetNav until  $NCL^i(j)$  is available
4   reset contention window
5 else
6    $k = NCL^i(j)$ 
7   Send $RTX^{i \rightarrow j}(k)$ 

```

RTX Receive (  $RTX^{s \rightarrow r}(k)$  )

```

1 if  $r == i \Rightarrow$  Node  $i$  is the intended receiver of the RTX
2   if  $(CAV^r(NCL^r(s)) \geq Time^r$  and  $NCL^r(s) \neq ConflictFree)$ 
3     Send $CTX^{s \rightarrow r}(0, n)$ 
4     SetNav( $n$ )
5   else
6     Send $CTX^{s \rightarrow r}(k, 0)$ 
7     Switchs( $k$ )
8 else Node  $i$  is not the intended receiver
9    $CAV^i(k) \leftarrow Time^i + T$ 
10  if  $i = Sender - Receiver \Rightarrow$  SetNav( $T$ ) and reset contention window

```

CTX Receive (  $CTX^{s \rightarrow r}(k, n)$  )

```

1 if  $r == i \Rightarrow$  Node  $i$  is the intended receiver of the CTX
2   if  $(k == 0)$ 
3     SetNav( $n$ )
4     reset contention window
5   else Switchr( $k$ )
6 else Node  $i$  is not the intended receiver
7   if  $k \neq 0$   $CAV^i(k) \leftarrow Time^i + T$ 
8   if  $i = Sender - Receiver \Rightarrow$  SetNav( $T$ ) and reset contention window

```

ACK Receive or ACK Send(  $ACK^{j \rightarrow i}(k)$  )

```

1  $CAV^i(m) \leftarrow Time^i + T \quad \forall m \in CH$ 
2  $CAV^i(k) \leftarrow Time^i$ 

```

---

contention window. After the NAV expires,  $i$  will again contend. If the received CTX is confirming then  $i$  will switch to the data channel and sends its data packet. If  $i$  is not the intended receiver and it hears a confirming CTX, then the CAV is updated according to duration in the header of CTX and if it is a rejecting CTX then nothing is done.

***Going back to control channel:*** Any time a node comes back to control channel, it resets CAV for all channels by adding maximum reservation duration to the current time. (In our case maximum reservation duration is equal to a packet transmission time) If there was a successful transmission on the data channel, then the CAV for this channel is available by writing the current time for this channel. On the other hand if a node comes back to the control channel and can not decode the packet, it will wait until the medium becomes idle.

### 4.3 Addressing Multi-Channel Coordination Problems

I now present how the CCR-MAC protocol solves the coordination problems described in section 4.1.

**Multi-channel Hidden Terminal Problem.** Consider again the Multi-channel Hidden Terminal Problem example of figure 4.1. Recall that when flow  $Bb$  arrives on the control channel during  $Aa$  transmission on data channel 1, it does not have sufficient information about the state of channel 1 because it has not heard the RTS/CTS packet of flow  $Aa$  while transmitting its own data packet on data channel 2. If it

selects channel 1 it will cause a collision to the disadvantaged flow Aa. Under CCR-MAC, the channel assignment protocol has global information about the topology and with sufficient number of channels allocates two different channels to each of these flows. In case that because of insufficient number of channels, the two flows are assigned the same channel node B sets channel 1 as unavailable and sets a timer to expire after the duration of a RTS/CTS/DATA/ACK transmission. Note that channel 1 may or may not actually be available, but node B sets it to unavailable, precisely because it does not have this information. Node B will compete for channel 1 only after the timer expires by that time any transmission on channel 1 will have completed. If any RTS/CTS for channel 1 is heard during this period, node B will defer further but will have synchronized for contention on channel 1.

**Missing Receiver Problem.** Consider the scenario shown in figure 4.3, where *A* wants to transmit to *B* when *B* is transmitting to *C*. CCR-MAC handles the Missing Receiver Problem as follows. If *A* receives from *B* an RTS intended to *C*, *A* will defer until the end of the ongoing transmission of *B* and examine its back-off stage. If it is already in high back-off stage, *A* sets its contention window size to the minimum value. In this way, *A* will fairly contend for the attention of *B* when *B* is in idle state. In contrast, in the naive protocol, *B* will transmit many packets before *A* decrements its back-off counter to zero.

**Mutual Deafness deadlock.** As in my channel assignment protocol, channels

are assigned to links instead of nodes, two nodes that intend to send messages to each other will always be tuned to the same channel and hence this deadlock will never occur.

**Contaminated Channel Availability Data Base.** In my proposed channel assignment and medium access design architecture, the network control center which has global information of all links is responsible for channel assignment. Two cases need to be considered: With sufficient number of channels, all channels that are assigned to links have zero contention degree and hence there is a zero chance of channel availability data base to become contaminated or in other words the chance of collision of data packets is zero. However in case of contention degree greater than zero, nodes need to track the availability of their assigned channel on the control channel. Since, there is always a chance of collisions of control packets, nodes still might lose the information in control packets and get contaminated channel availability state. However I emphasize that this happens only for links with contention degree greater than zero and furthermore happens for links that are far from the gateway node as the channel assignment procedure favors links connected to the gateways.

## Chapter 5

# Performance Evaluation

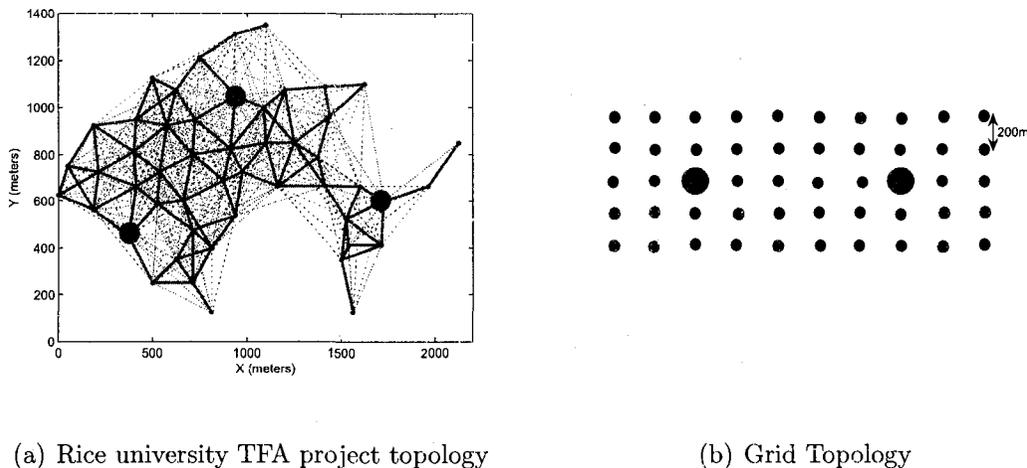
In this chapter, I evaluate the performance of my channel assignment and MAC protocol design through simulations. I compare my scheme with one-channel-per-gateway IEEE 802.11 and AMCP, a multi-channel MAC protocol in the class of TBCS protocols [31]. AMCP uses a separate control channel for channel negotiation before any data packet transmission. My experiments use the same MAC layer parameters as [31]. These parameters are summarized in table 5.

SIFS	10 $\mu s$
DIFS	50 $\mu s$
EIFS	364 $\mu s$
$\sigma$	20 $\mu s$
BasicRate	2 Mbps
DataRate	2 Mbps
PLCP length	192 bits @ 1 Mbps
MAC header (RTS,CTS,ACK,DATA)	(20,14,14,28) bytes @ BasicRate
Packet size	1000 bytes
$(cwmin, cwmax)$	(31,1023)
Retry Limit (Short,Long)	(7,4)
Channel switching delay	224 $\mu s$
MMAC ATIM window	20ms
MMAC Beacon interval	100ms

**Table 5.1** MAC layer parameters

### 5.1 Simulation Setup

For simulations, I modified the ns-2 simulator with CMU wireless extensions to perform the described MAC and channel assignment procedure. Simulations are performed for three different backbone topologies. In the first topology, the mesh coordinates correspond to a deployed mesh network [1]. This topology uses three



(a) Rice university TFA project topology

(b) Grid Topology

**Figure 5.1** Topology

gateways and is plotted in figure 5.1(a). For the second topology, I choose an ideal placement of nodes in a rectangular 5x10 grid topology and plot it in figure 5.1(b). In this topology, each node is at a distance of 200m from its immediate neighbors and there are two gateways in the middle of 5x5 square grids inside the original 5x10 rectangular grid topology. For the third topology I choose a general  $m$ -ary tree topology of depth  $L$ . Each node in these topologies has transmission range of 250m and interference range of 550m ([28]).

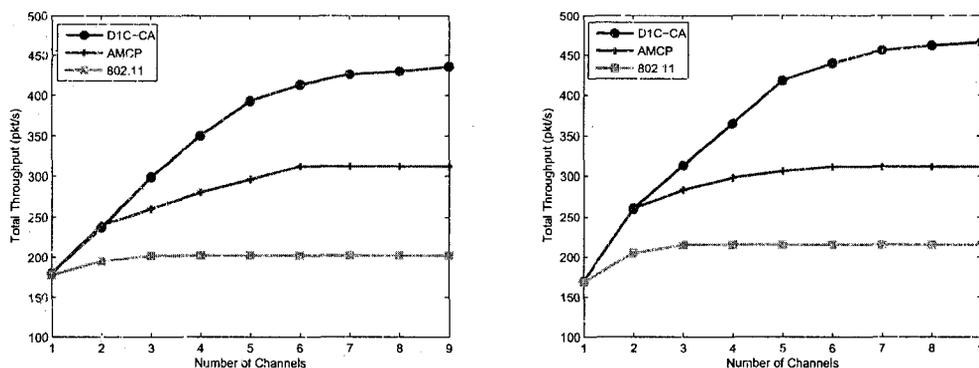
In all simulations, each node initially calculates the shortest path route to the nearest gateway and uses this static route for gateway access. In my scheme, this information is also provided for the NCC which runs the D1C-CA algorithm and allocates channels to all such links selected by the routing protocol. For 802.11 simulations, each gateway and its associated subgraph has a dedicated and separate

channel to avoid interference within subnetworks (companies like tropos [36] have constructed mesh networks with this structure) and the 802.11 MAC is used with RTS/CTS enabled.

In the first two topologies, 25 nodes are randomly selected as source/sink in upload/download gateway-traffic patterns. For the third topology all nodes have packets to transmit. I use a packet size of 1000 bytes and set the channel capacity to 2 Mbps. The channel switching delay is set to  $80 \mu\text{sec}$  ([4]) and each source node generates and transmits constant bit rate traffic via UDP. I drive the network to saturation as follows. For a particular scheme and number of channels, I run a series of simulations, increasing the offered load of all the flows proportionally, starting from a low value. I stop when the throughput does not increase any further with a further increase in the offered load.

## 5.2 Simulation Results

In this section I provide the simulation results for the described network topologies. I consider the aggregate and per-flow throughput results as the performance metrics and evaluate the impact of number of available channels, packet arrival rate and channel switching delay on the above performance metrics. These simulations are done for the TFA and grid topologies. I next consider the general  $m$ -ary tree topology and investigate depth and degree of the tree as the topology factors that impact the performance of the distance-1 channel assignment scheme.



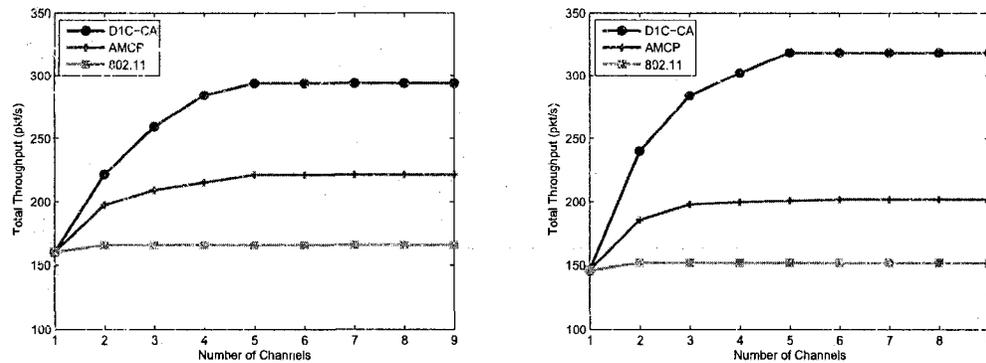
(a) Total upload throughput comparison      (b) Total download throughput comparison

**Figure 5.2** Aggregate throughput comparison for TFA topology

### 5.2.1 Aggregate channel utilization

Figures 5.2(a), 5.2(b), 5.3(a) and 5.3(b) correspond to aggregate upload/download simulation results for each of the topologies. With a single radio constraint at each node, the maximum aggregate throughput of a single link in isolation is limited to 184 pkt/s. Total throughput simulation results show an average of 150pkt/s achieved at each of the gateways with sufficient number of channels to have a valid D1EC (9 for figure 5.1(a) and 7 for grid topology). This shows that my approach can indeed utilize the increase in the number of channels and deliver close to the maximum throughput in topologies with high contention and collisions on the control channel.

In contrast, for AMCP, each time a collision on the control channel is received, the data base of available channels become inaccurate and hence there is a possibility of wrong-channel selection and collision on the data channel. Furthermore, in AMCP



(a) Total upload throughput comparison      (b) Total download throughput comparison

**Figure 5.3** Aggregate throughput comparison for grid topology

whenever a node  $a$ , comes back to the control channel, it only has information about availability of the previously used channel. Hence if some other node selects this channel for its future transmission, node  $a$  must wait on the control channel for a sufficient time to ensure a channel is available. This problem is specifically present in download traffic pattern simulation results in which the gateway node loses its previously-used channel to some other node and hence has to wait on the control channel until a channel becomes free. Consequently, AMCP does not efficiently utilize the available channels and saturates with small number of available channels. This problem is not limited to this protocol only. As I described previously, all protocols which belong to the TBCS class of solutions suffer from *contaminated channel availability data base* problem and hence perform a poor performance.

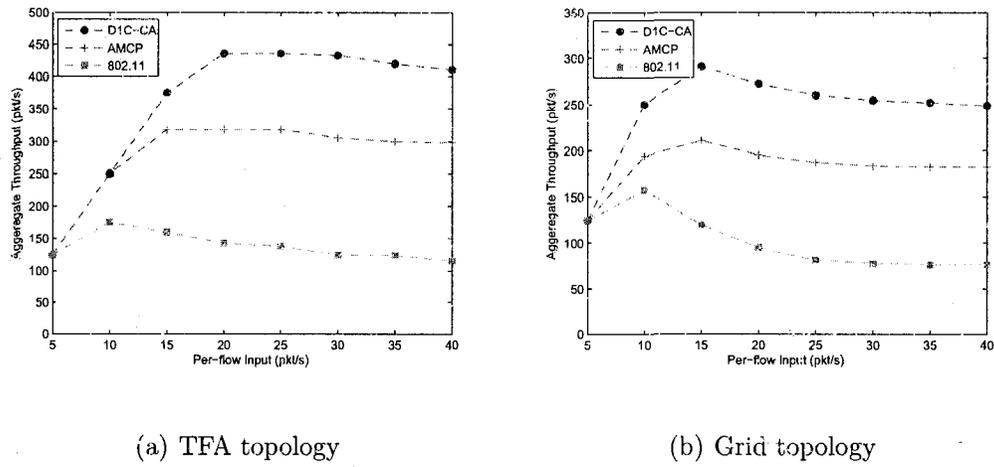
The same trend of performance is present with number of channels smaller than

$k_{DIEC}$ . Indeed as D1C-CA attempts to assign channels with zero contention degree to links connected to the gateway, high channel utilization is expected with only one additional channel as links connected to the gateway will no longer be a bottleneck. This observation is shown through the high increase in total throughput with only two available channels in figures 5.3(a) and 5.3(b). In contrast, for the topology of figure 5.1(a), one additional channel is not sufficient to color all links connected to each gateway as gateways interfere with each other. This is the main reason for low throughput improvement in figures 5.2(b) and 5.2(a) with 2 channels. With further increase in the number of channels, D1C-CA efficiently splits contention among different links. Moreover, by lowering the contention degree, data channel collision probability is minimized. This contrasts to AMCP in which nodes contend for all channels such that high collision probability is present on each of the channels.

Finally, the one-channel-per-gateway 802.11 simulation results show limited throughput improvement compared to link based channel assignment approaches. This is because of high interference that is present within each subnetwork compared to amount of interference between different subnetworks.

### 5.2.2 Effect of traffic load

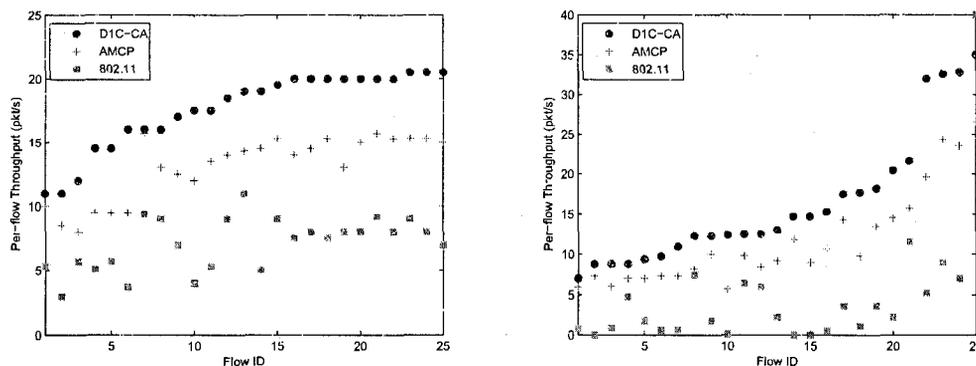
I now evaluate the effect of offered load. Figures 5.4(a) and 5.4(b) depict the aggregate throughput variation in uplink gateway traffic scenarios when 9 channels are available. Until 5 pkt/s, the load is too small to exploit the channels and hence



**Figure 5.4** Total throughput variation with offered load

all approaches yield the same performance. After that point, channelization becomes effective and multiple channels are exploited to increase aggregate throughput. In contrast, AMCP is limited to the use of available channels due to the inaccurate channel state information that is inherent in the TBCS.

Additional increase in offered load results in degradation of throughput in all approaches. However, this throughput degradation is more severe for 802.11 in the grid topology of figure 5.4(b). In this structured topology, more hidden terminals are present around the gateway nodes and data packet collisions are present even with RTS/CTS mechanism enabled. This is due to nodes that are within interference range of the gateway node as well as packet transmission duration that is large compared to the EIFS time. In contrast, other approaches show smooth throughput degradation with increase in the offered load. This is indeed because of channelization and the



(a) Upload per flow throughput in saturation (b) Upload Per flow throughput with backlogged flows

**Figure 5.5** Per flow throughput in TFA topology

fact that data packets are transmitted on different channels than RTS/CTS packets.

### 5.2.3 Fairness among flows

With the aggregate throughput metric, some flows might capture the medium completely and result in starvation of other flows. Hence I study per-flow throughput results to evaluate fairness among flows. In my simulations I observed fair throughput division among flows for download scenarios irrespective of the approach and the offered load. Hence, I only provide upload simulation results with 9 available channels for the topology of figure 5.1(a). The same trend of performance was observed with different number of channels and for grid topology.

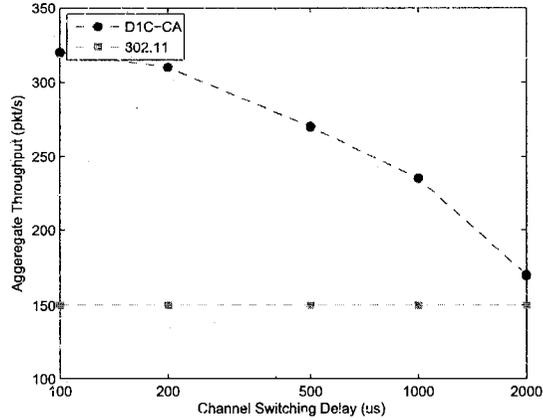
Figure 5.5(a) depicts per flow throughput results in the saturation region. As expected, both my scheme and AMCP achieve higher per-flow throughput results compared to 802.11. Furthermore, 23 out of 25 flows have further increased their

per-flow throughput in my approach.

On the other hand, the fairness characteristics become different under high traffic load conditions when each flow is fully backlogged and always has a packet to transmit (figure 5.5(a)). In this scenario, flows that are closer to the gateway or have fewer hidden terminals and contenders have a higher chance to transmit their data packets. This leads to starvation of many flows in 802.11 where 10 out of 25 flows have reached almost zero throughput. In contrast, fairness properties are better in both AMCP and my approach in which interfering links transmit their data packets on different channels and hence each flow receives a minimum amount of bandwidth. However as expected per-flow throughput results become different than results of figure 5.5(a), as links closer to the gateway have a higher chance to win the control channel and send more packets to the gateway node.

#### 5.2.4 Effect of channel switching delay

As in my proposed scheme nodes switch between control and data channels at packet level, the channel switching delay overhead can become the bottleneck. In figure 5.6 I have plotted such impact on aggregate achieved throughput. As observed from this figure, below 200  $\mu$ sec the impact of channel switching is negligible. This is because of packet transmission time which is large compared to the switching delay. However channel switching delay of 2 msec or higher decreases the throughput to the same level as 802.11. This overhead can be simply addressed by reserving a



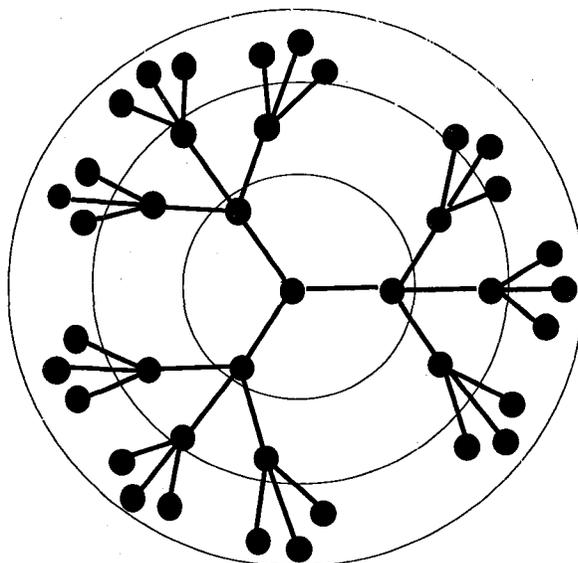
**Figure 5.6** Impact of channel switching delay for grid topology

channel for multiple packet transmissions instead of a single packet and can be easily incorporated with my MAC protocol.

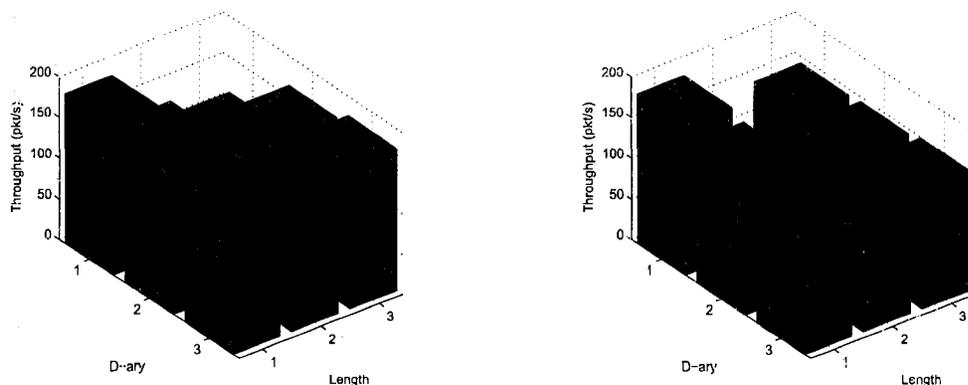
### 5.2.5 Topology Factors

In this section I explore the impact of mesh network topology on the throughput achieved by different methods. To this end, I consider the wireless mesh network as an  $m$ -ary tree topology of length  $L$ , where  $m + 1$  denotes the connectivity degree of any node and  $L$  denotes the depth of the tree in number of hops as shown in figure 5.7. I assume 12 available channels and that all nodes in the network have packets to transmit. I explore the impact of topology by varying the length and degree of the tree.

Figures 5.8(a) and 5.8(b) denote the aggregate download and upload throughput results for the static distance-1 channel assignment scheme. In the download scenario



**Figure 5.7** Ternary tree: Tree with 3 levels and connectivity degree  $m + 1 = 4$  of figure 5.8(a), with tree degree of greater than 1, achieved throughput is irrespective of the number of hops or degree of the tree and remains close to maximum achievable throughput of a single link. However with only one active branch in the tree, the node close to gateway becomes the bottleneck. The forwarding behavior of this node, causes the gateway to defer as there is only a single branch active in the tree. The same trend of behavior is present in the upload scenario of figure 5.8(b). However two distinctions are apparent compared to the download scenario. First, the aggregate throughput is less with more hops. This is due to high contention among the nodes around gateway that causes collisions around the gateway node, where as in download scenario the gateway node schedules all first hop downlink transmissions



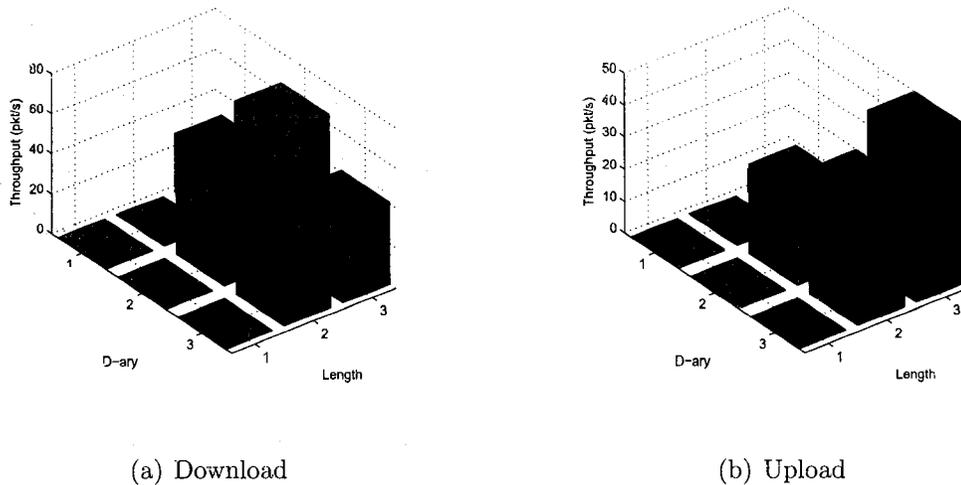
(a) Aggregate download throughput

(b) Aggregate upload throughput

**Figure 5.8** Tree topology

and hence lower contention and collision is present around the gateway node. Second, the aggregate throughput with degree 1 and length 3 is higher. This is due to the fact that the node close to gateway will have more opportunities to transmit to gateway compared to download scenario.

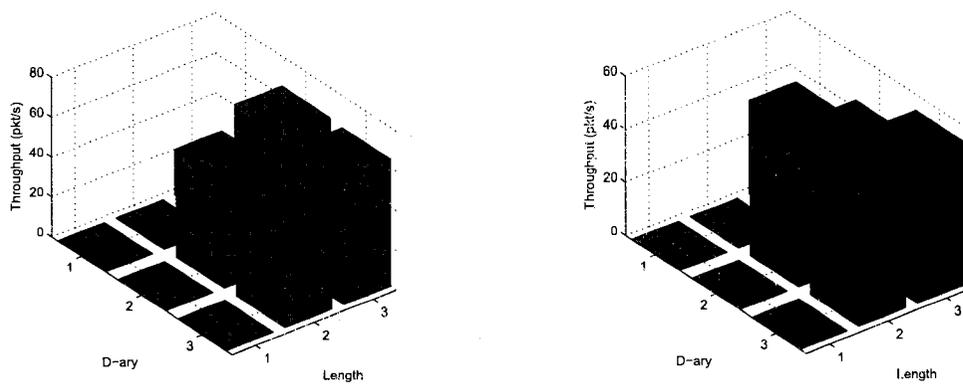
Figures 5.9(a) and 5.9(b) denote the relative gain of distance-1 channel assignment scheme compared to AMCP. With a tree of length 1, nodes form a wireless LAN setting (i.e., all internet traffic is one hop from the gateway node), where only one channel is used and hence no throughput improvement is achieved. However, as the length of the tree increases, D1C-CA approach efficiently eliminates the bottleneck around the gateway node and provides considerable performance improvement compared to AMCP. This performance improvement has a higher average in download scenario compared to upload scenario as AMCP suffers from channel information



**Figure 5.9** Relative gain of D1C-CA to AMCP

loss more frequently in download scenario.

Finally I evaluate the performance improvement of D1C-CA scheme compared to single channel 802.11. Note that using 802.11 as the access scheme in a mesh network, would result in subnetworks on different channels where each subnetwork has its root in a mesh gateway node. I provide relative gain of download and upload scenario throughput results in figures 5.10(a) and 5.10(b) respectively. With a tree of single hop length, the mesh network performs similar to wireless LAN networks and hence additional channels can not provide additional improvements. However increase in the length of the tree provides considerable performance gains. Furthermore, as all nodes in an 802.11 based network work on the same channel, length of the tree plays an important role as nodes two hop away from gateway are hidden from gateway and can cause potential data packet collisions.



(a) Download

(b) Upload

**Figure 5.10** Relative gain of D1C-CA to 802.11

## Chapter 6

### Related Work

I divide the related work into two broad categories: prior use of graph coloring algorithms in wireless networking, and alternative approaches to exploiting frequency diversity.

#### 6.1 Graph Theoretic Techniques

**Graph Theory-Based Coloring.** Relevant and widely applied graph theoretic techniques include *list coloring* and *labeling problem*. See [8, 19, 10] for surveys. A list coloring of a graph is an assignment of colors to each vertex from the list of available choices, such that two nodes that are connected with an edge get different colors. The  $L(h, k)$  labeling of a graph  $G$ , is an assignment of non-negative integers to the nodes of  $G$ , such that adjacent nodes are labeled with at least  $h$  apart and nodes that share a common neighbor node are at least  $k$  apart.

**Graph Theory-Based Channel Assignment.** Several applications of such theory to networking problems include mapping resource assignment problems for resources such as time, frequency, codes, and radios to different edge and vertex coloring problems [27, 6, 11, 21, 18, 3, 30, 5]. For example in the TDMA link scheduling problem with node exclusive interference model, links that share a node in common (i.e., according to our definition are at distance-0) should be given different time slots.

Similarly, with the RTS/CTS model, links that either share a node in common or interfere with each other (i.e., are at distance-0 or 1) should be given different channels. In other applications, channel assignment is used to increase spatial reuse for cellular networks by assigning the same channels to cells which are sufficiently apart.

**D1EC Problem.** In contrast to all such work, I am the first to formulate and investigate the D1EC problem, a specific form of graph edge coloring in which only edges at distance-1 are constrained. Furthermore, while most graph coloring problems assume sufficient number of colors (time slots, CDMA codes or frequencies), we consider that the number of available channels is constrained without any guarantee that the solution to the D1EC problem exists. This is important because many standards specify a fixed number of available channels, e.g., 12 channels in IEEE 802.11a.

## 6.2 Protocols to Exploit Frequency Diversity

A significant body of work exists in protocol design for exploitation of multiple orthogonal frequencies and here we summarize a representative sample.

### 6.2.1 Single-Radio Protocols

In SSCH[4], each node switches the channels periodically using a 13-hop pseudo-random sequence. Within a channel hop duration each node uses IEEE 802.11 DCF to transmit to its neighbors. SSCH is vulnerable to the HOL (head of the line) blocking as a link is reserved for multiple packet transmissions. Furthermore loss of

schedule because of collisions can cause severe decrease in throughput. Single-radio multi-channel MAC protocols include [31, 33]. In MMAC [33], nodes are synchronized and meet at a common channel periodically to negotiate channels for use in the next phase. In AMCP [31], a separate frequency channel is used for channel negotiation before each data packet transmission. These protocols belong to the TBCS class and have been considered by the IEEE 802.11s working group as potential approaches to support multi-channel capability in single radio mesh networks [15]. As described previously and demonstrated via simulations, such algorithms can be viewed as local and greedy and can yield inefficient allocations. Likewise, reference [37] proposes assigning channels to disjoint network “components” in order to increase capacity. However, in contrast to our work, if applied to a single-gateway mesh network, the component methodology would yield a static single 802.11 channel allocation.

### 6.2.2 Multi-Radio Protocols

[25, 24, 16, 14, 38] are examples of link based assignments with multiple radios. In [38, 14] a separate control channel is used for control message exchange and each user is equipped with 2 transceivers. These approaches also inherit the problems with TBCS approach. Ayda et al. [2] proposes a channel assignment strategy where the interface assignment is determined via a measurement based approach and at any point only one radio is selected for packet transmission. [28, 29, 35, 26, 23, 34] are examples of long term link based channel assignments. In [29] the authors proposed fixed

channel assignment algorithm when the traffic pattern is known. The author propose distributed channel assignment in [28] for a tree topology. In [35, 23] the authors propose channel assignment algorithms while keeping some measure of connectivity. Minimizing interference for multi radio channel assignments have also been proposed in [23, 34]. Shin et al. [32] propose a randomized distributed channel assignment method called SAFE that uses one-hop neighborhood information.

In contrast, my problem requires only that links that are a specific distance to be assigned different channels. More fundamentally, my results are achieved with only a single half-duplex transceiver.

## Chapter 7

### Conclusion

In this thesis I addressed the channel assignment problem in single radio wireless mesh networks. In my approach, channels are quasi statically assigned to a set of links selected for traffic forwarding by the routing protocol. Moreover, I designed a MAC protocol that supports parallel transmissions using the targeted channel assignment. I modeled channel assignment as a new type of graph coloring problem where edges at distance one are constrained. I proved that the problem is NP-complete and designed an efficient heuristic solution. I further investigated the performance of the assignment algorithm and provided bounds on the minimum number of needed channels such that maximum throughput is achieved. Finally, I provided extensive simulations and demonstrated considerable performance improvements compared to alternate schemes.

For future work I plan to address the following related problems: First, in my approach I assumed the routing paths for gateway traffic forwarding is fixed and determined in advance. Interaction between my specific channel assignment and routing can be exploited to further improve the performance. Second, the theoretical results that I provided in this thesis including upper bounds on distance-1 chromatic index and algorithm complexity analysis are done for the unit disk graph model.

Improving the results for a more general model is an interesting problem and needs further research.

## References

1. <http://tfa.rice.edu/coverage.html>, (2007).
2. A. Adya, P. Bahl, J. Padhye, A. Wolman, and L. Zhou, *A Multi-Radio Unification Protocol for IEEE 802.11 Wireless Networks*, Proc. IEEE International Conference on Broadband Networks (Broadnets), 2004.
3. M. Alicherry, R. Bhatia, and L. E. Li, *Joint Channel Assignment and Routing for Throughput Optimization in Multi-Radio Wireless Mesh Networks*, Proc. ACM MobiCom, 2005.
4. P. Bahl, R. Chandra, and J. Dunagan, *SSCH: Slotted Seeded Channel Hopping for Capacity Improvement in IEEE 802.11 Ad Hoc Wireless Networks*, Proc. ACM MobiCom, 2004.
5. H. Balakrishnan, C. L. Barrett, V. S. Anil Kumar, M. V. Marathe, and S. Thite, *The Distance-2 Matching Problem and Its Relationship to the MAC-Layer Capacity of Ad Hoc Wireless Networks*, IEEE Journal on Selected Areas in Communications, 2004.
6. A. A. Bertossi and M. A. Bonuccelli, *Code Assignment for Hidden Terminal Interference Avoidance in Multihop Packet Radio Networks*, IEEE Transactions on Networking (1995).

7. J. A. Bondy and U. S. R. Murty, *Graph Theory with Applications*, New York: McMillan, 1976.
8. Tiziana Calamoneri, *The  $L(h, k)$ -Labelling Problem: A Survey and Annotated Bibliography*, *The Computer Journal* (2006).
9. S. Cherry, *Wi-fi nodes to talk amongst themselves*, *IEEE Spectrum*, 2006.
10. D. Fotakis, G. Pantziou, G. Pentaris, and P. Spirakis, *Frequency Assignment in Mobile and Radio Networks*, 1999.
11. J. Garcia-Luna-Aceves and J. Raju, *Distributed Assignment of Codes for Multi-hop Packet-Radio Networks*, *Proc. of MILCOM*, 1997.
12. M. R. Garey and D. S. Johnson, *Computers and Intractability, A Guide to the Theory of NP-completeness*, W. H. Freeman and Company, 1979.
13. R.L. Graham, B.D. Lubachevsky, K.J. Nurmela, and P.R.J. stergd, *Dense packings of congruent circles in a circle*.
14. W. Hung, K. Law, and A. Leon-Garcia, *A Dynamic Multi-Channel MAC for Ad Hoc LAN*, *Proc. of 21st Biennial Symposium on Communications*, 2002.
15. IEEE, *Draft amendment: ESS mesh networking*, IEEE P802.11s Draft 1.00 (2006).

16. N. Jain and S. Das, *A Multichannel CSMA MAC Protocol with Receiver-Based Channel Selection for Multihop Wireless Networks*, Proc. of the 9th Int. Conf. on Computer Communications and Networks (IC3N), 2001.
17. R. Karrer, A. Sabharwal, and Edward W. Knightly, *Enabling large-scale wireless broad-band: the case for TAPs*, Proc. of HotNets, 2003.
18. M. Kodialam and T. Nandagopal, *Characterizing the Capacity Region in Multi-Radio and Multi-Channel Mesh Networks*, Proc. ACM Mobicom, 2005.
19. J. Kratochvil, Z. Tuza, and M. Voigt, *New Trends in the Theory of Graph Colorings: Choosability and List Coloring*, Contemporary trends in discrete mathematics, DIMACS Series in Discrete Mathematics and Theoretical Computer Science (1999).
20. R. Maheshwari, H. Gupta, and S. R. Das, *Multichannel MAC Protocols for Wireless Networks*, Proc. of SECON, 2006.
21. T. Makansi, *Transmitter-oriented code assignment for multihop packet radio*, IEEE Transactions on Communications (1987).
22. M. V. Marathe, H. Breu, H. B. Hunt, S. S. Ravi, and D. J. Rosenkrantz, *Simple Heuristic for Unit Disk Graphs*, Networks (1995).

23. M. K. Marina and S. R. Das, *A Topology Control Approach for Utilizing Multiple Channels in Multi-Radio Wireless Mesh Networks*, Proc. of BroadNets, 2005.
24. A. Nasipuri and S. R. Das, *Multichannel CSMA with Signal Power-based Channel Selection for Multihop Wireless Networks*, Proc. of IEEE Vehicular Technology Conference (VTC), 2000.
25. A. Nasipuri, J. Zhuang, and S. R. Das, *A Multichannel CSMA MAC Protocol for Multihop Wireless Networks*, Proc. of IEEE Wireless Communications and Networking Conference (WCNC), 1999.
26. K. Ramachandran, E. M. Belding, K. C. Almeroth, and M. M. Buddhikot, *Interference-Aware Channel Assignment in Multi-Radio Wireless Mesh Networks*, Proc. of IEEE Infocom, 2006.
27. S. Ramanathan, *A Unified Framework and Algorithm for (T/F/C)DMA Channel Assignment in Wireless Networks*, Proc. of IEEE Infocom, 1997.
28. A. Raniwala and T. Chiueh, *Architecture and Algorithms for an IEEE 802.11-based Multi-Channel Wireless Mesh Networks*, Proc. IEEE Infocom, 2005.
29. A. Raniwala, K. Gopalan, and T. Chiueh, *Centralized Channel Assignment and Routing Algorithms for Multi-Channel Wireless Mesh Networks*, ACM Mobile Computing and Communications Review (MC2R) 8 (2004).

30. G. Sharma, R. R. Mazumdar, and N. B. Shroff, *On the Complexity of Scheduling in Wireless Networks*, Proc. ACM Mobicom, 2006.
31. J. Shi, T. Salonidis, and E. Knightly, *Starvation Mitigation Through Multi-Channel Coordination in CSMA Multi-hop Wireless Networks*, Proc. of ACM MobiHoc, 2006.
32. M. Shin, S. Lee, and Y.-A. Kim, *Distributed Channel Assignment for Multi Radio Wireless Networks*, (2005).
33. J. So and N. Vaidya, *Multi-Channel Mac for Ad Hoc Networks: Handling Multi-Channel Hidden Terminals Using a Single Transceiver*, Proc. ACM MobiHoc, 2004.
34. A. P. Subramanian, H. Gupta, and S. R. Das, *Minimum interference channel assignment in multi-radio wireless mesh networks*, Tech. report, WINGS Lab, SUNY at Stony Brook, 2006.
35. J. Tang, G. Xue, and W. Zhang, *Interference-Aware Topology Control and Qos Routing in Multi-Channel Wireless Mesh Networks*, Proc. of ACM MobiHoc, 2005.
36. Tropos Networks, <http://www.tropos.com>.
37. R. Vedantham, S. Kakumanu, S. Lakshmanan, and R. Sivakumar. *Component*

*Based Channel Assignment in Single Radio, Multi-channel Ad Hoc Networks*, Proc. ACM Mobicom, 2006.

38. S.-L. Wu, C.-Y. Lin, Y.-C. Tseng, and J.-P. Sheu, *A New Multi-Channel Mac Protocol with On-Demand Channel Assignment for Multi-hop Mobile Ad Hoc Networks*, Proc. International Symposium on Parallel Architectures, Algorithms and Networks (I-SPAN), 2000.