Real-time services and multihop networks

Delay analysis of a multihop CDMA fixed relay network

IVAR GAITAN

KTH Information and Communication Technology

Master of Science Thesis
Stockholm, Sweden 2005

IMIT/LCN 2005-21

Real-time services and multihop networks

Delay analysis of a multihop CDMA fixed relay network

IVAR GAITAN

Master of Science Thesis performed at
Wireless Signal Processing and Networking Laboratory
Sendai, Japan
—
Examiner: Gerald Q. Maguire Jr.

Abstract

The next generation of cellular networks is expected to carry high speed IP traffic in a packet switched environment in order to accommodate data traffic for a wide range of services. Much research is thus focused on how to increase the transmission rate in cellular networks. A general consequence of increasing transmission rate in a radio link is a corresponding increase in transmit power, resulting in increased interference and reduced network capacity. The straightforward way to reduce this effect is to shorten the link distance, which in the cellular case means shrinking the cells. This, however, means increased control traffic for handoffs and location registration, as well as increased infrastructure costs.

An alternative approach, which has received increased attention lately, is the introduction of wireless multihop access networks to relay traffic between the wired infrastructure and the users. Such an access network must clearly accommodate all types of services expected to operate in next generation systems, including real-time services. However, wireless multihop networks have traditionally had problems meeting the delay requirements posed by such services.

In this thesis, we will study the delay performance through analysis and simulation of such a network, based on the Virtual Cellular Network proposal [1], in which geographically fixed wireless relays are deployed to act as both network nodes and user relays.

Abstract

Nästa generations mobilnät förväntas bära paketväxlad höghastighetstrafik för att kunna stödja en stor mängd varierande tjänster. Mycket forskning har däför fokuserat på transmissionshastigheten i mobilnäten. En generell konsekvens av att öka transmissionshastigheten i ett radiobaserat nätverk är en motsvarande ökning i transmissionsstyrkan, vilket resulterar i ökad interferens och följaktligen minskad kapacitet. En lösning på det problemet är att minska avståndet mellan sändare och mottagare, vilket i fallet med dagens mobila infrastruktur innebär mindre, och fler celler. Detta skulle dock innebära både ökad kontrolltrafik och större kostnader för planering och underhåll av infrastruktur.

En alternativ väg, som den senaste tiden fått ökad uppmärksamhet, är införandet av ett trådlöst vidarebefordrande accessnätverk mellan de mobila stationerna och det fasta nätet. Ett sådant nätverk måste naturligtvis kunna ackommodera samtliga typer av tjänster som förväntas utnyttja nästa generations mobilnät, inklusive realtidstjänster. Dock har kraven på låg fördröjning hos denna typ av tjänster traditionellt inneburit problem problem för vidarebefordrande trådlösa nätverk.

I denna rapport studeras fördröjningskaraktären hos ett sådant nätverk, baserat på konceptet Virtual Cellular Network [1], i vilket geografiskt fixerade radionoder vidarebefordrar trafik mellan mobila stationer och det fasta nätet.

Acknowledgements

This thesis was written while studying at the Wireless Signal Processing and Networking Laboratory, headed by prof. Fumiyuki Adachi and assoc. prof. Eisuke Kudoh, at Tohoku University in Sendai, Japan.

I want to express my deep gratitude to the following people for, in different ways, helping me in the process of writing this thesis.

Prof. Fumiyuki Adachi for letting me write this thesis at his lab and for always giving me time for questions and discussions.

Prof. Gerald Q. Maguire Jr. for his guidance and corrections, keeping me from straying too far off track.

The Kokubun family for their extraordinary generosity towards me and for letting me stay in their temple throughout my stay in Sendai.

Haris Gacanin for his invaluable "model", his humor, and for our long rewarding discussions about life and signal processing.

The people at the lab for their patience with my strange questions and their friendship, especially assoc. prof. Eisuke Kudoh, *Lalla* Soundous El Alami, Liu-san, Takeda-san, Imane Daou, Nakajima-san, Ishihara-san, Kawauchi-san, and Ku-san.

Annette Furuskog for her love and patience with this "very boring" work of mine and for carrying our child while I was away.

Contents

Chapter 1

Introduction

This chapter will give a general introduction to the area of interest, then state the problem which this thesis focuses on.

1.1 Introduction

A lot of research has been done, and continues to be done, to increase the link speed in cellular networks. This is part of a general vision of a heterogeneous wireless infrastructure in which a roaming user continually has high-speed access to a global network via various access technologies, among which the cellular system is one such network. To increase the bitrate in a radio channel, either the frequency bandwidth, or the signal-to-noise ratio ($\frac{S}{N}$) must increase, according to Shannon's channel capacity formula [2], $C = B\log_2(1 + \frac{S}{N})$, where C is the bitrate, and B is the bandwidth. Spectrum use is regulated by national and international associations, and thus the signal-to-noise ratio must be increased to improve the bitrate. If we keep the range of the radio link fixed, this means increasing the transmit power. A higher transmit power will in turn reduce frequency reusability, by increasing the interference radius from the transmitter [3].

In this chapter, we will briefly introduce a proposal network presented by the Wireless Signal Processing & Networking Laboratory at Tohoku University in Sendai, Japan, followed by a problem statement. In chapter 2, the required background is covered, including some basic radio concepts, a detailed description of the proposed network, an overview, with examples, of real-time services and their characteristics, and finally, a review of some related work. In chapter 3, delay factors are identified, and a few different models are presented. Evaluations, both analytically and by simulation, of the different models are presented in chapters 5 and 6, and finally, in chapter 7 some conclusions are drawn together with suggestions for further study.

The proposal

An obvious way to reduce the transmit power, and hence to increase the frequency reusability, is to reduce the link distance. In the present cellular system this means reducing the cell size, with the consequence of increased control traffic for handoffs and location registration. One proposal of how to deal with these issues is the Virtual Cellular Network (VCN) [4], developed by the Adachi laboratory at Tohoku University in Sendai. Next, the basic concept will be briefly introduced, while a more thorough review is left for a separate section.

In a VCN, a number of geographically fixed relays, called wireless ports, are deployed in the area surrounding a gateway to the wired network, called a central port. These relays are responsible for providing access to the users and for relaying traffic to and from the gateway by wireless multihop routing. Since the relays are fixed, they can easily be attached to a central power supply. Hence, their total energy consumption is not an issue. As every relay can act as a direct link to mobile stations, while at the same time relaying traffic from other ports, the peak transmit power can be reduced, this reduces interference distance, thus increasing the frequency reusability. Also, a shorter radio link distance means less propagation loss due to path loss, shadowing loss, and fading (see section 2.1), which, in turn, means increased signal-to-noise ratio *and consequently, higher bitrate.*

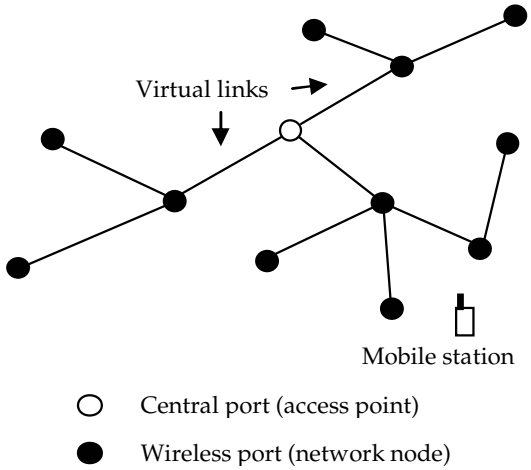


Figure 1.1: The Virtual Cellular Network concept

1.2 Problem statement

In the VCN concept, the benefits of introducing multihop increases, to a certain limit, with an increasing number of hops [5]. However, as the number of hops increases, the hop delay contribution also increases. Since such a network must be able to accommodate real-time services, the network delay must be kept below a certain level, while not impeding the beneficial effects of multiple hops.

This thesis discusses means to accommodate these two seemingly conflicting requirements – capacity and delay. The characteristics of the VCN concept and the constraints imposed by real-time services will be examined, and a network model is defined, analyzed, and evaluated.

Exercise

Suppose we need to direct a flow of real-time traffic over a network containing two wireless multihop subnetworks. This corresponds to the case with two users communicating in real-time, each at one end of such a subnetwork. Let's say that the upper bound delay contribution tolerated from these two subnetworks is set to T seconds. If all hops are identical in terms of delay contribution, we can approximate the maximum delay over each hop as $T/2M$ seconds, where M is the number of hops over one multihop subnetwork.

Considering an ideal, error-free, channel, it follows that $N_p/R_C \leqslant T/2M$, or $N_p \leqslant R_C T/2M$ where N_p is the packet size in bits, including protocol headers and trailers, and R_C is the channel capacity in bits per second (bps). To illustrate this relation, consider T=0.05s, M=5 hops, and R_C=512Kbps, reasonable values in this context. It follows that for these values, the maximum length of a packet, N_p, is 2500 bits, or 312.5B.

Bearing in mind that this was the estimate for an ideal situation, we might want to be more realistic by assuming that there is a frame error rate ρ associated with the channel and that we need to employ FEC and ARQ, possibly integrated in a hybrid ARQ model. Such a model adds redundancy to the bit stream, and provides a retransmission scheme in case of error. With a $\frac{3}{4}$ coding rate (which means that for every three bits of data, add 4 bits of redundancy) the maximum uncoded packet size becomes 1875 bits of data. The relative probability of retransmissions will further lessen this value.

Chapter 2

Background

In this chapter, some fundamental radio concepts needed to understand the basic problems presented in this thesis are reviewed. Next, a detailed presentation of the Virtual Cellular Network follows, to explain its salient characteristics. Following that, real-time services are briefly introduced, and finally, a review of some related work is presented.

2.1 Radio fundamentals

A message (encoded as a signal), analog or digital, can be sent over a radio link by combining it with a desired carrier frequency(ies), a process called *modulation*, and sending it through a transmitter. A radio signal consists of electromagnetic waves that travel through air at nearly the speed of light. The receiver will listen to the known carrier frequency, and extract the message (the decoded signal) by *demodulation*.

The signal can be distorted by the radio propagation environment. To protect a digital message from errors, two methods, Automatic Repeat reQuest (ARQ) and Feed-forward Error Correction (FEC), can be applied. ARQ detects errors and requests retransmission, while FEC adds redundant data to the message to enable correction to a certain extent. Of course, these methods come at a price: ARQ will add delay to individual packets, causing jitter in their respective sessions, while FEC will lower the overall user bitrate due to its redundancy (as more bits must be sent). These two concepts will be further described below.

As a radio signal propagates through the air, there are three main sources of propagation loss that affect the received signal power: path loss, shadowing loss, and fading loss. The received signal power can be expressed as

$$P_r = L_C G P_t \tag{2.1}$$

Where P_t is the transmitted signal power, G is the antenna gain, and $L_C = L_P L_S L_F$ is the propagation loss, where L_P, L_S, and L_F denote *path loss*, *shadowing loss*, and *fading loss* respectively.

Path loss

The path loss is an average propagation loss over wide areas. It's affected by distance, carrier frequency, and land profile. This is a straightforward loss, with the simple expression

$$L_P = A r^{-\alpha} \tag{2.2}$$

where A and α are propagation constants and r is the propagation distance. α takes a value of $3 \sim 4$ in urban areas, and the base value used in the VCN research project has been 3.5.

Shadowing

Shadowing is caused by the variation of propagation conditions in a relatively small area due to environmental obstacles such as buildings and structures, but also moving objects (including people, cars, and boats). Shadowing has been empirically determined to obey a log-normal distribution.

Fading loss

Many signals that propagate via different paths from the transmitter are superposed at the receiver to produce standing waves. When a receiver moves in these standing waves, a random variation in the signal level and phase occurs. These standing waves have short wavelengths (30cm at 1GHz) and can lead to significant drops in signal power for short movements.

This phenomena, called fast fading, will affect the channel between a mobile station and an end wireless port (i.e. the first or last hop). However, fading loss is not significant when analyzing the delay in a multihop network with fixed nodes.

2.2 Virtual Cellular Network (VCN)

A VCN is designed as a wireless multihop interface between the mobile station and the wired base stations of a cellular network. A virtual cell consists of a central port, which is a gateway to the fixed network, and a number of stationary, distributed wireless relays, called wireless ports (WP) (see Figure 2.1). Traffic is relayed between the central port and a mobile station via one or more hops. If more than one hop, the traffic goes via a wireless port. The responsibilities of the wireless ports are to directly communicate with mobile stations, and to relay signals to the central port. When communicating directly with a mobile station, a wireless port is denoted as an *end wireless port*.

By introducing multihop, reduced transmit power is achieved, which in turn contributes to a reduction in the interference power to other virtual cells, and thus improves the frequency efficiency [1].

Base station groups or uplink vs. downlink

A subgroup of wireless ports in a virtual cell act as an extension of the base station (BS) to a mobile station. They will be denoted as a *BS group* (Figure 2.2). This means that they will relay all the traffic between the mobile station and the central port. As a consequence, uplink signals from the mobile station will result in separate relay signals, one from each wireless port in the BS group, while the downlink signals can be multicast to the members of the group from the central port.

In [1], where the objective was to find the theoretically achievable performance limit, it was assumed that all wireless ports in a virtual cell constitutes a single BS group, and thus simultaneously communicate with each mobile station.

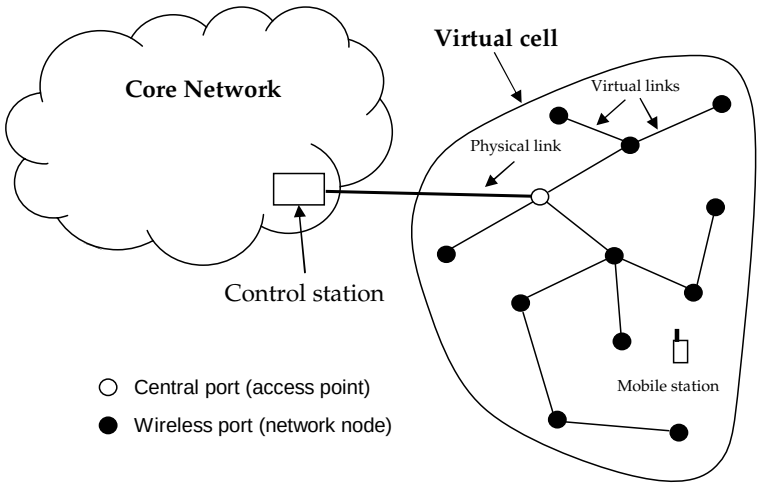


Figure 2.1: The Virtual Cellular Network concept

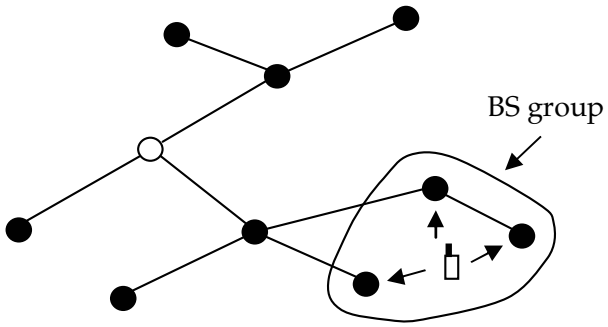


Figure 2.2: BS group

Uplink

In the uplink, the user signals received at the end wireless ports in a BS group, are relayed to the central port to be diversity combined. Two diversity combining methods are considered in [1]: Maximum Ratio Combining (MRC) and Selection Combining (SC). Using MRC, the signals from all receiving WPs are combined to produce a final version of the signal, while in SC, the signal with the largest signal-to-noise ratio is selected [6].

Downlink

For downlink transmission, the signal is being relayed from the central port, via a designated BS group, to the mobile station. For this process, two transmit diversity schemes are considered: multi-transmit diversity (MTD), and site selection diversity transmission (SSDT). In MTD, the signal is transmitted from all WPs in a BS group, using the same frequency channel, and then MRC combined at the mobile station. In SSDT, the WP with the best link is selected to transmit to the mobile station.

Multihop routing

To enable multihop routing among wireless ports, routes must be constructed between them. The construction and maintenance of multihop routes in a virtual cell is the responsibility of the virtual cell control layer. For a VCN, the route construction algorithm is designed to minimize the total transmit power of the wireless ports for relaying. Here, the total transmit power is defined as *the ensemble average of the sum of transmit powers of all WPs and the central port in an entire virtual cell* [1]. Because of the multiple transmissions of the same data in the uplink (see section 2.2), the total uplink transmit power is larger than that of the downlink, and is therefore used as the minimizing metric. A maximum number of hops is a constraining parameter of the algorithm.

Route construction algorithm

The route construction algorithm is a variant of the Bellman-Ford [7], and its basic outline is as follows:

1. Each WP periodically sends out a route construct request, containing information about the employed transmit power, source port address, number of hops traversed, transmitting port address, and the total required transmit power so far along the route.

2. Upon receiving a route construct request, a WP measures the received signal power, calculates the *required* transmit power, and adds it to the total required transmit power, before forwarding the route construct request. Additional route constructs request received from the same source, but with higher total transmit power are discarded.

3. When the central port receives a route construct request from the last relaying WP of a request chain, it responds with a route notification message containing destination port address (the next node in the chain), transmitting port address, the request source port address, and the required transmit power of the destination address.

4. The route notification message is relayed back through the request chain, setting the required transmit power for each WP, for traffic to the specific end WP that initiated the route setup.

The result is thus a point-to-multipoint routing scheme connecting all wireless ports with the central port, with minimum total transmit power and a bounded number of hops.

Simulations

Computer simulations have been performed to evaluate the VCN concept for the parameter values specified in Table 2.1 [1]. These simulations were all normalized to facilitate comparison with present cellular systems. As such, the results showed that by increasing the number of WPs in a cell (K) and the maximum number of hops (J), significant reductions in the total transmit power were achieved, regardless of the values of the other parameters. These results were due to increasing transmit/receive diversity gain.

Another simulated metric was the normalized frequency reuse distance D/R (where D is the frequency reuse distance, and R is the radius of the cell), for an allowable outage probability of 10%; D/R showed a corresponding decrease with increasing K and J.

Parameter	Comment
α	path loss exponent
σ	standard deviation of log-normal shadowing
K	number of WPs in a virtual cell
J	maximum number of hops
M	number of receive antennas
MRC/SC	uplink diversity combine method
MTD/SSDT	downlink transmit diversity scheme

Table 2.1: Simulation parameters

Disclaimer

It is important to note here, that the objective of the simulations was to find the theoretically achievable performance limit. Hence, a number of idealistic assumptions were made, such as the assumption that all WPs belong to the same BS group.

2.3 Real-time services

By definition the requirement for real-time services is a bounded delay tolerance for data delivery. This tolerance level can vary, of course, among different services, and there is no single value that separates real-time and non real-time services. Even though delay is the major constraint for real-time services, other properties such as jitter, packet loss, and bitrate can also be more or less constraining.

Communication services

Voice over IP (VoIP) does not require high bitrate and can mitigate loss to some extent due to the human intelligence. Delays up to 180 ms do not significantly affect quality, but above that threshold the quality starts to degrade [8]. Video conferencing is another service expected to be popular. Since this service is also based on human communication, it shares many properties with VoIP. However, video needs much higher bitrates and utilizes larger packets.

On-line games

Another group of real-time services is on-line games. These services are not homogeneous, and not only different classes of games, but different games within the same class, can have different network characteristics. This is basically due to varying intelligence in the software design in the respective application. In [9], two First-Person Shooter (FPS) games were measured - Halo and Quake3 - and it was found that while both games performed acceptably with up to 200ms delay, Halo would break down at 4% packet loss, while Quake3 was not degraded even at a 35% loss.

Current research

Much research has focused on voice communication. This is not only due to the expected user popularity, but also because of the extensive knowledge about its traffic model - audio packet sizes are generally uniform and the characteristics of human conversation are well known. Other human communication based services are also popular to model, while

analysis of online games appears relatively rarely in scientific papers dealing with real-time services.

2.4 Related work

The areas where wireless multihop networks are featured include mesh networks, ad hoc networks, and cellular network research. These areas sometimes overlap, particularly within cellular network research, where both ad hoc, and mesh networks have been proposed as complementary technologies. In this section I will describe these different areas, and how wireless multihop is being used. In particular, I will focus on relevant cellular network research activities.

Cellular networks

While the subject is not new, the last few years have seen a lot of papers published related to multihop assisted cellular networks. A few date back to the 20th century, but most have been published during the last five years. Many proposals include ad hoc architectures where the users themselves relay traffic to and from each other [10], [11], [12], [13], [14], and [15], while some introduce fixed relaying stations [16], [17], and [18]. A few proposals have been presented to extend the present 3G network with WLAN technology [14], [15], while most proposals that focused on next generation networks are hardware agnostic.

For simplicity, I will here discuss the ad hoc and fixed approaches separately.

A general overview of the subject of wireless multihopping has been produced by Yanikomeroglu in [19], where the concept of *intelligent relayers* is introduced. Such relayers are characterized by their selective (as opposed to blind) relaying, channel selection, and macrodiversity.

Number of hops

The two-hop model, where only one relay is introduced in the path between a mobile station and the base station was proposed in some papers. When the number of hops is not limited to two, but bounded, numbers of hops are not considered at all [13].

Mobility of nodes

Even though the majority of the papers suggest using an ad hoc network to extend the coverage, few of them analalyze the actual impact of mobility of the nodes.

Choice of metric

The choice of metric often reflects the conclusion. In [20], cost is the central metric, and hence, the conclusion is based on how well the suggested approach avoids high costs. In [13], several metrics are evaluated (spatial reuse, throughput, power, throughput per unit power, fairness, and impact of mobility), while [17] and [18] evaluate only the capacity.

Topology

The ad hoc approaches studied all assume a random distribution of the nodes, while [18] considers a poisson distribution, and [17] introduces a structured tree topology.

Ad hoc relaying

Ad hoc networks have no fixed infrastructure, apart from possible access points, which makes them desirable from an economic perspective. However, the mobility of the nodes makes them vulnerable to topology changes and limits their energy consumption, since they have no fixed power supply. Another issue is that when acting as a relay station, single radios will degrade throughput performance, because they can only send or receive at one time.

One interesting aspect of the ad hoc research mentioned is that in only one of the papers mentioned above is the actual mobility of the nodes considered [13]. Without mobility in an ad hoc network, the result becomes a fixed relay multihop network with a lot of unnecessary routing overhead.

In [11] and [13], a pure ad hoc network is found to be unsuitable for extending a cellular network. In [11], the conclusion is that an ad hoc network's performance depends on user density, and that a required QoS level cannot be cost efficiently guaranteed, while [13] concludes that the improved spatial coverage does not translate into better throughput performance without special modifications to the design.

Fixed relays

Networks with fixed relays do not have to cope with topology changes, although variations in channel quality must be dealt with. Additionally a fixed power supply means they do not have to worry about battery life. This gives the designer freedom to add complexity to the nodes, such as multiple radios and additional intelligence, but also adds infrastructure cost. Fixed relay networks have been described in [16], [17], and [18].

In [16], cooperative relaying using diversity to enhance capacity is presented. The topic of routing and channel selection is covered but the conclusion is that for a fixed-relay infrastructure, such issues are mostly trivial. The authors briefly introduce their wireless media system (WMS), which is a fixed-relay multihop network architecture, but it is not reviewed in detail.

The network in [17] utilizes very directional (beam) antennas for point-to-point communication between the relay stations. This results in higher quality channels for the relay links, but also costs in terms of network planning and configuration. These costs, however, are not covered in the paper.

In [18], relay nodes are distributed in a random fashion (Poisson process), and up- and downlink transmission are frequency separated, while the per-hop communication in either direction is possible through TDD, i.e. relays along a link chain must be synchronized.

Ad hoc and mesh networks

An abundance of research has focused on ad-hoc networks during the past decade. Since node mobility is a salient feature, the majority of the research has treated issues such as route discovery, link maintenance, and failure recovery. For implementation of such networks, standard WLAN hardware has mostly been adopted, such as 802.11 and HiperLAN, where relaying functionality is included.

The goal of mesh networks has been wireless connectivity in the present world. Thus, similar to the ad hoc case, much work has focused on how to adopt existing standards to wireless multihop.

A central problem with the existing, standard compliant, WLAN radios is that they operate on a single channel [21]. Hence, they cannot transmit and receive at the same

time, and multiple relay hops result in reduced capacity. Another limitation is the back-off algorithm coupled with the interference range, which, in times of congestion, results in unfairness and blocking [22].

The straightforward solution to this problem is to use more than one radio at each node, tuned to different channels. Although some limitations have been indicated [23], this solution has been a central theme of several research proposals [24], [25], and has recently been offered in a commercial product [26]. An interesting feature of this commercial mesh network is that the backhaul, i.e. the inter-node links, operates on a separate band from the users, which removes such sources of interference.

Relevance to the subject

Ad hoc networks are interesting as an extension of cellular networks, in order to provide coverage and lower infrastructure costs. The multi-radio mesh network approach shares similarities with some of the proposed fixed relay cellular extension networks proposed.

Chapter 3

Method

3.1 Purpose

The purpose of this chapter is to present a study of the relevant multihop delay mechanisms in a wireless CDMA network. As such, the focus of our efforts will be on transmission, error control, and processing in such a network.

Modeling, analysis, and simulation are the standard scientific procedures to understand such dynamic systems. We will thus introduce a system model that allows us to analyze the behavior we want to study, and which is free from elements that would add irrelevant complexity. We need to carefully decide what should be included and what should not, and these choices must be adequately justified. Once such a system model is established, we can analytically study the mechanisms we have chosen as relevant. This preliminary analysis provides useful insights to help us design a simulation, and to understand the simulation results. The simulation was performed to verify or refute our preliminary analysis. Its purpose is to explore the model we created and to produce the foundation for our conclusions.

3.2 Considerations

VCN legacy

The VCN proposal forms a base on which this study was performed. However, there are a number of questions and issues that are still open. As a result, more or less qualified assumptions have been made where necessary. The following is a list of the most salient issues still open in the VCN research.

- Allocation of the air interface. Will users and relaying nodes transmit on the same frequencies or should there be two (or more) different bands to allocate from?
- Channel selection. When a node is about to transmit, how should the channel code be selected?
- Multiplexing. How should packets be multiplexed between the nodes?
- Diversity exploitation. Spatial diversity has been mentioned, although rather sketchy.
- Method of relaying. Whether to use amplified or decoded relaying has not been discussed yet. This issue is briefly covered in this work.
- Flow and error control.

Real-time considerations

For real-time services, delays above a given threshold imply packet loss. However, packet loss is the combination of late arrivals and packet drops, which in turn is the sum of unrecoverable errors and queue buffer overflow. Since the main reason for a late arrival is excessive queuing delay, the size of the queue buffer determines the trade-off between packet loss due to buffer overflow or queuing delay.

Since end-to-end retransmission of lost or erroneous data is costly, many real-time services have some level of tolerance to packet loss. This can for example be due to redundancy in the data flow, or in some cases packet loss concealment algorithms at the receiver. However, with longer sequences of packet loss, i.e. bursts, this tolerance decreases. This is why not only the average packet delay is interesting, but more so the burstiness of both packet delay and drops.

3.3 Network characteristics

The network we study here is a wireless network, intended to provide connectivity to mobile users. The problem of designing a means to provide connectivity to mobile stations involves a lot of considerations, including mobility effects on the channel model, power consumption, and complexity of the mobile station. In this work, however, we limit our interest to the behavior of the backhaul, i.e. the wireless network supporting the mobile users, which is not constrained by many of the issues connected with mobility.

Channel characteristics

In wired networks, multihop delay is no longer a real problem thanks to low channel error rate and high transmission rate. In a wireless network, however, the channel suffers from environmental disturbances (fading) that raise the error rate, and will fluctuate with time. Also, the channel is a shared medium resulting in constraints due to interference, which restricts transmission rates for individual flows.

In many wireless channel models, a natural assumption is that one or more of the communicating parties is mobile. In this case, fast fading becomes a serious issue, causing much of the burstiness in the channel error rate. Because of this, in a wireless network with fixed nodes, there is reason to believe that the channel model should be less bursty and less error prone. The remaining error sources are primarily attenuation due to path loss, shadowing loss, and multipath fading, due to the non-stationary environment surrounding the nodes causing effects similar to fast fading.

CDMA

One of the properties of the network we study is its use of the CDMA multiple access method. Thus, it is reasonable to establish CDM as the multiplexing scheme for the inter-node communication. An important consequence is that, unlike TDM, several packets can be transmitted simultaneously. However, the price for this feature is that where the transmission rate for each packet is equal to the link's full capacity in TDM, only a fraction of this capacity is available for each packet transmission in CDM.

Multirate

Different services with different QoS requirements must be multiplexed together in a flexible way. This means using a multirate design, as it provides different bitrates. However, this offers differentiation in coding according to the traffic's requirements.

There are two basic approaches to offer variable bitrate to different services, the use of variable spreading factor (VSF) and multicode. With VSF, the CDM spreading ratio is reduced to increase the transmission rate for each code channel. Multicode works by allocating more than one code for a flow of traffic, which is then demultiplexed and transmitted over several code channels simultaneously, and then combined at the receiver node. In both of these schemes, time multiplexing of lower bitrate services onto higher bitrate code channels can be employed.

Error control

To achieve an acceptable error rate, redundancy is applied using error correcting codes (typically FEC). While this method will reduce the error rate, there may still be fluctuations and bursts that can push the peak error rate above the acceptable level. This can be addressed using transmit power control (TPC), by increasing the transmit power and thus the signal to noise (S/N) ratio, resulting in lower BER. ARQ can also be employed, either as an alternative or as a complement, by which we speak of a hybrid ARQ scheme (HARQ), as described in section 2.1.

Forward Error Correction (FEC)

As stated earlier in this document, the application of error correcting codes is a way to add redundancy to a flow of data in order to increase the tolerance to random transmission errors. The amount of redundancy added is described by the coding rate, which gives the ratio between number of data bits in an encoded packet and the total packet size. Hence, a low rate means more redundancy bits per information bit, which, in turn, means a lower packet error rate, but also longer transmission time per bit of user data. The parameters with which the performance of a coding scheme is measured are generally rate, coding gain, and decoding complexity. The coding gain is usually measured in dB as the reduction of required signal power to achieve a certain bit error rate.

There are several families of channel coding with different performance characteristics with respect to rate, complexity, coding gain. Traditionally, two types of codes have been in common use; block codes and convolutional codes. While popular, all codes based on these types have poor coding gain performance with respect to the theoretical limit. Lately however, two other types have shown coding gains approaching this limit: turbo codes and low density parity check (LDPC) codes. Although turbo codes show higher performance than traditional convolutional and block codes, they have limited use for real-time services due to their complexity and decoding latency. Furthermore, as the code block size is reduced, as required by many real-time applications, there is a loss in performance. LDPC codes, originally proposed by Gallagher in 1963 and rediscovered in 1996, have shown coding gain similar to that of turbo codes, but with significantly less decoding complexity, and consequently less decoding latency. Hence, LDPC is a promising technique for future wireless networking.

ARQ

ARQ defines the protocol for requesting retransmission of erroneously received packets. This negotiation takes place between two error controlling entities, which usually consist of the receiver and transmitter of a signal, or the communication endpoints.

HARQ

Hybrid ARQ is the combination of the above techniques. Error correcting codes will reduce the number of retransmissions, but at the same time increases the redundancy.

An important feature that is employed by HARQ schemes is that instead of discarding an erroneous packet and hoping for a successful retransmission, the erroneous packet is stored and combined with subsequent retransmissions in order to increase the probability of repairing the error. There are two different combination strategies, Chase combining (a.k.a. Type I HARQ) [27] and Incremental Redundancy (Type II HARQ) [28], [29]. Chase combining implies that identical copies of the packet are retransmitted and combined each round, increasing the received energy per bit with every retransmission. Incremental Redundancy, on the other hand, works by using a low rate code at the transmitter, generating a lot of redundancy, but only transmitting parts of the codeword each retransmission, thus increasing the probability of successful combining at the receiver each retransmission.

Chapter 4

Model

In this chapter we define the network model that will form the basis of our analysis and simulation.

4.1 Network

In our study we will consider a network of relay nodes using code division multiplexing (CDM). Since the choice of topology is irrelevant to our purpose, we will perform our study on a tandem network model. Forwarding is performed by link layer switching, i.e. there will be no action at the IP layer between the source and the destination. We will limit ourselves to considering single user communication throughout this network. Although this will correspondingly limit our results due to neglecting the effects of multiuser contention, however, the absence of queuing delay will allow us to focus on the static delay-contributing mechanisms of relaying.

Figure 4.1: A tandem n-hop network

Let K_i be the ith node in an n-hop network, where K_1 is the source of traffic, K_{n+1} is the destination, as depicted in Figure 4.1, and n is the number of hops. The nodes use pre-assigned non-conflicting frequency bands of bandwidth W_i, as described in [30]. Across each link i there is a packet error rate ρ_i which depends on the bit error rate and the FEC coding used. Packets are sent from the source K_1 to destination K_{n+1}, over relaying nodes $K_2 \dots K_n$.

As presented in section 3.3, there are different ways to accommodate transmission rates required by the services. These techniques are not regarded as substantial delay contributors, and the choice of method is not regarded as relevant to the present study. Instead, it is assumed that a required transmission rate can be accommodated if the service is not blocked, thus we will assume the same channel transmission rate R_c across all hops.

4.2 Channel model

A common perception is that a node cannot transmit and receive simultaneously using the same channel. This is because the transmitting signal power is much larger than the receiving signal power, which would then be drowned. Thus, a condition used throughout this study is that non-interfering channels have been allocated for each link in the network.

A popular way to model a fading AWGN channel is to use the Gilbert-Elliot two-state Markov model, representing good and bad state respectively, with fixed bit error rate (BER) for each state. This model has its merits when one or both of the communicating entities are mobile, which can generate sudden and drastic changes in the channel state. However, if we consider fixed nodes, as pointed out in section 3.3, the absence of mobility renders this model inadequate. Instead we will assume slower and not so drastic fluctuations in channel state, which we also assume are successfully battled with transmit power control (TPC). The result is a relatively stable packet error rate, represented in our model as a constant.

4.3 Delay model

Bertsekas and Gallager [31] show that the packet delay within a communication subnet can be described as the sum of delays on each subnet link traversed by the packet. Each link delay, between a source node and a destination node, can in turn be divided into four components: processing, queuing, transmission, and propagation delay. The modeling of these components will be reviewed below. If a packet transmission along the path turns out to be erroneous, a retransmission may be required, inflicting further delay on the packet. The modeling of this behavior is also covered.

Processing delay

The processing delay occurs between the time the packet is received at the source node of the link and the time the packet is assigned to an outgoing link queue for transmission. This consists of decoding the signal, and possibly checking for errors if error control is performed at the node. Depending on the coding scheme, decoding can produce a substantial delay. This is especially true for most types of turbo codes. In our model we will assume fixed decoding delay for every decoding attempt τ_c, be it the first transmission or subsequent retransmissions. This corresponds to an iterative decoder with a maximum number of iterations.

Queuing delay

Between the time the packet is assigned to a queue for transmission and the time it starts to be transmitted there is a queuing delay (if the queue is not empty, or if another transmission is ongoing on this outgoing channel). The study of queuing delay has received much attention, giving rise to the subject of queuing theory. In a situation with different classes of traffic, including real-time data, priority scheduling is often applied to minimize queuing delay for delay sensitive flows.

In this section, we are interested only in static delay sources and will disregard any effects from contention and queuing. As such, we consider only the case of a single user generating all traffic.

Transmission delay

From the time that the first bit of the packet is transmitted, to the time that the last bit is transmitted, is the transmission delay. The transmission delay of a packet of size N_p transmitted over a channel with transmission rate R_c is clearly expressed as N_p/R_c.

The packet size is the sum of the link layer header size N_h and the upper layer segment size N_s, which in turn, is the sum of the application data payload and higher layer protocol headers. Throughout this study, we will refer to the header size as N_h without any regard to the specific coding rate. However, the header is often coded with a lower rate than the rest of the packet, since the information contained within it is more sensitive than the packet payload.

Propagation delay

The propagation delay is the time between the transmission of the last bit at the source node of the link and the reception of the last bit at the destination node. This is proportional to the physical distance between transmitter and receiver. In a wireless network such as the VCN, the distance between the nodes will not produce propagation delays exceeding a few microseconds, which is small enough to be neglected in our analysis.

Retransmission delay

There is a chance that a packet is received erroneously at a node. If the node employs error control, a retransmission may be requested, leading to additional delay. While error control details are covered below, retransmission delay is the time between the detection of an erroneously received packet at the destination node, and the successful correct reception of the same packet. This delay is itself compound, and includes the sending of retransmission request to the destination node, the transmission of correction data, and the process of correcting the packet. Furthermore, this process may be repeated a number of times until the packet is successfully corrected, or the maximum number of retransmission is reached. At that point, the packet may be dropped or the whole transmission/retransmission process restarts.

In the context of this study, the delay of one retransmission is modeled as the sum of the transmission delay of the ARQ request, the transmission delay of the correction bits, and the processing delay. The packet size of an ARQ request is assumed to be the same as the link layer header size N_h, while the size of the packet with correction bits is written as N_r. The delay of one retransmission is thus written $\frac{1}{R_c}(N_r + N_h) + \tau_c$, where τ_c is the decoding delay. The total retransmission delay depends on the ARQ scheme used, which is presented later in this document.

Fitzek et al. presents an error control scheme that utilizes additional CDMA code channels [32].

4.4 Channel assignment

When a packet arrives at a relaying node, it will be forwarded to the next node in the chain using code division multiplexing (CDM). Either, a code is pre-allocated for each flow, or a code is assigned for each packet transmission independently. The former case is a circuit-switched type of scheme, where a service requests the necessary resources, which can be either granted or denied depending on resource availability across the entire path, and if granted it then enjoys a guaranteed QoS level throughout the connection lifetime.

The latter, packet-switched approach, works on a best-effort basis with no QoS guarantees. However, in this case a relaying node has still some freedom in offering QoS. A variable spreading factor (VSF) scheme can be employed, lowering the spreading factor for flows with high bitrate demands, or more than one code channel can be used by such flows in a multicode fashion.

In our model, however, we study a single user case, in which there is no shortage of resources. As a result, the method of assigning channels to traffic flows becomes irrelevant to our study. A fixed transmission rate R_c is assumed for all transmissions.

4.5 Relaying model

The relaying model describes the method of forwarding packets across a multihop path between the source and the destination. In wireless systems, there are two basic methods of forwarding an incoming signal; decoded relaying and amplified relaying. Decoded relaying corresponds to the case where a forwarding node digitally decodes and re-encodes the incoming signal before transmitting it. In the amplified relaying case, the forwarding node simply amplifies the signal before transmitting it. These two models are sometimes referred to as store-and-forward and amplify-and-forward, respectively. Another relaying model that has long been employed in wired systems is *virtual cut-through*.

Amplified relaying

Amplified relaying has some advantages over decoded relaying, primarily in terms of delay, since amplification does not require the whole packet to be received before forwarding, a process that multiplicatively increases the transmission and processing delay based on the number of hops. However, the decoding of packets is crucial for the operation of the type of network we are studying. The coded packets contain information regarding TPC, routing, etc., and thus, we will not consider amplified relaying further in the present study.

Decoded relaying

As stated above, decoded relaying means that each packet in a flow must be successfully received at a relaying node before being forwarded. The consequence is that the per hop transmission and processing delays are multiplied by the number of hops in the multihop chain. Large packets and low bitrates translates into longer packet transmission times, and the total transmission delay rises with the number of relaying stations. However, shorter packets usually means lower coding efficiency, leading to either more retransmissions or lower coding rate, the later results in larger packet sizes.

This problem, which is inherent in the store-and-forward method, is naturally less significant in networks with high transmission rates.

Virtual cut-through

Virtual cut-through, or cut-through switching, was originally introduced by Kermani and Kleinrock in 1979 [33], and can be viewed as a combination of the decoded and amplified relaying models presented above. When an incoming signal is received, the forwarding node decodes only the header, to resolve the destination address, before starting the transmission. Since the header of a packet is generally much shorter than the full packet, the reduction in transmission delay is substantial as compared to decoded relaying.

One feature of virtual cut-through is that there is no buffering by the forwarding node. As such, error control must be performed by the end parties of the communication. This property, together with the significantly higher error rates found in wireless networks, adds to the average delay and jitter with such magnitude that this forwarding model has not been considered for wireless environment.

The study of cut-through in a wireless context (defined by this model) will be performed in the next chapter. We will see what parameters affect the performance of the scheme, and under what conditions it may be a feasible alternative to decoded relaying.

Since the assumed weakness of cut-through resides in its inherent error control mechanism, where high error rates will produce increasingly long delays due to its end-to-end nature. Because of this, we will look at two different approaches that address end-to-end error penalty. The two models, called trailing error control, and partial cut-through, are presented below.

4.6 Error control and coding

For our purpose we need to establish how error control is to be performed in the present model.

Control scheme

The two basic ways of performing error control is between the transmitter and the receiver over each link in a multihop chain, or between the end parties of the communication, here denoted a per-hop error control and end-to-end error control, respectively. In this work we will also introduce *trailing error control* as an alternative scheme together with cut-through switching.

Per-hop error control

Per-hop error control implies that each node along the path is checking every packet for errors. No erroneous packet is forwarded, and if the maximum number of retransmissions has been reached, the packet is dropped.

End-to-end error control

If only the end parties of the communication perform error control, there is no need to decode the entire packet at each node, and retransmissions will not be performed by intermediate nodes. The result is faster forwarding between source and destination, thus this is the most common control scheme used in wired networks.

However, when an error occurs over an intermediate link, the erroneous packet will be forwarded until it reaches the destination, resulting in resource waste and additional delay for that individual packet. Additionally, not only is the erroneous packet transmitted over multiple hops before the error is detected, but the retransmission scheme is also performed between the end nodes, resulting in longer round-trip times for the negative acknowledgement (NAK) and corresponding retransmission packet, and, perhaps more importantly, increasing the error probability of the retransmission. Consequently, this control scheme is not well suited to error prone environments. This is also a reason why virtual cut-through has not been considered for wireless networks.

Trailing error control

The aim of trailing error control is primarily to decrease the error rate of the cut-through retransmissions, and thus decrease the loss rate while *also* decreasing the retransmission round-trip delay.

The mechanism relies on the relaying nodes temporarily storing each packet they are forwarding, i.e. to simultaneously buffer the packet while transmitting it, and when the last bit is received perform error detection. If the packet is found to be erroneous it is discarded, but if it is found to be error-free the packet is stored. When the destination has fully received the erroneous packet and detected the error, a NAK is sent back towards the source. Nodes along the path that have previously detected the error and discarded the packet will simply forward the NAK, but the first node with an error-free copy will intercept the NAK and respond with a retransmission. If Incremental Redundancy (IR) HARQ (see 3.3) is employed, the relay node does not possess the original, low-rate, codeword and cannot respond with only parity bits. The alternatives are then to either start a new IR session, re-coding the packet and issuing an original transmission, or to employ Chase combining, and respond with a transmission of the whole packet (coded with the high rate of the initial IR puncturing of the source).

The desired effect of this procedure is that for errors that occur during hops after the first hop, retransmissions will traverse fewer links, and consequently experience lower error rates.

Partial cut-through

A middle course between regular store-and-forward and virtual cut-through would be to assign error control responsibility to only a subset C of all the relaying nodes R along a path, i.e. $C \subseteq R$, where the end nodes are not considered to be relaying nodes, $\{src, dst\} \nsubseteq R$. At the control nodes, store-and-forward relaying is used, but at the non-control nodes, cut-through switching is employed. In defining the set of control nodes, the two extremes representing all nodes $C = R$, and no nodes $C = \varnothing$, correspond to pure store-and-forward with per-hop error control and virtual cut-through with end-to-end error control, respectively.

A simple and straightforward method to define C is to define a value, time-to-cut-through (TTCT), to represent the maximum number of consecutive cut-through-relaying nodes a packet may traverse. TTCT is defined as an integer between 0 and $n - 1$, where n is the number of hops. Setting TTCT to zero would mean decoded relaying, and setting it to $n - 1$ would mean virtual cut-through relaying. This value could be set on a per packet basis, as a link layer header flag, or at system level as a configuration parameter.

Flow control

The control scheme not only determines the control parties, but also the method of flow control. The three most common flow control strategies are Stop-And-Wait (SAW), Go-Back-N (GBN), and Selective Repeat (SR). With SAW, the sending party sends one packet at a time and waits for an acknowledgement before proceeding with the next packet. This is a simple and easy-to-analyze scheme, used in many analyses, but it may incur substantial resource waste if the acknowledgement round trip time is long and/or the packet size is large. Go-Back-N is a sliding-window scheme, allowing the sender up to N unacknowledged packets at any point in time. If one packet is found to be erroneous by the receiving party and a NAK is issued, the sender must retransmit not only the failed packet, but all subsequent packets as well. This scheme trades increased channel utilization for decreased

error penalty. Selective-Repeat is a more complex variant of GBN, where the packets are sequenced and the sender only needs to retransmit failed packets. The receiver has consequently the responsibility of reordering the packets.

In our model, we will assume the Stop-And-Wait protocol for simplicity.

Channel coding

To realistically model the packet error rate and the decoding latency as functions of SNR, the coding rate, and the coding scheme is, unfortunately, outside the scope of this study. Instead, we will choose a single coding rate R, packet error rate ρ, and the decoding latency τ_c as independent variables in each step of our analysis (i.e. for each analytic or simulation data point). To keep the results of this work relevant, the choice of these parameters for our analysis must be carefully made.

Hybrid ARQ

The HARQ scheme that we will model is Incremental Redundancy, or type II, with m possible retransmissions. This corresponds to one initial transmission of size N_p, and m shorter transmissions of size N_r.

Chapter 5

Analysis

In the previous chapter, we described our model and discussed the choices that were made. Now it is time to perform some more detailed analysis of that model. In this chapter, we will produce an analytic expression of the average packet delay under each of the relaying models described above, namely decoded relaying with per-hop error control, cut-through switching with end-to-end, trailing error control, and finally partial cut-through.

Decoded relaying is in this study considered to be the normal relaying model in wireless networks, and we will compare the performance of virtual cut-through by that of decoded relaying. Since the other schemes can be viewed as variations of the cut-through principle, they will be compared with virtual cut-through. To perform this comparison, we will analytically derive expressions for each of the techniques and when evaluating them, their performance will be normalized by that of decoded relaying and virtual cut-through, respectively.

This chapter is mainly concerned with expressions for average delay. Other important metrics in this study are the loss rate and the packet delay distribution, which will be examined in the next chapter.

5.1 Analytical model

LaBerge and Morris derived a general expression for the mean transfer delay of an M-stage HARQ protocol [34], where M denotes the maximum number of transmissions during the course of a HARQ session, i.e. it corresponds to $m + 1$ in our model. Applying the assumptions of perfect feedback channel and zero acquisition, synchronization, and reinitialization delays, we can write the expression as

$$\overline{D} = \frac{\sum_{i=1}^{M} \left(\prod_{k=0}^{i-1} P_{Rk} \right) \Gamma_i}{1 - \prod_{k=0}^{M} P_{Rk}} \tag{5.1}$$

where P_{Rk} is the probability that a repeat request is generated after the kth transmission, Γ_i is the transmission time associated with the ith transmission, and M is the maximum number of transmissions.

We will derive all subsequent expressions in this chapter based on this generalized model. However, to fit it into our model, we will make some redefinitions. First, P_{Rk} corresponds to our packet error rate, which we have declared to be the constant ρ. Second, the transmission time associated with the ith transmission, Γ_i, corresponds in our case to either the first transmission time, or the sum of the NAK and the following retransmission. That is, we

will mostly employ the notions Γ_T and Γ_R to distinguish between the initial transmission and the subsequent retransmissions, respectively. These parameters will, however, be defined for each of the relaying models.

Loss rate

The loss rate is the probability that a packet will not be successfully received during a HARQ session. For a simple session over one link with error probability ρ, the loss rate can be expressed as ρ^{m+1}, i.e. the probability of one erroneous transmission followed by m unsuccessful retransmissions. Although not explicitly stated in [34], the loss rate is found in the denominator of eq. 5.1, which is actually the rate of successful HARQ sessions, $1 - \rho^{m+1}$. As a result, to be able to produce comparable analytical expressions for our relaying models, based on eq. 5.1, we need to derive the loss rate for each model.

5.2 Parameters

For the analytical expressions and the simulations, the following parameter values are set when nothing else is stated. The first column describes default values, while the second column describes the ranges of values.

Parameter	Default value	Interval	Step size
n	5 hops	1 - 10 hops	1 hop
m	4		
N_p	240 bytes	120 - 480 bytes	30 bytes
N_r	$N_p/2$ bytes		
N_h	24 bytes		
R_c	384 Kbps	100 - 600 Kbps	50 Kbps
ρ	0.05	0.5 - 0.001	logarithmic
τ_c	5 ms	1 - 10 ms	1 ms

Table 5.1: Parameter values

5.3 Decoded relaying

Decoded relaying, or store-and-forward, with per-hop error control, is straightforward to analyze since there are no parallel events (because we are using the Stop-And-Wait ARQ protocol). Furthermore, in our model, all hops are identical in terms of packet error rate and channel transmission rate, which allows us to derive an expression for a single hop and then multiply that result by the number of hops.

As described in section 4.3, the packet size N_p is composed by the application payload N_a, higher layer protocol headers N_e, the coding scheme of rate R, and the link layer header N_h, expressed as $N_p = \frac{1}{R}(N_a + N_e) + N_h$. Let's define R_a as the bit generation rate by the real-time application, which means that $R_{pkt} = R_a/N_a$ is the packet generation rate. This leads to a theoretical lower bound on the (code) channel transmission rate at $R_{c,min} = \frac{1}{N_a} R_a N_p$. For transmission rates below this bound, traffic will be generated faster than it can be transmitted and packet delay will increase without bound.

In our model, P_{Rk} corresponds to the packet error rate ρ, which we hold constant. Also, Γ_m is represented by two different values, the transmission time for the first transmission, $\Gamma_T = \frac{N_p}{R_c} + \tau_c$, and the sum of the NAK delay, the retransmission delay, and the processing

delay, $\Gamma_R = \frac{1}{R_c}(N_h + N_r) + \tau_c$ for the relaying delay. Furthermore, the relation between M and our maximum number of retransmissions is $m = M - 1$. With these adaptations, equation 5.1 can be transformed into

$$\overline{D}_{dr} = \frac{\Gamma_T + \Gamma_R \sum_{i=1}^{m} \rho^i}{1 - \rho^{m+1}} \tag{5.2}$$

Observations

In figure 5.1, the behavior of equation 5.2 with the values from table 5.1 is plotted. Specifically, the relationships between the average packet delay and, respectively, packet error rate ρ, processing latency τ_c, channel transmission rate R_c, packet size N_p, and number of hops n, are displayed. It should be emphasized that the results rest on some idealized assumptions, and should not be taken as realistic predictions.

However, these results indicate some trends. The decoding latency, packet size, and hop count, all have a linear relation to the average delay, with coefficients 5.3, 3.2, and 10.4, respectively. This means that for the configuration given in table 5.1, every millisecond of decoding latency corresponds to 5.3 ms of network delay, while each byte added to the packet size results in additional 3.2 ms delay, and each hop in the multihop chain corresponds to 10.4 ms of network delay. The packet error rate and the transmission rate graphs, on the other hand, indicate that there is a minimum or maximum bound, respectively, beyond which the gain is no longer worth the effort. Also, several of these parameters are interconnected, such as packet size and packet error rate, and processing delay and packet error rate.

5.4 Virtual cut-through

A transmitted packet consists of two parts, the header, containing control information such as source, destination, transmit power, etc. and the body, containing application data together with higher layer protocol headers. Both the header and body are coded separately to enable resolution of header information before decoding the packet body.

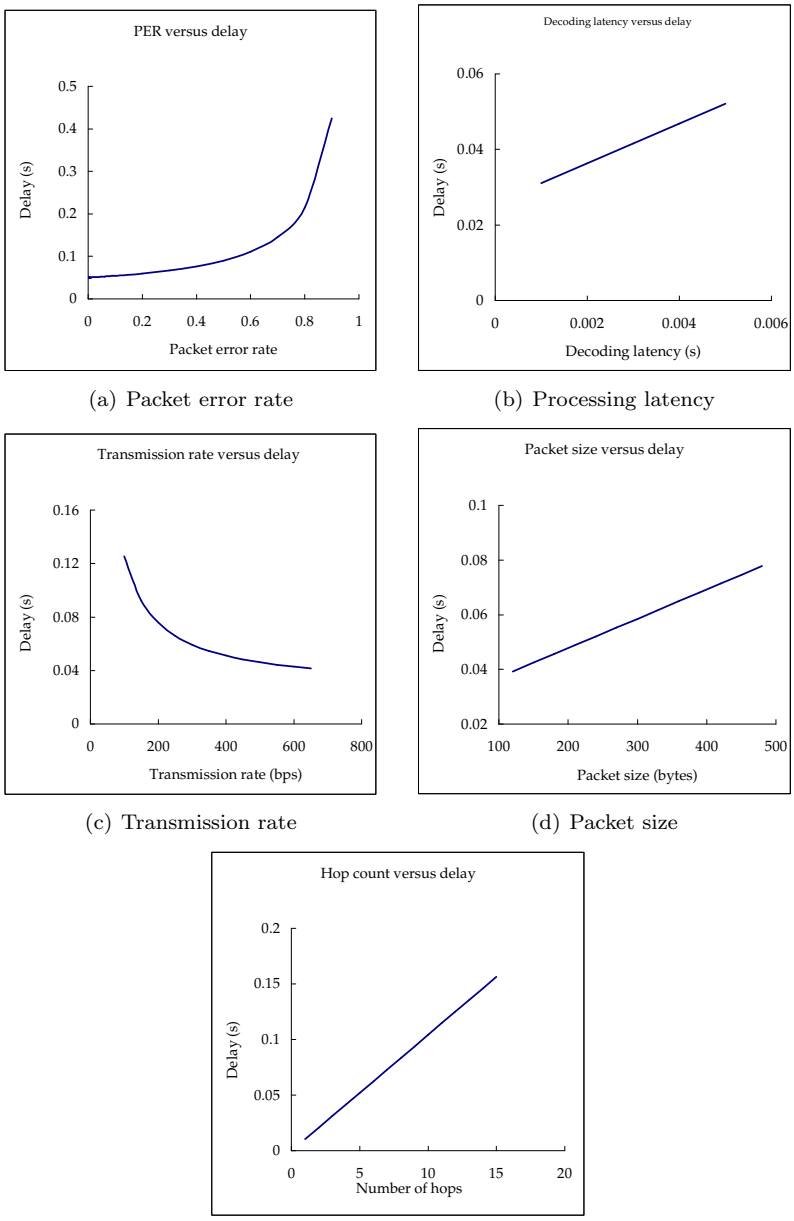
Since we want to apply equation 5.1 to this case, we need to modify Γ_T, Γ_R, and ρ, to fit our needs. The error free transmission delay of cut-through switching is the sum of the first full packet transmission, and the subsequent $n-1$ relay operations. A relay operation in this context is the reception of the N_h header bits, after which the packet begins to be forwarded. Since there is only decoding at the destination, we can define the initial transmission delay as $\Gamma_T = \frac{1}{R_c}(N_p + (n-1)N_h) + \tau_c$.

If an error is detected, a NAK of size N_h needs to be sent back to the source, after which an error correction packet of size N_r must be retransmitted to the destination (by cut-through), which will try to decode the packet during τ_c seconds. Hence, we define the retransmission delay component $\Gamma_R = \frac{nN_h}{R_c} + \frac{1}{R_c}(N_r + (n-1)N_h) + \tau_c$, or, slightly simplified, $\Gamma_R = \frac{1}{R_c}(N_r + (2n-1)N_h) + \tau_c$. We write the packet error rate over n hops as $\hat{\rho} = 1 - (1-\rho)^n$, and we can now rewrite equation 5.1 as follows.

$$\overline{D}_{ct} = \frac{\Gamma_T + \Gamma_R \sum_{i=1}^{m} \hat{\rho}^i}{1 - \hat{\rho}^{m+1}} \tag{5.3}$$

Observations

We want to know how the cut-through scheme performs in comparison with decoded relaying. To show this, we perform the same calculations as in the previous section,



(a) Packet error rate

(b) Processing latency

(c) Transmission rate

(d) Packet size

(e) Number of hops

Figure 5.1: Average packet delay in decoded relaying (eq. 5.2) as a function of different parameters.

and normalize the output with that of the store-and-forward expression. The results are displayed in Figure 5.2.

From these results, it is clear that the packet error rate and hop count have significant impact on the normalized performance of virtual cut-through. Actually, the sharp drop in the hop count graph is not a coincidence. Even at substantial error rates, 2-hop cut-through displays better performance than store-and-forward for the configuration given in Table 5.1.

5.5 Trailing error control

The reason for the fast degradation of performance of virtual cut-through under high error rates is primarily due to the properties of the end-to-end error control. Since every transmission traverses the whole network before a possible error can be detected, the transmission error rate accumulates over each link.

One approach to address this behavior of cut-through switching is the trailing error control scheme. The aim is reduce the number of hops that retransmissions need to traverse, and to consequently increase the probability of successful error correction. To achieve this, we let every node simultaneously buffer each packet, while forwarding it. This is to enable error detection after the packet has been received. If the packet is found to be erroneous, it is simply discarded, but if it is error-free it is stored in the node. When the destination node detects an error, it transmits a retransmission request back towards the source, but the first packet along the path that possesses an error-free copy intercepts the NAK, and responds with a retransmission. The result is an error control scheme which reduces the retransmission error probability, and hence the loss rate, compared to end-to-end error control, and consequently improves the performance of cut-through switching.

Analytic expression

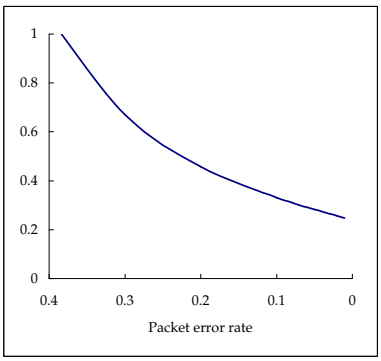
To derive the analytic expression of cut-through switching with trailing error control, we will first calculate the transmission delay and after that the loss rate. The transmission delay starts with the initial transmission, written as $\Gamma_T = \frac{1}{R_c}(N_p + (n-1)N_h) + \tau_c$. With probability ρ, there will be an error over the first link, resulting in a retransmission session $\Gamma_{Rn} \sum_{i=1}^{m} \hat{\rho}_n^{i-1}$, where $\Gamma_{Rn} = \frac{1}{R_c}(N_r + (2n-1)N_h) + \tau_c$ and $\hat{\rho}_n = 1 - (1-\rho)^n$ denote the single retransmission round-trip delay and the retransmission error probability, respectively. If the transmission over the first link was successful, there will be an error over the second link with probability $(1-\rho)\rho$, resulting in a retransmission session similar to the previous, but with parameters $\Gamma_{R(n-1)} = \frac{1}{R_c}(N_r + (2n-3)N_h) + \tau_c$ and $\hat{\rho}_{n-1} = 1 - (1-\rho)^{n-1}$. This is repeated for all links in the path, producing the expression:

$$\Gamma_T + \rho\Gamma_{Rn} \sum_{i=1}^{m} \hat{\rho}_n^{i-1} + (1-\rho)\rho\Gamma_{R(n-1)} \sum_{i=1}^{m} \hat{\rho}_{n-1}^{i-1} + (1-\rho)^2\rho\Gamma_{R(n-2)} \sum_{i=1}^{m} \hat{\rho}_{n-2}^{i-1} + \dots$$

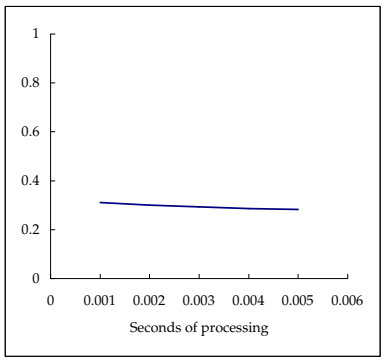
This expression can be formalized into the following equation.

$$D_{tr} = \Gamma_T + \rho \sum_{k=0}^{n-1} \left((1-\rho)^k \Gamma_{R(n-k)} \sum_{i=1}^{m} \hat{\rho}_{n-k}^{i-1} \right) \tag{5.4}$$

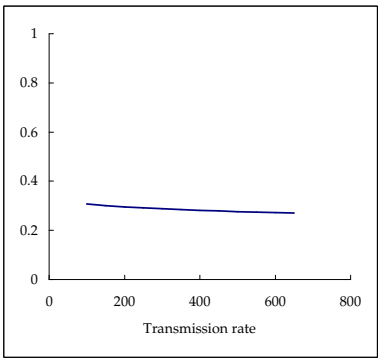
The loss rate can be derived in a similar fashion. The first term is that an error occurs during the original transmission, which is expressed as $1-(1-\rho)^n$. With probability ρ the error occurs over the first link, giving the loss rate $\hat{\rho}_n$. Next, with the probability of $(1-\rho)\rho$



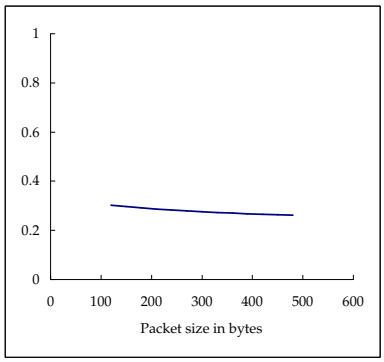
(a) Normalized C-T performance with a decreasing PER



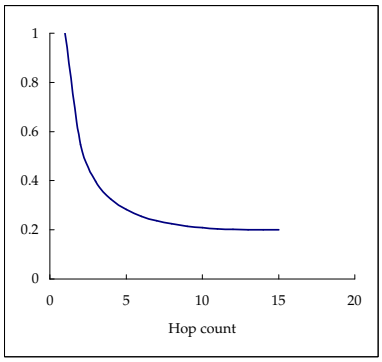
(b) Normalized C-T performance with an increasing processing time



(c) Normalized C-T performance with an increasing transmission rate



(d) Normalized C-T performance with an increasing packet size



(e) Normalized C-T performance under an increasing number of hops

Figure 5.2: Performance of virtual cut-through (C-T) normalized by corresponding performance of decoded relaying.

the loss rate is $\hat{\rho}_{n-1}$. The final expression will thus, in accordance with the transmission delay, be written as follows.

$$R_{loss} = \rho(1-(1-\rho)^n)\sum_{k=1}^{n}(1-\rho)^{k-1}\hat{\rho}_k^m \tag{5.5}$$

To keep our result consistent with the previous analyses, we need to include the loss rate in our equation. The final expression can thus be written as

$$\overline{D}_{te} = \frac{\Gamma_T + \rho\sum_{k=0}^{n-1}\left((1-\rho)^k\Gamma_{R(n-k)}\sum_{i=1}^{m}\hat{\rho}_{n-k}^{i-1}\right)}{1-\rho(1-(1-\rho)^n)\sum_{k=1}^{n}(1-\rho)^{k-1}\hat{\rho}_k^m} \tag{5.6}$$

Observations

When evaluating this scheme, we see in Figure 5.3 that for the default parameter settings shown in Table 5.1, trailing error control performs only slightly better than the regular end-to-end error control. As expected, however, the relative performance increases with higher error rates and an increasing number of hops, since these parameters directly affect the probability of a retransmission request, i.e. the end-to-end packet error rate.

5.6 Partial cut-through

The third variant of cut-through, partial cut-through, is a combination between decoded relaying and virtual cut-through. Given a multihop path of n hops, we apply virtual cut-through over every h hops. Thus, there will be $\lceil n/h \rceil - 1$ subsequent decoded relaying, with virtual cut-through in between. If h is not a factor of n, the last cut-through operation is performed over $n \mod h$ hops.

To express this analytically, we re-define Γ_T, Γ_R, and $\hat{\rho}$ as in section 5.4, writing $\Gamma_T(h) = \frac{1}{R_c}(N_p + (h-1)N_h) + \tau_c$, $\Gamma_R(h) = \frac{1}{R_c}(N_r + (2h-1)N_h) + \tau_c$, and $\hat{\rho}(h) = 1-(1-\rho)^h$, respectively. We then define $h \in \{1 \ldots \lceil n/2 \rceil, n\}$, and the expression becomes as follows.

$$\overline{D}_{pct} = \min_{h\in\{1\ldots\lceil n/2\rceil,n\}} \frac{\Gamma_T(h) + \Gamma_R(h)\sum_{i=1}^{m}\hat{\rho}(h)^i}{1-\hat{\rho}(h)^{m+1}} \cdot \frac{n}{h} \tag{5.7}$$

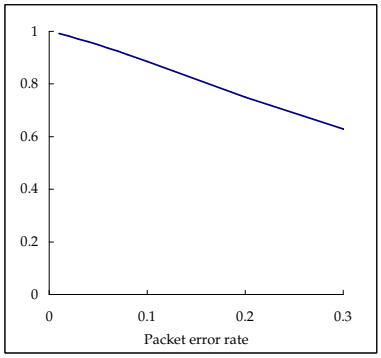
Observations

A principal observation, when evaluating this expression and comparing it with virtual cut-through, is that at low to moderate error rates, partial cut-through behaves identical to virtual cut-through. Only at substantial error rates ($> 20\%$) does it outperform this scheme. In figure 5.4, the performance of partial cut-through normalized with that of virtual cut-through at a 30% error rate is displayed.

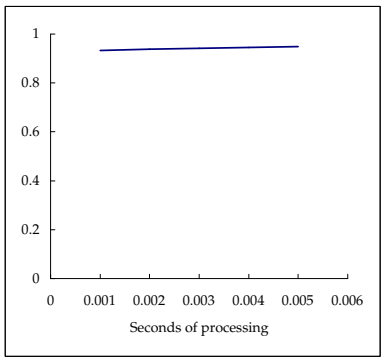
5.7 Discussion

In a multiuser environment, or with multiple contending services, the issue of queuing is present. In such a case statistical multiplexing may be favorable.

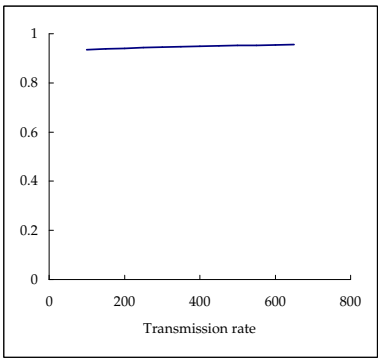
In the analytical comparison between these four schemes (decoded relaying, and three variants of cut-through switching) it is clear that the cut-through scheme outperforms decoded relaying at low to moderate error rates.



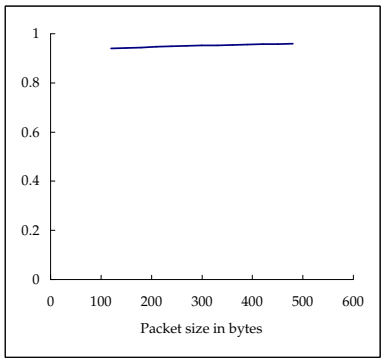
(a) Normalized T-E performance with a decreasing PER



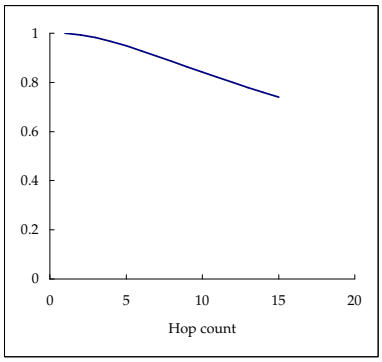
(b) Normalized T-E performance with an increasing processing time



(c) Normalized T-E performance with an increasing transmission rate



(d) Normalized T-E performance with an increasing packet size



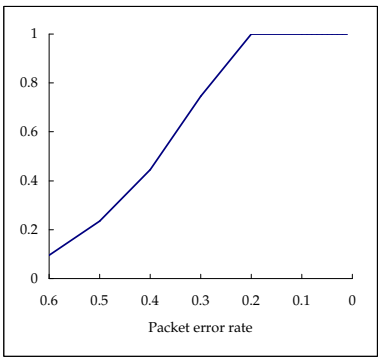
(e) Normalized T-E performance with an increasing number of hops

Figure 5.3: Performance of cut-through with trailing error-control (T-E), normalized by corresponding performance of virtual cut-through.

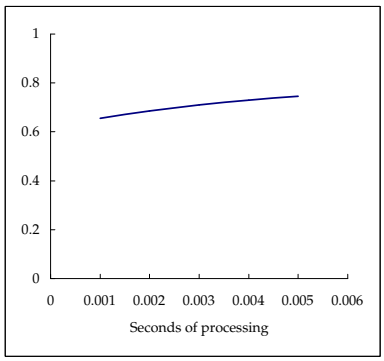
However, we need to know more than just the *average* delay, specifically packet drop rate and delay distribution. Thus we can see what portion of the packets would meet the maximum tolerable delay bound.

Drop rate

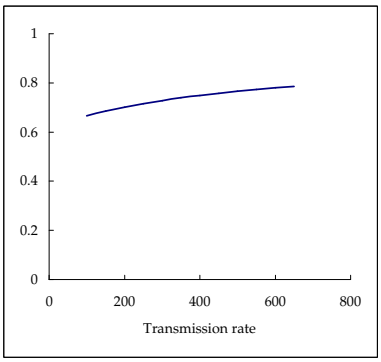
Figure 5.5 shows the drop rate of the per-hop, end-to-end, and trailing error control schemes studied above with the given default configuration.



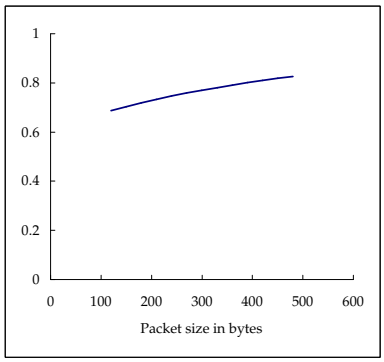
(a) Normalized PCT performance with a decreasing PER



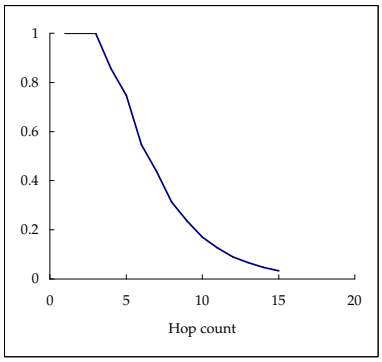
(b) Normalized PCT performance with an increasing processing time



(c) Normalized PCT performance with an increasing transmission rate



(d) Normalized PCT performance with an increasing packet size



(e) Normalized PCT performance with an increasing number of hops

Figure 5.4: Performance of partial cut-through (PCT), normalized by corresponding performance of virtual cut-through.

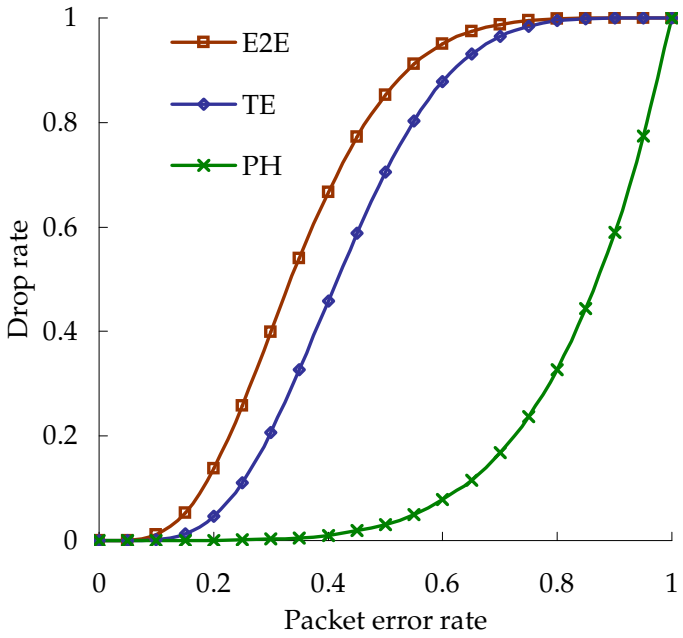


Figure 5.5: Drop rate of end-to-end error control (E2E), trailing error control (TE), and per-hop error control (PH) as a function of packet error rate.

Chapter 6

Simulation

In this chapter we will study the delay characteristics of our network model by simulation. Due to the limited time for these studies, simulation was limited to the decoded relaying forwarding scheme. We will discuss relevant metrics and observe how they perform in our simulated network.

6.1 Simulation environment

Traffic

We need to choose a real-time service to use for our simulation. Since a lot of attention has been placed on voice communication, we have chosen this traffic type. When modeling voice traffic we need to consider the CODEC, the packet contents, and the packet arrival model.

CODECs

For traffic generation, there are several different coding standards to consider, as presented in Table 6.1. They differ in data rate, algorithmic delay, perceived quality, and random-loss resilience. While a comprehensive discussion of the different CODEC characteristics is beyond the scope of this thesis, we will make a brief comparison between the G.711 and G.723.1 CODECs, which represent two opposite extremes in terms of the characteristics listed above. While G.711 tops the list in terms of data rate, it introduces a very low algorithmic delay, as opposed to G.723.1, which operates at a significantly lower rate, but has a corresponding significant algorithmic delay. As for quality, the data rate and voice quality of a CODEC are correlated, meaning that a higher data rate implies a higher quality. Finally, comparing tolerance to random losses, G.711 with Packet Loss Concealment (PLC) outperforms G.723.1 at the expense of a small additional algorithmic delay [35].

In our simulation we choose to model the G.711 CODEC for the following reasons; (a) very small packet sizes would limit FEC efficiency, (b) larger packets do not imply significantly longer transmission delays for the high transmission rate networks we are considering, (c) as packet loss rate will be non-negligible, the CODEC property of loss tolerance becomes important, and (d) the performance of a high data rate VoIP service can give an indication of the behavior of other high data rate real-time communication services such as video conferencing.

Coding standard	Data rate	Algorithmic delay	MOS
G.711	64 Kbps	3.75 ms (PLC)	4.3-4.4
G.729	8 Kbps	15 ms	4.0-4.2
G.728	16 Kbps	3-5 ms	4.0-4.2
G.723.1	5.3, 6.3 Kbps	37.5 ms	3.5-4.0

Table 6.1: Common coding standards (data from [35]). The Mean Opinion Score (MOS) is a subjective method of quality assessment with scores ranging from 1 to 5 with increasing perceived quality.

Packet constitution

RFC1890 [36] specifies that the default packetization period should be 20ms. For G.711, this results in a packet payload of 160 bytes, and a generation rate of 50 packets per second (pps). Each packet also has protocol headers for IP, UDP, and RTP, in total up to 40 bytes. However, by using compressed RTP (cRTP) [37] [38], these headers can be reduced to 2-4 bytes. At the link layer, we apply an IEEE 802.11 MAC header of 34 bytes, resulting in an uncoded packet size of 198 bytes.

We also need to add error correcting bits to the packet. We choose a low $\frac{3}{8}$ coding rate with a puncturing scheme resulting in an increased $\frac{3}{4}$ rate. In our HARQ scheme, this puncturing results in one original, high-rate coded, transmission of 264 bytes and 4 possible retransmissions of 66 bytes. To the resulting packet of 264 bytes, we add the physical layer header bits. Using the physical header size of the IEEE 802.11 protocol (24 bytes), our final packet size is 291 bytes. The corresponding retransmission packet size is 90 bytes.

Control traffic, i.e. positive and negative acknowledgements, is modeled simplistically as a single MAC + physical header packet 58 bytes in size.

Packet generation

Each packet source in our simulation generates traffic that arrives at one of the network nodes, excluding the access point. Packet generation is traditionally modeled as an on/off process with talkspurts and gaps exponentially distributed in length. In accordance with ITU-T P.59 [39], the mean spurt and the mean gap is set to 1.004s and 1.587s, respectively. During a spurt, traffic is generated at a constant rate of 50 pps, resulting in a 116.4Kbps activity bitrate. As the mean gap is 1.587s, the source activity is only 39%, resulting in an average bitrate of 45.1Kbps. If we want to include the probability of retransmissions, $\sum_{i=1}^{m} \rho^i$, we need to add an extra 1Kbps for a 10% packet error rate, derived from the expression $39\% \cdot 50\text{pps} \cdot 58\text{bytes} \cdot 8\text{bits} \cdot \sum_{i=1}^{4} 0.1^i = 1.005\text{Kbps}$, which gives us a probabilistic, average source bitrate of 46.1Kbps.

Simulation setup

Our simulation setup, as depicted in figure 6.1, consists of a number of traffic sources distributed in the network, providing the network load. One source, called the measurement source, is assigned to be our point of measurement, and the one-way packet delays and losses pertaining to this source are recorded for analysis. Here, the other sources are providing the network load and are called load sources. These later sources provide background traffic.

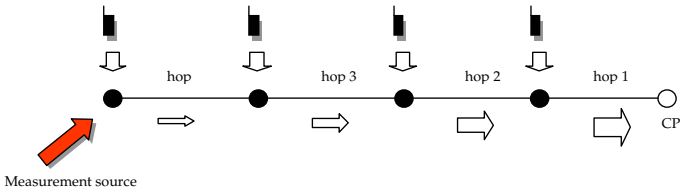


Figure 6.1: Simulation setup

Topology

We will consider a tandem topology where one of the end nodes is the access point to the wired network. The number of nodes in the chain determines the maximum number of hops, and by changing the size of the network, we change the hop count parameter in our simulation.

For a given network load there is a fixed number of load sources present. These load sources are evenly spread among the nodes, excluding the access point. When the number of load sources is not a simple multiple of the number of source nodes, sources will be assigned to the nodes in increasing order, beginning with the node closest to the access point. The measurement source is always assigned to the end source node (i.e. the node farthest from the access point).

HARQ

Our HARQ model is subject to a few simplifications. We assume that whenever a correction transmission arrives without error, it will succeed in correcting the original transmission, and when it arrives with an error, it will not provide sufficient correction. This is a simplification, since an erroneously received correction still has a chance to succeed when combined with the original packet and existing redundancy bits. There is also a probability that an error free correction packet would not succeed in correcting the original packet if the errors are too great.

Also, transmissions of correction packets are subject to the same error rate as regular packet transmissions. In our simulation, these packets are smaller and uncoded, therefore the error probability is likely to be much higher, i.e. resulting in a lower rate of correction packets being in error.

6.2 Metrics

Average delay and its standard deviation

This metric is crude and not very informative, since it says nothing about the variation and distribution of individual packet delays, but it can be useful together with its standard deviation as an indicator of the effect of parameters (such as number of hops and packet error rate) on the overall delay performance.

Delay distribution

The delay distribution tells us how many of the received packets arrive within a specific time. This metric can be used as an indication of the number of hops that can be supported while still providing an acceptable delay.

Loss and delay episodes

When considering packet loss, the distribution of loss is important. While single packet losses may be tolerable, loss of several packets will degrade any service as the loss bursts grow in length. We will use the term *loss episode* of length n to denote a sequence of n consecutively lost packets, with $n \in \{1 \dots N\}$ where N is the total number of packets sent in a simulation run. Consequently, a loss episode of length 1 denotes a single lost packet while a loss episode of length N means that all packets in the simulation run were lost.

Tolerance level

For a real-time service, packets arriving after the deadline will be discarded. Hence, we can regard a late packet (i.e. a packet arriving after the deadline) as a packet loss. As a result, we need to study the loss episodes originating from the combination of lost and late packets.

6.3 Simulation

In the previous chapter we examined analytically the average packet delay performance with respect to five parameters: packet error rate (PER), transmission rate, number of hops, packet size, and processing latency. For the simulation we have chosen to focus on the first three (PER, transmission rate, and number hops), and also to study the effect of competing traffic, represented by a network load. To examine the influence of these parameters, we have chosen the configuration values shown in Table 6.2.

Statistical data is collected for each possible configuration by having every source send 100,000 packets, corresponding to an average of 85 minutes of conversation. To measure the statistical significance, each configuration is run 5 times, and the standard deviation of the average packet delay is calculated. To reduce the effects of simultaneous startup, we only start measuring after the 50th packet has been sent by the measurement source, which corresponds to 1 second of a talkspurt. Since this period is exponentially distributed with a 1.004s mean, there is an even chance that the measurement source will still be in spurt mode, while half of the load sources have switched to gap mode (and possibly switched back again), or that the measurement node has switch to gap mode and back again (possibly several times).

Parameter	**Configuration values**
Transmission rate	1 or 2 Mbps
Network size	1-9 hops
Packet error rate	1%, 5%, 10%
Network load	25%, 75%

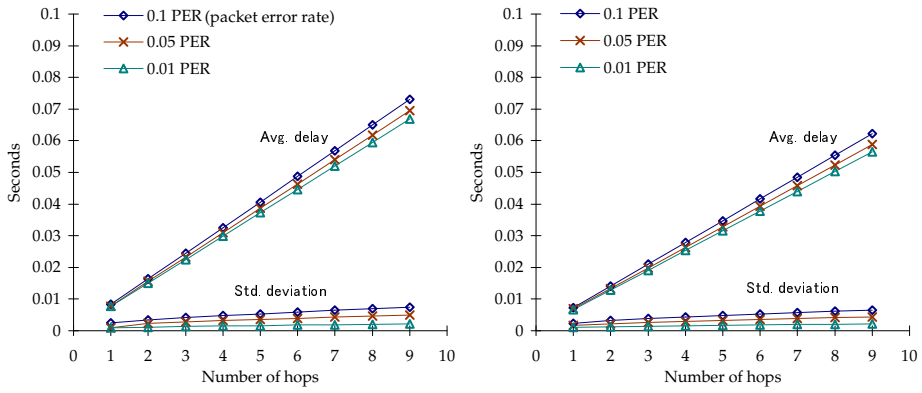
Table 6.2: Simulation parameters

Queuing

In our simulation we employ a first come first serve (FCFS) queuing policy, with equal priority among the sources.

6.4 Average delay and jitter

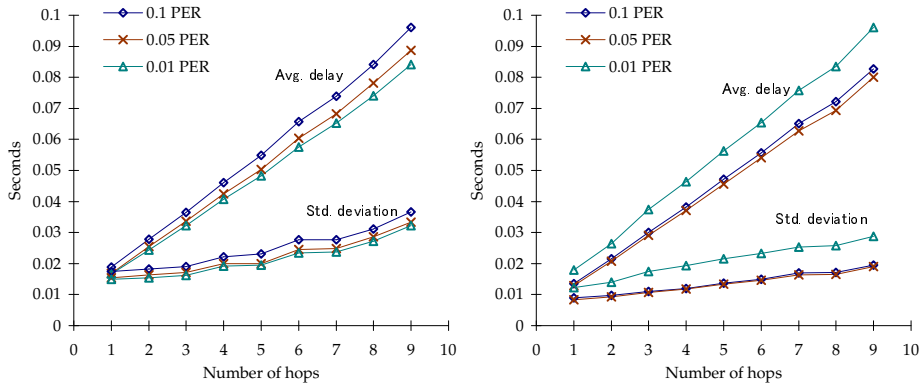
First, we look at the low network load case, depicted in Figure 6.2, we note that the average packet delay increases linearly with the number of hops (as expected). This linear relationship is stable, and corresponds to a low average packet queuing delay (< 1 ms). In addition to the low queuing delay we observe low delay variation, indicated by the standard deviation in the figure. Worth noting is also the relatively modest effect of packet error rate. The difference in average delay between the 1% and 10% packet error rate case ranges linearly from negligible at 1 hop to around 10ms at 9 hops. These properties, the linearity in average delay vs. number of hops and the modest effect of error rate, are similar for both values of transmission rate, but perhaps a little more accentuated in the 2Mbps case. The maximum difference in average delay between the two configurations is only 10ms at 9 hops.



(a) 1 Mbps link transmission rate (b) 2 Mbps link transmission rate

Figure 6.2: Average delay performance for 25% loaded networks, with different link transmission rates.

In the 75% load case, shown in figure 6.3, the differences between the two configurations are more significant. At 1Mbps, we see that the delay variation is significantly greater than in the 25% load case. Also, the linearity is somewhat distorted, indicated not only by the instability of the curve, but by the greater slope over the last two hops compared to the slope over the first two hops. Correspondingly, the average delay is not only greater than in the 25% load case, but *the increase with the number of hops* is also greater. In (b), the 1% packet error rate graph clearly displays a larger delay than both 5% and 10%, which are very similar. This behavior can be explained by the increased queuing delay as bursts of error-free packets contend for resources. As the error rate increases, more retransmissions are requested, smoothing out the arrival bursts, and consequently **decreasing** the queuing delay.



(a) 1 Mbps network (b) 2 Mbps network

Figure 6.3: Average delay performance for 75% loaded networks, with different transmission rates.

6.5 Delay distribution

Next, we study the delay distribution. If we consider the Global IP Sound (GIPS) enhancement of the G.711 CODEC, the random packet loss tolerance is claimed to reach 30% [40]. Packet loss is the sum of packet drops and late arrivals, and the number of dropped packets is the sum of unrecoverable errors and queue overflows. Here, we will look at the delay distribution at a 5% packet error rate, with a 50ms delay tolerance.

In Figures 6.4 and 6.5, we see the packet delay distribution in the lightly loaded network with low (1Mbps) and higher (2Mbps) link transmission rates, respectively. Given the 50ms delay tolerance, the low rate network can achieve nearly 100% performance with 5 hops, while the higher rate network achieves similar performance with 6 hops. In both cases a higher hop count degrades the performance significantly, but the high rate network performs acceptably at 7 hops **if** the packet drop rate is modest and the late arrivals are not too bursty.

Turning to the high load cases in Figures 6.6 and 6.7, we see that the high rate network may be able to perform acceptably with 4 hops, if the packet drop rate is modest and late arrivals not too bursty. For the low rate case, however, even a 2-hop network performs only conditionally, depending on drop rate. Worth recalling here is that at high loads, the pressure on the queues are substantial, resulting in frequent packet drops.

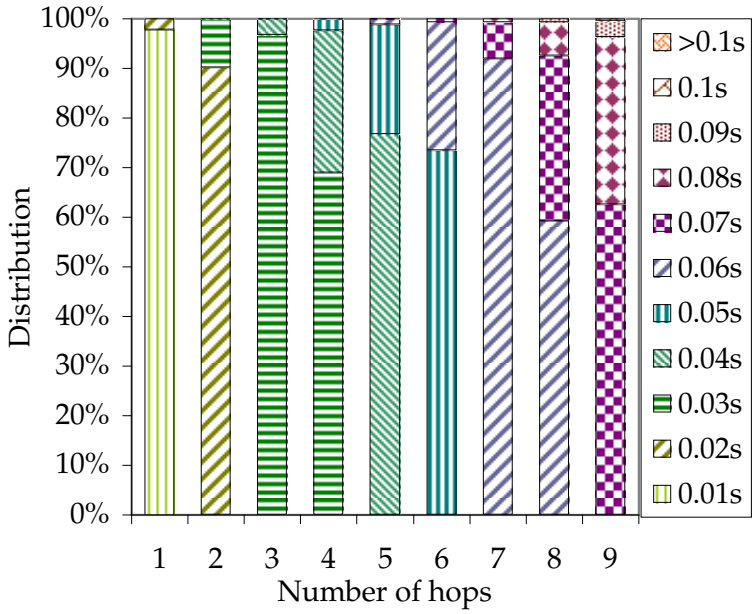


Figure 6.4: Delay distribution for the 5% error rate scenario with 25% network load and 1Mbps transmission rate.

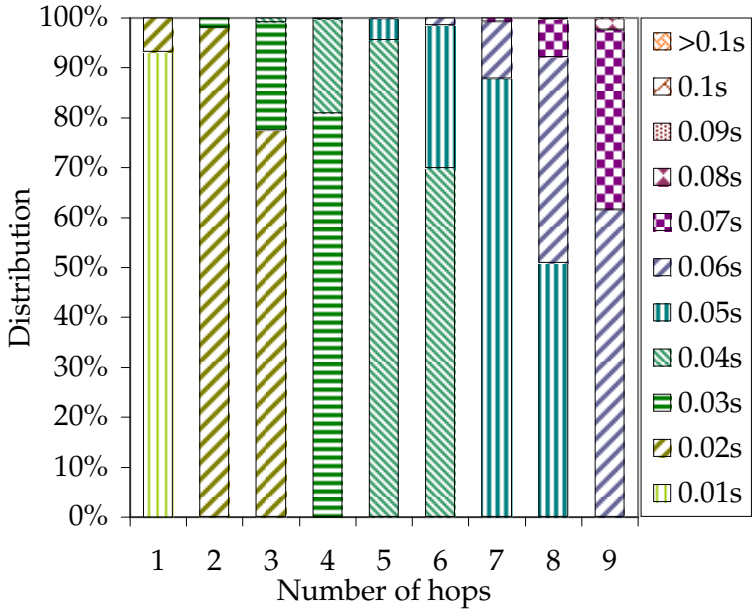


Figure 6.5: Delay distribution for the 5% error rate scenario with 25% network load and 2Mbps transmission rate.

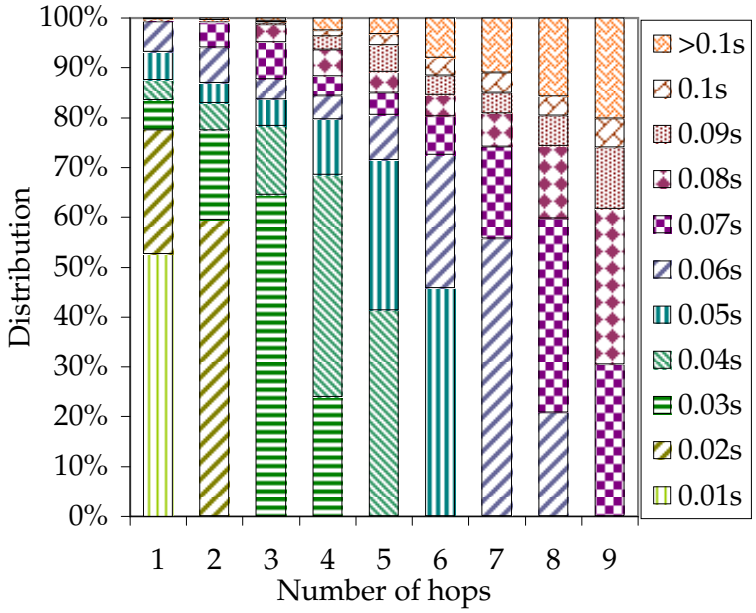


Figure 6.6: Delay distribution for the 5% error rate scenario with 75% network load and 1Mbps transmission rate.

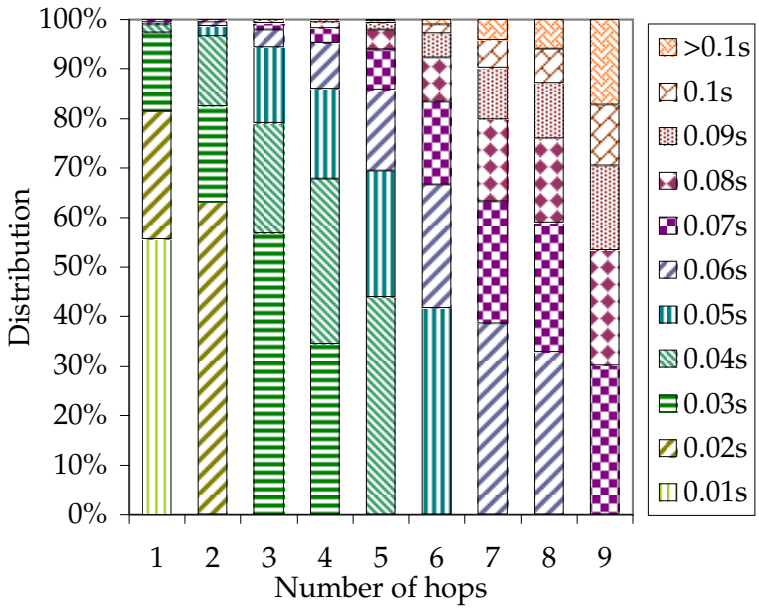


Figure 6.7: Delay distribution for the 5% error rate scenario with 75% network load and 2Mbps transmission rate.

6.6 Packet loss

The packet loss observed is a combination of packets dropped by the network and packets with delays above 50ms. However, we need to examine the burstyness of packet losses to get a clearer picture of how these lost packets may affect the service.

The previous section examined the delay distribution, and from those findings we are now interested in packet loss characteristics at 5 and 6 hops in the 25% loaded, low rate configuration, and at 6 and 7 hops in the high rate case, respectively. Figure 6.8 shows the distribution of bursts of varying length for each of these cases. In the low rate configuration, there's an average of 25 ($\pm$7) bursts of length 2 and 0.5 ($\pm$0.5) of length 3 at 5 hops, and none of greater length. An increase in hop count to 6 hops has a dramatic effect on the packet loss, as shown in the figure. Burst-lengths of up to 8 are present (2 ($\pm$1) occurrences), which corresponds to a 160 ms sample loss.

Turning to the 75% load configuration, we first look at the low rate case. As depicted in Figure 6.9a, even at a single hop we have a non-zero probability of burst-lengths of up to 100 (0.2 ($\pm$0.4)), which corresponds to 2 seconds of loss. Increasing the link rate, we see a significant decrease in burst-length (Figure 6.9b), with 0.2 ($\pm$0.4) probability of 27 as the peak value. Adding one hop, we see a probabilistic peak burst-length at 58 in Figure 6.10, which is over 1.1 seconds of loss.

The above results suggest that a network load of equal priority traffic as high as 75% of the network capacity strains the networks ability to support quality VoIP services, even in a single-hop configuration. However, in the 25% load case, we can see that the number of usable hops is clearly bounded, and that the maximum number of hops is to some extent dependant on the loss tolerance, which in the case of our VoIP service is represented by the packet loss concealment (PLC) efficiency of the CODEC.

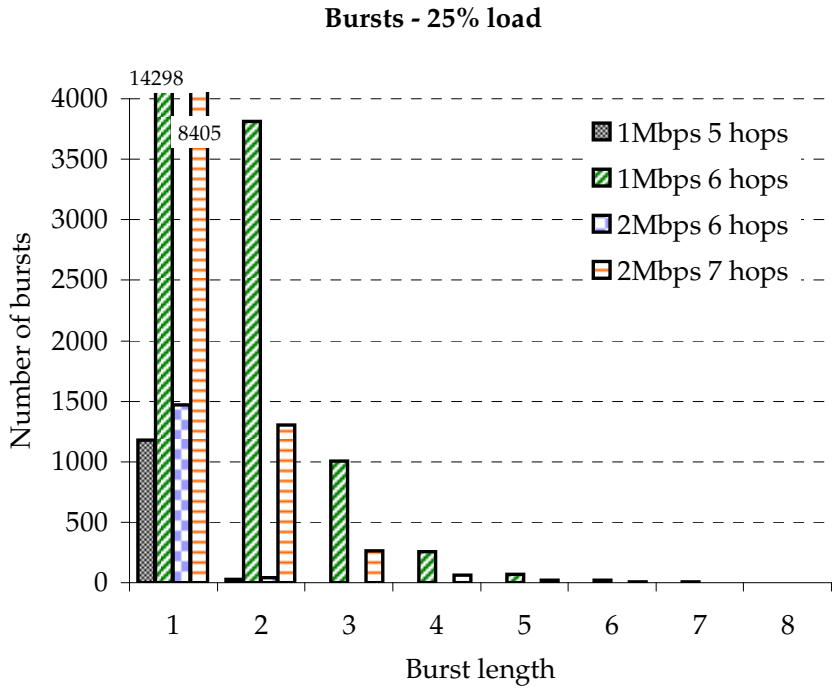
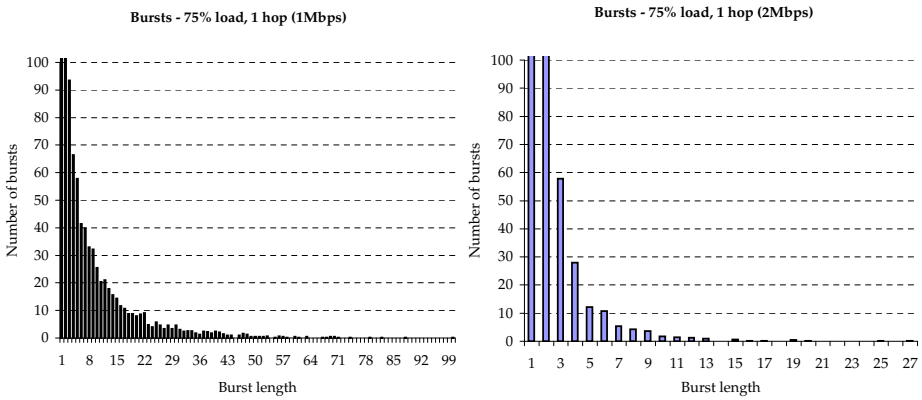


Figure 6.8: Burst distribution in the 25% load configuration.



(a) 1 Mbps capacity

(b) 2 Mbps capacity

Figure 6.9: Burst distribution in the 75% load configuration at 1 hop.

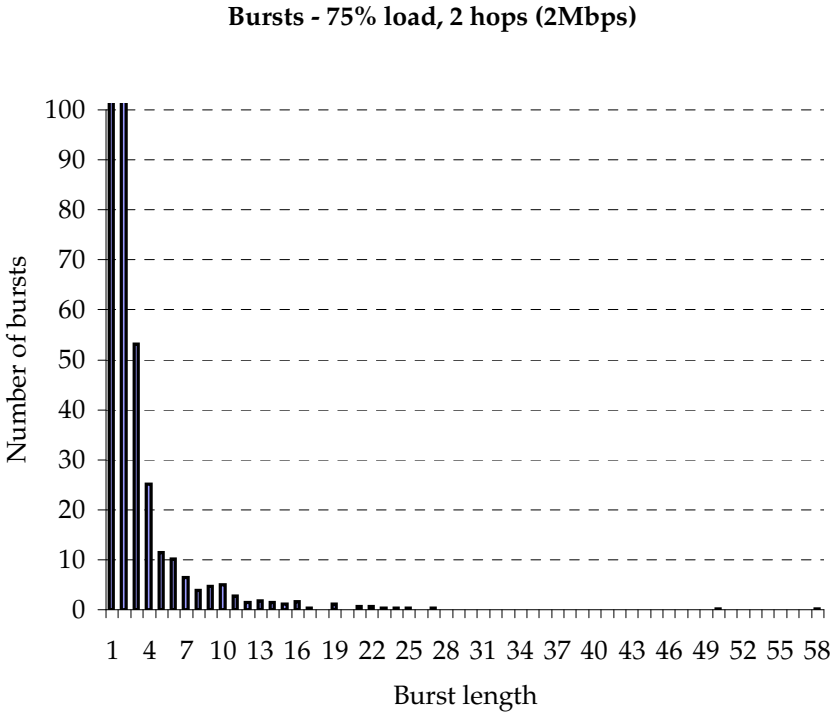


Figure 6.10: Burst distribution in the 75% load / 2Mbps configuration at 2 hops.

Chapter 7

Conclusions

In this thesis we have analyzed the link layer operation of a wireless multihop network, and its effect on packet delay. In particular, multihop forwarding has been analyzed. Four alternative forwarding approaches have been presented and analyzed, and one has been evaluated by simulation.

7.1 Conclusions

Analysis

We have seen that for decoded relaying, parameters such as decoding latency and packet size contribute to the average delay in direct proportion to the number of hops in a multihop chain. Packet error rate and channel transmission rate, however, display thresholds, beyond which the change in parameter values result in diminishing effects on delay.

As already known, virtual cut-through displays superior delay performance to decoded relaying under modest error rates. However, given a relatively high (but still realistic) transmission rate, cut-through still outperforms decoded relaying even when error rates are quite substantial (at rates above 10^{-1}).

One of the most important assumptions of this study is the stability of the error rate. Because of the low delay and limited jitter tolerance of real-time services, the bursty nature of the error rate in traditional wireless channel models give rise to relatively long and sudden periods of very low relaying performance, with increased delay and jitter as a result. However, in an environment without mobility, it is assumed that this burstiness is significantly damped. With the assumption of stability in the error rate, the cut-through technique proves to be viable, even for relatively high error rates. Furthermore, by employing the trailing error control scheme, the cut-through penalty even at high error rates can be reduced.

The main degrading property of virtual cut-through under high error rates is the loss rate. This parameter reaches (in many cases) unacceptable levels while, according to our analytical model, the average delay still performs well. This is partly due to the way the model incorporates packet loss. While trailing error control results in reduced loss rate, decoded relaying still outperforms by orders of magnitude all variants of cut-through with respect to this parameter. However, since we cannot see the characteristics of packet loss in this model, the question of what is an acceptable error rate remains.

Decoding

Preliminary results suggest that since higher decoding latency incurs significantly less penalty on virtual cut-through than on store-and-forward, while higher packet error rates degrading cut-through, the combination of relatively low coding rate and virtual cut-through would perhaps prove more delay efficient than high coding rate and store-and-forward.

Simulation

Through simulation, we could observe the burstiness of packet loss due to drops and delay for decoded relaying in a competing-traffic environment. While the 75% loaded network could not support our voice service even with a single hop, the 25% loaded network performed acceptably with up to 6 hops at 1Mbps bitrate, and 7 hops with the double this transmission rate. In our simulation environment, the packet error rate, ranging from 1% to 10%, did not affect performance in any significant way.

One conclusion to be drawn from this is that the number of hops in our network model is clearly bounded, and that this bound depends more on contention and link speed than on packet error rate.

7.2 Further study

As indicated before, there are several areas that need further exploration in order to validate the results of this thesis.

Simulations

Further simulation is needed to compare all of the forwarding schemes presented in chapter 5. However, the simulation of one scheme showed the importance of simulation to understand burst loss behavior vs. the analytic results which only describe statistically the average behavior.

Diversity

In this work, spatial diversity, or cooperative relaying, has been ignored. This is an central feature of wireless networks and how it can affect the delay and packet loss performance should be addressed in further studies.

Channel model

The behavior of the bit error rate in a fading channel with fixed nodes and fast TPC should be explored in order to verify or refute the assumption of stable packet error rate.

Medium access

The distribution of channel bandwidth between the inter-node links and the users is still an open issue. One approach is to use separate bands for user access and backhaul communication, as in [26] where the backhaul uses 802.11a, which operates in the 5GHz band, while users access the network via 802.11b/g, which use the 2.4GHz band.

Channel assignment

In this thesis, we have assumed non-interfering, pre-assigned frequency channels for each multihop link. This is idealized and not very realistic. How the frequencies should be distributed and assigned over the links so as to minimize interference is an important topic that needs more exploration.

Chapter 8

Appendix

Decoded Relaying

$$\Gamma_T = \frac{N_p}{R_c} + \tau_c \tag{8.1}$$

$$\Gamma_R = \frac{1}{R_c}(N_h + N_r) + \tau_c \tag{8.2}$$

$$D = \frac{\Gamma_T + \Gamma_R \sum_{i=1}^{m} \rho^i}{1 - \rho^{m+1}} \tag{8.3}$$

Virtual C-T

$$\Gamma_T = \frac{1}{R_c}(N_p + (n-1)N_h) + \tau_c \tag{8.4}$$

$$\Gamma_R = \frac{nN_h}{R_c} + \frac{1}{R_c}(N_r + (n-1)N_h) + \tau_c \tag{8.5}$$

$$\Gamma_R = \frac{1}{R_c}(N_r + (2n-1)N_h) + \tau_c \tag{8.6}$$

$$D = \frac{\Gamma_T + \Gamma_R \sum_{i=1}^{m} \hat{\rho}^i}{1 - \hat{\rho}^{m+1}} \tag{8.7}$$

Trailing Error

$$D = \sum_{k=1}^{n} (1-\rho)^{k-1} \left(\Gamma_{Tk} + (n-k+1)\Gamma_R \sum_{i=1}^{m} \rho^i \right) \tag{8.8}$$

$$D = \frac{1}{1 - n\rho^{m+1}} \sum_{k=1}^{n} (1-\rho)^{k-1} \left(\Gamma_{Tk} + (n-k+1)\Gamma_R \sum_{i=1}^{m} \rho^i \right) \tag{8.9}$$

$$D(1) = \frac{\Gamma_{T1} + \Gamma_R \sum_{i=1}^{m} \rho^i}{1 - \rho^{m+1}} \tag{8.10}$$

Bibliography

[1] E. Kudoh and F. Adachi, "Power and Frequency Efficient Wireless Multi-hop Virtual Cellular Concept," *IEICE Transactions on Communications*, vol. E88-B, pp. 1613–1621, Apr. 2005.

[2] C. E. Shannon, "A Mathematical Theory of Communication," *Bell System Technical Journal*, vol. 27, pp. 379–423,623–656, 1948.

[3] T. S. Rappaport, *Wireless Communications*. New Jersey: Prentice Hall, 1996.

[4] E. Kudoh and F. Adachi, "Power and frequency efficient virtual cellular network," in *Proc. of the 57th IEEE Vehicular Technology Conference (VTC)*, pp. 2485–2489, Apr. 2003.

[5] E. Kudoh and F. Adachi, "Impact of frequency- selective fading on distributed dynamic channel assignment in a DS-CDMA multi-hop virtual cellular network," in *Proc. of the 60th IEEE Vehicular Technology Conference (VTC)*, pp. 26–29, Sept. 2004.

[6] Y. Akaiwa, *Introduction to Digital Mobile Communication*. New York: Wiley-Interscience, 1997.

[7] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*. Boston: MIT Press, Sept. 2001.

[8] R. G. Cole and J. H. Rosenbluth, "Voice over IP Performance Monitoring," *Computer Communication Review*, vol. 31, pp. 9–24, Apr. 2001.

[9] S. Zander and G. Armitage, "Empirically Measuring the QoS Sensitivity of Interactive Online Game Players," in *Proceedings of the Australian Telecommunications Networks & Applications Conference*, (Sydney), ATNAC 2004, ATNAC, Dec. 2004.

[10] P. Herhold, W. Rave, and G. Fettweis, "Relaying in CDMA Networks: Pathloss Reduction and Transmit Power Savings," in *Proc. of the 57th IEEE Vehicular Technology Conference (VTC)*, pp. 2047 – 2051, IEEE, Apr. 2003.

[11] P. Lungaro and E. Wallin, "QoS, Coverage and Capacity in Adhoc/Cellular Networks," in *Proceedings of the Affordable Wireless Services and Infrastructure, 1st Annual Workshop*, June 2003.

[12] Y.-D. Lin and Y.-C. Hsu, "Multi-hop cellular: a new architecture for wireless communication," in *Proceedings of IEEE INFOCOM*, pp. 1273–1282, IEEE, Mar. 2000.

[13] H.-Y. Hsieh and R. Sivakumar, "On Using the Ad-hoc Network Model in Cellular Packet Data Networks," in *Proceedings of ACM MobiHoc*, pp. 36–47, ACM, June 2002.

[14] H. Luo, R. Ramjee, P. Sinha, L. Li, and S. Lu, "UCAN: A Unified Cellular and Ad-Hoc Network Architecture," in *Proceedings of ACM MOBICOM*, pp. 353–367, ACM, Sept. 2003.

[15] J. C. Park and S. K. Kasera, "Enhancing Cellular Multicast Performance Using Ad Hoc Networks," in *IEEE Wireless Communications and Networking Conference*, pp. 2175 – 2181, IEEE, Mar. 2005.

[16] R. Pabst, B. H. Walke, D. C. Schultz, P. Herhold, H. Yanikomeroglu, S. Mukherjee, H. Viswanathan, M. Lott, W. Zirwas, M. Dohler, H. Aghvami, D. D. Falconer, and G. P. Fettweis, "Relay-Based Deployment Concepts for Wireless and Mobile Broadband Radio," *IEEE Communications Magazine*, p. 0, 2004.

[17] H. Bolukbasi, H. Yanikomeroglu, D. D. Falconer, and S. Periyalwar, "On The Capacity of Cellular Fixed Relay Networks," in *Canadian Conference on Electrical and Computer Engineering*, pp. 2217 – 2220, IEEE, May 2004.

[18] A. N. Zadeh and B. Jabbari, "Performance analysis of multihop packet CDMA cellular networks," in *Proceedings of GlobeCom 2001*, pp. 2875–2879, 2001.

[19] H. Yanikomeroglu, "Fixed and Mobile Relaying Technologies for Cellular Networks," in *Proc. of the Second Workshop on Applications and Services in Wireless Networks (ASWN'02)*, pp. 75 – 81, July 2002.

[20] K. Johansson, J. Markendahl, and P. Zetterberg, "Relaying access points and related business models for low cost mobile systems," in *Proc. of the Austin Mobility Roundtable*, Mar. 2004.

[21] IEEE, *ANSI/IEEE Std 802.11, 1999 Edition*. New Jersey: IEEE, 1999.

[22] S. Xu and T. Saadawi, "Does the IEEE 802.11 MAC protocol work well in multihop wireless ad hoc networks?," *IEEE Communications Magazine*, vol. 39, pp. 130–137, June 2001.

[23] J. Robinson, K. Papagiannaki, C. Diot, X. Guo, and L. Krishnamurthy, "Experimenting with a Multi-Radio Mesh Networking Testbed," in *Proc. of the 1st workshop on Wireless Network Measurements (WiNMee)*, Apr. 2005.

[24] A. Raniwala and T. cker Chiueh, "Evaluation of A Wireless Enterprise Backbone Network Architecture," in *Proceedings of IEEE Hot Interconnects 12*, IEEE, Aug. 2004.

[25] A. Adya, P. Bahl, J. Padhye, A. Wolman, and L. Zhou, "A Multi-Radio Unification Protocol for IEEE 802.11 Wireless Networks," Tech. Rep., Microsoft Research, July 2003.

[26] Mesh Dynamics, "http://www.meshdynamics.com," Jan. 2005.

[27] D. Chase, "Code combining – a maximum-likelihood decoding approach for combining an arbitrary number of noisy packets," *IEEE Transactions on Communications*, vol. 33, pp. 385–393, May 1985.

[28] D. M. Mandelbaum, "An adaptive-feedback coding scheme using incremental redundancy," *IEEE Transactions on Information Theory*, pp. 388–389, May 1974.

[29] S. Lin and P. S. Yu, "A Hybrid ARQ Scheme with Parity Retransmission for Error Control of Satellite Channels," *IEEE Transactions on Communications*, vol. 30, pp. 1701–1719, July 1982.

[30] S. Sugawara, E. Kudoh, and F. Adachi, "A DS-CDMA Cellular System Using Band Division and Channel Segregation Distributed Channel Allocation," Tech. Rep., IEICE, RCS2004-52, May 2004.

[31] D. Bertsekas and R. Gallager, *Data Networks*. New Jersey: Prentice Hall, 1992.

[32] F. H. P. Fitzek, B. Rathke, M. Schlager, and A. Wolisz, "Quality of Service Support for Real-Time Multimedia Applications over Wireless Links using the Simultaneous MAC-Packet Transmission (SMPT) in a CDMA Environment," in *Proceedings of MoMuc*, IEEE, Oct. 1998.

[33] P. Kermani and L. Kleinrock, "Virtual Cut-Through: A New Computer Communication Switching Technique," *Computer Networks*, vol. 3, pp. 267–286, 1979.

[34] E. F. C. LaBerge and J. M. Morris, "Expressions for the Mean Transfer Delay of Generalized M-Stage Hybrid ARQ Protocols," *IEEE Transactions on Communications*, vol. 52, pp. 999–1009, June 2004.

[35] Angus Ma, "Voice over IP (VoIP)," Tech. Rep., Spirent Communications, http://www.spirentcom.com/documents/100.pdf, 2001.

[36] H. Schulzrinne, "RTP Profile for Audio and Video Conferences with Minimal Control (RFC 1890)." IETF, 1996.

[37] S. Casner and V. Jacobson, "Compressing IP/UDP/RTP Headers for Low-Speed Serial Links (RFC 2508)." IETF, 1999.

[38] T. Koren, S. Casner, J. Geevarghese, B. Thompson, and P. Ruddy, "Enhanced Compressed RTP (CRTP) for Links with High Delay, Packet Loss and Reordering (RFC 3545)." IETF, 2003.

[39] International Telecommunication Union, "ITU-T Recommendation P.59: Artificial conversational speech," Mar. 1993.

[40] Global IP sound, "GIPS Enhanced G.711," http://www.globalipsound.com/datasheets/EG711.pdf, 2005.

IMIT/LCN 2005-21

www.kth.se