Maximizing Performance in Long Distance Wireless Networks for Developing Regions

Sergiu Nedevschi

Electrical Engineering and Computer Sciences
University of California at Berkeley

Technical Report No. UCB/EECS-2009-27
http://www.eecs.berkeley.edu/Pubs/TechRpts/2009/EECS-2009-27.html

February 10, 2009

Maximizing Performance in Long Distance Wireless Networks for Developing Regions

by

Sergiu Nedevschi

Dipl.Eng.(Technical University of Cluj-Napoca, Romania) 2002
M.S. (University of California at Berkeley) 2005

A dissertation submitted in partial satisfaction of the
requirements for the degree of
Doctor of Philosophy

in

Computer Science

in the

GRADUATE DIVISION
of the
UNIVERSITY OF CALIFORNIA, BERKELEY

Committee in charge:
Professor Eric A. Brewer, Chair
Professor Ion Stoica
Professor John Chuang

Fall 2008

The dissertation of Sergiu Nedevschi is approved:

Chair Date

Date

Date

University of California, Berkeley

Fall 2008

Maximizing Performance in Long Distance Wireless Networks for Developing Regions

Abstract

Maximizing Performance in Long Distance Wireless Networks for Developing Regions

by

Sergiu Nedevschi

Doctor of Philosophy in Computer Science

University of California, Berkeley

Professor Eric A. Brewer, Chair

Today we are witnessing a large disparity between the levels of network connectivity in industrialized countries, and the ones in the developing world. This *digital divide* is caused by the uneven distribution of wealth around the world, and has the effect of reinforcing this polarization, by providing increased economic opportunities to people that already afford access to information technology. This divide is partially addressed by the growth of wireless and cellular technologies, but these technologies remain financially unviable in rural regions, with sparse and financially-constrained users.

To address rural and remote network coverage, we propose the use of multi-hop wireless networks relying on long-distance point-to-point links. By using inexpensive, off-the-shelf Wi-Fi radios, and connecting them to high-gain directional antennas, we can build inexpensive, high-throughput links exceeding tens or even hundreds of kilometers in length.

In theory, these wireless long-distance (WiLD) networks have the potential to deliver low-cost connectivity to remote areas. Unfortunately, the performance achieved using the standard 802.11 MAC in long links is very poor, with high and asymmetric packet loss rates, and with low throughput over wireless paths spanning multiple hops.

In this dissertation, we understand the causes for low performance in these scenarios, and build MAC- and PHY-layer mechanisms that address these problems and maximize end-to-end network performance. Using extensive measurements we identify the primary sources of performance degradation. To deal with these problems we design and build WiLDNet, a system that includes a spatial-reuse TDMA MAC and a combination of FEC and ARQ-based link-layer loss recovery mechanisms. We deploy WiLDNet in real-world networks, and show that it eliminates most packet losses and increases link utilization, delivering good end-to-end UDP and TCP throughput. We incorporate the lessons learned from our deployments in the design of JazzyMAC, a MAC that maximizes network-wide throughput and minimizes packet delay by using variable-length transmission slots which change dynamically according to traffic.

We demonstrate the appropriateness of our solutions by deploying them in several networks in developing countries, including the Aravind Eye Hospital network in India that uses our technology to provide telemedicine services to many thousands of patients.

Professor Eric A. Brewer
Dissertation Committee Chair

To my wonderful parents.

Contents

List of Figures

List of Tables

Acknowledgments

I am grateful to my professors, advisors, collaborators and friends that made my years in Berkeley an amazing experience. I was very fortunate to have Eric Brewer as an advisor and role model. He taught me a lot about computers, but much more about what it means to be a good and successful human being. He showed me that the best way to live a successful life is to look around carefully, to find the real problems of the world, to understand how to best use our abilities and talents in order to make a difference, and to follow that vision with tenacity and dedication. He opened my eyes to the idea of using technology to help the less fortunate around the world, and he led me through my years in Berkeley every step of the way.

Collaborators: I want to thank Rabin Patra, a great friend, colleague and roommate, who marked every aspect of my life in Berkeley. We became friends on the first day we both came to Berkeley, and we remained friends for the entire time. We worked together on almost all of our papers, class projects, and TIER projects. We prepared and took the prelim together, we found an advisor together, and we were office mates and house-mates for most of our stay here. I learned a lot from him, both in terms of knowledge and moral fiber. I wish I could be as selfless and helpful as he is to everyone, and I wish I knew as much as he knows about everything. Whenever I have a Computer Science (or general) question I go to him. He usually has the answer, and if he doesn't have it, I can be assured that we'll discuss it until we find one. But I'm most grateful for all the fun we had together: we traveled to India together (twice), we jumped out of the same plane, we climbed Alaskan glaciers, we windsurfed, skied, snow-mobiled, hiked, rock-climbed, bodybuilt, drank and partied. Joyojeet Pal was also part of many of these, especially the latter few. He is the first of the three musketeers: Joyojeet, Rabin and Sergiu (or the third in the Indian version of the musketeers: Lal, Bal and Pal). He's the first because he looked after Rabin and I, just like a big brother would. I never had a big brother and I can't imagine a better one than Joyojeet – someone you can have crazy fun with, travel the world with, but most importantly someone to trust for advice whenever in trouble (in my case mostly girls-related).

I am thankful to have met Melissa Ho, an amazing friend and collaborator. She helped me countless times, we traveled together, and we had a great time talking about every possible subject, many times an existential or philosophical one. She's the reason that I often ask myself questions I forgot to ask, and I'm grateful for that.

Lakshminarayanan Subramanian is a dear friend (and mini-advisor). We also jumped from the same plane (to celebrate his PhD degree). His most important contributions to my life were four-fold: First, he taught me that self-confidence is the most important ingredient in being successful; second, that thinking of how to pitch a project or a paper is sometimes as important as the project itself; third, that you cannot be successful unless you are hungry for it; and fourth, that enumerating your contributions in the introduction of a paper submission is essential in making it simpler for the reviewer – she will like it.

Sonesh is an awesome friend and colleague. He made our dream of making a social impact come true, by managing the deployment of our technology in the Aravind Eye Hospitals in India. But we also had our share of shared adventures: we traveled together to India and Romania, we floated (no need to swim) in salt-saturated water, we climbed mount Shasta (with Anmol and Atul), we traveled half a country measuring wireless coverage from a Jeep, and we took pictures of sheep in the Carpathians.

I was honored to be part of the TIER group; after being part of it, I can honestly say I understand much better the meaning of the words “collaborate”. I am happy that I had the chance to work (and play) with Anmol Sheth, Divya Ramachandran, Yahel Ben-David, Mike Rosenblum, Mike Demmer, Bowei Du, RJ Honicky, Omar Bakr, Renee Kuriyan, Jennifer Bussell, Madelaine Plauche, Sushant Jain and all the rest.

My days (and nights, given my nocturnal schedule) in the Intel Research Lab were a lot of fun, and I am grateful to all the researchers and interns that I had the privilege to work with. I was very lucky to have Sylvia Ratnasamy as a mentor here, she is a truly extraordinary person, one of the smartest people I know, but also one of the kindest and most supportive. Working with her helped me mature as a researcher, gave me confidence, and inspired me to explore new interesting areas, for which I am deeply grateful. I also want to thank Gianluca Iannaccone, Kevin Fall, Nina Taft, Rob Ennals, Tyson Condie, Mazier Manesh, Atul Singh, Alan Mainwaring, Amrita Lokre, Allan Knies and all the other people at Intel Research for their help and support.

In the last year I had the privilege to work with Lucian Popa, who is a smart and dedicated collaborator, a dear friend, and an awesome skiing partner. He inspired me to think deeper and work harder, and helped me tremendously in my research of energy efficient networks.

In my first couple of years in Berkeley I had a great time studying Computer Architecture in Soda Hall, and I want to thank Yatish Patel, Sonesh Surana, Mark Whitney, Nemanja Isailovic, Sukun Kim, Norm Zhou and Yury Markovskiy for being there to guide me and help me integrate. I owe a lot to La Shana, who was there to advise and support me through all the requirements of my degree.

Friends: My days in Berkeley involved a lot of work, but also a lot of partying, traveling and fun, and I owe it all to my friends, that made me feel loved and appreciated. I am especially lucky to date Roycie, a smart, wild and amazing girl, who made every bad day a little better, and every good day an awesome one. My Romanian brothers Andi, Mihai, Vali, Bezea, Oana, Nick, and my I-House friends were with me for crazy parties, chill beer nights and spring breaks in Jamaica, Costa Rica and Mexico; I doubt anyone could wish for better days. I want to thank all of the I-House party crew for teaching me the value of living in a diverse world, and my Romanian friends for keeping me close and proud of my roots.

Family: This entire experience of graduate school would have not been possible without the support of my family. I can't imagine better parents than mine, Anca and Sergiu. They are my role models, my best friends, my biggest fans and my most dedicated supporters. Their unconditional love made me the man that I am, and I owe everything to them. I was also one of the lucky children to be the "pride and joy" of four wonderful grandparents, who played an essential role in my upbringing. I only wish they could be here to see me graduate. I also want to thank my friends and relatives from Romania for staying close even when far away, returning home to see them every winter recharged my batteries, and gave me renewed motivation.

Chapter 1

Introduction

It is widely accepted that technology is, and has historically been, one of the greatest enablers for an improved quality of life. Information and communication technology (ICT) in particular has been essential in increasing the quality and accessibility of healthcare and sanitation, education, governance and many other areas essential to social and human development.

Among these technologies, the Internet has played a central role in bringing about today's *digital revolution*, by allowing individuals and businesses to collaborate with others across town or globe, enabling them to buy and sell things, to share and learn information, and in general to *communicate* with each other in ways and at scales never possible before. This new and improved means to accessing information proved invaluable, empowering its users and bringing a wealth of opportunity for businesses, educational institutions and scholarly research alike.

Unfortunately, most of the gains of the digital revolution have been restricted to the industrialized countries, where the economic and technical means necessary to buy and operate computers and telecommunication infrastructure were available. But because computers and network access are empowering, providing large economic opportunities to their users, not having access to these resources becomes a great competitive disadvantage, further widening the economic gap between people and regions can that afford access to these resources in the first place and the ones that can not.

This phenomenon, called *the Digital Divide*, can be used to reflect the divide within one country or population, or to denote the global division between industrialized countries and developing countries. The phenomenon has been widely recognized by international organizations [117], governments [16, 25], corporations [108], NGOs [122] and academic researchers [37] as an important issue that needs to be addressed. To this end, in the year 2000, the United Nations adopted the following goal as one of its Millennium Development Goals [118]: "Make available the benefits of new technologies – especially information and communication technologies". To the same end, governments, NGOs, researchers and private companies have started a large number of development projects seeking to bridge the digital divide by building and deploying information and communication technologies for development(ICTD) [37].

This dissertation is focused specifically on the problem of building networking technology that enables the easy deployment of inexpensive communications infrastructure in developing regions.

The work proposes a novel, inexpensive and high-performance networking solution for delivering rural network connectivity. The solution relies on deploying multi-hop wireless networks built using long-distance point-to-point wireless links. Each of these links can be tens or even hundreds of kilometers in length, and costs as little as US$1000. The links are built using inexpensive commodity hardware (Wi-Fi radios, directional antennas and wireless routers), and run novel medium-access and link-layer network protocols, tailored to deliver superior performance in long-distance, multi-hop operation.

The solutions proposed in this work are designed and implemented jointly with other members of the TIER research group, and they were deployed in many rural and urban network deployments all around the world. Our technology is currently used by the Aravind [23] hospitals and rural health centers in India, delivering high-quality telemedicine services to tens of thousands of users. The same solution was also used to demonstrate the feasibility of an unamplified, 382km wireless link [50] in Venezuela, delivering 6Mbps bidirectional throughput, and relying only on commodity hardware.

1.1 Motivation

In the following we present our motivation for researching inexpensive communication technologies for rural network infrastructure in developing regions.

1.1.1 The need: Today's Communications Divide

The Digital Divide is manifested in the inequal access to computing devices, but also in the inequal penetration of telecommunication and Internet infrastructures. To understand the latter, let us examine the worldwide ICT penetration statistics for the year 2007, as provided by the International Telecommunication Union [9].

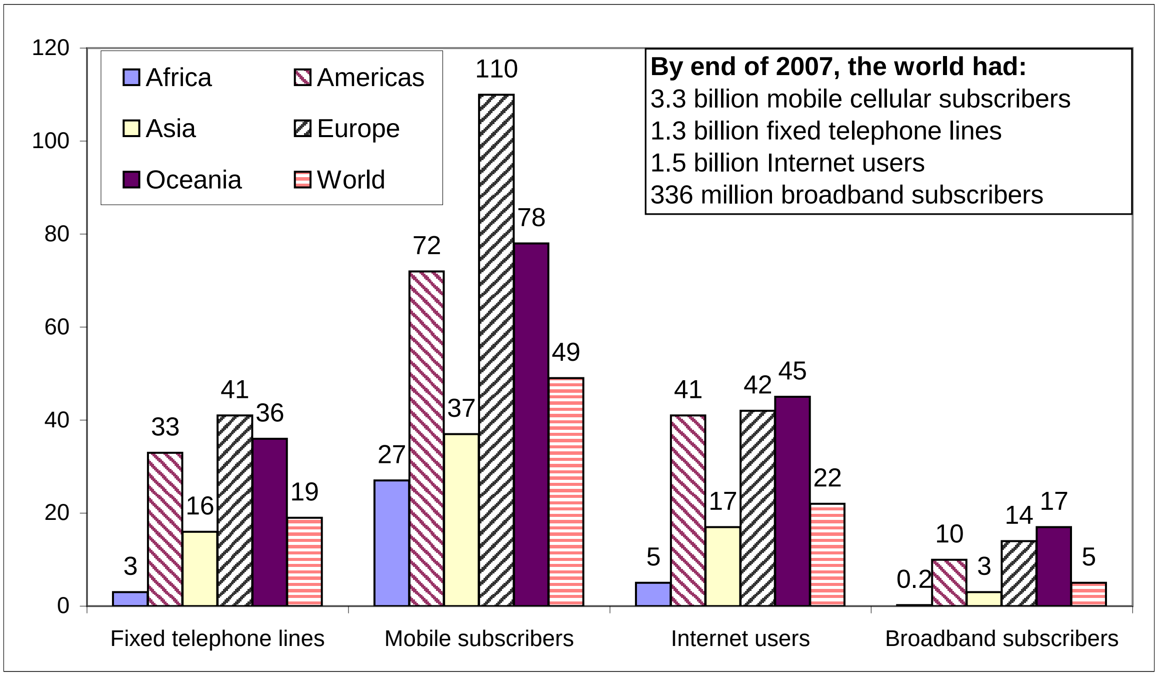


Figure 1.1: ICT penetration by continent in 2007 (per 100 inhabitants)

We observe a very large disparity between telephone and Internet penetration across continents, with only 3 fixed telephone lines and 5 Internet users per 100 inhabitants in Africa, in comparison with a much more substantial penetration in Europe and the Americas. However, looking at continent granularity data hides the fact that this penetration is even lower in some countries (*e.g.,* essentially no Internet penetration in Liberia, below 1% penetration in 15 African countries and 7 Asian countries, including large ones like Bangladesh [9]). Moreover, most of the ICT penetration in developing countries happens in urban centers, with a very large fraction of the rural areas remaining unserved by any networking infrastructure.

Some of this divide will be naturally addressed by market forces and the rise of wireless networks and mobile telephony, which has seen a very quick growth developing

World Regions	Population (2008)	% Population (Penetration)	Usage % of World	Usage Growth 2000-2008
Africa	955,206,348	5.3%	3.6%	1,032.2%
Asia	3,776,181,949	14.0 %	37.5 %	363.4 %
Europe	800,401,065	48.0 %	27.2 %	265.7 %
Middle East	197,090,443	21.3 %	3.0 %	1,176.8 %
North America	337,167,248	73.4 %	17.5 %	129.1 %
Latin America/Caribbean	576,091,673	23.8 %	9.7 %	659.3 %
Oceania/Australia	33,981,562	59.5 %	1.4 %	165.1 %
WORLD TOTAL	6,676,120,288	21.2 %	100.0 %	291.3 %

Table 1.1: Worldwide Internet Usage Statistics

countries. This trend is illustrated in Table 1.1, that shows worldwide Internet usage statistics in 2008 [10]. We see that regions with small ICT penetration (Africa, Asia, the Middle East) have seen much higher growth rates in the last 8 years than European and American markets, which are much closer to saturation.

Although this tendency is promising, it is also somewhat deceiving, because it reflects the increase in ICT penetrations seen almost exclusively in urban areas, not in rural regions. For example, in India the mobile phone penetration in the year 2006 was 33 percent in urban areas, as opposed to only 2 percent in the rural areas. Even though the majority of the developing world population lives in rural areas – 74 percent in India [56] – most rural areas remain uncovered.

Therefore, extending network coverage to rural regions of the world remains a big challenge due to a combination of limited purchasing power and low density of rural users. Since 70 percent of the capital cost of typical networks is in the access network as opposed to the backbone, these networks depend on a certain user density for profitability. Hence urban areas tend be covered by multiple carriers, while rural areas are typically covered by a single carrier or none at all.

For example, even the best known "rural" cellular system, Grameen Telecom [57] in Bangladesh, avoids rural-only base stations. Instead, by exploiting the high population density of Bangladesh, Grameen places base stations such that they cover both urban, higher income users as well as lower income users in the rural areas; typically, there is no coverage for rural areas that are not near an urban base station [78].

1.1.2 The opportunities

Although the need for universal network connectivity is more pressing than ever before, the opportunity for providing such connectivity is also at its peak. There are several things that make this the right time to attempt such universal connectivity: the impact of *Moore's Law*, the emergence of a *shared model of computing*, the *rise of wireless communication*, and a more supportive business environment.

The impact of Moore's law has brought the cost of computing down to fractions of a cent per user for web sites [37], making the costs of shared infrastructure very small. The combined impact of Moore's Law and a cost-efficient design, targeting users in the developing world, has also resulted in the production of low-cost personal devices. Initiatives such as the MIT-initiated "one laptop per child" [14] and Intel's competing Classmate [62] drove the cost of low-end laptops to figures approaching US$100.

In parallel with the reduction of device and infrastructure costs, another decisive factor in decreasing ICT cost per user is the emergence of a model of computing in which devices are shared among a large number of potential users. The shared model of computing has been adopted by a multitude of telecenter projects [72, 82, 90, 123] all around the world, and by projects leveraging the use of shared cellphones such as Grameen Village Phone [39, 101, 124]. Refurbished or new computers are also shared in schools [27] and hospitals [112]. To make sharing more efficient, inexpensive solutions introducing multiple inexpensive thin clients per server have been developed, such as HP's 441 systems that supports 4 keyboards and 4 screens connected to a single Linux-based machine, and NComputing's X300 project that uses low cost terminals connected over Ethernet with a single PC. Solutions proposing the use of multiple mice per computer to stimulate in-classroom collaboration and learning using shared computers have also been proposed [83, 85, 87], and were proven to significantly improve the learning process.

Second, the high-volume production of wireless communications, particularly cellular and Wi-Fi, has brought its cost down as well. For example, the majority of villages in Bangladesh, almost entirely without telephony a decade ago, now have shared cellular phones. For rural areas, wireless infrastructure appears to be the first kind of infrastructure that is affordable. We believe that successful wireless infrastructure may lead to sufficient increases in rural incomes to make other infrastructure investments viable, such as water and power distribution.

Moreover, there now exists a supportive business environment for ICT projects. The diffusion of technology worldwide and the growing access to capital have created a favorable environment for entrepreneurship and experimentation. This environment, combined with the success of franchising as a way to deploy large-scale ICT projects, means that there is a viable path from research to large-scale impact.

1.1.3 The obstacle: Inadequacy of Existing Solutions

Until now we have discussed both the stringent need for rural network connectivity, and the existing opportunities to provide this connectivity. We believe that networking technology has a large role to play in rural areas of developing regions, but that technology to date has been a poor fit in these areas, prompting for the need for research exploring more appropriate networking technologies.

The main obstacles in deploying existing technology in rural areas of developing regions are low purchasing power and the low user density, making the available alternatives too expensive. For example, the typical investment cost for a telephone land-line is US$500, which is acceptable in the US, where 90 percent of households can afford to pay $30 a month for telephone service. In contrast, in India, more than 60 percent of the population can afford at most US$5 a month for communications [68].

This makes wire-line solutions economically unviable, especially for low-density, remote areas. Satellite networks have the advantage that they provide connectivity even in most remote places, independent of any wireline infrastructure, making them the only available choice in many locations. Unfortunately, VSAT equipment is also very expensive. In developing regions, VSAT installation costs amount to over US$10,000, with recurring monthly costs of over US$2,000 for a 1 Mbps link [110]. These high costs are driven by the scarce satellite transponder capacity, which needs to be amortized among large user populations to make it affordable. As a consequence, in low user-density regions, VSAT is affordable only for businesses or wealthy users.

Networks with a base-station model such as WiMax, and cellular networks like

GPRS and CDMA, have an asymmetric design philosophy where expensive base stations are amortized by large number of cheap clients over many users. In low-density regions, such base stations simply do not cover enough users to be economically viable. In some cases this issue can be overcome by government policies sponsoring cellular rural coverage: China has dictated good coverage as policy, despite the economic issues. Other countries either subsidize rural users through taxation, much like the US universal access tax, or require some rural coverage as part of spectrum allocation. In Appendix A we analyze some of these alternatives in detail.

All of the above arguments, namely the *need for rural network connectivity*, the *opportunities to provide it today*, and the *inadequacy of existing networking alternatives* prompts us to investigate an alternative solutions for rural network connectivity. We design these novel wireless solutions specifically for coverage in areas with sparse user populations infrastructure, featuring cost and performance advantages that make it competitive in these environments.

1.2 Wi-Fi-Based Long-Distance (WiLD) Networks

In recent years, the technologies specified by the IEEE 802.11 series of open standards [61] (also called Wi-Fi) have experienced a massive adoption in the industrialized nations, being the most used technology in the deployment of wireless local area networks. Wi-Fi radios can deliver high network throughput (up to 54Mbps), and the economies of scale (an estimated 213 million Wi-Fi chipsets shipped in the year 2006 alone [100]) have driven down the cost of Wi-Fi radios to a few dollars per unit.

Besides the edge given by low cost, high performance and free spectrum, Wi-Fi technology has the advantage of being specified by open standards, and of using open-source drivers [24]. This facilitated a large body of research analyzing and optimizing 802.11 performance [33, 41, 59, 70, 77, 103, 125], or investigating quality-of-service mechanisms in 802.11 networks [22, 75, 103, 116].

1.2.1 Outdoor Wi-Fi links

Given the above mentioned advantages, researchers have also investigated the use of 802.11 radios outside the intended scope of short-range indoor networks. By connecting Wi-Fi radios to outdoor antennas, researchers have successfully used Wi-Fi radios to form outdoor multi-hop mesh networks [20, 34, 55, 66, 97].

Among the first groups investigating this possibility was MIT's Roofnet project [36] that deployed a Wi-Fi campus mesh network consisting of a few tens of nodes. The outdoor links spanned hundreds of meters or even kilometers [20], taking advantage of outdoor omnidirectional antennas. Other groups, including the Digital Gangetic Plains [5] from IIT Kanpur, and our own group (TIER Berkeley [114]) took this idea further, by using high-gain directional antennas to amplify the wireless signal and enable the deployment of links spanning tens of kilometers; Raman *et al.* [32, 92] deployed tens of such links, some more than 30km long. In parallel, the TIER group also deployed a campus wireless testbed with links of comparable range.

Results from early experiments, which were later confirmed by several studies [40, 107], showed that long links can incur very high packet loss rates, making the useful high-level throughout (especially TCP) very low. However, the same experiments indicated that,

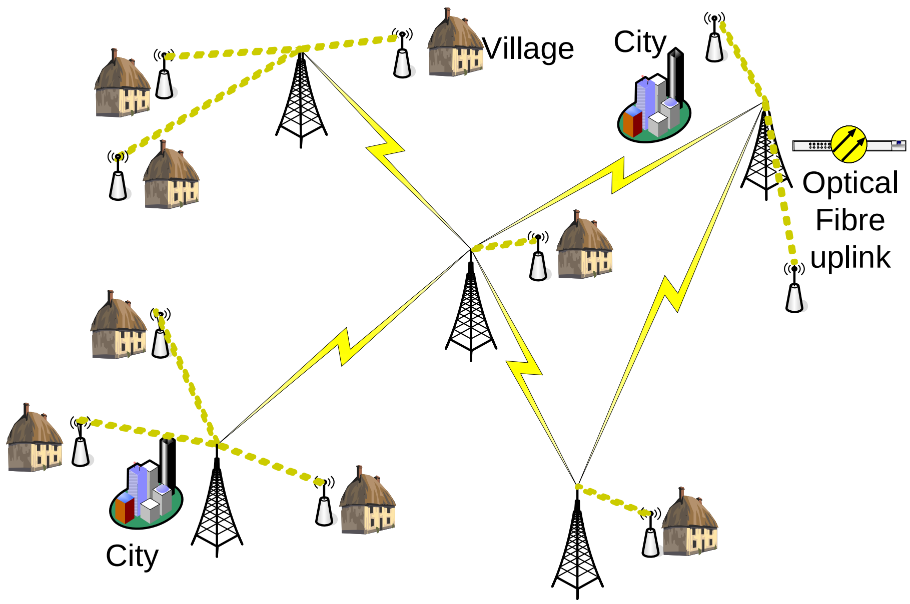


Figure 1.2: Example of envisioned deployment model

when high-power 802.11b cards (23dBm or more) and directional antennas were used, the signal strength of the links was not an issue, allowing in theory the deployment of links hundreds of kilometers long. This made us believe (and in retrospect [79, 84] we were right) that the high loss rate was a result of the inadequacy of the 802.11 carrier-sensing MAC in long distance operation, a problem that could be overcome by smart protocol redesign. Later studies [84, 93, 99] demonstrate the feasibility of implementing TDMA-based medium access using commodity 802.11 radios, which represents the necessary support for high-performance long-distance operation, as we will show in the later chapters of this dissertation.

1.2.2 Deployment Model and Envisioned Applications

Assuming we can solve the performance problem associated with 802.11 long distance operation, we now have the opportunity to deploy inexpensive (below US$1000 per link, including radios, antennas and wireless) links running at several *Mbps* and spanning tens, even hundreds of kilometers.

This enables us to envision a new model for providing network coverage to rural areas. This model, illustrated in Figure 1.2, uses a combination of a) long-distance (5-100 km) *point-to-point* high bandwidth backhaul links, connecting towns and villages, and b) medium range *point-to-point* or *point-to-multipoint* access links distributing this connectivity to schools, hospitals, Internet kiosks and individual users. This model renounces the ability to provide "carpet coverage" in the way cellular and WiMax deployments do, choosing to instead connect select remote areas with high-bandwidth connections.

This architecture is a great fit for the set of applications that we target in these environments [37]: Internet and VoIP telephony kiosks, remote education, telemedicine and e-governance. All of these applications require moderate to high-bandwidth connectivity, and many of them feature packet delay constraints. For example, the kiosk connection is shared among many simultaneous users, and some users require bandwidth guarantees to carry out VoIP calls. Telemedicine applications, such as the ones we used in the Aravind healthcare centers [112], are many times based on teleconferencing, and require high resolution video feeds for accurate remote diagnosis. Low delay is also important to allow VoIP calls and interactive tele-conferencing.

1.3 Goal and Summary of Contributions

WiLD networks have the potential to deliver high bandwidth, cost-effective connectivity, but the performance delivered in early experiments and deployments was very low. Individual links exhibit very high and asymmetric packet loss (sometimes as high as 80% [107]), making high-level protocols (especially TCP) experience a combination of low bandwidth and large packet delay. Moreover, the performance degrades drastically as we scale network paths to more than one hop, making paths exceeding three hops unusable.

The goals of this dissertation are to understand the factors preventing WiLD networks from operating at their potential performance, and then to design and implement the mechanisms that would allow this technology to reach its potential. The contributions of this thesis are the mechanisms that enable the achievement of *high-bandwidth and low delay in large-scale WiLD networks*.

We take an incremental approach, where we start with a) improving the performance of *individual links*, we continue with b) optimizing entire *end-to-end paths*, and finish with c) maximizing *network-wide performance*. Each of these steps involves a diagnostic part – a study of the causes for low performance and potential solutions – and continues with the design, implementation and evaluation of performance-maximizing solutions. All our solutions are implemented at the MAC and link layer, but sometimes require additional information from higher levels of the network stack (*e.g.,* information regarding network topology).

We summarize the main contributions of this thesis in the following:

- *Why don't WiLD links perform well?* characterization of loss and inefficiencies in WiLD links. The first contribution is the identification and measurement of the primary causes of high loss and low bandwidth in long distance link [107]. By performing a detailed measurement study, we find these causes to be both protocol and channel related. The protocol-related causes are a) the breakdown of the carrier-sense mechanism, leading to packet collisions and inter-link interference, and b) an inadequate flow-control mechanism, inappropriate for long distances. The main channel-related cause is the interference with other Wi-Fi traffic in the vicinity of one of the link endpoints.

- *Maximizing performance in WiLD links:* A medium access approach to eliminate link inefficiencies and avoid interference. The second contribution is the design and implementation of a TDMA-based medium access control protocol that eliminates all the *avoidable* sources of loss (collisions and interference with other WiLD packets). The MAC also implements an efficient, sliding-window flow-control with per-hop bulk acknowledgments. This brings the bandwidth of WiLD links close to their theoretical potential (more than 6Mbps for 802.11b links).

- *Maximizing performance in end-to-end paths:* Per-hop loss recovery mechanisms. The third contribution maximizes end-to-end performance, by design and implementing the appropriate per-hop loss recovery mechanisms that enable higher level protocols (mainly TCP) to achieve high end-to-end bandwidth. The combination of ARQ and FEC enables the easy negotiation of the tradeoff between bandwidth and delay. The second and third contributions are implemented as part of the WiLDNet [84] system, and deployed in several real-world WiLD networks in developing countries.

- *How well can WiLD networks work?* Understanding the capacity of WiLD networks We also look at WiLD networks as a whole, in an attempt to maximize network wide performance. An important contribution is understanding the theoretical capacity of a WiLD network. We use linear programming techniques to quantify this capacity, and then compare the throughput achieved when using existing practical MAC approaches (including WiLDNet) against the maximum throughput achievable by optimal scheduling link transmissions according to traffic. We conclude that knowledge of traffic information is essential in maximizing network capacity.

- *Maximizing network-wide WiLD performance:* An adaptive MAC that optimizes link utilization by using local information of traffic demand. We revisit the MAC protocol design, and propose a novel scheme that improves link utilization and reduces packet delay by independently adapting link transmission slots for each link (and link direction), as a function of current traffic demand. This new MAC, called JazzyMAC, maintains the constraint of non-interference between adjacent links, and is proven to outperform TDMA MACs using fixed slot sizes in all situations.

1.4 Roadmap

Chapter 2 presents an overview of WiLD networks, including background on the 802.11 MAC, the use of Wi-Fi radios in medium and long-distance links, and existing research on PHY and MAC level techniques to extend the range of point-to-point Wi-Fi links. Chapter 3 presents a detailed measurement study of the sources of loss and inefficiency in long-distance links. Chapter 4 presents the design and implementation of WiLDNet, a system that includes an efficient TDMA MAC, and appropriate link-level loss recovery mechanisms. Chapter 5 investigates the maximum throughput that could potentially be achieved in WiLD networks, while maintaining interference avoidance constraints. Chapter

6 presents JazzyMAC, an improved MAC protocol that uses traffic information to dynamically adapt the size of link transmission slots on each link, in order to maximize network wide throughput, and maintain a low average delay. Finally, Chapter 7 summarizes the contributions, and limitations of the work presented in the dissertation, as well as the real-world impact of this work. Appendices A and B present additional information on alternative technologies for network connectivity in developing regions, with particular focus on CDMA450, one of the most (if not the most) promising alternative for rural connectivity.

Chapter 2

Overview of WiLD Networks

This chapter starts with an overview of the 802.11 family of standards, and of the projects that use 802.11 radios to build meshes and long-distance directional (WiLD) networks. We then discuss physical and medium-access layer techniques that can be used to increase the range of WiLD links, and summarize related work in these areas.

2.1 IEEE 802.11 Overview

The 802.11 family of radio protocols have enjoyed an incredible level of popularity in Europe and the United States, with hundreds of millions of radios manufactured and sold every year [100]. By implementing a common set of standard protocols, and operating in unlicensed wireless spectrum, these radios are highly interoperable, allowing customers to use 802.11 equipment in any Wi-Fi networks, without fear of vendor lock in. The highly competitive market drove the volumes up, and the prices down, making these radios very attractive for developing markets.

The original IEEE802.11 standard defines a medium access control (MAC), and three kinds of physical layers: direct sequence spread spectrum (DSSS), frequency-hopping spread-spectrum (FHSS) and infrared (IR). Maintaining backward compatibility to the original DSSS 802.11 specifications, provisioned for 1 and 2 Mbps, the 802.11b was adopted to support rates of 5.5 and 11Mbps as well, operating in the 2.4GHz band. Another extension was 802.11a, which uses orthogonal frequency division multiplexing (OFDM) as the physical layer. 802.11a can support data rates ranging from 6Mbps to 54Mbps, operating in the 5.8 GHz band. The 802.11g extension combines the modulation techniques of both 802.11b and 802.11a. It is backward compatible with 802.11b, using the 2.4 GHz band, but achieving rates up to 54Mbps.

In addition to these original specifications, QoS enhancements for delay-sensitive applications (such as VoIP) were introduced in the 802.11e standard, through modifications to the original MAC specifications. To increase link throughput, a pending standard amendment(IEEE 802.11n) has also been proposed. This builds on previous 802.11 standards by adding multiple-input multiple-output (MIMO) and channel-bonding (40MHz) operation to the physical layer, and frame aggregation to the MAC layer. This addition increases datarate from 54Mbps to a maximum of 600Mbps. Besides these above standards, many vendor specific extensions were also introduced, providing higher speeds, better security mechanisms and even increased range.

Today, 802.11 WLANs are considered the wireless versions of the Ethernet wired WLANs. The original 802.11 MAC supports two modes of operation: the Distributed Coordination Function (DCF), and the Point Coordination Function (PCF). DCF provides contention-based access. The Ethernet CSMA/CD mechanism, not feasible with wireless transceivers that cannot listen while they send, was replaced by a CSMA/CA mechanism. PCF, on the other hand, provides contention-free access. The two modes can be used alternatively in time, in the sense that a contention-free period of the PCF can be followed by a contention period of DCF.

2.1.1 The DCF Mode

DCF employs the CSMA/CA mechanism and works as follows. A station (including the AP) with a packet ready for transmission senses whether or not the channel is busy. If the channel is idle for an interval called *DCF Inter Frame Space* (DIFS), the station starts packet transmission. Otherwise, the station continues to monitor the channel status (busy or idle). After finding the channel idle for a DIFS interval, the station waits for an additional random backoff period, measured in discrete time intervals, called slots. For each slot time in which the channel remains idle, the backoff interval is decremented by one. When the interval value reaches zero, the station starts packet transmission. During this backoff period, if the channel is sensed busy in a slot time, the decrement of the backoff interval stops (i.e., is frozen) and backoff is resumed only after the channel has been idle for a full DIFS.

The backoff mechanism for the DCF is exponential. The backoff interval is uniformly chosen between 0 to CW slot time units, where CW is the congestion window. The value of CW depends on the number of failed transmissions for the packet. At the first transmission attempt, CW is set to $CWmin$, and increased exponentially at each retry, until it reaches the maximum contention window value $CWMax$, where it levels. After a

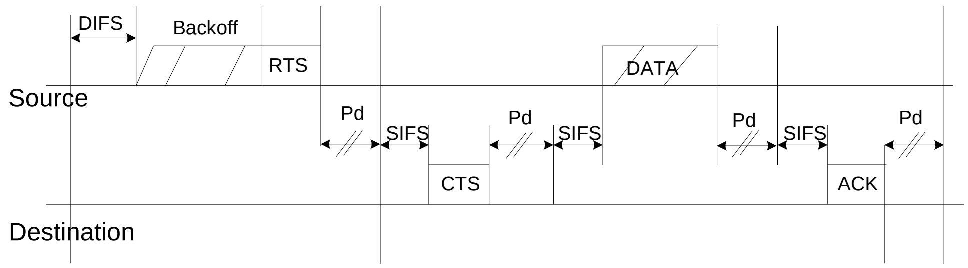


Figure 2.1: Packet Transfer in 802.11 DCF Mode

successful transmission CW is reset to the minimum value.

The receiver is required to send an ACK packet for each successfully received packet. A simple stop-and-wait protocol is used. The sending station is expected to receive the ACK within a SIFS (Short Inter Frame Space) interval after the packet transmission is completed. If the ACK does not arrive at the sending station within a specified ACK_timeout period, or it detects transmission of a different packet on the channel, the original transmission is considered failed and retransmission must be performed. The protocol also implements an (optional) RTS/CTS mechanism that solves the hidden terminal problem. The mechanism uses a network allocation vector (NAV) to specify the expected duration of the current transfer (including the expected response to the current packet). The value of the NAV indicates the amount of time is expected to pass until the channel becomes idle, and this value is included as a field in each packet. Thus, the NAV acts as a virtual carrier sense mechanism. The MAC uses the combined physical and virtual sensing to avoid collisions.

The functioning of the DCF mode is illustrated in figure 2.1. The Pd value represents propagation delay. The figure is self explanatory.

2.1.2 The PCF Mode

PCF is a contention-free mode of operation, that assumes an access-point node (AP). The AP polls its associated mobile stations one after another, by sending polling messages. If the AP has data to send to the station being polled, this data can be included in the poll message. If the polled station has data for the AP, it is sent in the response message. When applicable, an acknowledgment (which acknowledges receipt of a previous data frame from the AP) can also be included in the response message. This mode of operation is intended to support QoS by enabling the AP to regulate the bandwidth allotted to each of the mobile stations. However, it is ill-suited to long-distance outdoor links (discussed later), and because it is optional, very few cards actually implement it.

The 802.11e standard enhances DCF and PCF, through a new coordination function called the Hybrid Coordination Function(HCF). With HCF there are two access methods: Enhanced Distribution Channel Access (EDCA), an extension of DCF, and HCF Controlled Channel Access (HCCA) which is similar to PCF.

2.1.3 EDCA

This channel access method is similar to DCF, but introduces traffic priority classes. This allows high priority traffic a better chance to contend for the medium. Each priority level is assigned a different Transmit Opportunity (TXOP), which is the bounded interval during which a station can send as many frames as possible. In practice, different transmit opportunities are achieved by varying two DCF-specific parameters: the maximum size of the congestion window (CW), and the size of the DIFS interval, here called AIFS.

Wi-Fi Multimedia (WMM) certified APs enable EDCA, while other 802.11e enhancements are optional.

2.1.4 HCCA

This method works a lot like PCF, and is considered the most advanced coordination function, allowing QoS to be configured to great precision. Unfortunately, HCCA support is not mandatory, and we are not aware of any APs implementing this method.

2.2 802.11 in Long Distance Networks

Given the low cost and free spectrum, researchers and Wi-Fi enthusiasts have experimented with deploying 802.11 radios in outdoor links, either as mesh networks or planned point-to-point in links.

One of the first Wi-Fi mesh networks was built by the MIT Roofnet [36] project, using small, low cost single-board linux-based computers mounted on building rooftops, and equipped with 802.11b radios. To amplify the signal, these radios were connected to external omnidirectional antennas. The Roofnet deployment featured a few tens of nodes, and many more links, ranging from a few hundred meters to a few kilometers in length.

A very similar deployment was BARWN (the Bay Area Research Wireless Network [13]), that aimed to deliver broadband Wi-Fi connectivity at very low-costs (< $300 per Customer Premises Equipment unit). The BARWN deployment also featured point-to point long links (some accross San Francisco Bay), by connecting the Wi-Fi radios to directional antennas in line-of-sight conditions.

The High Performance Wireless Research and Education Network [7], an NSF-funded research project based in UC San Diego deployed a large network of directional wireless links, with many nodes in hard-to-reach remote areas in rural California. Some of these links use (high cost) proprietary technology, while some of them use 802.11 radios. HPWREN deployed one of the earliest very long Wi-Fi links – a 72km link in 2002.

A project exploring the deployment of longer Wi-Fi links in order to connect together remote villages in developing regions was the Digital Gangetic Plains [5] research project at IIT Karapur in India. The testbed deployed in India consisted of about a dozen point-to-point long links, the longest of which was 39km. To enable such long links, the Digital Gangetic Plains utilized high-gain 24dBi parabolic grid directional antennas, and to ensure line-of-sight the link endpoints were often placed on top of tall communication towers.

The Berkeley TIER group was also involved in similar experiments, deploying a wireless testbed comprising several links in the Bay Area, with some links exceeding 40km in length. The TIER routers, featuring the protocol modifications discussed later in this dissertation, were then deployed in several real-world networks in India and Ghana. Moreover, unamplified links of 279km and 382km were demonstrated [50] in Venezuela, in collaboration with the EsLaRed fundation [6].

Although the standard 802.11 protocol and commercial APs and network cards are appropriate for short-distance connectivity, high-performance long-distance links are only feasible by using a specialized set of PHY and MAC level range extension techniques. We begin by discussing the means of extending the physical range of these links, and then continue by discussing the 802.11 protocol issues related to long-distance operation.

2.3 Extending Physical Range

In establishing a long-distance link, the most important issue is to ensure that the wireless signal is strong enough to permit communication. A point-to-point link consists of two radios, each using an *antenna*, the two being separated by the physical path to be travelled by the wireless signal. In order to establish communication between the endpoints, a certain minimum level of signal needs to be collected by the receiving antenna and presented as input to the receiving radio. Determining whether the link is feasible is called a *link budget* calculation. The link feasibility depends on the power of the transmit radio and the amplification given by the antennas, but also on the energy losses incured in cables, antenna connectors, and along the wireless path.

2.3.1 Link Budget Calculation

The parameters involved in link budget calculation are:

Transmit power: This is usually expressed either in milliwatts or in dBm, and for commercial 802.11 radios this usually ranges between 30mW to 200mW, and recently as high as 600mW (28dBm) [15].

Antenna gain: This is the gain in signal strength resulting from the amplification provided by use of an antenna. Antennas amplification is symmetric, acting both on transmitted and received signals. Typical gains are of 8dBi for omnidirectional antennas, 12dBi for sector antennas, and 19-24dBi for parabolic directional antennas [8].

Receiver sensitivity: This is the minimum level of received signal at which the communication is still distinguishable. This is expressed in negative dBm, and depends on the datarate used by the transmitter. At smaller datarates (*e.g.,* 1Mbps) the wireless symbols are easier to distinguish than the ones at high datarates (*e.g.,* 54Mbps). Typical receiver sensitivity levels range between -70dBm and -95dBm.

Losses in cables and connectors: Additional energy is lost due to attenuation along cables connecting the radios to their antenns, and in the antenna connectors. Cable attenuation depends on the cable quality and the signal frequency. For example, at 2.4GHz, typical attenuations range from 0.05dB to 0.8dB per foot of cable. Connectors can attenuate by 2 or 3dB.

Path loss: The most significant signal attenuation happens over the air. There are several factors contributing to this: *free space loss*, *attenuation* and *multipath*. *Free space loss* happens because power is diminished by the geometric spreading of the wave. This factor increases with distance (since the wave spreads more the further it travels), but is independent of environment and other factors, and therefore there are no available techniques to alleviate this.

Another factor contributing to path loss is the *attenuation* that takes place as the wave passes through or around obstacles such as trees, walls or windows. This attentuation is difficult to quantify, and can be as much as 10-20dB for trees or walls. However, this part of the attenuation can be avoided by ensuring clear line of sight between transmitter and receiver, *i.e.,* ensuring that a cone around the direct line between the transmitter are received, called the *Fresnel zone* [17], is free of any obstacles. We discuss this issue in more detail later on in this chapter.

Finally, signal is also attenuated due to *multipath*, or *signal dispersion*, which happens because, as RF energy leaves the antenna, it spreads out. Part of this energy travels to the receiving antenna on the direct path, but part bounces off the ground and other obstacles. Since the reflected signal travels a longer path, it arrives at the receiver later than the direct signal. The receiver combines signals coming on different paths, but due to the difference in the experienced delay (which translates into a phase shift), these signals can either sum up, or they can cancel each other. In practice, multipath is a very location-specific phenomenon, being much more pronounced in enviromnets conducive of reflections (*e.g.,* urban areas). However, our experiments [107] show this to be a small factor in very long rural directional links.

If we consider the combined effects of free space loss, attenuation and multipath, the path loss can be expressed as:

$$Loss(db) \approx 92.45 + 20 \cdot log(F) + 10 \cdot n \cdot log(D) + L_{attenuation} \tag{2.1}$$

where F is the communication frequency in GHz, D is the distance in km, n is the free space loss exponent, which depends on environment, and $L_{attenuation}$ is the attenuation given by solid obstacles along the path. An n equal to 2 represents the ideal case with essentially no multipath, an exponent of 3 is typical for outdoor environments, while a value of 4 is representative of indoor links.

Having all the factors that influence the feasibility of a link, let us discuss an example of computing the link budget. Let us assume that we have a 20km outdoor with perfect line of sight and no multipath ($n = 2$ and $L_{attenuation} = 0$). The link uses high-powered radios transmitting at 200mW (23dBm) and featuring a receive sensitivity of $-90dB$. Each radio is connected to a directional antenna with a gain of 19dBi. In this scenario, the path loss can be computed as: $Loss(db) = 92.45 + 20 \cdot log(2.4) + 10 \cdot 2 \cdot log(20) + 0 \approx 126dB$. Then, the receive signal strength equals:

```
Transmit power (23dB)
+ TX Antenna gain (19db)
- TX Cable loss (4db)
- Path loss (126db)
+ Antenna 2 gain (19dB)
- RX Cable loss (4dB)
= -73db
```

Since the network card has a sensitivity of -90dB (smaller than -73dB), the link is feasible, and we have some room (17dB) to accomodate attenuation due to obstacles or multipath.

2.3.2 Range Extension Techniques

Given the terms to consider in computing the link budget, the obvious ways to extend the link range is by either increasing gain or by decreasing loss.

Increasing gain: One way to increasing signal strength is to use high-power radios, such as Ubiquity's XR series of radios [15], that can transmit at up to 28dBm (600mW). Even higher transmit power levels can be achieved by employing stand-alone power amplifiers, but this

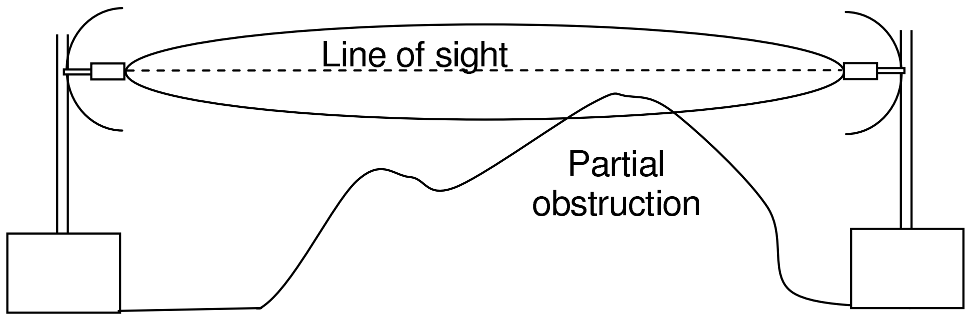


Figure 2.2: An example where the Fresel zone is partially obstructed, even though the visual line of sight appears clear

alternative is generally expensive, and rarely needed. Another important gain contributor is the use of high-gain directional antennas at both link ends. High gain antenna may be of many designs, but all allow transmitting a narrow signal beam, often nulling out nearby interference sources. Today, 24dB and even 30dB Wi-Fi directional antennas are available commercially, at prices below US$100. Very cheap solutions, taking advantage of ordinary cans (Cantennas [12]) and cookware (WokFi) can also be used to add 10+ db of gain.

Decreasing loss: The most important aspect in reducing path loss is to ensure clear line-of-sight between link endpoints. The line of sight notion (LOS) is very intuitive when talking about visible light, since it simply means a straight line without obstacles between endpoints. It is less intuitive however when talking about propagation of wireless signals. Without going into details, in order for the wireless signal to preserve its energy, a conical zone surrounding the direct line between the link endpoints needs to remain clear of obstacles. This zone is called the first Fresnel zone [60] (there are many Fresnel zones, but we are mainly concerned with the first). Figure 2.3.2 presents an example where the Fresnel zone is partially obstruted, while the visual line is clear. The way to compute the radius of the first Fresnel zone at each point along the path takes into consideration the distance from the endpoints, and of course the frequency of the transmitted signal:

$$r \approx 17.31 \cdot \frac{\sqrt{D_1 \cdot D_2}}{F \cdot (D_1 + D_2)} \tag{2.2}$$

where r is the radius of the zone in meters, D_1 and D_2 are distances to the link end points in meters, and F is the frequency in MHz.

In order to ensure clear line-of-sight, the network plannners must consider elevation profiles and other possible obstacles. High-elevation locations, such as rooftops or communication towers are usually a good choice to place link endpoints. Given that Wi-Fi routers and antennas are very light, many inexpensive solutions for tall communication towers are available [21].

Decreasing Interference: A technique to increase the Signal to Noise and Interference Ratio(SINR) is to decrease the channel bandwidth. In this mode of operation, the signal strength remains the same as for wider channels, but white-noise interference is lower (proportionally to the width of the channel), resulting in increased card sensitivity. To take advantage of this opportunity, the 802.11-2007 standard adds 10 MHz and 5 MHz OFDM modes to the 802.11a standard, and extend the time of cyclic prefix protection from 0.8 s to 3.2 s, quadrupling the multipath distortion protection. Commonly available 802.11a/g chipsets, such as the ones built by Atheros [2], already support the OFDM "half-clocking" and "quarter-clocking" specified in the 2007 standard.

2.4 Extending Range at the MAC Layer

The techiques presented in the previous section can be used to create point-to-point 802.11 links spanning many tens of kilometers, and featuring good wireless signal. But this is only one of the necessary steps in making these links functional. The other piece of the puzzle is making the 802.11 MAC protocol perform properly in these settings.

2.4.1 Performance of the standard 802.11 MAC in long links

Because the 802.11 MAC was designed to operate in short-distance, broadcast environments, the protocol makes certain assumptions about packet propagation delay and carrier sensing that make it a bad fit for long-distance links. In the following we discuss the use of PCF and DCF operation modes in long links.

PCF in Long-Distance Links: Timing requirements in the 802.11 standard make 802.11 PCF infeasible. In particular, the ACK sent by the polled station must be received by the AP within the SIFS time interval (see Figure 2.3a), which for 802.11b is 10 μs. This corresponds to a round trip of approximately 3km, limiting links to less than 1.5km. Another practical problem related to PCF is that, because it is optional, the mode is only supported in very few (if any) of the current wireless cards on the market.

DCF in Long-Distance Links: In contrast to PCF, DCF can be used in long links, because the timing is more lenient for ACK timeouts. The difference is illustrated in Figure 2.3. In this mode, if an ACK is not received in time, the transmitter needs to backoff before retransmitting. Therefore there is a larger time opportunity for the ACK to arrive in time. Because of this, the ACK_Timeout (not explicitly specified by the standard)

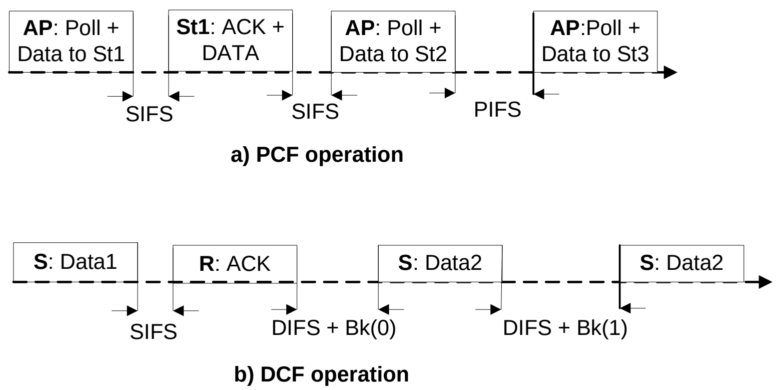


Figure 2.3: PCF and DCF timing

is larger than a DIFS interval in typical implementations. Some protocol implementations use a timeout of $2 \cdot SIFS + DIFS$, supporting links of up to 10km, and more lenient implementations are also available.

Although usable for long-distance directional links, DCF has several shortcomings:

- Low throughput. This is due to protocol inefficiencies induced by the large propagation delay. If used, the RTS/CTS mechanism adds a round trip delay to every transmission. The employed stop-and-wait protocol, which requires an ACK for each packet, adds another round trip delay for every packet. This overhead can be significant, especially for the 802.11b, where the long PHY preamble and header are sent at the lowest supported datarate, namely 1Mbps. Finally, the random backoff also adds to the inefficiency.

- High packet loss. Given the long propagation delay, listening for a clear channel does not guarantee that the channel is clear indeed (since the other end of the link might have innitiated a transmission that has not yet arrived at this end). This increases the opportunity for packet collisions. The directional nature of the links also results in the amplification of the hidden terminal effect [115], which in turn results in inter-link interference, as we see in Chapter 3. This results in very large packet loss rates.

2.4.2 802.11 MAC layer modifications

The problems exhibited by the 802.11MAC make the performance in long links very poor. To address this issue, several researchers (including the TIER group) have proposed and implemented MAC modifications, or even new MAC implementations. In the following we briefly discuss these projects. Some of this research was performed concurrently with the research presented in this dissertation.

Given the inadequacy of carrier sense in long-distance, directional settings, an alternative solution that emerged as a good option was to use a time division multiple access (TDMA) style medium access control, with link ends taking turns transmitting for pre-established fixed periods (called TDMA slots).

Roofnet: The Roofnet mesh deployments rely on the standard CSMA/CA MAC, but in order to locking WiFi stations to a single AP, they use the the "pseudo-IBSS" (also "pseudo-adhoc") mode provided by the MadWifi [24] driver, a mode in which stations don't need to be associated in order to send or receive packets. They also modify the driver to provide a better integration with the Click [71] modular router, by providing better visibility and control of MAC parameters (*e.g.,* packet datarate and SINR) at higher network layers.

Overlay MAC: Rao *et al.* [98] implement an Overlay MAC Layer (OML), that works on top of the 802.11 MAC. Their solution is implemented using the Click modular router, and uses loosely synchonized clocks to divide the time into equal size slots. Although OML

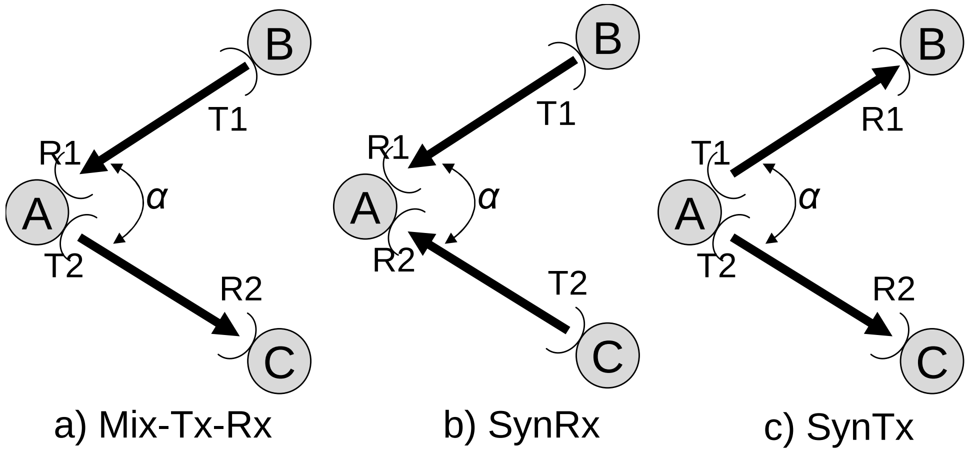


Figure 2.4: SynOp Cases

is tailored for short-range broadcast environments, and for network hosts equipped with a single WiFi radio, it is the first working implementation of a TDMA-style access solution for WiFi networks.

SynOp: Raman *et al.* [92, 93] propose to increase the *spectral efficiency* of WiLD networks by allowing several adjacent links to operate on the same wireless channel. This can be achieved by synchronizing among adjacent links to the same router, such that all either transmit simultaneously, or receive simultaneously. This idea builds on the *burst synchronization* concept, well established in the wireless literature [105]. Given the large relevance of this observation to the research presented in the later chapters, we discuss this mode of operation in detail.

Consider the adjacent point-to-point links depicted in Figure 2.4, separated by an angle α. Now consider the following three usage scenarios:

Mix-Tx-Rx: In this scenario, depicted in Figure 2.4a), T2's transmission interferes with R1's reception, due to the physical proximity between the radios and the presence of antenna side-lobes. Therefore, operating the links in this mode is infeasible.

SynRx: During simultaneous receive, (Figure 2.4b)), T2's transmission is seen as interference at R1, and T1's transmission is seen as interference at R2. For the interfering signal to be ignored, the difference between useful signal and interference must be larger than a certain threshold $Th_{isolation}$, which depends on modulation and data-rate; *e.g.,*with 802.11b at 11Mbps, $Th_{isolation} \approx 10dB$ [92, 107]. Fortunately, this level of isolation can usually be ensured through the difference in gain levels provided by the directional antennas, if the links are separated by a sufficiently large angle. If we denote the difference between the antenna gain of the main lobe and the gain at an angle α away from the main lobe by S_{alpha} (also called the rejection level at angle α), then adjacent links are interference free under the following condition [92]:

$$|P_{R1} - P_{R2}| < S_{\alpha} - Th_{isolation}, \tag{2.3}$$

where P_{R1} and P_{R2} are the receive power levels at R1 and R2 respectively.

For example, if links use typical $24dBi$ grid antennas [8], an angular separation of more than 10° (half the width of the antenna main lobe) translates into an isolation of at least $25dBi$ (sometimes larger, not monotonically increasing with the separation angle). This means that 802.11b links receiving simultaneously are interference-free if $|P_{R1} - P_{R2}| < 15dB$. This can be easily satisfied by a large range of values ($e.g.,P_{R1} = P_{R2}$), and even if the path loss of the two links is very different, the condition can be satisfied by adjusting the radio transmit power accordingly (by reducing the TX power on the stronger link).

- *SynTx:* With simultaneous transmissions, as in Figure 2.4c), interference may occur at nodes B and C, but not at node A. Once again, R_1 may see interference from T_2, and R_2 from T_1. Given the symmetry of the two links, ensuring non-interference during SynTx can be done by enforcing a similar condition to that in equation 2.3.

In summary, simultaneous synchronized operation (SynOp) can allow multiple adjacent WiLD links to simultaneously use the same wireless channel *provided* the links are separated by a sufficiently large angle α and the radio transmit powers are chosen to satisfy the constraint from equation 2.3. Given the gain pattern of typical grid directional antennas [8], an angular separation α larger than $30°$ provides generous isolation between adjacent links; this has also been demonstrated experimentally [92, 93] and later validated in the TIER network deployments [84, 112].

2P: Raman *et al.* [93] build on the SynOp idea, and design 2P, a Spatial-reuse TDMA (STDMA) MAC. In 2P, long-distance links alternate between *transmit* and *receive* slots of fixed lengths. These slots are synchonized according to the SynOp principle, with nodes either transmitting or receiving, but not both simultaneously. These constraint can be efficiently met in bipartite network topologies, as they allow nodes to use all of their links simultaneously and alternate as a group between send mode and receive mode. Therefore 2P constrains the network topology to be bipartite.

Figure 2.4.2 shows an example of such a bipartite network. Using 2P, all nodes in partition A first transmit on all of their links (for a time slot of size $t_{A\rightarrow B}$). Following this, all nodes in partition B transmit on all their links (for a time slot of $t_{B\rightarrow A}$). The ratio between these slot sizes regulates the bandwidth allocation for every network link between the two partitions. In practice, $t_{A\rightarrow B}$ and $t_{B\rightarrow A}$ are almost always set to be equal since this

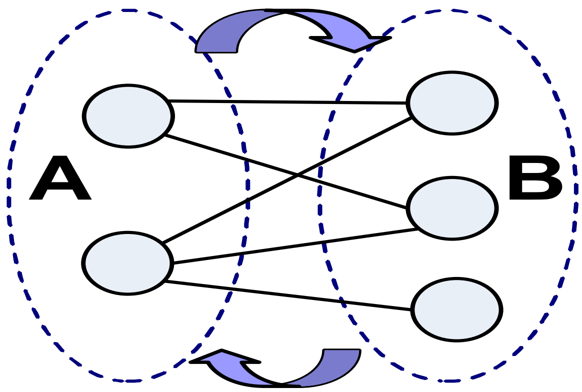


Figure 2.5: Example of 2P operation

maximizes throughput for traffic paths spanning more than two hops [91, 93].

2.5 Summary

Thic chapter covered background information on WiLD links. It presented the details of the 802.11 protocol, and insight of the way the protocol works in long links. We then discussed how to provision long-distance links at the physical level, what is the equipment to be used and how do characteristics of various link elements affect the link budget. Finally, we discussed related work on MAC modifications or additions that enable proper long-distance operation.

Chapter 3

Characterization of WiFi Long Distance Links

In this chapter we characterize the network performance achievable by using the 802.11 protocol over long-distance wireless links. Achieving good signal strength proves an easy task, by leveraging the use of directional antennas in line-of-sight conditions. However, despite this promising result, the bandwidth achieved over these links is low, and the packet loss is usually very high and bursty. We therefore perform a detailed measurement study of the sources of packet loss and other inefficiencies in WiLD network settings. We consider both loss and inefficiencies caused by the inappropriateness of the 802.11 protocol for long

The material presented in this chapter is collaboration work, done together with Anmol Sheth, Rabin Patra, Sonesh Surana, Lakshminarayanan Subramanian, and Eric Brewer, all part of the TIER group. Most of the material was previously published as "Packet Loss Characterization in WiFi-Based Long Distance Networks" [107], and the material presenting protocol-induced losses and inefficiencies was previously published as "WiLDNet: Design and Implementation of High-Performance WiFi Based Networks" [84] distance, directional operation, but also losses induced by the wireless channel, which are also exacerbated by distance and directionality. We then characterize the nature, magnitude and time variability of these losses, in order to inform the design of appropriate loss-recovery mechanisms.

We find that the 802.11 protocol is ill-suited for operation in WiLD scenarios, suffering from substantial inter-link interference and packet loss due to collisions on a single link. Among channel-induced losses, we find external WiFi interference to be the most prominent.

Taking into account the loss characteristics, we end the chapter by discussing prelimary methods to alleviate both channel and MAC-induced losses, which we refine, implement and evaluate in the following chapter.

3.1 Loss in Long-Distance Links

Despite the promise of low-cost connectivity, the performance of the first deployed WiLD network links has been abysmal. This poor performance is primarily triggered by the high loss variability observed on WiLD links. Figure 3.1 shows the loss rate measured over two of our links ("K-P" and "B-R") over a period of 3 hours on different days. The loss rate was averaged over 30-second intervals for a 1 Mbps unidirectional UDP CBR traffic flow with the MAC-layer ACKs turned off and retries set to zero. For the K-P link, the sender and receiver roles were reversed at the end of every 30 second time interval. This allowed us to test the symmetry of our links.

We make the following important observations: 1) WiLD links demonstrate high

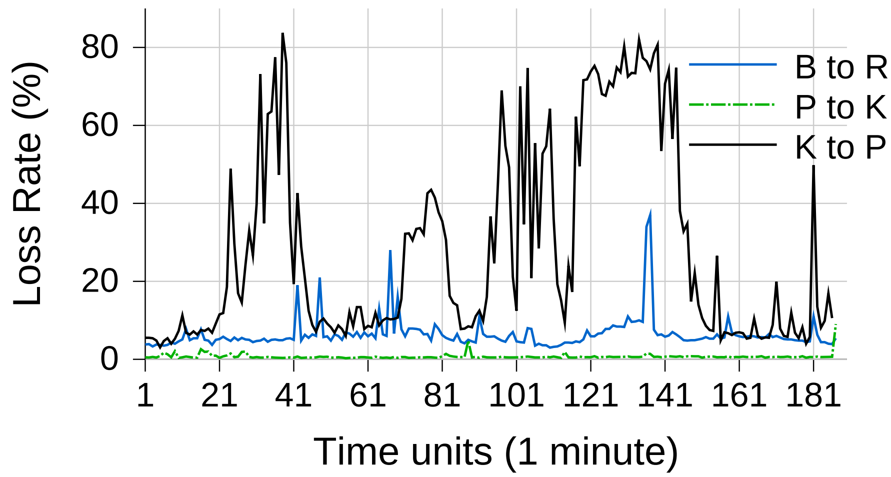


Figure 3.1: Packet loss variation over a period of about 3 hours. The loss rate was averaged over 30-second intervals for a 1440 byte UDP CBR flow of 1 Mbps at 802.11b datarate of 11Mbps

variability of loss rate; and 2) the loss rate can be highly asymmetric across a link, and 3) bursts vary in magnitude as well as duration. For example, on the K to P link, loss bursts ranged in magnitude from 15–80% and the duration of bursts also varied from a transient high burst to a long burst lasting over 25–30 minutes. In contrast, the reverse path (P to K) had almost 0% loss for the entire duration. In addition to the high variability of the loss rate, there is also a residual loss that is always present and remains constant over long time periods. This residual loss ranges between 0–10% and varies with each link.

Although Figure 3.1 shows only two links in our testbed, the above behavior is characteristic of all our urban links. In contrast, our rural links consistently show loss rates close to zero with a maximum of less than 2%. We explore these differences further and point out that many WiLD links have one end in an urban area. In addition, the losses shown here are only those due to the channel; the 802.11 protocol itself also induces losses.

3.2 Overview and Methodology

In this chapter we study the characteristics of 802.11 long distance links. Our study is based on a real-world WiLD network deployment consisting of 6 links with lengths varying from 2–20 km. Unlike existing WiLD deployments [40], our testbed includes both rural and urban links. In addition to the real deployment, we perform detailed experiments using a wireless channel emulator, which enables repeatable and controlled experiments over a wide range of link distances.

3.2.1 Contributions

We study the following characteristics of 802.11 WiLD links:

- **802.11 Protocol Shortcomings:** The stock 802.11 MAC protocol is ill-suited for WiLD links due to the breakdown of CSMA over long distances and propagation delays (section 3.3). Here, we pinpoint some fundamental shortcomings of the 802.11 MAC protocol.

- **Channel loss characterization:** We analyze three well known causes for channel losses in wireless environments, namely, *external WiFi interference*, *non-WiFi interference* and *multipath interference* (section 3.4). Among these, we show that external WiFi interference is the most significant source of packet losses in WiLD environments and the effect of multipath and non-WiFi interference is not significant. This is in contrast to the results of the Roofnet mesh network [20] where the authors observed multipath to be the most significant source of packet loss.

- **Loss variability analysis:** We classify the loss patterns over time into two basic categories: *bursts* and *residual* loss. We further classify bursts into short and long bursts. We make three important observations (section 3.5): (a) Although the burst arrival patterns can be approximately modeled based on a Poisson process, the duration and magnitude of a burst are harder to predict; (b) The residual loss characteristics over certain links are stationary, while some others exhibit non-stationary behavior even over daily timescales; (c) The loss variability observed in our urban links significantly differs from that under rural settings as observed in prior work [40].

- **Impact on TCP performance:** We measure the effect that protocol-induced losses and channel losses have on end-to-end TCP performance, and how link parameters (*e.g.,* maximum number of packet retransmissions) affect the TCP throughput (section 3.6).

- **Loss remedies:** Having identified external WiFi interference as the primary source of losses in WiLD links, we discuss three potential remedies to mitigate these losses (section 3.7) (a) 802.11 frequency channel adaptation; (b) 802.11 PHY datarate adaptation, and (c) adaptive FEC. We evaluate the effectiveness of each of these remedies.

The focus of our link characterization study is significantly different from other wireless-based loss measurement studies [20, 104]. The work done by Raman et al. [40] is the only other measurement-based study of WiLD deployments of which we are aware. However, the two studies are orthogonal: we focus on loss variability characterization, determining the impact of different sources of losses and remedies for loss alleviation, their work focused more on performance analysis of 802.11 network at various layers in the network stack and the effect of other parameters (weather, SNR, payload, datarate) on loss. Our work also differs from mesh networks like Roofnet [20] in that WiLD networks, as we show, have very different loss characteristics, with loss much more due to external interference than multipath effects.

3.2.2 Experimental Methodology

We perform our packet loss characterization measurements on a WiLD network testbed comprising of links in both rural and urban environments. Table 3.1 summarizes some of the urban and rural links in our deployments. The links range from 2–20 km in length.

The two main characteristic of WiLD links that differentiate them from links in a multi-hop urban mesh deployment [20] are the longer distances and the use of high-gain directional antennas (24 dBi, 8 degree beam-width). The two endpoints of each link have direct line-of-sight (LOS). In multihop settings, nodes have one radio per fixed point-to-point link to each neighbor, which can independently operate on different channels.

Link	Distance (km)	Environ.	Antenna height(m)
K-P	20	Urban	50
B-R	8	Urban	30
M-P	2	Urban	40
T-A	11	Rural	20
T-S	13	Rural	25
W-N	15	Rural	20

Table 3.1: List of our urban and rural WiLD testbed links.

In addition to the testbed, we also use a wireless channel emulator (Spirent

5500 [11]) to study each source of packet loss in isolation. The emulator allows us to place the two ends of the link in separate RF-isolated boxes (80dB isolation) and then emulate in real time the RF channel between them. The Spirent 5500 accurately emulates radio channel characteristics with channel loss, fast and slow fading and delay spreads. This enables us to emulate links of any length or loss profile with repeatable results. We perform tests by connecting the channel emulator to the same radios used in our WiLD deployments.

We use Atheros 802.11 a/b/g radios for all our experiments. The wireless nodes are 266 MHz x86 Geode single-board computers running Linux 2.4.26. We use *iperf* to measure throughput. All our results are based on CBR UDP traffic streams. Unless otherwise stated, for all our experiments we turn off MAC-layer ACKs and set the maximum retries limit to zero. This allows us to measure the real channel loss rate in absence of any MAC-layer acknowledgments and retries.

We instrument the stock Atheros *madwifi* driver to log fine-grained information for each frame received and transmitted. In addition to capturing all the frames on the link, we also capture and log frames being transmitted by external WiFi sources. This is achieved by creating a virtual network interface set in "monitor mode" on the same channel as the primary interface. This technique is equivalent to using two physical network interfaces, one being the primary and the other a passive monitor. To summarize, we collect the following information for every frame: complete 802.11 MAC header and IP payload, received signal strength, PHY layer transmit datarate, timestamp, PHY and CRC errors, and the noise floor immediately after the frame is received. We also modify the Atheros driver to pass up frames with CRC and PHY errors.

Using the WiLD testbed and the channel emulator, we explore two categories of loss: *channel losses* induced by the wireless channel and *protocol-induced losses* by the 802.11 MAC protocol. Specifically, for channel-induced losses we investigate: a) External WiFi interference, b) External non-WiFi interference and c) multipath interference. The absence of any mobility of the end points and high SNR eliminate fading and path loss as possible sources of packet loss. For 802.11 protocol induced losses, we investigate: a) timeouts due to propagation delay, and b) the breakdown of CSMA over long distances.

3.3 802.11 Protocol Shortcomings

In this section, we study the three main limitations of the 802.11 protocol: the inefficient link-layer recovery mechanism, collisions in long-distance links, and inter-link interference. These limitations make 802.11 ill-suited even in the case of a single WiLD link. Based on extensive experiments, we also show that modifying the driver-level parameters of 802.11 is insufficient to achieve good performance.

Inefficient Link-Layer Recovery

The 802.11 MAC uses a simple stop-and-wait protocol, with each packet independently acknowledged. Upon successfully receiving a packet, the receiver node is required to send an acknowledgment within a tight time bound (ACKTimeout), or the sender has to retransmit. This mechanism has two drawbacks:

- As the link distance increases, propagation delay increases as well, and the sender waits for a longer time for the ACK to return. This decreases channel utilization.

- If the time it takes for the ACK to return exceeds the ACKTimeout parameter, the sender will retransmit unnecessarily and waste bandwidth.

We illustrate these problems by performing experiments using the wireless channel emulator. To emulate long distances, we configure the emulator to introduce a delay to emulate links ranging from 0–200 km.

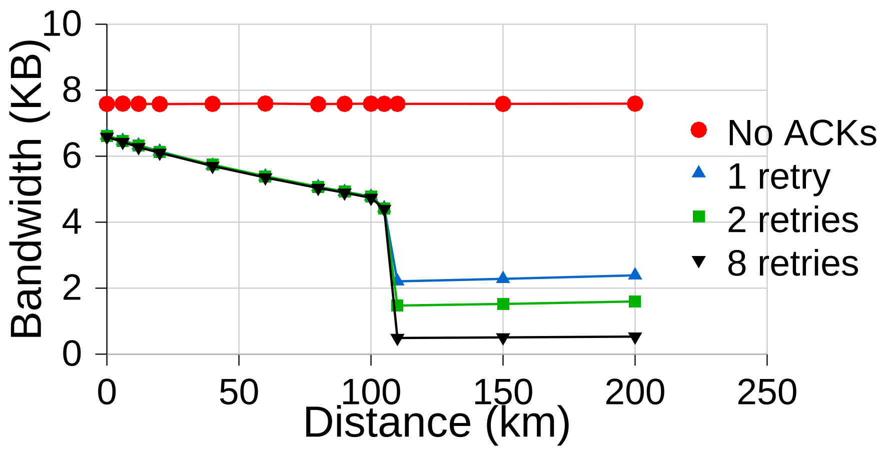


Figure 3.2: Unidirectional UDP throughput

Figure 3.2 shows the performance of the 802.11 stop-and-wait link recovery mechanism over increasing link distances. With the MAC-layer ACKs turned off (No ACKs), we achieve a throughput of 7.6 Mbps at the PHY layer data rate of 11 Mbps. When MAC ACKs are enabled, we adjust the ACK timeout as the distance increases. In this case, the sender waits for an ACK after each transmission, and we observe decreasing channel utilization as the propagation delay increases. At 110 km, the propagation delay exceeds the maximum ACK timeout (746μs for Atheros, but smaller and fixed for Prism 2.5 chipsets) and the sender always times out before the ACKs can arrive. We notice a sharp decrease in received throughput, as the sender retries to send the packet repeatedly (even though the packets were most likely received), until the maximum number of retries is reached (this happens because, if an ACK is late, it is ignored). This causes the received throughput to stabilize at $BW_{110km}/(no\_of\_retries + 1)$.

3.3.1 Collisions on long-distance links

The 802.11 protocol uses a CSMA/CA channel-access mechanism, in which nodes listen to the medium for a specified time period (DIFS) before transmitting a packet, thus ensuring that the channel is idle before transmission. This translates to a maximum allowable distance at which collisions can be avoided of about 15km for 802.11b (DIFS is 50μs), 10.2 km for 802.11a and 8.4 km for 802.11g. For longer links it is possible for a node to start transmitting a packet unaware of another packet transmission at the other end. As the propagation delay increases, this probability of loss due to collisions increases.

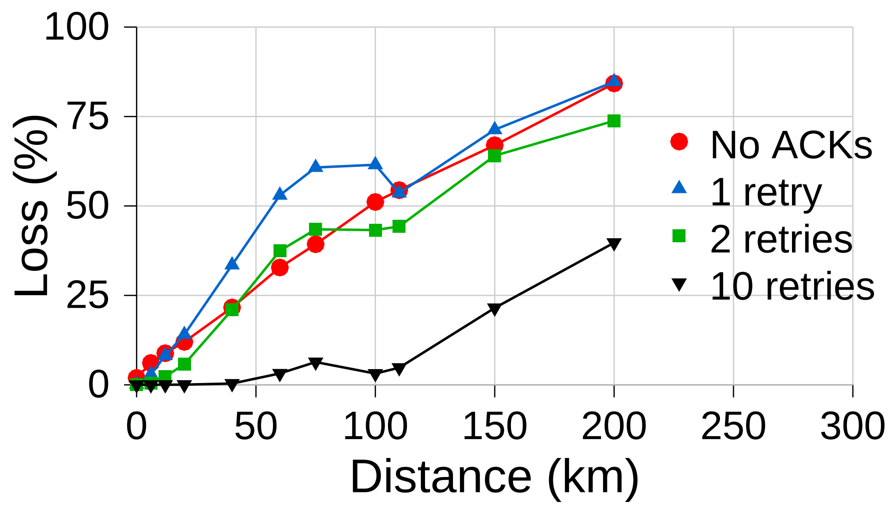


Figure 3.3: Bidirectional UDP loss

We illustrate the above-mentioned effect by using a simple experiment: we send

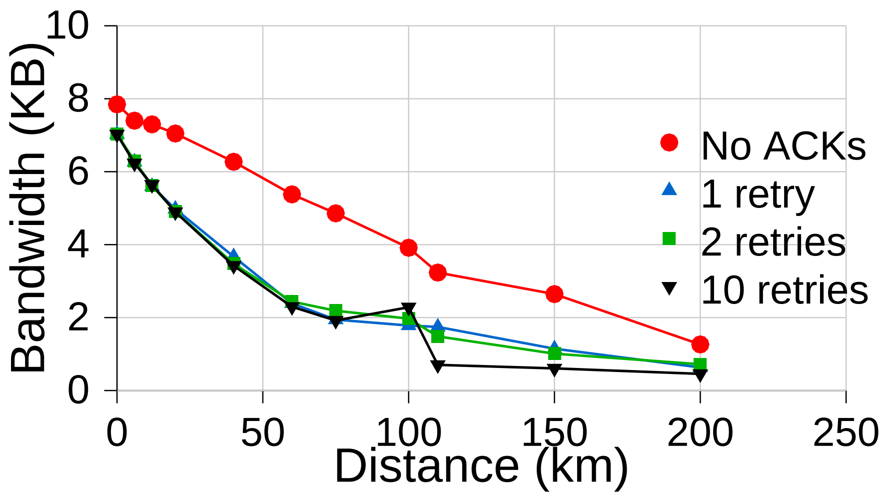


Figure 3.4: Bidirectional UDP throughput

bidirectional UDP traffic at the maximum possible sending rate on the emulated link and measure the percentage of packets successfully received at each end. Figure 3.3 shows how the packet loss rate increases with distance. Figure 3.4 shows the sum of the throughputs achieved at both ends for bidirectional UDP traffic as we increase the distance for a link. Note that there are no losses due to attenuation or outside interference in this controlled experiment; all of the losses are due to collisions.

A possible solution to this issue would be to increase the DIFS time interval in order to permit longer propagation delays. However, just as in the case of the ACK timeout, this approach would decrease channel utilization substantially for longer links. Furthermore, we are not aware of any 802.11 chipsets that allow the DIFS interval to be configured.

3.3.2 Multiple Link Interference

Another important source of errors is the interference between adjacent 802.11 links operating in the same channel or in overlapping channels. Although interference between adjacent links can be avoided by using non-overlapping channels, there are numerous reasons that make it advantageous to operate adjacent links on the same frequency channel, as described by Raman *et al.* [93]. Moreover, there are WiLD topologies such as the Akshaya network [21] where different channels cannot be allocated to all the pairs of adjacent links, given the high connectivity degree of several nodes.

Inter-link interference occurs because the high-power radios create a strong RF field in the vicinity of the radio, enough to interfere with the receptions at nearby radios. Directional antennas also have sufficiently high gain (4–8 dBi) side lobes [32] in addition to the main lobes.

The first type of problem occurs when multiple radios attached to the same node attempt to transmit at the same time. As soon as one radio starts transmitting after sensing the carrier to be idle, all other radios in the vicinity find the carrier to be busy and backoff. This is desirable in a broadcast network to avoid collisions between two senders at any receiver node. However, in our network where each of these radios transmits over point-to-point long distance links to independent receivers, this backoff leads to suboptimal throughput. A second problem occurs when packets being received at one link collide with packets simultaneously transmitted on some other link on the same node. The signal strength of packets transmitted locally overwhelms any packet reception on other local radios.

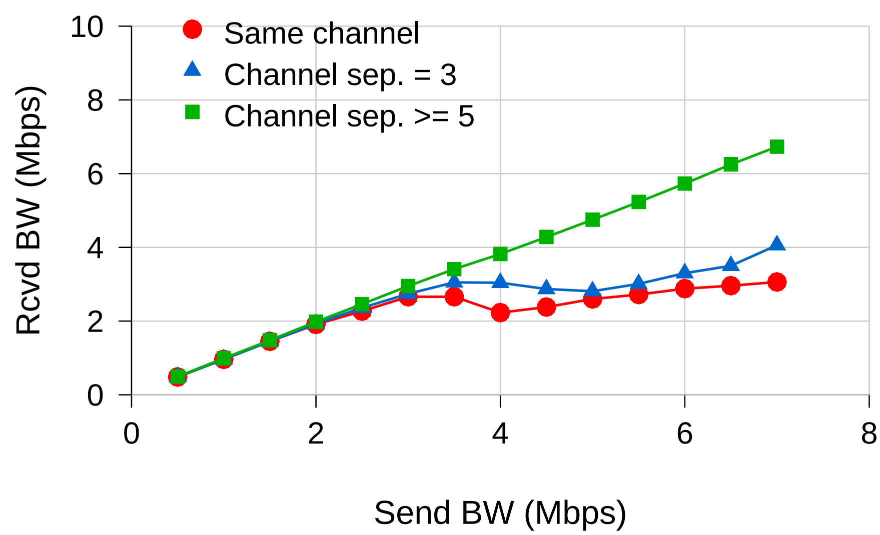


Figure 3.5: UDP throughput

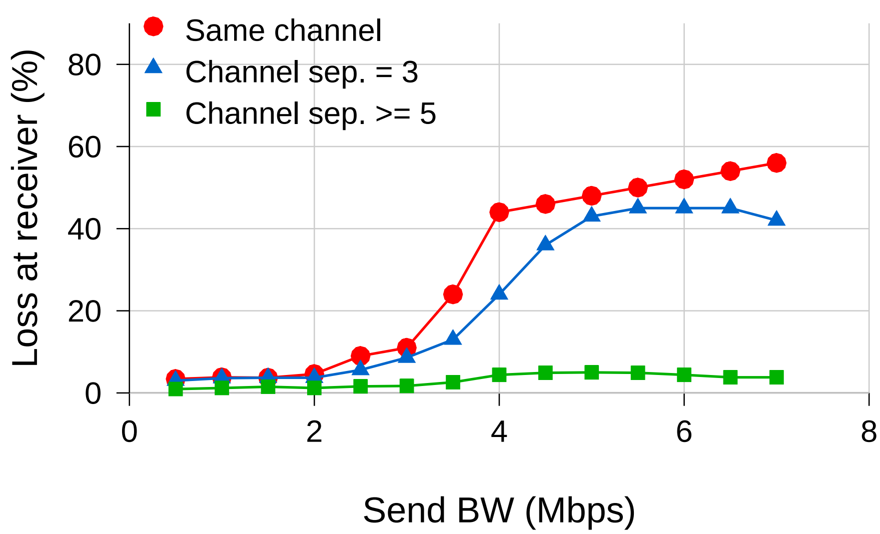


Figure 3.6: UDP loss at receiver

In order to illustrate and quantify the magnitude of these effects, we perform experiments on the real-world setup featuring two adjacent links, K – P and P – M, using constant rate UDP traffic, and configuring the radios to operate in 802.11b mode at a bitrate of 11Mbps.

First, we simultaneously transmit UDP traffic from P to both K and M. We gradually increase the packet sending rate and vary the channel separation between the links. We notice that the cummulative send throughput on the two links can be as high as 14.20 Mbps when operating on non-overlapping channels (separation ≥ 4), but that this throughput drops to only 7.88 Mbps when on the same channel. We continue by performing a similar experiment, in which UDP traffic from node M to node K is relayed through P. Figure 3.5 illustrates the achieved end-to-end throughput for different values of the channel spacing between the adjacent links, while Figure 3.6 presents the end-to-end packet loss rate for the same experiment. We find that indeed interference reduces the utilization of the individual links and significantly increases the link loss rate (even in the case of partially overlapping channels). For links operating on the same channel, loss rates can be as high as 60%, and even for a channel spacing of 4 the loss rate due to interference can exceed 40%.

Therefore, the maximum channel diversity that one can simultaneously use at a single node in the case of 802.11(b) is restricted to 3 (channels 1,6,11) which may not be sufficient for many WiLD networks. This motivates the need for an interference-avoiding mechanism that allows the efficient operation of same-channel adjacent links, as proposed later in chapter 4.

3.3.3 Implications

In the following we summarize the implications of our findings related to the inefficiencies of the 802.11 protocol when used in the context of long-distance, directional links:

- Per packet acknowledgments and a stop-and-wait flow-control are extremely inefficient in long distance links. These problems can be alleviated by disabling the per-packet acknowledgments and switching to a more efficient flow-control mechanism that can permit many packets to be in-flight simultaneously, and therefore achieve high throughput in spite of the large propagation delays.

- Due to long propagation delays, the propability of collisions between packets sent by link ends is very high. This indicates that more appropriate medium access policies, that synchronize between link endpoints and do not simply rely on carrier sensing, are required.

- Inter-link interference between adjacent links operating on overlapping channels is a significant problem. This could be alleviated by employing mechanisms that synchronize transmissions accross multiple adjacent links, avoiding inter-link interference.

We design and implement mechanisms addressing all the issues above in our system (WiLDNet) presented in chapter 4.

3.4 Channel-Induced Losses

In this section we investigate the packet losses due to the inherent nature of the wireless channel in long-distance, directional links. These losses are independent of the medium-access protocol used by our links.

3.4.1 External WiFi Interference

We begin by investigating external WiFi interference as a potential source of packet loss in WiLD links. Any WiFi traffic that is not a part of the primary WiLD link is categorized as external WiFi interference. Based on the measurements performed on our WiLD testbed and the wireless channel emulator, we show three key results:

- In the presence of external WiFi interference, the loss rate is strongly correlated with the amount of external traffic received on the same and adjacent channels. In contrast, due to the omni-directional antennas used in the Roofnet deployment [36], no such strong correlation was observed.
- Packet loss due to external WiFi interference is far more significant in WiLD deployments than local mesh networks.
- The loss due to external WiFi interference depends on the relative power level between the primary and external traffic, their channel separation, and the rate of external interference.

Correlation of loss rate and external WiFi traffic: To measure the effect of external WiFi traffic interference on our WiLD links we create a virtual interface in monitor mode as described in Section 3.2.2. A CBR traffic source of 1 Mbps is used to generate traffic on the WiLD link and the loss rate is averaged every minute.

Figure 3.7 shows the loss rate across all (rural and urban) our WiLD links. We observe that the loss rate of the urban links vary across a wide range (4–70%). In contrast, all the rural WiLD links have a very small loss rate. The maximum loss rate observed in

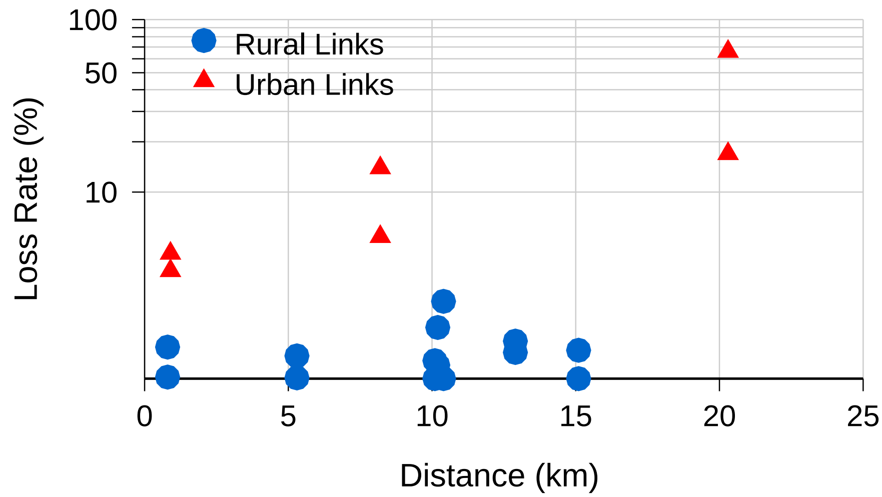


Figure 3.7: Scatter plot of loss rates observed in links deployed in urban and rural areas

all our rural WiLD links was 2%.

To study this contrast between the rural and urban links, we collected detailed packet level MAC traces. By parsing the MAC header source and destination fields, we are able to count the number of frames received from external WiFi sources. In the traces collected over all our rural links we see negligible external WiFi traffic. However, significant amount of external WiFi traffic was captured from the traces collected in the urban WiLD deployment.

Figure 3.8 shows a scatter plot between the loss rate and the absolute number of external WiFi traffic frames received on an urban link (K $\rightarrow$ P) for a period of 6 hours. The figure shows that a subset of the loss rate samples are strongly correlated with the external traffic. For the other subset of the samples, the loss rate increases even when there is no significant increase in WiFi traffic on the same channel.

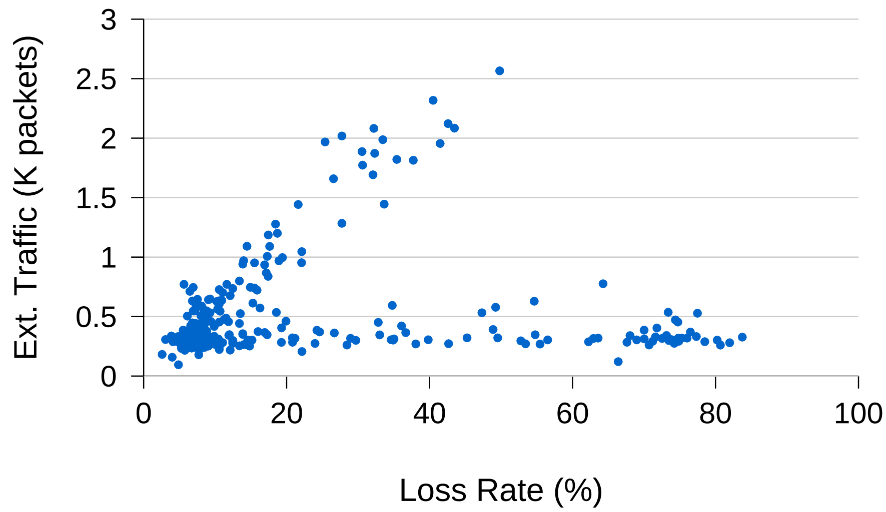


Figure 3.8: Loss rate vs. external/ traffic observed on WiLD link

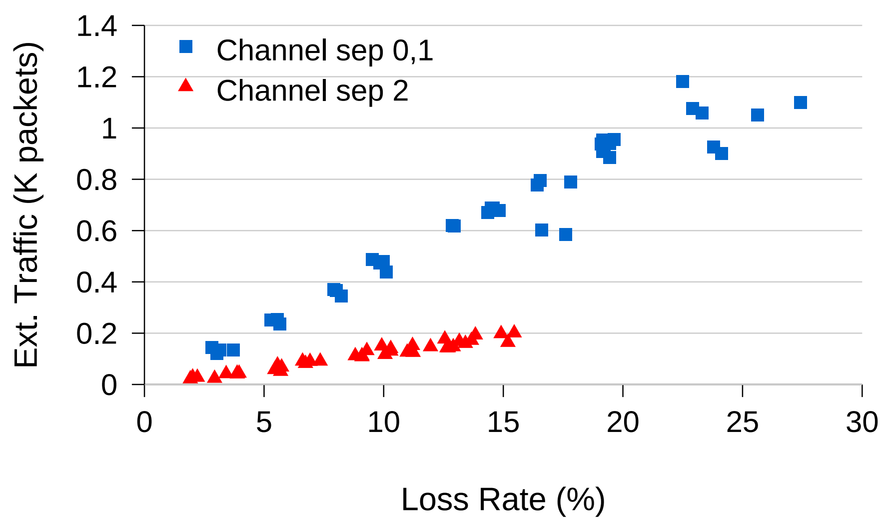


Figure 3.9: Loss rate vs. external traffic observed in wireless emulator

To investigate this further, we perform a controlled experiment using the wireless channel emulator. To model interference from an external traffic source, along with the primary link traffic we introduce a controlled interference source at the receiver. The traffic rate of the interference source was varied from 0.1 to 1 Mbps and the traffic rate on the primary link was kept fixed at 5 Mbps. Figure 3.9 shows a scatter plot of the loss rate and the total number of frames received from the external interference source. From the graph, we observe that for a given loss rate, the amount of external traffic captured by the monitor device depends on the channel separation of the primary and interference source.

The above observed trend is the same as that in Figure 3.8. At a channel separation of 0 and 1, the receiver can receive both the primary link traffic as well as the frames from the interference source. Hence, the loss rate is directly correlated with the amount of external WiFi traffic captured by the monitor interface. At a channel separation of 2, the receiver is not able to receive the frames from the external interference source. However, the signal spillage of the interference source in the primary channel is sufficient to cause frame corruption. From the traces we observed that almost 100% of the lost frames contained CRC errors.

Effect of hidden terminals in WiLD networks: Unlike WiLD deployments, where we have observed significant correlation between loss rate and external interference, it has been observed that there is no significant correlation in outdoor mesh-network deployments (Roofnet [36]). The primary difference between WiLD and mesh-network links is that WiLD links are longer (10–100 km) and use high gain (30dBi) directional antennas (≈ 8 degree beam width), while mesh-network links are shorter (0.5–2 Km) and use low gain (8dBi) omni-directional antennae. In a mesh-network deployment, an external interference source (I) that is within range of the omni-directional transmitter (Tx) would be able

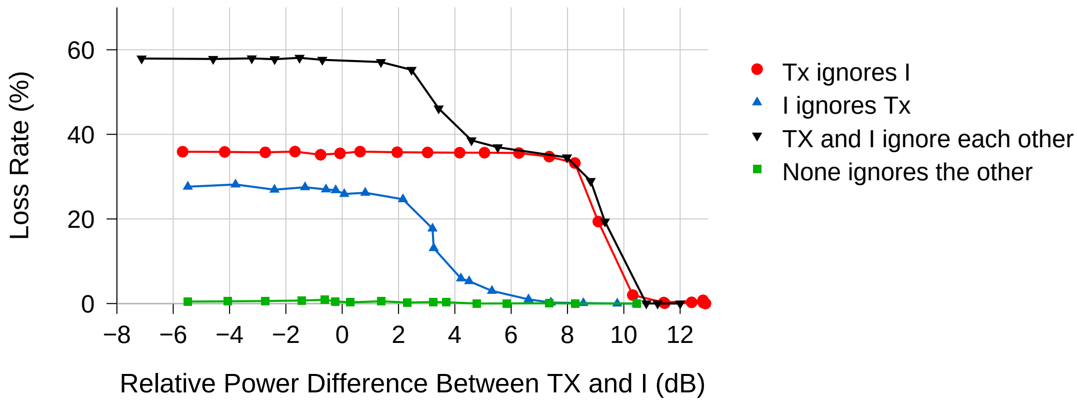


Figure 3.10: Losses due to different hidden terminal effects. Both main and interfering traffic is 1440 byte UDP CBR packets at 11Mbps PHY datarate of 802.11b.

to sense the medium to be free and backoff its transmission. However in WiLD links, the transmissions are highly directional and the propagation delays are higher. These factors in combination exacerbate the *hidden terminal* problem in WiLD networks. The transmitter and the interference source can erroneously sense the medium to be free leading to collisions whenever they are out of range of each other (because of the directional nature of transmission) or when they cannot sense the medium to be busy in time to backoff (because of the longer propagation delays).

Collisions at the receiver can manifest in two different situations: a) When I doesn't hear Tx, and initiates a transmission when the medium is busy with an ongoing packet transmission from Tx, and b) When Tx doesn't hear I, and causes a collision by interrupting an ongoing packet transmission from I.

To isolate the above two cases and measure the performance degradation due to each case, we perform controlled experiments using two WiFi links. We simultaneously send packets from both Tx (512 Kbps traffic) and I (3Mbps), and measure the packet loss rate on the primary link ($Tx \rightarrow Rx$) with MAC-layer ACKs disabled.

To create the situation where Tx cannot hear I, we disable the Clear Channel Assessment (CCA) at Tx, which simply causes Tx to ignore I. We also eliminate propagation delay between Tx and I so that I's CCA works perfectly. We reverse the operations to create the situation in which I cannot hear Tx, but Tx hears I perfectly.

We then run four experiments, reflecting the losses in four situations: when Tx can't hear I, when I can't hear Tx, when neither can hear each other (representative of cases in WiLD networks), and when both Tx and I hear each other (representative of most cases in urban mesh networks).

Figure 3.10 shows the loss rate for each of the above four cases. In the case where I ignores Tx, to overcome the interferer completely (achieve 0% loss), packet transmissions from the Tx have to be 7dB stronger than the interfering transmissions. This threshold, at which the primary link is loss free, is much higher (12dB) in the case where Tx ignores I. We explain this assymetry in the following.

When Rx begins receiving a packet P_I sent by the interferer, Rx's radio calibrates the channel parameters based on the packet preamble, and then continues with receiving the packet data, without searching for another packet preamble until P_I is fully received. If a subsequent packet P_{Tx} is sent by Tx, this packet can be properly received by Rx only if its energy is large enough to determine the receiver to drop the receival of P_I. Moreover, Rx needs to identiy P_{Tx}'s preamble quickly, before this preamble is over. This can happen only if the signal received from Tx is much stonger (at least by 12dB) than the signal received from I. On the other hand, if Tx is the first one to transmit a packet P_{Tx}, it is easier for Rx

to successfully finish the receival of this packet. Even if the interferer sends a subsequent packet P_I, the receival of P_{Tx} is successful as long as the signal received from Tx remains 7dB stronger than the interference.

When neither of Tx and I can hear each other, both the above two types of collisions are possible. Hence the loss rate is the sum of the losses generated by the above two types of collisions. However, when both Tx and I are in range of each other, resembling a mesh-network, losses due to collisions are close to zero. In this case, CSMA ensures that the two transmitters, Tx and I, share the medium properly.

From the above experiment we conclude that the effect of hidden terminals, causing collisions at the receiver, are greatly exacerbated in WiLD networks compared to urban mesh networks.

Effect of relative power and rate of external interference: To study the effect of relative power and rate of the external WiFi traffic on the loss of the primary link, we perform two experiments using the wireless channel emulator.

In the first experiment, we fix the relative power between the interference source and primary WiLD link, and vary the rate of the external interference source. The received signal strength of the interfering source was approximately 6dB higher than the primary link traffic. From Figure 3.11 we observe that for channel separations of 0, 1 and 2, the loss rate increases as the rate of the external interference increases. However, beyond a channel separation of 2, there is no significant interference from the external WiFi traffic source and the loss rate is almost zero.

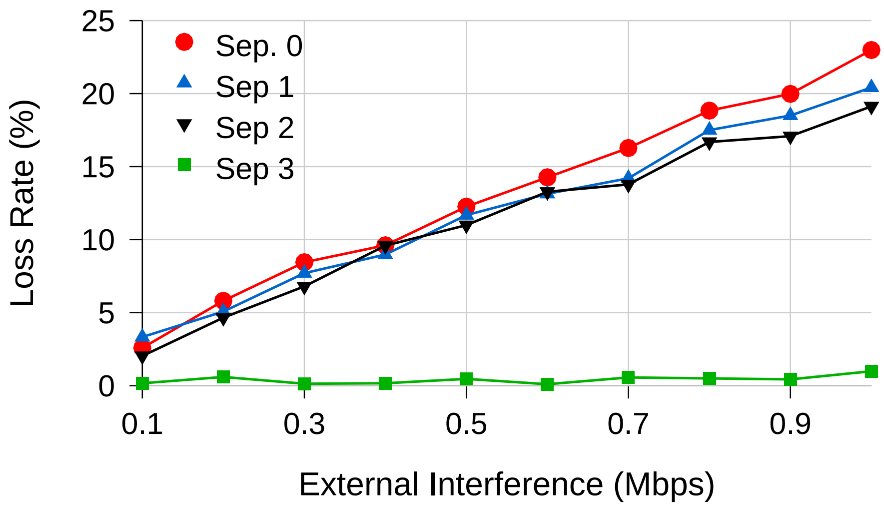


Figure 3.11: Loss rate at different channel separations: Varying interference rate

Figure 3.12 shows the variation in loss rate for different relative power levels of the interference source and WiLD link. In this experiment, we fix the power level of the primary WiLD link traffic and vary the relative power of the primary link to the power of the interferer from -15dBm to +13dBm. The primary link CBR traffic rate is fixed at 512 Kbps, while the interferer transmits at a rate of 3 Mbps.

We observe that when the interference source is on the same channel, even a 12dB lower signal could lead to packet loss on the primary WiLD link. When the interference source is significantly higher than the WiLD link (6dB and beyond), the loss rate is very high ($\geq$50%) for channel separations 0, 1 and 2. This corresponds to the situation where any collision results in the capture of the packet on the primary link. Beyond a channel separation of 2, we do not observe any loss on the primary link.

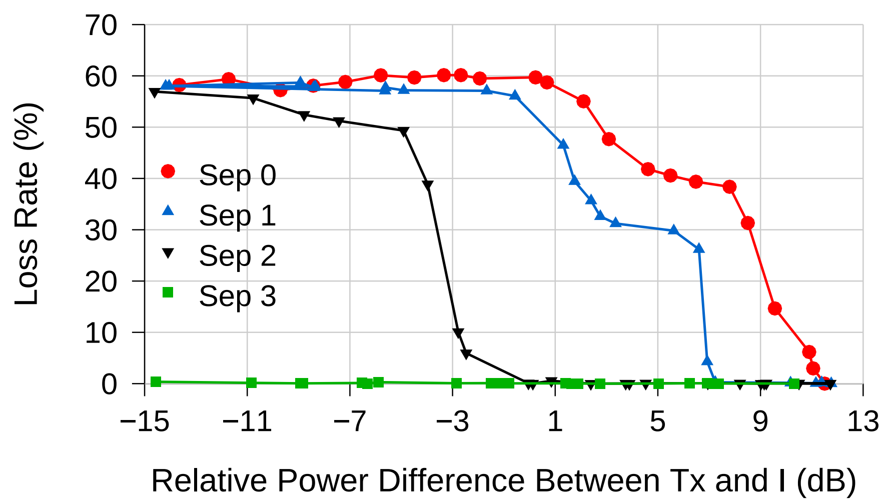


Figure 3.12: Loss rate at different channel separations as we vary the transmit power of the interference source

Implications: To summarize, we make the following observations related to the effects of external WiFi interference on WiLD links:

- We conclude that external WiFi interference is a significant source of packet loss in WiLD networks. Any deployment of WiLD networks in dense urban deployments has to take into account external WiFi interference.

- When calculating the link budget for urban links, it is beneficial to over-provision the received power. A high signal strength could potentially immunize the WiLD link from external WiFi traffic.

- MAC layer adaptation algorithms like adaptive channel switching, rate adaptation, and adaptive FEC could significantly reduce the loss due to external WiFi interference. In section 3.7 we evaluate each one of these as potential remedies to mitigate external

WiFi interference.

3.4.2 Non-WiFi Interference

The 802.11b communication protocol operates in the 2.4 GHz shared ISM band. This frequency band is shared with a host of other non-802.11 devices, such as microwave ovens, cordless phones, baby monitors, etc. Most of these non-802.11 devices do not follow a channel-access protocol, which could lead to a significant amount of interference caused by these devices.

In Sheth et al. [106], the authors were able to detect and measure non-WiFi interference by sampling the noise floor of the Atheros chipset. The authors observed that in presence of external non-WiFi noise, the noise floor linearly increases with increasing noise. We performed the same experiment on our WiLD testbed, where we sample the noise floor for every packet received. In presence of external noise causing high loss, we would expect the noise floor to be correlated with the loss rate. However, based on extensive measurements carried out on the urban links we do not see any correlation between noise floor and loss rate. In fact, the noise floor remains mostly constant with minor 1–2 dB variations.

In addition to the above test, we also check for wide-band non-WiFi noise. A wide-band noise source would cause interference across the entire 802.11 spectrum. Ideally, this can be measured using a spectrum analyzer and detecting a rise in power across the entire spectrum. However, using a spectrum analyzer is infeasible on the outdoor WiLD links. Thus, to detect wide band noise in our WiLD deployment we synchronize the two ends of a link to rotate across channel 1, 6 and 11 periodically. The sender generates 1 Mbps UDP CBR traffic on each channel and the receiver measures the loss rate on each channel.

In presence of any wide-band noise, we would expect to observe a correlation among loss rates across all three channels. However, based on long-term experiments performed on three urban links, we determined that there was no statistically significant correlation, and thus no significant broadband noise.

3.4.3 Multipath Interference

Multipath interference is a well known source of packet loss in WiFi networks [20, 46]. It occurs when a RF signal takes different paths from a source to a destination node. Hence, along with the primary line-of-sight signal, the receiver also receives multiple secondary reflections that distort the primary signal. Although it is difficult to measure the exact delay between the primary and secondary paths on our WiLD deployments using commodity off the shelf equipment, based on the experiments using the wireless channel emulator we conclude the following:

1. The order-of-magnitude lower delay spreads in WiLD deployments significantly reduce the interference due to multipath in WiLD deployments.

2. If WiLD links are deployed in dense urban deployments with non-line-of-sight, multipath interference could lead to significant loss at the higher 802.11b data rates of 5.5 and 11 Mbps.

Multipath interference in Roofnet and WiLD deployments: Based on link-level measurements performed in the Roofnet deployment [20], the authors conclude that multipath interference was a significant source of packet loss in the Roofnet deployment. However unlike urban 802.11 mesh deployments, multipath interference is significantly lower in WiLD network deployments due to the order-of-magnitude lower delay spreads. The two factors contributing to lower delay spreads in WiLD networks are the long distance between the two end hosts and the line-of-sight deployment of the nodes. The strong line-of-sight component in WiLD deployments ensures that the attenuation of the primary signal is only due to path loss, and most of the secondary paths are due to reflections from the ground. In comparison to our WiLD deployment, the Roofnet deployment has shorter links and non-LOS deployments. The median link length is 0.5 km and the longest link is 2.5 km, and links are rarely line-of-sight.

Table 3.13 shows the delay between the primary path and secondary path assuming the antenna is mounted at a height of 30 meters and reflection is only from the ground. The two delays are computed for a secondary path reflecting at the quarter point and at the mid-way point between the transmitter and the receiver. Although multipath reflections arriving at the receiver are not constrained to these distances, the table provides the relative difference in delay spreads observed in the Roofnet deployment and WiLD deployment. As the length of the link increases, the primary and the secondary path travel almost the same distance, and hence the delay between the primary and secondary reflection reduces. As seen from the table, there is an order-of-magnitude difference between the delay in WiLD links and medium range Roofnet links. Aguayo et al. [20] also observed that the RAKE receiver is able to tolerate delay spreads upto 0.3–0.4 μsec.

To further validate that multipath interference is not a significant source of packet loss, we perform measurements over WiLD links deployed in rural environments. Our hypothesis was that most of the loss in our urban deployment was due to external WiFi

Dist. (km)	Delay spread (μsec)
0.5	(4.75, 3.59)
1.0	(2.4, 1.80)
2.0	(1.1, 0.90)
4.0	(0.6, 0.45)
8.0	(0.3, 0.22)
16.0	(0.15, 0.11)
32.0	(0.07, 0.06)
100.0	(0.02, 0.01)

Figure 3.13: Delays between a primary and secondary reflection

interference. Hence, in absence of external interference the WiLD links deployed in the rural areas should not have any loss. Figure 3.7 validates our hypothesis, which shows loss rates observed across three such rural links. From this figure we observe that the maximum loss rate observed was 1.7% with low variance.

Effect of non-line-of-sight dense urban deployments: To study the effect of multipath when WiLD links are deployed in absence of line-of-sight, we perform controlled experiments in the wireless channel emulator. Due to the lack of analytical models, we build an artificial model consisting of 12 reflected paths with the path delay increasing in steps of 0.18 μsec and the power exponentially decaying. Hence the maximum delay between the primary and the longest secondary path is 2.16μsec. Figure 3.14 shows the loss rate for payloads of size 768, 1024 and 1280 bytes and the four data rates of 802.11b. From the figure we observe that the lower data rates are resilient to multipath and the length of a frame does not affect the loss rate.

Looking closer at the packet traces collected at the receiver node, we observe that almost 100% of the errors are CRC errors. We find loss rates to be close to zero at 1

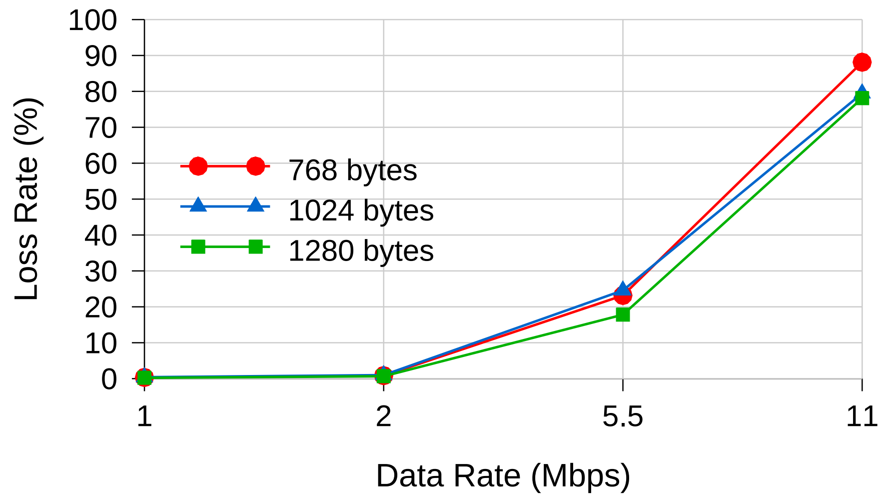


Figure 3.14: Multiple reflections at the receiver. Power is exponentially decaying and delay is increasing linearly in steps of 0.2 us

and 2Mbps, but significantly higher at 5.5Mbps and 11Mbps. This happens because the modulations used at lower datarates (BPSK for 1Mbps and QPSK for 2Mbps) are resilient to multipath, while CCK, used for 5.5Mbps and 11bps, is less resilient. In the following we present our results for 5.5 and 11Mbps. Figure 3.15 shows a histogram of the number of bytes corrupted as a percentage of total number of CRC error frames received. We observe that the distribution has a heavy tail and almost 90% of the CRC error frames have less than 10% of the bytes corrupted in the payload. This has implications on the recovery mechanisms in presence of multipath.

Implications: Our findings have the following implications:

- The higher data rates of 11 Mbps and 5.5 Mbps are much more sensitive to multipath interference as compared to the lower data rates of 1 Mbps and 2 Mbps. This has implication on the rate selection algorithm for applications that can trade-off band-

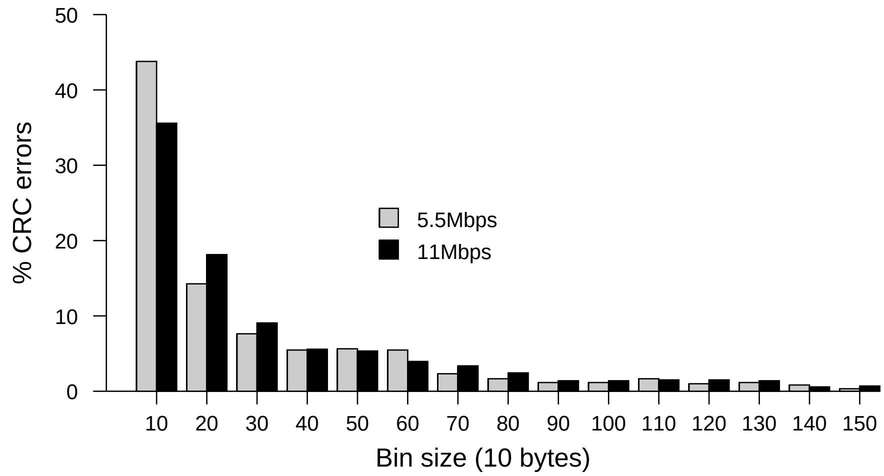


Figure 3.15: Distribution of number of bytes being corrupted.

width for a loss free channel. In presence of significant multipath interference, a rate selection algorithm for such applications could directly move from 11 Mbps to 1 or 2 Mbps instead of stepping down to 5.5 Mbps.

- From 3.15 we observe that 90% of the corrupted frames have less than 10% of bytes corrupted. Also from Figure 3.14 we notice that increasing the length of the payload does not affect the loss rate. Hence, an alternate approach to rate adaptation could be to divide the payload into smaller blocks and encode these blocks to add in sufficient redundancy to tolerate the CRC errors due to multipath interference.

3.5 Loss Variability

In this section, we analyze the variability of packet loss over time on the WiLD links. We first propose a simple mechanism we use to classify loss periods as either bursts or residual losses and then individually describe the loss characteristics for bursts and residual losses.

3.5.1 Burst-Residual Separation

We observe that all the links in our testbed exhibit a bi-modal loss variation over time where the loss-rate at any given time can be classified into into two categories: *bursts* and *residual* losses. While bursts refer to time-periods with sharp spikes in the loss rate, residual losses refer to the losses that constantly occur in the underlying channel over time. Unlike previous studies on WiLD links in rural environments [40], we observe a non-zero residual loss-rate in most of our links in urban environments.

To classify each time-period into either a bursty or residual loss-period, we determine a *demarcation region* for the loss distribution on a given link. We estimate parameters p_1 and p_2 ($> p_1$) such that a significant majority ($> 99\%$) of the loss samples fall in the regions $[0, p_1)$ and $(p_2, 1]$. All loss periods with the loss-rate in the range $[0, p_1)$ are classified residual and those in the range $(p_2, 1]$ are classified bursty. The remaining samples are considered transition phases. If adjacent loss periods of a transition period are bursty, then the transition phase is also classified as bursty.

3.5.2 Burst characteristics

To analyze burst characteristics, we need to measure the variability of three parameters associated with bursts: duration, arrival pattern and magnitude.

Burst duration and arrival: Based on the duration of bursts, one can classify a burst as either as a *short burst* or a *long burst*. Across our links, we observe a majority of the bursts to be short bursts that last for less than 0.3s. However, in certain links, especially those in urban environments, we observe a continuous burst period that can last up to 70s. The characteristic arrival pattern that we observed for long bursts is that a single long burst is followed by a string of other long bursts separated by short time-periods (in the order of a few seconds). Overall, the entire string of long bursts that occur together in time lasts for several minutes representing time periods where the underlying channel experiences very high loss rates. Based on the results in Section 3.4.1, we conclude that these elongated bursts occur due to interference from external WiFi traffic sources.

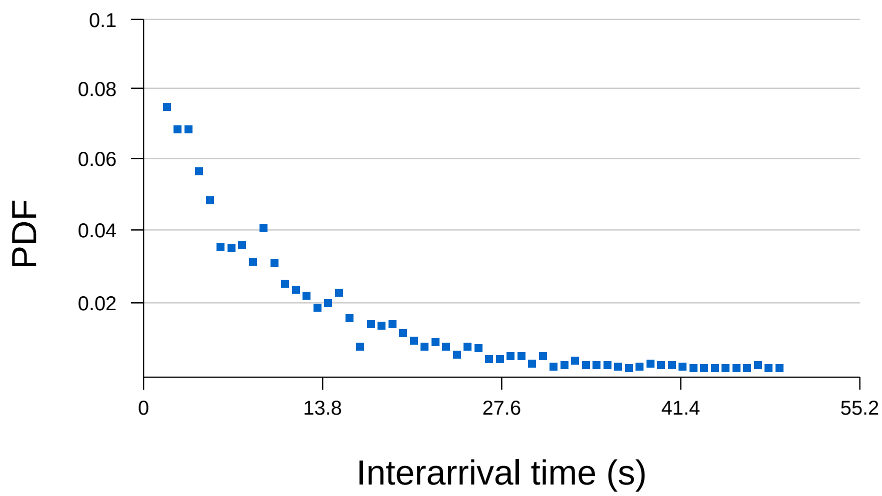


Figure 3.16: Prob. distribution of inter-arrival time of bursts

We next focus on the arrival pattern of short bursts. Figure 3.16 shows the distribution of inter-arrival times between bursts with short durations for the R-B link in our testbed. For this link, we observe that the underlying distribution of inter arrival time resembles an exponential distribution with a mean inter-arrival time of 15s. In addition, we observe that the inter-arrival time distribution is stationary across various time-periods.

These observations suggest that the underlying arrival process can potentially be modeled based on a Poisson arrival process. We observe a similar behavior across all the links in our testbed.

Burst-loss magnitude: We found burst magnitudes to be very hard to predict. For both short spikes and long-duration bursts, the loss-rate varied across the entire spectrum between 10% and 60%. Even within a single burst, we observed the loss-rate across episodes to fluctuate rapidly. Given that our links operate in static environments, such wild fluctuations in very short periods appear to be triggered due to external WiFi interference as opposed to multi-path fading channel conditions.

3.5.3 Residual loss characteristics

Every link in an urban environment in our testbed exhibits a non-trivial residual loss rate where packet losses occur at regular intervals as opposed to bursts. The residual loss rate varies between 1% and 10% in our urban links in the testbed. However, residual loss-rates are negligible in our rural links. Based on analyzing the loss distributions over different timescales for different links, we make two observations. First, except for one specific link (K-P), we observed that the residual loss distribution is stationary over hourly time scales while on the K-P links, the distribution is time-varying. Second, we observe that the residual loss rate on any link remains roughly constant over a few minutes even in the presence in short bursts during such periods.

3.5.4 Implications

In summary, we make three observations. First, we can classify the loss sample at any time period into three categories: short burst, long burst or residual. Second, while the arrival of short bursts can be approximately modeled based on a Poisson arrival process, the arrival of long bursts are highly correlated in time and not memory-less. Finally, unlike rural links which exhibit negligible residual losses, we observe a non-negligible residual loss-rate in urban environments.

3.6 Impact on TCP Performance

Taken together, the protocol shortcomings of 802.11 and channel induced losses significantly lower end-to-end TCP performance. The use of stop-and-wait over long distances reduces channel utilization. In addition, we see correlated bursty collision losses due to interference from unsynchronized transmissions (over both single-link and multi-hop scenarios) as well as from external WiFi sources. Under these conditions, TCP flows often timeout resulting in very poor performance. The only configurable parameter in the driver is the number of packet retries. Setting a higher value on the number of retries decreases the loss rate, but at the cost of lower throughput resulting from lower channel utilization.

To better understand this trade-off, we measure the aggregate throughput of TCP flows in both directions on an emulated link while varying distance and introducing a channel packet loss rate of 10%. Figure 3.17 presents the aggregate TCP throughput with various number of MAC retries of the standard 802.11 MAC. Due to increased collisions and larger ACK turnaround times, throughput degrades gradually with increasing distances.

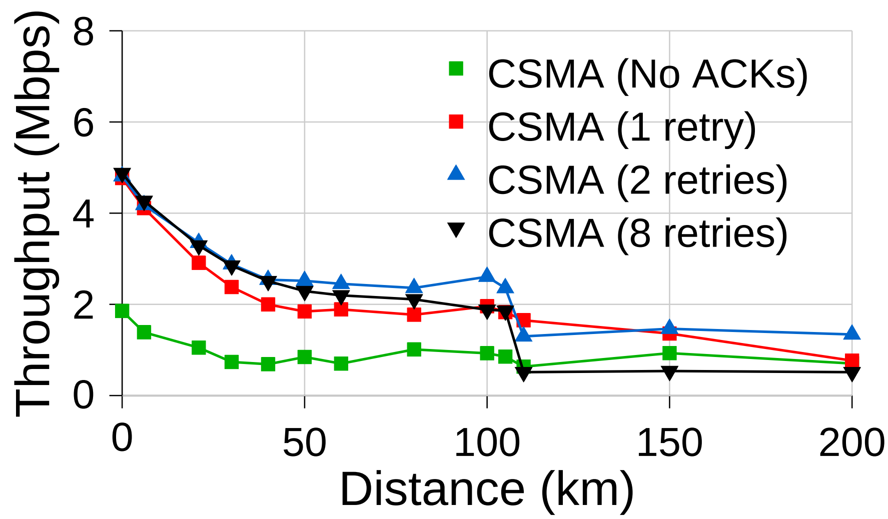


Figure 3.17: Cumulative throughput for TCP in both directions simultaneously over standard CSMA with 10% channel loss on emulated link. Traffic is 802.11b at PHY layer datarate of 11Mbps.

3.7 Remedies

Having identified protocol inadequacy and external WiFi interfernce as the main sources of packet loss in WiLD networks, we outline the potential remedies to mitigate external WiFi interference.

3.7.1 Remedies for protocol inadequacies

We defer most of the discussion of remedies for protocol inadequacies for the next chapter, where we redesign the medium access policy to address all the shortcomings identified here. At a high level, these solutions are:

TDMA based WiLD MAC protocol: As previously discussed, an un-synchronized channel-access mechanism relying solely on carrier sensing can cause severe collisions even for plain long distance point-to-point links, and is prone to inter-link interference. This necessitates a MAC protocol that synchronizes the transmissions from both the end points of the link, as well as in between adjacent links. One way to achieve this is by using a TDMA-style MAC approach, and we discuss the details of such a solution in the following chapter.

Sliding-window flow-control and adaptive link recovery: An alternate approach that mitigates the under-utilization of the medium due to the large timeouts and propagation delay is to relax the constraint of having only a single un-acknowledged frame. We propose a sliding-window based flow-control approach, in which the receiver acknowledges a set of frames at once (bulk ACKs). This can be implemented in combination with a TDMA MAC, and arrange for all the frames sent in one TDMA slot to be acknowledged together. The MAC protocol for WiLDNet (next chapter) combines TDMA based slot allocation with adaptive link loss recovery using bulk ACKs.

3.7.2 Remedies for external interference losses

We continue by evaluating adaptive frequency selection, rate adaptation and adaptive forward error correction (FEC) algorithms as potential remedies for losses due to external interference. For each, we simulate the adaptation algorithms and measure the improvements gained for real loss traces from our testbed and experiments performed on the wireless channel emulator.

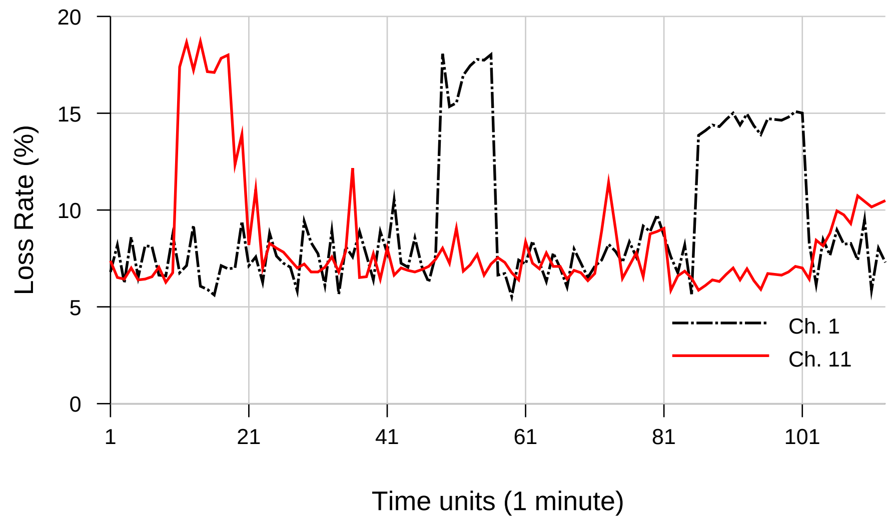


Figure 3.18: Loss variation over time across channels 1 and 11; loss rate averaged every 1 minute.

3.7.3 Frequency Channel Adaptation

A simple solution to mitigate external WiFi interference could be to select an alternate less congested channel and switch to that channel. To motivate this simple technique we perform a channel switching experiment on our WiLD deployment on the K-P link. The source and destination switch between channel 1 and 11 synchronously every 30 seconds. Figure 3.18 shows the variability of loss rate across the two channels for a period of about 2 hours. We can observe that both channel 1 and 11 show bursts that stretch upto a few minutes. It is important to note that by averaging the loss rate over 30 seconds we are not capturing the transient changes in the channel conditions.

Given the above loss trace across the two channels, Table 3.2 compares different channel switching algorithms by the achieved loss rate and the count of channel switches

	Loss	No
No adapt	(9.2, 8.3)	0
Lowest rate	6.8	40
Oracle (5%)	7.01	26
Change $\geq$ 10%	7.76	8

Table 3.2: Channel switching algorithms for the trace (loss rate and no. of switches)

required. In the base case (No adapt), where the channel is fixed at either channel 1 or 11, the average loss rate across the entire trace is either 9.2 or 8.3%. If the receiver has complete knowledge of the loss rate on both channels 1 and 11 at the beginning of a time interval (Oracle), then switching to the least lossy channel at any given time achieves the lowest loss rate (at 6.8%); but this comes at a cost of frequent switches of the channel. Adding a small hysteresis of 5% (Oracle 5%) for channel switching reduces the number of switches from 40 to 26 without increasing the average loss rate significantly. In absence of knowledge of loss rates on other channels, we can use the simple approach of jumping to the alternate channel when the loss rate on the current channel exceeds a threshold (e.g. 10% in Change $\geq$ 10%).

Although the reduction in loss rate shown in Table 3.2 by the different algorithms is only of the order of 1-2%, the advantages of channel switching could be significant in presence of long or high-loss bursts.

Implications of channel switching: Even though adaptive channel switching seems to be a viable solution, large scale WiLD mesh deployments require careful channel assignment to avoid interference between multiple radios mounted on the same tower [84, 93]. Switching the frequency channel on one link could lead to a cascading effect requiring other links to also change their operating channel. Hence, although it could mitigate interference,

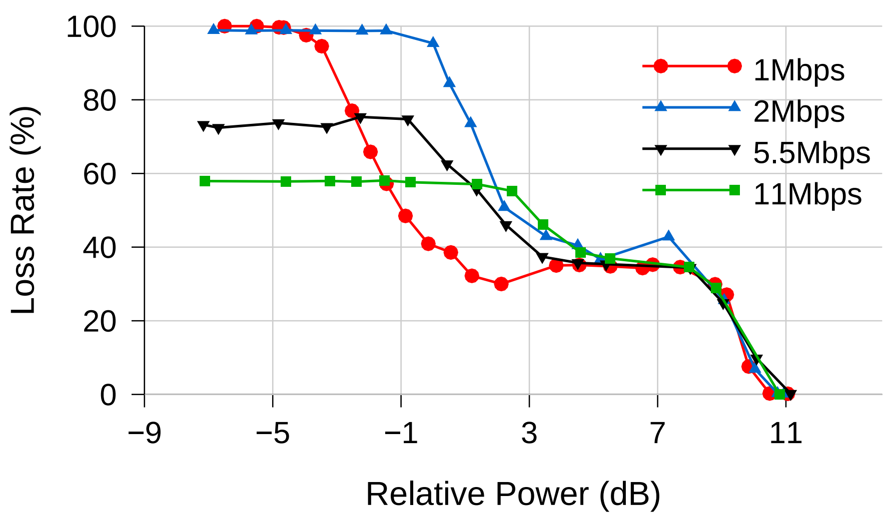


Figure 3.19: Loss rate for 802.11b encoding rates at varying relative power of transmitter compared to interferer.

it is not always possible to switch a frequency channel in a large scale deployment.

3.7.4 Rate Adaptation

Figure 3.19 shows the variation of loss rates as the relative power of the primary transmitter is increased with respect to that of the interference source for different 802.11 datarates.

We observed that in presence of external WiFi interference, data rate adaptation could either degrade the performance further or cause no effect on the loss rate. From figure 3.19 we see that when the received signal strength of the primary transmitter is higher than that of the interference source (from 0 to 12 dB), there is no difference in the loss rate for different 802.11b datarates. Whereas, when the interferer is stronger than the transmitter, reducing the data rate actually exacerbates the performance. This is because the increased transmission time of the frame increases the probability of a collision with the external traffic.

Implications for datarate selection: Most of the 802.11 radios have built in rate-adaptation algorithms which selects a lower rate with resilient encoding on experiencing high loss. However, the above analysis shows that in the presence of loss due to external WiFi interference, it is not worthwhile to adapt the data rate. Rather, we propose using other techniques such as adaptive FEC and link-layer retransmissions to mitigate the loss.

3.7.5 Adaptive Forward Error Correction

As discussed in the previous two sections, both channel and rate adaptation may not be feasible in large-scale WiLD mesh deployments. Furthermore, they only provide coarse-grain adaptations, which may not be suitable for QoS specific applications like video streaming. In this section we propose adaptive FEC as a solution to achieve fine-grained control. With an estimate of the channel loss variability, adaptive FEC allows addition of the "right" amount of redundancy to cope with the channel losses.

We evaluate a simple Reed-Solomon based adaptive FEC mechanism. Time is divided into slots (25 ms) and at the end of each slot the receiver informs the transmitter of the loss observed in the previous slot. Based on this link information, the transmitter adjusts the redundancy for the next round. To deal with transient spikes in loss rate, the sender maintains a moving window average of the loss rate (WinSize = 10). The application traffic is assumed to be a CBR traffic source (1.8 Mbps) and consuming only half the available bandwidth (3.8 Mbps at 11 Mbps); there is sufficient bandwidth per slot to introduce 100%

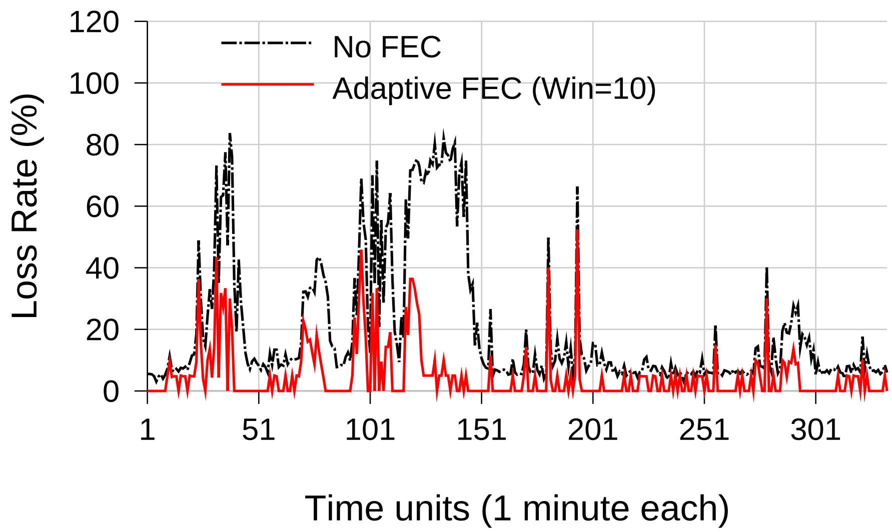


Figure 3.20: Comparison of loss rate observed with and without adaptive FEC

redundancy. To differentiate between new incoming packets and backlogged packets that were dropped in the previous slot, two separate queues are maintained. New incoming packets are queued at the beginning of every slot in the *application queue*. A *backlog queue* (2 x application queue length) maintains a queue of packets that have been transmitted, but not yet successfully acknowledged. Packets in the application queue are given higher priority as compared to the backlogged packets. Packets are dequeued off the backlog queue for re-transmission only when there is additional bandwidth remaining in the current slot. In absence of the backlog queue, packets transmitted with inadequate redundancy would be dropped.

Figure 3.20 shows a loss trace on the M-P link. The traffic source was a 1.5 Mbps UDP CBR traffic generator and the loss rate was averaged over 1 minute for a duration of approximately 6 hours. Here again, the MAC-layer ACKs were turned off and retries set to

	Loss Rate	Percent pkts delayed
No FEC	19.98	12.67
Oracle FEC	0	0
Adapt FEC (Win = 10)	4.78	2.86

Table 3.3: Summary of the effectiveness of adaptive FEC over the trace shown in Figure 3.20
.

zero. From the above figure we observe that the link was extremely bursty with bursts as high as 70–80% lasting for 20–30 minutes. Table 3.3 shows the performance comparison of the adaptive FEC algorithms. We measure the average loss rate and percentage of packets delayed. A packet is marked as delayed if it is received in a slot later than its scheduled slot. The baseline case is when there is no FEC being applied (No FEC). In this case, the average loss rate across the entire 6 hour period is 19.98% and approximately 12.67% packets were delayed. If the exact loss rate could be predicted for each slot (Oracle), then the loss rate and delayed packets are 0. However, in practice the channel loss rate cannot be predicted accurately, especially since the loss rate is determined by the external WiFi interference. A simple approach is to maintain a moving average window of the loss rate and for every time slot encode the frames to transmit additional redundant packets determined by the current value of the moving average. Table 3.3 shows that this simple approach (Adapt FEC) significantly reduces the loss rate to 4.78% and the packets delayed to 2.86%. Figure 3.20 shows the loss rate along with the original loss rate. From the figure we observe that the above simple approach can tolerate long bursts of high loss rate. However, FEC cannot adapt to transient high bursts.

3.8 Related Work

Although there have been numerous research studies on packet loss characterization and methodologies, here, we only focus on those works that are closely related to ours.

In a recent study [40], the authors present a detailed performance study of 10 WiLD links. This is the only other study of WiLD links performance that we are aware. The paper analyzes the behavior of WiLD links for varying packet sizes, data rates, link lengths, SNRs and weather conditions. The study also mentions external interference as one of the causes for high packet loss in these types of links.

Other measurement based studies:

The Roofnet project [20, 35] characterizes the causes of packet losses in an urban multi-hop 802.11b network. The authors conclude that the main source of packet loss in their urban mesh deployment was due to multipath interference. The authors observed a weak correlation between factors such as S/N ratio and link distance on loss rates. Also there was no correlation between loss rate and external WiFi interference. Jamieson *et al.* [66] experimentally evaluate the limitations of carrier-sense with respect to achieving high throughput in multi-hop environments. Garetto *et al.* [54] show that CSMA performs very badly in multihop wireless networks, and that this is not due to any particular implementation of a CSMA protocol, but is indeed a general coordination problem in multihop wireless networks.

A large number of measurement based studies have also been carried out to study the source of packet loss in indoor large scale 802.11 deployments [58, 65, 67, 104]. The authors in [67, 104] study the performance of 802.11 in a conference setting, where a large number of clients are using the wireless network. The authors observed both short- and long-term variability in link quality and performance degradation under heavy usage of the wireless network. The authors also point out that rate fallback exacerbates the link quality, leading to a higher number of retransmissions and dropped frames.

3.9 Conclusions

We perform a detailed study of channel induced (WiFi, non-Wifi, and multipath interference) and protocol induced (timeouts, breakdown of CSMA) losses in WiLD settings. Our main result is that most of the losses arise due to external WiFi interference on same and adjacent channels. This result is in contrast to loss studies of urban mesh networks, where multipath is reported to be the most significant source of loss. We also show that 802.11b protocol limitations make it unsuitable not just for point-to-multipoint links, as claimed in prior work, but also unsuitable for simple point-to-point links. In addition, we analyze the loss variability in both urban and rural links and show that urban links suffer from a higher degree of residual loss. Finally, we propose and analyze the effectiveness of three remedial strategies to mitigate the losses caused by external WiFi interference.

Chapter 4

WiLDNet: How to deal with loss and interference in WiLD networks

.

In this chapter, we discuss the design and implementation of WiLDNet, a system that builds efficient MAC and link-layer mechanisms for long-distance directional wireless links. These mechanisms eliminate the 802.11 protocol inefficiencies identified in Chapter 3, by avoiding packet collisions and inter-link interference. To address the protocol shortcomings, WiLDNet makes several essential changes to the 802.11 MAC protocol, but continues to exploit standard (low-cost) WiFi network cards. To better handle losses and improve link utilization, WiLDNet uses an adaptive loss-recovery mechanism using FEC and bulk acknowledgments. Based on a real-world deployment, WiLDNet provides a 2–5 fold improvement in TCP/UDP throughput (along with significantly reduced loss rates)

WiLDNet was designed and implemented in collaboration with Rabin Patra, Sonesh Surana, Lakshminarayanan Subramanian, Anmol Sheth and Eric Brewer [84] in comparison to the best throughput achievable by conventional 802.11. WiLDNet can also be configured to adapt to a range of end-to-end performance requirements in terms of bandwidth, delay and loss.

4.1 Contributions

To summarize the findings from the previous chapter, the stock 802.11 protocol has fundamental *protocol shortcomings* that make it ill-suited for WiLD environments. These include: (a) the *802.11 link-level recovery* mechanism results in low utilization; (b) at *long distances frequent collisions occur because of the failure of CSMA/CA*; (c) WiLD networks experience *inter-link interference*, which introduces the need for synchronizing packet transmissions at each node. Another significant problem is the fact that WiLD links experience very *high and variable packet loss rates* induced by external factors (primarily external WiFi interference in our deployment); under such high loss conditions, TCP flows hardly progress and continuously experience timeouts.

In order to address these shortcomings, we design WiLDNet, which brings the following contributions:

1. Using a TDMA-based approach to medium access, instead of the standard CSMA/CA mechanism: The stock 802.11 CSMA/CA mechanism is inappropriate for WiLD settings since it cannot assess the state of the channel at the receiver, making it prone to collisions and interference. Instead, we propose a TDMA mechanism that provides the appropriate coordination between link endpoints, as well as across several adjacent links operating on the same channel.

2. Bulk acknowledgments and sliding-window flow-control to improve link utilization: The current 802.11 protocol uses a stop-and-wait link recovery mechanism, which when used over long distances with high round-trip times leads to under-utilization of the channel. In order to pipeline the transmission of multiple packets, and therefore improve link utilization, WiLDNet uses a slinding window flow-control, and bulk acknowledgment of entire sequences of packets.

3. Adaptive loss recovery mechanisms: In our WiLD network deployments, we found that external WiFi interference is the primary source of packet loss. The emergence of many WiFi deployments, even in developing regions, will exacerbate this problem. In WiLDNet, we use an adaptive loss-recovery mechanism that uses a combination of FEC and bulk acknowledgments to significantly reduce the perceived loss rate and to increase the end-to-end throughput. We show that WiLDNet's link-layer recovery mechanism is much more efficient than a higher-layer (transport) recovery mechanisms such as Snoop [30].

4. Application-based loss recovery configuration: Different applications have varying requirements in terms of bandwidth, loss, delay and jitter. In WiLDNet, configuring the TDMA and recovery parameters (time slot period, FEC, number of retries) provides a tradeoff spectrum across different end-to-end properties. We explore these tradeoffs and show that WiLDNet can be configured to suit a wide range of goals.

4.2 WiLDNet Design

In this section, we describe the design of WiLDNet. The system follows three main design principles. First, the system is tailored to a broad range of network application and traffic patterns. It is configurable, providing knobs that can be used to negotiate the tradeoff between different end-to-end properties, including delay, bandwidth, loss, reliability and jitter. Second, all mechanisms proposed are implementable using commodity off-the-shelf 802.11 radio cards. Third, the design is lightweight, making WiLDNet deployable on resource-constrained, low cost single-board computers (266-MHz CPU and 128 MB memory) used in our deployments.

4.2.1 Bulk Acknowledgments

We begin with the simple case of a single WiLD link, with each node having a half-duplex radio. As shown earlier, when propagation delays become longer, the default CSMA mechanism cannot determine whether the remote peer is sending a packet in time to back-off its own transmission and avoid collisions. Moreover, such a contention-based mechanism is overkill when precisely two hosts share the channel for a directional link.

Thus, a simple and efficient solution to avoid these collisions is to use an echo protocol between the sender and the receiver, which allows the two end-points to take turns sending and receiving packets. Hence, from a node's perspective, we divide time into send and receive time slots, with a burst of several packets being sent from one host to its peer in each slot.

Consequently, to improve link utilization, we replace the stock 802.11 stop-and-wait protocol with a sliding-window based flow-control approach in which we transmit a *bulk acknowledgment* (bulk ACK) from the receiver for a window of packets. We generate a bulk ACK as an aggregated acknowledgment for all the packets received within the previous slot. In this way, a sender can rapidly transmit a burst of packets rather than wait for an

ACK after each packet.

The bulk ACK can be either piggybacked on data packets sent in the reverse direction, or sent as one or more stand-alone packets if no data packets are ready. Each bulk ACK contains the sequence number of the last packet received in order and a variable-length bit vector ACK for all packets following the in-order sequence. Here, the sequence number of a packet is locally defined between the pair of end-points of a WiLD link.

Like 802.11, the bulk ACK mechanism is not designed to guarantee perfect reliability. 802.11 has a maximum number of retries for every packet. Similarly, upon receiving a bulk ACK, the sender can choose to advance the sliding window skipping unacknowledged packets if the retry limit is exceeded. In practice, we support different retry limits for packets of different flows. The bulk ACK mechanism introduces packet reordering at the link layer, which may not be acceptable for TCP traffic. To handle this, we provide in-order packet delivery at the link layer either for the entire link or on a per-flow basis.

4.2.2 Designing TDMA on Lossy Channels

To address the inappropriateness of CSMA for WiLD networks, 2P [93] proposes a contention-free TDMA based channel access mechanism. 2P eliminates inter-link interference by synchronizing all the packet transmissions at a given node (along all links which operate on the same channel). In 2P, a node in transmission mode simultaneously transmits on all its links for a globally known specific period, and then explicitly notifies the end of its transmission period to each of its neighbors using marker packets. A receiving node waits for the marker packets from all its neighbors before switching over to transmission mode. In the event of a loss of a marker packet, a receiving node uses a timeout to switch into the

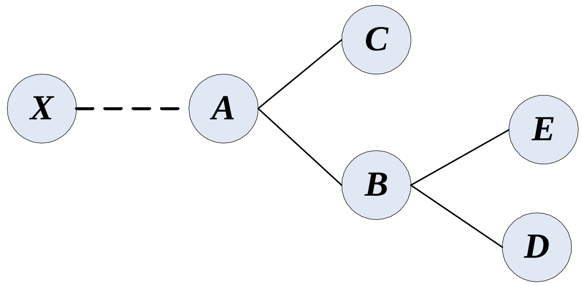


Figure 4.1: Example topology to compare synchronization of 2P and WiLDNet.

transmission mode.

The design of 2P, while functional, is not well suited for lossy environments. Consider the simple example illustrated in Figure 4.1, where all links operate on the same channel. Consider the case where (X, A) is the link experiencing high packet loss-rate. Let T denote the value of the time-slot. Whenever a marker packet transmitted by X is lost, A begins transmission only after a timeout period T_0 $(\geq T)$. This, in turn, delays the next set of transmissions from nodes B and C to their other neighbors by a time period that equals $T_0 - T$. Unfortunately, this propagation of delay does not end here. In the time slot that follows, D's transmission to its neighbors is delayed by $T_0 - T$. Hence, what we observe is that the loss of marker packets has a "ripple effect" in the entire network creating an idle period of $T_0 - T$ along every link. When markers along different links are dropped, the ripples from multiple links can interact with each other and cause more complex behavior.

Ideally, one would want $T_0 - T$ to be very small. If all nodes are perfectly time synchronized, we can set $T_0 = T$. However, in the absence of global time synchronization, one needs to set a conservative value for T_0. 2P chooses $T_0 = 1.25 \times T$. The loss of a marker packet leads to an idle period of $0.25 \times T$ (in 2P, this is 5 ms for $T = 20$ ms). In bursty losses, transmitting multiple marker packets may not suffice.

Given that many of the links in our network experience sustained loss-rates over 5–40%, in WiLDNet, we use an implicit synchronization approach that aims to reduce the value of $T_0 - T$. In WiLDNet, we use a simple loose time synchronization mechanism similar to the basic linear time synchronization protocol NTP [76], where during each time slot along each link, the sender acts as the master and the receiver as the slave. Consider a link (A, B) where A is the sender and B is the receiver at a given time. Let t_{send_A} and t_{recv_B} denote the start times of the slot as maintained by A and B. All the packets sent by A are timestamped with the time difference (δ) between the time the packet has been sent (t_1) and the beginning of the send slot(t_{send_A}). When a packet is received by B at time t_2, the beginning of B's receiving slot is adjusted accordingly: $t_{recv\_B} = t_2 - \delta$.

As soon as B's receive slot is over, and $t_{send\_B} = t_{recv\_B} + T$ is reached, B starts sending for a period T.

Due to the propagation delay between A and B, the send and corresponding receive slots are slightly skewed. The end-effect of this loose synchronization is that the value of $T_0 - T$ is limited by the propagation delay across the link even with packet losses (assuming clock speeds are roughly comparable). Hence, an implicit synchronization approach significantly reduces the value of $T_0 - T$ thereby reducing the overall number of idle periods in the network.

4.2.3 Adaptive Loss Recovery

To achieve predictable end-to-end performance, it is essential to have a loss recovery mechanism that can hide the loss variability in the underlying channel. Achieving such an upper bound (q) on the loss-rate perceived by higher level applications is not easy in our settings. First, it is hard to predict the arrival and duration of bursts. Second, the loss distribution that we observed on our links is non-stationary even on long time scales (hourly or daily). Hence, a simple model cannot capture the channel loss characteristics.

In WiLDNet, we can either use retransmissions or FEC to deal with losses (or a combination of both). A retransmission based approach can achieve the loss-bound q with minimal throughput overhead but at the expense of increased delay. An FEC based approach incurs additional throughput overhead but does not incur a delay penalty especially since it is used in combination with TDMA on a per-slot basis. However, an FEC approach cannot achieve arbitrarily low loss-bounds mainly due to the unpredictability of the channel.

Tuning the Number of Retransmissions

To achieve a loss bound q independent of underlying channel loss rate $p(t)$, we need to tune the number of retransmissions. One can adjust the number of retransmissions $n(t)$ for a channel loss-rate $p(t)$ such that $(1-p(t))^{n(t)} = q$. Given that our WiLD links support in-order delivery (on a per-flow or on whole link basis), a larger $n(t)$ also means a larger maximum delay, equal to $n(t) * T$ for a slot period T. One can set different values of $n(t)$ for different flows. We found that estimating $p(t)$ using an exponentially weighted average is sufficient in our links to achieve the target loss estimate q. A purely retransmission based recovery mechanism has minimal throughput overhead as only the lost packets are retransmitted but this comes at a cost of high delay due to the long round-trip times over WiLD links.

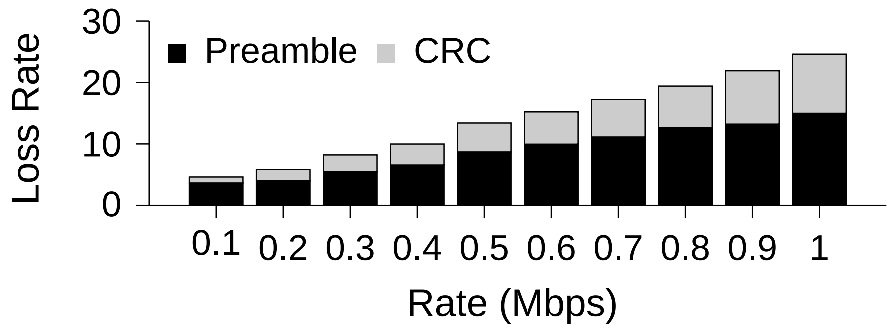


Figure 4.2: Proportion of CRC and preamble errors in channel loss. Traffic is at UDP CBR packets of 1440 bytes each at 802.11b PHY datarate of 11Mbps. Main link is sending at 2Mbps. The sending rate of the interferer increases from 0.1Mbps to 1Mbps.

Adaptive FEC-Based Recovery

Designing a good FEC mechanism in highly variable lossy conditions requires accurate estimation of the underlying channel loss. When the loss is underestimated, the redundant packets cannot be decoded at all making them useless, but overestimating the loss rate leads to unnecessary overhead.

Motivating inter-packet FEC: We can perform two types of FEC: inter-packet FEC (coding across packets) or intra-packet FEC (coding redundant blocks within a packet). Based on extensive measurements on a wireless channel emulator we observe that in presence of external WiFi interference, lost packets can be categorized into either CRC errors or preamble errors. A CRC error packet is received by the driver with a check sum error. However, an error in the preamble leads to the entire packet being dropped completely. Figure 4.2 shows the breakup of the loss rate with increasing external interference. We observe although the proportion of preamble errors decreases as external interference increases, it still causes at least 50% of all errors. Moreover a substantial number of the CRC error packets were truncated. We choose not to perform intra-packet FEC because it can only help recover packets that have CRC errors. Hence, we chose to perform inter-packet FEC.

Estimating redundancy: We apply FEC in combination with TDMA. For every time slot of N packets, we add $N-K$ redundant packets to K original packets. To estimate the redundancy factor, $r = (N-K)/K$, we choose a simple but not perfect estimation policy based on a weighted average of the losses observed in the previous M time slots. Here, we specifically chose a small value of $M = 10$ because it is hard to predict the start of a burst. Secondly, a small value of M, can quickly adapt to both the start and end of a loss burst saving unnecessary redundant FEC packets. For a time slot of $T = 10ms$, $M = 10$ corresponds to $200ms$ (with symmetric slot allocation in both directions) to adapt to a change in the loss behavior. Also due to non-stationary loss distributions, the benefit of using more complicated distribution based estimation approaches [109] is marginal. This type of FEC is best suited for handling residual losses and bursts that are longer than the time required for loss estimation mechanism to adapt.

4.3 Implementation

In this section, we describe the implementation details of WiLDNet. Our implementation comprises two parts: (a) driver-level modifications to control or disable features implemented in hardware (Section 4.3.1), and (b) a *shim* layer that sits above the 802.11 MAC (Section 4.3.2) and uses the Click [71] modular router software to implement the functionalities described in Section 4.2.

4.3.1 Driver Modifications

The wireless cards we use in our implementation are the high power (200–400 mW) Atheros-based chipsets. To implement WiLDNet, we have to disable the following 802.11 MAC mechanisms:

- We disable **link-layer association** in Atheros chipsets using the *AdHoc-demo* mode.
- We disable **link-layer retransmissions and automatic ACKs** by using 802.11 QoS frames with WMM extensions set to the no-ACK policy.
- We disable **CSMA** by turning off the Clear Channel Assessment (CCA) in Atheros chipsets. With CCA turned off, the radio card can transmit packets right away without waiting for a clear channel.

4.3.2 Software Architecture Modifications

In order to implement single-link and inter-link synchronization using TDMA, the various loss recovery mechanisms, sliding window flow-control, and packet reordering for in-order delivery, we use the Click modular router [71] framework. We use Click because it enables us to prototype quickly a modular MAC layer by composing different Click elements together. It is also reasonably efficient for packet processing especially if loaded as a kernel module. Using kernel taps, Click creates fake network interfaces, such as *fake0* in Figure 4.3 and the kernel communicates with these virtual interfaces. Click allows us to intercept packets sent to this virtual interface and modify them before sending them on the real wireless interface and vice versa.

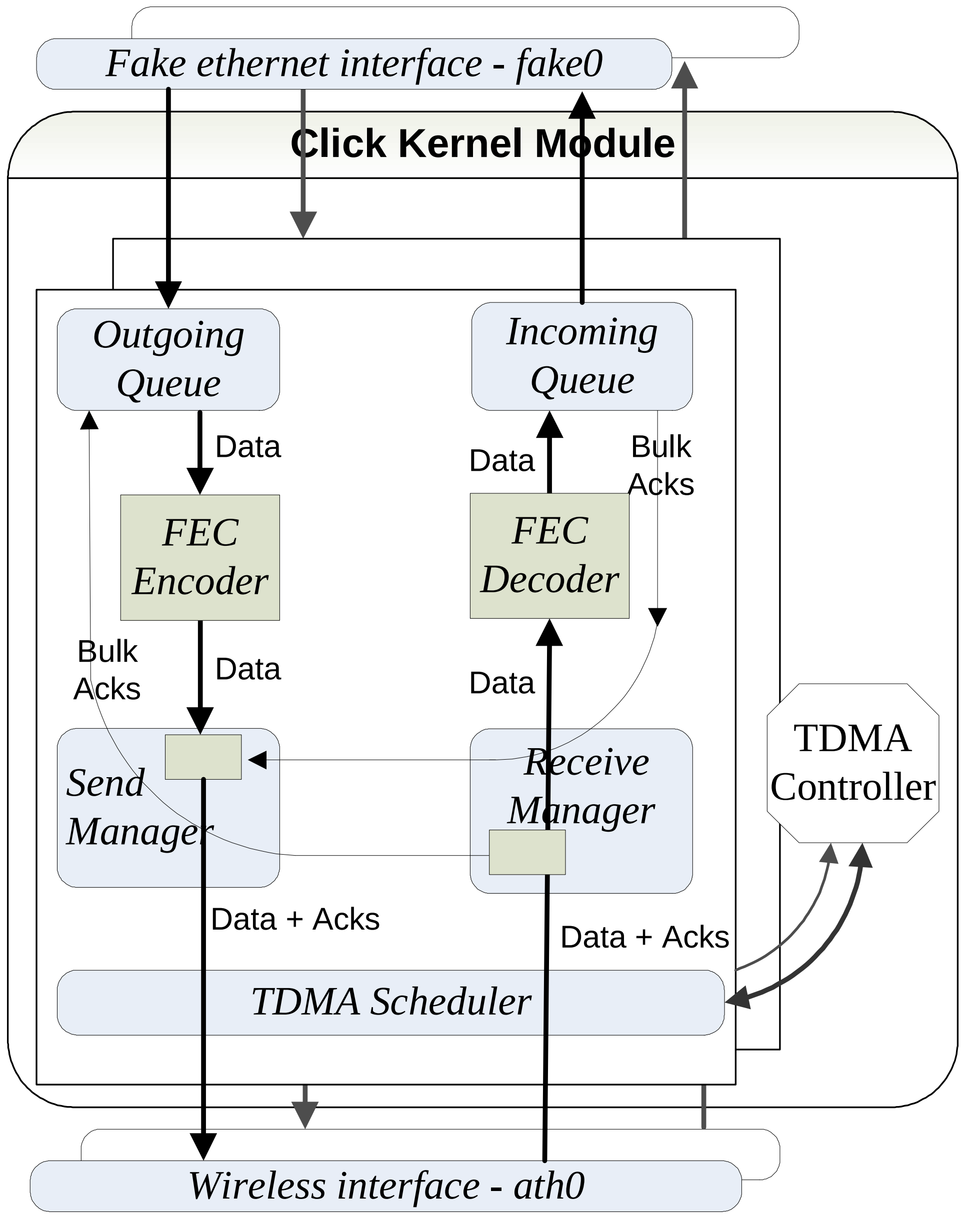


Figure 4.3: Click Module Data Flow

Figure 4.3 presents the structure of the Click elements of our layered system, with different functionality (and corresponding packet header processing) at various layers:

Incoming/Outgoing Queues: The mechanisms for sliding window packet flow, bulk ACKs, selective retransmission and reordering for in-order delivery are implemented by the incoming/outgoing queue pair. Packet buffering at the sender is necessary for retransmissions, and buffering at the receiver enables reordering. In-order delivery and packet retransmission are optional, and the number of retries can be set on a per-packet basis.

FEC Encoder/Decoder: An optional layer is responsible for inter-packet forward error correction encoding and decoding. For our implementation we modify a FEC library [102] that uses erasure codes based on Vandermonde matrices computed over $GF(2^m)$. This FEC method uses a (K, N) scheme, where the first K packets are sent in their original form, and $N - K$ redundant packets are generated, for a total of N packets sent. At the receiver, the reception of any K out of the N packets enables the recovery of the original packets. We choose this scheme because, in loss-less situations, it introduces very low latency: the original K packets can be immediately sent by the encoder (without undergoing encoding), and immediately delivered to the application by the decoder (without undergoing decoding).

TDMA Scheduler and Controller: The Scheduler ensures that packets are being sent only during the designated send slots, and manages packet timestamps as part of the synchronization mechanism. The Controller implements synchronization among the wireless radios, by enforcing synchronous transmit and receive operation (all the radios on the same channel have a common send slot, followed by a common receive slot).

Timing issues

We do not use Click timers to implement time synchronization because the underlying kernel timers are not precise at the granularity of our time slots (10–40ms) on our hardware platform (266MHz CPU). Also packet queuing in the wireless interface causes variability in the time between the moment Click emits a packet and the time the packet is actually sent on the air interface. Thus, the propagation delay between the sending and the receiving click modules on the two hosts is not constant, affecting time slot calculations. Fortunately, this propagation delay is predictable for the first packet in the send slot, when the hardware interface queues are empty. Thus, in our current implementation, we only timestamp the first packet in a slot, and use it for adjusting the receive slot at the peer. If this packet is lost, the receiver's slot is not adjusted in the current slot, but since the drift is slow this does not have a significant impact. In the future we intend to perform this timestamping in the firmware — that would allow us to timestamp every packet accurately just before packet transmission.

Another timing complication is related to estimating whether we have time to send a new packet in the current send slot. Since the packets are queued in the wireless interface, the time when the packet leaves Click cannot be used to estimate this. To overcome this aspect, we use the notion of *virtual time*. At the beginning of a send slot, the virtual time t_v is same as current (system) time t_c. Every time we send a packet, we estimate the transmission time of the packet on the channel and recompute the virtual time: $t_v = max(t_c, t_v) + duration(packet)$. A packet is sent only after checking that the virtual time after sending this packet will not exceed the end of the send slot. Otherwise, we postpone the packet until the next slot.

4.4 Experimental Evaluation

The main goals of WiLDNet are to increase link utilization and to eliminate the various sources of packet loss observed in a typical multi-hop WiLD deployment, while simultaneously providing flexibility to meet different end-to-end application requirements. We believe these are the first actual implementation results over an outdoor multi-hop WiLD network deployment.

Raman *et al.* [93] show the improvements gained by the 2P-MAC protocol in simulation and in an indoor environment. However, a multi-hop outdoor deployment also has to deal with high losses from external interference. 2P in its current form does not have any built-in recovery mechanism and it is not clear how any recovery mechanism can be combined with the marker-based synchronization protocol. Hence, we do not have any direct comparison results with 2P on our outdoor wireless links. Also, the proof-of-concept implementation of 2P was for the Prism 2.5 wireless chipset and it would be non-trivial to implement the same in WiLDNet using features of the Atheros chipset.

Our evaluation has three main parts:

- We analyze the ability of WiLDNet to maintain high performance (high link utilization) over long-distance WiLD links. At long distances, we demonstrate 2–5x improvements in cumulative throughput for TCP flows in both directions simultaneously.

- Next, we evaluate the ability of WiLDNet to scale to multiple hops and eliminate inter-link interference. WiLDNet yields a 2.5x improvement in TCP throughput on our real-world multi-hop setup.

- Finally, we evaluate the effectiveness of the two link recovery mechanisms of WiLDNet: Bulk Acks and FEC.

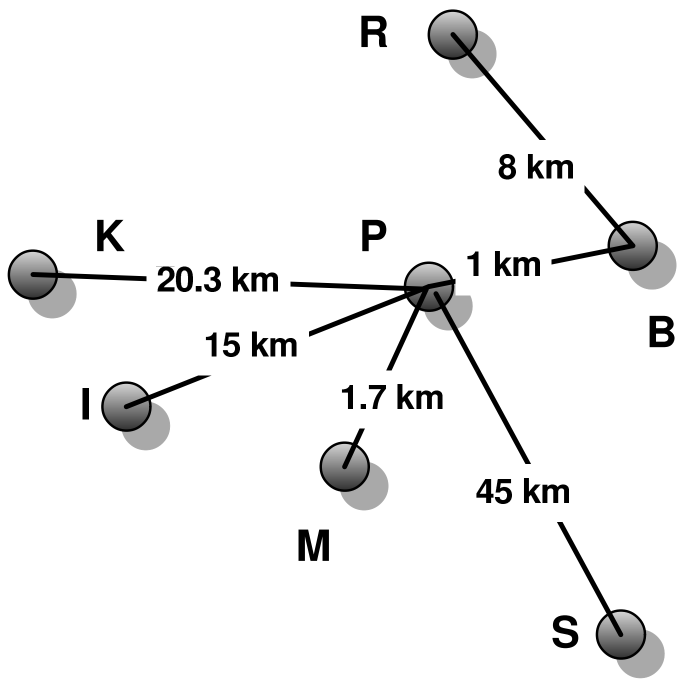


Figure 4.4: Overview of the WiLD campus testbed (not to scale)

4.4.1 Experimental Setup

We use three different experimental setups to conduct measurements and to evaluate WiLDNet.

Campus testbed: Figure 4.4 is our real-world campus testbed on which we have currently deployed WiLDNet. The campus testbed consists of links ranging from 1 to 45 km, with end points located in areas with varying levels of external WiFi interference. We also use one of the links in our Ghana network, a link measuring 65km.

Wireless Channel Emulator: The channel emulator (Spirent 5500 [11]) enables repeatable experiments by keeping the link conditions stable for the duration of the experiment. Moreover, by introducing specific propagation delays we can emulate very long links and hence study the effect of long propagation delays. We can also study this in isolation of external interference by placing the end host radios in RF isolation boxes.

Indoor multi-hop testbed: We perform controlled multi-hop experiments on an indoor multi-hop testbed consisting of 4 nodes placed in RF isolated boxes. The setup was designed to recreate conditions similar to long outdoor links where transmissions from local radios interfere with each other but simultaneous reception on multiple local radio interfaces is possible. We can also control the amount of external interference by placing an additional wireless node in each isolation box just to transmit packets mimicking a real interferer. The amount of interference is controlled by the rate of the CBR traffic sent by this node. The indoor setup features very small propagation delay on the links; we use it only to perform experiments evaluating TDMA scheduling and loss recovery from interference.

We use Atheros 802.11 a/b/g radios for all our experiments. The wireless nodes are 266 MHz x86 Geode single-board computers running Linux 2.4.26. The choice of this hardware platform is motivated by the low cost ($140) and the low power consumption ($<$ 5W). We use *iperf* to measure UDP and TCP throughput. The madwifi Atheros driver was modified to collect relevant PHY and MAC layer information.

4.4.2 Single Link

In this section we demonstrate the ability of WiLDNet to eliminate link under-utilization and packet collisions over a single WiLD link. We compare the performance of WiLDNet (slot size of 20ms) with the standard 802.11 CSMA (2 retries) base case.

The first set of results show the improvement of WiLDNet on a single emulator link with increasing distance. For these experiments, we use 802.11b radios, operating at

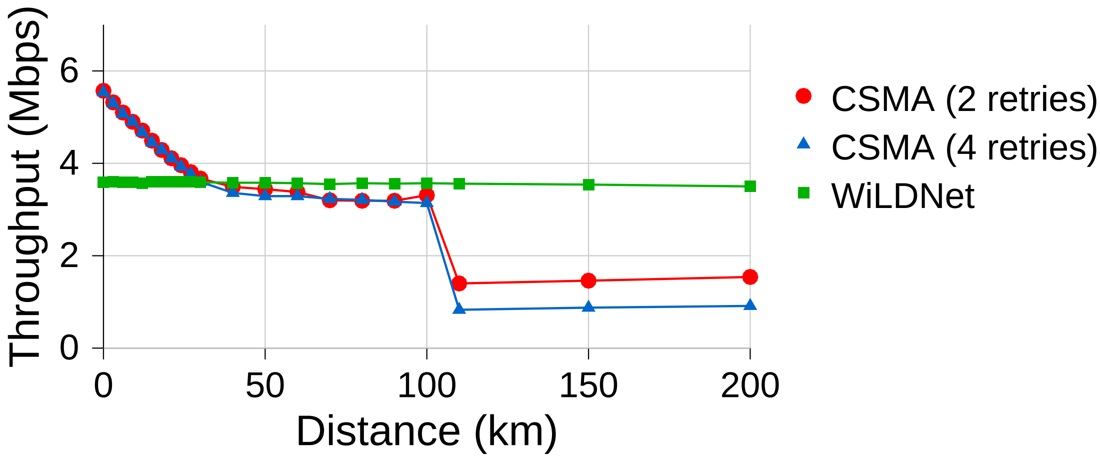


Figure 4.5: TCP flow in one direction

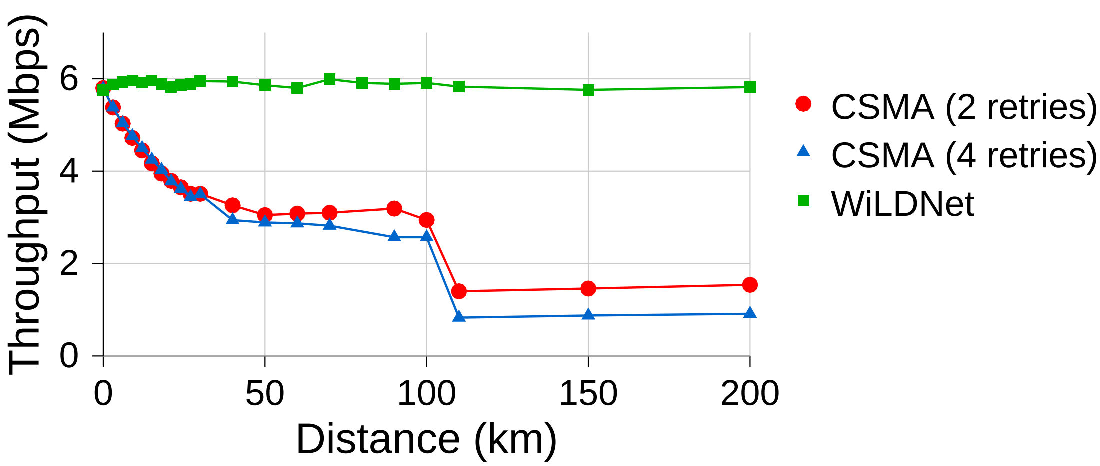


Figure 4.6: TCP flow in both directions

a datarate of 11Mbps. Figure 4.4.2 compares the performance of TCP flowing only in one direction. The lower throughput of WiLDNet, approximately 50% of channel capacity, is due to symmetric slot allocation between the two end points of the link. However, over longer links (>50 km), the TDMA-based channel allocation avoids the under-utilization of the link as experienced by CSMA. Also, beyond 110 km (the maximum possible ACK

Link	Distance	Loss rates	802.11 CSMA (Mbps)		WiLDNet (Mbps)	
	(km)	(%)	One dir	Both dir	One dir	Both dir
B-R	8	3.4	5.03 (0.02)	**4.95** (0.03)	3.65 (0.01)	**5.86** (0.05)
P-S	45	2.6	3.62 (0.20)	**3.52** (0.17)	3.10 (0.05)	**4.91** (0.05)
Ghana	65	1.0	2.80 (0.20)	**0.68** (0.39)	2.98 (0.19)	**5.51** (0.07)

Table 4.1: Mean TCP throughput (flow in one direction and cumulative for both directions simultaneously) for WiLDNet and CSMA for various outdoor links (distance and loss rates). The standard deviation is shown in parenthesis for 10 measurements. Each measurement is for TCP flow of 30s at a 802.11b PHY-layer datarate of 11Mbps.

timeout), the throughput with CSMA drops rapidly because of unnecessary retransmits (Section 3.3). Figure 4.4.2 shows the cumulative throughput of TCP flowing simultaneously in both directions. In this case, WiLDNet effectively eliminates all collisions occurring in presence of bidirectional traffic. TCP throughput of 6 Mbps is maintained for all distances.

Table 4.1 compares WiLDNet and CSMA for some of our outdoor wireless links. We show TCP throughput in one direction and the cumulative throughput for TCP simultaneously flowing in both directions. Since these are outdoor measurements, there is significant variation over time and we show both the mean and standard deviation for the measurements. We can see that as the link distance increases, the improvement of WiLDNet is more substantial. Infact, for the 65 km link in Ghana, WiLDNet's throughput at 5.5 Mbps is about 8x better than standard CSMA.

4.4.3 Multiple Hops

This section validates that WiLDNet eliminates inter-link interference by synchronizing receive and transmit slots in TDMA resulting in up to 2x TCP throughput improvements over standard 802.11 CSMA in multi-hop settings.

Setup	802.11 CSMA (Mbps)			WiLDNet (Mbps)		
	Dir 1	Dir 2	Both	Dir 1	Dir 2	Both
2 nodes	5.74 (0.01)	5.74 (0.01)	**6.00** (0.01)	3.56 (0.03)	3.53 (0.02)	**5.85** (0.07)
3 nodes	2.60 (0.01)	2.48 (0.01)	**2.62** (0.01)	3.12 (0.01)	3.12 (0.01)	**5.12** (0.03)
4 nodes	2.23 (0.01)	2.10 (0.01)	**1.99** (0.02)	2.95 (0.05)	2.98 (0.04)	**4.64** (0.24)

Table 4.2: Mean TCP throughput (flow in each direction and cumulative for both directions simultaneously) for WiLDNet and standard 802.11 CSMA. Measurements are for linear 2, 3 and 4 node indoor setups recreating outdoor links running on the same channel. The standard deviation is shown in parenthesis for 10 measurements of flow of 60s each at 802.11b PHY layer datarate of 11Mbps.

The first set of measurements were performed on our indoor setup (Section 4.4.1) where we recreated the conditions of a linear outdoor multi-hop topology using the RF isolation boxes. Thus transmissions from local radios interfere with each other but multiple local radio interfaces can receive simultaneously. We then measure TCP throughput of flows in the one direction and then both directions simultaneously for both standard 802.11 CSMA and WiLDNet (with slot size of 20ms). All the links were operating on the same channel. As we see in Table 4.2, as the number of hops increases, standard 802.11's TCP throughput drops substantially when transmissions from a radio collide with packet reception on a nearby local radio on the same node. WiLDNet avoids these collisions and maintains a much higher cumulative TCP throughput (up to 2x for the 3-hop setup) by proper synchronization of send and receive slots.

We can also see that although WiLDNet has more than 2x improvement over standard 802.11, the final throughput (4.6Mbps) is still much smaller than the raw throughput of the link (6-7Mbps). This can be attributed to the overhead of synchronization and packet processing in Click running on our low-power (266MHz) single board routers. A more efficient synchronization mechanism implemented in the firmware (rather than Click) would

Description (Mbps)	One direction	Both directions
Standard TCP: same channel	**2.17**	**2.11**
Standard TCP: diff channels	3.95	4.50
WiLD TCP: same channel	**3.12**	**4.86**
WiLD TCP: diff channels	3.14	4.90

Table 4.3: Mean TCP throughput (flow in single direction and cumulative for both directions simultaneously) comparison for WiLDNet and standard 802.11 CSMA over a 3-hop outdoor setup ($K \leftrightarrow P \leftrightarrow M$). Averaged over 10 measurements of TCP flow for 60s at 802.11b PHY layer datarate of 11Mbps.

deliver much better improvement.

We also measure this improvement on our outdoor testbed between the nodes K and M relayed through node P. We again compare the TCP throughput for WiLDNet and standard 802.11 CSMA with links operating on the same channel. In order to quantify the effect of inter-link interference, we also perform the same experiments with the links operating on different, non-overlapping channels, in which case the inter-link interference is almost zero.

The results of these experiments, presented in Table 4.3, show that, for same channel operation, the cumulative TCP throughput in both directions with WiLDNet (4.86 Mbps) is more than twice the throughput observed over standard 802.11 (2.11 Mbps). The improvement is substantially lower for the unidirectional case (3.14 Mbps versus 2.17 Mbps), because the WiLD links are constrained to send in one direction only roughly half of the time.

Another key observation is that WiLDNet is capable of eliminating almost all inter-link interference. This is shown by the fact that the throughput achieved by WiLDNet is the same, whether the links operate on the same channel or on non-overlapping channels. This

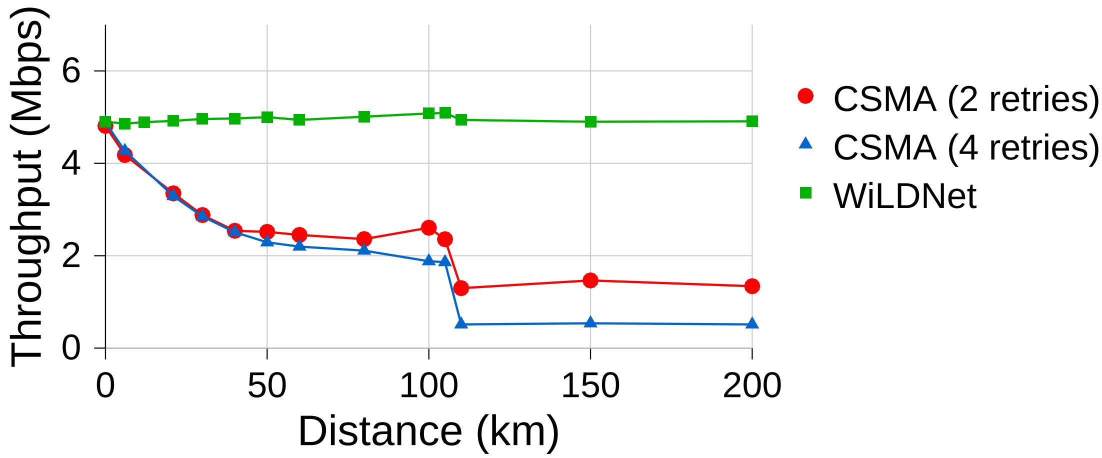


Figure 4.7: TCP flow in both directions, 10% channel loss

result indicates that WiLDNet increases *spectral efficiency*, allowing for the same channel to be reused on multiple adjacent links, and making channel allocation much simpler.

4.4.4 WiLDNet Link-Recovery Mechanisms

Our next set of experiments evaluate WiLDNet's adaptive link recovery mechanisms in conditions closer to the real world, where errors are generated by a combination of collisions and external interference. We evaluate both the bulk ACK and FEC recovery mechanisms.

Bulk ACK Recovery Mechanism

For our first experiment, presented in Figure 4.7, we vary the link length on the emulator, and we introduce a 10% error rate through external interference. We again measure the cumulative throughput of TCP flows in both directions for WiLDNet and standard 802.11 CSMA. As can be seen, WiLDNet maintains a constant throughput with

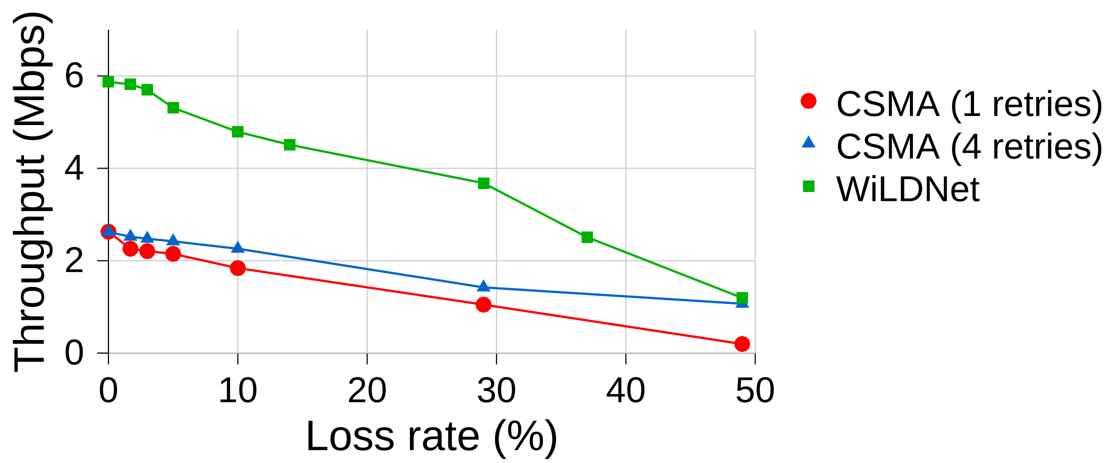


Figure 4.8: Comparison of cumulative throughput for TCP in both directions simultaneously for WiLDNet and standard 802.11 CSMA with increasing loss on 80km emulated link. Each measurement was for 60s TCP flows of 802.11b at 11Mbps PHY datarate.

increasing distance as opposed to the 802.11 CSMA. Due to the 10% error, WiLD incurs a constant throughput penalty of approximately 1 Mbps compared to the no-loss case in Figure 4.4.2.

In our second experiment we fix the distance in the emulator setup to 80 km, and vary channel loss rates. The results in Figure 4.8 show that WiLDNet maintains roughly a 2x improvement over standard CSMA's recovery mechanism for packet loss rates up to 30%.

Forward Error Correction (FEC)

To measure the jitter introduced by the FEC mechanism, we performed a simple experiment where we measured the jitter of a flow under two conditions: in the absence of any loss and in the presence of a 25% loss. Figure 4.9 illustrates the jitter introduced by WiLDNet's FEC implementation. We can see that in the absence of any loss, when only

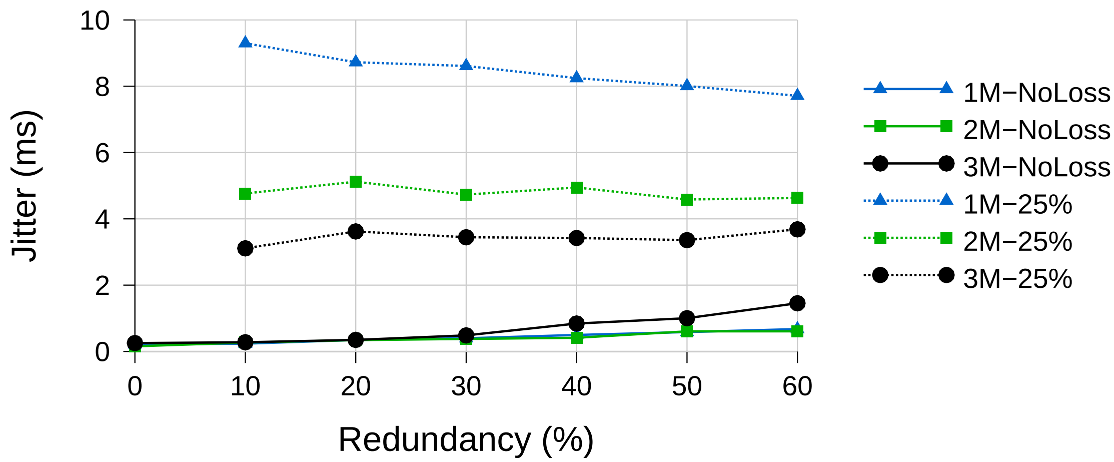


Figure 4.9: Jitter overhead of encoding and decoding for WiLDNet on single indoor link. Traffic is 1440-byte UDP CBR packets at PHY datarate of 11Mbps in 802.11b.

encoding occurs, the jitter is minimal. However, in the presence of loss, when decoding also takes place, the measured jitter increases. However, the magnitude of the jitter is well within the acceptable limits of many interactive applications (voice or video), and decreases with higher throughputs (since the decoder waits less for redundant packets to arrive).

Moreover, considering the combination of FEC with TDMA, the delay overheads introduced by these methods overlap, since the slots when the host is not actively sending can be used to perform encoding without incurring any additional delay penalties.

4.5 Tradeoffs

One of the main design principles of WiLDNet is to build a system that can be configured to adapt to different application requirements. In this section we explore the tradeoff space of throughput, delay and delivered error rates by varying the slot size, number of bulk retransmissions and FEC redundancy parameters. We observe that WiLDNet can

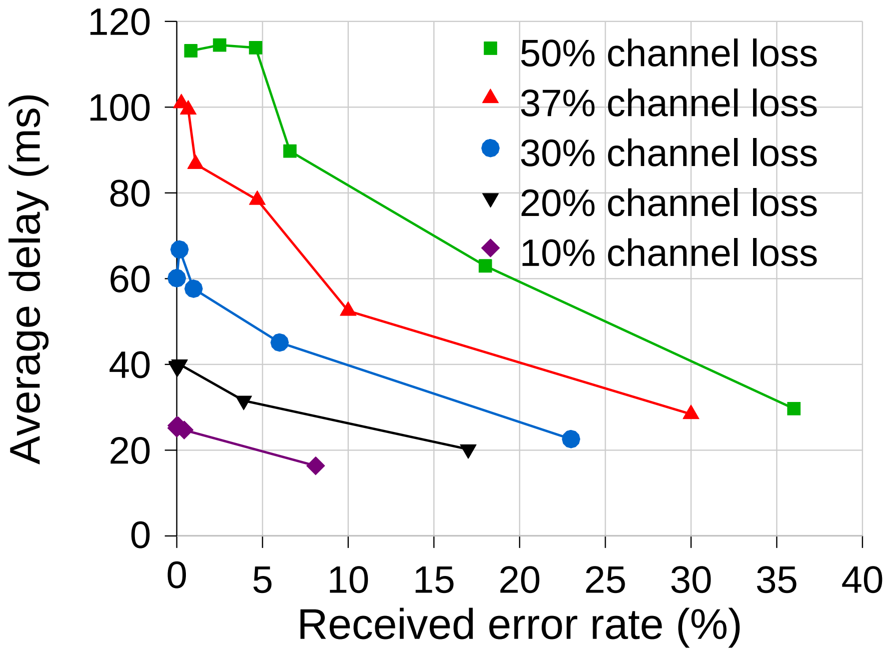


Figure 4.10: Average delay with decreasing target loss rate (X-axis) for various loss rates in WiLDNet on single emulated 60km link (slot size=20ms).

perform in a wide spectrum of the parameter space, and can easily be configured to meet specific application requirements.

4.5.1 Choosing number of retransmissions

The first tradeoff that we explore is choosing the number of retries to get a desired level of final error rate on a WiLD link. Although retransmission based loss recovery achieves optimal throughput utilization, it comes at a cost of increased delay; the loss rate can be reduced to zero by arbitrarily increasing the number of retransmissions at the cost of increased delay. This tradeoff is illustrated in Figure 4.10 which shows a plot of delay versus error rate for varying channel loss rates (10% to 50%). Retries are decreased from 10 to 0 from left to right for a given series in the figure. All the tests are with unidirectional UDP at 1 Mbps for a fixed slot size of 20ms on a single emulator 60km link. We can see that as we try to reduce the final error rate at the receiver, we have to use more retries and this increases the average delay. In addition, we also observe that larger the number of retries, larger the end-to-end jitter (especially at higher loss rates).

This tradeoff has important implications for applications that are more sensitive to delay and jitter (such as real time audio and video) as compared to applications that require high reliability. For such applications, we can achieve a balance between the final error rate and the average delay by choosing an appropriate retry limit. For applications that require improved loss characteristics without incurring a delay penalty, we need to use FEC for loss recovery.

4.5.2 Choosing slot size

The second tradeoff that we explore is the effect of slot size on TCP and UDP throughput. Our experiments are performed on a 60 km emulated link (Figure 4.11). As discussed in Section 4.2.2, switching between send and receive slots incurs a non-negligible overhead for the Click based WiLDNet implementation. This overhead although constant for all slot sizes, occupies a higher fraction of the slot for smaller slots sizes. As a consequence, at small slot sizes the achieved throughput is lower. However, the UDP throughput levels off beyond a slot size of 20 ms. We also observe the TCP throughput reducing slightly at higher slot size. This is because the bandwidth-delay product of the link increases with slot size, but the send TCP window sizes are fixed. UDP throughput does not decrease at higher slot sizes.

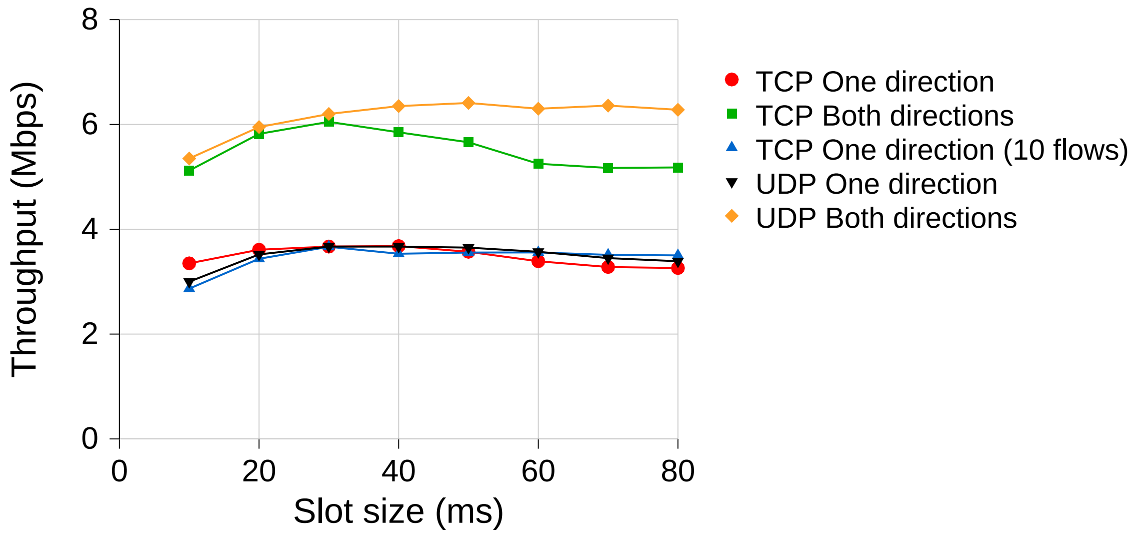


Figure 4.11: Throughput for increasing slot sizes (X-axis) in WiLDNet for various types of traffic on single emulated 60km link.

In the next experiment, we measure the average UDP packet transmission delay while varying the slot size, for several channel error rates. The results are presented in Figure 4.12; each series represents a unidirectional UDP test (1 Mbps CBR) at a particular channel loss rate with WiLDNet using maximum number of retries. Figure 4.12 shows the increase in delay with increasing slot size. It is clear that slot sizes beyond 20 ms do not result in substantially higher throughputs, but they do result in much larger delay. However, if lower delay is required, smaller slots can be used at the expense of some throughput overhead consumed by the switching between the transmit and receive modes.

4.5.3 Choosing FEC parameters

The primary tunable FEC parameter is the redundancy factor $r = (N-K)/K$, also referred to as throughput overhead. Although FEC incurs a higher throughput overhead

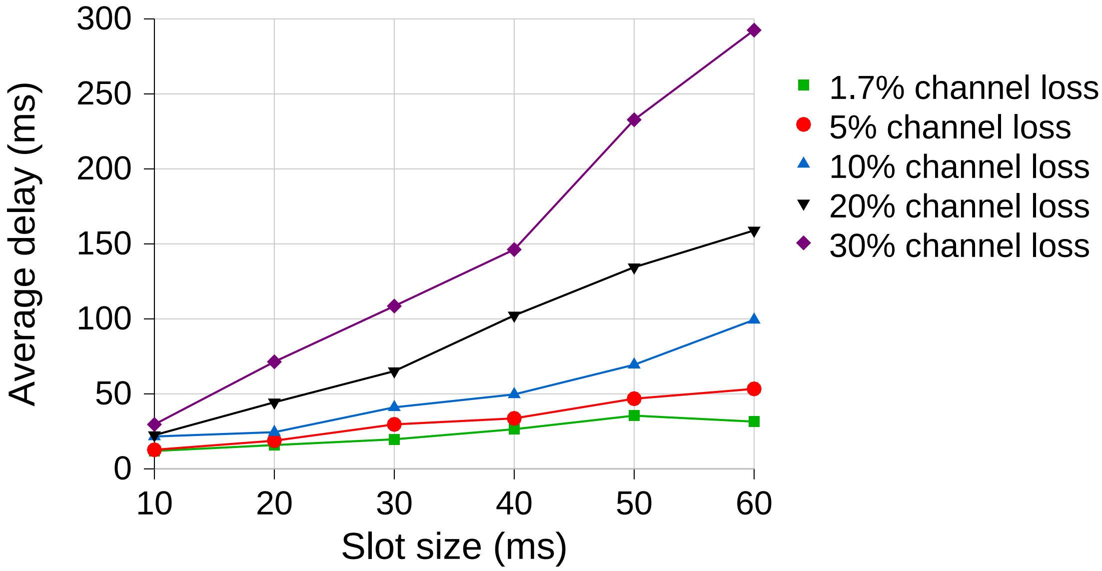


Figure 4.12: Averag delay at increasing slot sizes (X-axis) for various loss rates in WiLDNet on single emulated 60 km link.

than retransmissions, it incurs a smaller delay penalty as illustrated earlier in Section 4.4.4. To analyze the tradeoff between FEC throughput overhead and the target loss-rate, we consider the case of a single WiLD link (in our emulator environment) with a simple Bernoulli loss-model (every packet is dropped with probability p). Figure 4.13 shows the amount of redundancy required to meet three different target loss-rates of 10%, 5% or 1% as the raw channel error rates (namely p) increase. We see that in order to achieve very low target loss-rates, a lot of redundancy is required (for example, FEC incurs a 100% overhead to reduce the loss-rate from 30% to 1%). Also, when a channel is very bursty and has an unpredictable burst arrival pattern, it is very hard for FEC to achieve arbitrarily low target loss-rates.

For applications that can tolerate one round of retransmissions, we can use a combination of FEC and retransmissions to provide a tradeoff between overall throughput

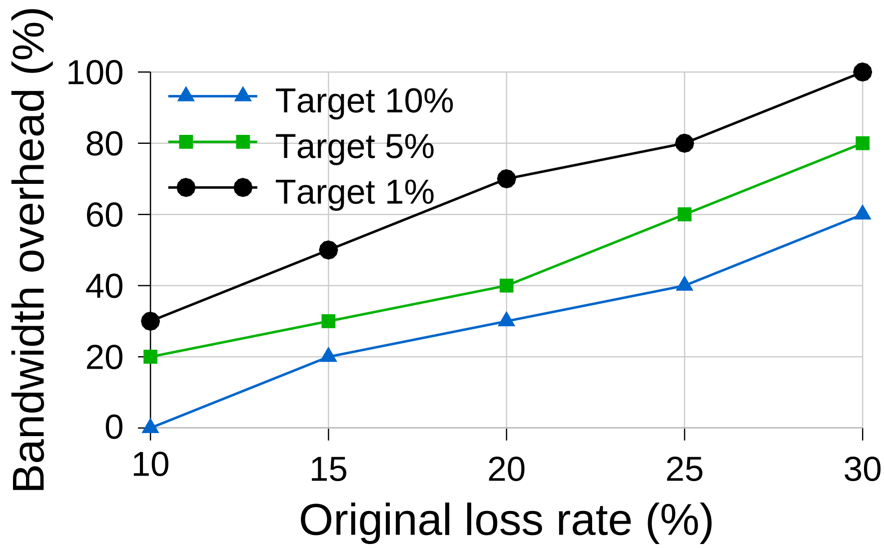


Figure 4.13: Throughput overhead vs channel loss rate for FEC on single emulated 20km link. Traffic is 1Mbps CBR UDP.

overhead, delay and target loss-rate. In the case of a channel with a stationary loss distribution, OverQoS [109] shows that the optimal policy to minimize overhead is to not use FEC in the first round but use it in the second round to pad retransmission packets. With unpredictable and highly varying channel loss conditions, an alternative promising strategy is to use FEC in the first round during bursty periods to reduce the perceived loss-rate.

4.6 Related Work

Long Distance WiFi: The use of 802.11 for long distance networking with directional links and multiple radios per node, raises a new set of technical issues that were first illustrated in [32]. Raman *et al.* builds upon this work [93, 92] and proposes the 2P MAC protocol. WiLDNet builds upon 2P to make it robust in high loss environments. Specifically, we use an implicit synchronization mechanism that works better in lossy environment, we introduce sliding-window flow control for increased througput, and we build adaptive loss-recovery mechanisms relying on bulk acknowledgments and packet-level FEC.

Other wireless loss recovery mechanisms: There is a large body of research literature in wireless and wireline networks that have studied the tradeoffs between different forms of loss recovery mechanisms. Many of the classic error control mechanisms are summarized in the book by Lin and Costello [74]. OverQoS [109] performs recovery by analyzing the FEC/ARQ tradeoff in variable channel conditions and the Vandermonde codes are used for reliable multicast in wireless environments [102].

TCP transport-aware recovery mechanisms over multi-hop wireless networks have been extensively studied in prior works [52, 63], and transport-aware link layer approaches such as Snoop [30], and I-TCP [29] have been proposed. The TCP Simultaneous-Send problem, as described in [55], is the problem when channel contention between the data and ACK packets of single flow reduces throughput, and the authors alleviate the situation by skipping TCP ACKS at the TCP receiver. However, we can avoid this contention through synchronization at the link layer, and the TCP stack on the end hosts on our network does not need to be modified at all.

To compare the WiLDNet bulk ACK recovery mechanism with recovery at a higher layer such as Snoop, we experimented with a version of the original Snoop protocol [31] that we modified to run on WiLD links. Basically, each WiLD router ran one half of Snoop, the fixed host to mobile host part, for each each outgoing link and integrated all the Snoops on different links into one module.

We measured the performance of modified Snoop as a recovery mechanism over both standard 802.11 (CSMA) and over WiLDNet with no retries. We found that WiLDNet was still 2x better than Snoop. We also saw that Snoop was better than vanilla CSMA only at lower error rates (less than 10%). Thus, this indicates that higher layer recovery mechanisms might be better than stock 802.11 protocol, but only at lower error rates.

Other WiFi-based MAC protocols: Several recent efforts have focused on leveraging off-the-shelf 802.11 hardware to design new MAC protocols. Overlay MAC Layer (OML) [99] provides a deployable approach towards implementing a TDMA style MAC on top of the 802.11 MAC using loosely synchronized clocks to provide applications and competing nodes better control over the allocation of time-slots. SoftMAC [80] is another platform to build experimental MAC protocols. MultiMAC [49] builds on SoftMac to provide a platform where multiple MAC layers co-exist in the network stack and any one can be chosen on a per-packet basis.

4.7 Future Work and Conclusion

In this chapter we design WiLDNet, a solution that addresses the shortcomings of the 802.11 protocol in WiLD settings. We implement and deploy our system in real long distance scenarios, and show that our solution is efficient, successful in avoiding interference, and provides 2–5X improvement in TCP throughput over the conventional 802.11 MAC.

In future chapters we discuss how WiLDNet was deployed in several developing countries, including Indian hospitals that use it for telemedicine and remote consultation [112]. WiLDNet was also used to set a world record, with a 382 km WiLD link delivering a bidirectional throughput of 6 Mbps [50].

Chapter 5

Capacity of WiLD Networks

In this chapter we measure the maximum achievable network capacity in WiLD networks operating on the same channel. We then quantify the gap between the throughput achieved in practice by WiLDNet and other TDMA-based MAC protocols, and the maximum throughput computed offline using complete topology and traffic demand information. We find this gap to be very large in practical topologies, and this motivates us to look for practical capacity maximization mechanisms that take into account dynamic traffic demand information.

In previous chapters we presented mechanisms to deal with inter-link interference and packet loss, embodied in WiLDNet. Our solution was successfully implemented and deployed in real WiLD networks serving many thousands of users in developing countries. Nonetheless, we found that WiLDNet, as well as other parallel efforts relying on TDMA MACs (2P [93]), feature a number of significant limitations. These limitations stem from the reliance on a TDMA schedule with fixed-time slots that does not adapt transmission opportunities to dynamic traffic variations. In order to understand the extent of these limitations, we compute an optimal link transmission schedule that maximizes network throughput given complete knowledge of the traffic demand and network topology.

5.1 Computing Network Capacity

In the following we discuss methods to compute offline link transmission schedules that optimize network throughput. We borrow from the prior work of Jain *et al.* [63] in the more general context of multi-hop wireless networks that exhibit inter-link interference. In this work, optimal link scheduling is framed as a *max-flow optimization problem*, with an additional constraint that avoids inter-link interference by enforcing that interfering links never schedule transmissions simultaneously. We adapt this generic method to interference and other constraints encountered in WiLD networks.

The inputs of the throughput-optimal link scheduling problem are the *network topology* and the *traffic demand*. Computing the link transmission schedule is challenging because wireless interference makes network links depend on one another, with the decision to transmit on one link affecting the availability of other links. Consequently, *link scheduling* and *routing* decisions are inter-dependent. One can approach link scheduling in one of the two ways: a) solve it as a joint optimization problem together with routing, or b) fix routing and find the optimum link scheduling that observers the given routing constraints. The former is optimal, while the latter is more amenable to implementation.

We look at three flavors of offline link-scheduling optimization algorithms, which make different assumptions about routing. The first is the most optimistic one, solving the joint optimization problem of routing and link scheduling, and allowing for multipath routing. The second one is similar to the first, with the only difference given by an additional *single-path routing constraint*. The third is the least optimistic (and therefore closest to practical), and assumes a fixed routing that does not change with traffic demand.

5.1.1 Multi-path routing

This scenario assumes complete knowledge of traffic and topology, and does not impose any constraints on packet routing. In this case, the optimal link transmission schedule that maximizes global network throughput can be computed using a linear programming (LP) formulation, as proposed by Jain *et al.* [63] in the more general context of multi-hop wireless networks.

We summarize this LP solution in the following. In the absence of interference, the maximum achievable throughput in a network with nodes N_G and links L_G, with the traffic demand specified by a set of source-destination pairs $F \subseteq N_G \times N_G$, can be found by solving the following max-commodity problem:

$$max \sum_{\{s,d\} \in F} \sum_{l_{si} \in L_G} f_{si}^{sd}$$

Subject to:

$$\sum_{l_{ij} \in L_G} f_{ij}^{sd} = \sum_{l_{ji} \in L_G} f_{ji}^{sd}, \forall s, d, n_i, | \quad \{s, d\} \in F, n_i \in N_G \setminus \{s, d\}$$

$$\sum_{l_{is} \in L_C} f_{is}^{sd} = 0, \qquad \forall \{s,d\} \in F$$

$$\sum_{l_{di} \in L_C} f_{di}^{sd} = 0, \qquad \forall \{s,d\} \in F$$

$$\sum_{\{s,d\} \in F} f_{ij}^{sd} \leq Cap_{ij}, \qquad \forall i,j | l_{ij} \in L_G$$

$$f_{ij}^{sd} \geq 0, \qquad \forall i,j,s,d | l_{ij} \in L_G, \{s,d\} \in F$$

Here f_{ij}^{sd} denotes the flow between source s and destination d on link l_{ij}, and Cap_{ij} represents the capacity of link l_{ij}.

In order to account for interference, we add the constraint that interfering links must not be simultaneously used. Given the *directed connectivity graph* G of a wireless network, the links that interfere with one another are specified by means of an undirected *conflict graph* C. The vertices of the conflict graph C correspond to the directed edges l_{ij} in the original graph G. There is an edge between vertices l_{ij} and l_{pq} in C if the links l_{ij} and l_{pq} cannot be activated (transmitted on) simultaneously.

In the context of WiLD networks, the conflict graph can be computed simply by enforcing that there is an edge between any two vertices corresponding to links l_{ix} and l_{yi}, where one originates in node i and the other ends in the same node i, since a node is not allowed to transmit and receive simultaneously. Fig. 5.1(a) presents an example connectivity graph G for a WiLD network, and Fig. 5.1(b) shows its associated conflict graph. Optionally, if angular separation between long distance wireless links is not enough to offer good isolation for simultaneous reception on both links (this isolation is ensured in most cases) additional edges in the conflict graph must be added.

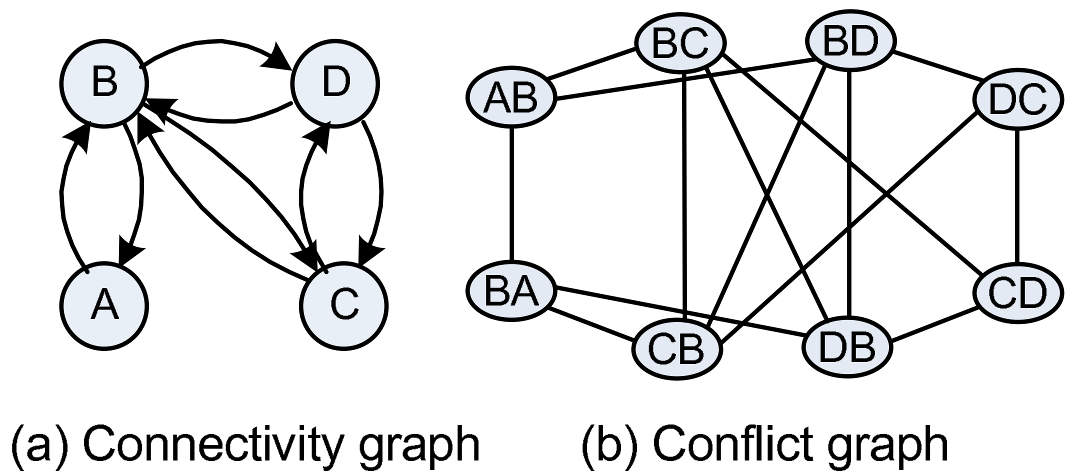


Figure 5.1: Example of a connectivity graph corresponding to a WiLD network topology, and its associated conflict graph

Having the conflict graph C, we know that vertices belonging to a given *independent set* [1] in C, which represent links in the original connectivity graph G, can be scheduled simultaneously. Therefore an *independent set* in the conflict graph corresponds to a *schedulable set* of links in the original graph. To avoid interference, the link schedule must ensure that, at any time, all the scheduled links belong to a common schedulable set.

Any link transmission schedule that alternates among schedulable link sets is a feasible one (it avoids interference). Therefore, the optimization problem becomes finding how much of the total time to spend in each of the schedulable sets, so that the bandwidth is optimized. If we have K total schedulable sets $S_1, \ldots, S_K$, let $\lambda_i, 0 \leq \lambda_i \leq 1$ denote the fraction of time allocated to set S_i.[2] We then can add the following constraints to the

[1] Given a graph, an independent set is a set of graph vertices such that there is no edge between any two vertices in the set.

[2] The fraction of time allotted to a set can be zero, *i.e.,* the set is not used in the schedule.

original LP formulation:

$$\sum_{i=0}^{K} \lambda_i \leq 1, \text{(since at most one schedulable set is active at a time)}$$

$$\sum_{\{s,d\} \in F} f_{ij}^{sd} \leq \sum_{l_{ij} \in S_i} \lambda i Cap_{ij}$$

Computing the schedule using this algorithm is optimal [63], but also $NP-hard$ (since it is $NP-hard$ to find all the schedulable sets), with exponentially many independent (schedulable) sets to consider. A good lower bound can be computed by only considering some of the *maximal independent sets* (finding all of them is also NP-hard).

We will use the solution of this LP program (for small, computationally tractable cases) as an idealized upper bound on how much throughput can be obtained given the network and traffic demand.

5.1.2 Single-path routing

The second scenario makes a first step towards a more practical solution, given that most practical routing algorithms are single-path, and many transport-layer protocols rely on the same assumption. We use this scenario to understand what can be achieved in terms of maximizing throughput under a single-path routing constraint. In the context of the LP formulation, this constraint can be posed [63] by introducing a set of binary variables z_{ij}^{sd} that indicate whether a link l_{ij} is used for forwarding traffic in the flow between s and d. We then enforce that at any node in the network, there is at most one out-going edge for a given flow:

$$\sum_{\{s,d\}\in F} f_{ij}^{sd} \leq Cap_{ij}\dot{z}_{ij}^{rd}, \forall l_{ij}, \text{where } z_{ij}^{sd} \in \{0,1\}$$

$$\sum_{l_{ij}\in L_G} z_{ij}^{sd} \leq 1, \forall i \in N_G, \forall \{s,d\} \in F$$

This makes the LP formulation an integer LP problem, which is NP-hard to solve, but tractable in practice for small enough problems.

5.1.3 Fixed routing

The previous scenario assumes single-path routing, but the routing decision is taken such as to optimize overall throughput, taking into account the traffic demand. However, most routing protocols compute a fixed routing, since adjusting routes to traffic demand is difficult. The third scenario considers the case when the network uses a fixed, shortest path routing (commonly used in practice, and useful in optimizing other routing metrics such as *delay*). In this case, we compute the routes R_{sd} for every source-destination pair $\{s,d\} \in F$ beforehand, and we change the flow-conservation constraints to the following simpler constraints:

$$f_{ij}^{sd} = f_{jk}^{sd}, \qquad \forall s,d,i,j,k|\{s,d\} \in F, l_{ij}, l_{jk} \in R_{sd}$$

$$f_{ij}^{sd} = 0, \qquad \forall s,d,i,j|\{s,d\} \in F, l_{ij} \notin R_{sd}$$

5.1.4 Whole-node transmissions

Besides the constraints for *obeying link capacities*, *avoiding interference*, and *routing scheme*, other constraints can be added in order to reflect the limitations introduced by characteristics of practical solutions.

One such example is the constraint imposed by approaches such as 2P and WiLDNet, which schedule every node to simultaneously transmit on all of its links. This approach is suboptimal, because it prevents two neighboring nodes from making simultaneous and independent transmissions to their respective neighbors. Nonetheless, we investigate this possibility for two important reasons: it is very easy to implement practical MAC schemes following such constraints, and it makes the search for schedulable sets much easier and tractable for larger network sizes. This happens because in this scenario, the schedulable sets of links can be computed simply by computing the *independent sets* of nodes in the original connectivity graph. Since the connectivity graph is much smaller in size than the conflict graph, computing schedulable sets is feasible for much larger networks.

5.2 The Throughput Gap

We use the solutions to these LP problems to present the potential for improvement over existing algorithms. Because neither WiLDNet nor other existing TDMA-style MAC protocols support single-channel operation over non-bipartite topologies, we begin by proposing a few simple ways to adapt these algorithms for use in any topology. We then compare the throughput achieved by these algorithms to the optimal network capacity.

5.2.1 Operation in General Topologies

Enforcing the network topology to be bipartite can be limiting, because it constrains the ways in which networks can be gradually extended. For example, consider the case when a new network node A is added to the network, and A has line of sight to nodes B and C. If B and C are already connected to each other, node A can only connect to one of the two (in order to maintain the bipartite constraint). For node A this implies that a) it cannot have redundant links, making network connectivity less reliable, and b) it is served at suboptimal network capacity.

Raman [91] proposes a solution to address this when several non-overlapping channels are available, by dividing the network into bipartite subgraphs operating on different channels, and using 2P on each of these subgraphs.

However, under the constraint of single-channel operation, or if we choose to use a single channel for spectral efficiency reasons, such an approach cannot be used. We therefore investigate the following intuitive ways to adapt the TDMA scheme in which nodes send or receive on all of their links for use in non-bipartite topologies:

1. *FT*: Fixed-slot TDMA according to vertex colors. First compute the minimum vertex coloring of the graph. Then nodes transmit in TDMA slots, according to their color. Colors are scheduled for transmission in a round-robin fashion, and therefore each node sends once every K slots, where K is the number of colors. For bipartite graphs (which can be colored with 2 colors), the behaviour of this algorithm is the equivalent to the that of WiLDNet.

2. *FT-CUT*: Fixed-slot TDMA over maxcut. We first compute the maximal subgraph

that is bipartite and contains all the network nodes — i.e. a *maxcut* in the original graph. We then use WiLDNet on the maxcut, keeping other links as backups.

The latter approach features two types of links: some that are used for the entire time (to either send or receive), and others that are never used in normal operation. The former uses all the links, but all of them are only used for part of the time ($2/K$ of the time, with K being the chromatic number of the network connectivity graph). Dynamic slot sizes can work with either approach and we compare the efficiency of these approaches, with and without adaptive slot sizes, later in this dissertation.

5.2.2 Throughput Comparison

We now compare the optimal throughput computed offline against the throughput achieved by existing fixed-slot approaches (FT and FT-CUT). We perform our comparison on the following topologies: a) a 20-node random graph, with an average connectivity degree of 3; b) the real WiLD topology (14 nodes and 19 links) as used in the Aravind Eye Hospital; c) a realistic WiLD topology constructed using the method presented by Raman [91]. We assume a uniform link capacity of 10 Mbps.

To measure saturation throughput (in terms of the maximum number of flows successfully accommodated by the network), we generate an amount of traffic exceeding the maximum capacity. We use CBR flows, with a bandwidth of 500 kbps. We generate unidirectional flows between random source and destination pairs.

For the offline algorithms, we solve the linear programs generating the throughput-optimal solutions using the ILOG CPLEX [4] optimizer. To evaluate the performance of the online algorithms, we perform simulations using a modified version of the Java-based network simulator developed by Jain [64]. Given that the number of flows accommodated by the network depends on the order in which we add flows, we generate 5 such random flow orderings, and for each ordering we add the flows one by one until we reach saturation. For each run, we find the point when the maximum number of flows was successfully served by the network, and we average among the results obtained in each run. We use this method to compute the throughputs for 5 random topologies for each size, and present the average of these results.

We use 6 variants of the LP optimization, as described in section 5.1, in order to illustrate the effects of constraining the routing to be single path or shortest path, but also to understand the effect of constraining nodes to simultaneously send on all their links. The latter constraint makes link scheduling easier to implement in practice (since node scheduling is easier than link scheduling), and was observed in WiLDNet and 2P.

For the variant using single-path routing and independent link scheduling, also denoted LP-SP(O), we only show the results using the Aravind topology (the smallest of our topologies), since for the other topologies the resulting integer LP formulation becomes too large.

Figure 5.2 shows the result of our comparison. As expected, we find a very large gap between the throughput achieved by practical approaches and the maximum potential throughput. Even with constraints of fixed routing and simultaneous transmission on all the links of a node, the LP solution computed offline outperforms practical solutions by more than a factor of two.

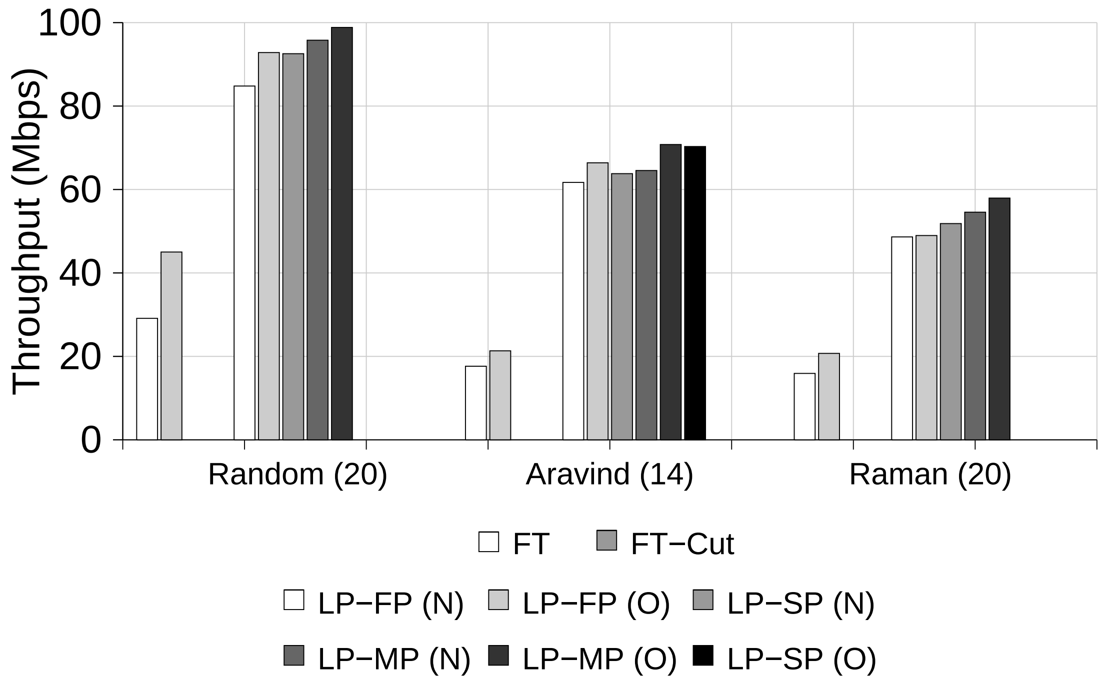


Figure 5.2: Throughput comparison for the following algorithms: 1)LP-MP (O): LP, multipath; 2) LP-SP(O): LP, single path; 3)LP-FP (O): LP, fixed path; 4)LP-MP (N): LP, multipath, nodes send to all links simultaneously; 5)LP-SP (N): LP, single path, nodes send to all links simultaneously; 6)LP-FP (N): LP, fixed path, nodes send to all links simultaneously; 7)FT-Cut: FT over maxcut; 8)FT: FT over the original topology

Another important finding is the fact that *fixed routing* and *simultaneous transmission on all of a nodes's links* are acceptable constraints, since the network throughput achieved observing these constraints is very close from the optimal throughput, achieved using multipath routing and independent link scheduling (LP-MP(O)). This observation holds for all our topologies, with the most constrained solution (LP-FP(N)) being within 15% of the optimal one (LP-MP(O)).

Comparing among practical algorithms, we also note that the algorithm running over the maxcut of the network topology (FT-CUT) outperforms FT, which runs over the entire network topology. This happens because if we use FT, and the graph has a *chromatic*

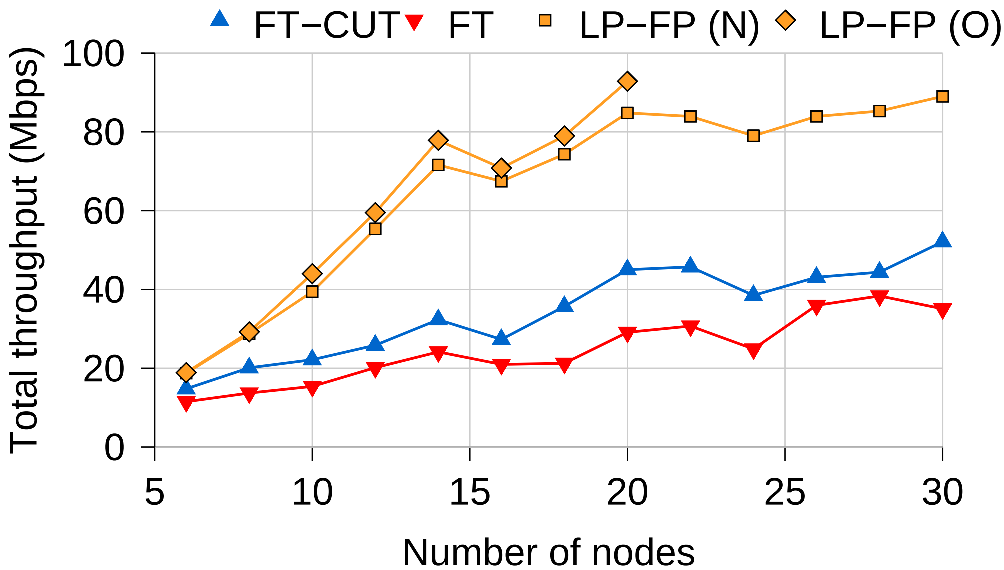


Figure 5.3: Maximum throughput for *unidirectional* CBR flows for various protocols with increasing network size. These are random topologies (avg. deg:3).

number K, each node gets to send only once every K slots, which is inefficient for $K \geq 3$.

In the following experiment we investigate whether the observed gap between practical and optimal link scheduling is maintained for networks of other sizes. Figure 5.3 compares the network throughput delivered by the practical scheduling algorithms (FT and FT-CUT), and the offline schedules computed using fixed routing, in networks of increasing sizes (# of nodes). Since independent link scheduling (LP-FP(O)) is too computationally intensive beyond a network size of 20, we also solve the (smaller) problem using node scheduling (LP-FP(N)), and use this solution as an approximation for LP-FP(O).

For each network size, we generate 5 random topologies with average connectivity degree of 3, and use traffic similar to the one used in the previous experiment. Every measurement point corresponds to the throughput averaged among all the topologies of the same size, and for each topology we run the experiment 5 times, each time with different traffic.

We find that, at small sizes (*e.g.,* 5 nodes), the difference between practical and optimal approaches is small, but this difference increases quickly as we exceed 10 nodes, and remains high afterward.

These findings show that existing practical approaches are inefficient over a large spectrum of network topologies, which motivates the development of a new more dynamic MAC layer based on the insights presented above.

5.3 Related work

Raman *et al.* [91] propose a TDMA scheduling that uses coloring to divide a graph into bipartite subgraphs, and then use the 2P approach [93] in each subgraph; the coloring is done such that the mismatch between the desired and the achieved upload/download ratio for each link is minimized. Spatial-reuse TDMA has been considered by researchers for medium sharing in packet radio networks. Both link scheduling [43, 81], where individual graph links are scheduled, as well as broadcast scheduling [42, 45, 96] where graph nodes are scheduled, have been considered. In these, the goal is to come up with a transmission schedule of time slots such that all links/nodes are scheduled within a minimum number of time-slots (i.e., with minimum schedule length). This problem is NP-complete for both the link and node scheduling variants [95], and efforts have focused on other issues such as a distributed implementation [42, 43, 45, 96]. Researchers have also considered restricted classes of graphs [95], since the problems are NP-Complete on general graphs.

Another dimension that has been considered is scheduling to adapt to current/expected traffic patterns [28, 81, 113]. Similar to the Raman's work [91], bipartite graphs also figure in [28, 113]; in these studies it is shown that non-bipartite graphs lead to significantly less efficient solutions.

5.4 Conclusion

In this chapter we use linear optimization techniques to quantify the network capacity of typical WiLD links, using complete knowledge of topology and traffic demand. We then compare the maximum achievable throughput with the throughput achieved by simple TDMA-based practical approaches, agnostic of traffic demand. We find that the additional performance achievable by incorporating knowledge of traffic demand can be very large, motivating the use of a traffic-informed link scheduling approach, which we cover in the following chapter.

Chapter 6

JazzyMAC: Maximizing Capacity in WiLD Networks

.

In this chapter we describe JazzyMAC, a second-generation MAC protocol for WiLD networks, which uses purely local traffic information to adapt the link transmission duration on various links. This brings significant bandwidth improvements over WiLDNet and other TMDA-style protocols, often as large as 50–100%, especially for the common case of asymmetric traffic. It also significantly reduces the gap between throughput achievable in practice and the capacity limits of the network. Finally, the use of dynamic slot sizes enables better tradeoffs between throughput and delay and can often provide lower latency for the same bandwidth.

The ideas behind JazzyMAC were the fruit of the collaboration with Rabin Patra, Sonesh Surana, Sylvia Ratnasamy, Lakshminarayanan Subramanian and Eric Brewer, and were previously published [79]

As we have seen in the previous chapter, using traffic information to inform link scheduling is essential in achieving superior throughput. Therefore, we start by looking into practical ways in which local traffic information can be used to inform transmission scheduling in order to increase link utilization and decrease per-hop delay (section 6.1). We then incorporate these ideas in the design of JazzyMAC (section 6.2), and use detailed simulations to evaluate the performance of this protocol for various traffic types and in various network topologies (section 6.3).

6.1 Opportunities for Improvement

In this section we discuss several opportunities for improving the performance achieved with WiLDNet and similar TDMA-style MAC approaches.

6.1.1 Improving Throughput

WiLDNet and other existing MAC solutions for wireless long-distance networks feature a static TDMA slot allocation, which is simple, robust, and easy to deploy in practice. However, throughput of these networks can be drastically increased by exploiting two important avenues.

1) Adapting to Traffic Demand: Current MAC solutions for WiLD networks feature a static TDMA slot allocation. This approach is simple, robust, and easy to deploy. However we conjecture that higher throughputs could be achieved by having nodes adapt their slot sizes by using current traffic information. The following examples illustrate this intuition:

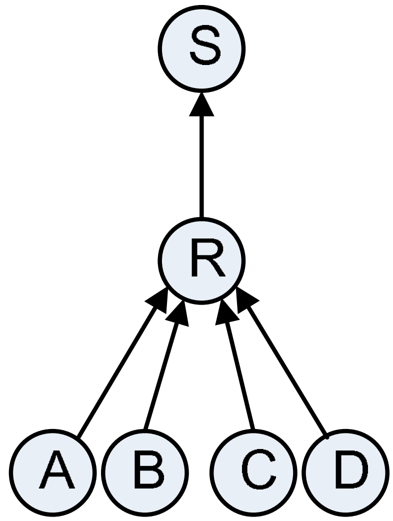


Figure 6.1: Example fork topology

Example 1: Single link: Consider the simplest case of a network with a single link between nodes A and B and assume that the traffic demand only exists from A to B. In this scenario, the highest throughput would be achieved by configuring the link to transmit from A to B for (almost) the entire time. This can be achieved by allocating large transmit slots in the direction $A \rightarrow B$, and very short transmit slots in the reverse direction. If subsequently the direction of traffic flow is reversed, then the optimal slot allocation would correspondingly change, with longer slots from B to A. If we were to use such an adaptive approach, the unidirectional traffic could always be served at close to the full link capacity. Unfortunately, approaches using fixed per-link slot sizes cannot deliver similarly high throughputs. Instead, in these approaches, the link is always scheduled to transmit for $x\%$ of the time in direction $A \rightarrow B$ and $1 - x\%$ in the reverse direction, with a typical setting of $x = 50\%$.

Example 2: A fork *topology*: Figure 6.1 illustrates yet another example. In this scenario, we have a sink node S, and several source nodes A,B,C, and D connected to the sink through relay node R. Let us assume all links have the same datarate, and analyze the optimal slot size allocation for relay R. If only one of the sources (say A) sends traffic to the sink, the slot allocation that maximizes throughput is the one in which node R has equally sized transmit and receive slots. In this case, R receives data for 50% of the time, and relays this data for the remainder 50% of the time. Now assume that we have two sources sending to S. In this case, the bandwidth-optimal solution would be to have R receive for 1/3 of the time (from both senders), and then relay this data to S in the remaining 2/3 of the time. Thus, R would have a transmit slot twice as long as the receive slot. Similarly, if all four sources are sending traffic, the best scenario would be the one in which the transmit slot at R is 4 times longer than its receive slot.

In each of the previous examples, a simple strategy to take advantage of local traffic information is to monitor the volume of traffic on outgoing links and then adapt the size of TDMA transmit slots to be proportional to the volume of traffic to be transmitted. This is the fundamental intuition behind JazzyMAC.

2) Allowing neighboring transmissions that overlap: Current MAC protocols such as 2P and WiLDNet require that a node maintain all of its links in transmit mode for the same (fixed) time duration. However, there are several situations where this can be needlessly inefficient. For example, consider the topology presented in Figure 6.2, in which traffic flows are represented by arrows. In this topology, since nodes A and B are neighbors, they can never simultaneously operate in transmit mode (as per current protocols). However, it is possible that the traffic demand is such that A only needs a portion of its transmit slot to B (from say, $t = 0$ to $t = 6$). In this case, we can allow B to start transmitting

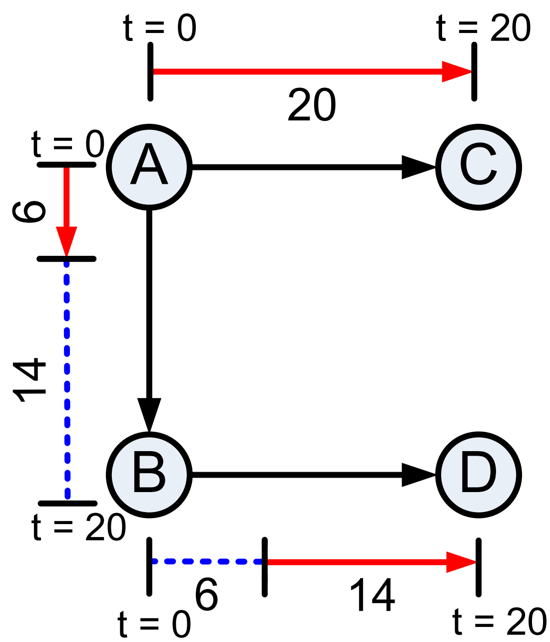


Figure 6.2: Overlap of transmissions

to a third node (D) at an earlier time ($t = 6$) rather than having to wait until the end of A's transmission slot ($t = 20$). This means that, for a portion of their transmission slots, both A and B can transmit simultaneously while still respecting all the invariants required to avoid interference. Such *neighboring-but-independent* transmissions have the potential to further increase network channel utilization and our JazzyMac protocol is designed to exploit these opportunities.

6.1.2 Improving the bandwidth vs. delay tradeoff

Besides network throughput, another issue of particular interest in long-distance networks is the per-packet delay. Although a large fraction of the popular applications over WiLD networks are delay-sensitive such as telemedicine [112] and VoIP [21], existing solutions introduce significant per-hop delays.

One of the main reasons for these large delays is the TDMA approach adopted by current protocols, and the fact that practical constraints prevent TDMA slot sizes from being particularly small. This happens because switching between a *sending slot* and a *receiving slot* cannot be done instantaneously; it requires a non-zero *guard time* in which packets are neither transmitted nor received [84].

A lower bound for the size of this guard time is the round-trip propagation delay, which is significant in long-distance networks. For example, a $75km$ link has a round-trip delay of $0.5ms$. Also, in order to maintain synchronization in the network, the size of the guard time is constrained by the round-trip delay of the *longest* link in the network [93].

Besides propagation delay, existing implementations feature additional constraints that make this guard time much larger in practice. This is especially true of implementations on top of WiFi hardware, because the TDMA mechanisms are not supported in the PHY layer (and firmware), but implemented either in the WiFi driver or above it. This introduces additional (sometimes variable) delays between the time a packet is sent from the driver and the actual time that the packet is sent over the air. Because of these inefficiencies, the guard time in WiLDNet is $3ms$.

Having a large slot guard time t_{switch} limits the minimum slot size. This in turn affects the average per-hop delay, which is proportional to the slot time. For example, the average delay when very lightly utilized is

$$delay_{avg} = \frac{(t_{switch} + t_{slot})^2}{2(2t_{slot} + t_{switch})} \approx \frac{t_{slot}}{4}, \tag{6.1}$$

while the maximum per-hop delay at close to saturation utilization is $\approx 2t_{slot}$. Figure 6.3

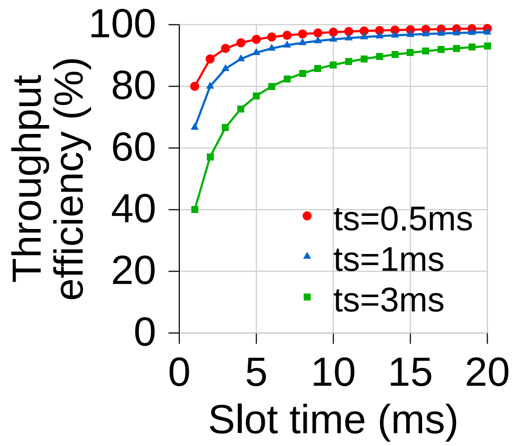


Figure 6.3: Throughput efficiency vs. slot time

plots the bandwidth as a function of slot time, assuming guard times t_{switch} of $0.5ms$, $1ms$ and $3ms$.

Since existing approaches use fixed slots, the bandwidth vs. delay tradeoff is fixed, usually to a value that favors bandwidth while sacrificing delay (*e.g.,* a $10ms$ slot). In small deployments this is acceptable, but with larger-scale networks the average hop count increases, the end-to-end delay penalty could become prohibitive for interactive applications.

We believe that dynamic slot adaptation can alleviate this problem. This would allow for the bandwidth-delay tradeoff to be negotiated differently for different links, taking into account traffic demand. Links seeing low utilizations could utilize small TDMA slots and deliver low per-hop delay, since maximum link bandwidth would not be necessary to serve the traffic demand. Conversely, for highly utilized links the tradeoff could be shifted towards supporting higher maximum bandwidth, by using larger slots (*e.g.,* $20ms$). This approach would allow the network to achieve the best of both worlds: small average delays and maximum bandwidth efficiency when required.

6.2 JazzyMac Design

This section presents JazzyMAC, a novel medium access control protocol for long-distance wireless networks that addresses the limitations identified previously. Specifically, JazzyMac makes the following key improvements:

- *Adaptive slots:* rather than require fixed-length transmission slots, JazzyMac allows each link to adapt the length of its transmission slots dynamically, based on locally observed traffic load. Adaptive slots lead to more efficient bandwidth allocations and greater flexibility in navigating the tradeoff between throughput and delay.
- *Allow parallel neighboring-but-independent transmissions:* the protocol is specifically designed to allow neighbors to proceed with parallel independent transmissions, as exemplified in section 6.1, which contributes to increased throughput.
- *Generalized topologies:* scheduling in JazzyMac does not require that the topology be bipartite, making the protocol applicable to arbitrary topologies.

JazzyMac achieves the above using simple and fully distributed algorithms that rely only on readily available local state. This makes JazzyMac practical for implementation in existing radios and hardware platforms.

6.2.1 Protocol Description

We now describe the JazzyMac protocol. Every node A is associated with a node-wide **mode** of operation, which can be either transmit (`TX`) or receive (`RX`). Each network link AB is associated with a **token**, T_{AB}, that is at all times in the possession of either node A or B and only the node holding the token can transmit on the associated link. In addition, each token is associated with a **timeout value**, v_{AB}, that controls *when* the node holding the token is allowed to transmit over the associated link. Finally, we introduce a network-wide parameter `max_slot` that bounds the maximum length of any transmission slot.

Given the above protocol state, the basic operation of JazzyMac is guided by the following four rules:

1. *token exchange rule:* When a node (say) B has completed its transmission over link AB, it computes a timeout value v_{AB} that estimates the time in the future when node B will be willing to receive traffic *from* A (we describe how v_{AB} is computed shortly). Node B then hands the tuple (T_{AB}, v_{AB}) to node A. If node A receives this token at time t, we say that token T_{AB} is *valid* after time $t + v_{AB}$.

2. *mode rule:* A node B that is in receive mode can transition to transmit mode only when it holds the token (whether valid or not) for *all* its links. Likewise, a node returns to receive mode when it has released the tokens for *all* its links.

3. *transmission rule:* A node A can transmit over link AB only when the following two conditions are true: (1) node A is in transmit mode and (2) node A holds token T_{AB},

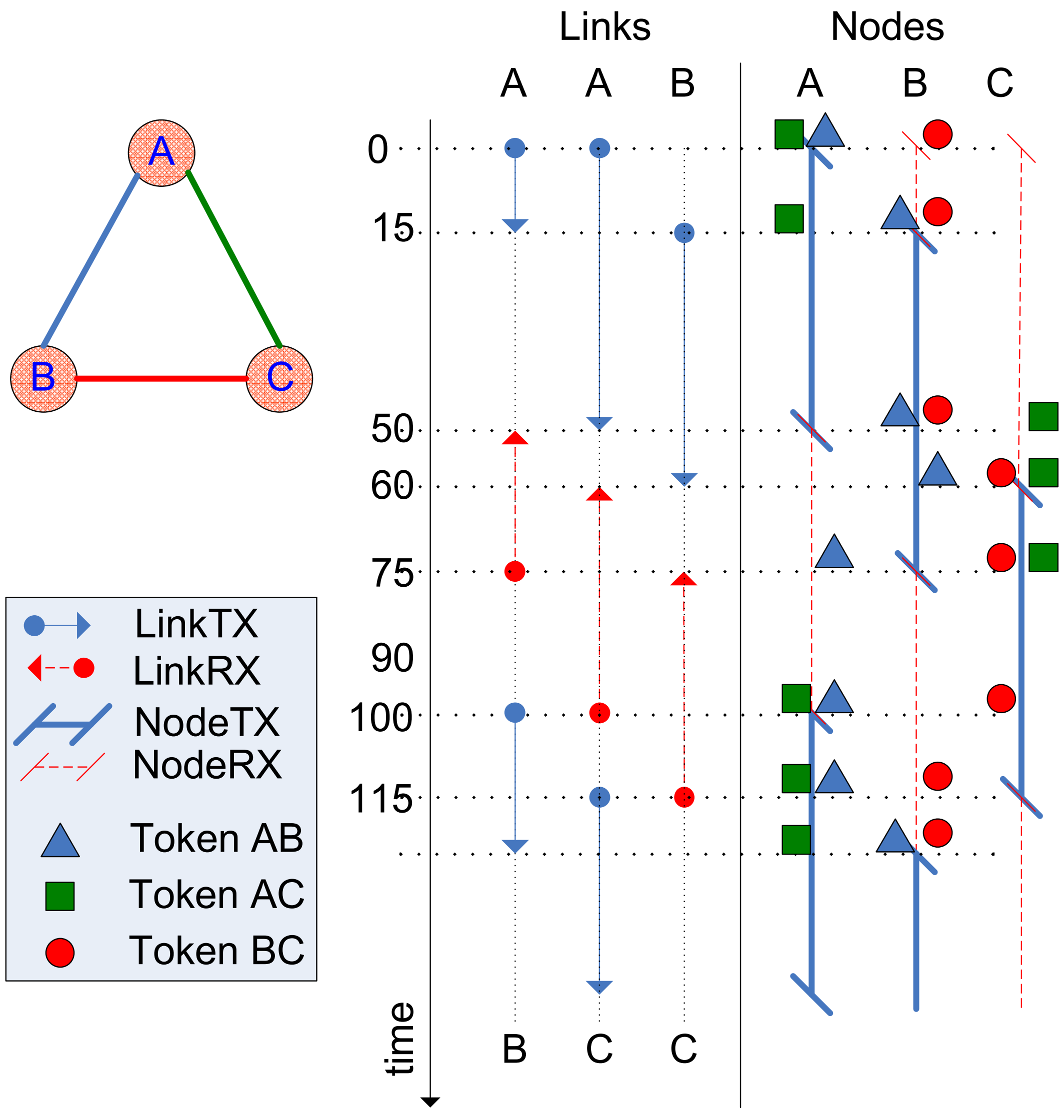


Figure 6.4: Scenario featuring three nodes and three links. The figure presents the network topology, and illustrates how data is sent and received on each of the network links. The figure also shows how nodes transition between TX and RX states, as well as the distribution of the link tokens among the three nodes.

and T_{AB} is valid. (Note that, by the mode rule, A being in transmit mode ensures it has T_{AB}).

4. *slot rule:* A node A can transmit on link AB for no longer than `max_slot` time units.

Figure 6.4 illustrates the operation of JazzyMac for a simple 3 node scenario.

Assume that node A initially holds the tokens for links AB (T_{AB}) and AC (T_{AC}), while node B holds the token T_{BC}. The timeline proceeds as follows.

1. At $t = 0$, since node A has all the tokens, it is in node-wide TX mode and starts transmitting on both its links.

2. At $t = 15$, A's transmission to B ends, and token T_{AB} is passed to B. Note that A's transmission to C lasts much longer (50 time units). Therefore, T_{AB} is passed with a timeout $v_{AB} = 35$, the additional time until node A finishes its transmission to C.

3. Now, at $t = 15$, node B has all its tokens and hence transitions into a node-wide TX mode. However only token T_{BC} is valid, and therefore B starts transmitting only to node C. In prior MACs, to avoid collisions, B would transmit to C only when A finished *all* its transmissions. With JazzyMAC, we can permit such *neighboring-but-independent* transmissions without resulting in any collisions.

4. At $t = 50$, A releases token T_{AC} and transitions into node-wide RX mode. B's token T_{AB} becomes valid and it starts transmitting over link AB.

5. At $t = 60$, C transitions to TX, and so on.

Note that the use of a node-wide mode of operation controlled by the above rules ensures that JazzyMac respects the fundamental limitation of inter-link interference in WiLD networks. Specifically, a node A never transmits on a link AB while receiving on another link (say) CA and also never transmits on link AB while node B is itself transmitting on some link BC.

The use of token timeouts v_{AB} allows neighboring nodes to simultaneously transmit provided these transmissions are independent. For example, in the above scenario, nodes A and B can simultaneously transmit between times 15 and 50. This allows JazzyMac to move beyond the strict alternation imposed by solutions based on bipartite scheduling. In addition, we show in Section 6.2.3 that the above rules suffice to ensure that JazzyMac is deadlock and starvation free.

We now address two additional questions not addressed by the above protocol description.

1. *How long does a link transmission last?* The `max_slot` parameter sets the upper limit on slot lengths. To select a good slot length, JazzyMac selects a slot length based on its locally observed traffic demand. Our implementation uses the *per-link outgoing queue length* as a measure of traffic demand on the link in question. Let tt_{AB} denote the estimated time to transmit all the the packets queued for transmission over link AB. The slot length for link AB is then selected to be the minimum of tt_{AB} and `max_slot`. This policy allows busy links to transmit longer and less used links to transmit for shorter periods, as demanded by the current traffic, while the `max_slot` bound ensures a minimum per-link bandwidth and a maximum per-link packet delay (as will be discussed in section 6.2.3).

2. *How are timeout values v_{AB} calculated?* As described above, when node A finishes its transmission on link AB, it must calculate a timeout period v_{AB} that estimates the time when node A exits transmit state and is ready to receive traffic from B. The difficulty is that in order to estimate v_{AB}, node A must estimate the time *in*

```
Event:  Receive token T_l
```

$l.state = preTX$
$T_l.owner = A$
$T_l.timeoutDone = now + T_l.timeout$
Schedule Timeout done event for token T_l
$l.state = preTX$
if, for every $ll \in links(A)$, $T_{ll}.owner = A$
    $A.state = TX$

```
Event:  Timeout done for token T_l
```

$l.state = TX$
$l.endTxTime$ = computeLinkTXDuration(l) + now
l.startTransmission()
Schedule TX done event for link l

```
Event:  TX done for link l
```

$l.state = doneTX$
if, $(\forall ll \in links(A), ll \neq l)$: $ll.state \neq doneTX$
    // l is first link to finish TX
    $A.endTXTime$ = computeNodeTXEndTime(A)
$T_l.timeout = A.endTXTime - now$
give T_l to peer

`computeNodeTXEndTime ()` {

$$\text{return} \min_{l \in links(A)} \begin{cases} l.endTxTime & , l.state = TX \\ min\{max\_slot, QLen(l)\} + & , l.state = preTX \\ \quad + \, l.timeoutDone & \end{cases}$$

}

`computeLinkTXDuration(l)` {
    $maxTXEnd = min\{max\_slot, QLen(l)\}$
    $if\ \exists\ ll \in links(A), \text{s.t. } ll.state = afterTX$
        $maxTXEnd = min\{maxTXEnd, A.endTXTime\}$
    return $maxTXEnd$
}

Figure 6.5: Protocol description.

the future when it will be done transmitting on *all* its links. We implement this by estimating a remaining-transmission time for each link individually and setting v_{AB} to the maximum of these estimates. For the links that are done transmitting, the estimated pending transmission time is zero; for links that are already transmitting, the overall slot time is already known (calculated using #1 above) and hence the remaining transmission time is known. For links over which transmission hasn't yet begun (*e.g.*,, if the token for the link is still inactive), we estimate the remaining transmission time as the sum of the time left to the activation of the link token and the time required to transmit packets currently buffered at the link's outgoing queue. After estimating all the per-link transmissions end times, the latest of these times is selected as v_{AB}, and subsequently advertised to peers when exchanging tokens. Once the end of the node-wide TX has been established, all links will make sure not to transmit past this time.

The detailed protocol operation is presented in Figure 6.5. This completes the description of the basic JazzyMac protocol operation. In addition, we must specify *a)* how is the protocol bootstrapped (in terms of the initial token assignment), and *b)* how does JazzyMac recover from token losses and node failures. We describe our bootstrapping protocol in the following section, and discuss recovery mechanisms in Section 6.2.4.

6.2.2 Protocol Bootstrapping

The protocol liveness and efficiency depend on the initial assignment of link tokens. Although the long-term functioning of our protocol is distributed and requires only local

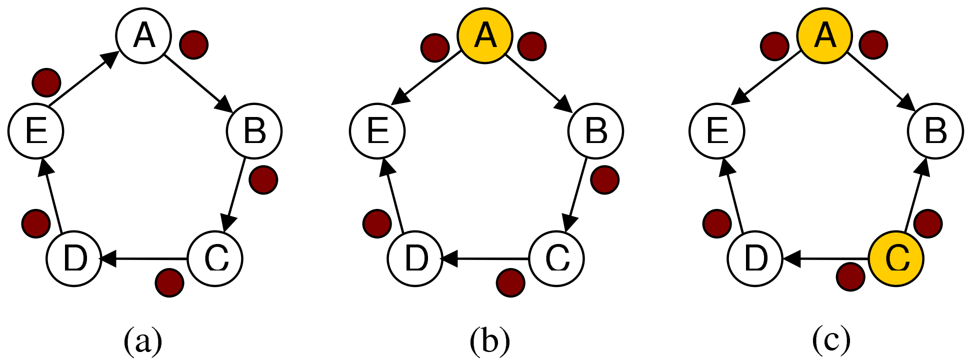


Figure 6.6: Example initial token assignments

information, the initial token assignment will be computed globally during the network planning phase (in future work we plan to investigate distributed coloring in dynamically changing networks).

In order to illustrate the effect of different initial assignments, let us examine different possibilities for the 5-node cycle, as presented in Figure 6.6. Assigning a link token is similar to establishing an initial direction for the given link, as indicated by the arrows in the figure. In this example, we make the simplifying assumption that all transmission slots have the same length. Some possible initial states are:

- If we decide to start by giving one token to each node, the protocol will be in a deadlock situation, since none of the nodes can proceed with their transmissions (Figure 6.6(a)).

- If we begin in the state in which only one node (node A) has all its tokens (Figure 6.6(b)), then node A sends first, followed by B, then by C, then D, then E and finally A again, *i.e.,*one node at a time. Thus, each link transmits for 20% of the time in each direction, and is idle for 60% of the time (given equal-sized transmit slots).

- If we begin with the initial assignment presented in Figure 6.6(c), where at the beginning both nodes A and C can immediately (and simultaneously) start transmitting, then nodes A and C go together, followed by the pair B and E, then by A and D, then C and E, and so on. The sets of nodes that send at one time keep changing, with 2 nodes always transmitting simultaneously. In this scenario (which is the optimum for fixed slots), links send for 40% of the time in each direction.

Thus we see that the steady-state performance of JazzyMac is determined by the initial protocol state. We therefore aim to assign an initial state that allows JazzyMac to ensure and maintain the following correctness and performance-related properties:

- *a)* deadlock-free operation
- *b)* starvation-free operation (every node gets the opportunity to send),
- *c)* a lower bound on the fraction of time in which a link can send in each direction (provided that the link requires this much time for transmission), and
- *d)* an upper bound on the per-link packet delay time.

We therefore propose the following bootstrapping algorithm:

1. Color the vertices of the network graph with the minimum number of colors K such that no two adjacent vertices have the same color.
2. The tokens are assigned to the link end that has the lowest color (the two ends must be colored differently).

6.2.3 JazzyMac Properties

In the following we prove that JazzyMac is deadlock-free (assuming the bootstrapping strategy introduced earlier), and that it observes a set of performance guarantees in terms of *link utilization* and *per-hop maximum packet delay*. In the interest of space, we only give the formal proof for the simpler case assuming fixed slot sizes, and provide the intuition for why the same properties hold for the general case of dynamic slot sizes.

Fixed slot size case: In this simplified case, time can be regarded as a sequence of equally sized time slots.

For our proof, we introduce the following abstraction that describes the protocol in a manner equivalent to our token-based description. Imagine that each node has a non-decreasing sequence number. Let $S_i(A)$ be the sequence number of node A in slot i, and let $T_i \subseteq G$ be the set of nodes transmitting during slot i. In the initial slot, the sequence number of every node is equal to the vertex color used to bootstrap the algorithm: $S_0(A) = color(A)$. After nodes transmitting in slot i finish sending, they recompute the value of their sequence number to be one larger than the maximum sequence number of their neighbors:

$$S_{i+1}(A) = 1 + \max_{X \in neighbors(A)} S_i(X), \forall A \in T_i$$

while the non-transmitting nodes $B \notin T_i$ keep their sequence number unchanged ($S_{i+1}(B) = S_i(B)$).

Using these sequence numbers, the condition to be fulfilled by node A in order for A to belong to the set T_i (meaning that A has the tokens for all its links) can be expressed as:

$$A \in T_i \iff S_i(A) < S_i(N),\ \forall N \in neighbors(A) \tag{6.2}$$

.

We continue by stating the following property.

Property 1 *During any time slot, the difference in sequence numbers between any two network nodes remains strictly smaller than the number of colors used for graph coloring:*

$$\max_{A \in G} S_i(A) - \min_{B \in G} S_i(B) \leq colors(G) - 1 \tag{6.3}$$

Proof By induction. We use the initial assignment of sequence numbers as the base case, and for this base case property (1) holds, because all nodes have sequence numbers between 1 and the maximum number of colors. For our inductive step, we assume that the property holds in slot n, and we must prove it for slot $n+1$. Since in slot n there will be at least one node that will have a sequence number smaller than the ones of its neighbors, $T_n \neq \{\}$. Also, the set M of nodes that have the *minimum* sequence number in the entire network is a subset of T_n. Every node $A \in T_n$ will transmit and then set its sequence number to be $1 + \max_{neighbors(A)} S_n(A)$. Since $M \subset T_n$, and all the sequence numbers of nodes $A \in T_n$ increase by at least one, it means that in slot $n+1$:

$$\min_{B \in G} S_{n+1}(B) \geq \min_{B \in G} S_n(B) + 1 \tag{6.4}$$

On the other hand, $\max_{N \in neighbors(A)} S_n(N) \leq \max_{P \in G} S_n(P)$, and therefore the maximum sequence number in the network will not increase by more than one:

$$\max_{B \in G} S_{n+1}(B) \leq \max_{B \in G} S_n(B) + 1 \tag{6.5}$$

From (6.3,6.4,6.5) it follows that:

$$\max_{A \in G} S_{n+1}(A) - \min_{B \in G} S_{n+1}(B) \leq colors(G) - 1 \tag{6.6}$$

, which concludes our proof.

Property 2 *The protocol does not result in deadlock or node starvation.*

Proof Knowing that property (1) holds, it becomes obvious to show that there is no starvation. This follows from the fact that, at every slot, the minimum sequence number in the network increases by at least one (as previously shown) and therefore in any K consecutive slots, the minimum sequence number increases with at least K. But since, at any time, all the nodes have sequence numbers that differ by at most K, we can conclude that every node must transmit at least once every K slots. Therefore *none of the nodes will starve*. This also implies that there is also no *deadlock*.

The proof above entails the following properties:

Property 3 *Every node can choose to send on each of its links for at least* $1/K$ *of the link capacity.*

Property 4 *The maximum delay between two consecutive opportunities to send on any link is smaller than* $K \times max\_slot$

These two properties establish performance guarantees, the former introducing a lower bound on link utilization, and the latter introducing an upper bound on per-link delay.

Dynamic slot sizes The properties above also hold for the general JazzyMac protocol, and we provide a brief intuition for it here. The first observation we make is that a node using variable slots goes through the same sequence of node-wide TX and RX states as when using fixed slots. Furthermore, the token exchanges performed by a node during a particular TX or RX state is also same as in the fixed slot case. The difference between the two scenarios is given by the fact that, with variable slots, nodes have the option to give up tokens *earlier* than in the fixed case. These observations can be used to show that a particular token exchange in the variable slot case can only happen earlier than the same exchange in the fixed slot case. Therefore, at any time from the beginning of operation, each link would have had at least as many opportunities to transmit as in the fixed slot size case. This means that the protocol does not suffer from starvation, and obeys similar bandwidth bounds.

6.2.4 Dealing with Loss

In the previous section we presented the behaviour of the JazzyMac protocol under ideal conditions, in which packet losses, link and node failures do not happen. In the following we will briefly discuss how can JazzyMac be provisioned to deal with some of these possible occurrences.

Even though JazzyMac eliminates interference at co-located radio, other sources of packet loss such as *external interference* can still cause packet loss in long distance links [40, 107]. This can lead to *loss of link tokens*, which will affect the functioning of our protocol.

Consequently, any JazzyMac implementation should take precautionary measures in order to minimize the probability of losing tokens. There are several way to make the protocol more resilient to such occurrences, including piggybacking tokens on several data packets, and sending multiple copies of the token in small packets.

However, in the unlikely event that the loss still occurs, our protocol must recover properly. This is a delicate issue: simply assuming token receival after waiting for a certain *timeout* period is not adequate, because it breaks the inter-node ordering established during bootstrapping, possibly leading to starvation or low-performance steady-state operation. For example, consider the chain A-B-C-D, and assume the loss of token T_{AB}. This will prompt B to wait, which in turn would prompt C to wait for token T_{BC}. Now if node C assumes to have received token T_{BC} after a timeout, we arrive into a situation where both nodes B and C believe that they hold token T_{BC}. Moreover, if node C goes ahead to transmit to its neighbors, the ordering between nodes is broken. In order to maintain the original inter-node ordering, we must make sure that the lost tokens are recovered, while the rest of the token exchanges remain unaffected.

We propose a solution that involves adding a sequence number S_{AB} to each token T_{AB}, which is set to 0 during bootstrapping. At every valid exchange of T_{AB}, S_{AB} is incremented. The solution works as follows: If a token T_{AB} is lost, the recipient (B) will wait for it, which will prompt other nodes, including A, to wait as well. After a *timeout* given by the maximum time between successive link transmissions ($K \times max\_slot$), every node will resend the tokens they have sent last. Duplicate tokens (that have previously been received) will be ignored, and the lost token (resent by A) will be properly recovered.

A problem with this approach is that simultaneous token retransmissions by several hosts can interfere with each other or other packets. To minimize the probability of such occurrences, the tokens can be sent in small packets, at random intervals after the timeout and retransmitted periodically until successful.

The same sequence numbers can be used to detect a link (or node) that is permanently down: if a node A does not receive any retransmitted tokens on a quiescent link AB for a long time, the link is marked *down*. From that moment on, A will not wait for tokens on the link AB. Instead, it will transmit a copy of the token during every TX state, and assume its instantaneous return. In the event that a node dies, all its links will be individually marked as being down.

When a node B re-joins the network after a period of inactivity, it first listens on all of its links (to verify that they are still active), and then advertises its presence by sending special *join request* packets on all its active links. These requests are repeated periodically, until a response is received. Upon receiving a join request, the neighboring nodes will mark the respective link active again, and respond by sending the link token to B. Upon receiving all the link tokens, B will resume normal operation. As in the case of token retransmission, join requests can create an acceptable amount of interference.

6.3 Evaluation

In this section we evaluate the performance of JazzyMAC over a range of topology types and sizes, and various patterns of traffic. Our findings show that:

1. JazzyMAC significantly improves the maximum network throughput achievable by existing approaches: 15–100% in typical topologies, depending on traffic.

2. The throughput improvements are consistent across many topologies and traffic patterns. Improvements are highest in asymmetric traffic patterns, when dynamically sized slots can offer much better bandwidth allocation compared to fixed-slot approaches.

3. Our protocol more than halves the gap between previously demonstrated throughput and the optimal network throughput computed with idealized slots (no switching overhead).

4. Dynamic slot sizing improves the delay-throughput tradeoff, offering increased maximum throughput when required, and decreased packet delay at average network utilizations.

5. Having the ability to operate over non-bipartite topology is significant, and allows throughput increases as large as 80% with asymmetric traffic distributions.

6.3.1 Methodology

In our experiments we compare the performance of fixed TDMA (FT) approaches such as WiLDNet and 2P against JazzyMAC(JZ). We compare these approaches running

	Fixed Slots	Adaptive Slots
Original Graph	Fixed TDMA (FT)	JazzyMAC (JZ)
Maxcut	Fixed TDMA on cut (FT-CUT)	JazzyMAC on cut (JZ-CUT)

Table 6.1: Algorithm and topology combinations

on both the original network connectivity graph (FT and JZ), and the maximum bipartite subgraph (FT-CUT and JZ-CUT). These variants are also summarized in table 6.1.

We run our experiments using a version of the Java-based network simulator developed by Jain [64], modified with MAC-level support for our protocols. We consider a range of network topologies and sizes, and various traffic demand patterns. We assume a link capacity of 10 Mbps. The link propagation delay is not considered explicitly, but is accounted for in the slot switch time (guard time) t_{switch}, which we conservatively set to $1ms$ in the default case. Unless otherwise specified, we use a slot size (or maximum slot size for adaptive algorithms) of $20ms$. We assume no packet or token losses, and no node or link failures. For the optimal LP formulations we use the experimental setup described in Section 5.2.

Topologies: We consider several types of topologies: a) random topologies, with varying degrees of connectivity and of varying sizes; b) an actual real-world topology, derived from the one used in the Aravind Eye Hospital in India [112]; c) typical mesh WiFi topologies, using the construction method introduced by Raman [91], which we denote as the *Raman* topology henceforth.

Traffic: We assume traffic consisting of many CBR flows, 500 Kbps each. We choose CBR flows because this traffic is representative of applications supported today in rural wireless deployments: VoIP, telemedicine and streaming of educational content.

We consider the following patterns of traffic demand, ranging from very *asymmetric* to very *symmetric*: a) one source to many randomly distributed destinations (*single-source, many sinks*); b) unidirectional CBR flows, with randomly chosen source-destination pairs (*unidirectional*); c) pairs of CBR flows in opposite directions (*bidirectional*), with many random source-destination pairs. We leave the evaluation of other types of traffic (*e.g.,*TCP) for future work.

Performance metrics: We measure performance in terms of maximum network throughput and average delay.

Method: We generate several networks (typically 5) for each topology type and network size. Next, we generate several ordered sets (typically 5) of CBR flows, using the traffic patterns described above. We then run a simulation using each of these flow sets. During a simulation run, we start with an unused network, and incrementally add CBR flows from the ordered flow set, until the network reaches saturation.

6.3.2 Performance in Random Topologies

In order to understand the behaviour of various algorithms with increasing network load, we start our evaluation by looking at one graph, a 30-node random graph with an average degree of 3, and we use randomly generated *unidirectional* CBR traffic.

We perform the experiment by adding one CBR flow at a time; we then measure how many of these flows can be accommodated by the network. We consider a flow to

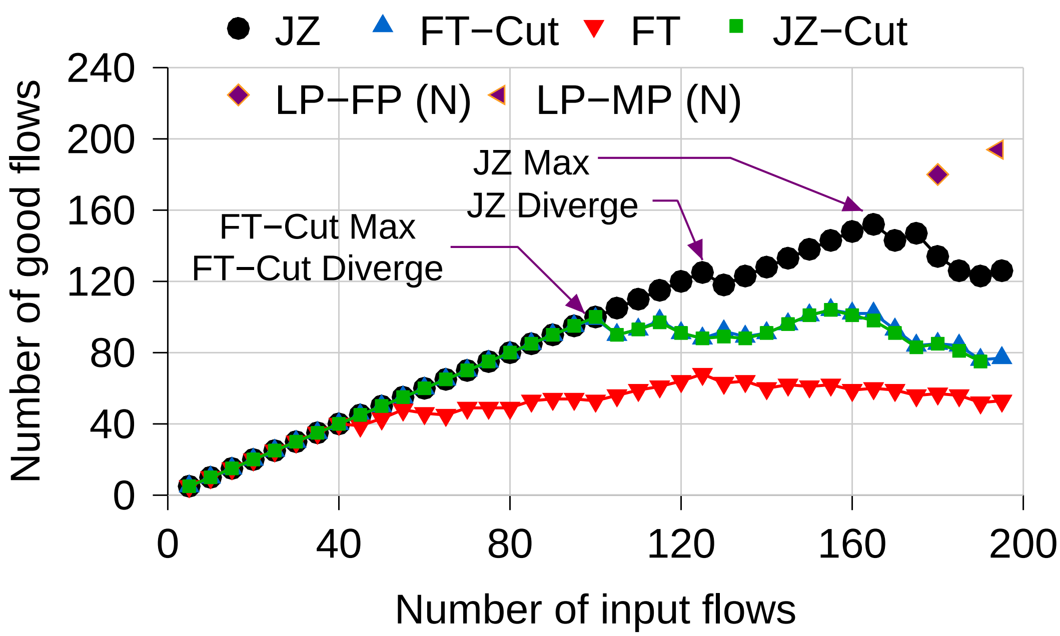


Figure 6.7: Number of successful flows as we add *unidirectional* CBR flows on a 30-node random graph (average degree of 3) for various MAC protocols.

be accommodated successfully (*i.e.,* a *"good"* flow) if it receives 90% of its packets and its per-packet delay is not continuously increasing.

As we can see from our results, plotted in Figure 6.7, initially all flows are accommodated, but as more flows are added, some links become saturated, and the corresponding flows on these saturated links suffer. For each experiment, we emphasize two metrics: a) *max point*, which is the maximum number of good flows supported at any time during the experiment (we use this as a proxy for maximum throughput), and b) *divergence point*, which is the maximum number of good flows that could be successively added from the beginning without having to drop any flow. We highlight this second metric because in practice, flows arrive in random order and, once accepted, they cannot be dropped. These

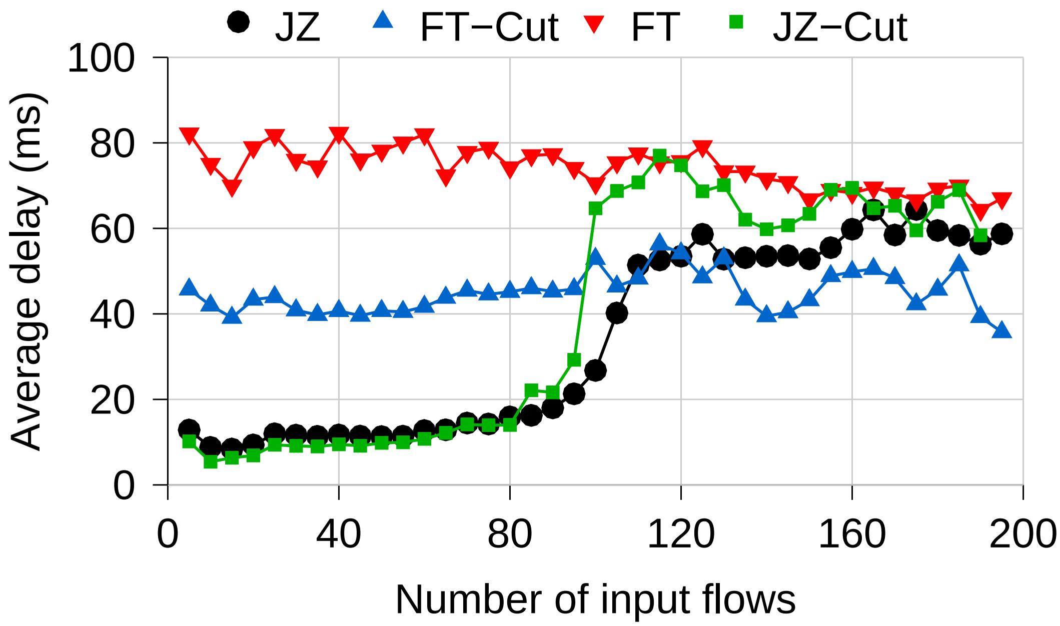


Figure 6.8: Average delay of successful flows as we add *unidirectional* CBR flows on a 30-node random graph (average degree of 3) for various MAC protocols.

two metrics are illustrated in Figure 6.7. For this particular network, JazzyMAC accommodates 46% more flows than FT-CUT and 137% more than FT. JazzyMAC is also 32% better at the *divergence point* than FT-CUT and over 200% better than FT.

Figure 6.8 shows what happens to the average per-flow delay as we increase the volume of traffic in the network. As expected, JazzyMAC operates at lower slot sizes, and therefore sees consistently smaller delays than FT and FT-CUT, which use a fixed slot size of 20ms. At low utilizations, these delay improvements are between 2–4x over FT-CUT and between 4–8x over FT. Among the fixed slot approaches, we see that FT-CUT does much better than FT, even though it uses fewer links, which is a common trend throughout our experiments.

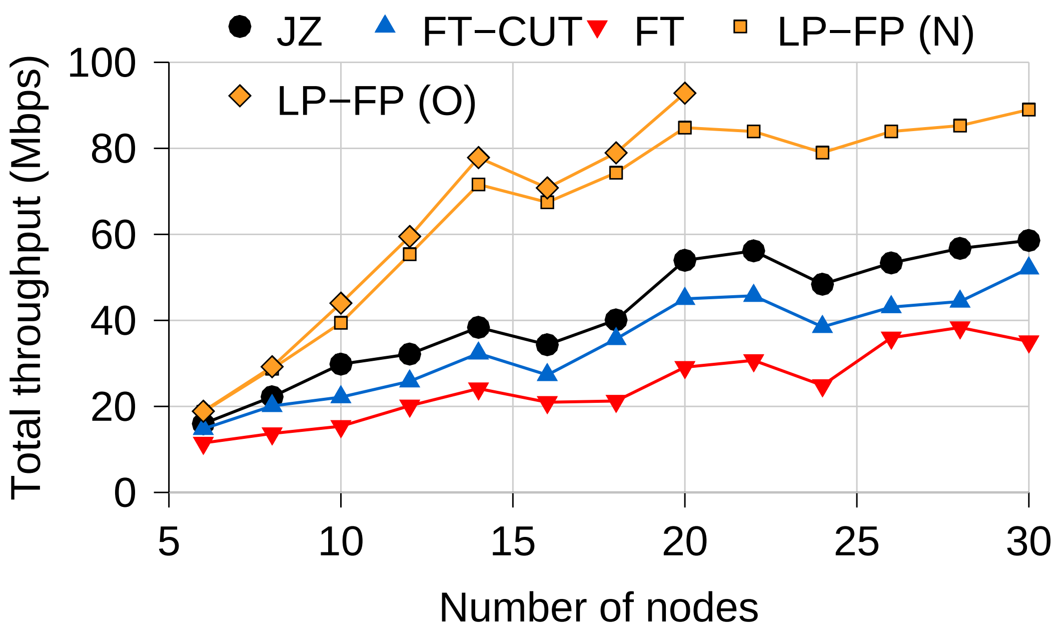


Figure 6.9: Maximum throughput for *unidirectional* CBR flows with increasing network size. LP-FP(O) is the fixed path optimum (not tractable beyond size 20), and LP-FP(N) is the more tractable approximation (described in the previous Chapter). For random topologies (average degree of 3)

Next we investigate whether JazzyMAC's improvements are consistent over a range of network sizes. We start by revisiting the experiment discussed in Section 6.1, where we measure the gap in maximum throughput between practical and optimal algorithms. For this experiment we span several network sizes, and compute average results over 5 random topologies for each network size, and 5 sets of unidirectional CBR traffic demands for each topology. Our results, presented in Figure 6.9 confirm that JazzyMAC outperforms fixed slot approaches in all cases, and that it reduces the gap to optimal throughput.

Nonetheless, this gap remains large. Upon closer inspection, we find this to be the case because the solution of the LP chooses the best set of flows (that maximizes total throughput) from the large pool of input flows. In contrast, JazzyMAC and the other practical algorithms are constrained to accept flows in the order they arrive (which is non-optimal), making our comparison unfair. To compensate for this effect, we compute the equivalent of the *divergence point* for the LP solution, using the following iterative approach: we start with a small set of flows, and incrementally add flows to it. At each step, we run the LP solver to see if the current flow set is *feasible*, meaning that the LP solver can find a link transmission schedule that accommodates all of the flows in the set. If this is the case, we add more flows, otherwise we stop, and use the network throughput achieved using the largest *feasible* flow set as our divergence point. We then compare the *divergence throughput* of the optimal approaches to the *divergence throughput* of JazzyMAC and other practical algorithms, and plot the results in Figure 6.10. We can see that, after eliminating the unfairness in our comparison, JazzyMAC effectively halves the gap to optimal throughput.

6.3.3 Effect of Traffic and Topology

We now examine the performance of JazzyMAC under various traffic patterns and various topology types.

One source, many sinks: In our first experiment we use traffic from a single source to many sinks, which is representative of the case when a when a video server streams content to many destinations.

Figure 6.11 plots the (a) *maximum bandwidth* and (b) the number of flow additions until the first network flow fails (*i.e.,* the *divergence point*), achieved by our algorithms in the three types of topologies described previously (random, Aravind, and Raman). The

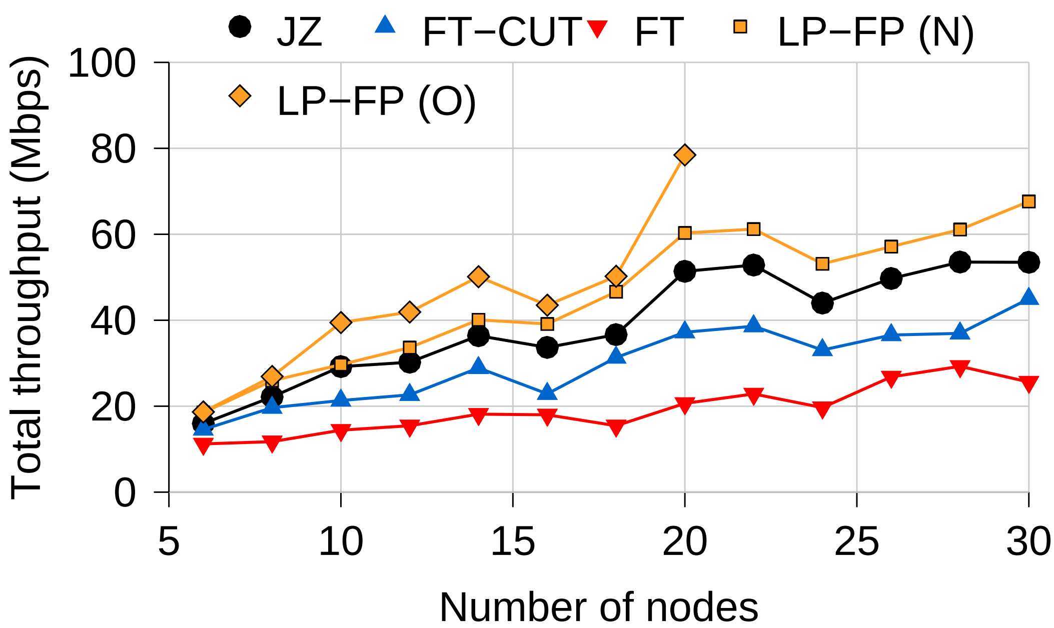
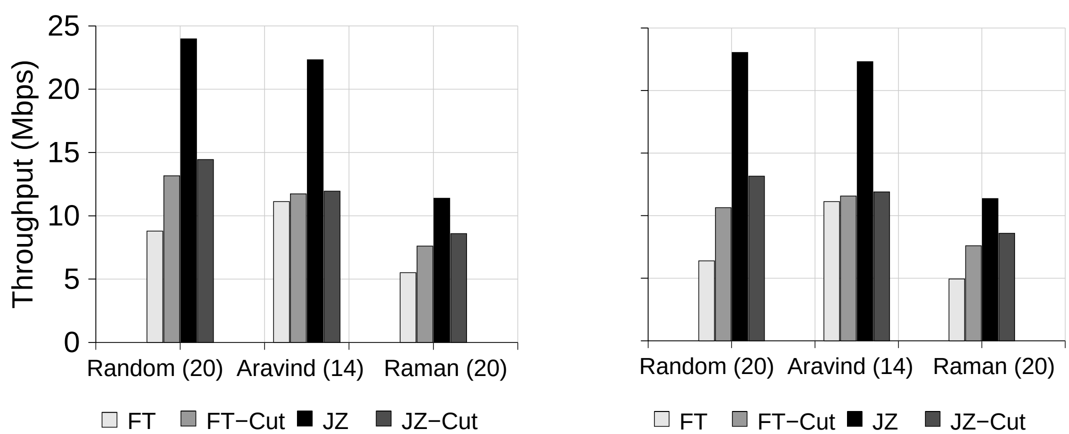


Figure 6.10: Divergence throughput or *unidirectional* CBR flows with increasing network size. For random topologies (average degree of 3)



(a) Max Bandwidth (b) Diverge point

Figure 6.11: Throughput for various topologies, random CBR flows from one source to all the nodes.

random and Raman topologies have 20 nodes, while the Aravind topology has 14 nodes. For the random and Raman cases, we generate 5 topologies and for each topology we run

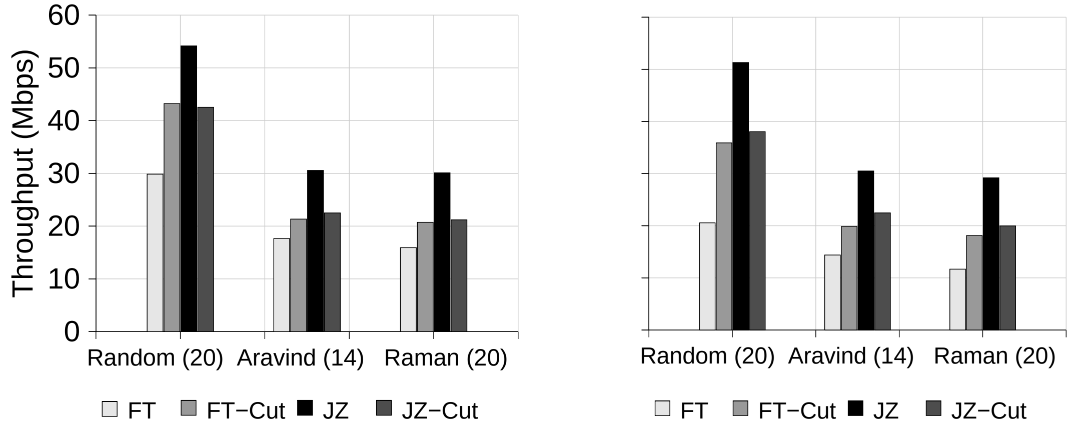
5 sets of CBR flows and average over all of them.

We find that for this asymmetric traffic distribution, JazzyMAC achieves dramatically higher throughput across all topologies, with improvements as large as 100% over FT-CUT, and even larger over FT. We also note that, while FT performs better over the maxcut (*i.e.,* FT-CUT > FT), JazzyMAC performs much better over the original graph: it is able to make productive use of the extra links.

Many sources and sinks: We perform the same comparisons for the other two traffic patterns as well: random unidirectional (Figure 6.12) and bidirectional (Figure 6.13) flows. In the first case, the improvements over FT-CUT are between 25–50% for the *max point*, and 40–60% for the *divergence point*. In the second case, the throughput improvements are much lower, 15–45% for the *max point* and 20–50% for the *divergence point*.

Overall, we see that JazzyMAC consistently outperforms the other protocols across all the topologies and traffic types. We also find that the relative throughput improvements given by JazzyMAC are larger for more asymmetric traffic. This is to be expected, given that variable slot sizes are most useful in asymmetric traffic conditions, where traffic demands are very different in different directions on the same link, but also across different links. For symmetric traffic, which naturally requires similar slot sizes, JazzyMAC's throughput improvements are more modest.

Another important observation can be derived from the relative ordering (in terms of achieved throughput) of the four measured protocols: JZ > JZ-CUT $\approx$ FT-CUT > FT, which holds true across all our topologies and traffic patterns. On one hand, this finding confirms that, for fixed slot approaches, it is more opportune to operate on the maximum



(a) Max Bandwidth

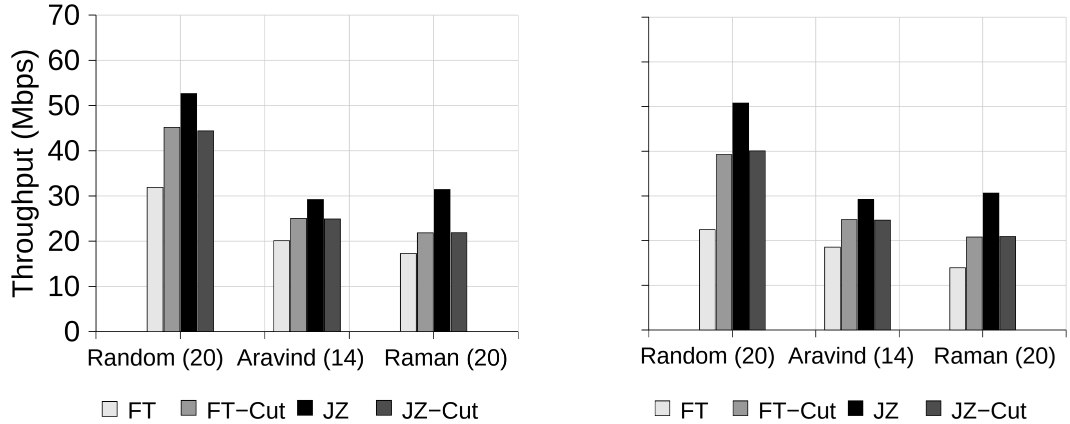
(b) Diverge point

Figure 6.12: Throughput for various topologies with *unidirectional* random CBR flows.

bipartite subgraph of a given network topology (as done by 2P and WiLDNet) rather than on the original topology (since FT-CUT > FT). On the other hand, if dynamic slots are used, it becomes more profitable to operate on the original non-bipartite topology (JZ > JZ-CUT). This confirms the importance of having an approach that takes advantage of all the network links, increasing network capacity (reflected in the increased throughput achieved by JZ), but also improving fault-tolerance.

6.3.4 Bandwidth vs. Delay Tradeoff

We also look at how JazzyMAC enables a better combination of *average delay* and *maximum throughput* than existing fixed-slot approaches. We perform our experiments using a random topology of 20 nodes with an average connectivity degree of 3, and using random unidirectional CBR traffic. To change the tradeoff between throughput and delay, we vary the *fixed* slot sizes of FT-CUT and FT uniformly, between $3ms$ and $12ms$, in 5 steps. For JazzyMAC, we vary the value of the *maximum* slot size in the same range. For



(a) Max Bandwidth (b) Diverge point

Figure 6.13: Throughput for various topologies with *bidirectional* random CBR flows.

all of the algorithms, we assume a slot switching overhead of $1ms$.

We then measure the *maximum network throughput*, and the *average end-to-end delay* experienced at half of the saturation throughput of the network. We plot the tradeoff between maximum throughput and average delay in Figure 6.14.

We fist note that setting JazzyMAC's upper bound on the slot size to the largest value ($12ms$) is clearly the best setting in terms of throughput, and as good as any other setting in terms of delay. This result confirms that JazzyMAC's adaptation mechanism is effective in dynamically adapting slot size to achieve both high throughput when needed and low delay at average utilizations.

We also find that, as suggested by previous experiments, JazzyMAC outperforms FT-CUT and FT by a large margin, in terms of both throughput and delay. Among the fixed-slot approaches FT-CUT performs best, and increasing its fixed slot size beyond $6ms$ has diminishing bandwidth benefits.

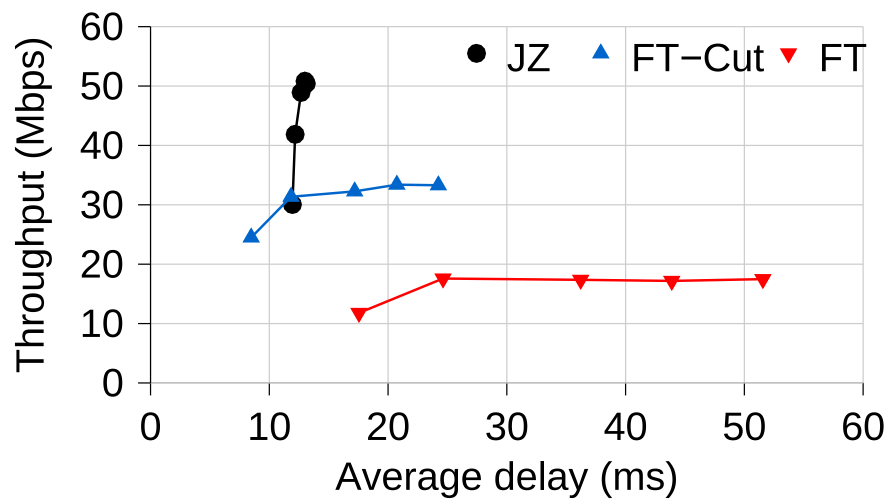


Figure 6.14: Tradeoff between maximum throughput and average delay. This is for 20-node random network (average degree of 3). Slot size increase from $3ms$ to $12ms$.

6.4 Related work

Since CSMA/CA is ill-suited for multi-hop mesh network settings, in both short- and long-range links [53, 66, 107], several TDMA-based protocols have been proposed for mesh networks. Djukic and Valaee [48] propose min-max heuristics to provide offline algorithms to minimize delay when link bandwidths *are known in advance*. Wang et al. [121] provide centralized and distributed algorithms to maximize throughput by taking into account interfering links. In contrast, our new protocol needs no future knowledge of traffic, is fully distributed, provides flexible delay-bandwidth guarantees, and can dynamically adjust to varying traffic demands.

6.5 Conclusion

In this chapter, we identify opportunities to increase the performance of WiLD networks, by using dynamic slot sizing that reflects instantaneous traffic demand.

We then use this insight in designing JazzyMAC, an efficient and practical MAC protocol. Our evaluations over a wide range of topologies and traffic demand distributions show that JazzyMAC is successful in exploiting asymmetric traffic, time varying traffic, and non-bipartite topologies to achieve higher throughput and better control over latency than previous TDMA-based approaches. In our experiments JazzyMAC consistently outperforms the existing protocols with respect to both bandwidth and latency, and closes the gap between the network bandwidth achieved by practical algorithms and the maximum achievable bandwidth computed with ideal, offline approaches.

Chapter 7

Conclusions

Overall, the work presented in this dissertation meets our objectives: we build wireless networking technology that can be used to provide inexpensive, high performance networking connectivity to remote areas and places with few potential users. We began by understanding the constraints imposed by the wireless long-distance, multi-hop mode of operation. We continued by designing and implementing the MAC and link-layer mechanisms required to provide high performance operation at large scale, and we evaluated it by using it in real-world network deployments, used to provide telemedicine and other services to thousands of users. Incorporating experiences from our deployments, we then revisited our MAC design, and proposed a solution that maximizes total network throughput while maintain small per-packet delay. JazzyMAC is still under implementation, and remains to be tested on the field, but our preliminary results indicate this to be a strict and often substantial improvement over WiLDNet, allowing high-bandwidth, low-delay connectivity for delay-sensitive, bandwidth intensive applications such as telemedicine or remote education.

7.1 Contributions

In the following, we summarize our specific contributions in each of the major chapters of this work:

7.1.1 Performance Characterization of WiLD links

We performed a detailed measurement study, and identify all the important causes of loss and inefficiency in WiLD networks. This enabled us to design the appropriate MAC an link-layer solutions to deal with loss and increase throughput.

- We found that long-distance Wi-Fi links feature high and variable loss rates (sometimes as high as 80%). These losses are much higher in urban links, and can be highly asymmetric in different directions of a link. The loss has two components: a *residual* component (usually a few percent), and a large *bursty* component.

- We found the standard 802.11 MAC to be inappropriate for individual long links. The causes for this are a) the stop-and wait flow-control, which is inefficient at large propagation delays, and b) the breakdown of the carrier sensing mechanisms, which is again affected by the long propagation. We illustrated the inefficiency of the stop-and-wait approach by showing that unidirectional throughput decreases gradually with distance (half at 50km). We showed the effect of collisions by noting the large increase in packet loss with link distance, when sending traffic in both directions.

- We discovered that the standard 802.11 MAC is also inappropriate for multiple adjacent directional links using the same wireless channels, given the inadequacy of carrier sensing in directional (non-broadcast) environments, leading to cross-link interference. This interference leads to significant packet loss (about 30% when using half of the channel capacity, and above 50% when sending at capacity), and makes multi-hop networks very inefficient.

- Among channel-induced losses, we found external Wi-Fi interference to be the most significant, given the hidden terminal effects caused by the directionality of transmissions, and accentuated by large propagation delay. This results are very different from results in urban meshes (which do not feature directional links) – in these networks, most of the loss was due to multipath.

- We investigated appropriate loss avoidance and recovery mechanisms. We concluded that a TDMA MAC design avoids protocol induced losses. We also found that, since interference is the largest cause of losses, operating at high datarates is better that operating at low datarates, because packets are shorter, and the collision probability is minimized. Finally, we saw that a large number of lost packets are truncated, making byte-level FEC very inefficient, and recommending packet-level FEC.

7.1.2 WiLDNet

We designed, implemented, deployed and evaluated WiLDNet, a system that incorporates a new MAC layer protocol and link recovery mechanisms. We then demonstrated how WiLDNet successfully avoids protocol-induced losses, recovers from other losses, and increases throughput efficiency, demonstrating close to capacity TCP throughput over multi-hop paths.

- We built our new MAC using off-the-shelf Atheros radios running standard firmware. In order to achieve this, we modified the MadWiFi driver to first disable carrier sensing, per packet ACKs and the stop-and-wait flow-control, and then implemented TDMA access and sliding-window flow-control in the Click modular router.

- The sliding-window flow-control improved the throughput efficiency of our protocols, allowing throughput to remain at the same levels ($\approx$6Mbps bidirectional) irrespective of link distance.

- WiLDNet has an implicit synchronization mechanism for TDMA, which is more resilient in the face of packet losses than existing implementation. We showed experimentally that our MAC successfully avoids all protocol-induced losses, and is resilient to other sources of loss.

- We designed and built link-level recovery mechanisms that combine ARQ an packet-level FEC. We then showed that our solution enables achieving arbitrarily low loss rates, while allowing the flexible negotiation of the bandwidth/delay tradeoff for every link, as desired by various applications.

- We deployed WiLDNet and demonstrated experimentally that, both in single link deployment the throughput is not affected by distance, only by interference experienced locally at endpoints. In multi-hop paths we also showed that the performance obtained using the same channel on all links is the same as the performance obtained if using different channels, confirming that we eliminate inter-link interference, and validating the spectral-efficiency of our solution.

7.1.3 Capacity of WiLD Networks

Using linear programming optimization, we evaluated the optimal WiLD network throughput in different topologies.

- We adapted linear programming methods [63] for computing the maximum throughput of multihop wireless networks to the constraints specific to WiLD networks, by considering the interference patterns featured in these networks. We also considered additional constrains derived from practical solutions for routing and transmission scheduling, and evaluated network capacity for several network topologies.
- We showed that *fixed routing* and *constraining nodes to simultaneously send on all of their links* are acceptable constraints, yielding network capacities close to optimal.
- We showed that STMDA-based algorithms using fixed transmission slots on the entire network yield better throughput when utilized over the maximum bipartite subgraph of the network topology then when used over all of the network links.
- Our results indicated a very large gap between optimal solutions that take into account traffic information, and algorithms using fixed transmission slots that do not adapt link scheduling to traffic demand.

7.1.4 JazzyMAC

Inspired by our real-life deployment experience, where we found that packet delay can become an issue in long end-to-end paths, and by our study of network capacity, we revisited the problem of efficient MAC design, and proposed JazzyMAC, a MAC protocol that uses local traffic information to maximize link utilization and maintain low delay.

- We designed a MAC protocol that maintains the same type of interference avoidance constraints introduced by SynOp and used in WiLDNet and 2P, but uses local traffic information to adapt the size of link transmission slots, such that only links that require longer transmission opportunities get this opportunities, while links featuring low utilizations use short transmission slots. We proved that our protocol avoids inter-link interference, maintains a deadlock and starvation free operation, and provides some link utilization and link delay guarantees.

- JazzyMAC improves the maximum network throughput achieved by WiLDNet by 15 to 100% accross various topologies , with larger improvements for asymmetric traffic, when differentiated link transmission slots provide a larger gain. JazzyMAC consistently outperforms WiLDNet across a variety of topologies and traffic patterns.

- JazzyMAC also improves the average delay experienced in the network, by dynamically negotiating the bandwidth/delay tradeoff, and achieving both better saturation throughput and smaller average delay.

- This new MAC removes the constraint of operating in bipartite topologies, being deployable in any topology, and making use of the capacity of all of the network's links.

7.2 Limitations

Our solutions feature several limitations. Some of these are specific to the assumed network architecture, while some are specific to the design of our protocols. We begin by discussing the former.

Line-of-sight requirement: An important limitation of our solution is the requirement for line-of-sight between link endpoints. This makes network topology highly dependent on geography and elevation profile, vegetation on the ground, and many other factors. Communication relays must often be placed in locations with high elevation, and tall communication towers are often needed. When naturally high locations are not available (*e.g.,* across plains, or when connecting remote locations across bodies of water), long links become very hard to deploy, with very tall towers needed to compensate for Earth's curvature. For example, 40m towers are required to clear the horizon for a 45km link. All these considerations make network design challenging, with each link requiring careful planning. The use of directional antennas makes antenna alignment difficult, especially if place on top of tall communication towers, and in situations when appropriately skilled staff are not available.

No solution for point-to-multipoint access: Our solution is suited well for backbone links connecting cities and villages, and for access links serving remote areas with sparse users. But once connectivity is brought to a particular village or city, an effective way to distribute this connectivity to many tens of users situated at *medium range* (2-10km) from the point of connectivity is needed in many cases. Standard Wi-Fi access points can provide this, but only at short ranges (a few hundred meters). Deploying individual point-to-point links to each individual user can be too costly and inefficient. We therefore need a complementary solution for cost-efficient medium range point-to-multipoint connectivity.

Besides limitations pertaining to the network architecture, there are other limitations stemming from our protocol design choices:

No simultaneous transmissions on interfering links: In order to fully avoid interference, our solutions impose the constraint of non-simultaneous transmission on potentially-interfering links. But in practice interference happens to various degrees, from mild to severe, which depends on the amount of data to be transferred, modulation and other factors. Therefore, in some situations, network capacity could be increased by allowing simultaneous transmission on interfering links, and then recovering from packet losses as needed.

Centralized bootstrapping mechanism for JazzyMAC: Although JazzyMAC is a distributed protocol that only requires local information for long-term operation, the bootstrapping of the protocol is currently centralized and requires knowledge of the entire network topology. This can be acceptable for small deployments, but not for large-scale networks. We believe that distributed bootstrapping is possible, since it relies on vertex coloring, which can be distributed.

7.3 Future work

In the following we take a look at some of the research issues we plan to address in the future.

7.3.1 Short term

We intend to tackle several remaining issues with the design and implementation of JazzyMAC and WiLDNet:

- *Implement and evaluate JazzyMAC in real deployments.* We plan to deploy JazzyMAC in two stages. The first step would be to build a prototype MAC implemented using the Click modular router. This would have all the MAC features, allowing us to evaluate the protocol under various types of traffic, and using different topologies. Our second step would be to build the protocol at driver level, which would allow us to have more accurate timing, and to support very small transmission slots.

- *Design a distributed bootstrapping mechanism.* As previously mentioned, one of the limitations in our design is the centralized bootstrapping strategy. We therefore plan to design and implement a distributed bootstrapping algorithm, relying on existing distributed coloring algorithms [69].

- *Evaluate and deploy 802.11a and 802.11n WiLD networks.* Our current experiments and deployments have exclusively used the 802.11b PHY, which has good RF propagation at 2.4GHz, and features a robust modulation. However, our solution can work well with other PHY layers, without requiring modifications. We therefore plan to investigate the performance achievable with 802.11a/g and 802.11n PHYs. Recent experiments, and data collected from the AirJaldi [1] network, advocate the use of 802.11a in long-distance deployments, given the smaller width of the Fresnel cone produced at higher frequencies, making it easier to achieve line of sight. 802.11a would also allow for higher datarates, essential in backbone links. 802.11n is also very promising, given even higher datarates, and the possibility to use MIMO for beamforming, increasing signal strength in a particular direction.

7.3.2 Long term

Our long-term research plans include:

- *The design and implementation of a point-to-multipoint MAC solution.* As mentioned previously, a complementary point-to-multipoint solution for access would be very important. We are investigating two versions of this solution, one maintaining the compliance with the 802.11 MAC and allowing standard clients to connect to the network, and another one based on a TDMA MAC featuring similarities to OML [98]. This will be done in collaboration with Rabin Patra and other TIER members. The design and implementation of the point-to-multipoint solution is Rabin Patra's dissertation subject.

- *The use of steerable antennas, power adaptation and adaptive client association to increase capacity and range in point-to-multipoint operation.* Steerable antennas can be used by point-to-multipoint APs, or by nodes in a wireless mesh, to direct communication to one client at a time, eliminating broadcast interference and increasing capacity and range, as illustrated by a large body of research [44, 94]. We have already started looking at this problem [86], and we intend to use the low-cost steerable antenna developed by TIER member Omar Bakr to implement and evaluate our solutions.

7.4 Real-world impact

Two exciting aspects about the work presented in this dissertation are a) its adoption in several real-world networks in developing regions, and b) the fact that some of these deployments demonstrated **long-term financial and operational sustainability** (as shown in our recent studies [111, 112]), confirming the appropriateness of our solutions for affordable network connectivity in rural areas of developing countries.

With the effort of several TIER researchers, including Sonesh Surana, Rabin Patra, RJ Honicky, Manuel Ramos, Yahel Ben-David, Melissa Ho, Matt Podolsky, Michael Rosenblum and others, WiLDNet has been deployed in several networks in India, Ghana, Guinea Bissau and the Philippines [114]. Some of these deployments have been carried out by TIER members and some have been performed by other groups using our equipment.

The most notable of these deployments is the Aravind telemedicine network in Theni, southern India. In 2007, the network deployed under the coordination of TIER member Sonesh Surana consisted of five vision centers, connected to the main hospital in Theni (Figure 7.1). The network had a total of 11 wireless routers (6 endpoints, 5 relay nodes) and used 9 point-to-point links. The links ranged from just 1 km to 15 km. Six of the wireless nodes were installed on towers, heights of which range from 24-42 m; the others used short poles on rooftops or existing tall structures, such as the chimney of a power plant on the premises of a textile factory.

This telemedicine network provided video-conferencing for 3000 rural patients per month, and approximately 500,000 patient examinations per year. Given this success, the Theni network expanded to 25 rural vision centers, expanding this model to their

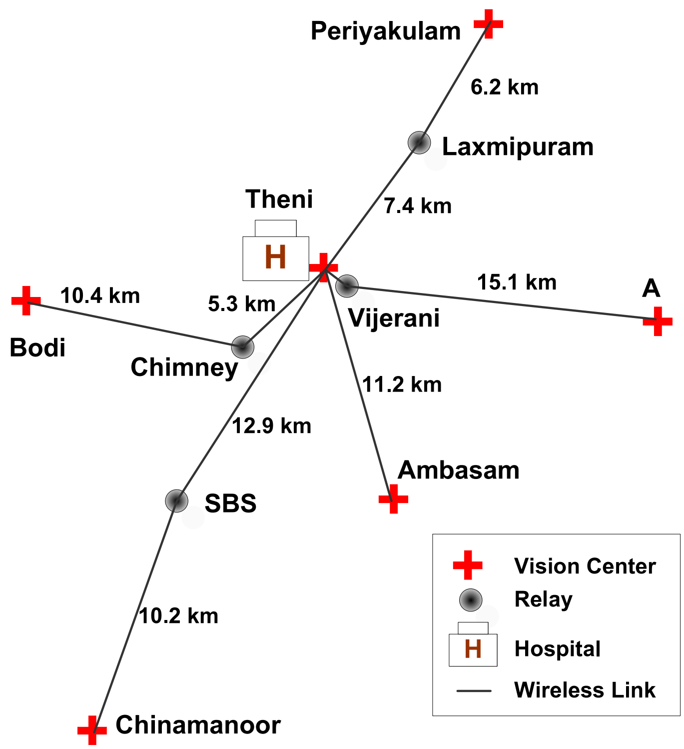


Figure 7.1: Aravind Telemedicine Network. Theni hospital is connected to 5 vision centers. The other nodes are all relays.

hospitals in Madurai and Tirunelveli where they have added more vision centers. The network is currently financially viable, and self-sustaining operationally, without requiring further technical assistance from the TIER group. The achieved financial and operational sustainability [111] prompted the Aravind hospital to plan a further expansion to 50 clinics around 5 hospitals, aiming to carry out 500,000 annual eye examinations.

Another notable deployment is the one in Ghana, coordinated by RJ Honicky. Currently, this educational network connects the University of Ghana Legon campus to their remote campuses at the Korle-Bu Medical School and the City campus. This network is under construction, and will feature links as long as 90km, some using very large custom-made directional antennas manufactured locally.

Our technology has also been used to establish a wireless world record. The Isolate [6] team, lead by Ermanno Pietrosemoli and assisted by TIER researchers Rabin Patra and Sonesh Surana used WiLDNet to establish world records for longest unamplified Wi-Fi links. Two such links, one 279km long, and one 382km long were demonstrated [50]. The links used high-powered 802.11b radios, and achieved bidirectional throughputs close to 6 Mbps.

Bibliography

[1] AirJaldi, Empowering Communities Through Wireless Network. http://drupal.airjaldi.com.

[2] Atheros Wireless Solutions. http://www.atheros.com.

[3] CDMA Development Group. http://www.cdg.org.

[4] CPLEX: LP Solver. http://www.ilog.com.

[5] Digital Gangetic Plains. http://www.iitk.ac.in/mladgp/.

[6] Fundacion Escuela Latinoamericana de Redes. http://www.eslared.org.ve/.

[7] High Performance Wireless Research and Education Network. http://hpwren.ucsd.edu/.

[8] HyperLink HG2424G 2.4GHz 24dBi High Performance Die Cast Reflector Grid Wireless LAN Antenna. http://www.hyperlinktech.com/multimedia/datasheets/DS_HG2424G-NF.PDF.

[9] International Telecommunications Union - Statistics. http://www.itu.int/ITU-D/ict/statistics/.

[10] Internet World Stats. `http://www.internetworldstats.com/stats.htm/`.

[11] Spirent Communications. `http://www.spirentcom.com`.

[12] Super Cantenna. 12dBi Wireless Booster Antenna. `http://www.cantenna.com`.

[13] The Bay Area Research Wireless Network. `http://barwn.org`.

[14] The One Laptop Per Child Project. `http://laptop.org`.

[15] Ubiquity Xtreme Range 2 WiFi MiniPCI Radio. `http://www.ubnt.com/products/xr2.php`.

[16] United States Agency for International Development. `http://usaid.gov`.

[17] Wireless Networking in the Developing World. A practical guide to planning and building low-cos telecommunication infrastructure. Second Edition. `http://wndw.net`.

[18] Zapp Telemobil. `http://www.zapp.ro`.

[19] A.Balaji, T. A. Gonsalves, and R. Kumar. The corDECT Wireless Local Loop: Capacity Predictions. *Conference on Personal Wireless Communication, 1999.*

[20] Daniel Aguayo, John Bicket, Sanjit Biswas, Glenn Judd, and Robert Morris. Link-level Measurements from an 802.11b Mesh Network. In *ACM SIGCOMM*, August 2004.

[21] The Akshaya E-Literacy Project. `http://www.akshaya.net`.

[22] Andreas Almquist Anders Lindgern and Olov Schelen. Quality of Service Schemes for IEEE 802.11 Wireless LANs: An Evaluation. *Module Networks and Application*, 2003.

[23] Aravind Eye Hospital. *Annual Activities Report, 2004*. Aravind Eye Hospitals, 2004.

[24] Atheros. MadWiFi driver for Atheros Chipsets. `http://sourceforge.net/projects/madwifi/`.

[25] A.B. Atkinson. *New Sources of Development Finance*. Oxford University Press, 2005.

[26] Levent Aydin, Eduardo Esteves, and Roberto Padovani. Reverse Link Capacity and Coverage Improvement for CDMA Cellular Systems Using Polarization and Spatial Diversity. In *IEEE International Conference on Communications (ICC)*, volume 3, pages 1887–1892, 2002.

[27] Azim Premji Foundation. Impact of Computer Aided Learning on Learning Achievements: A Study in Karnataka and Andhra Pradesh, May 2004. `http://www.azimpremjifoundation.org/downloads/ImpactofCALonLearningAchievements.pdf`.

[28] S. Baatz, M. Frank, C. Kuhl, P. Martini, and C. Scholz. Bluetooth Scatternets: An Enhanced Adaptive Scheduling Scheme. *IEEE INFOCOM*, 2002.

[29] Ajay Bakre and B. R. Badrinath. I-TCP: Indirect TCP for Mobile Hosts. *ICDCS*, 1995.

[30] Hari Balakrishan. *Challenges to Reliable Data Transport over Heterogeneous Wireless Networks*. PhD thesis, University of California at Berkeley, August 1998.

[31] Hari Balakrishnan, Srinivasan Seshan, Elan Amir, and Randy Katz. Improving TCP/IP Performance over Wireless Networks. In *ACM MOBICOM*, November 1995.

[32] Pravin Bhagwat, Bhaskaran Raman, and Dheeraj Sanghi. Turning 802.11 Inside-out. *ACM SIGCOMM CCR*, 2004.

[33] G. Bianchi. Performance Analysis of the IEEE 802.11 Distributed Coordination Function. *IEEE Journal on Selected Areas in Communications*, 18(3):535–547, 2000.

[34] J. Bicket, D. Aguayo, S. Biswas, and R. Morris. Architecture and Evaluation of an Unplanned 802.11b Mesh Network. *ACM MOBICOM*, 2005.

[35] John Bicket, Daniel Aguayo, Sanjit Biswas, and Robert Morris. Architecture and Evaluation of an Unplanned 802.11b Mesh Network. In *ACM MOBICOM*, August 2005.

[36] Sanjit Biswas and Robert Morris. Opportunistic Routing in Multi-Hop Wireless Networks. *Hotnets-II*, November 2003.

[37] Eric Brewer, Michael Demmer, Bowei Du, Melissa Ho, Matthew Kam, Sergiu Nedevschi, Joyojeet Pal, Rabin Patra, Sonesh Surana, and Kevin Fall. The Case for Technology in Developing Regions. *Computer*, 38(6):25–38, 2005.

[38] CDMA Development Group (CDG). Opportunity for All: Using Wireless to Provide Universal Access to Telecom Services. *White Paper*, page 5, October 2004.

[39] Chandler Burr. Grameen Village Phone: Its Current Status and Future Prospects. In *Business Services for Small Enterprise in Asia: Developing Markets and Measuring Performance*. International Labour Organization, April 2000.

[40] Kameswari Chebrolu, Bhaskaran Raman, and Sayandeep Sen. Long-Distance 802.11b Links: Performance Measurements and Experience. In *ACM MOBICOM*, 2006.

[41] C.M. Cheng, P.H. Hsiao, HT Kung, and D. Vlah. Adjacent Channel Interference in Dual-radio 802.11 a Nodes and Its Impact on Multi-hop Networking. *IEEE Globecom*, 2006.

[42] I. Chlamtac and S. Kutten. A Spatial Reuse TDMA/FDMA for Mobile Multi-hop Radio Networks. *IEEE INFOCOM*, 1985.

[43] I. Chlamtac and S. Lerner. A Link Allocation Protocol for Mobile Multihop Radio Networks. *IEEE Globecom*, 1985.

[44] Romit Roy Choudhury, Member-Xue Yang, Senior Member-Ram Ramanathan, and Senior Member-Nitin H. Vaidya. On Designing MAC Protocols for Wireless Networks Using Directional Antennas. *IEEE Transactions on Mobile Computing*, 5(5):477–491, 2006.

[45] I. Cidon and M. Sidi. Distributed Assignment Algorithms for Multi-hop Packet-radio Networks. In *IEEE Transactions on Computers, vol. 38, no. 10, pp. 1353-1361*, October 1989.

[46] Martin V. Clark, K.K. Leung, B. McNair, and Zoran Kostic. Outdoor IEEE 802.11 Cellular Networks: Radio Link Performance. *IEEE ICC*, 2002.

[47] Comsys. The Comsys VSAT Report, 10th edition. 2007.

[48] Petar Djukic and Shahrokh Valaee. Link Scheduling for Minimum Delay in Spatial Re-use TDMA. In *IEEE INFOCOM*, May 2007.

[49] Christian Doerr, Michael Neufeld, Jeff Filfield, Troy Weingart, Douglas C. Sicker, and Dirk Grunwald. MultiMAC - An Adaptive MAC Framework for Dynamic Radio Networking. In *IEEE DySPAN*, November 2005.

[50] Ermanno Pietrosemoli. Long Distance WiFi Trial. *International Summit for Community Wireless Networks*, 2007. `http://www.eslared.org.ve/articulos/Long%20Distance%20WiFi%20Trial.pdf`.

[51] ETSI. ETSI Standard for Digital Video Broadcasting (DVB); Framing structure, channel coding and modulation for 11/12 GHz satellite services Digital Video Broadcasting (DVB); Framing structure, channel coding and modulation for 11/12 GHz satellite services. 1997.

[52] Z. Fu, P. Zerfos, H. Luo, S. Lu, L. Zhang, and M. Gerla. The Impact of Multihop Wireless Channel on TCP Throughput and Loss. *IEEE INFOCOM*, 2003.

[53] Violeta Gambiroza, Bahareh Sadeghi, and Edward Knightly. End-to-End Performance and Fairness in Multihop Wireless Backhaul Networks. In *ACM MOBICOM 2004*, September 2004.

[54] Michele Garetto, Theodoros Salonidis, and Edward Knightly. Modeling Per-Flow Throughput and Capturing Starvation in CSMA Multi-hop Wireless Networks. In *IEEE INFOCOM 2006*, April 2006.

[55] Sumathi Gopal and Dipankar Raychaudhuri. Experimental Evaluation of the TCP Simultaneous-Send Problem in 802.11 Wireless Local Area Networks. In *ACM SIGCOMM E-WIND Workshop*, August 2005.

[56] Government of India. Census India, 2001. http://www.censusindia.net.

[57] Grameen Bank. Grameen Telecom Overview. http://www.grameen-info.org/grameen/gtelecom/.

[58] Tristan Henderson, David Kotz, and Ilya Abyzov. The changing usage of a mature campus-wide wireless network. In *ACM MOBICOM*, 2004.

[59] M. Heusse, F. Rousseau, G. Berger-Sabbatel, and A. Duda. Performance anomaly of 802.11 b. *IEEE INFOCOM*, 2003.

[60] H.D. Hristov. *Fresnel zones in wireless links, zone plate lenses, and antennas*. Artech House Boston, 2000.

[61] IEEE 802.11. Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification. 1999.

[62] Intel Corporation. Classmate PC. http://www.intel.com/intel/worldahead/classmatepc.

[63] Kamal Jain, Jitendra Padhye, Venkat Padmanabhan, and Lili Qiu. Impact of Interference on Multi-hop Wireless Network Performance. In *ACM MOBICOM*, September 2003.

[64] Sushant Jain, Kevin Fall, and Rabin Patra. Routing in a Delay Tolerant Network. In *ACM SIGCOMM*, pages 145–158, New York, NY, USA, 2004. ACM.

[65] Kyle Jamieson, Bret Hull, Allen Miu, and Hari Balakrishnan. Understanding the real-world performance of carrier sense. In *ACM SIGCOMM E-WIND Workshop*, 2005.

[66] Kyle Jamieson, Bret Hull, Allen K. Miu, and Hari Balakrishnan. Understanding the Real-World Performance of Carrier Sense. In *ACM SIGCOMM E-WIND Workshop*, August 2005.

[67] Amit P. Jardosh, Krishna N. Ramachandran, Kevin C. Almeroth, and Elizabeth M. Belding-Royer. Understanding link-layer behavior in highly congested ieee 802.11b wireless networks. In *ACM SIGCOMM E-WIND Workshop*, 2005.

[68] A. Jhunjhunwala. Towards Enabling India through Telecom and Internet Connections. *Variously published essay available online at www. tenet. res. in*, 2000.

[69] Ryouhei Kawano and Toshiaki Miyazaki. Distributed Coloring Algorithm for Wireless Sensor Networks and Its Applications. In *CIT '07: Proceedings of the 7th IEEE International Conference on Computer and Information Technology*, pages 997–1002, Washington, DC, USA, 2007. IEEE Computer Society.

[70] Murali Kodialam and Thyagarajan Nandagopal. Characterizing the Achievable Rates in Multihop Networks. In *ACM MOBICOM*, September 2003.

[71] Eddie Kohler, Robert Morris, Benjie Chen, John Jannotti, and M. Frans Kaashoek.

The Click Modular Router. *ACM Transactions on Computer Systems*, 18(3):263–297, August 2000.

[72] R. Kuriyan and K. Toyama et al. Integrating Social Development and Financial Sustainability: The Challenges of Rural Computer Kiosks in Kerala. *Information and Communication Technologies and Development (ICTD)*, 2006.

[73] Lawrence Harte. *Introduction to EVDO: Physical Channels, Logical Channels, Network, and Operation*. Athlos, July 2004. ISBN: 193281308X.

[74] Shu Lin and Daniel Costello. *Error Control Coding: Fundamentals and Applications*. Prentice Hall, 1983.

[75] S. Mangold, S. Choi, P. May, O. Klein, G. Hiertz, and L. Stibor. IEEE 802.11 e Wireless LAN for Quality of Service. *Proc. European Wireless*, 18:32–39, 2002.

[76] David L. Mills. Internet Time Synchronization: The Network Time Protocol. In *Global States and Time in Distributed Systems, IEEE Computer Society Press*. 1994.

[77] A. Mishra, M. Shin, and W. Arbaugh. An empirical analysis of the IEEE 802.11 MAC layer handoff process. *ACM SIGCOMM Computer Communication Review*, 33(2):93–102, 2003.

[78] S.M. Mishra, J. Hwang, D. Filippini, R. Moazzami, L. Subramanian, and T. Du. Economic Analysis of Networking Technologies for Rural Developing Regions. *Internet And Network Economics: First International Workshop, WINE 2005, Hong Kong, China, December 15-17, 2005: Proceedings*, 2005.

[79] Sergiu Nedevschi, Rabin Patra, Sonesh Surana, Sylvia Ratnasamy, Lakshminarayanan Subramanian, and Eric Brewer. An Adaptive, High Performance MAC for Long-Distance Multihop Wireless Networks. *ACM MOBICOM*, 2008.

[80] Michael Neufeld, Jeff Fifield, Christian Doerr, Anmol Sheth, and Dirk Grunwald. SoftMAC - Flexible Wireless Research Platform. In *HotNets-IV*, November 2005.

[81] R. Orgier. A Decomposition Method for Optimal Scheduling. *24th Allerton Conference*, 1986.

[82] Joyojeet Pal, Sergiu Nedevschi, Rabin Patra, and Eric Brewer. A Multi-disciplinary Approach to Studying Village Internet Kiosk Initiatives: The case of Akshaya. *Policy Options and Models for Bridging Digital Divides*, March 2005.

[83] Joyojeet Pal, Udai Singh Pawar, Eric A. Brewer, and Kentaro Toyama. The case for multi-user design for computer aided learning in developing regions. In *WWW '06: Proceedings of the 15th international conference on World Wide Web*, pages 781–789, New York, NY, USA, 2006. ACM.

[84] Rabin Patra, Sergiu Nedevschi, Sonesh Surana, Anmol Sheth, Lakshminarayanan Subramanian, and Eric Brewer. WiLDNet: Design and Implementation of High Performance WiFi Based Long Distance Networks. *USENIX/ACM NSDI*, 2007.

[85] Rabin Patra, Joyojeet Pal, Sergiu Nedevschi, Madelaine Plauche, and Udai Pawar. Usage Models of Classroom Computing in Developing Regions. *Information and Communication Technologies and Development (ICTD)*, 2007.

[86] Rabin Patra, Sonesh Surana, Sergiu Nedevschi, and Eric Brewer. Optimal Scheduling and Power Control for TDMA based Point to Multipoint Wireless Network. *ACM SIGCOMM Workshop on Networked Systems for Developing Regions (NSDR)*, August 2008.

[87] Udai Singh Pawar, Joyojeet Pal, Rahul Gupta, and Kentaro Toyama. Multiple mice for retention tasks in disadvantaged schools. In *CHI '07: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 1581–1590, New York, NY, USA, 2007. ACM.

[88] Perry LaForge. CDMA Enhancements Build on a Strong Foundation. *Business Briefing: Wireless Technologies 2003*, page 2, 2003.

[89] Qualcomm. 3G Today: Third Generation Wireless Technology Newsletter. 2006.

[90] K. Ramamritham and et al. A. Bahuman. Innovative ICT Tools for Information Provision in Agricultural Extension. *Information and Communication Technologies and Development (ICTD)*, 2006.

[91] Bhaskaran Raman. Channel Allocation in 802.11-based Mesh Networks. In *IEEE INFOCOM*, April 2006.

[92] Bhaskaran Raman and Kameswari Chebrolu. Revisiting MAC Design for an 802.11-based Mesh Network. In *HotNets-III*, 2004.

[93] Bhaskaran Raman and Kameswari Chebrolu. Design and Evaluation of a new MAC Protocol for Long-Distance 802.11 Mesh Networks. In *ACM MOBICOM*, August 2005.

[94] Ram Ramanathan. *Antenna Beamforming and Power Control for Ad Hoc Networks*. Wiley-IEEE Press, 2004.

[95] S. Ramanathan and Errol L. Lloyd. Scheduling algorithms for multi-hop radio networks. In *ACM SIGCOMM*, pages 211–222, New York, NY, USA, 1992. ACM Press.

[96] R. Ramaswami and K. K. Parhi. Distributed algorithm for efficient and interference-free broadcasting in radio networks. *IEEE INFOCOM*, 1988.

[97] Ashish Raniwala and Tzi cker Chiueh. Architecture and Algorithms for an IEEE 802.11-based Multi-channel Wireless Mesh Network. In *IEEE INFOCOM*, March 2005.

[98] Ananth Rao and Ion Stoica. An Overlay MAC Layer for 802.11 Networks. *Technical Report, UC Berkeley*, 2004.

[99] Ananth Rao and Ion Stoica. An Overlay MAC layer for 802.11 Networks. In *ACM MOBISYS*, Seattle,WA, USA, June 2005.

[100] Research and Markets. The Wi-Fi Chipset Market: Portable Connectivity Applications Drive Volumes. 2007.

[101] Dr. Don Richardson, Ricardo Ramirez, and Moinul Haq. Grameen Telecom's Village Phone Programme in Rural Bangladesh: A Multi-Media Case Study. *TeleCommons Development Group*, March 2000.

[102] Luigi Rizzo. Effective Erasure codes for Reliable Computer Communication Protocols. *ACM CCR*, 1997.

[103] JW Robinson and TS Randhawa. Saturation throughput analysis of IEEE 802.11 e enhanced distributed coordination function. *Selected Areas in Communications, IEEE Journal on*, 22(5):917–928, 2004.

[104] Maya Rodrig, Charles Reis, Ratul Mahajan, David Wetherall, and John Zahorjan. Measurement-based characterization of 802.11 in a hotspot setting. In *ACM SIGCOMM E-WIND Workshop*, 2005.

[105] W.H. Sheen, C.C. Tseng, and J.S. Ho. Burst synchronization of slotted random access with preamble power ramping in the reverse link of CDMA systems. *Wireless Communications, IEEE Transactions on*, 2(5):953–963, 2003.

[106] Anmol Sheth, Christian Doerr, Dirk Grunwald, Richard Han, and Douglas Sicker. Mojo: a distributed physical layer anomaly detection system for 802.11 wlans. In *ACM MOBISYS*, New York, NY, USA, 2006. ACM Press.

[107] Anmol Sheth, Sergiu Nedevschi, Rabin Patra, Sonesh Surana, Lakshminarayanan Subramanian, and Eric Brewer. Packet Loss Characterization in WiFi-based Long Distance Networks. *IEEE INFOCOM*, 2007.

[108] J. Snider, R.P. Hill, and D. Martin. Corporate Social Responsibility in the 21st Century: A View from the World's Most Successful Firms. *Journal of Business Ethics*, 48(2):175–187, 2003.

[109] Lakshminarayanan Subramanian, Ion Stoica, Hari Balakrishnan, and Randy Katz. OverQoS: An Overlay Based Architecture for Enhancing Internet QoS. In *USENIX/ACM NSDI*, March 2004.

[110] Lakshminarayanan Subramanian, Sonesh Surana, Rabin Patra, Melissa Ho, Anmol Sheth, and Eric Brewer. Rethinking Wireless for the Developing World. *Hotnets-V*, 2006.

[111] Sonesh Surana, Rabin Patra, Sergiu Nedevschi, and Eric Brewer. Deploying a Rural Wireless Telemedicine System: Experiences in Sustainability. *IEEE Computer*, 41(6):48–56, June 2008.

[112] Sonesh Surana, Rabin Patra, Sergiu Nedevschi, Manuel Ramos, Lakshminarayanan Subramanian, Yahel Ben-David, and Eric Brewer. Beyond Pilots: Keeping Rural Wireless Netorks Alive. *USENIX/ACM NSDI*, 2008.

[113] L. Tassiulas and S. Sarkar. Maxmin fair scheduling in wireless networks. *IEEE INFOCOM*, 2002.

[114] The Technology and Infrastucture for Emerging Regions (TIER) Research Group. http://tier.cs.berkeley.edu.

[115] Athanasia Tsertou and David I. Laurenson. Revisiting the Hidden Terminal Problem in a CSMA/CA Wireless Network. *IEEE Transactions on Mobile Computing*, 7(7):817–831, 2008.

[116] Teresa Sheausan Tung. *QoS in Wireless Networks*. PhD thesis, University of California at Berkeley, November 2006.

[117] United Nations. The Global Alliance for Information and Communication Technologies and Development. `http://www.un-gaid.org/`.

[118] United Nations. UN Millennium Development Goals. `http://www.un.org/millenniumgoals/`.

[119] Vieri Vanghi, Aleksandar Damnjanovic, and Branimir Vojcic. *The CDMA2000 System for Mobile Communications: 3G Wireless Evolution*. Prentice Hall, March 2004. ISBN: 0131416014.

[120] Venugopal V. Veeravalli and Andrew Sendonaris. The Coverage-Capacity Tradeoff in Cellular CDMA Systems. *IEEE Transactions on Vehicular Technology*, 48(5):1443–1450, September 1999.

[121] Weizhao Wang, Yu Wang, Xiang-Yang Li, Wen-Zhan Song, and Ophir Frieder. Efficient Interference-Aware TDMA Link Scheduling for Static Wireless Networks. In *ACM MOBICOM*, September 2006.

[122] M. Warschauer. *Technology And Social Inclusion: Rethinking the Digital Divide*. MIT Press, 2004.

[123] B Wellenius. Sustainable Telecenters: A Guide for Government Policy. *The World Bank Group, Private Sector and Infrastructure Network, Note 251*, 2003.

[124] World Bank and IIM, Ahmedabad. Grameen Telecom: The Village Phone Program. *World Bank Empowerment Case Studies*, 2003.

[125] Y. Xiao. Performance analysis of priority schemes for IEEE 802.11 and IEEE 802.11 e wireless LANs. *Wireless Communications, IEEE Transactions on*, 4(4):1506–1515, 2005.

Appendix A

Other Technologies for Rural Network Connectivity

In this appendix we discuss some of the most promising alternative technologies for providing network coverage in developing regions. Although there is an abundance of networking technologies that could be used, many of them are not designed for environments with sparse and financially constrained users, making them economically unviable [37] (cellular technologies, WiMax, fiber). Other technology, specifically designed for these types of environments (CorDect [19], ham radios) are inexpensive, but provide low bandwidth, inadequate when sharing connectivity among many users, or when using high-resolution audio/video applications.

Besides WiLD networks, two other technologies show a lot of promise in providing cheap, good quality network connectivity that is both cheap and to remote ares: VSAT and CDMA450. VSAT has traditionally been the only choice for connecting areas that lack other forms of communication infrastructure. CDMA450 is a much newer technology, and can provide a combination of inexpensive long-range communication and broadband performance, making it a good alternative in regions that allow 450MHz spectrum licensing. We discuss the advantages and disadvantages of these in the following.

A.1 VSAT

One of the traditional options for providing network connectivity in remote places has been the use of Very Small Aperture Terminal (VSAT) stations.

A.1.1 Overview

A VSAT station is a two-way satellite ground station with a dish antenna that is smaller than 3 meters (most VSAT antennas range from 75 cm to 1.2 m, hence the name). VSATs access geostationary satellites (satellites in geosynchronous orbit), to relay data from some small earth terminals to other small terminals (in the case of mesh configurations), or to very large, "master" earth stations (in the case of a star configuration).

VSAT datarates vary depending on technology, but they typically range from narrowband to at most 4 Mbit/s. The upload and download channels are asymmetric, with higher download rates.

Nearly all VSAT systems are now based on IP, with a very broad spectrum of applications. VSAT has been used in conjunction with application requiring easy narrowband communication, such as point of sale transactions, but also for providing broadband Internet services in remote locations. As of December 2006, the total number of VSAT stations in service exceeded 1 million, with VSAT service revenues exceeding US$3.88 billion [47].

VSAT networks can be configured in one of the following ways: *a) star topology*, featuring a large, central hub, connecting to small terminals via satellite, *b) mesh topology*, with VSAT terminals connected directly with other small terminals, or a combination of the two.

A.1.2 Advantages

VSATs present several advantages when deployed in remote areas of developing regions. These are:

- *Great coverage:* These systems can be easily deployed in any location that has a clear view to *Clarke Belt*, which is most of the locations on earth.
- *Independence on wireline infrastructure:* VSATs don't require additional local network infrastructure, and are essential in situations where traditional infrastructure fails (*e.g.,* in case of natural disasters).
- *Efficient multicast and broadcast:* Modern VSAT systems support efficient broadcast mechanisms, such as the one specified in the DVB-S standard [51], enabling the delivery of the same content to many users, at no additional cost. This can be especially relevant for applications such as broadcasting educational content for remote education.

A.1.3 Disadvantages

On the other hand, VSAT systems also feature a number of significant shortcomings, both in terms of cost and performance:

- *High cost:* In developing regions, VSAT equipment installation can costs over US$10,000, with a recurring monthly cost of over US$2,000. This makes the technology only available to a few private businesses and government-subsidized institutions.

- *High Latency:* Given the large distance that the wireless signal has to travel to reach a geostationary satellite, orbiting at 22,300 miles above the Earth's surface, VSAT links experience round-trip times of 500 milliseconds at minimum, making them a poor choice for interactive applications. This effect is partly alleviated by hardware acceleration of a few protocols, such as TCP (using advance spoofing of acknowledgment packets), and HTTP (using pre-fetching of recognized HTTP objects). However, these acceleration mechanism do not help in applications involving interactive voice and video content, such as VoIP, telemedicine, and so on.

A.2 CDMA450

CDMA2000 [119] is a family of third-generation CDMA cellular communications standards that supports voice and data traffic. *CDMA20001x* (also known as *1x*, *1xRTT* or *IS-2000*) is the core air interface standard of CDMA 2000 and it uses a single pair of radio channels (1.25MHz each for forward and reverse links) to transmit both voice and data with a peak data rate of 153 Kbps in each direction.

The newer data standard called *CDMA 1xEVDO* (*Evolution Data Optimized*) [73] adds capabilities of high speed data services to CDMA 2000 by devoting a second pair of channels for packet switched data transmission. The first version called Release 0 offers peak data rates of 2.44 Mbps on the forward link (base station to handset) and 153 Kbps on the reverse link (handset to base station). The newer version known as Revision A will offer higher speeds (3.1 Mbps for downlink and 1.8 Mbps for uplink).

The technology that is most interesting for rural emerging regions is *CDMA450*, which is standard CDMA2000 technology operating in the 450MHz band. As a result CDMA450 can offer the same range of high speed data technologies such as 1xEVDO, but at a potentially lower cost by taking advantage of the lower carrier frequency, which features better signal propagation, and thus allows for larger, fewer cells.

Today, CDMA2000 has 264 million subscribers in 58 countries (in 2006 [89]). In addition to that, high speed 1xEVDO data services are now availed by about 42 million subscribers in 24 countries. CDMA450 however is a relatively young technology; the first commercial CDMA450 deployment was launched in Romania by Zapp Telemobil [18] in December 2001. Since then, its popularity has increased rapidly. Zapp estimates that currently there are about seven million CDMA450 subscribers worldwide, and as of June 2006 the CDMA Development Group (CDG) [3] reported that commercial CDMA450 networks have been deployed in 18 countries around the world and an additional 27 networks will soon be launched or are undergoing trials.

A.2.1 Advantages

There are several well known reasons why CDMA450 is an appropriate connectivity solution for rural areas. Some of these reasons arise from the advantages in using CDMA technology regardless of the frequency, while others arise from the particular characteristics of the 450MHz frequency spectrum.

In this section we present the main CDMA450 strengths, and the resulting implications for providing rural connectivity. In the next section, we revisit some of these strengths by also considering the practical limitations that can prevent real deployments from reaching peak performance capabilities.

- *Large cell size* The main advantages of using 450MHz are its superior propagation characteristics and better penetration compared to commonly used frequencies (800 / 900 / 1800 / 1900MHz), leading to longer ranges; the resulting larger cell sizes reduce the number of base stations required to cover a given area. Table A.1 shows typical cell radii for various frequencies according to an International Telecommunication Union (ITU) study cited in a CDG white paper [38]. Thus, using 450MHz has the potential to provide significant savings in upfront capital expenditure (CapEx).

Frequency (MHz)	Cell radius (km)	Cell area (km^2)	Relative Cell Count
450	48.9	7521	1
850	29.4	2712	2.8
950	26.9	2269	3.3
1800	14.0	618	12.2
1900	13.3	553	13.6
2500	10.0	312	24.1

Table A.1: Cell Radius vs. Frequency

- *Flexible cell size* CDMA technology manages the trade-off between capacity and coverage by leveraging its embedded power control mechanism and adapting the cell size dynamically to either serve very long ranges (when top capacity is not needed), or to deliver high capacity (where top coverage ranges are not needed). The effect of dynamically shrinking the cell range, also called "cell breathing", happens when, as more users connect to the base station, the effective cell range decreases. As described by Veeravalli and Sendonaris [120], an increase in the number of active users in the cell causes the total interference seen at the receiver to increase. This requires an increase in received power seen at the base station for each user, since each user has to maintain a certain signal-to-interference ratio at the base station for satisfactory performance. However as there is a maximum limit on the transmit power from the terminals, an increase in the number of users results in a decrease in the maximum distance a handset can be placed from the base station.

  Although cell breathing can preclude the system from simultaneously achieving maximum capacity and maximum coverage, in practice the scenarios where both high capacity and large cell size are simultaneously needed are very rare. In rural environments with sparse users, coverage is the determining factor, and the system is never used to capacity; on the other hand, in urban morphologies, capacity is the driving factor, and cell sizes can be very small, with each of them being used to capacity.

  The ability to have flexible cell sizes relaxes a number of constraints related to cell placement planning, and offers a way to enable cost-effective deployments in both rural and urban environments, essential for the successful adoption of the technology.

But fortunately, CDMA450 is competitive for both rural and urban environments; operators can use the same technology to simultaneously target both urban and rural markets.

- *Well suited for low-density rural deployment and high-capacity urban deployment* We have already argued that CDMA450 is well suited for rural areas given that it enables a larger range per base station, ideal for covering low density subscriber regions. It is also known that CDMA450 can provide high capacity in an efficient manner. In terms of spectral efficiency, CDMA450 has the same characteristics as standard CDMA2000. Although a direct "apples to apples" comparison is difficult (and very controversial) to make between competing cell phone standards, most studies have shown the CDMA2000 technology to be among the most spectrally efficient; one study by Deutsche Bank cited in a 2003 industry briefing [88] ranks it the best with respect to spectral efficiency. In general, CDMA450 can deliver high capacity efficiently, saving on spectrum licensing cost.

A.2.2 Disadvantages

Although CDMA450 is a good alternative for rural connectivity, here are a number of practical limitations that make CDMA450 challenging to deploy in such scenarios.

- *Less than maximum cell range* An important advantage of CDMA450 in rural settings is the extended cell range, sometimes exceeding 50 km. Unfortunately, CDMA is interference limited, and cell breathing prevents achieving maximum cell range at high loads. As an example, by Zapp Romania's calculations [18], operating at 50%

capacity causes a loss of 3dB in the link budget, which then results in a 20% loss in range, after which the range decreases drastically.

- *Large antennas and antenna spacing* At lower frequencies, the required antenna sizes are much larger, both at the base stations and the handsets. This results in larger and heavier handsets and more expensive base stations. The use of a low frequency carrier also affects implementation of antenna receive diversity for both the wireless handsets and base stations. Receiver diversity involves combining the signals from multiple receive antennas to enhance the quality of the received signal.

  At the base stations, cellular systems use several diversity techniques in order to improve receiver performance in fading channel environments. Among these, *spatial* and *cross polarization* diversity are the preferred techniques. For spatial diversity, antenna elements need to be well separated in order for their respective channel fading processes to be uncorrelated. It has been determined through measurements that horizontally spaced antennas need to be separated by 10 to 30 times the wavelength in order for the correlation between antenna observations to be less than 0.7 [26]. At 850MHz this corresponds to an antenna spacing of 3.5 to 11 meters, which is challenging, but achievable; however at 450MHz this translates to distances of 6.5 to 20 meters, which is much more challenging to deploy, given that antennas must be mounted on the same tower. Thus, CDMA450 operators are usually constrained to employ cross-polarity diversity, which provides slightly less of a gain than spatial diversity.

  Similar concerns also affect handsets. Due to the larger handset antenna sizes, it is difficult for CDMA450 handset producers to design small handsets featuring antenna diversity. To compete with small 850MHz handsets, 450 mobile handset producers might have to drop handset antenna diversity altogether, or might have to opt for suboptimal antenna sizes and spacing, making the phones less sensitive overall.

- *Low volumes for CDMA450 terminals* CDMA450 relies on established technology from the CDMA2000 family, which means that most of the network equipment enjoys the benefits of high-volume production. However, when it comes to CDMA450 terminals, the production volume is still considerably lower than production volume for competing bands (850/1900). Out of the 318 million CDMA phone users worldwide, only 7 million are CDMA450, and there are fewer manufacturers involved in the CDMA450 terminal market.

- *Large required customer base* As with any other cellular technology, in order for the deployment to be cost-efficient, the customer base (and thus the scale of the deployment) must exceed a certain threshold, at which the investment in the core network is justified financially. In other words, any CDMA450 must exceed at least a few tens of thousands of users; a small, incremental rural-only deployment targeting only a few hundreds or thousands of users cannot be envisioned. This is especially true for deployments supporting voice, where the core network includes circuit switching components, which come in coarser granularity of incremental capacity, and are much more expensive than IP-based data switching equipment. An EVDO-only deployment, however, can be of smaller scale since it only requires cheaper data switching equipment at the core.

A.3 WLL using CDMA450

Another alternative for connectivity in rural areas is the use of CDMA450 in Wireless Local Loop (WLL) deployment model. This model takes advantage of standard CDMA450 technology, but fixes the location of terminals, allowing for additional increases in cell range. Although the use of the 450MHz spectrum enables large cell sizes, we have seen that sometimes cell breathing and the requirement for large antennas can reduce the maximum cell size achievable in practice. Luckily, in rural scenarios, additional range extensions are possible by adopting a fixed wireless deployment model, often called *wireless local loop* (WLL). In addition to the potential cost advantages of WLL, we argue that a fixed wireless model can sometimes be better suited for data transfer in rural areas where devices will be shared at fixed locations such as Internet cafes or kiosks.

Let us look at a few technical improvements achievable with fixed wireless terminals that are otherwise impossible using mobile wireless terminals:

A.3.1 Increased effective transmit power at terminals

In cellular systems the transmission power of handsets is orders of magnitude smaller than the one of base stations (0.2W vs. 23W) and thus the reverse data link (from terminal to base station) is often the range bottleneck.

- *Directional Antennas*: The fixed location of a WLL terminal enables the use of directional antennas pointed at the omnidirectional base station. In the 450MHz frequency band, inexpensive Yagi directional antennas with gains of 9–12 dB are easy to deploy.

- *Higher transmit power in terminals*: A significant advantage of fixed terminals is the

CDMA450 standard that allows them to transmit at much higher powers compared to the limited power output of mobile handsets. Increasing the terminal transmit power from 200mW (typical for a mobile handset) to values of 500mW or even 2W (as supported by some WLL cellular terminals) can significantly increase range and signal quality. These improvements can add 4–10 dB to the reverse link budget.

By using these two techniques, terminals would be able to maintain the same SINR (Signal to Interference Noise Ratio) at the base station from farther distances thus negating the cell breathing effect.

A.3.2 Better receive signal at receivers

- *Directional Antennas:* The use of these antennas symmetrically increases the quality of the received signal as well.
- *Receive Diversity:* Multiple receiver antennas connected to a single terminal can be used to boost receiver signal strength. For example, a Yagi and an omni antenna can be attached to a single terminal, which uses processing to combine the signals from the two antennas. As shown in the following section, significant forward link improvements can be seen by using receive antenna diversity.

To quantify the range and bandwidth increases in a WLL deployment model, we perform detailed range measurements in a real CDMA450 deployment, in collaboration with Zapp Romania [18]. Our findings, discussed in detail in Appendix A.3.2 illustrate how the use of directional antennas and antenna diversity extends cell range and increases link quality, especially at the edge of the cell. Our experiments also show that, given favorable geography, CDMA450 and CDMA450 EVDO links can exceed ranges of 50km and maintain good download and upload performance.

Appendix B

Range Experiments with WLL CDMA450

In order to measure the improvements caused by using directional antennas and dual receive antennas at the fixed terminal side, we perform several range tests for various cell sites in the ZAPP CDMA1xEVDO cellular network.

We present our findings for a site placed atop a large hill at a ground elevation of 727 m. The antenna height is 30 m relative to the base of the tower on the hill. We use this base station location because it is situated in a rural area and overlooks a wide plain with few obstacles or changes in altitude. This provides us with a more uniform testing environment, where the signal strength varies relatively uniformly with distance. In these measurements we do not exceed distances of 50 km, a software limitation imposed at the base station, which in general is avoidable with some reconfiguration.

For the fixed terminal, we use the Zapp-branded Z020 wireless EVDO modem with

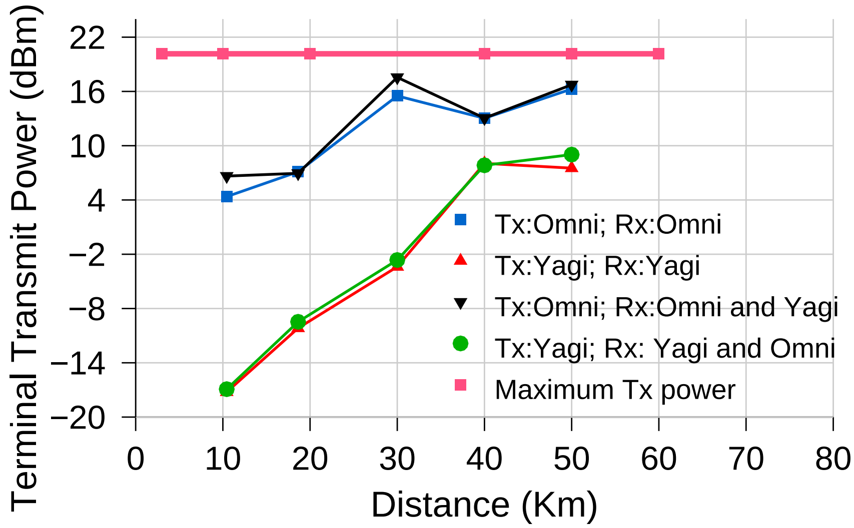


Figure B.1: Phone TxPower vs. Distance for various antennas

receive diversity enabled. At all measurement sites the terminal antennas are mounted at a height of 2m above ground level. Given that the surrounding area is a wide plain the ground elevation of all the measurement sites is roughly the same.

We configured our terminal and target base station to communicate on a channel reserved for testing, and not used by any nearby base stations or terminals, thus avoiding interference with any other source. The terminal was connected to a laptop running Qualcomm's QCait software to collect information such as the Signal-to-Noise-plus-Interference Ratio (SINR) and the terminal transmit power (Tx Power) at the terminal.

We start by investigating potential increases in EVDO cell radius. We compare four scenarios: the first uses a single 1.5 dB gain omnidirectional antenna; the second uses a single 9 dB directional Yagi antenna, aligned towards the base station; the third utilizes a combination of two antennas, in which the omni is used as the primary antenna, both for

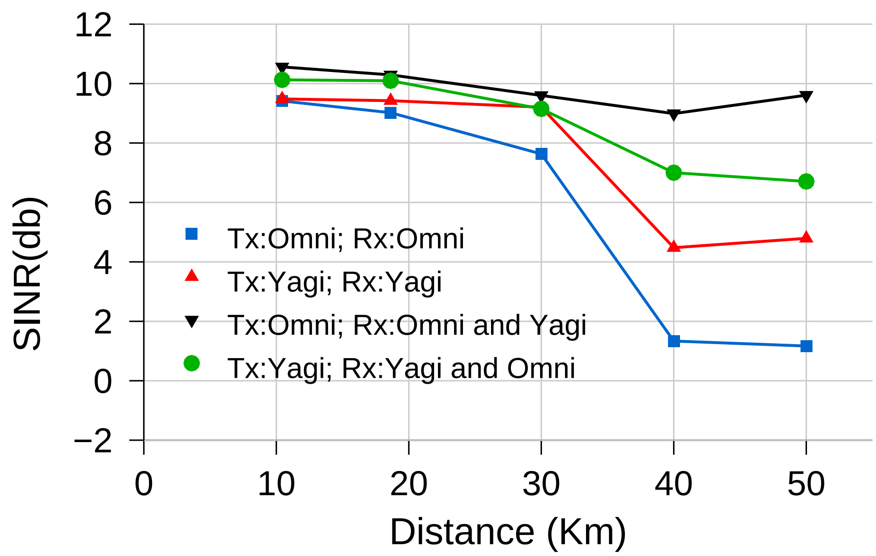


Figure B.2: SINR vs. Distance for various antennas

transmit and receive, while we use an additional Yagi only for receive diversity; finally, the fourth scenario has the Yagi as the primary antenna, while the additional omni is used for diversity.

In each of these setups we measure the average transmit power used by the wireless terminal when sending data packets to the base station. This is relevant because it reflects how close the terminal is from the cell edge. In order to properly receive the terminal's packets, the base station requires the terminal to send packets at a specific power level, which is higher as the terminal is closer to the cell edge. When the required transmit power exceeds the maximum possible power output, the terminal is effectively out of range. The results of our first experiment are presented in Figure B.1, and show that the determining factor for the transmit power, and thus cell size, is solely the antenna used as the primary. The antenna used for diversity makes no difference in this case. In both of the scenarios

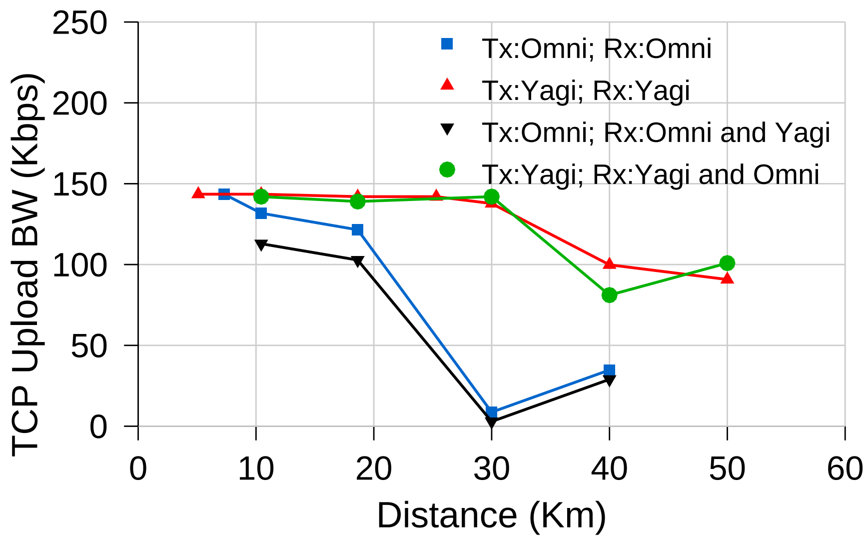


Figure B.3: Upload throughput vs Distance for various antennas

that use a Yagi as the primary, the power output required is much lower than for the cases when an omni is used for transmission. At large distances (50 km), the transmit power required is close to its maximum when the omni is used at the primary, while there is a significant reserve (in power and thus range) when the directional antenna is used. We conclude that all configurations can function at ranges of up to 50 km, and this range can be further increased with Yagi antennas.

Even though it is not a bottleneck in terms of range, we also examine the effect of different antennas on the forward link, by measuring the SINR (Signal to Interference plus Noise Ratio) at the terminal for each of the scenarios already presented. The results of these measurements are shown in Figure B.2.

We observe that in terms of SINR, and thus in terms of the forward link quality, the scenarios using receive diversity outperform the ones using a single antenna, with SINR

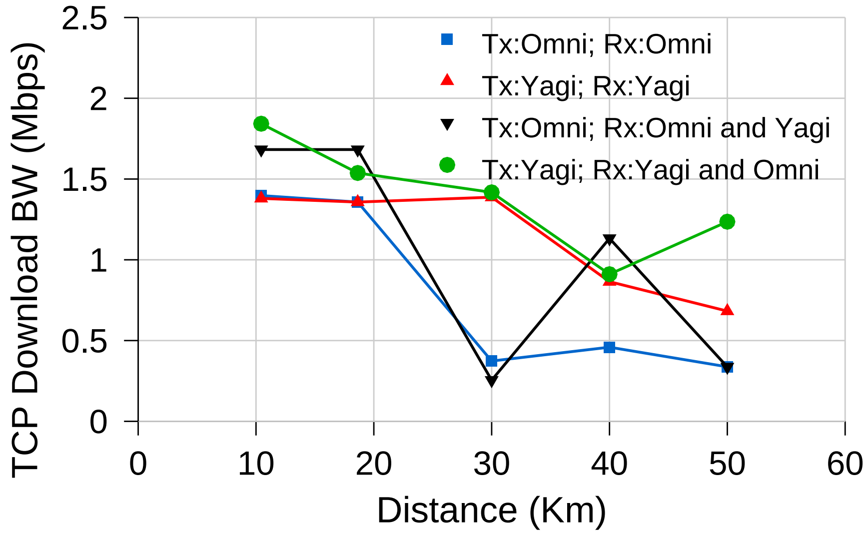


Figure B.4: Download throughput vs Distance for various antennas

improvements ranging between 2–9 dB.

We continue by examining how these effects of different antenna configurations on Tx power and SINR reflect on data transfer performance. Since TCP is the data transport protocol most used in existing network applications, we use TCP throughput as an indication of link quality.

The TCP upload measurements, presented in Figure B.3, can be correlated very well with the transmit power measurements, confirming that the quality of the reverse link is only determined by the gain of the primary antenna, which is good for directional Yagis, and bad for omnidirectional antennas.

Finally, we investigate the effect of various antennas on the TCP download speeds, and we present the results in Figure B.4. A slightly unexpected result is that, same as for uploads, the configurations using the Yagi as the main antenna perform better. This is surprising because one might think that SINR is the determining factor for the forward link performance. This effect can be explained by the fact that the download throughput is influenced by the upload throughput if TCP is used, since TCP acknowledgments travel in the reverse direction. The receive diversity is also useful, the best performance being obtained using a Yagi on the primary with an additional omni for receive diversity.

In conclusion, we show that: a) directional antennas can be used to increase cell range; b) both receive diversity and directional antennas improve link performance, especially close to the cell edge, and c) in single-user conditions, CDMA450 EVDO can exceed ranges of 50 km and maintain good download and upload performance.