Origins and Solutions of Proportional Unfairness in IEEE 802.11 Multi-hop Wireless Networks

by

Jingpu Shi

A THESIS SUBMITTED IN FULL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE

Doctor of Philosophy

APPROVED, THESIS COMMITTEE:

[Signatures]

Dr. Edward W. Knightly, Chair
Professor
Electrical and Computer Engineering

Dr. David B. Johnson
Associate Professor
Computer Science

Dr. Ashutosh Sabharwal
Assistant Professor
Electrical and Computer Engineering

Houston, Texas

December, 2007
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.
Origins and Solutions of Proportional Unfairness in IEEE 802.11 Multi-hop Wireless Networks

Jingpu Shi

Abstract

Proportional unfairness exists in multi-hop wireless networks where IEEE 802.11 is employed as the medium access control protocol. Some flows may receive very high throughput while some others may receive very low or even zero throughput. This thesis investigates the origins of unfairness and proposes solutions to improve fairness in such networks.

To analyze unfair MAC contention, this thesis first studies multi-hop topologies consisting of two disjoint flows and four nodes. All possible topologies depending on whether the different nodes are within range of each other are systematically identified and further classified into three groups. We show that in the first group the two senders are in radio range of each other and contend fairly. However, long-term unfairness arises in the second group where the two senders do not have complete information of the channel and the topology is asymmetric. In the third group where the channel information each flow obtains is incomplete but the topology is symmetric, the system achieves long-term fairness yet endures significant durations in which one flow dominates channel access before relinquishing the channel. We develop analytical models to characterize the MAC performance in the groups where either long- or short-term unfairness arises.

To analyze the compounding effect of MAC and congestion control on fairness, we identify an important topology that is necessarily embedded in mesh networks, in which a one-hop TCP flow contends with a two-hop TCP flow for gateway access. We describe how the congestion control and the MAC jointly lead to unfairness between the two flows, and analytically model the system with a six-dimensional Markov chain model. Motivated
by the model, we devise and evaluate a simple Contention Window Policy that only requires the one-hop mesh nodes to increase their minimum contention window. Extensive experiments and simulations show that regardless of its simplicity, this window policy has a powerful effect in improving fairness of the network.

To address unfairness in mobile ad-hoc networks, we devise a distributed multi-channel medium access protocol, Asynchronous Multichannel Coordination Protocol (AMCP), that not only increases aggregate throughput, but more importantly, addresses the fundamental coordination problems that arise in a multi-hop network. AMCP uses a dedicated control channel for control message exchange and multiple data channels for data transmissions. We analytically derive and experimentally validate a lower throughput bound for any flow in an arbitrary topology under the operation of AMCP.
Acknowledgments

I would like to thank many people who have guided, helped and supported me on the path toward where I am today. Without them, this thesis would not be possible.

I am deeply thankful to my advisor Dr. Knightly. He is an excellent professor, wonderful advisor, and an efficient manager. His insights guided me through difficult times when I was lost in research and did not know where I was going. His high standard in research and publication pushed me to challenge the status quo and to think out of the box. His help and support were essential in my completion of this thesis!

I am also thankful to Dr. Sabharwal and Dr. Johnson for their willingness to serve on my thesis committee. I thank them for their discussions, comments, guidance, and the hard time they gave me in my defense!

I was extremely lucky to have Omer Gurewitz as my colleague. Right after he arrived, he challenged me in every assumption I made in my work. The fight then quickly turned into great collaborations. Omer is a faithful friend, a great collaborator and an exceptional researcher. Thanks Omer, for your great collaborations and tremendous help!

My collaborations with colleagues played a big part in my dissertation work. During the past several years, I extensively worked with Michele Garetto, Theodoros Salonidis, Vincenzo Mancuso and Joseph Camp. Their hard work, professionalism, and great insight made our collaborations very productive, successful and enjoyable.

Rice Networks Group (RNG) is a supportive and dynamic group. We discussed new ideas and presented research proposals in weekly research meetings. The group members are enthusiastic and have great insight into networking research. I would like to thank all RNG members I have not mentioned so far: Ehsan Aryafar, Anastasios Giannoulis, Ahmed Khattab, Eugenio Magistretti, Stanislav Miskovic, Joshua Robinson and Jasmin Shrestha.
The following friends spent a great amount of their time helping me prepare for my defense presentation and take care of the logistics: Yanjun Sun, Vincenzo Mancuso, Ehsan Aryafar and Alireza Keshavarz-Haddad. The Chinese community at Rice also deserves my thankfulness. My days at Rice would be much less enjoyable without Yang Sun, Yanjun Sun, Bo Zhang and Guohui Wang.

Heather Dodge and Dee Rashdi helped me smoothly navigate through the administrative procedures. As the graduate coordinator of my department, Heather effectively helped me meet deadlines, fill out forms appropriately, set up the teleconference system for my defense, and much more. Due to her expertise in the graduate study policies at Rice, I spent as less time as possible dealing with administrative issues and was able to stay focused on my study and research.

All the efforts I spent in my PhD program became worthy because of my family. I thank my mother Yuqing Zhang, my father Dengyun Shi and my two sisters Jingqin Shi and Jing Shi for their self-sacrificing love. They are always there for me in both high points and low points of my life. Finally, special thanks go to my wife Lifei Zhang. Her unconditional love not only supported and encouraged me in every aspect of my life, but also shaped me. I know that her love to me will remain the same even if the sea becomes empty and the stone decays. In my last days at Rice, my son Andy Shi came to this world. He not only brought me joy, but kept me going in difficult times. In desperation I sometimes lost all motives to get a PhD except this: A PhD will allow me to buy more diapers for my baby!

I would like to thank the group who share the same faith with me: Trae Vacek, Chris Sneller, William Chan, Dajing Liu, Bibo Jiang, Yu Lin and Jiashun Sheng. American Chinese Fellowship is my wonderful faith family, where my brothers and sisters help each other and rejoice together. Thank you all for your spiritual support!

Finally, I believe that God who created the universe is the source of all wisdom. “The fear of the LORD is the beginning of wisdom; all who follow his precepts have good understanding. To him belongs eternal praise.” (Psalm 111: 10).
Contents

Abstract i
Acknowledgments iii
List of Illustrations viii
List of Tables xiii

1 Introduction 1
1.1 Analysis of Unfair MAC Contention 2
1.2 Analysis of Joint Effect of MAC and Congestion Control on Fairness 4
1.3 Solutions to Improve Fairness 5
1.4 Contributions 7
1.5 Thesis Overview 8

2 Analysis of MAC Protocol Origins of Unfair Contention 9
2.1 Topology & Network Geometry 10
   2.1.1 Two Flow Topologies 10
   2.1.2 Spatial Analysis 12
   2.1.3 Comparing Scenario Likelihood 15
2.2 Impact of MAC Protocols 19
2.3 Asymmetric Incomplete State 25
   2.3.1 General Decoupling Model of an 802.11 Station 26
   2.3.2 Analysis of AIS Flows 29
   2.3.3 Non-Continuously-Backlogged Flows 33
   2.3.4 Model Validation Through Simulations 35
   2.3.5 Model Validation Through Experiments 39
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.4</td>
<td>Symmetric Incomplete State and Short Term Unfairness</td>
<td>41</td>
</tr>
<tr>
<td>2.4.1</td>
<td>Analytical Model</td>
<td>42</td>
</tr>
<tr>
<td>2.4.2</td>
<td>Simulations and Model Validation</td>
<td>46</td>
</tr>
<tr>
<td>2.5</td>
<td>Related Work</td>
<td>49</td>
</tr>
<tr>
<td>2.6</td>
<td>Summary</td>
<td>51</td>
</tr>
<tr>
<td>3</td>
<td>Compounding Effect of MAC and Congestion Control</td>
<td>52</td>
</tr>
<tr>
<td>3.1</td>
<td>Analysis</td>
<td>53</td>
</tr>
<tr>
<td>3.1.1</td>
<td>Broader Topology</td>
<td>57</td>
</tr>
<tr>
<td>3.2</td>
<td>Verification</td>
<td>57</td>
</tr>
<tr>
<td>3.2.1</td>
<td>Experiment Results</td>
<td>58</td>
</tr>
<tr>
<td>3.3</td>
<td>Modeling</td>
<td>60</td>
</tr>
<tr>
<td>3.3.1</td>
<td>System Model</td>
<td>61</td>
</tr>
<tr>
<td>3.3.2</td>
<td>Model Description</td>
<td>62</td>
</tr>
<tr>
<td>3.3.3</td>
<td>Transition Probability Computation</td>
<td>64</td>
</tr>
<tr>
<td>3.3.4</td>
<td>Throughput Computation</td>
<td>66</td>
</tr>
<tr>
<td>3.3.5</td>
<td>Model Validation Through Simulations</td>
<td>67</td>
</tr>
<tr>
<td>3.3.6</td>
<td>Model Validation Through Experiments</td>
<td>67</td>
</tr>
<tr>
<td>3.4</td>
<td>Discussion</td>
<td>68</td>
</tr>
<tr>
<td>3.5</td>
<td>Related Work</td>
<td>69</td>
</tr>
<tr>
<td>3.6</td>
<td>Conclusion</td>
<td>71</td>
</tr>
<tr>
<td>4</td>
<td>Minimum Contention Window Policy</td>
<td>72</td>
</tr>
<tr>
<td>4.1</td>
<td>Minimum Contention Window Policy</td>
<td>72</td>
</tr>
<tr>
<td>4.2</td>
<td>Evaluation of the Minimum Contention Window Policy</td>
<td>74</td>
</tr>
<tr>
<td>4.2.1</td>
<td>The Basic Topology</td>
<td>74</td>
</tr>
<tr>
<td>4.2.2</td>
<td>Extending the Basic Topology: Two Branches</td>
<td>78</td>
</tr>
<tr>
<td>4.2.3</td>
<td>Downstream Traffic</td>
<td>78</td>
</tr>
<tr>
<td>4.2.4</td>
<td>Larger Scenarios via Simulation</td>
<td>80</td>
</tr>
</tbody>
</table>
4.3 Related Work .......................................................... 82
4.4 Conclusion ............................................................. 82

5 Asynchronous Multi-channel Coordination Protocol 83
  5.1 Motivation and protocol design issues .............................. 84
    5.1.1 Unfairness in IEEE 802.11 Single-channel Multi-hop Wireless Networks ............................................. 84
    5.1.2 Fairness Improvement Through Multiple Channels ............. 87
    5.1.3 Multi-channel Coordination Problems .......................... 88
  5.2 Asynchronous Multi-channel Coordination Protocol (AMCP) ........ 91
    5.2.1 Overview ......................................................... 91
    5.2.2 Protocol Description ............................................ 92
    5.2.3 Addressing Multi-channel Coordination Problems ............. 94
  5.3 Protocol Analysis .................................................. 97
    5.3.1 Bottleneck Analysis ............................................. 97
    5.3.2 Lower Bound Analysis .......................................... 98
  5.4 Performance Evaluation ........................................... 101
    5.4.1 Single-hop Topologies ......................................... 102
    5.4.2 Multi-hop Topologies .......................................... 105
  5.5 Related work ........................................................ 112
  5.6 Conclusion .......................................................... 113

6 Conclusions ........................................................... 114

Bibliography ............................................................ 116
Illustrations

2.1 Twelve scenarios for two flows sharing a wireless channel. 11
2.2 Node placement and topology of Scenario 11. The two circles centered at
A and a indicate respective transmission range of station A and a. The
circle centered at B indicates possible positions of station b given the
distance between B and b is fixed. 13
2.3 Different locations of station B lead to different computations for $\beta$. 15
2.4 Different value of $d_2$ decides how to further divide $T(\bar{A} \cap a)$ into different
regions. 16
2.5 Probability of each of the eleven scenarios occurring as a function of
normalized hop distance. 17
2.6 Probabilities of the two groups, assuming identical transmission and
sensing range. 18
2.7 Hop distance distribution in a multi-hop wireless network. 19
2.8 Probabilities of the two groups, in case of different transmission and
sensing range. 20
2.9 Short-term normalized throughput for the twelve scenarios. The ‘×
’marks correspond to two-way handshake simulations; the dot marks
correspond to four-way handshake simulations. 21
2.10 Example topologies of the AIS class. 23
2.11 Example topologies of the SIS class. (b) is a special case of (a), when two Mobile Units (MU1 and MU2) are sending packets simultaneously to the same Access Point (AP) in infrastructure mode, as encountered in the classic "hidden terminal" problem. ......................................... 24
2.12 Flow throughput comparison between a 10 second snapshot and a 120 second snapshot. ................................................................. 25
2.13 A station's channel view and embedded discrete time renewal process. .... 26
2.14 Channel occupation state as perceived by node a while sender A attempts to initiate a new data transfer. The top diagram (a) refers to Scenario 12, the bottom (b) to Scenario 11. ......................................................... 30
2.15 Throughput of Flow A vs. data payload size (with RTS/CTS). ............ 36
2.16 Throughput of Flow B vs. data payload size (with RTS/CTS). ............ 37
2.17 Throughput of Flow A vs. data payload size (basic access). ............... 38
2.18 Throughput of Flow A vs. arrival rate of flow B (with RTS/CTS, Scenario 12). .......................................................... 39
2.19 Throughput of Flow A vs. arrival rate of flow B (basic access, Scenario 11). 40
2.20 Connectivity graph of the TFA backhaul with appropriate scaling for distance between nodes. ................................................. 40
2.21 Model predictions vs. measurement results. (with RTS/CTS, scenario 11). 41
2.22 Steady state probabilities of the Markov model in case of RTS/CTS mechanism, default 802.11b parameters. ............................. 45
2.23 Packets successfully sent by one flow every 100 ms, in all four cases. .... 48
2.24 Comparison of the number of packets sent by one flow during intervals of 100 ms in scenario 10, with or without RRTS .......................... 50

3.1 The basic topology ......................................................... 53
3.2 TCP DATA and TCP ACK are contending for channel access. ............ 54
3.3 Illustration of bi-stability with alternation of \((A, B)\) and \((B, GW)\) transmissions. ................................................................. 55

3.4 Illustration of multiple control loops and a shared medium. .............. 56

3.5 Two-branch scenario. \(A \rightarrow B \rightarrow GW\) is one branch. \(C \rightarrow GW\) is another branch of the mesh. ................................................................. 57

3.6 Connectivity graph of the TFA network backhaul topology with appropriate scaling for distance between nodes. There are nearly 2,500 residential users within the coverage area. Nodes \(A, B\) and \(GW\) are the nodes used throughout the measurements as the basic topology within the TFA network. Node \(C\) used for the second branch topology discussed in Section 3.1.1. ................................................................. 58

3.7 Two-hop chain contention with RTS/CTS off measured at six different times of the day. ................................................................. 59

3.8 Two-branch contention with RTS/CTS on. ........................................ 61

3.9 Queues at different mesh points. ....................................................... 62

3.10 Illustration of channel states. ........................................................... 63

3.11 Analytical model predictions compared to simulation results. ............ 68

3.12 Analytical model predictions compared to measurement results. ........ 69

4.1 Queue behavior as a function of the minimum contention window of \(B\). ... 74

4.2 RTS/CTS enabled: unfairness with default \(CW_{\text{min}}\) and Minimum Contention Window Policy result in the basic scenario of MirrorMesh. ... 75

4.3 RTS/CTS mechanism disabled: unfairness with default \(CW_{\text{min}}\) and Minimum Contention Window Policy result in the basic scenario of MirrorMesh. ................................................................. 76

4.4 Unfairness and Minimum Contention Window Policy result in a two-hop chain of MirrorMesh with the most common default minimum contention window setting for commercial devices \((CW_{\text{min}} = 32)\). ................................. 77
4.5 Unfairness and Minimum Contention Window Policy result with 500 Byte
TCP packets. Left: RTS/CTS on, right: RTS/CTS off. ....................... 77
4.6 Two branch topology. TCP flow A is a two-hop flow. TCP flow B and C
are two one-hop flows. ............................................. 78
4.7 Unfairness in a two-hop chain of MirrorMesh with default and increased
contention windows at node B and C. ................................ 79
4.8 Unfairness and Minimum Contention Window Policy result in a
downstream two-hop chain of MirrorMesh nodes. ..................... 79
4.9 TCP flows in four-hop chain topology. ............................... 81
4.10 TCP throughput in a four-hop chain topology, with minimum contention
windows all 32 (IEEE 802.11) and the one hop neighbor changed to 128
(New window). .................................................. 81

5.1 Information Asymmetry (IA) example .................................. 85
5.2 Flow In The Middle (FIM) example .................................. 86
5.3 The use of multiple channels to address starvation. ................ 88
5.4 The Multi-Channel Hidden Terminal Problem. ....................... 89
5.5 Missing Receiver Problem. ......................................... 90
5.6 AMCP addressing the Multi-channel Hidden Terminal problem. .... 96
5.7 Comparison of lower bound to minimum throughput achieved in an
arbitrary contention region as a function of the number of interfering flows. 101
5.8 Throughput as a function of number of channels. .................... 103
5.9 Aggregate throughput when arrival rate varies. ...................... 104
5.10 Effect of channel switching delay. .................................. 105
5.11 AMCP performance in the basic single-channel starvation scenarios. 107
5.12 Per-flow throughput in an arbitrary topology of single-hop backlogged
flows using 12 channels. .......................................... 108
5.13 Download scenario. .............................................. 109
5.14 Per-flow throughput in an arbitrary topology of multi-hop flows using 12 channels.
Tables

2.1 The value of D to be used in (2.12) ................................. 32
2.2 Parameters setting for the MAC and physical layers .............. 36
2.3 Transition probabilities of the Markov Model .................. 43
2.4 Average throughput per flow $T$ (in pkt/s), conditional packet loss
   probability $p$, and average transient time $\Delta t$ (in ms) obtained in Scenario 8
   for four different settings of parameters ............................ 47

3.1 Transition probabilities of the Markov Model, when one of the queues is
   empty ........................................................................ 65

4.1 Parameters setting for the MAC and physical layers ............ 80

5.1 MAC layer parameters ............................................... 102
Chapter 1

Introduction

Multi-hop wireless networks are wireless networks where not all nodes are within radio range of each other. Due to their ability to allow out-of-range devices to communicate via multi-hop paths consisting of intermediate forwarding nodes, multi-hop wireless networks are expected to lead in the next wave of wireless technology deployment. As an example, large-scale wireless mesh network deployments are planned and underway in cities across the world. According to In-Stat, the market will grow from 248 cities in 2005 to 1,500 cities in 2010.

It is desired that a multi-hop wireless network is capable of providing fairness to flows. There are various definitions for fairness, among which proportional fairness is one of the most commonly used in the wireless research community, introduced by Kelly in [34] as follows. Consider a network where there are a set $S$ of flows. A rate vector $(x_i^*, i \in S)$ is proportionally fair if it is feasible and if for any other feasible rate vector $(x_i, i \in S)$, the sum of the proportional change is non-positive:

$$\sum_{i \in S} \frac{x_i - x_i^*}{x_i^*} \leq 0. \quad (1.1)$$

It has been shown in [34] that a proportional rate vector $(x_i^*, i \in S)$ is a maximizer of the optimization problem $\sum_{i \in S} \log(x_i)$ over the feasible region. Regardless of many definitions for fairness, this thesis focuses on proportional fairness. Unless otherwise specified, the term fairness used throughout this thesis refers to proportional fairness.

In today’s multi-hop wireless networks, severe unfairness exists [6, 11, 24] due to wireless nodes contending with each other unfairly. Since flows within each other’s transmission range share the same wireless medium, if two or more of them have packets to transmit at the same time, they need to contend for channel access. The contention mechanism is
provided by the Medium Access Control (MAC) protocol. Among existing MAC protocols, the IEEE 802.11 protocol [22] is widely deployed in both single-hop and multi-hop wireless networks. While the IEEE 802.11 protocol allows nodes to contend fairly in single-hop wireless networks, where all wireless nodes are within the radio range of each other and thus obtain the same channel state, in a multi-hop wireless network, it has been known that nodes can contend unfairly with each other [6, 32]. As a result of this unfair contention, some flows may receive very low or even zero throughput. As we will show in this thesis, even in a multi-hop network consisting of only two flows, the achieved rate by IEEE 802.11 can be far off from the desired proportional fair rate.

Clearly, for the multi-hop wireless technology to succeed, it is critical that unfairness in such networks is understood and addressed. The objective of this thesis is to investigate unfairness in IEEE 802.11 multi-hop wireless networks and propose solutions to improve fairness. More specifically, this thesis will

- demonstrate the existence of unfairness in multi-hop topologies consisting of only two flows.

- analyze the origins of unfairness.

- analytically model unfairness.

- propose solutions to improve fairness.

- implement and evaluate the proposed solutions.

1.1 Analysis of Unfair MAC Contention

The performance of IEEE 802.11 is well understood in single-hop wireless networks [8, 9, 37, 45], where all wireless nodes are within range of each other. In such networks, all nodes have the same and complete view of the channel state and therefore contend fairly with each other. The throughput each flow receives is a function of the total number of flows in the network and can be accurately predicted using existing technologies [8].
In contrast, in multi-hop wireless networks different wireless nodes may obtain different channel information and thus contend unfairly with each other. The topology of the network can become a dominating factor in deciding how the flows contend and what throughput they receive. For example, consider a scenario where flow $A$ is contending with flow $B$ for channel access. Flow $A$ may receive throughput close to the maximum channel capacity or close to zero, depending on the topology that flow $A$ and flow $B$ form.

This thesis systematically characterizes the performance of IEEE 802.11 protocol in multihop topologies consisting of four nodes and two disjoint contending flows. For two flows, there are twelve distinct scenarios, according to whether the different nodes are in radio range of each other. These scenarios are further classified into three groups: Senders Connected (SC) group where the two senders of the two flows are within radio range, Asymmetric Incomplete State (AIS) group where the two senders are not within radio range and the topology is asymmetric, and finally Symmetric Incomplete State (SIS) where the two senders are not within radio range and the topology is symmetric. Simulations are performed to characterize the MAC performance in each of the twelve scenarios. The simulation results demonstrate that severe long-term starvation (i.e., asymptotic low throughput) arises in the AIS class, and the SIS class incurs short-term starvation (i.e., low throughput within a certain time interval) and exhibits bi-stability: in one stable-state, one flow captures the channel and starves the other flow; after a while the system switches to the other stable state in which the other flow captures the channel.

This thesis then develops analytical models to study the performance of IEEE 802.11 protocol in the two-flow scenarios where unfairness arises. It proposes a novel model in which different nodes have their private view of the channel state. This model captures long-term unfairness that arises in the AIS class. A two-dimensional Markov model is developed to characterize the SIS class. This model captures the time scale of short-term unfairness as a function of key protocol parameters.
1.2 Analysis of Joint Effect of MAC and Congestion Control on Fairness

Sliding window congestion control algorithms such as TCP are designed to alleviate congestion in the network. Such an algorithm impacts the sending rate of the MAC protocol and thus may change the unfairness profile of the network. In the literature, it is known that TCP does not perform well in multi-hop wireless networks and magnifies the MAC problems [14, 26]. In [16], the authors conclude that the poor performance of TCP is due to its congestion window being larger than its optimal value, which is determined to be one third of the number of hops from the source to the destination.

This thesis performs analysis on two TCP flows contending with each other on top of IEEE 802.11 protocol and shows that, even the TCP window is fixed to its optimal value suggested in [16], TCP can still perform poorly and lead to unfairness. To analyze the joint impact of TCP congestion control algorithm and the IEEE 802.11 MAC on unfairness, this thesis considers a scenario named the basic topology, which can not be avoided in a mesh network. In the basic topology, a two-hop flow shares the same gateway with a one-hop flow. This thesis shows that severe unfairness arises in this basic topology: the two-hop TCP flow receives close to zero throughput while the one-hop flow receives most of the gateway bandwidth. The compounding effect of MAC and congestion control on unfairness is characterized as below.

First, as the analysis for the two-flow scenarios suggests, the medium access protocol induces bi-stability in which pairs of nodes alternate in capturing system resources. The transport protocol, specifically, the DATA-ACK control loop, makes the system spend much longer time in one stable state than in the other stable state and favors the one-hop TCP flow. Finally, the two-hop flow’s transmitter often incurs a high penalty in terms of loss, delay, and consequently, throughput, in order to re-capture system resources.

An analytical model is developed to study the unfairness in the basic topology. The model omits many intricacies of the system (TCP slow start, fading channels, channel coherence time, etc.) and instead focuses on the minimal elements needed such that un-
fairness manifests. Namely, the model uses a discrete-time Markov chain embedded over continuous time to capture a fixed end-to-end congestion window, a carrier sense protocol, and all end-point and intermediate queues. Validated through simulations, the model predicts unfairness in the basic scenario with the one-hop TCP flow receiving much larger throughput than the two-hop TCP flow. Since the model only considers a fixed congestion control window, we conclude that the DATA-ACK control loop and a sliding window alone are sufficient to induce unfairness, even if the TCP window is fixed to its optimal value suggested in [16].

1.3 Solutions to Improve Fairness

Our analysis shows that in multi-hop IEEE 802.11 wireless networks, when contending for channel access, some nodes may have advantages over others. If the transmissions from both the advantaged and disadvantaged flow occur in the same frequency and overlap in time, unfairness will arise. To improve fairness, this thesis explores solutions to either shift over time flow transmissions that would contend unfairly or distribute these transmissions to different frequency bands. Taking advantage of the fact that in mesh networks the traffic are to and from the gateway, we devise a simple contention window policy to improve fairness. For mobile ad-hoc networks, we design a multi-channel protocol that not only improves the aggregate throughput of the network, but more importantly, improves the throughput of those flows that are disadvantaged when contending with other flows on one channel.

The model used to analyze the basic scenario yields a contention window policy to improve fairness in mesh networks. According to this policy, the gateway’s directly-connected neighbors should increase their minimum contention window to a value greater than that of other nodes. The model also characterizes why the policy is effective in that it forces all queuing to occur at the gateway’s one-hop neighbors rather than elsewhere. Because these nodes have a perfect channel view of both the gateway and their neighbors that are two-hop away from the gateway, MAC bi-stability is eliminated such that the subsequent penalties
are not incurred. This thesis evaluates through extensive experiments and simulations the effectiveness of the contention window policy in improving fairness.

To mitigate unfairness in ad-hoc networks, this thesis explores the use of multiple channels under the constraints of a single half-duplex radio at each node. Multi-channel wireless technologies have the potential to improve fairness by distributing the flow transmissions that would contend unfairly on the same channel to different channels. Still, achieving this goal is challenging, especially when each node can transmit or receive on only a single channel and link at a time. This results in the following generic multi-channel coordination problems: (1) Control packets sent on a certain channel fail to inform neighboring nodes currently communicating on a different channel and (2) Control packets intended for a certain receiver may fail because the receiver is currently on a different channel. In light of the above coordination problems we revisit the basic design principles of multi-channel MAC protocols. A fundamental design choice is whether to use a dedicated control channel or transmit both control and data information on all channels. AMCP utilizes a dedicated control channel to address both single-channel and multi-channel coordination problems. To combat the bottleneck caused by the control channel, we compute the maximum number of data channels that can be supported by the control channel as a function of the protocol parameters. This allows one to quantitatively perform appropriate sizing on the control channel capacity.

This thesis then derives a lower bound on the throughput of any AMCP flow in an arbitrary topology. The basic technique is to construct a hypothetical, low-throughput scenario on the control channel and to model the impact of the aggregate channel hopping pattern of the interfering flows. The lower bound depends on system parameters and the number of interfering nodes within the neighborhood of each flow. Therefore, it can be computed using only local information. Through extensive simulations we demonstrate that the throughput achieved by AMCP can approach the lower bound in highly congested contention regions while being much higher in multi-hop scenarios. We design experiments to isolate and expose each fundamental single- and multi-channel coordination problem and show how
AMCP addresses these issues and describe why existing multi-channel solutions do not. As AMCP switches channels at packet level, we evaluate via simulations the performance degradation of AMCP due to channel switching delay.

1.4 Contributions

This thesis makes the following contributions.

- Discovery of minimal multi-hop scenarios in which unfairness manifests.
  
  This thesis enumerates all twelve distinct scenarios consisting of two disjoint flows, and shows that MAC-induced short- or long-term unfairness exists in scenarios where senders are disconnected and thus obtain incomplete channel state. This thesis also identifies the basic topology in mesh networks and shows that the congestion control and the MAC mechanism can jointly induce unfairness. Identification of these simple scenarios enables one to propose tractable analytical models to gain insights how protocol inefficiencies lead to unfairness or flow starvation.

- Novel mathematical models to characterize multi-hop wireless networks.
  
  Three mathematical models are developed in this thesis to characterize the MAC and congestion control mechanisms in multi-hop wireless networks. In these models, each node has its own private view of the wireless channel. The first model characterizes long-term unfairness due to contending flows having incomplete and asymmetric channel information. The second model captures short-term unfairness due to contending flows having incomplete but symmetric information. The third model captures a fixed end-to-end congestion window, a carrier sense protocol, and all endpoint and intermediate queues. These models analytically characterize unfairness in simple multi-hop wireless topologies, provide insights how to improve fairness, and enable others to predict flow throughput in arbitrary IEEE 802.11 multi-hop wireless networks [19].
• A novel multi-channel protocol to improve fairness.

Another contribution of this thesis lies in the design of AMCP, a distributed multi-channel protocol. Distributed contention-based multi-channel protocols have been proposed in [2, 28, 41, 48, 53]. All these protocols have focused on increasing aggregate network throughput. However, with increased aggregate throughput, some flows may still receive very low or even zero throughput. In contrast, AMCP's counter-starvation mechanism provides minimum rate guarantee and effectively improves the throughput of the starving flows.

• A novel contention window policy to improve fairness.

This thesis proposes a Minimum Contention Window Policy to mitigate starvation in mesh networks. This policy can be realized via mechanisms in standard protocols such as IEEE 802.11e. It only requires modification of a single MAC parameter, but has a power effect in improving fairness.

1.5 Thesis Overview

The rest of the thesis is organized as follows. Chapter 2 characterizes the MAC protocol origins of unfairness. Chapter 3 analyzes the compounding effect of the congestion control mechanism and MAC mechanism on unfairness. A contention window policy to improve fairness for mesh networks is described and validated in Chapter 4. Chapter 5 presents a distributed multi-channel protocol to improve fairness for mobile ad-hoc networks. Chapter 6 concludes the thesis.
Chapter 2

Analysis of MAC Protocol Origins of Unfair Contention

In an IEEE 802.11 single-hop wireless network where all nodes are within radio range, the evolution of the wireless channel is promptly learned by all other nodes in the network through Medium Access Control mechanisms, by either virtual carrier sensing or physical carrier sensing. Therefore, all nodes in the network obtain the same and complete channel state and contend fairly with each other. The performance of IEEE 802.11 in such networks has been well understood [8]. This chapter shows, however, in a multi-hop wireless network where not all nodes are within radio range, when the wireless channel evolves, the new channel state may not be learned by all other nodes in the network. As a result, nodes in a multi-hop wireless network obtain incomplete and inconsistent channel information, which lead to some nodes having advantages over others when they contend for channel access.

This chapter studies the unfair contention among wireless nodes under the IEEE 802.11 protocol family in a multi-hop wireless network. To understand the characteristics of the MAC protocol in such a network, we study the multi-hop topology consisting of four nodes and two contending flows. We enumerate the twelve distinct scenarios and classify them into three groups based on their geometric properties. We develop a spatial model that predicts the likelihood of the scenarios and groups when nodes are placed randomly. Moreover, we use simulations with random waypoint mobility and routing protocols to further characterize scenario likelihood. We then perform simulations to characterize the short- and long-term unfair contention between the two flows in each of the three classes. We develop analytical models to characterize the classes where either long- or short-term unfair contention arises.
2.1 Topology & Network Geometry

In this section, we first identify all feasible topologies in which exactly two directional flows are communicating. Next, using a grouping of these topologies, we perform a spatial analysis that characterizes the likelihood of each of these topologies occurring under random node placement. Finally, we compare scenario likelihood via the model and simulations that incorporate mobility.

2.1.1 Two Flow Topologies

We consider four stations that are communicating pairwise, where two senders are transmitting a one-way data flow to their two respective receivers. When two stations are within radio range of each other (i.e., the received SNR is above the carrier sense threshold) we refer to them as having a connection or link between them. Depending on the distances and propagation paths between the four stations, a link may or may not be established between the two flows.

Denote station $A$ as the sender for flow $A$ and station $B$ as the sender for flow $B$, and stations $a$ and $b$ as their respective receivers. Links are named by using the names of the stations that they interconnect. There are four possible inter-flow links: $AB$, $ab$, $aB$ and $Ab$. In a general topology, each one of these links may be present or not, yielding $2^4 = 16$ different scenarios. Notice that links $Aa$ and $Bb$ are always present, given that senders are connected with their respective receivers. However, scenarios where link $Ab$ exists and link $aB$ does not are symmetric to those where $aB$ exists and $Ab$ does not. After ruling out four such cases, we have twelve distinct scenarios depicted in Figure 2.1. For example, Scenario 1 in Figure 2.1 depicts the case in which the two flows are out of radio range and hence operate independently.

Omitting Scenario 1 which is trivial to analyze, we classify the remaining eleven scenarios into three groups as follows.

- Senders Connected (SC) - Scenarios 2-7, in which link $AB$ is present.
Figure 2.1: Twelve scenarios for two flows sharing a wireless channel.

- Asymmetric Incomplete State (AIS) - Scenarios 11 and 12, in which senders are disconnected, and only one of links $Ab$ or $aB$ is present (we assume it is always $aB$), resulting in asymmetric connection between the two flows.

- Symmetric Incomplete State (SIS) - Scenario 8, 9 and 10, in which senders are disconnected, and either both $Ab$ and $aB$ are present or neither is present (in the latter case $ab$ is present) resulting in symmetric connection between the two flows.

We will show that the flow pairs obtain dramatically different performance profiles according to which of the three groups represents their topology. We demonstrate this using simulations for both two-way and four-way handshakes in Section 2.2 and via analytical models in Sections 2.3 and 2.4. Note that factors such as time-varying channel or transmission power may or may not affect the existence of a link between the two flows. If the SNR variation caused by these factors is not large enough to cause the received SNR to cross the
carrier sense threshold, the existence of the link is not affected. If the received SNR crosses
the carrier sense threshold, the topology of the two flows changes accordingly. However, at
a given time, the topology of the network is one of the scenarios shown in Fig 2.1 and we
characterize the performance of the network for that moment.

2.1.2 Spatial Analysis

Now we develop a model to compute the probability that each scenario in Figure 2.1 occurs
in a random graph. Our approach is to view a scenario as three joint events, and to conduct
a spatial analysis to compute the probability of each of these events occurring. We assume
the four stations are uniformly distributed in the network and that the size of the network is
large enough so that border effects are negligible. We describe in detail only the derivation
for Scenario 11. The same approach can be applied to predict the occurrence probabilities
of the other scenarios.

We consider a simplified propagation model in which \( r \) is the radio range of a station.*
Let \( s \) denote the physical area (size) of the network and \( T(\cdot) \) represent a region of the plane
satisfying certain geometric conditions. In particular, \( T(A) \) indicates the region within
radio range of station \( A \) and \( T(\bar{A} \cap B) \) represents the region within radio range of \( B \) but
outside that of \( A \). The distance between station \( A \) and \( a \) is denoted by \( d_1 \). Similarly, the
distance between station \( B \) and \( b \) is denoted by \( d_2 \). The coordinates of station \( z \) are denoted
by \( x_z \) and \( y_z \), respectively, where \( z \in \{ A, a, B, b \} \).

Node placement for Scenario 11 is shown in Figure 2.2, where the angle \( \beta \) indicates
whether station \( b \) is within \( T(A \cup a) \). Scenario 11 can be decomposed into the following
three events: (i) \( d_1 \leq r \); (ii) station \( B \) is within \( T(\bar{A} \cap a) \) given that the first event occurs;
(iii) \( d_2 \leq r \) and station \( b \) is within \( T(\bar{A} \cap \bar{a}) \) given that the first two events occur. Note that
due to the symmetry of region \( T(\bar{A} \cap a) \), we only need to consider cases where \( y_B \geq 0 \) and
then apply a multiplicative factor of 2.

*For simplicity, we assume that the transmission range is equal to the sensing range for all stations. We
remove this assumption later.
Figure 2.2: Node placement and topology of Scenario 11. The two circles centered at $A$ and $a$ indicate respective transmission range of station $A$ and $a$. The circle centered at $B$ indicates possible positions of station $b$ given the distance between $B$ and $b$ is fixed.

We now compute the probability of each of the three events occurring. In event (i), the probability of station $a$ falling into a doughnut area comprised in the interval $[d_1, d_1 + \Delta d_1]$ with $A$ at the center is given by

$$\lambda_1 = \frac{2\pi(d_1 + \Delta d_1)^2 - 2\pi d_1^2}{s} = \frac{2\pi \Delta d_1^2 + 4\pi d_1 \Delta d_1}{s}. \quad (2.1)$$

When $\Delta d_1 \to 0$ we can neglect the second order term, obtaining

$$\lambda_1 = \frac{4\pi d_1 \Delta d_1}{s}. \quad (2.2)$$

In event (ii), the probability of station $B$ falling into a small square region in $T(\bar{A} \cap a)$ defined by the interval $[x_B, x_B + \Delta x_B]$ on the $x$ axis and $[y_B, y_B + \Delta y_B]$ on the $y$ axis is given by

$$\lambda_2 = \frac{\Delta x_B \Delta y_B}{s}. \quad (2.3)$$

In event (iii), the joint probability that the distance between station $B$ and $b$ is within the interval $[d_2, d_2 + \Delta d_2]$ and station $b$ is within $T(\bar{A} \cap \bar{a})$, is given by

$$\lambda_3 = \frac{1 - \beta}{2\pi} \times \frac{2\pi(d_2 + \Delta d_2)^2 - 2\pi d_2^2}{s} = \frac{1 - \beta}{2\pi} \times \frac{2\pi \Delta d_2^2 + 4\pi d_2 \Delta d_2}{s}, \quad (2.4)$$
where $\beta$ is the angle shown in Figure 2.2. Neglecting the second order term, this equation becomes

$$\lambda_3 = \frac{1 - \beta}{2\pi} \times \frac{4\pi d_2 \Delta d_2}{s},$$

(2.5) when $\Delta d_2 \to 0$.

From Equations (2.2), (2.3) and (2.5), the probability that Scenario 11 occurs given $x_B$, $y_B$, $d_1$ and $d_2$ is

$$p'_{11} = \lambda_1 \times \lambda_2 \times \lambda_3$$

$$= \frac{4\pi d_1 \Delta d_1}{s} \times \frac{\Delta x_B \Delta y_B}{s} \times \frac{(1 - \beta)}{2\pi} \times \frac{4\pi d_2 \Delta d_2}{s}$$

(2.6)

$$= \frac{8\pi^2}{s^3} d_1 d_2 \Delta x_B \Delta y_B (1 - \beta).$$

Finally the probability of Scenario 11 occurring is

$$p_{11} = \int_0^r \int_0^r \int_{d_1 - \frac{r}{2}}^{r + d_1} \int_{f_1(d_1, x_B)}^{f_2(d_1, x_B)} 2 \times p'_{11} \, d(y_B) \, d(x_B) \, g(d_2) \, d(d_2) \, g(d_1) \, d(d_1),$$

(2.7)

where $g(.)$ is a given probability density function of the distance between a transmitter and a receiver. The $x$ coordinate of any point within $T(\bar{A} \cap B)$ lies in the interval $[\frac{d_1}{2}, r + d_1]$, which explains the bounds of the third integral.

To solve Equation (2.7), we need to compute $\beta$, $f_1(d_1, x_B)$ and $f_2(d_1, x_B)$. As shown in Figures 2.3 and 2.4, computations for $\beta$, $f_1(d_1, x_B)$ and $f_2(d_1, x_B)$ are different when station $B$ falls into different regions of $T(\bar{A} \cap a)$. Therefore, we need to compute $p_{11}$ in different regions and sum the results.

To compute $\beta$, we consider Figure 2.3 which indicates that if $d_2$ is fixed, $T(\bar{A} \cap a)$ can be divided by the dashed circle into two areas, denoted by $R1$ and $R2$ respectively. $\beta$ is computed differently in $R1$ than in $R2$ because the circle centered at $B$ intersects with other circles differently in different regions. However, the computation for $\beta$ is trivial in either $R1$ or $R2$ although its expression is tedious, so we omit it here.

To determine $f_1(d_1, x_B)$ and $f_2(d_1, x_B)$, we further divide region $R1$ and $R2$ in Figure 2.3 into smaller regions. Let $\Gamma$ denote such a value that when $d_2 = \Gamma$, $x_Q = d_1$, where $x_Q$ is the $x$ axis of the intersection $Q$ shown in Figure 2.4. When $d_2 \leq \Gamma$, $T(\bar{A} \cap a)$ is divided into
Figure 2.3: Different locations of station B lead to different computations for $\beta$.

five areas labeled by I, II, III, IV and V as shown in Figure 2.4(a). When $d_2 > \Gamma, T(\bar{A} \cap a)$ is divided into five different areas shown in Figure 2.4(b). $f_1(d_1, x_B)$ and $f_2(d_1, x_B)$ can be determined within each of the five regions and Equation 2.7 can be solved numerically. Similar to $\beta$, $f_1(d_1, x_B)$ and $f_2(d_1, x_B)$ are easy but tedious to obtain, thus we omit their expressions here.

A similar analysis can be conducted for computing the probability that each of the other 10 scenarios in Figure 2.1 occurs, denoted by $p_i$, where $i \in \{2, 3, 4 \ldots 12\}$.

2.1.3 Comparing Scenario Likelihood

We now compare the likelihood of the three classes occurring. In the context of a multi-hop wireless network, it is especially interesting to evaluate the probability of each of the three classes occurring between direct neighbors as a function of hop distance. We assume that the distance between transmitters and receivers is constant, and we compute $p_i$ as a function of $d$, where $d = d_1 = d_2$.

We can obtain results independent of flow density by computing the conditional probability that a particular Scenario $i$ occurs, given that some connection exists between the
two flows, i.e., they are not isolated. This is then the probability of Scenario $i$ occurring conditioned on the event that any of the scenarios except Scenario 1 occurs. Thus it is given by $p_i/p$, where $p = \sum_{i=2}^{12} p_i$. Results validated by Monte Carlo simulations are shown in Figure 2.5 as a function of the normalized hop distance, which is the actual distance between a sender and a receiver divided by the radio range. Simulations are run dropping two pairs of nodes at a given distance from each other uniformly at random in a square area with wrap-around (to avoid border effects), and seeing which case they map into (excluding Scenario 1). By so doing, results are insensitive of area size and node density.

We observe that Scenario 11 (belonging to the AIS class and one with problematic performance) emerges as the dominating scenario when the distance between sender and receiver increases.

Denote the respective probabilities of the Senders Connected (SC), Asymmetric Incomplete State (AIS), and Symmetric Incomplete State (SIS) classes occurring as $\epsilon_{SC}$, $\epsilon_{AIS}$ and $\epsilon_{SIS}$. We have $\epsilon_{SC} = \{p_2 + p_3 + p_4 + p_5 + p_6 + p_7\}/p$, $\epsilon_{SIS} = \{p_8 + p_9 + p_{10}\}/p$, and $\epsilon_{AIS} = \{p_{11} + p_{12}\}/p$. The three probabilities above are reported in Figure 2.6 as a function
Figure 2.5: Probability of each of the eleven scenarios occurring as a function of normalized hop distance.

of the normalized hop distance.

Results indicate that the AIS and SIS classes account for a significant fraction of all possible scenarios. Increasing the hop distance, the likelihood of class AIS approaches that of the SC class despite accounting for two vs. six scenarios. The ratio between AIS and SIS probabilities is about 2.

In multi-hop wireless networks, the distribution of hop distances depends on the routing protocol deployed. From Figure 2.6, it is clear that routing protocols can have a significant impact on the probability of each class occurring. To evaluate this impact, we conducted simulation experiments to measure the hop distance distribution resulting from the operation of current routing protocols in large-scale networks with mobility. In our simulations, we consider 300 stations randomly deployed in a 2000 m × 2000 m area. A random waypoint model [55] is used to simulate mobility, and connections are randomly established among the nodes. The routing algorithm considered is Distance Sequence Distance Vector (DSDV) (we tested other routing algorithms and obtain similar results). To compute the normalized hop distance distribution, we calculate the hop distance of every link traversed
by each packet and divide it by the maximum transmission range. After averaging the results of several simulation runs, we obtained the distribution of normalized hop distances reported in Figure 2.7. Compared to the results in [39], in which the impact of routing protocol is not considered, our simulation experiments suggest that in a multi-hop wireless network, about 40 percent of normalized distances are within the [0.9, 1] interval. This can be explained by the fact that routing protocols select minimum-hop paths to reach the destination. Combined with the results in Figure 2.6, this means that current routing protocols make the scenarios of classes AIS and SIS more likely to occur in a random network, by favoring larger hop distances when choosing next hops. The occurrence probabilities of classes SC, AIS and SIS under routing can be computed from Figures 2.6 and 2.7, assuming random distribution of two-hop flows.

In the previous analysis, we assumed identical transmission and sensing range for all stations. In a real network, however, sensing range and transmission range of a station are usually different. In order to examine the impact of this factor, we vary the sensing range of the nodes, while keeping the transmission range fixed. We assume the distance between
Figure 2.7: Hop distance distribution in a multi-hop wireless network.

senders and receivers is equal to the transmission range. Results are shown in Figure 2.8, as a function of the ratio between sensing range and transmission range. Although the occurrence probabilities of the AIS and SIS classes decrease when the ratio between sensing range and transmission range increases, the probability of class AIS occurring is significant over the whole range of values encountered in practice. The ratio between AIS and SIS is always around 2.

2.2 Impact of MAC Protocols

In this section we qualitatively assess by simulation experiments the performance of IEEE 802.11 in the two-flow subgraphs identified by the spatial analysis of Section 2.1. Our objective is to identify the critical performance issues that can arise when a typical CSMA-based MAC protocol is employed to arbitrate channel access between the two flows.

The main channel access mechanisms implemented in the IEEE 802.11 standard [15] are the “basic access” two-way handshake without RTS/CTS and the four-way handshake with RTS/CTS. We study how both access methods perform in each of the twelve possible
Figure 2.8: Probabilities of the two groups, in case of different transmission and sensing range.

scenarios comprising two flows.

We consider a data rate of 11 Mbps, and a fixed packet size of 500 bytes. Both flows are continuously backlogged with UDP traffic. We measure the average throughput of each flow during consecutive periods of 0.4 seconds. Each simulation experiment lasts 20 seconds, thus we collect 50 throughput samples for each flow. In Figure 2.9, we plot the normalized throughput of flow $B$ versus the normalized throughput of flow $A$ in each measurement interval. The "$\times$" marks correspond to the two-way handshake access method; the dot marks correspond to the four-way handshake. Dashed lines represent ideal fairness, since they consist of the points where the throughput of the flows are equal. Note that to precisely determine the proportional fair rate in an IEEE 802.11 network is beyond the scope of this thesis. To estimate the ideal proportional fair rates in the two-flow scenarios considered in this chapter, we make a simplifying assumption that the aggregate of the two flow’s received throughput remains constant when the throughput of each individual flow changes. Under this assumption, the proportional fair rate is achieved when the two flows receive equal throughput.
Figure 2.9: Short-term normalized throughput for the twelve scenarios. The ‘×’ marks correspond to two-way handshake simulations; the dot marks correspond to four-way handshake simulations.

Omitting Scenario 1 in which the flows are isolated due to spatial reuse, we make the following observations on the other 11 scenarios.

- For the Senders Connected (SC) class consisting of Scenarios 2-7, most of the throughput points reside close to the dashed line, indicating that neither short-term nor long-term fairness problems exist.

- For the Asymmetric Incomplete State (AIS) class consisting of Scenarios 11 and 12, the variance of the throughput points is small. However, these points largely deviate from the fairness line, indicating severe unfairness at all time scales and flow starvation.

- For the Symmetric Incomplete State (SIS) class consisting of Scenarios 8, 9 and 10,
the throughput points are symmetrically scattered around the dashed line, indicating short-term unfairness, but in the long term, the throughputs of the two flows are the same.

Since the SC class does not encounter fairness problems and because its performance can be analyzed with existing techniques [8], we will not further consider it. We focus instead on the problematic AIS and SIS classes. In Sections 2.3 and 2.4 we will develop a detailed analysis of these two classes. Here we provide a qualitative explanation of the behavior observed in the simple experiments of Figure 2.9.

**Origins of AIS Long-Term Unfairness.** Figure 2.10 shows example topologies of Scenarios 11 and 12 of the AIS class. The core property of the AIS class is the asymmetric view of the channel state possessed by the two flows. When transmitters are not in range of each other, channel state information is necessarily incomplete because transmitters cannot sense when the other flow is transmitting. This lack of information affects the two AIS flows in very different ways because of the asymmetry of the topology. In particular, flow \( A \) lacks the necessary information to compete fairly with flow \( B \), while flow \( B \) does not suffer from the incomplete channel state information. This disparity is due to the fact that sender \( A \) does not sense any packets belonging to flow \( B \), and consequently, completely ignores the activity of the other flow. On the other hand, sender \( B \) can hear the control packets sent by node \( a \) (CTS and/or ACK), and hence can detect the activity of the other flow. While sender \( B \) knows exactly when to start contending for the channel, sender \( A \) has to discover an available time-slot randomly, without any coordination with sender \( B \). This fact results in many transmission attempts of sender \( A \) without any response back from receiver \( a \), most often because \( A \) attempts to transmit in the middle of a transmission of flow \( B \), when receiver \( a \) cannot receive correctly the packets sent by \( A \), or is not able to reply. Consequently, sender \( A \) is forced to timeout and to repeatedly double its contention window. As a result, the probability of flow \( A \) capturing the channel is significantly smaller than that of flow \( B \).

**Origins of SIS Short-Term Unfairness.** Figure 2.11(a) reports an example topology
Figure 2.10: Example topologies of the AIS class.

of Scenario 8 of the SIS class. The classic "hidden terminal" problem [6, 33], depicted in Figure 2.11(b), is a special case of Scenario 8 when the two flows have the same receiver. The core property of the SIS class is the symmetric view of the channel state possessed by the two flows while the channel state information is incomplete. This property results in short-term unfairness but long-term fairness. The origin of the short-term unfairness lies on the binary exponential backoff mechanism coupled with the large packet loss probability that characterizes all scenarios of the SIS class. The large packet loss probability is due to incomplete channel state information. In particular, a sender does not sense the other sender's activity, thus it can start transmitting the first packet of the two-way or four-way handshake while the other sender is also attempting to transmit. Indeed, a sender does not stop decrementing the backoff counter as soon as the other sender starts transmitting. This fact significantly increases the collision probability of the flows. After experiencing a collision, a source doubles its contention window, thus reducing the chances of attempting a new transmission in the next available slot. On the contrary, a source resets its contention window to the minimum value after a successful transmission, increasing the likelihood of a new transmission attempt in the near future. Thus, in all scenarios of the SIS group, the system endures significant durations in which one flow dominates channel access with many repeated transmissions, while the other flow is forced to repeatedly double its contention window significantly reducing the chance to seize the channel. However, this
problem affects the two flows equally, because the geometric relationships in the scenarios of the SIS group are symmetric. The two flows alternate capturing the channel and dominating over the other flow. Therefore, flow pairs belonging to the SIS group do not suffer from long-term unfairness (essentially because of symmetry). Instead, simulations show significant short-term unfairness, as illustrated in Figure 2.9, which is clearly undesirable as it can adversely affect delay-sensitive applications, such as voice.

(a) Scenario 8.  
(b) Hidden terminal problem.

Figure 2.11: Example topologies of the SIS class. (b) is a special case of (a), when two Mobile Units (MU1 and MU2) are sending packets simultaneously to the same Access Point (AP) in infrastructure mode, as encountered in the classic "hidden terminal" problem.

**Impact of Mobility.** In a network where stations move randomly, a flow is expected to belong to different local subgraphs constantly changing over time. Will this mobility alleviate the unfairness problem observed in Figure 2.9? We conducted a simulation experiment where 40 stations move according to the random waypoint model in a 1000x1000 region with a speed uniformly distributed in a [7, 15] m/s interval. 20 connections are established between randomly chosen pair of stations. Figure 2.12 shows that although mobility averages out unfairness over time-scales of 120 seconds, severe unbalance is observed during time windows of 10 seconds, in which two dominating flows starve all of the other flows. This means that severe fairness problems still exists over time scales associated with the speed of the nodes.
Figure 2.12: Flow throughput comparison between a 10 second snapshot and a 120 second snapshot.

2.3 Asymmetric Incomplete State

In this section we develop an analytical model to study the behavior of flow pairs belonging to the Asymmetric Incomplete State (AIS) group, comprising Scenarios 11 and 12 of Figure 2.1. Examples of topologies that satisfy the geometric properties of these two scenarios are illustrated in Figure 2.10(a) and 2.10(b). In both cases, flow $B$ achieves a significantly higher throughput as compared to flow $A$ for CSMA both with and without RTS/CTS (see Figure 2.9). The only difference between the two cases is that in scenario 12 the receivers are in radio range of each other, whereas this is not true in scenario 11. Below, we show how this topology difference affects performance.

Our objective is to analytically compute the throughput of the two flows to both characterize the root cause of the starvation of flow $A$ and to evaluate the impact of key system parameters on the extent of the starvation.

The remainder of the section is organized as follows. In Section 2.3.1 we introduce a general model of the behavior of a backlogged source employing the standard 802.11 DCF.
This model will be applied in Section 2.3.2 to the particular scenarios of the AIS group. In Section 2.3.3 the analysis is extended to the case of non-backlogged sources in order to assess the impact of the starvation problem in more general network scenarios. Finally, numerical results and model validation are presented in Section 2.3.4 and 2.3.5.

2.3.1 General Decoupling Model of an 802.11 Station

Our modeling framework for the AIS group contrasts with existing techniques (e.g., [8]) in that we account for the fact that, in a general topology, the channel state as perceived by a station can be different from node to node. In [8], all stations are assumed to be in range of each other so that they share a common view of the channel. In contrast, we build a model representing the channel state as seen by each individual source, instead of the channel state shared by all nodes. Yet, in the scenarios of the AIS group, the behavior of each station can still be decoupled from that of the other stations, as done in [8]. This property significantly simplifies the analysis of the interaction among the two flows, as we show below.

Channel State as Seen by a Single Source. The behavior of an arbitrary station employing a CSMA protocol such as the DCF function of 802.11 can be abstractly represented by a temporal diagram such as the one illustrated in Figure 2.13. We identify 4 different states: (i) idle channel; (ii) channel occupied by successful transmission of the station; (iii) channel occupied by a collision of the station; (iv) busy channel due to activity of other nodes, detected by means of either physical or virtual carrier sensing (i.e., NAV).

![Figure 2.13: A station's channel view and embedded discrete time renewal process.](image)

The durations of the time intervals during which the channel remains in the four states above are denoted by $\sigma, T_s, T_c$, and $T_b$, respectively. In Figure 2.13, the time instants of a possible state change are pointed to by arrows placed below the temporal axis. While $\sigma$ is
a constant equal to one 802.11 time slot, the duration of the other intervals can be variable (with general distribution) depending on the access mechanism (basic access or RTS/CTS), the frame size, and the sending rate of the transmitting node(s).

**Analysis of the Behavior of a Single Source.** In order to analyze the behavior of a station, we make the fundamental assumption that the channel evolution can be described by a renewal process: at each switching time the next state does not depend on the current state, and the four states occur with fixed probabilities \( \Pi_s, \Pi_c, \Pi_\sigma \) and \( \Pi_b \). The resulting process is thus semi-Markov. Notice in Figure 2.13 that the durations of \( T_s, T_c \) and \( T_b \) comprise an idle slot at the end of the interval, which occurs deterministically, so that it is not considered as an individual event of the overall stochastic process (hence these special slots are marked with a dotted arrow in Figure 2.13).

At the end of an idle slot, the station decrements its backoff counter, and starts transmitting in the next interval if the counter reaches zero. Let \( \tau \) be the probability that the station sends out a packet after an idle slot, under the assumption that it is always backlogged (we will remove this assumption later in the section). Let \( p \) be the probability that a transmission of the station is not successful. The probability \( p \) is usually referred to as the *conditional* packet loss probability [8]. We also introduce \( b \), as the probability that the channel becomes busy after an idle slot due to activity of other nodes (assuming that the station does not start transmitting). Using these probabilities, we can specify the occurrence probability of each of the four channel states at the switching instants as follows,

\[
\begin{align*}
\Pi_s &= \tau (1 - p), \\
\Pi_c &= \tau p, \\
\Pi_\sigma &= (1 - \tau) (1 - b), \\
\Pi_b &= (1 - \tau) b.
\end{align*}
\]  (2.8)

**Computation of the throughput.** Using renewal theory, the throughput of a station (expressed in packet/s), is given by

\[
T_P = \frac{\Pi_s}{\Delta},
\]  (2.9)
where $\Delta$ is the average duration of a channel state (in seconds). The final expression for the throughput of a station is then given by

$$T_p = \frac{\tau(1 - p)}{\tau(1 - p)T_s + \tau p T_c + (1 - \tau)(1 - b)\sigma + (1 - \tau) bT_b}. \quad (2.10)$$

Now, the probability $\tau$ is a deterministic function of $p$, which depends only on backoff parameters such as the window size, the number of backoff stages, etc. For 802.11, the expression of $\tau$ as a function of $p$ has been first computed in [8]. More recently, it has been shown that one can easily write similar expressions of $\tau$ as a function of $p$ for a large class of backoff mechanisms, employing arbitrary window distributions and backoff multipliers [37].

The complete expression of $\tau$ for 802.11, which takes into account the maximum retransmission limit jointly with the maximum window size, is given by

$$\tau = \frac{2q(1 - p^{m+1})}{q(1 - p^{m+1}) + W_0 [1 - p - p(2p)^m (1 + p^{m-m'}q)]}, \quad (2.11)$$

where $q = 1 - 2p$, $W_0$ is the minimum window size, $m$ is the maximum retry limit, and $m'$ is the backoff stage at which the window size reaches its maximum value, $m' \leq m$.

The average durations $\bar{T}_s$ and $\bar{T}_c$ of a successful transmission or of a collision in which the station is involved are also known a priori (see [8]). It turns out that the only unknown variables are the occurrence probability $b$ of a busy period, its average duration $\bar{T}_b$, and $p$, the conditional packet loss probability. These quantities are specific to each station, and their values derive from the interaction of the station with the rest of the network.

In the next section, we apply the above modeling technique to the study of the AIS scenarios. However, we remark that this methodology has general applicability to modeling long-term throughput of flows in arbitrary networks with any number of nodes. In complex topologies comprising more than two flows, evaluation of the variables $b$ and $\bar{T}_b$ for each node turns out to be the most difficult task, whereas their computation is quite simple in scenarios comprising only two flows. While our study in this paper is restricted to two-flow scenarios, computation of $b$ and $\bar{T}_b$ in more general network scenarios can be found in [19]. The analysis of all possible combinations of flow pairs provides instead the basis
to evaluate the packet loss probability $p$ (the other fundamental variable that we need to compute the throughput) of a node in an arbitrary topology. That is, two-flow scenarios are the building blocks that can be used to evaluate the collision probability of a transmitter in any network topology: a transmission on a link is successful if it does not collide with any other transmission on neighboring links. Therefore, the careful analysis of all flow pairs presented in this paper is the necessary first step toward the throughput analysis in arbitrary networks.

2.3.2 Analysis of AIS Flows

Now we apply the general model introduced in Section 2.3.1 to independently study the behavior of the two transmitting nodes $A$ and $B$ in the scenarios of the AIS group. We add an index $A$ or $B$ within brackets to all quantities defined in Section 2.3.1 to distinguish between the values of the two stations. For example, $\tau(A)$ is the transmission probability of node $A$. For simplicity, we assume that the payload size of all data frames is constant. The analysis can be extended to the case of variable payload sizes.

We start by considering the behavior of flow $A$. As described in Section 2.2, sender $A$ does not detect any activity on the channel produced by flow $B$, neither by means of physical nor virtual carrier sensing. As a consequence, $b(A) = 0$. It turns out that the only parameter that we need to compute for flow $A$ is the conditional collision probability $p(A)$, or its complement $s(A)$, the conditional success probability.

The transmission attempts of sender $A$ are not coordinated with those of sender $B$, and occur at random points in time according to the backoff process of $A$. Our approximation is to assume that each transmission attempt of $A$ is an independent random look at the activity of flow $B$. Hence we need to characterize the activity of flow $B$. A fundamental property of flow $B$ is that all transmission attempts of sender $B$ are successful, i.e., $p(B) = 0$. Indeed, transmissions of $B$ could only collide with the control packets sent by $a$, but the probability that this happens is negligible because of the lack of synchronization: almost always $B$ or
a avoid collisions by sensing the channel busy and refraining from transmitting.†

Under saturated traffic, the activity of flow $B$ is a sequence of successful transmissions, separated by a random number of backoff slots uniformly distributed in the minimum window size $W_0$ (the window size of sender $B$ is never increased, since $p(B) = 0$). Occasionally, sender $B$ receives a $CTS$ or $ACK$ packet from node $a$, freezing its backoff counter for the remaining part of the successful packet exchange of flow $A$.

The only chance flow $A$ has to successfully transmit is when the initial packet of the two-way or four-way handshake (a DATA frame or an RTS frame) happens to arrive during those short gaps in which sender $B$ is in the backoff phase. More precisely, we have to examine the channel occupation state as perceived by the receiving node $a$ while node $A$ is trying to initiate a new data transfer, which is illustrated in Figure 2.14 for both scenarios of the AIS group. Notice that we remove the amount of time in which receiver $a$ is actively transmitting or receiving from sender $A$, because we are considering the channel around $a$ conditioned on the fact that $A$ starts transmitting a new packet.

![Figure 2.14: Channel occupation state as perceived by node $a$ while sender $A$ attempts to initiate a new data transfer. The top diagram (a) refers to Scenario 12, the bottom (b) to Scenario 11.](image)

The temporal evolution of the channel as perceived by node $a$ can be divided into cycles

---

†A collision would occur only if the time instants at which nodes $a$ and $B$ start placing a packet on the channel are separated in time by less that the propagation delay between the two nodes, which is a rare event.
of variable duration $C$. Each cycle comprises a successful data transfer of flow $B$, of duration $T_s$, and a variable number $i$ of slots $\sigma$ corresponding to the backoff phase of sender $B$. The "gap" $G$ during which a packet originated by node $A$ can be received by node $a$ is also reported in Figure 2.14. We observe that this gap comprises also the DIFS space at the end of $T_s$. In Scenario 11, node $a$ does not receive the ACK of node $b$, thus the gap is enlarged by the duration of an ACK and a SIFS. A key observation is that for flow $A$ to be successful, not only node $A$ must start transmitting during the gap, but the entire packet that node $A$ places on the channel must fit into the same gap. This explains why we do not consider as potential gaps the short inter-frame spaces in between the packets of flow $B$, nor even, in Scenario 11, the space corresponding to the (unheard) CTS of node $b$, because $\text{RTS} > \text{CTS} + 2 \text{SIFS}$.

If the RTS/CTS mechanism is not used, the situation is very similar to that represented in Figure 2.14, with the only difference that there are no RTS and CTS packets. Also, notice that in this case an entire DATA packet must fit into the gap in order to be successfully received by node $b$.

In order to compute the conditional packet loss probability $p(A)$, we make the simplifying assumption that the initial packet sent by node $A$ (RTS or DATA packet) arrives at an arbitrary point in time during a cycle $C$. Also, we assume that all transmission attempts of node $A$ randomly and independently sample a point within a generic cycle $C$. Although this is only an approximation, the analytical predictions produced by the resulting model are quite accurate (see Section 2.3.4).

We observe that since the duration of a cycle is variable (the number $i$ of slots is randomly chosen by the transmitting node $B$), we must consider the fact from renewal theory that the probability of arriving within a cycle of duration $C$ is proportional to $C$.

The final expression of the packet loss probability of flow $A$ is as follows

$$p(A) = 1 - \frac{2}{W_0[2T_s + (W_0 - 1)\sigma]} \sum_{i=0}^{W_0-1} \max(0, D + i\sigma),$$

(2.12)

where $D$ is a parameter that depends on the access mechanism used (basic access or
Table 2.1: The value of D to be used in (2.12).

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>RTS/CTS - scenario 11</td>
<td>ACK + DIFS - RTS - SIFS</td>
</tr>
<tr>
<td>RTS/CTS - scenario 12</td>
<td>DIFS - RTS</td>
</tr>
<tr>
<td>Basic Access - scenario 11</td>
<td>ACK + DIFS - DATA - SIFS</td>
</tr>
<tr>
<td>Basic Access - scenario 12</td>
<td>DIFS - DATA</td>
</tr>
</tbody>
</table>

RTS/CTS) and on the considered scenario. Table 2.1 provides the value of D for all combinations of cases. Notice that D can take negative values, which explains the max operator in Equation (2.12). If D is positive, Equation (2.12) simplifies to

\[
p(A) = \frac{2(T_s - D)}{2T_s + (W_0 - 1)\sigma}, \quad D > 0. \tag{2.13}
\]

We observe that \(p(A)\) can be directly computed as a function of all known system parameters. Thus, we can already compute the throughput of flow A by first obtaining \(\tau(A)\) from Equation (5.6), and then substituting both \(\tau(A)\) and \(p(A)\) into Equation (5.5) (recall that \(b(A) = 0\)).

We now turn to the analysis of flow B. We have already seen that \(p(B) = 0\), thus we can obtain from Equation (5.6) \(\tau(B) = 2/(W_0 + 1)\). The only unknown variable of flow B is \(b(B)\), the probability that node B, after an idle slot during the backoff phase, starts receiving a control packet from \(a\) (CTS or ACK), after which B sets the NAV and suspends its activity, allowing the packet exchange of flow A to complete successfully. The duration \(T_b\) of this suspension is equal to \(T_s\) minus the duration of the first packet (RTS or DATA) sent by A, which is not heard by B.

Since we have already independently computed the throughput of flow A, we know the rate at which sender B has to suspend its activity during the backoff phase. Indeed, the following equation, similar to Equation (5.5), has to be satisfied

\[
T_P(A) = \frac{[1 - \tau(B)]x}{\tau(B)T_s + [1 - \tau(B)](1 - x)\sigma + [1 - \tau(B)]xT_b}, \tag{2.14}
\]
from which one can obtain the unknown variable $x = b(B)$, to be used into the expression of the throughput of flow $B$.

We remark that as a result of our analysis, the throughputs of both flows are available in closed form expressions. This is made possible by the hypothesis that both flows are backlogged. In the next section, the analysis is extended to the case of non-backlogged sources.

2.3.3 Non-Continuously-Backlogged Flows

The starvation problem observed in the scenarios of the AIS group is particularly severe when flow $B$ is continuously backlogged and transmits at the maximum achievable rate, occupying the largest possible fraction of channel time with its own transmissions, and leaving few gaps to be discovered by flow $A$. Therefore it is important to model these scenarios under more general assumptions, i.e., when flow $B$ does not utilize all of the available bandwidth. This can happen because flow $B$ represents a variable rate flow that empties its transmission queue, or if the sender or receiver of flow $B$ are deferring elsewhere, e.g., if flow $B$ senses the activity of other flows in the network by means of either physical or virtual carrier sensing.

In our analysis, we assume that the maximum throughput achievable by a station is known. For example, if the transmission queue of a source is fed by an arrival process of data packets from the upper protocol layers at rate $\lambda$, the achieved rate clearly cannot exceed this value. If the queue is backlogged but the station senses the activity of other flows in its neighborhood, the achievable throughput will be limited by the resulting share of the channel capacity. In this paper we limit ourselves to the analysis of two-flow scenarios, therefore we do not deal with the problem of solving the interaction of many flows in arbitrary topologies. Regardless, we can model the behavior of two-flow scenarios embedded in a large topology by considering each scenario in isolation from the network, and assuming that the transmission queues of the senders are fed by a given arrival rate of packets $\lambda$, that can either represent the actual data rate offered by the upper protocol layers, or the
maximum rate resulting from the interaction with the rest of the network.

Moreover, we assume that the actual throughput $T$ achieved by a station as a function of $\lambda$ is equal to the input rate $\lambda$ up to a saturation value $T_{satur}$, after which it remains constant and equal to $T_{satur}$. Our analysis in the previous sections has actually computed the saturation throughput values $T_{satur}(A)$ and $T_{satur}(B)$ when both flows are backlogged.

Now we extend the analysis to the case in which senders are fed by arbitrary input rates $\lambda(A)$ and $\lambda(B)$. We add a new probability $e$ to the description of the behavior of a single station introduced in Section 2.3.1, which is the conditional probability that the transmission queue of the station is empty, given that the station can potentially start a new transmission (i.e., when its backoff counter is zero and the channel has been sensed idle for a time slot). The occurrence probability of each of the four channel states at the switching instants are modified as follows,

$$
\begin{align*}
\Pi_s &= \tau(1 - p)(1 - e), \\
\Pi_c &= \tau p(1 - e), \\
\Pi_o &= [(1 - \tau) + \tau e](1 - b), \\
\Pi_b &= [(1 - \tau) + \tau e] b, 
\end{align*}
$$

and the throughput expression in Equation (2.9) must be changed accordingly. As a result, the only unknown variable that we need to compute is the probability $e$. This can be easily done by considering that $T = \min(\lambda, T_{satur})$ as described above, and by assuming that all other variables are known: if $T|_{e=0} < \lambda$, then $e = 0$ (the source is saturated); otherwise $e$ is equal to the value $e^*$ such that $T|_{e^*} = \lambda$, which is easily obtained by inverting the throughput formula.

The solution of the scenarios of the AIS group under arbitrary input rates requires an iterative approach: each flow is studied assuming that the throughput of the other flow is given. The independent analysis of each flow allows to compute a new estimate of its throughput, to be used in the analysis of the other flow in the next step of the iteration. After a few iterations we obtain the fixed-point solution.
For flow $A$, we only need to compute the collision probability $p(A)$, or its complement $s(A)$. To do so, we model the activity of flow $B$ as perceived by $a$ while sender $A$ attempts to transmit as an alternating on-off process. The on period has a fixed duration $T_{ON}$, equal to the portion of the cycle in Figure 2.14 not occupied by the gap $G$, which depends only on the access method and the specific scenario. The off period is the gap available for flow $A$, that now can contain also periods of time in which the queue of sender $B$ is empty. The average duration $\bar{T}_{OFF}$ can be computed from the following expression,

$$\frac{1}{T_{ON} + \bar{T}_{OFF}} = \frac{T(B)}{1 - T(A)\bar{T}_b}, \quad (2.16)$$

which states that the rate at which on periods occur (the inverse of the average duration of a cycle) must be equal to the throughput of flow $B$, normalized by the fraction of time in which the channel is not occupied by successful transmissions of flow $A$. We assume that the duration of the off period is exponentially distributed, and that the transmission attempts of sender $A$ arrive at random point in time. We obtain the following formula for the successful probability of flow $A$,

$$s(A) = \frac{\bar{T}_{OFF}}{T_{ON} + \bar{T}_{OFF}} e^{-\frac{d}{\bar{T}_{OFF}}}, \quad (2.17)$$

where $d$ is the duration of the first packet sent by $A$ (RTS or DATA). Equation (2.17) states that a transmission attempt of flow $A$ is successful if the first packet arrives during the off period of flow $B$ and if it is fully received by $a$ before the beginning of the next on period. A new estimate of the throughput of flow $A$ can now be derived using the collision probability $p(A) = 1 - s(A)$.

For flow $B$, the only unknown is the probability $b(B)$, which is updated using the same reasoning applied at the end of Section 2.3.2.

### 2.3.4 Model Validation Through Simulations

Here we validate our analysis of the AIS group and compare the analytical predictions of the throughput of the two flows with simulation results obtained with ns. We consider
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>SIFS</td>
<td>10 µs</td>
</tr>
<tr>
<td>DIFS</td>
<td>50 µs</td>
</tr>
<tr>
<td>EIFS</td>
<td>364 µs</td>
</tr>
<tr>
<td>σ</td>
<td>20 µs</td>
</tr>
<tr>
<td>BasicRate</td>
<td>2 Mbps</td>
</tr>
<tr>
<td>DataRate</td>
<td>11 Mbps</td>
</tr>
<tr>
<td>PLCP length</td>
<td>192 bits @ 1 Mbps</td>
</tr>
<tr>
<td>MAC header (RTS,CTS,ACK,DATA)</td>
<td>(20,14,14,28) bytes @ BasicRate</td>
</tr>
<tr>
<td>(CW&lt;sub&gt;min&lt;/sub&gt;,CW&lt;sub&gt;max&lt;/sub&gt;)</td>
<td>(31,1023)</td>
</tr>
<tr>
<td>Short Retry Limit</td>
<td>7</td>
</tr>
<tr>
<td>Long Retry Limit</td>
<td>4</td>
</tr>
</tbody>
</table>

Table 2.2: Parameters setting for the MAC and physical layers.

stations operating according to the 802.11b standard, with a data rate of 11 Mb/s. The common parameters at the MAC and physical layers for all of the experiments of this paper are reported in Table 5.4.

We first consider the case of continuously backlogged sources. In Figure 2.15 we compare the throughput achieved by flow A (in pkt/s) in the two scenarios of the AIS group. We vary the data payload size from 100 bytes to 1500 bytes, and we use the RTS/CTS access mechanism to transfer all data frames, irrespective of their size. Figure 2.16 reports the corresponding results for flow B.

![Figure 2.15: Throughput of Flow A vs. data payload size (with RTS/CTS).](image)
Figure 2.16: Throughput of Flow B vs. data payload size (with RTS/CTS).

We observe that despite a number of approximations, the model’s predictions provide an excellent match with simulation results for all payload sizes and in both scenarios. As expected, the throughput of Flow B is significantly larger than the throughput of flow A (notice the different scales on the y axis). As predicted by the model, starvation is partially alleviated in scenario 11, when nodes a and b are not in range of each other. As already explained, this is due to the fact that in this case, the gap in which node A can successfully send to a is enlarged by the duration of an ACK (see Figure 2.14).

In Figure 2.17 we compare the throughput achieved by flow A in scenario 11, considering the basic access method and assuming that the maximum retry limit is equal to either the Short Retry Limit or the Long Retry Limit as specified in the 802.11 standard, irrespective of the data payload size (notice that the Short Retry Limit corresponds in the model to $m = 6$, while the Long Retry Limit corresponds in the model to $m = 3$).

We observe that flow A achieves very different throughput depending on the packet payload size used. Flow A is completely starved (i.e. achieves zero throughput) when the size of a DATA packet exceeds the maximum possible gap left free by flow B. For very small payload sizes, the throughput of flow A can actually be higher than the throughput obtained employing the RTS/CTS mechanism due to the reduced MAC overhead.

We also observe that flow A achieves significantly higher throughput when the Long
Retransmission Limit \((m = 3)\) is used, with respect to the case in which the Short Retransmission Limit \((m = 6)\) is used. This is due to the fact that when \(m = 3\), sender A spends less time in backoff, because after reaching the maximum retransmission limit, the window is reset to the minimum value \(W_0\). This increases the aggressiveness of sender A, which is more likely to find an available gap in the activity of flow B.

Finally, we consider flows that are not continuously backlogged. The most interesting case to analyze is when we limit the rate of flow B so as to leave more time available to flow A. In Figures 2.18 and 2.19 we plot the throughput achieved by the flows as a function of the input rate \(\lambda(B)\), while keeping flow A always backlogged. Figure 2.18 refers to scenario 12 with the RTS/CTS mechanism, whereas 2.18 refers to scenario 11 with basic access. The data payload size is constant and equal to 1000 bytes, and \(m = 6\). We also report on the plots the sum of the throughput of the two flows.

We observe that the asymmetry between the flows results in nearly strict priority of flow B over flow A. Indeed, flow B achieves a throughput exactly equal to the input rate up to a sharp saturation point, whereas flow A, even if backlogged, only gets a fraction of the remaining channel capacity. Notice that there is a significant loss in aggregate throughput when the two flows obtain similar throughput. This is due to the time wasted by flow A during backoff, even when flow B does not have packets to transmit, leaving the channel...
idle for a large fraction of time. By limiting the rate of flow $B$, it is possible to give the flows the same rate, at the expense of a loss in the aggregate throughput. This illustrates the trade-off between fairness and capacity (aggregate throughput) in networks as it is realized with an unfair access mechanism.

### 2.3.5 Model Validation Through Experiments

Here we validate our analysis of the AIS group and compare the analytical predictions with real experiment results measured in the Technology For All network (TFA). The Technology For All (TFA) network is an operational mesh network that provides Internet access in a densely populated urban neighborhood in Houston. The network has two tiers including a backhaul tier which wirelessly forwards data and an access tier which provides access between end-users and the mesh infrastructure. Fig. 2.20 depicts the spatial distribution of the TFA network backhaul tier and the connectivity map. In the figure, nodes are depicted as connected if a direct transmission can occur between the two backhaul nodes. All links are omni-directional with the exception of a directional link (shown in black) which serves as an additional point of capacity for the network. Each mesh node runs software derived from open-source LocustWorld mesh networking software. Each mesh node has
Figure 2.19: Throughput of Flow A vs. arrival rate of flow B (basic access, Scenario 11).

A single SMC 2532-b 802.11b wireless adapter with 200 mW transmission power to serve both backhaul and access traffic. Each wireless card connects to a 15 dBi omni-directional antenna with a vertical beamwidth of 8 degrees. The backhaul antennas are attached to the sides of homes at 10m height, and at slightly greater height (maximum of 20m) at libraries, schools, and businesses.

Figure 2.20: Connectivity graph of the TFA backhaul with appropriate scaling for distance between nodes.

We select four nodes in the TFA network to form the topology of Scenario 11 shown in
Fig. 2.1. In each experiment performed in TFA, we generate backlogged traffic using Iperf v.1.7.0 and measure the achieved throughput. Our measurement intervals are 30 seconds. The PHY rate is set to 11 Mbps. The data packet size in our experiments is 1500 bytes. Other MAC and physical layer parameters are default IEEE 802.11b parameters. We test the case with RTS/CTS on and show the result in Fig. 2.21. We observe that as the model predicts, in the TFA network flow B receives much higher throughput than flow A. The difference between the model prediction and the measurement result for flow B is due to the performance anomaly effect [25]. In the SMC 2532-b wireless card deployed in TFA, although the transmission rate is fixed to 11M Bps, when the card experiences three consecutive collisions, it will set its rates to 2Mbps. High collision probability of flow A forces this flow to transmit most of its data packets at 2Mbps, thus significantly reducing the throughput of flow B which is transmitting at 11Mbps.

![Figure 2.21: Model predictions vs. measurement results. (with RTS/CTS, scenario 11).](image)

2.4 Symmetric Incomplete State and Short Term Unfairness

In this section we develop an analytical model to study the behavior of flow pairs belonging to the Symmetric Incomplete State (SIS) group, comprising Scenarios 8, 9 and 10 in Figure
2.1. We explore the resulting short-term unfairness by means of analysis and examine the impact of various protocol parameters.

2.4.1 Analytical Model

The main difficulty in analyzing the scenarios of the SIS group resides in the fact that the behavior of the two flows is tightly correlated: when one flow starts dominating over the other, the states of the flows clearly cannot be considered to be independent, and we therefore cannot employ the decoupling technique adopted in Section 2.3. In order to correctly analyze the system, it is necessary to consider the joint behavior of the two flows.

Thus, we represent the system state as the pair \((S_A, S_B)\), where \(S_A\) and \(S_B\) represent the current backoff stages of transmitters \(A\) and \(B\), \(0 \leq S_A, S_B \leq m\), respectively. Recall that \(m + 1\) is equal to the maximum retransmission limit, which plays a fundamental role in the behavior of the flows as we show in Section 2.4.2. The total number of states is \((m + 1)^2\) yielding a computationally efficient solution.

Using our bi-dimensional state description, we build a discrete time Markov Chain embedded over continuous time at the time instants in which both senders can (potentially) start transmitting the first packet of a new data exchange (either the RTS or the DATA packet), provided that their backoff counter is equal to zero. We use the same channel view as depicted in Figure 2.13, but consider only time epochs at which both transmitters can attempt a new transmission.

We assume that the backoff counter of a station is geometrically distributed, instead of uniformly distributed, over the current window. By doing so, we can exploit the memoryless property of the geometric distribution and avoid explicitly incorporating in the state description the remaining number of backoff slots of each station. Our simulations indicate that this approximation does not compromise the model’s accuracy. The parameter \(\gamma_i\) of the geometric distribution that characterizes the backoff counter at stage \(i (0 \leq i \leq m)\) is given by \(\gamma_i = \frac{2}{W_i - 1}\), where \(W_i\) is the window size of backoff stage \(i\). Consequently, at each time epoch a station in stage \(i\) attempts a new transmission with probability \(\gamma_i\).
Transition Probabilities

We consider the behavior of the two flows in Scenario 8 (or equivalently, Scenario 9). The key point to analyze the system dynamics is the computation of the (conditional) packet collision probability $p$. We observe that sender $A$ (for example) does not know if sender $B$ has started accessing the channel until a packet sent by $B$ has been fully received at $b$, triggering the transmission of a CTS or ACK that can be immediately sensed by $A$. This leads to the following typical situation: sender $A$ starts transmitting a new packet, but before it is fully received by $a$, sender $B$ also starts transmitting a packet, resulting in a collision at the receivers in which both packet are destroyed. A transmission from sender $A$ is successful only if sender $B$ does not attempt to transmit in all transmission opportunities that occur during the duration of the first packet (either the RTS or the DATA packet) sent by $A$. The number of such transmission opportunities is given by the duration of the first packet (RTS or DATA) expressed in the number of backoff slots, and denoted by $f$. The (conditional) successful probability of sender $A$ in state $(i, j)$ is then given by $(1 - \gamma_j)^f$ (recall that $j$ is the backoff stage of sender $B$, and $\gamma_j$ the corresponding transmission probability).

<table>
<thead>
<tr>
<th>from state</th>
<th>to state</th>
<th>probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>$i, j$</td>
<td>$i, j$</td>
<td>$(1 - \gamma_i)(1 - \gamma_j)$</td>
</tr>
<tr>
<td>$i, j$</td>
<td>$0, j$</td>
<td>$\gamma_i (1 - \gamma_j)^f$</td>
</tr>
<tr>
<td>$i, j$</td>
<td>$i, 0$</td>
<td>$(1 - \gamma_i)^f \gamma_j$</td>
</tr>
<tr>
<td>$i, j$</td>
<td>$(i + 1) \mod m, (j + 1) \mod m$</td>
<td>otherwise</td>
</tr>
</tbody>
</table>

Table 2.3: Transition probabilities of the Markov Model.

The transition probabilities stemming from the generic state $(i, j)$ are summarized in Table 3. The first row of the table is the self transition corresponding to the case in which both flows do not start transmitting a new packet. The second and third rows refer to
successful transmissions from sender $A$ or $B$, respectively. Notice that in this case the backoff stage of the station successfully transmitting is reset to the initial value (stage 0). Finally, the last row corresponds to a collision event for both flows, with the consequent increase of the backoff stage – if the backoff stage reaches the maximum retransmission limit, it is reset to 0, which explains the modulus operator.

**Performance Metrics**

By numerically solving the Markov Chain, which is ergodic for any choice of parameters, we obtain the stationary distribution $\pi = \{\pi_{i,j}\}, \forall i, j$. An example of such a distribution is reported in Figure 2.22, obtained using the set of parameters in Table 5.4 and considering the RTS/CTS access mechanism. The plot clearly suggests a bi-stable behavior, in which the most likely states are those in which one flow maintains a small value of window size (thus obtaining high throughput), while the other falls into deeper and deeper backoff stages (thus obtaining low throughput). Only when the poor flow reaches the maximum retransmission limit, it resets its window and competes equally with the rich flow.

Long-term performance metrics can be obtained directly from the solution of the Markov Chain. From renewal-reward theory, the throughput of either flow is given by

$$T = \frac{\sum_{i,j} \pi_{i,j} \gamma_i (1 - \gamma_j)^f}{\Delta},$$

where $\Delta$ is the average duration of a step. The duration of a successful transmission is equal to $T_s$, as defined in Section 2.3. The idle slot is $\sigma$, while a collision has an approximate average duration of $T_c + \sigma f/2$, assuming that the colliding packet starts on average in the middle of the packet that is transmitted first. $\Delta$ is computed as the average of the duration of all possible events in all states, weighted by their respective probabilities.

The average conditional collision probability $p$ for each flow can be computed as

$$p = \frac{\sum_{i,j} \pi_{i,j} P_c(i, j)}{\sum_{i,j} \pi_{i,j} [P_c(i, j) + \gamma_i (1 - \gamma_j)^f]},$$

where $P_c(i, j)$ is the collision probability at state $(i, j)$ (fourth row of Table 3).
Figure 2.22: Steady state probabilities of the Markov model in case of RTS/CTS mechanism, default 802.11b parameters.

**Transient Analysis**

As SIS scenarios are long-term fair, we are most interested in evaluating the short-term unfairness of these cases. In particular, it would be desirable to have an indication of the average amount of time during which one flow experiences poor throughput while the other gets most of the available channel capacity. To this purpose, we consider the system states \((m, 0)\) and \((0, m)\), where one flow has reached the last backoff stage, while the other is at stage 0. These two states are expected to be reached in the two symmetric conditions in which one flow strongly dominates over the other. The average amount of time necessary to transition from one of these states to the other provides a good estimate of the duration for the system to switch from one equilibrium point to the other, i.e., the system's time-scale of unfairness.

To compute this, we exploit the symmetry of our system, and reduce the problem to finding the average time necessary to re-enter one of the above states (for example \((m, 0)\))
after having left it. To simplify notation, let \( i \) be state \((m, 0)\).

We proceed as follows. First, we remove all self-transitions of state \( i \), since we want to consider re-entries to this state only after a change of state. To do so, transitions leaving state \( i \) are re-normalized to sum up to one.\(^1\) The time actually spent in state \( i \) in between two successive visits to this state will be considered separately at the end of the computation.

After applying this modification to the transition probabilities, we recompute the stationary distribution of the model obtaining a different state vector \( \pi' \). From the renewal-reward theorem applied to cycles defined by visits to state \( i \), we know that the average number of visits to state \( j \) between returns to state \( i \) is given by \( E(V_{ij}) = \pi'_j / \pi'_i \). Moreover, the average number of transitions of type \( j \rightarrow k \) between returns to state \( i \) is given by [52]

\[
E(V_{ijk}) = \pi'_j p_{jk} / \pi'_i, \quad \forall i, j, k.
\] (2.18)

Using this result, we can compute the average time to return to state \( i \) after leaving it by summing all durations associated with transitions \( j \rightarrow k \), weighted by their expected average number of occurrences given by Equation (2.18). Finally, we add the average time spent in state \( i \) before leaving it, which was initially removed. This can be easily done by considering that the average number of steps spent in a state is geometrically distributed, with parameter equal to the exit probability from the state.

### 2.4.2 Simulations and Model Validation

In this section we validate the analytical model of the behavior of the two flows in Figure 2.11(a) and investigate several properties of this scenario with a focus on short-term unfairness. Our results are summarized in Table 2.4, in which we compare analytical predictions with \( n.s \) simulations in four different cases: (C1) RTS/CTS access, \( m = 6 \), \( CW_{\text{max}} = 1024 \); (C2) RTS/CTS access, \( m = 8 \), \( CW_{\text{max}} = \infty \); (C3) basic access, \( m = 3 \), \( CW_{\text{max}} = 1024 \); (C4) basic access, \( m = 6 \), \( CW_{\text{max}} = 1024 \).

---

\(^1\)Indeed, following all transitions leaving \( i \), the poor flow (flow \( A \)) resets its window to the minimum value, exiting the starvation condition and starting to compete fairly with the rich flow.
<table>
<thead>
<tr>
<th>case</th>
<th>model</th>
<th>ns</th>
<th>SC</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>T</td>
<td>p</td>
<td>Δt</td>
</tr>
<tr>
<td>C1</td>
<td>218</td>
<td>0.25</td>
<td>235</td>
</tr>
<tr>
<td>C2</td>
<td>229</td>
<td>0.11</td>
<td>982</td>
</tr>
<tr>
<td>C3</td>
<td>125</td>
<td>0.69</td>
<td>15</td>
</tr>
<tr>
<td>C4</td>
<td>222</td>
<td>0.37</td>
<td>59</td>
</tr>
</tbody>
</table>

Table 2.4: Average throughput per flow T (in pkt/s), conditional packet loss probability p, and average transient time Δt (in ms) obtained in Scenario 8 for four different settings of parameters.

Case C1 corresponds to the default parameters of 802.11 when RTS/CTS is used, having the maximum retransmission limit equal to 7, the Short Retry Limit. Case C2 also employs RTS/CTS, but increases the maximum retransmission limit to 9 and does not bound the maximum window size. Case C3 corresponds to the default parameters of 802.11 with basic access, having maximum retransmission limit 4, and the Long Retry Limit. Case C4 differs from C3 in that the maximum retransmission limit is set to 7, the Short Retry Limit.

We compare both long-term performance metrics (average throughput and collision probability) and our characterization of the short-term unfairness by means of the average time Δt to transition from state \((m, 0)\) to \((0, m)\). This latter quantity is computed analytically using the approach described above and has also been measured in simulation via inspection of the backoff stage of the two flows. The last column of Table 2.4 reports the per-flow throughput (in pkt/s) achieved in each case when senders are connected (SC).

In addition to the excellent agreement between analysis and simulations in all cases, we make the following observations. All configurations achieve similar throughput (around 220 pkt/s) except case C3, for which we observe a severe penalty (62%) mainly due to the large packet loss probability combined with the small retransmission limit, which leads to many packets being dropped by the MAC. In the other cases the throughput loss with respect to the case in which senders are connected is not significant using RTS/CTS (around
10\%, cases C1 and C2), while it is important in case C4, in which basic access is employed (34%).

![Graphs C1, C2, C3, C4 showing packet successful sent every 100 ms.]

Figure 2.23 : Packets successfully sent by one flow every 100 ms, in all four cases.

The maximum throughput is achieved in case C2, where we have increased both the maximum number of retransmissions and the maximum window size with respect to standard values. The small gain in throughput comes at the expense of a severe short term unfairness, as the average transition time from one equilibrium point to the other approaches 1 second. This confirms the significant impact of $m$ on the time-scales of short-term unfairness.

Surprisingly, the basic/access mechanism with $m = 6$ achieves the smallest value of $\Delta t$ while obtaining a throughput comparable to that obtained using the RTS/CTS mechanism. This is remarkable, as the RTS/CTS mechanism was actually proposed to overcome the hidden terminal problem in infrastructure mode (see Figure 2.11(b)). Unfortunately, it
leads to further short-term unfairness than that obtained when the basic access mechanism is used.

As a further validations of our conclusions, we present in Figure 2.23 simulation results showing the average number of packets successfully sent by one of the two flows every 100 ms, in the window \([50s - 60s]\), for each of the four cases. We also show the average number of packets obtained when senders are connected, for comparison. We observe that the throughput oscillates dramatically in the cases having a large value of \(\Delta t\), enduring prolonged intervals in which the achieved rate is either very high or very low.

## 2.5 Related Work

**Problematic Topologies.** Variants of some of the scenarios we investigated are known to incur poor performance. Among them are the classic hidden terminal and exposed terminal problems examined in \([6, 33, 36]\). Likewise, AIS scenarios are described in \([6, 7, 32, 33, 51]\). Yet no prior studies comprehensively analyze all scenarios within a single analytical framework nor predict scenario likelihood in a random graph.

**Modeling Throughput.** The fully connected topology and Senders Connected (SC) group can be accurately modeled using \([8]\) and hence we do not consider them here. In contrast, we model topologies in which senders are disconnected, which leads us to develop new modeling techniques for both the AIS and SIS groups.

In \([42, 43]\), a queuing analysis of Scenario 11 in Figure 2.1 is developed under (unrealistic) assumptions that (i) the time between retransmissions (i.e. the backoff delay) is negligible, and (ii) the maximum number of retransmissions allowed for each packet is unlimited. Consequently, the model matches simulations well only under light traffic conditions and with very large packets transmitted at low data rate. In contrast, our analysis incorporates all details of 802.11 DCF, applies to both saturated and non-saturated conditions, incorporates variable packet sizes, and shows the significant impact of the maximum retransmission limit and access method (two- or four-way handshake). In \([23]\) the authors present a throughput analysis that incorporates topology dependent relations, and report
preliminary results for the ring topology neglecting the impact of the binary exponential backoff.

**Modeling Short-Term Unfairness.** While a number of studies have analyzed short-term unfairness due to various aspects of the wireless medium ranging from channel errors to contention, (see [5, 12, 13] for example), none accurately characterize the time scale in which 802.11 stations alternate domination in SIS scenarios. In particular, in our analysis of the SIS group, we model short-term unfairness of IEEE 802.11 with and without RTS/CTS, derive the behavior of the flows from the collision probability, analyze the transient behavior of the system, and accurately predict the time-scale of this unfairness as a function of key system parameters.

**Receiver Oriented Media Access.** A number of receiver-based access mechanisms, in which channel contention is initiated by the receiver, not the sender, have been proposed to improve fairness and performance in various scenarios [6, 18, 50]. The Request-for-Request-to-Send (RRTS) solution is an example receiver-oriented mechanism originally proposed in MACAW [6]: whenever a station receives an RTS to which it cannot respond (due to deferral), it contends during the next contention period on behalf of the sender, reserving the channel by means of an additional control packet called RRTS. The RRTS message solicits the sender to immediately send a new RTS.

![Graph](image)

**Figure 2.24:** Comparison of the number of packets sent by one flow during intervals of 100 ms in scenario 10, with or without RRTS

We have investigated the effectiveness of this solution in both the AIS and SIS group,
implementing the RRTS packet exchange in the \textit{ns} simulator. We have found that this solution has a fundamental limitation in the fact that it requires the receiver of a flow to correctly decode the RTS packet sent by the sender. Unfortunately, this occurs extremely rarely when a transmission attempt is not successful. Consider the scenarios of the AIS group: the RTS packet sent by \textit{A} typically arrives at \textit{a} during a transmission of flow \textit{B}, or it is destroyed by a new packet sent by \textit{B} before it is fully received by \textit{a}. Simulation results confirm that the RRTS mechanism does not help in the scenarios of the AIS group.

In the SIS group, the RRTS mechanism is also not useful in scenarios 8 and 9, because during a collision event both RTS packets are destroyed at the receivers. The only exception is scenario 10: in this case, the two flows are connected only through the receivers; the receiver of a flow is not in range of the transmitter of the other flow, and correctly receives the RTS sent by the sender. This scenario leads to short-term unfairness, and can be analyzed in a similar way as described in Section 2.4. Figure 2.24 shows that the RRTS mechanism partially mitigates the observed unfairness. Notice that the occurrence probability of Scenario 10 is only about 10\%, as reported in Figure 2.5.

\section*{2.6 Summary}

We have systematically and comprehensively investigated flows' unfair contention under the IEEE 802.11 protocol in the multi-hop topologies consisting of two flows and four nodes. We identified all possible topologies, classified them into three geometric groups, and computed their likelihood under random node placement. In each case, we showed how fundamental properties of two- and four-way handshake IEEE 802.11 yield short-term unfairness in one group, and long-term unfairness in another. We demonstrated that these problematic scenarios are highly likely to occur in a random graph, and developed analytical models that can accurately predict the performance in each of these scenarios. These models are able to precisely predict each flow's throughput in scenarios with long-term unfairness and the time scale that flows alternate domination in scenarios with short-term unfairness.
Chapter 3

Compounding Effect of MAC and Congestion Control

Chapter 2 analyzes the two-flow scenarios and shows that the IEEE 802.11 protocol may result in unfair contention even between two flows. A sliding window congestion control algorithm such as TCP utilizes a congestion window to limit the sending rate of the MAC protocol and therefore may dramatically change the throughput profile of the flows in the network. In this chapter, we are interested to explore the joint effect of the IEEE 802.11 MAC and window based flow control such as TCP.

It has been established that TCP does not perform well in wireless networks [14, 30, 31, 38]. In [16], the authors conclude that the poor performance of TCP is due to its congestion window being larger than its optimal value, which is determined to be one third of the number of hops from the source to the destination. Different from the prior work, this chapter shows that it is the sliding window congestion control and IEEE 802.11 MAC jointly induce unfairness. Even the TCP window is fixed to its optimal value suggested in [16], TCP can still perform poorly and lead to unfairness.

To explore the joint effect of MAC and TCP, we first identify a topology that is fundamental to a mesh network, in that this topology is necessarily embedded in any multi-hop mesh network. By analysis and experiments, we then demonstrate that severe unfair contention arises in this scenario, caused by the interaction of the MAC and TCP. We finally develop an analytical model to analyze their interaction and how they joint induce unfairness.
3.1 Analysis

We first consider a topology that is necessarily embedded in any larger mesh network topology given that mesh networks are defined as multi-hop wireless networks with gateways. This topology is shown in Fig. 3.1, in which two mesh nodes, A and B, are located two and one hop away from the gateway node, GW, respectively. Mesh nodes A and GW do not sense each other’s transmission, i.e., they are hidden from each other. Both A and B transmit a TCP flow to the gateway node GW. We call this scenario the basic scenario and name the two TCP flows TCP Flow A and TCP Flow B, respectively.

![Diagram](image)

Figure 3.1: The basic topology

In Fig. 3.1, if we replace the two TCP flows with two UDP flows, the scenario becomes a variation of the SC class described in chapter 2, which predicts that the two UDP flows will contention fairly because their senders are connected and have complete information of the channel. We now examine the joint behavior of the congestion control mechanism and the medium access mechanism in the basic topology shown in Fig. 3.1.

Medium Access and Bi-stability.

We first characterize the behavior of two flows in the scenario shown in Fig. 3.2, where the gateway node GW and two-hop mesh node A contend with each other for transmitting TCP ACK and TCP DATA, respectively. This is a variation of the SIS class shown in Fig. 2.1. According to Chapter 2, the SIS class consists of cases (8), (9) and (10) in Fig. 2.1. In all the three cases, if we combine the two receivers of the two flows, these scenarios become the scenario shown in Fig. 3.2. The analysis in Chapter 2 shows that the scenario in Fig. 3.2 exhibits bi-stability, which is explained in detail as below.

Assume the transmission queues of A and GW are backlogged at a given time, and both nodes are in the minimum contention stage. Since the two senders, namely A and
Figure 3.2: TCP DATA and TCP ACK are contending for channel access.

GW, are hidden from each other, a transmission from one sender succeeds only when it fits within the other sender’s backoff interval. Note that when the packet size of one sender is comparable to or larger than the contention window of the other sender, the probability of collision between the two senders is very high. For example, in IEEE 802.11b with default parameters, the collision probability between two RTS transmissions respectively from the two senders is 0.7. The collision probability for data packets with RTS/CTS off is even higher (e.g., nearly 1 for packets larger than 750 bytes in 802.11b). Thus, when both nodes are in an early backoff stage, the system is likely to experience collisions. After a series of collisions, the backoff window of both nodes will become sufficiently large such that one of the nodes will successfully transmit a packet.

Assume without loss of generality that node GW finally succeed in transmitting a packet. After this successful transmission, node GW resets its contention window back to its minimum size, while node A keeps a high contention window. In order for node A to succeed in its next transmission attempt, it must fit its packet in a small backoff interval of node GW, which is an unlikely event. After a resulting collision, the probability to succeed for each node is asymmetric, because the contention window of GW is much smaller than that of A. This process can repeat many times such that only node GW manages to transmit packets, while node A keeps increasing its contention window. When the contention window of A is high, GW can transmit multiple packets between two consecutive transmission attempts by A. A similar phenomenon has been previously illustrated in [6, 20].

To summarize, when mesh node GW (A) wins the channel, it enters a success state in which it transmits a burst of packets, while A (GW) enters a fail state in which it starves
and does not succeed in transmitting any packets. The success state can terminate for three reasons: (i) the probability of the node with higher contention window to win is low but not zero; (ii) the losing node drops the packet and resets its contention window after it reaches its maximum retry limit; (iii) the transmission queue of the winning node is emptied.

Figure 3.3: Illustration of bi-stability with alternation of \((A,B)\) and \((B,GW)\) transmissions.

Note that since node \(B\) is in sensing range with both \(A\) and \(GW\), it contends fairly with the node that is in the success state and interleaves its packets with the burst generated from this node. The collision avoidance mechanism in CSMA/CA causes bi-stability, in which node pairs \((A,B)\) and \((B,GW)\) alternate in transmission of multiple packet bursts. In particular, the system alternates between a state in which \(A\) and \(B\) jointly capture the system resources for multiple transmissions while \(GW\) is idle, and a state in which \(GW\) and \(B\) transmit while \(A\) is idle. This bi-stability is depicted in Fig. 3.3.

**Asymmetry Induced by TCP.** TCP causes the system to spend dramatically different times in the two stable states. Specifically, TCP’s congestion control mechanism creates a closed-loop system between each sender-receiver pair in which the transmission of new packets is triggered by the reception of acknowledgments. The basic scenario contains two nested transport loops, one for each flow. We term the one-hop and the two-hop loops as the inner loop and outer loop respectively, as depicted in Fig. 3.4(a). When in the stable state in which \((A,B)\) bursts and \(GW\) is in the fail state, both the outer and inner loops are broken (Fig. 3.4(b)), and hence, \((A,B)\)'s burst length is upper bounded by \(A\)'s TCP congestion window. Conversely, when \((B,GW)\) bursts, only the outer loop is broken, and the inner loop is self-sustaining due to the loop’s own ACK generation (Fig. 3.4 (c)). Consequently, the duration for \(GW\) and \(B\) to jointly capture the channel is not bounded. As a result, the
system spends much more time in the state in which \((B, GW)\) captures the channel than in the state in which \((A, B)\) captures the channel.

**Severe Penalties.** Due to asymmetric bi-stable states, node \(A\) and node \(GW\) exit their fail states differently, leading to a severe penalty only for the TCP flow originating from node \(A\). Recall that a node exits its fail state in the three ways described above. When \(GW\) is in the fail state, node \(A\)'s limited burst is not likely to drive \(GW\) to drop a packet. Hence, \(GW\) will most likely exit its fail state by case \((iii)\), i.e., the transmission queue of \(A\) is emptied. The penalty that node \(GW\) incurs is small due to short duration of its fail state. Furthermore, this penalty is shared by both TCP Flow \(A\) and TCP flow \(B\). On the other hand, when node \(A\) is in the fail state, the inner loop is self-sustaining, hence, the gateway queue is rarely empty. Consequently, node \(A\) most likely exits its fail state by case \((ii)\), i.e, by dropping the packet. The penalty node \(A\) incurs is high, including both the long duration of its fail state (MAC penalty) and TCP timeout, a duration which exponentially grows with multiple drops of the same TCP segment. This penalty is only paid by TCP Flow \(A\).
3.1.1 Broader Topology

A variation of the basic topology is shown in Fig. 3.5 (left), where $A$ and $C$ transmit a two-hop TCP flow and a one-hop TCP flow to the gateway node $GW$, respectively. In this case, although node $C$ does not forward traffic for node $A$, the same reasoning of unfairness origins applies. The gateway $GW$ and $A$ are out of carrier sense range yielding bi-stable behavior. When $GW$ and $C$ obtain the channel, the one-hop loop is self-sustaining. When $A$ and $B$ obtain the channel, $GW$ is in fail state and both loops are broken. Consequently, the burst size of $A$ is limited by its congestion window.

![Diagram](image)

Figure 3.5: Two-branch scenario. $A \rightarrow B \rightarrow GW$ is one branch. $C \rightarrow GW$ is another branch of the mesh.

3.2 Verification

According to the analysis above, TCP congestion control and CSMA medium access jointly cause long-term unfairness in the scenarios shown in Fig. 3.1 and 3.5. To validate that TCP flows indeed incur unfairness in these scenarios, we perform experiments in the operational Technology For All network introduced in 2.3.5. The connectivity graph of TFA and the nodes we selected for our experiments are shown in Fig. 3.6.

In each experiment performed in TFA, we generate TCP traffic using *Iperf* v.1.7.0 and
Figure 3.6: Connectivity graph of the TFA network backhaul topology with appropriate scaling for distance between nodes. There are nearly 2,500 residential users within the coverage area. Nodes A, B and GW are the nodes used throughout the measurements as the basic topology within the TFA network. Node C used for the second branch topology discussed in Section 3.1.1.

measure the achieved throughput. Before each experiment, we measure the throughput when each flow is singly active to ensure good channel state. We use Kismet v.2006.04.R1 to collect MAC-level traces at selected network nodes. Unless stated differently, our measurement intervals are 120 seconds, the maximum PHY rate is 11 Mbps, and the radio band is channel 6 of the 2.4 GHZ ISM band. By default, the RTS/CTS mechanism is not used by the TFA mesh nodes.

3.2.1 Experiment Results

Unfairness in the basic topology. Here, through experiments conducted in the TFA network, we demonstrate that the two flows shown in Fig. 3.1 receive dramatically different throughput. We perform experiments at different times of the day and night, in order to consider different background load.

For our measurements, we select the gateway GW, second-hop node A and first-hop node B in TFA. Although A is physically close to the gateway, it is not in radio range of the gateway due to the propagation environment, i.e., A and the gateway are neither in
transmission range nor in interference range of each other due to obstacles such as trees and houses. All of A's packets to and from the gateway are forwarded by node B, as verified by observing the routing table.

In each trial of the experiment, we simultaneously generate a TCP flow from the two-hop node A and a TCP flow from the one-hop node B to the gateway GW. Thus, in all experiments, all three nodes mutually contend for channel access in support of both uplink data and downlink acknowledgments.*

![Graph showing throughput](image)

Figure 3.7: Two-hop chain contention with RTS/CTS off measured at six different times of the day.

Fig. 3.7 depicts the throughput of the two flows when RTS/CTS is off and illustrates that severe unfairness occurs. In particular, the one-hop TCP flow from node B dominates whereas the two-hop TCP flow from node A receives nearly zero throughput in all experiments. Similar results are observed for cases with RTS/CTS on. Note that to precisely

---

*To ensure that our results are not unique to injecting a single flow from both nodes in the presence of many background flows, we also generate aggregate flows from both nodes and obtain nearly identical results which are not shown.
determine the proportional fair rates in this scenario is beyond the scope of this thesis. Here to estimate the fair rates, we simply assume that the aggregate utilization in this clique \(^\dagger\) does not change when the per-flow throughput of Flow A and Flow B changes. Under this assumption, the maximizer of the following optimization problem are the proportional fair rates of the basic scenario:

\[
\begin{align*}
\text{maximize} & \quad \log(x) + \log(y) \\
\text{subject to} & \quad x + 2y = C \\
\text{over} & \quad x \geq 0, y \geq 0,
\end{align*}
\tag{3.1}
\]

where \(y\) and \(x\) are the throughput of Flow A and Flow B respectively and \(C\) is a constant value. By solving this optimization problem, we observe that to achieve proportional fairness, \(y\) has to be equal to \(\frac{x}{2}\).

**Unfairness in the two-branch topology.** To verify unfairness in the scenario shown in Fig. 3.8 (left), in TFA we select another one-hop node \(C\) besides node \(A\), \(B\) and \(GW\). As depicted in Fig. 3.5 (left), two TCP flows are active on the two branches \(A \rightarrow B \rightarrow GW\) and \(C \rightarrow GW\), respectively. Fig. 3.5 (right) depicts the result of the experiment and shows that unfairness does persist in this two branch topology. As expected, the behavior of the TCP flow pair \(A \rightarrow B \rightarrow GW\) and \(C \rightarrow GW\) is strictly analogous to the behavior of the pair \(A \rightarrow B \rightarrow GW\) and \(B \rightarrow GW\) discussed above. In this experiment, RTS/CTS is on. Similar unfairness is observed when RTS/CTS is off.

### 3.3 Modeling

In this section, we develop an analytical model to study the compounding effects of medium access and congestion control on unfairness. We employ a highly simplified system model in order to isolate and study the root causes of unfairness under the simplest conditions in which they arise.

\(^\dagger\)Defined as a network where spatial reuse is not possible.
3.3.1 System Model

As described in Section 3.1, the DATA-ACK control loop is a key factor in unfair contention between the two flows. Consequently, we model only one aspect of congestion control, the sliding window, and in particular, we consider a fixed congestion control window. When the corresponding analytical model predicts unfairness, we can conclude that among congestion control’s many mechanisms, the DATA-ACK control loop and a sliding window alone are sufficient to induce unfairness.

For medium access, we also consider a simplified system model with an idealized physical layer in which node pair (GW, B) and node pair (A, B) can communicate without channel errors. We do not consider physical layer capture effect, i.e., we assume that overlapped transmissions fail. We consider that the initial contention window of node i is given by $CW_{\text{min},i}$, and the contention window evolves according to the binary exponential backoff scheme. Moreover, we assume that the backoff counter of each station is geometrically distributed over the current window. This assumption allows us to exploit the memoryless property of the geometric distribution and to avoid tracking the number of mini-slots already elapsed. This assumption is common and has been previously validated, e.g., [20, 37, 45]. Our model captures both RTS/CTS on as well as pure CSMA with
RTS/CTS off.

In addition to medium access and end-to-end sliding window, we also model the queues at each node. We assume that a node contains a separate queue for each subflow, e.g., node B has a queue for downlink ACKs to node A, a queue for uplink DATA originating from A, and a queue for uplink DATA originating from B. Moreover, each time a node gains channel access, each of the node’s queues receives service with equal probability. This assumption provides a memoryless property thereby aiding the model’s tractability.

We will show that while this system model omits many aspects of our experimental system, it nonetheless captures unfairness.

### 3.3.2 Model Description

As shown in Fig. 3.9, six sub-flows originated from the three mesh nodes need to be modeled. Included in the six sub-flows are three upstream DATA flows and three down-stream ACK flows, traversing to and from the gateway node, respectively. Correspondingly, we need to track the queue occupancy of the six sub-flows as shown in the figure.

![Diagram showing network queues](image)

**Figure 3.9:** Queues at different mesh points.

Eight channel states are identified including three *DATA transmission states* occupied by upstream DATA transmissions on links 1, 2, and 3; three *ACK transmission states* occupied by ACK transmissions on links 4, 5, and 6; one *collision state* occupied by RTS (or DATA if the RTS/CTS mechanism is not used) collisions between the second-hop and gateway node; and one *idle state* occupied by an idle mini-slot to characterize when all nodes
are counting down their backoff counters. These channel states are illustrated in Fig. 3.10, where the time instants of a possible channel state switch are pointed by arrows placed below the temporal axis. We label a transmission channel state using the index of the link on which this transmission occurs. For example, channel state 4 refers to transmission on link 4. We denote the duration of the transmission states, the collision state, and the idle state by \( T_i(1 \leq i \leq 6) \), \( T_c \) and \( T_5 \), respectively.

![Figure 3.10: Illustration of channel states.](image)

We now construct a Markov chain model embedded over continues time at mini-slot boundaries in which all three nodes can (potentially) start transmitting the first packet of a new data exchange (either the RTS or the DATA packet), provided that their queue is not empty and their backoff counter reaches zero.

For the ease of presentation, in notations we use \( a, b \) and \( g \) to represent nodes \( A, B \) and \( GW \), respectively. For node \( i \), \( (i \in \{a, b, g\}) \), the success probability of the geometric distribution that characterizes the backoff counter is given by \( e_i = \frac{q}{CW_i} \), where \( CW_i \) is the current contention window of node \( i \). With \( e_i \) computed above, the mean backoff interval with geometric distribution is set to be the same as with the system’s actual uniform distribution. Consequently, at any state-switching time epoch, a node with contention window \( CW_i \) attempts a new transmission with probability \( e_i \).

We denote the length of queue \( i \) for link \( i \) as \( Q_i \). Let \( Q_g = Q_a + Q_b \) be the aggregate queue length at node \( GW \), and \( W_a \) and \( W_b \) be the fixed congestion window for flow \( A \rightarrow GW \) and flow \( B \rightarrow GW \), respectively. Note that \( W_a \) and \( W_b \) are constant values. Because the middle node is in radio range of the two other nodes, the collision probability between
the middle node and one of the other two nodes is very small,\footnote{To collide, the middle node has to send the first packet of a data transmission within the propagation delay of one of the outer nodes.} compared to the collision probability between $A$ and $GW$. We therefore assume that the middle node never doubles its backoff counter, i.e., $CW_b = CW_{\text{min},b}$.

In order to capture both the MAC contention status and the queue behavior, we represent the system state as $S = \{Q_1, Q_2, Q_3, Q_g, \Omega_a, \Omega_g\}$, where $\Omega_a, \Omega_g$ denote the current backoff stage of node $A$ and $GW$, respectively. Although the length of some of the queues is not incorporated in the system state, they can all be expressed with $(Q_1, Q_2, Q_3, Q_g)$ as follows:

\begin{align*}
Q_4 &= W_b - Q_3 \\
Q_5 &= Q_g - (W_b - Q_3) \\
Q_6 &= W_a + W_b - (Q_1 + Q_2 + Q_3 + Q_g)
\end{align*}

(3.2)

### 3.3.3 Transition Probability Computation

To compute the transition probabilities given a system state, we first use the queue occupancy to determine the set of nodes that are contending for channel access. Since the next state that the system switches to depends on the contention outcomes, we compute the probability that each of the possible contention outcomes occurs. The key to compute these probabilities is to handle hidden terminals, which is described below.

We now consider system state $(Q_1, Q_2, Q_3, Q_g, \Omega_a, \Omega_g)$ in which each queue of Fig. 3.9 has packets to send. This is the state in which the computation of the transition probability is most involved due to the fact that all nodes are contending. We therefore show the computation of the transmission probabilities through this example. For system states in which not all queues have packets to send, the transition probability can be similarly computed.

With all queues backlogged, all three nodes contend for channel access at the next state switching time, in which node $i$, ($i \in \{a, b, g\}$) attempts to transmit a packet (RTS or data packet depending on which hand-shake mechanism is used) with probability $e_i = \frac{2}{CW_i}$. Let
\( f \) denote the duration of this contending packet expressed in the number of mini-slots. The second hop node \( A \) successfully transmits a packet only if (1) it attempts to transmit in the next mini-slot, (2) the middle node does not attempt to transmit in the next mini-slot, and (3) the gateway does not attempt to transmit in the next \( f \) mini-slots. Thus, the successful transmission probability of the second hop node is given by

\[
e_a (1 - e_b) (1 - e_g)^f,
\]

which is the transition probability from the current state to \((Q_1 - 1, Q_2 + 1, Q_3, Q_g, 0, \Omega_g)\).

All of the possible next states and their transition probabilities can be computed similarly and are summarized in Table 3.1. When collision occurs, both the second hop and the gateway increase their backoff to the next stage, e.g., after \( k \) collisions \( CW_i = 2^k CW_{\text{min},i} \) for binary exponential backoff. If the backoff stage reaches the maximum retry limit denoted by \( R_L \), it is reset to 0, which explains the modulus operator. When a node with more than 1 non-empty queue wins contention, these queues have equal probability to transmit their head-of-line packet, which explains the division operator.

<table>
<thead>
<tr>
<th>which link</th>
<th>to state</th>
<th>probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>link 1</td>
<td>((Q_1 - 1, Q_2 + 1, Q_3, Q_g, 0, \Omega_g))</td>
<td>(e_a (1 - e_b)(1 - e_g)^f)</td>
</tr>
<tr>
<td>link 2</td>
<td>((Q_1, Q_2 - 1, Q_3, Q_g + 1, \Omega_a, \Omega_g))</td>
<td>(1 - e_a, e_g(1 - e_g)^f)</td>
</tr>
<tr>
<td>link 3</td>
<td>((Q_1, Q_2, Q_3 - 1, Q_g + 1, \Omega_a, \Omega_g))</td>
<td>(1 - e_a, e_g(1 - e_g)^f)</td>
</tr>
<tr>
<td>link 4</td>
<td>((Q_1, Q_2, Q_3 + 1, Q_g - 1, \Omega_a, 0))</td>
<td>(1 - e_a, e_g(1 - e_g)^f)</td>
</tr>
<tr>
<td>link 5</td>
<td>((Q_1, Q_2, Q_3, Q_g - 1, \Omega_a, 0))</td>
<td>(1 - e_a, e_g(1 - e_g)^f)</td>
</tr>
<tr>
<td>link 6</td>
<td>((Q_1 + 1, Q_2, Q_3, Q_g, \Omega_a, \Omega_g))</td>
<td>(1 - e_a, e_g(1 - e_g)^f)</td>
</tr>
<tr>
<td>colliding</td>
<td>((Q_1, Q_2, Q_3, Q_g, (\Omega_a + 1)%R_L, (\Omega_g + 1)%R_L))</td>
<td>((1 - e_g)(e_a + e_p - e_a e_p - e_a (1 - e_p)^f - e_p (1 - e_a)^f))</td>
</tr>
<tr>
<td>none</td>
<td>((Q_1, Q_2, Q_3, Q_g, \Omega_a, \Omega_a))</td>
<td>otherwise</td>
</tr>
</tbody>
</table>

Table 3.1: Transition probabilities of the Markov Model, when one of the queues is empty.
3.3.4 Throughput Computation

After computing all transition probabilities, we can numerically solve the Markov Chain and obtain the stationary distribution $\Pi = \{\Pi_i, 1 \leq i \leq H\}$, where $H$ is the total number of system states, given by

$$H = R_L^2(W_a + 1)^2(W_b + 1)(W_a + W_b + 1). \quad (3.3)$$

We now compute binary matrix $\varphi_i$ for the transmission channel state $i$, $(1 \leq i \leq 6)$, $\varphi_c$ for the collision state, and $\varphi_\delta$ for the idle state. These matrices have the same dimension as the transition matrix and can be computed as follows. Suppose the system makes a transition from system state $m$ to system state $n$, where $m$ and $n$ are index of the system state. When making this transition, if a successful data transmission on link 1 occurs, we set $\varphi_1(m, n) = 1$; otherwise, we set $\varphi_1(i, j) = 0$. Similarly, when making this transition, if a collision occurs, we set $\varphi_c(m, n) = 1$. If none of the nodes attempts a new transmission, we set $\varphi_\delta(m, n) = 1$. Let $M$ be the $H \times H$ transition matrix. Then the occurrence probability of each channel state can be computed as

$$p_i = \sum (\Pi \times (\varphi_i \cdot M)), \quad i \in \{1, 2, 3, 4, 5, 6, c, \delta\}, \quad (3.4)$$

in which, the operator $\cdot$ denotes inner product, and the operator $\sum$ denotes the operation that adds all elements of a vector. The throughput of the two flows originating from node $A$ and $B$ is then expressed in pkts/s as,

$$\lambda_a = \frac{p_6T_6}{\Delta}, \quad \lambda_b = \frac{p_4T_4}{\Delta}, \quad (3.5)$$

in which $\Delta$ is the average duration of the channel states, computed as the average of the duration of all channel states, weighted by their respective probabilities. Recall that $T_6$ and $T_4$ are the duration of transmission state 6 and 4, respectively. To compute the duration of the collision state, we assume that, on average, the colliding packet starts in the middle of the packet that is transmitted first.
3.3.5 Model Validation Through Simulations

Before we validate the model, we look at how we can improve fairness in the basic scenario. As our above analysis shows, collisions between node A and node GW break the TCP loops and cause the system to spend much more time in one stable state than in the other. Since A and GW are hidden terminals with respect to each other, the transmissions of their contending packets are not coordinated through carrier sense. If they both have packets to transmit at a given time, with high probability a collision would occur. To reduce the probability of collisions between A and GW, we can reduce the backlog of these two end nodes. This can be done by increasing the minimum contention window of node B. With increased minimum contention window, node B contends less aggressively with the two end nodes, and the queuing of the system is shifted to B and thus the backlog of the two end nodes is reduced. While we will discuss the solution in detail in Chapter 4.2, we here know that \( CW_{\text{min}} \) of node B is an important system parameter in improving fairness. Thus, we vary this parameter in our model validation.

We first validate the assumption that backoff counter is geometrically distributed using ns-2 simulations. The parameters used in both the model and the simulations are default parameters of IEEE 802.11b. Because the siximensional Markov chain leads to a large state space as shown by Equation (3.3), we numerically solve the model for \( W_a = W_b = 3 \), i.e., both flows are modeled as having a fixed congestion window of 3 packets. To simulate fixed-window congestion control, we set the congestion window of TCP in the simulator to 3. We vary the minimum contention window of \( CW_{\text{min}} \) of the middle node and evaluate the impact on throughput. We observed in Fig. 3.12 that the model accurately predicts the throughput of the two flows.

3.3.6 Model Validation Through Experiments

Through measurements in real networks, we validate whether the model captures the trend of the two TCP flows when \( CW_{\text{min}} \) of node B increases. The platform used in our experiments is termed \textit{MirrorMesh}. MirrorMesh nodes are desktop PCs with a Linux Operating
Figure 3.11: Analytical model predictions compared to simulation results.

System (kernel 2.6) and Atheros wireless card (Madwifi v. 0.9.2 driver) that allows $CW_{\text{min}}$ to be changed. Each desktop PC connects to an external omni-directional antenna. All parameters for MAC and physical layer are according to IEEE 802.11b standard except the minimum backoff window, which is 16 by default in Atheros chip set.

Fig. 3.12 reports the model predictions and the measurement results. We observe that when $CW_{\text{min}}$ increases, the trend of the two flows’ throughput predicted by the model agrees with that measured on our platform.

### 3.4 Discussion

In mesh networks, the basic topology shown in Fig. 3.1 or its variation shown in Fig. 3.5 (left) is necessarily embedded in larger scenarios such as long-chain and broad-tree topology. In these larger scenarios, although there are other factors that affect the behavior of the contending flows, since all flows finally converge to the gateway, the embedded basic scenario plays an important role in determining the throughput of each flow. Indeed, our
extensive experiments demonstrate that in these larger scenarios, similar unfairness occurs, i.e., the first hop flow(s) starve flow(s) that are more than two hop away from the gateway. The results of these experiments are reported in [46].

Finally, we comment on the number of radios used in each mesh node. In our work, we consider one backhaul radio with or without a second access radio, thereby covering commercial architectures of Tropos, Cisco, Nortel, and others. Nevertheless, with multiple radios, if the number of radios is not sufficient to allocate orthogonal channels to every interfering wireless link, the results of this paper are still pertinent. In fact based on the previous subsection, whenever a two-hop transmitter is assigned the same channel with a one-hop transmitter, unfairness can occur.

3.5 Related Work

There is significant prior work investigating fairness, congestion control, and medium access in multi-hop wireless networks. Below, we separately address related studies concern-
ing single-hop and multi-hop flows given the fundamental difference between these two categories as described below.⁵

We refer to a flow as a single-hop flow if the source of the flow can reach its destination within one hop. Single-hop flows can exist in both single-hop topologies in which all nodes sense each other’s transmission and multi-hop topologies in which they do not. Single-hop flow studies showed both analytically as well as by simulation that in a fully backlogged scenario without flow control mechanisms (e.g., UDP traffic), network resources can be shared unevenly between contending flows. It was shown that MAC mechanisms ranging from binary exponential backoff to the use of carrier sense itself can cause unfairness [3, 6, 19].

In contrast, we consider multi-hop flows, which yield a significant difference from single-hop flows. For example, the memory introduced due to receipt and subsequent forwarding of the same packet adds multiple dimensions to the modeling problem as we describe in Section 5.3.

Poor performance of multihop TCP flows has been previously established [21, 49]. Furthermore severe unfairness has been observed when multiple TCP flows compete for the same wireless medium [27, 30, 31, 38]. Finally, a simplified model of IEEE 802.11 MAC and TCP features for multi-hop flows can be found in [35], where a single TCP flow is modeled over a two-hop chain assuming that the TCP transmission windows is fixed and neglecting MAC collisions (and hence neglecting binary exponential backoff issues).

None of these prior works identified nor modeled unfairness in the basic topology discussed here, which is the minimum and fundamental topology that inherently exists in mesh networks. Furthermore, none of these works identified that the congestion control mechanism changes the duration that the system spends in each of the two bi-stable states.

⁵We omit discussions concerning scheduled access wireless networks, as they are intrinsically different from random access wireless networks which are discussed in this work.
3.6 Conclusion

In this chapter we identified the existence of severe unfairness in the basic topology that is necessarily embedded in any multi-hop mesh networks. We then analyzed how TCP and MAC can jointly induce unfairness contention. We finally modeled the compounding effect of IEEE 802.11 MAC and sliding window congestion control on unfairness.
Chapter 4

Minimum Contention Window Policy

As our previous analysis shows, in an IEEE 802.11 multi-hop network where not all nodes are within range of each other, nodes may have different view of the channel state. When contending for channel access, some flows may have advantages over other flows. As a result of this unfair contention, some flows may receive very little throughput or even experience throughput starvation.

To improve fairness in such networks, one approach is to provide the contending flows with complete information of the channel state so that these flows may contend fairly with each other. In a distributed multi-hop network without centralized control, however, providing complete channel information to contending flows necessitates additional message exchange among flows. According to [10], the overhead due to message exchange is a multiplicative factor to throughput deduction of the network payload. This thesis will not pursue this direction further. With incomplete channel information, to improve fairness we let nodes that contend unfairly with each other transmit either at different times or at different frequencies. This chapter devises a simple contention window policy for mesh networks that shifts flow transmissions over time so that the problematic contention that lead to unfairness is less likely to occur compared to the original contention window settings. Next chapter explores the use of multiple channels to distribute flow transmissions to different frequency spectrum.

4.1 Minimum Contention Window Policy

In chapter 3 we developed an analytical model to study the joint impact of the MAC and the congestion control on unfairness in mesh networks. In summary, in the basic topology
shown in Fig. 3.1, node A and GW collide with high probability when A has TCP DATA packets and GW has TCP ACK packets to transmit. This high collision probability causes bi-stability of the system. TCP congestion control mechanism makes the time that the system spends in one stable state much longer than that in the other state and favors the one-hop flow.

The analytical model proposed in chapter 3 is validated in Fig. 3.12, which also reveals the solution to improve fairness. In particular, the figure shows that increasing the contention window of node B has the desired effect of providing fairness among the two flows. Consequently, we propose the following policy to improve fairness and counter starvation.

**Minimum Contention Window Policy:** All nodes that are directly connected to the gateway should increase their minimum contention window.

Analysis of the model’s state probabilities reveals the effect of the policy on the system queues shown in Fig. 3.9. Fig. 4.1 shows that when the minimum contention window of the first-hop node B increases, the probability that both Q₁ and Q₉ are empty dramatically increases and the probability that both Q₁ and Q₉ have packets to transmit at a given time dramatically decreases. The transmissions from node A and GW are shifted over time compared to the case with default minimum contention window. Recall that Q₁ is the queue at second-hop node A and Q₉ is the aggregate queue at gateway node GW. Having both of these queues empty indicates that most packets in the system are queued at B. Since B always contends fairly for the channel due to its ability to sense both A or GW (see Section 3.2), this is the ideal queuing point within the system. Consequently, collisions between A and GW are almost zero. Indeed, the model indicates that with large CW_{min} for the first-hop node, A and GW will rarely collide and rarely increase their backoff window.

Thus, the model indicates that the Minimum Contention Window Policy results in minimal queuing at the gateway and two-hop node for flows employing a sliding window protocol. Without these queues, the MAC protocol’s bi-stable behavior is broken. Without bi-stability, the “penalty to exit fail state” is very rarely incurred.
4.2 Evaluation of the Minimum Contention Window Policy

In this section, we evaluate our contention window policy's ability to counter starvation. As described in Section 4.1, the policy sets the minimum contention window of the gateway's immediate neighbors to a value larger than all other nodes. To evaluate our solution, we use our platform MirrorMesh introduced in Section 3.3.6.

To implement our $CW_{\text{min}}$ policy we need to change the minimum contention window of all of the gateway's immediate neighbors. Here, we describe the results of a set of experimental tests designed to validate the Minimum Contention Window Policy in the field. We perform all experiments on our platform MirrorMesh. The parameters for MAC and physical layer are according to IEEE 802.11b standard except the minimum backoff window, which is 16 by default in Atheros chip set. MirrorMesh contains no user-generated background flows such that the only traffic is that generated by our tests.

4.2.1 The Basic Topology

Here, we experimentally validate our Minimum Contention Window Policy on MirrorMesh. In this set of experiments, we measure per-flow throughput and compute aggregate log util-
ity for the basic topology, both for the default $CW_{\text{min}}$ and for increased $CW_{\text{min}}$ as recommended by the Minimum Contention Window Policy. Each experiment lasts 120s and the packet size is set to 1500 bytes unless stated otherwise.

We consider the scenario depicted in Fig. 3.1, in which nodes $A$ and $B$ both transmit packets to the gateway node, $GW$. As in TFA, we first verify that all links are operational and that $A$ and $GW$ are out of range.

**RTS/CTS on.** In this experiment, we enable RTS/CTS and set $CW_{\text{min}}$ of node $A$, $B$, and $GW$ to the default value of 16. Fig. 4.2(left) depicts a severe throughput imbalance and confirms that the system behavior for this scenario is consistent between MirrorMesh and TFA. We increase $CW_{\text{min}}$ of node $B$ to 128 and repeat the experiment. The result is also shown in Fig. 4.2(left), which indicates significantly improved throughput for flow $A \rightarrow B \rightarrow GW$. In this case, $A$’s throughput is improved from 0.056 Mbps to 1.23 Mbps. The aggregate log utility is improved from -0.6931 to 0.6523. One can easily follow Equation 3.1 and compute the upper bound of this metric to be 3.2917 by setting $C$ to the capacity of the channel.

![Graph](image)

**Figure 4.2 :** RTS/CTS enabled: unfairness with default $CW_{\text{min}}$ and Minimum Contention Window Policy result in the basic scenario of MirrorMesh.

**RTS/CTS off.** Fig. 4.3 report results for the case that RTS/CTS is disabled. We consider $CW_{\text{min}} = 16$ for all nodes as well as $CW_{\text{min}} = 128$ for node $B$. The results indicate that the Minimum Contention Window Policy is equally effective. With increased $CW_{\text{min}}$
at node $B$, the throughput of flow $A$ is improved from 0.025 Mbps to 1.44 Mbps and the aggregate log utility is improved from -1.91 to 0.996. The reason is that, as discussed in Section 5.3, our solution results in having all queued packets at $B$. Consequently, the hidden nodes, $A$ and $GW$, are not backlogged such that the probability that both $A$ and $GW$ have packets to send simultaneously and collide is negligible, irrespective of the RTS/CTS mechanism.

![Graph showing throughput comparison]

Figure 4.3: RTS/CTS mechanism disabled: unfairness with default $CW_{min}$ and Minimum Contention Window Policy result in the basic scenario of MirrorMesh.

**Baseline** $CW_{min} = 32$. In the above experiments, the baseline $CW_{min}$ is set to 16, the default value for Atheros. However, for most commercial wireless cards, including those deployed in TFA, $CW_{min}$ is set to 32 as recommended by the IEEE 802.11 standard. Consequently, we evaluate our policy for $CW_{min} = 32$ on MirrorMesh. We first set $CW_{min}$ to 32 for $A$, $B$ and $GW$ and collect the measurement results. Then we enlarge $CW_{min}$ of node $B$ to 128 and 256 and report the result for both cases in Fig. 4.4. With larger CWmin at node $B$, the throughput of flow $A$ is improved from 0.18 Mbps ($CW_{min} = 32$) to 1.02 Mbps ($CW_{min} = 128$) to 1.22 Mbps ($CW_{min} = 256$), and the aggregate log utility for the three windows are -0.28, 0.61 and 0.41 respectively. We observe that the proportional fairness is greatly improved with minimum contention window 128. If we further increase this window to 256, max-min fairness is achieved.

**Adoption of small packet size.** Because realistic traffic does not have only 1500 byte
Figure 4.4: Unfairness and Minimum Contention Window Policy result in a two-hop chain of MirrorMesh with the most common default minimum contention window setting for commercial devices ($C_W \text{min} = 32$).

packets, we next test the impact of packet size on the unfairness problem and on our solution. As shown in Fig. 4.5, the mere adoption of small packet sizes does not significantly affect the unfairness problem and our Minimum contention window policy is effective. The case with RTS/CTS on is shown in Fig. 4.5(left). When CWmin at node B is creased from 16 to 128, throughput of flow A is increased from 0.064 Mbps to 0.73 Mbps and the aggregate log utility is from -1.9 to -0.67. Fig. 4.5(right) shows the case with RTS/CTS off. The aggregate log utility is increased from -0.8886 to -0.2298.

Figure 4.5: Unfairness and Minimum Contention Window Policy result with 500 Byte TCP packets. Left: RTS/CTS on, right: RTS/CTS off.

When RTS/CTS is on, contending packets are RTS packets rather than data packets.
When RTS/CTS is disabled, data packets that are usually larger than RTS packets contend for channel access. This results in more severe bi-stability effect than with RTS packets. Our analysis in Section 3.1 and 5.3 captures that smaller data packets do not mitigate the unfairness problem, and that the Minimum Contention Window Policy is equally effective.

### 4.2.2 Extending the Basic Topology: Two Branches

In Section 3.1, we showed that severe unfairness occurs for flows not only on one branch in a mesh network, but also on two branches. In this experiment, we evaluate our solution in the scenario shown in Fig. 4.6 in which three flows are active on two branches. Fig. 4.7 reports that flow A is starved, whereas flow B and C almost equally split the bandwidth. We then invoke the Minimum Contention Window Policy by increasing $CW_{min}$ for both B and C, both of the gateway’s one-hop neighbors. As shown in Fig. 4.7, with our solution, the throughput of the second hop flow is dramatically improved from 0.01 Mbps for $CW_{Min} = 16$ to 0.25 Mbps for $CW_{Min} = 128$ and to 0.56 Mbps for $CW_{Min} = 256$. Correspondingly, the aggregate log utility is improved from -3.8 to -1.23 to -1.08. Note that in this case, the log utility is upper bounded by 3.2. The improvement is because with increased $CW_{min}$, most of the packets of flow $A \rightarrow B \rightarrow GW$ are queued at node $B$, and therefore contend with node $C$ more fairly.

![Diagram of two branch topology](image)

**Figure 4.6**: Two branch topology. TCP flow A is a two-hop flow. TCP flow B and C are two one-hop flows.

### 4.2.3 Downstream Traffic

Thus far, we have considered upstream data traffic. In Fig. 3.5, we reverse the direction of the flows such that DATA packets are transmitted from $GW$ to nodes $A$ and $B$, respectively,
Figure 4.7: Unfairness in a two-hop chain of MirrorMesh with default and increased contention windows at node B and C.

and TCP ACKs are transmitted from A and B to GW. In this scenario, both the hidden terminal effect in the MAC layer and the flow loops enforced by the sliding window congestion control remain the same as in uploading scenario. In this experiment, we show the presence of the unfairness problem and the effectiveness of our solution for downstream flows.

Figure 4.8: Unfairness and Minimum Contention Window Policy result in a downstream two-hop chain of MirrorMesh nodes.

Fig. 4.8 shows the throughput A and B receive when $CW_{\text{min}}$ is 16 for all three nodes and the throughput of A and B when $CW_{\text{min}}$ is set to 32. As predicted, unfairness indeed occurs in the download scenario, and our solution allows the two-hop downstream TCP flow to receive significantly higher throughput than the default window. With increased CWMin at node B, the aggregate log utility of the network is improved from -2.78 to
\begin{center}
\begin{tabular}{|c|c|}
\hline
Param & Value \\
\hline
SIFS & 10 $\mu$s \\
DIFS & 50 $\mu$s \\
EIFS & 364 $\mu$s \\
$\sigma$ & 20 $\mu$s \\
Basic Rate & 2 Mbps \\
Data Rate & 11 Mbps \\
PLCF Rate & 1 Mbps \\
$\left(CW_{\text{min}}, CW_{\text{max}}\right)$ & $(32,1024)$ \\
Short Retry Limit & 7 \\
\hline
\end{tabular}
\end{center}

Table 4.1: Parameters setting for the MAC and physical layers.

\section{Larger Scenarios via Simulation}

We use the \texttt{ns}-2 simulator to validate our solution in more general scenarios. We begin our simulations with a longer chain topology where spatial reuse is present. We then perform simulations on a topology in which three branches are connected to the gateway, with each branch further diverging. In all simulations, we use TCP-Reno for congestion control and IEEE 802.11b for medium access control. We use the MAC and physical parameters of Table 5.4.

\textbf{Four hop chain topology.} In a four-hop chain topology as depicted in Fig 4.9, spatial reuse is possible and there are an increase in the number of nodes out of carrier sense range. In this simulation, we explore whether these factors change the nature of the problem and solution. In this scenario, four mesh nodes simultaneously create long-lived TCP connections to the gateway. In previous experiments, separate queues are used for separate flows at first hop(s). In this simulation, we use one aggregate queue at the first hop to explore whether our window policy is effective in the this case.

Fig. 4.10 depicts the simulation results for all nodes having $CW_{\text{min}} = 32$ and node 1 (the gateway's one-hop neighbor) having $CW_{\text{min}} = 128$ following the Minimum Contention Window Policy. We observe that with all nodes having the same minimum con-
tention window, the first hop node receives an order of magnitude larger throughput than the sum of the throughput received by all other nodes. In contrast, by changing $CW_{min}$ of the gateway's neighbor to 128, throughput of the flows more than one hop away from $GW$ is much improved. Indeed, our Minimum Contention window policy improves the aggregate log utility from -14.0005 to -6.1415. Note that log utility of this scenario is upper bounded by 1.156.

![Gateway diagram](image)

**Figure 4.9**: TCP flows in four-hop chain topology.

![Throughput chart](image)

**Figure 4.10**: TCP throughput in a four-hop chain topology, with minimum contention windows all 32 (IEEE 802.11) and the one hop neighbor changed to 128 (New window).

In a longer chain topology, while spatial reuse is possible, nodes farther away from the gateway have less forwarding responsibility and are more lightly loaded. In contrast, nodes that are one and two hops away from the gateway still share the medium with all flows
and consequently, are the bottleneck. Thus, the unfairness problem in a longer chain has the same nature as in the two-hop chain topology, and our solution is just as effective in improving fairness.

4.3 Related Work

MAC-level solutions to unfairness among single-hop flows have been previously proposed including suggested modifications to binary exponential backoff [6, 51] and the hand-shake mechanism [6]. Likewise, in the context of 802.11e (which addresses QoS and service differentiation), some proposals allow different system parameters (Contention Window, SIFS and DIFS, etc.) for different traffic classes [40, 44, 47], thereby achieving performance differentiation.

To improve performance of congestion control in multi-hop wireless networks, proposals include hop-by-hop distributed congestion control [1, 54] and joint re-design of congestion control and medium access [14, 17]. Transport-level counter-starvation policies have also been proposed in which the TCP protocol is modified by adaptively slowing down the transmission rate [16, 29], limiting the TCP transmission window [16, 49] or modifying RED [16, 38].

Our solution is lightweight solution specifically designed for mesh networks. Our counter-starvation policy only modifies basic MAC protocol parameters and does not require any transport, network, nor MAC protocol modifications, nor does it necessitate any control message exchange.

4.4 Conclusion

In this chapter we devised a simple counter-starvation policy in which nodes one-hop away from the gateway increase their minimum contention window. We implemented and empirically validated the solution on MirrorMesh, an outdoor network test-bed, and through simulations.
Chapter 5

Asynchronous Multi-channel Coordination Protocol

In Chapter 4, we proposed a contention window policy to improve fairness for mesh networks. This chapter devises mechanisms to alleviate severe unfairness for ad-hoc networks through the use of multiple channels, which provides an opportunity to distribute transmission that would contend unfairly on a single channel to different frequency spectrum.

As this chapter shows, however, the flows receiving low throughput under single channel protocols may not necessarily receive the extra channel capacity made available by the use of multiple channels. Indeed, While previous multi-channel MAC protocols [2, 4, 28, 41, 48, 53] have been shown to increase aggregate network throughput, they do not provide mechanisms that prevent severe unfairness in multi-hop wireless networks. We show that without proper coordination of transmissions, the aggregate throughput may increase with the number of channels, but certain flows may still receive very low or even zero throughput.

This chapter proposes a multi-channel protocol to mitigate severe unfairness under the constraint of a single half-duplex radio at each node and the absence of centralized knowledge or infrastructure support. The solution, called Asynchronous Multi-channel Coordination Protocol (AMCP), uses the simple distributed access primitives of IEEE 802.11 DCF and provides analytical minimum rate guarantee for each flow in the network. Despite its simplicity, AMCP is designed to address unfairness of CSMA protocols in single-channel multi-hop wireless networks as well as the dual coordination problems that arise by the introduction of multiple channels.
5.1 Motivation and protocol design issues

Chapter 2 shows that the Asymmetric Incomplete Class (AIS) consisting of two-flows incurs long-term unfairness. In [19], another coordination problem named Flow-in-the-Middle (FIM) is identified to incur unfairness in the network. Further, it is shown in [19] that the superposition of these problems cause unfairness in general multi-hop wireless networks. Here we review these problems to motivate our protocol design. We use case (11) shown in Fig. 2.1 to review the the AIS class and then present the Flow-in-the-Middle (FIM) scenario. For the ease of presentation, in this chapter we term case (11) Information Asymmetry to explicitly represent the fact that the two flows in this scenario has asymmetric channel information.

We show that multiple channels can be used to improve fairness and compare two broad classes of solutions. We study two generic coordination problems inherent in a multi-channel system, namely the Multi-Channel Hidden Terminal problem identified in [48] and the Missing Receiver problem which we identify in this paper. These multi-channel coordination problems manifest in both classes of solutions and may cause performance degradation if not addressed properly.

5.1.1 Unfairness in IEEE 802.11 Single-channel Multi-hop Wireless Networks

When all transmitters are within range of each other it can be shown that IEEE 802.11 protocols provide fair access opportunities to all flows. Unfortunately, in a multi-hop topology where not all nodes are within range of each other, such protocols do not perform well, even if coordination enhancements such as RTS/CTS control packet exchanges [6] are used. More specifically, throughput distributions arise in which a few flows capture all bandwidth while many other flows get very low or even zero throughput. Such starvation phenomena are not merely due to having a different number of contenders for each flow, which is natural in a multi-hop topology; rather, they are due to coordination problems when CSMA-based access is used in a multi-hop environment. Here we illustrate these coordination problems that cause starvation through two characteristic examples.
**Information Asymmetry (IA).** The IA problem arises when the senders of two contending flows are not within radio range and have an asymmetric view of the channel state. Fig. 5.1(a) shows the topology of the IA problem, where the transmitter $B$ of flow $Bb$ is within radio range of the receiver $a$ of flow $Aa$ not in range of transmitter $A$. If both flows are backlogged, flow $Bb$ will receive significantly higher throughput than flow $Aa$. This is because the transmitter $B$ of flow $Bb$ knows exactly when to contend for the channel (through the control packets sent by the receiver of flow $Aa$). On the other hand, sender $A$ cannot sense the activity of flow $Bb$ and has to discover an available time slot only through random back-off. Since for efficiency purposes the ratio of data transmission interval to the idle slot size is usually large, most of these random attempts occur during the transmission of flow $Bb$ and result in collisions at receiver $a$. Repeated collisions trigger timeouts at sender $A$, which repeats doubling its contention window. As a result, the collision probability of flow $Aa$ is close to 1, while the collision probability of flow $Bb$ is close to 0. Figure 5.1(b) shows the channel state experienced by flow $Aa$.

![Diagram](image)

(a) Example Topology  
(b) Flow activities: Flow $Bb$ does not experience collisions. The random attempts of node $A$ to find an idle interval within the transmissions of flow $Bb$ result in RTS failures and exponential back-off.

**Figure 5.1**: Information Asymmetry (IA) example

**Flow-in-the-Middle (FIM).** The FIM problem arises when the sender of a flow senses the activity of neighboring nodes that are not within range with respect to each other. This behavior is illustrated in the three-link scenario of Fig. 5.2. If all flows are backlogged, the
middle flow $Bb$ will receive very low throughput, while the outer flows ($Aa$ and $Cc$) will receive maximum throughput. This is not due to high loss probability, but rather to the lack of transmission opportunities for the middle flow. More specifically, when one of the outer flows (say flow $Aa$) captures the medium, the transmitter of the middle flow $Bb$ will sense and defer but the transmitter of the other outer flow $Cc$ will continue contending and initiate transmission. When flow $Aa$ ends transmission, it will contend and initiate transmission, while flow $Bb$ now defers due to flow $Cc$. Fig. 5.2(b) shows that the misaligned concurrent transmissions of the outer flows may be sensed by the transmitter $B$ of the middle flow for extended periods of time. The middle flow has a chance to access the medium only when both outer flows are in the back-off phase (the vertical lines interval in Fig. 5.2(b)). Unfortunately, such occurrences become increasingly rare especially as the ratio of data transmission interval to the back-off interval increases.

![Diagram of Example Topology and Channel Activity](image)

(a) Example Topology  
(b) Channel activity sensed by the middle flow.

Figure 5.2 : Flow In The Middle (FIM) example

Note that both IA and FIM problems are not specific to the 802.11 DCF access mechanism. They are generic coordination problems that arise due to the asymmetry of the multi-hop topology and due to the use of carrier sense. In a general topology some flows experience the combined effect of both IA and FIM problems and their throughput may even reach zero. For an analytical model of starvation phenomena in single-channel CSMA multi-hop networks, see [19].
For convenience, in the rest of the chapter we use the term "advantaged flows" to refer to flows with geometry advantage (such as flow $Bb$ in Fig. 5.1(a) and the outer flows $Aa$ and $Cc$ in Fig. 5.2(a)) and the term "disadvantaged flows" to refer to flows with geometry disadvantage (such as flow $Aa$ in Fig. 5.1(a) and flow $Bb$ in Fig. 5.2(a)). We also maintain the convention of using capital letter for the transmitter and lowercase letter for the receiver of each flow.

5.1.2 Fairness Improvement Through Multiple Channels

In both the IA and FIM scenarios, the disadvantaged flow is unable to identify an idle interval because transmissions are generally misaligned and their durations are much larger than the back-off interval.

Clearly, unfairness would be eliminated if all transmissions occurred on orthogonal channels. Potential solutions can be classified into two approaches, as exemplified in Fig. 5.3 for the case of two flows. In the first approach, the entire transmission (including control and data transmissions) of each flow is scheduled on a different channel (Fig. 5.3(a)). The reason this approach can avoid unfairness is straightforward: an advantaged flow will not starve a disadvantaged flow because they both transmit on different channels. In the second approach, control packets are transmitted on a separate control channel and data packets of different flows are distributed to different data channels (Fig. 5.3(b)). This approach also alleviates unfairness: as the data packets have moved to different channels, contention occurs only on the control channel between short control packets, whose length is comparable to the back-off interval.

Each approach has its advantages and disadvantages. The advantage of the first approach is that it does not require the overhead of a dedicated channel for control messages and can potentially reduce the contention between the advantaged flows and the disadvantaged flows to zero. However, it can lead to logical partition where two nodes within range are unable to communicate. This is a significant challenge, especially when a node has only one transceiver and can only transmit or listen to one channel at a time. In the second
approach nodes immediately return to the control channel after finishing their data transmissions. The advantage of this approach is that nodes have a common channel (but not time) reference to coordinate their transmissions. The downside is that a dedicated control channel introduces overhead, which can be significant if its capacity is not appropriately designed.

5.1.3 Multi-channel Coordination Problems

Regardless of the solution approach, it is challenging to coordinate transmissions over different channels in an asynchronous setting 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 state on other channels. Hence, when it finishes communication it may attempt to exchange information with its neighbors while they are currently on other channels. To design an efficient protocol we must be able to
accurately characterize this lack of coordination. We investigate two generic coordination problems, classified with respect to their effect on the intended actions of control packets.

In the **Multi-channel Hidden Terminal Problem**, *control packets sent on a certain channel fail to inform neighboring nodes currently communicating on a different channel.*

An instance of this generic problem was first identified in [48]. To illustrate this problem we use a “naive” protocol that is a straightforward extension to IEEE 802.11 DCF for a multi-channel setting: The RTS/CTS control packets are exchanged on a dedicated control channel and reserve data channels for data packets. Nodes return to control channel immediately after they finish their data transmissions.

Now we consider again the two-flow topology of the IA scenario in Fig. 5.1. In this example, we assume that the protocol operates with two data channels. As shown in Fig. 5.4, a control packet exchange of disadvantaged flow $Aa$ may occur when the advantaged 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. Flow $Aa$ can receive very low throughput if there are many advantaged flows within its radio range.

![Figure 5.4: The Multi-Channel Hidden Terminal Problem.](image)

Although in this example we used a naive protocol where all control messages are exchanged in a dedicated control channel, it is evident that the problem is also present when control messages are transmitted on different channels. The multi-channel hidden
terminal problem limits the ability of control packets to block interfering flows. If no proper measures are taken it may result in very poor performance.

The **Missing Receiver Problem** arises when *control packets sent on a certain channel to access an intended receiver fail because this node is currently on a different channel (acting either as transmitter or receiver).*

To illustrate the problem, we consider the simple three-node scenario of Fig. 5.5, where node $A$ transmits to node $B$ and node $B$ transmits to node $C$. We first consider the naive protocol version where all control messages are transmitted on different channels. In Fig. 5.5, 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 delay for flow $AB$ and decrease its throughput.

![Figure 5.5: Missing Receiver Problem.](image)

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 other’s 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 $B$. Hence flow $AB$ will starve if no proper measures are taken.
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.

## 5.2 Asynchronous Multi-channel Coordination Protocol (AMCP)

We first illustrate the basic principles of AMCP and then present its implementation. Finally, we show how it addresses the multi-channel coordination problems.

### 5.2.1 Overview

Following the second approach of Section 5.1.2, AMCP uses a dedicated control channel on which nodes contend to reserve data channels by exchanging RTS/CTS packets according to 802.11 DCF. Upon successful control packet exchange, both the sender and the receiver switch to the reserved data channel, denoted by $x$, and transmit a data packet. After a data packet is successfully transmitted on channel $x$, the sender and receiver return to the control channel and set all channels as unavailable except $x$. They may contend for data channel $x$ immediately or contend for other data channels after the timers of these channels expire.

The RTS/CTS control packets serve a dual purpose: first, they aid two link endpoints to negotiate on commonly available data channels; second, they inform neighboring nodes to set the overheard data channels piggy-backed in RTS/CTS as unavailable for an entire data transmission interval. However, a node overhearing an RTS/CTS will not always defer for the entire data transmission; under certain conditions, it may initiate contention after the overheard RTS/CTS.

The exact deferring rules (described in Section 5.2.2) implement an efficient coordination scheme where nodes stay on the control channel long enough to learn about which channels to compete, while at the same time not always waiting for the entire data packet
transmission, thus increasing throughput. We proceed to describe the exact protocol operations.

5.2.2 Protocol Description

Structures and Variables

We assume one control channel and $N$ data channels, indexed from 1 to $N$. All channels are orthogonal with respect to each other. Each node has a single transceiver, hence it can either transmit or listen, but not both. Also it can listen to or transmit on one channel at a time. To execute AMCP, each node maintains the following structures and variables:

- A local $N$-entry Channel Table. Each table entry corresponds to a data channel and consists of a bit called $avail\_bit$ indicating channel availability, and a timer called $avail\_timer$ indicating the remaining time a channel is not available. Each time the channel becomes unavailable ($avail\_bit = 0$), its timer is set to expire after a data transmission duration. When the timer expires, the corresponding channel becomes available ($avail\_bit = 1$). By default, when a node joins the network all its $avail\_bits$ are set to zero.

- An integer $prefer$ variable takes values from 0 to $N$. If non-zero, this variable indicates that a node prefers to compete for the data channel indexed by $prefer$. If zero it indicates no preference.

Reservation/Transmission Cycle

Initially all nodes reside on the control channel. We now describe the protocol actions that occur when node $A$ has a packet intended to node $a$. We denote a neighboring node of $A$ or $a$ as node $C$.

Step 1: Channel selection. Node $A$ selects a data channel by inspecting its channel table. Among the available data channels, the channel indexed by $prefer$ is selected if $prefer$
is non-zero and available. Otherwise one of the available data channels is randomly chosen. If no data channel is available, the node waits until any of the avail_timers expires.

**Step 2: Channel contention.** Suppose that data channel $x$ is selected. Node $A$ inserts the index $x$ to its RTS packet and contends on the control channel using the 802.11 DCF CSMA/CA mechanism. In AMCP, a control channel’s NAV interval expires at the end of a RTS/CTS transmission, rather than the end of a DATA/ACK transmission as in IEEE 802.11.

**Step 3: Channel negotiation.** When node $a$ receives the RTS packet, it inspects the status of channel $x$ in its channel table. If $x$ is available, node $a$ replies to $A$ with a Confirming CTS packet containing index $x$. Then, it switches to data channel $x$ and waits for a DATA packet. If channel $x$ is not available, node $a$ replies to $A$ with a Rejecting CTS packet containing index 0 and a list of its available data channels, and remains on the control channel.

If node $A$ receives a Confirming CTS, it switches to channel $x$ and transmits the DATA packet to $a$. If $A$ receives a Rejecting CTS, it randomly selects a channel available in both its channel table and the channel list included in the CTS packet, then it inserts the index of this channel in a RTS packet and begins a new contention cycle on the control channel.

**Step 4: Data transmission.** Upon reception of the DATA packet, node $a$ responds with an ACK on data channel $x$, then switches back to the control channel. Upon reception of the ACK packet, $A$ also switches back to the control channel. The packet transmission has completed successfully.

**Step 5: Setting channel availability.** After $A$ returns to the control channel it sets its prefer variable to $x$; $A$ also sets the avail_bit unavailable and starts avail_timer for all other data channels except $x$. Node $a$ sets its prefer variable and Channel Table in the same way. Node $A$ restarts step 1 if there is a packet in its transmission queue.

We note that errors in the transmitted control and data packets are handled with timeout mechanisms similar to 802.11. If a timeout occurs while a node resides on a data channel, the node returns to the control channel, sets its prefer variable to 0, sets the avail_bit
unavailable and starts \textit{avail}\_\textit{timer} for all data channels.

\textbf{Overhearing nodes’ deferral rules}

Let $C$ be a neighbor of either $A$ or $a$. When $C$ overhears an RTS packet, it first updates its channel table by setting its $\text{avail}\_\text{bit}(x) = 0$ and sets $\text{avail}\_\text{timer}(x)$ to expire at the end the full data packet transmission (for a duration equal to CTS + DATA + ACK). When node $C$ hears a Confirming CTS, it sets its $\text{avail}\_\text{bit}(x) = 0$ and starts $\text{avail}\_\text{timer}(x)$ in the same way. When it hears a Rejecting CTS, no action is needed. Note that this deferring rule is only with respect to channel $x$. Node $C$ can compete for other available channels after deferring for the duration of an RTS/CTS exchange.

There is only one exception to the above deferring rules. When $C$ wants to transmit to $A$ and hears an RTS from $A$, intended to $a$, it will defer until the end of the entire transmission of flow $Aa$, and set its contention window size to the minimum value. Similarly, when $C$ wants to transmit to $a$ and hears a CTS from $a$ intended to $A$, it will defer until the end of the entire transmission of flow $Aa$ and set its contention window size to the minimum value. This scheme provides an opportunity for $C$ to address the Missing Receiver Problem.

\textbf{5.2.3 Addressing Multi-channel Coordination Problems}

To present how AMCP solves the coordination problems described in 5.1.3, we consider the topology in Figure 5.1(a) and suppose there are 2 data channels and 1 control channel.

\textbf{Multi-channel Hidden Terminal Problem.} Consider again the Multi-channel Hidden Terminal Problem example of Fig. 5.4. 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 AMCP, node B sets channel 1 as unavailable and sets a timer to expire after the duration of a RTS/CTS/DATA/ACK transmission (as specified in step 5 of protocol
operations). 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.

However, node B does not necessarily remain idle after the channel 1 timer is set. Since its transmission on channel 2 was successful, this channel is available and B will start contending immediately for this channel (through its prefer variable). Fig. 5.6(a) shows the scenario where B succeeds in capturing channel 2. In case B fails due to another flow Cc that contended for channel 2, it will also set a timer for channel 2 and defer contention until either of the two channel timers expires.

The worst-case scenario for flow Bb upon its arrival on the control channel is depicted in Fig. 5.6(b). Here flow Cc wins channel 2 and then flow Aa wins channel 1 before the channel 1 timer expiration. Although B has lost both contentions, it has synchronized on both data channels and will contend when either of these transmissions ends. Flow Bb has an advantage in capturing either channel in future access attempts: it can compete for both channels, counting down a single back-off counter. On the other hand, each of flows Aa and Cc will only compete for its preferred channel, according to a fresh back-off counter.

Summarizing, the simple waiting scheme of AMCP on the control channel effectively addresses the Multi-channel Hidden Terminal Problem by providing fair channel access opportunities to contending flows.

**Missing Receiver Problem.** Consider the scenario shown in Fig. 5.5, where A wants to transmit to B when B is transmitting to C on a different channel. AMCP 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.
(a) Flow $Bb$ contends and captures channel 2, upon return to the control channel.

(b) Flow $Bb$ synchronizes to contend on both channel 1 and channel 2, upon return to the control channel.

Figure 5.6: AMCP addressing the Multi-channel Hidden Terminal problem.

to zero.

With AMCP, the key reason of $A$ quickly synchronizing with $B$ is that all control messages are transmitted on a dedicated control channel, where $A$ can hear another RTS from $B$ when $A$’s first RTS to $B$ collides with the RTS of $B$ to $C$.

In case node $B$ is the receiver on link $BC$, node $A$ performs the same actions as above when it hears the CTS of $B$ to $C$. Therefore, AMCP effectively addresses both manifestations of the Missing Receiver Problem.
5.3 Protocol Analysis

We now derive the analytical properties of AMCP. In Section 5.3.1 we derive the maximum number of data channels that can be supported by the control channel. In Section 5.3.2 we derive a lower bound on the throughput achieved by any flow in an arbitrary topology.

5.3.1 Bottleneck Analysis

For any multi-channel protocol having a dedicated control channel, two potential bottlenecks exist: the number of data channels and the bandwidth of the control channel. Let $M$ be the maximum number of data channels occupied by data transmissions when the control channel is saturated by control message exchanges. Let $T_d$ be the duration of a successful data transmission (including DATA and ACK), $T_r$ be the duration of an RTS packet, and $T_c$ be the duration of an CTS packet, all expressed as time slots.

We observe that in steady state when the control channel is saturated, there are always $M$ on-going transmissions on $M$ data channels. Furthermore, within the time period of $T_d + T_r + T_c$, exactly $M$ flows return to the control channel. Hence in steady state, $M$ flows should successfully exchange control packets and switch to $M$ respective data channels. Since the control channel is saturated, there is no idle interval between two successive RTS/CTS exchanges, if we neglect small overhead intervals, such as SIFS and DIFS. Thus $M$ is given by,

$$M = \left\lfloor \frac{T_d + T_r + T_c}{T_r + T_c} \right\rfloor. \quad (5.1)$$

From Equation (5.1), we observe that $M$ increases when the data transmission time $T_d$ increases. For example, the control channel can drive more data channels if we reserve a data channel for multiple data packets. Also note that $M$ derived above is for a single contention region. In a multi-hop network, $M$ can be much larger because the control channel is spatially reused.
5.3.2 Lower Bound Analysis

In this section, we compute a lower bound of per-flow throughput achieved by our protocol in an arbitrary multi-hop wireless network. We first construct a hypothetical, low-throughput scenario for a tagged flow, then compute its collision probability \( p \) by modeling the process by which control packets of other nodes arrive on the control channel as a Poisson process. We then apply the analytical model proposed in Chapter 2 to compute the throughput of the tagged flow, which serves as a lower bound on the throughput achieved by any flows in an arbitrary topology.

**Construction of the hypothetical scenario.** We consider a tagged flow \( Aa \) that has \( N \) neighboring nodes in a network employing AMCP. We construct the scenario where flow \( Aa \) achieves very low throughput given \( N \) neighbors as the case that all of its \( N \) neighbors are backlogged and always transmit to receivers that are not in range of \( Aa \) using the minimum back-off window. We also assume that these \( N \) nodes are transmitting independently, in the sense that they do not sense and hence coordinate with each other's transmission. Furthermore, they are placed such that they are advantaged with respect to flow \( Aa \). More specifically, we consider an IA scenario where these \( N \) nodes are within range of receiver \( a \) and out of range of transmitter \( A \). In this scenario, most control packets of flow \( Aa \) will collide, thus forcing flow \( Aa \) to double its contention window. Notice that transmitter \( A \) is not able to sense the activity of the interfering flows. This scenario is hypothetical and only used to derive a lower bound of the throughput of flow \( Aa \) given \( N \) neighboring nodes in its contention region.

Since in this scenario the interfering nodes transmit independently, their control packets arrive on the control channel independently. Consequently, we assume that the aggregate process formed by the control packet arrivals of the \( N \) interfering nodes is Poisson. While this process is not strictly Poisson, we validate the bound via simulations below.

**Computation of the conditional packet loss probability.** To compute the throughput of flow \( Aa \) in the hypothetical, low-throughput scenario, we first need to compute the collision probability \( p \) when node \( A \) attempts to transmit an RTS packet to \( a \). Similar
to [8], we refer to $p$ as the conditional collision probability.

Let $X(t)$ be the Poisson process that represents the number of successful control packet arrivals of the $N$ interfering nodes, given a starting point in time. Let $\alpha$ be the arrival rate of control packets and $T$ be the arrival interval. Note that $\alpha$ is a deterministic value and $T$ is a random variable.

We assume nodes can always find a data channel to transmit a data packet upon successful RTS/CTS exchange. The arrival rate $\alpha$ of $X(t)$ is given by:

$$\alpha = \frac{N}{T_d + T_r + T_c}. \quad (5.2)$$

Since $X(t)$ is a Poisson process, any interval $T$ between two successive control packet exchanges of the interfering flows is exponentially distributed with the following CDF,

$$F_T(t) = P(T \leq t) = 1 - e^{-\alpha t}. \quad (5.3)$$

The RTS/CTS exchange between $A$ and $a$ will fail if it cannot fit within an idle gap $T - (T_r + T_c)$ between two successive control packet exchanges. This corresponds to the event $T - (T_r + T_c) < T_r$ (or $T < 2T_r + T_c$), which occurs with probability $p = F_T(2T_r + T_c)$.

Combining with Equations (5.2) and (5.3), we derive the final expression for the conditional packet loss probability $p$:

$$p = 1 - e^{-(2T_r + T_c)T_d + \frac{N}{T_d + T_r + T_c}}. \quad (5.4)$$

**Throughput computation.** We compute the throughput of the tagged flow $Aa$ using a general model for backlogged flows sharing an 802.11 multi-hop network introduced in Chapter 2. In that model, the channel view of each node comprises of a sequence of time intervals that correspond to 4 different states: (i) idle channel; (ii) channel occupied by successful transmission of the tagged station; (iii) channel occupied by a collision of the station; (iv) busy channel due to activity of other stations, detected by means of either physical or virtual carrier sensing (the NAV). The time intervals during which the station remains in each of the four states above are denoted by $\sigma$, $T_s$, $T_c$, and $T_b$, respectively.

According to the model in [20], the throughput of the tagged flow $Aa$ is given by:

$$T_p = \frac{\tau(1-p)}{\tau(1-p)T_s + \tau p T_c + (1-\tau)(1-b)\sigma + (1-\tau)bT_b}, \quad (5.5)$$
where $\tau$ is the probability that the node attempts to send a packet after an idle slot, $b$ is the probability that the channel becomes busy after an idle slot due to activity of other nodes and $p$ is the conditional packet loss probability.

The probability $\tau$ is a deterministic function of $p$ and is given by [37]:

$$\tau = \frac{2q(1 - p^{m+1})}{q(1 - p^{m+1}) + W_0[1 - p - p(2p)^{m'}(1 + p^{m-m'}q)]}, \quad (5.6)$$

where $q = 1 - 2p$, $W_0$ is the minimum window size, $m$ is the maximum retry limit, and $m'$ is the backoff stage at which the window size reaches its maximum value. The average durations $\bar{T}_s$ and $\bar{T}_c$ are fixed and can be found in [8].

In this hypothetical scenario, the transmitter node $A$ does not defer its transmission due to the activity of other nodes. Setting $b = 0$ in Equation ((5.5)) yields:

$$T_P = \frac{\tau(1 - p)}{\tau(1 - p)\bar{T}_s + \tau p \bar{T}_c + (1 - \tau)\sigma + (1 - \tau)}, \quad (5.7)$$

Using Equations (5.4) and (5.6), in Equation (5.7), we can now compute the throughput of the tagged flow $Aa$ in the hypothetical scenario which serves as a lower bound on the throughput achieved by any flow in an arbitrary topology as a function of number of interfering flows and system parameters.

**Lower bound validation.** We now validate the lower bound with simulations obtained with ns. Both RTS/CTS packets and data packets are transmitted at 2 Mbps. We vary the number of flows $N$ and place them in a $700m \times 700m$ area such that they belong to the same contention region. This means that only one flow can transmit successfully at a time, however it is not necessary that all transmitters or receivers are within range. For each $N$, we generate 10 data points each corresponding to the minimum rate achieved by a different contention region. Fig. 5.7 shows the minimum rates as data points and the lower bound as the analytical curve, as computed by our model.

We observe that in general the minimum rates are greater than the lower bound while in several cases the bound is tight.
Figure 5.7: Comparison of lower bound to minimum throughput achieved in an arbitrary contention region as a function of the number of interfering flows.

5.4 Performance Evaluation

We evaluate AMCP in both single-hop and multi-hop topologies using the ns-2 simulator with CMU wireless extensions. Unless otherwise specified we use the MAC parameters of Table 1. According to these parameters, the maximum rate achieved by a backlogged flow in isolation is 184 pkt/s. The simulator physical layer parameters have been set so that the transmission range of each node is approximately 250m.

We begin with experiments on single-hop topologies to study the main protocol properties and illustrate the interplay between various parameters—number of channels, traffic load, control channel capacity, number of nodes, channel switching delay—that affect performance. Performance is measured in terms of aggregate throughput gain with respect to IEEE 802.11 DCF using a single channel.

We then move to multi-hop topologies, where we demonstrate the properties of AMCP: starvation mitigation, increase of aggregate utilization and addressing the fundamental coordination problems of both single channel and multi-channel systems, as elaborated in section 5.1. We also compare AMCP with MMAC, a single-radio, multi-channel protocol
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>SIFS</td>
<td>10 μs</td>
</tr>
<tr>
<td>DIFS</td>
<td>50 μs</td>
</tr>
<tr>
<td>EIFS</td>
<td>364 μs</td>
</tr>
<tr>
<td>σ</td>
<td>20 μs</td>
</tr>
<tr>
<td>BasicRate</td>
<td>2 Mbps</td>
</tr>
<tr>
<td>DataRate</td>
<td>2 Mbps</td>
</tr>
<tr>
<td>PLCP length</td>
<td>192 bits @ 1 Mbps</td>
</tr>
<tr>
<td>MAC header (RTS,CTS,ACK,DATA)</td>
<td>(20,14,14,28) bytes @ BasicRate</td>
</tr>
<tr>
<td>Packet size</td>
<td>1000 bytes</td>
</tr>
<tr>
<td>(CW_{min}, CW_{max})</td>
<td>(31,1023)</td>
</tr>
<tr>
<td>Retry Limit (Short,Long)</td>
<td>(7,4)</td>
</tr>
<tr>
<td>Channel switching delay</td>
<td>224μs</td>
</tr>
<tr>
<td>MMAC ATIM window</td>
<td>20ms</td>
</tr>
<tr>
<td>MMAC Beacon interval</td>
<td>100ms</td>
</tr>
</tbody>
</table>

Table 5.1: MAC layer parameters

proposed in [48]. MMAC uses a globally synchronized control/data periodic frame (termed beacon interval). During the control subframe (termed ATIM window) flows contend on a default channel to reserve channels (including the default channel) for the data subframe. The flows that succeed in reserving a channel during the ATIM window contend during the data subframe using RTS/CTS 802.11 access mechanism. Our experiments use the same MMAC parameters as [48] (Table 5.4).

5.4.1 Single-hop Topologies

In this series of experiments all nodes are within range of each other and are equally divided in a transmitter and receiver set. This yields a set of single-hop disjoint flows with distinct transmitter-receiver pairs. The case where a node is both sender and receiver is considered in the multi-hop experiments.

**Effect of number of channels.** Fig. 5.8 depicts the aggregate throughput achieved by AMCP as a function of the total number of channels for 15 backlogged flows (30 nodes). The capacity of the control channel and each data channel is 2 Mpbs. The case of AMCP with 2 channels is equivalent to single-channel 802.11, which provides the reference line
in Fig. 5.8. The aggregate throughput increases linearly until 7 channels. After that point, it increases with a slower rate with additional channels; at 8 channels it reaches the limit of 1100 pkt/s where the control channel is saturated. This behavior agrees with our bottleneck analysis: for the parameters in this experiment, Equation (5.1) predicts that the control channel can drive up to 8 data channels.

![Throughput Graph](image)

Figure 5.8: Throughput as a function of number of channels.

**Effect of traffic load.** We evaluate the performance of AMCP under non-backlogged conditions. Fig. 5.9 depicts the aggregate throughput of AMCP and IEEE 802.11 in a 15-flow topology as the input rate of each flow increases, when a total of 4 channels are used. Until 10 packets/s, the load is too low to exploit the additional data channels and AMCP yields similar performance to 802.11. After that point, channelization becomes effective and AMCP reaches an aggregate throughput gain equal to the number of data channels.

We note that existing multi-channel MAC protocols can achieve similar or slightly higher aggregate throughput than AMCP. For example, for 4 channels and under heavy load, DCA [53] also achieves three times the aggregate throughput of 802.11, similar to AMCP. This is because both AMCP and DCA dedicate a separate channel for control traffic. On the other hand, MMAC transmits control and data packets over 4 channels and
achieves an additional gain of 20%-30%. However, DCA requires two radio transceivers per node and MMAC requires global synchronization. AMCP uses a single transceiver and no global synchronization.

![Graph showing throughput vs. packet arrival rate](image)

Figure 5.9: Aggregate throughput when arrival rate varies.

**Effect of channel switching delay.** Since AMCP switches channels at the packet level, channel switching delay due to hardware limitations can be a source of overhead. According to the IEEE 802.11 specification [22] this parameter can reach 224\(\mu\)s. Fig. 5.10 shows a graceful decrease of aggregate throughput as channel switching delay increases from 0 to 5ms. At 224\(\mu\)s, the throughput decrease is very small. This can be explained by the fact that 224 \(\mu\)s is small compared to the duration of a data transmission. After 3ms, throughput goes below the single-channel maximum throughput of 184 pkt/s. For hardware with such high channel switching delays, the overhead can be addressed by reserving a channel for multiple data packets. Such functionality is easy to incorporate in the AMCP channel reservation mechanism.
5.4.2 Multi-hop Topologies

In this series of experiments we compare the performance of AMCP, MMAC, and single-channel 802.11 in static and mobile multi-hop topologies using both single-hop and multi-hop flows. We also consider specific scenarios that isolate inefficiencies that arise in the design of multi-channel protocols, namely the random channel selection problem due to collisions of control packets and the head-of-line (HOL) problem due to lack of packets to fill a channel reservation window.

Single-hop Flows

**Single-channel unfairness scenarios.** We first investigate the ability of AMCP and MMAC to address the IA and FIM coordination problems (Fig. 5.1 and Fig. 5.2, respectively). These scenarios can easily be addressed by random channel selection if a large number of channels are available. Here we consider the case when a total of three channels are available.

We observe from Fig. 5.11(a) that AMCP provides equal and maximum throughput
to each flow, despite that, topologically, flow $Bb$ has more information about the channel. Furthermore, the simulation shows that the two flows persist transmitting on different channels. This is a desired property and shows that AMCP successfully separates the two flows and reduces their interaction.

Under MMAC, flow $Bb$ achieves 80% the maximum throughput of 802.11 and AMCP. This is the maximum throughput allowed by MMAC since the ATIM window is 20% of the beacon period. However, the key observation is that the disadvantaged flow $Aa$ receives only 2/3 of the maximum MMAC throughput. This is because the IA problem still exists in both the control subframe and the data subframe: the ATIM packet size is comparable to the back-off window size;* since its control packets collide, the transmitter of flow $Aa$ is not informed about channel reservations in its neighborhood and is forced to perform random channel selection. $Aa$ may choose the same data channel as $Bb$ and, consequently, its data packets may be destroyed due to the IA problem.

Similarly, in Fig. 5.11(b) AMCP provides equal and maximum throughput to all flows. As in the IA scenario, the flows quickly coordinate and keep transmitting on the right channels: flow $Aa$ and flow $Cc$ on one data channel and the middle flow $Bb$ on the other. In contrast, MMAC does not equalize the throughputs but is again subject to random channel selection: in this case, the transmitter of the middle flow is not able to decode the colliding ATIM control packets of the outer flows.

We now explain how AMCP addresses the IA and FIM problems. In the IA scenario, flow $Bb$ does not experience collisions and will persist transmitting on one of the two data channels (e.g. channel 1). The receiver of flow $Aa$ will be informed about this decision through the control packets of flow $Bb$. The transmitter $A$ of flow $Aa$ starts without any knowledge of which channel to use. Since contention in the control channel has been reduced by the removal of data packets, it is easier for $A$ to access the receiver. In case it picked channel 1, $A$ will be informed by the receiver and will compete and acquire

---

*To allow more ATIM control packet exchanges in the 20 ms ATIM window, the back-off window size cannot be set too small.
Figure 5.11: AMCP performance in the basic single-channel starvation scenarios.

channel 2 in its next access attempt. After that point both flows will continue transmitting on different channels.

In the FIM scenario flows do not experience collisions and therefore prefer to transmit on the same channel. Since the outer flows are not within range there may be an undesirable situation where they have preference for channels 1 and 2, respectively. In this case the middle flow $Bb$ is blocked but only temporarily, until its channel timers expire. It will then contend on the control channel for any of the two data channels. When it acquires any of the two channels (e.g. channel 1) the outer flows are informed and will compete for channel 2. From this point on, since the flows do not experience collisions they will continue transmitting on orthogonal channels.

**Arbitrary topology / single-hop flows.** We now consider an arbitrary topology of 100 nodes placed in a $2000m \times 2000m$ area. The nodes are arbitrarily divided in 50 disjoint single-hop flows. Fig. 5.12 depicts the per-flow throughput under AMCP, MMAC, 802.11 as well as the AMCP lower bound, all with respect to the AMCP throughput sorted in decreasing order.

As expected, both AMCP and MMAC achieve higher aggregate throughput than 802.11.
Figure 5.12: Per-flow throughput in an arbitrary topology of single-hop backlogged flows using 12 channels.

Furthermore, under 802.11 16 out of 50 flows receive close to zero throughput. The aggregate log utility of this network with IEEE 802.11, MMAC and AMCP is -90.9, -3.7, and 13.2 respectively. While determining the utility achieved by optimal proportional fairness rate is beyond the scope of this thesis, by assuming each flow receives the maximum capacity, we compute the upper bound of the network utility to be 34.65.

We observe that AMCP achieves higher per-flow throughput than MMAC and 802.11. Under AMCP, all flows receive above 105 pkt/s and each flow receives higher throughput than its predicted lower bound. Under MMAC all flows receive throughput above 75 pkt/s, yet always lower than AMCP; furthermore, 27 out of 50 flows receive throughput below the corresponding AMCP lower bound. Part of this inefficiency is due to the 20% ATIM window overhead; however, the flows with much lower throughput indicate that the random channel selection problem can be a source of inefficiency even if several channels (12 in this case) are available in the system.
Multi-hop flows

Next, we move to more sophisticated scenarios involving multi-hop flows. Multi-hop flows induce non-disjoint single-hop flows which include the missing receiver problem and the head-of-line (HOL) problem in addition to the problems we have experimented so far. We first consider a scenario that isolates and illustrates these two additional problems. Finally, we consider an arbitrary scenario where all the problems are present and also evaluate the effect of mobility.

**Download scenario.** In the static 20-node topology of Fig. 5.13(a), a designated gateway node sends traffic to all other nodes through a tree structure. In this download scenario multiple channels are of little help because the bottleneck is the radio constraint at the root node. The maximum per-flow fair rate is $184 / 19 = 9.68 \text{ pkt/s}$.

The per-flow throughputs under backlogged conditions are shown in Fig. 5.13(b). Two key observations are in place. First, AMCP delivers close to maximum per-flow throughput in a scenario where the missing receiver problem is strongly present. With IEEE 802.11,
AMCP, and MMAC the network log utility is -56.2 -74.3 -54.5. Note that the log utility of this network is bounded by -32.4.

Second, MMAC delivers substantially lower throughput than both AMCP and 802.11. This is not due to the missing receiver problem because MMAC uses synchronized contention. It is also not due to the random channel selection problem because the number of channels is not the bottleneck in this scenario. The problem arises because each node intends packets to multiple outgoing neighbors. During the 20ms control subframe, each node contends for the link corresponding to the HOL packet in its queue. Upon success, for the next 80ms-data subframe it will contend and transmit in the reserved channel only for this link. Hence, the data subframe can be fully utilized only if a sufficiently high number of packets of this link immediately follow the HOL packet. Unfortunately this is not likely to happen if this node intends packets to multiple neighbors and is the source of inefficiency in this scenario.

There appears to be no easy solution to the HOL problem. On one hand a node could be allowed to reserve a channel for multiple links during the data subframe. This would require both significant changes to the MAC protocol as well as sophisticated queue management that would increase protocol complexity. On the other hand, the data subframe can be reduced to fit packet transmissions of a single link. However this increases the overhead due to the control subframe. Optimal sizing of the global control/data subframe is hard to perform without a-priori knowledge of traffic requirements. In addition, no sizing would suit all nodes in the network.

The HOL problem is not specific to MMAC. It exists in any multi-channel protocol that attempts to reserve a channel for several packet transmissions (e.g. SSCH [4]). If not addressed properly, it can produce substantial overhead that counter-balances the gain due to multiple channels. On the other hand, the HOL problem is not present in AMCP because contention occurs on a per-packet basis.

**Multi-hop flows and mobility.** To study mobility and the joint effects of the above factors, we consider a mobile scenario of 50 nodes in a 1000m × 1000m area and form
10 multi-hop flows with arbitrary source-destination pairs. We use the random waypoint mobility model where nodes move at 1 m/s.

![Graph showing throughput comparison](image)

Figure 5.14: Per-flow throughput in an arbitrary topology of multi-hop flows using 12 channels.

To test MAC protocol performance we need to operate at relatively high loads. Under such conditions, a dynamic MANET routing protocol can cause frequent route changes due to lost routing packets, which in turn can have a dominating degrading effect in overall performance. To decouple the effect of routing, we pre-compute shortest path routes based on the initial topology and keep the routes fixed during each run. We then consider only the experiments where no route breakage occurred. In this way, we can test how the MAC protocols react to mobility viewed as changes of the network contention regions.

Fig. 5.14 shows the achieved throughputs under a per-flow UDP load of 30 pkt/s. Each data point is the average of 10 mobility scenarios. The aggregate log utility achieved with IEEE 802.11, MMAC and AMCP is -24.2, -21.05, and -15.3 respectively. AMCP appears robust in terms of delivery ratio since each flow achieves throughput close to 30 pkt/s. In contrast, several flows receive much lower throughput under MMAC and 802.11. Overall, MMAC outperforms 802.11. However, in 7 out of the 10 flows it receives substantially lower throughput than AMCP and flows 6 and 7 receive very low throughput similar to
802.11. This inefficiency is due to the superposition of ATIM window overhead, the random channel selection problem and the HOL problem.

5.5 Related work

Distributed contention-based CSMA multi-channel protocols have been proposed in [2, 28, 41, 48, 53]. The works in [41] and [28] assume nodes can receive packets on all channels simultaneously. Current hardware does not support listening on an arbitrary number of channels. The DCA protocol [53] uses a separate transceiver for the control channel. This is an expensive solution considering that control traffic is much lower than data traffic. The additional transceiver could instead be used to double network data throughput. Indeed, MUP [2] uses two transceivers each assigned to a different channel and executing a separate instance of IEEE 802.11 DCF; Packets are scheduled on the transceiver (and channel) that experience the least contention. It is evident that MUP would experience starvation within each channel much like a single-radio / single-channel IEEE 802.11 DCF system.

MMAC [48] uses a single transceiver and resolves the access problem using a synchronized control/data periodic frame. Since it uses globally synchronized contention, MMAC avoids the Multi-channel Hidden Terminal and Missing Receiver problems. It is also able to use more channels than solutions that use a dedicated control channel like AMCP. However, it inherits all the problems of scheduled access with a synchronized control/data frame structure. Furthermore, as demonstrated in Section 5.4, its performance can be severely degraded due to the random channel selection and HOL problems.

SSCH [4] is a single-transceiver, multi-channel protocol. Each node hops between channels using a 13-hop pseudo-random sequence. Within a channel hop duration a node uses IEEE 802.11 DCF to transmit data or control packets (which advertise its channel hopping schedule) to its neighbors. The channel hopping sequences have been designed such that any two nodes will overlap in at least one of the 13 hops. Each node uses the channel hopping schedules of its neighbors to transmit to them, by tuning the corresponding hops in its own hopping schedule. SSCH follows the first general solution approach we
presented in Section 5.1. It avoids the control channel bottleneck by distributing both control and data packets to different channels. However, since it reserves a channel for multiple data transmissions on the same link it is subject to the HOL problem. Furthermore, since every node may decide to change its hopping schedule to transmit to others, the Missing Receiver problem can be extremely severe. In the three-node example of Fig. 5.5, even if node $A$ knows the hopping schedule of $B$, its access attempt on a certain hop may fail when $B$ has tuned to the hopping schedule of $C$. This will result in several RTS failures of $A$ during the 10ms duration of this hop. This problem becomes more severe when $B$ has several outgoing links. In AMCP, node $B$ will coordinate with node $A$ through the control channel, irrespective of the number of its outgoing links.

Regardless of assumptions on hardware or infrastructure support, all protocols in [2, 4, 28, 41, 48, 53] have focused on increasing aggregate network throughput and do not provide any form of analytical per-flow throughput guarantees.

5.6 Conclusion

We have presented AMCP, a distributed medium access protocol that utilizes multiple channels to improve unfairness in a multi-hop wireless network. AMCP is simple, yet effectively alleviate the unfairness problem that would occur with a single channel and further addresses coordination problems inherent in a multi-channel system.

Using a simple analytical model we have shown that AMCP provides a lower throughput bound for a flow within any contention region of an arbitrary topology. Our experiments for arbitrary topologies have shown that AMCP utilizes multiple channels to yield much higher per-flow throughput than the lower bound and a significant aggregate throughput gain with respect to a single-channel system. Its benefits are achieved using minimal amount of resources: only a single half-duplex radio transceiver and no infrastructure support such as global time synchronization or channel pre-distribution mechanisms.
Chapter 6

Conclusions

In this thesis, we have analyzed the origins of unfairness in multi-hop wireless networks that employ IEEE 802.11 as the medium access control protocol. We focused on two flows, the least number of flows that allow us to study the inter-flow contention. We isolated and studied the MAC effect on unfair contention, and then investigated the compounding effect of MAC and Congestion Control on unfairness. Based on our analysis, we have explored the solution space to improve fairness.

Under IEEE 802.11, flows can contend unfairly in multi-hop wireless networks. Using analysis of two-flow topologies, we demonstrate that if not all nodes are within transmission range of each other, they may have incomplete channel information. As a result, their mutual contention may lead to some flows receiving much higher throughput than others.

Specifically, we have systematically and comprehensively investigated two flows contending with each other in a multi-hop topology. We identified all possible topologies consisting of four nodes and two flows, classified them into three geometric groups, and computed their likelihood under random node placement. In each case, we showed how fundamental properties of two- and four-way handshake IEEE 802.11 protocols yield short-term unfairness in one group, and long-term unfairness in another. We developed analytical models that characterizes the performance of IEEE 802.11 protocol in each of each of the scenario where either short-term or long-term unfairness arises.

Using a mesh network scenario where a one-hop TCP flow contends with a two-hop TCP flow for gateway access, we analyzed how the IEEE 802.11 protocol and the congestion control mechanism jointly induce unfairness in the network. We described how originating factors that lead to unfairness stem from interaction between the transport layer's
congestion control and the MAC layer's collision avoidance. We analytically modeled the system with a six-dimensional Markov chain model that captures the MAC's collision avoidance, the sliding window congestion control, and all end and intermediate queues of the network. The model further motivated us to devise a Contention Window Policy that requires the nodes one-hop away from the gateway increase their minimum contention window. We finally implement and empirically validate the solution on our test-bed and by simulations.

We have presented AMCP, a distributed medium access protocol that utilizes multiple channels to improve fairness in a multi-hop wireless network. AMCP is simple, yet effectively addresses the root causes of unfairness and prevents further coordination problems inherent in a multi-channel system. AMCP resolves the above problems by asynchronously coordinating transmissions over a dedicated control channel.

Using a simple analytical model we have shown that AMCP provides a lower throughput bound for a flow within any contention region of an arbitrary topology. Our experiments for arbitrary topologies have shown that AMCP utilizes multiple channels to yield much higher per-flow throughput than the lower bound and a significant aggregate throughput gain with respect to a single-channel system. Its benefits are achieved using minimal amount of resources: only a single half-duplex radio transceiver and no infrastructure support such as global time synchronization or channel pre-distribution mechanisms.
Bibliography


