Performance modeling and analysis of mobile Internet access via cellular networks

Master's thesis

Author: Taoying Yuan
Email: tyuan@few.vu.nl
Student ID: #1430610

University: Vrije Universiteit Amsterdam
Department of Computer Science

Supervisors: Prof. dr. R. D. van der Mei
Dr. S. Bhulai

vrije Universiteit amsterdam

Acknowledgements

I worked on this thesis between December 2004 and June 2005 at the Vrije Universiteit Amsterdam. I would not have been able to carry out the work without the following people, whom I am deeply grateful to.

First I would like to thank my supervisors: Professor Rob van der Mei and Dr. Sandjai Bhulai for all their valuable help during the last seven months. They have been a great source of inspiration for me and were always available to discuss new ideas or to answer my questions. They have helped me to get back on track several times.

I would like to give my deepest thanks to my boyfriend, Rik Farenhorst, for his enormous support and understanding throughout the last six months. Without his help, this work would not have been possible for me. I would also like to thank my parents for their support and encouragement during my work and their endless support during all my years in school.

Taoying Yuan
Amsterdam, June 2005

Executive Summary

Over the next few years, mobile Internet access services, (e.g. the 802.11 WLAN), providing Internet access "anywhere" and "any time" are likely to experience dramatic growth. A key success factor in the commercial success of these services is performance, for example, in terms of throughput. Hence, for service providers it is crucial to be able to predict the performance of their networks under various scenarios, including the number of subscribers and the locations of the Access Points. This advocates the need for the development, validation and analysis of quantitative models to describe and predict the performance experienced by the end user for any given set of system parameters.

In this thesis our research goal is to understand the impact of the mobility of the mobile terminal on performance of internet access via cellular networks**.** This is accomplished by the following four parts.

1. Derive and validate an approximation closed-form expression for the aggregated throughput.
2. Develop the model to predict the performance of multiple classes' mobile users in a single cell scenario. And then, derive and validate closed-form expressions for the average throughput and the total amount received amount of received data.
3. Extend our model to a "multiple-class mobile users in a multiple-cell"- scenario and derive the closed-form expressions of the performance.
4. Illustrate how to use those expressions to optimize the design of the cellular networks based on 802.11 protocols.

Table of contents

List of Figures

List of Tables

Chapter 1
Introduction

1.1 Introduction

This chapter describes the global contents and context of this Master's thesis. The aim of this thesis is to understand the impact of the mobility of the mobile terminal on performance of mobile internet service via WLAN networks**.** Models are developed that measure performance as experienced by the end users in this domain (in terms of throughput and total amount of received data), for a given set of design choices and realistic user-behavior scenarios.

The following paragraphs first explain the research motivation, after which the research goals and the relevant research questions are presented. This chapter concludes by discussing the structure of the remainder of this thesis.

1.2 Motivation for the research

Over recent years, there has been an increasing trend towards personal computers and workstations becoming "portable" and "mobile", and wireless technology now reaches or is capable of reaching virtually every location on the face of the earth. Using wireless network interfaces, mobile devices can be connected to the public telephone network in the same way as wired telecommunications, or to the Internet in the same way as desktop computers are connected, using the Ethernet, a token ring, or point-to-point links. Millions of people exchange information every day using cellular telephones, and other wireless communication devices. The market for wireless communications has enjoyed tremendous growth. With the tremendous success of wireless telephony and messaging services, it is hardly surprising that mobile Internet access services (e.g., via cellular networks such as GPRS, UMTS, WLAN) which provide Internet access "anywhere" and "any time" are likely to experience dramatic growth.

Nowadays, there are three main wireless access technologies: Bluetooth, WLAN 802.11 and 3G Wireless networks (e.g., GSM/GPRS, UMTS). Figure 1.1 shows the comparison between them [Cesana 03]. The coverage of a transmitting node in a Bluetooth network is narrow (typical maximum range 10m), the coverage of a base station in 3G Wireless networks is wide, (typical maximum range 1000m), and the coverage of an access point in a Wireless Local Area Network (e.g., WLAN 802.11) is medium, (typical maximum range 300m in outdoor scenarios). The typical maximum peak speed of Bluetooth is 1Mbps, the one of 3G is 115 Kbps, the one of 802.11b is 2 to 4 Mbps, and the one of 802.11a is 24 to 45 Mbps. Bluetooth is targeted on low costs, 10US$ in 2001 evolving towards 5 US$ in 2005. Since 3G mobile cellular systems work in licensed bands (e.g., 900, 1800 1900 MHz), it is usually expensive for the service providers to get the license. However, WLAN 802.11 has moderate costs, 35 to 40 US$, because it works in the non-licensed spectrum (2.4 and 5 GHz). Fast development of 802.11b WLANs has made 802.11b WLAN cards currently even cheaper than Bluetooth cards. Although 3G Wireless networks can provide high ubiquity and mobility to customers and mobile cellular technologies are evolving from GPRS to UMTS, WLAN hotspots or networks are still spreading rapidly. They are cheap to install and offer much higher data speeds than UMTS, although in a limited area.

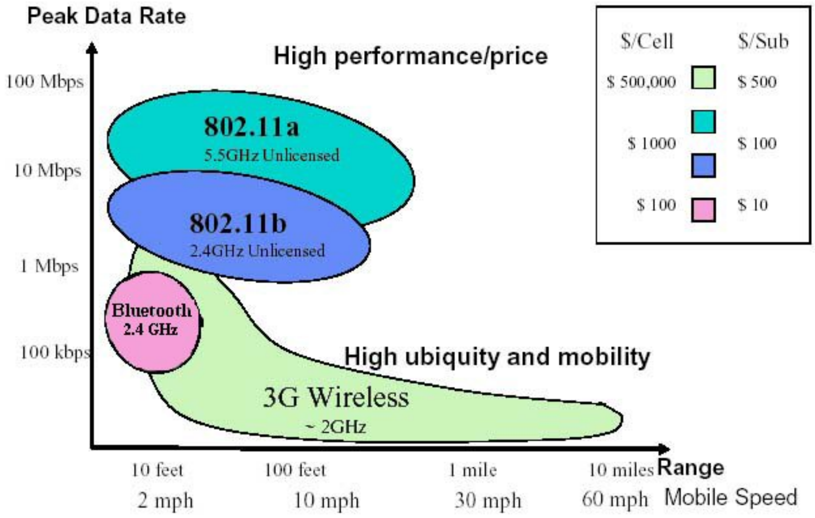


Figure 1.1 Current wireless network systems

WLAN technology was initially designed and implemented for use inside companies and buildings, but later WLAN networks have spread out to public places such as airports, hotels and cafes, which see a high passage of people. The hotspots are deployed as “network islands” but in many places we are seeing hotspots merging or expanding to cover parts of cities or larger regions. Wi-Fi services are even already available on international flights operated by Lufthansa and Japan Airlines. That is, passengers are able to check e-mail, send instant messages and surf the Web at 30,000 feet. [Peters 05] Figure 1.2 shows the number of WLAN total public users, and the total market penetration [Alonso 02]. It is predicted that there will be in total 20.5 millions WLAN users and the total market penetration will be 2.8 billions US$.

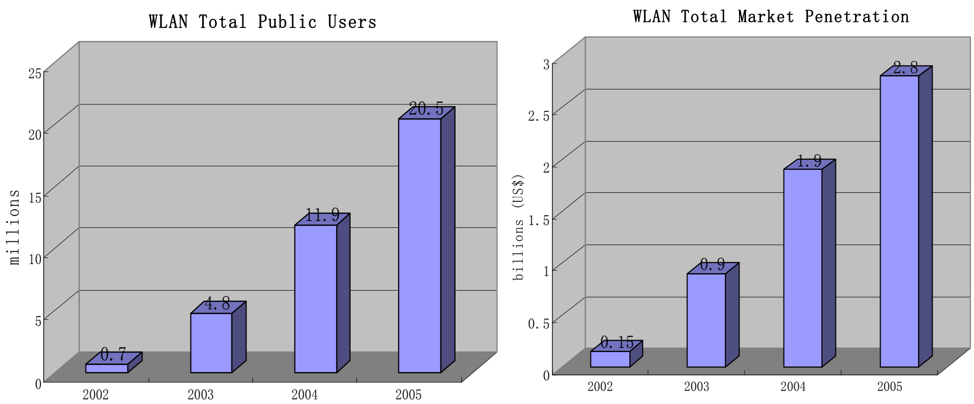


Figure 1.2 WLAN total public users and total market penetration

The wireless LAN industry is today one of the fastest growing segments of the communications industry. The first generation of wireless LAN specifications, referred to as 802.11, was developed for the 2.4 GHz microwave band and supported data speeds of 1-2Mbp/s. The introduction of the 802.11b specifications, known as "WiFi", sparked the launch of cheaper wireless LAN products, offering data speeds of up to 11Mbp/s (although, as with all wireless LAN standards, data speeds diminish as more users work concurrently in each WLAN cell) [Pareek 05]. Today, the next generations of 802.11 - 802.11a, 802.11g, and 802.11i. – are aimed at various stages of development and implementation, and at increasing the speed, security and quality of wireless LANs. This is shown in Table 1.1 Until now, all the standards from 802.11a to 802.11m have been released. According to the Industrial Economics and Knowledge Center (IEK) of the Industrial Technology Research Institute (ITRI), 802.11n products will become available in 2006 and start to take off in 2007. However, first-generation devices will start emerging this year. Suppliers show keen interest in the development of 802.11. For instance, Planex Communication Inc. has introduced a pre-802.11n-compatible WLAN card and a WLAN Access Point (AP) in cooperation with Airgo of the United States. Gemtek Technology Co. Ltd plans to increase its R&D efforts on 802.11n and WiMAX products during 2005. CyberTan Technology Inc. will develop WLAN models that adhere to both 802.11n and 802.11i standards. D-Link Corp. has likewise revealed plans to develop and launch 802.11n products with enhanced user-friendly features [Tech 05]. Furthermore, several smaller cities in US, such as Chaska, Minn., have deployed citywide Wi-Fi. Large cities such as Philadelphia and San Francisco see wireless broadband technology as a low-cost solution to providing broadband access to low-income residents. Philadelphia plans to have its citywide Wi-Fi network up and running by summer 2006 [Reardon 05]. In Adelaide (Australia) the company Agile communications is also offering WiFi throughout the city for free [CityLan].

Protocol version	Main Function	Protocol version	Main Function
802.11a	5GHz OFDM PHY	802.11b	2.4GHz CCK PHY
802.11c	802.11 bridging	802.11d	International roaming
802.11e	QoS/efficiency enhancements	802.11f	Inter AP protocol
802.11g	2.4GHz OFDM PHY	802.11h	5GHz regulatory extensions
802.11i	Security enhancements	802.11j	Japan 5GHz band extensions
802.11k	Radio resource measurement	802.11l	Skipped (typographically unsound)
802.11m	Maintenance	802.11n	High throughput PHY

Table 1.I WLAN standards versions

Whereas wide area cellular networks are traditionally speech-oriented and WLANs are typically installed as (extensions to wireline) office data networks, the evolution of services and the desire for seamless roaming underlies the significant current interest in the integration of both network types. Swisscom [Swisscom 04] started to combine GPRS, UMTS and WLAN technologies, which can offer users a nationwide broadband network. In addition, the mobile terminal company DoCoMo [NTT 04] started to produce dual-mode 3G and wireless LAN phones. This is very useful for business users wishing to work while on the move (e.g., to access e-mails, their company intranet or the Internet) as well as for residential users with a need for bandwidth-intensive applications such as the transfer of audio and video files.

Moreover, a broadband wireless access technology “WiMAX” (Worldwide Interoperability for Microwave Access) is also developing, which provides high-throughput broadband connections over long distances. It is also called "802.16", which seems to be a promising extension of WiFi, but with a much broader coverage than WiFi (radius up to 50 km in rural areas), with speeds up to 70 Mbps [McCullagh 05].

Whether it is 3G, WLAN or WiMax, the customer is interested in services, not access technology. A key success factor in the commercial deployment of these services is performance, for example, in terms of throughput and total amount of retrieved data. For instance, GPRS can provide broad coverage and constant quality of service but the achieved bit rates are relatively low. In WLAN on the other hand, the wireless channel has a higher bit rate, but is shared among all users. This results in a high bit rate, if only a few users are served. Hence, for service providers it is crucial to be able to predict the performance of their cellular networks under various scenarios, such as the number of subscribers, the locations of the base stations, and the mobility pattern of the users. This advocates the need for the development, validation and analysis of quantitative models to describe and predict the performance experienced by the end user, for any given set of system parameters. Examples of these parameters are: the locations of the base stations, the number of subscribers, the durations of sessions, the download volume per session, the average load for each of the cells, etc.

The IEEE 802.11 standard is the de facto standard for wireless local area networks (WLANs). It has been deeply studied in the last few years, and a multiplicity of papers has appeared in the literature, covering almost all aspects of the standard. A common problem in mobile cellular networks is that the amount of transmission capacity is limited and shared among the different users within the same cell. However, up to the knowledge of the authors, no satisfactory analytical model of the interaction between the user mobility and the network performance has been presented so far.

1.3 Research goal and approach

The goal of this project is *to understand the impact of the mobility of the mobile terminal on performance of internet access via cellular networks*.

To realize this goal, the following research questions are addressed. The key research questions can be stated as follows:

How does the mobility of the user impact the performance of his mobile internet access?

How can we use those validated expressions to do the optimizations?

Although simulations play an essential role in comparing competing design alternatives, simulations of wireless network systems, especially wireless links, are sometimes unreliable. This may lead to incorrect design choices. Generally, higher accuracy in modeling requires more complexity of computation. In this case, it is desirable to have models with low computation complexity, while maintaining the desired accuracy. If we can develop and study mathematical models, it is not only fast but it can also give us some insight of the relationship between the factors in the network. Another advantage is that we can do "what-if" analyses, i.e., we can predict the performance under any given scenario. This can never be done easily using simulations.

To answer those research questions, quantitative models are developed that predict the performance as experienced by the end users in this domain (in terms of throughput and total amount of retrieved data), for a given set of design choices and realistic user-behavior scenarios. Simulations are done to validate those expressions of performance obtained from the models. After that, system optimization can be done by using those validated approximation expressions for performance as a function of system parameters.

1.5 Structure of this thesis

The remainder of this thesis is organized in the following way:

The next chapter starts with starts with an introduction to the wireless networks. After this, introductions of the IEEE 802.11 protocol, the TCP protocol and the Network simulator are given. After these introductions, the related work in this area complements this chapter.

Chapter 3 first derives the performance expression for the single mobile station model, which has been validated by the simulations. Based on the results of the previous step, the multiple stations in a single-class scenario are analyzed. The expressions of the average throughput and the total amount of received data of any number of stations are derived, which also was thoroughly validated by the simulations.

Chapter 4 focuses on the multi-class users in the single cell scenario, that is, the users drive through one AP cell with different average velocities. The approximating expressions of the performance are derived. Simulations are run for validating the models. At the end of this chapter, the practical optimization usage of our model is analyzed.

Chapter 5 focuses mainly on investigated the performance of multiple-class users across multiple AP cells. We derived simple closed-form expressions for the average throughput and the total amount of received data for any class of users. After that, we provided two examples on how our model can be used to analyze practical problems.

Chapter 6 presents the conclusions. First, a summary of the developed models are given. Other topics in this chapter include a discussion on what this thesis contributes to the research community. The last paragraph focuses on future work.

Chapter 2
Background

2.1 Introduction

This chapter starts with an introduction to the wireless networks. Then, introductions of the IEEE 802.11 protocols are given in Section 2.3. The related details of the TCP protocol are present in Section 2.4, and the introductions of Network simulator are given in Section 2.5. In Section 2.6 related work in this area complements this chapter.

2.2 Wireless networks

"Wireless" and "*network"* have become two commonly used words today. Many companies either have products or present future products that are to be used in wireless network environments. The concept of wireless networks incorporates several different networking technologies, communication ranges and transmission bandwidths. They range from local coverage networks (as IEEE 802.11) to large wide area coverage networks, such as the third generation mobile telephony systems (e.g., GPRS, UMTS).

The General Packet Radio Service (GPRS) [ETSI 01] is part of the evolution path towards third generation (3G) mobile systems. It is designed to provide packet-based data service in a cellular system based on GSM. It is an overlay to the circuit switched GSM-networks. GPRS increases the possible bandwidth for data transmission by introducing packet switching. Inherently, packet-switching enables a more flexible and efficient utilization of the radio resources.

In this thesis we concentrate on close range networks, which are often called Wireless Local Area Networks (WLANs). Recently, hardware prices have dropped drastically for infrastructure equipment, and as a result of this, WLANs are deployed almost everywhere. The most common standard for these networks today is the IEEE 802.11 standard [IEEE 99]. There exist other standards such as HiperLan/2 [HipGF], and HomeRF [HRF], but they are not so widely used. WLAN usually operates on the license free ISM (Industry, Science and Medical) frequency band at 2.4GHz. And the extension 802.11a is moving towards 5 GHz which enables higher transfer rates but decreases the coverage area. The transmission bandwidth ranges from 1Mbit/s to approximately 50Mbit/s and the possible communication distance ranges from 30 to 1500m.

2.3 The IEEE 802.11 standard

Since this thesis focuses on the TCP performance of the IEEE 802.11 protocol, we only give an explanation of the general information about 802.11 standards that are related to our research. Further details can be found in [IEEE 99] and [802.11 b].

2.3.1. The 802.11 protocol architecture

The 802.11 standard was first standardized by IEEE in 1997 and revised in 1999. It defines two layers, shown in Figure 2.1. The first layer is the Physical layer (PHY), which specifies the modulation scheme used and signaling characteristics for the transmission through radio frequencies. The second layer is the medium access control (MAC) layer. This layer determines how the medium is used. Mobility of wireless stations may be the most important feature of a wireless LAN. A WLAN would not serve much purpose if stations were not able to move freely from location to location either within a specific WLAN or between different WLAN ‘segments’. For compatibility purposes, the 802.11 MAC must appear to the upper layers of the network as a ‘standard’ 802 LAN. The 802.11 MAC layer is forced to handle station mobility in a fashion that is transparent to the upper layers of the 802 LAN stack. This forces functionality into the 802.11 MAC layer that is typically handled by upper layers.

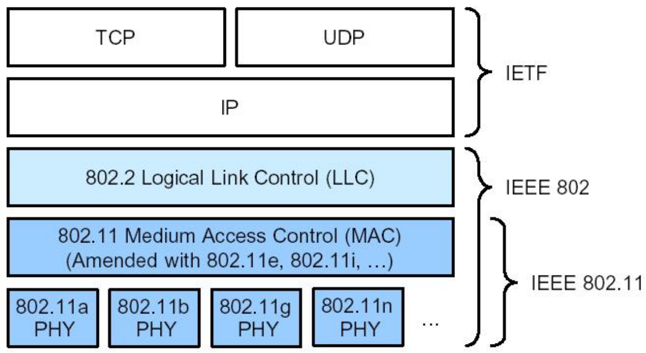


Figure 2.1 The 802.11 protocol model

The 802.11 architecture is comprised of several components and services that interact to provide station mobility transparent to the higher layers of the network stack, shown in Figure 2.2. An 802.11 network, in general, consists of Basic Service Sets (BSSs) that are interconnected with a Distribution System (DS). Each BSS consists of mobile nodes (referred to as *stations*, Wireless LAN Station (STA)). The station is the most basic component of the wireless network. A station is any device that contains the functionality of the 802.11 protocol, e.g., a laptop PC, handheld device, or an Access Point. Stations in a BSS gain access to the DS and to stations in “remote” BSSs through an *Access Point* (AP). The access points communicate amongst themselves to forward traffic from one BSS to another to facilitate movement of stations between BSSs. The access point performs this communication through the distribution system. The distribution system is the backbone of the wireless LAN and may consist of either a wired LAN or a wireless network. Before a station can access the wireless medium it has to be associated with an AP. A station can be associated with only one AP at any given time. A network of interconnected BSSs, as in Figure 2.2, in which mobiles can roam without loss in connectivity, is frequently referred to as an Extended Service Set (ESS). The IEEE 802.11 standard also specifies an additional “ad-hoc” architecture for a WLAN. Ad-hoc WLANs are characterized by lack of an AP, no functionality to support mobility, and support for data transfer *only* between stations that belong to the same WLAN. Therefore, an ad hoc WLAN is simply an independent BSS where a station communicates directly with one or more other stations. In our studies, we focus on the ESS as in Figure 2.2.

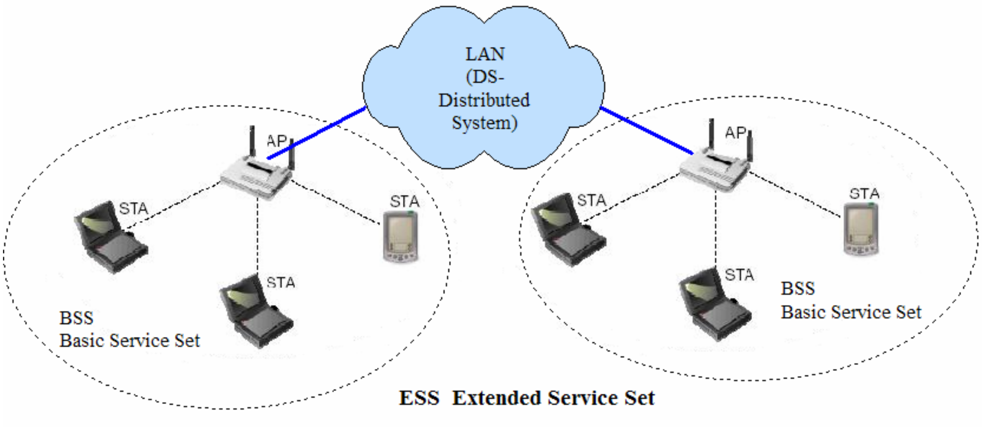


Figure 2.2 Components of an 802.11 WLAN system

2.3.2. The 802.11 MAC layer

The 802.11 MAC layer provides functionality to allow reliable data delivery for the upper layers over the wireless PHY media. The data delivery itself is based on an asynchronous, best-effort, connectionless delivery of MAC layer data. The standard provides two modes of operation. The basic access method in the 802.11 MAC protocol is the *Distributed Coordination Function* (DCF) which is best described as the *Carrier Sense Multiple Access with Collision Avoidance* (CSMA/CA) protocol [IEEE 99]. In addition to the DCF the 802.11 also incorporates an alternative optional access method known as the *Point Coordination Function* (PCF) – an access method that is similar to "polling" and uses a point coordinator (the AP) to determine which station has the right to transmit. Ad-hoc WLANs support only the DCF since they lack an AP. Most of the WLAN cards actually available on the market do not implement PCF for complexity reasons. Since the PCF is not the primary concern of my study, its operation is not covered here. DCF offers two access methods; Basic Access (BA) and ready to send/clear to send (RTS/CTS).

The basic access method

When using the DCF basic access mode, a station, before initiating a transmission, senses the state of the channel to determine if another station is transmitting. If the medium is determined to be idle for an interval that exceeds the *Distributed InterFrame Spacing* (DIFS), the station proceeds with its transmission. If the medium is busy, the station defers until after a DIFS is detected and then generates a random backoff period for an additional deferral time before transmitting. This minimizes collisions during contention between multiple stations. The backoff period is used to initialize the *backoff timer.* The backoff timer is decremented only when the medium is idle; it is frozen when the medium is busy. After a busy period the decrementing of the backoff timer resumes only after the medium has been free longer than DIFS. A station initiates a transmission when the backoff timer reaches zero. The length of the backoff interval, expressed in slots, is uniformly chosen in the set {0,1, …, CW-1}, where CW denotes the actual contention window size. At the beginning, the contention window takes an initial value of CWmin for each frame queued for transmission. To reduce the probability of collisions, the contention window of the stations involved in the collision is doubled and another transmission attempt is made. The contention window cannot grow indefinitely, but may reach a maximum value of 2^{γ}CWmin = CWmax, and then the contention window will remain at CWmax for the remaining retries;

moreover, if a packet incurs m collisions, where m⩾γ, it is dropped. The set of CW values are 7 (CWmin), 15, 31, 63, 127, 255 (CWmax) [IEEE 99].

Since no channel load sensing mechanism is provided, i.e., a transmitter cannot determine if the data frame was successfully received by listening to its own transmission as in wired LANs, an explicit acknowledgement is necessary to inform the station of the success/failure of its transmission. To do that, *Immediate positive acknowledgements* are employed to determine the successful reception of each data frame. This is accomplished by the receiver initiating the transmission of an acknowledgement frame after a time interval *Short InterFrame Spacing* (SIFS), which is less than DIFS, immediately following the reception of the data frame. In case an acknowledgement is not received the data frame is presumed lost and a retransmission is scheduled (by the transmitter). This access method referred to as the *Basic Access mechanism* is summarized in Figure 2.3 [Litjens 03_1].

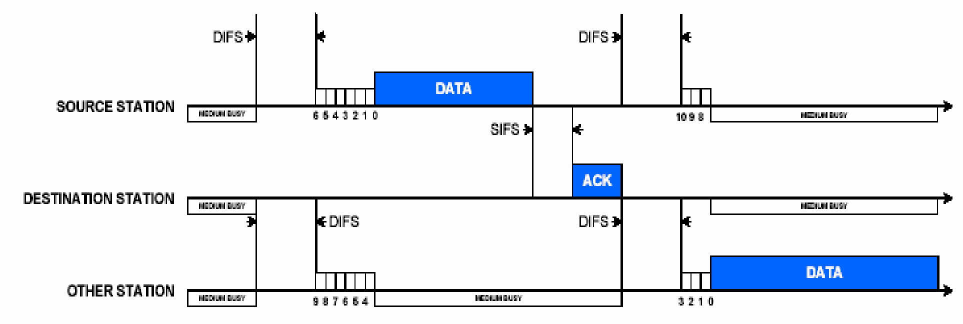


Figure 2.3 Basic access mechanism

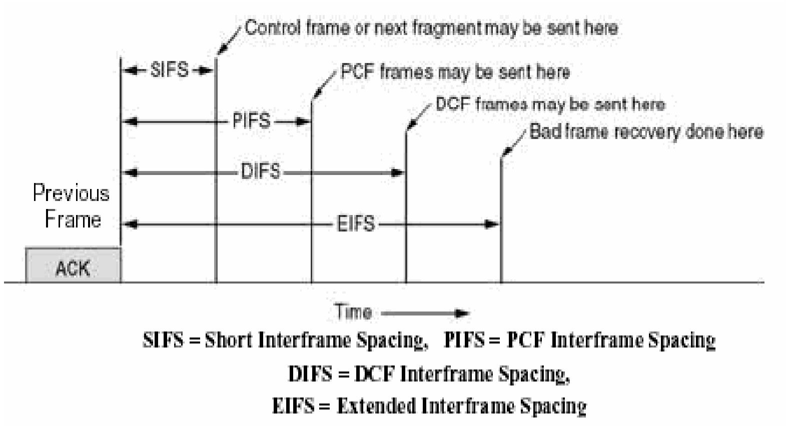


Figure 2.4 InterFrame spacing in 802.11

The standard encompasses also an *Extended Interframe Spacing* (EIFS), which is used whenever the PHY indicates to the MAC that a frame reception was not successfully completed. EIFS is provided to let the transmitting station retransmit the packet without interference from the receiving station. However, under the assumption of ideal channel conditions, EIFS does not have a great impact on network performance. This is due to the fact that EIFS acts at the receiver side. As an example, consider a pair of nodes exchanging data. In such a situation, collisions may clearly occur. However, since a transmitting node cannot receive data, nodes will become aware of the failure of a transmission attempt only by means of the ACK timeout expiration. Thus, EIFS will never be used. Hence, even if EIFS may actually be used when there are more than two nodes contending for the channel, we will, in the following, neglect its impact on network performance. The different Mac layer interframe space time is summarized in Figure 2.4 [IEEE 99].

RTS/CTS access method

The DCF also provides an alternative way of transmitting data frames that involve transmission of special short *Request To Send* (RTS) and *Clear To Send* (CTS) frames prior to the transmission of the actual data frame. This RTS/CTS mechanism is used to avoid the well-known hidden terminal problem of CSMA-based MAC protocols. In the basic access method, a data frame could be corrupted by transmissions due to stations that are hidden from the transmitter any time during the transmission of the data frame. The above leads to an increased probability of collisions, and the transmission time wasted as a result of each collision. This wasted time is significant since the transmitter is made aware of a collision only after it times out waiting for the corresponding acknowledgement. The RTS/CTS exchange attempts to perform a fast collision detection and a transmission path check. A successful exchange of RTS and CTS frames attempts to reserve the channel for the time duration needed to transfer the data frame under consideration. The rules for the transmission of an RTS frame are the same as those for a data frame under basic access, i.e., the transmitter sends an RTS frame after the channel has been idle for a time interval exceeding DIFS. On receiving an RTS frame the receiver responds with a CTS frame (the CTS frame acknowledges the successful reception of an RTS frame), which can be transmitted after the channel has been idle for a time interval exceeding SIFS. After the successful exchange of RTS and CTS frames the data frame can be sent by the transmitter after waiting for a time interval SIFS. In case a CTS frame is not received within a predetermined time interval, the RTS is retransmitted following the backoff rules as specified in the basic access procedures outlined above. The channel access method using RTS and CTS frames is summarized in Figure 2.5 [Litjens 03_1].

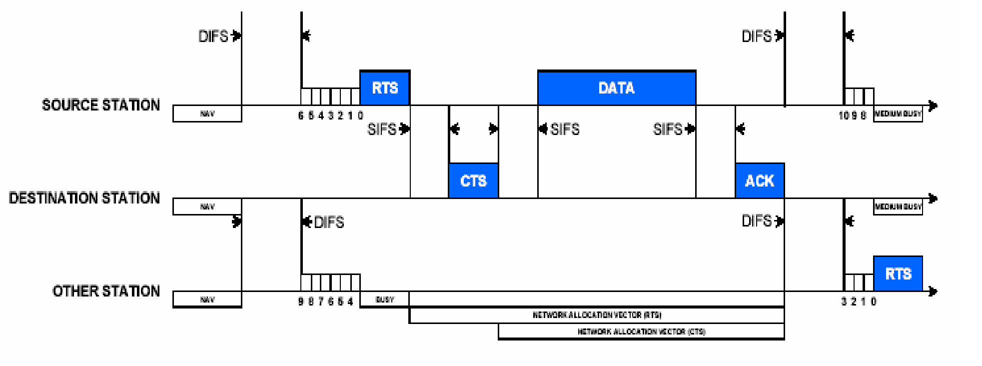


Figure 2.5 The RTS/CTS Access Mechanism

The RTS and CTS frames contain a *duration field* that indicates the period the channel is to be reserved for transmission of the actual data frame. This information is used by stations that can hear either the transmitter and/or the receiver to update their *Net Allocation Vectors* (NAV) - a timer that is always decreasing if its value is non-zero. A station is not allowed to initiate a transmission if its NAV is non-zero. The use of NAV to determine the busy/idle status of the channel is referred to as the *Virtual Carrier sense* mechanism. Since stations that can hear either the transmitter or the receiver resist from transmitting during the transmission of the data frame under consideration the probability of its success is increased. However, this increase in the probability of successful delivery is achieved at the expense of the increased overhead involved with the exchange of RTS and CTS frames, which can be significant for short data frames.

2.3.3. The physical layer

The 802.11 physical layer (PHY) is the interface between the MAC and the wireless media where frames are transmitted and received. 802.11 provides three different PHY definitions. Both *Frequency Hopping Spread Spectrum* (FHSS) and *Direct Sequence Spread Spectrum* (DSSS) support 1 and 2 Mbps data rates. An extension to the 802.11 architecture (802.11a) defines different multiplexing techniques that can achieve data rates up to 54 Mbps. Another extension to the standard (802.11b) defines 11 Mbps and 5.5 Mbps data rates (in addition to the 1 and 2Mbps rates) utilizing an extension to DSSS called *High Rate-Direct Sequence Spread Spectrum* (HR-DSSS), also known as Complementary Code Keying (CCK). However, the MAC mechanisms described in the above section are unchanged for IEEE 802.11b, so generally 802.11b can be treated as a legacy 802.11 with a higher access rate [802.11b]. Currently 802.11b WLAN is widely applied and has replaced most legacy 802.11 WLANs. So we focus on the DSSS, and, furthermore, use the parameters described for operations in the 2.4 GHz ISM band, known as 802.11b.

The 802.11 PHY defines a packet structure with three major parts, the preamble, header, and payload. Two different preambles and headers are defined: the mandatory Long Preamble and header and an optional Short Preamble and header. Figure 2.6 shows the format for the Long PHY protocol data units (PPDU). Most 802.11b stations mainly support only the Long Preamble and header mode. The Short Preamble and header mode is intended for operations where maximum throughput between the stations is desired and interoperability with legacy equipments is not a consideration. Our research is mainly based on the 802.11b standards and only the Long Preamble and header mode are considered.

Furthermore, different parts are transferred with different data rates. For example, in the case of the Long Preamble and header a long preamble of 144 bits is used. Both preamble and header are transmitted with a 1 Mbps data rate. The bits of the Payload (real data), however, use 802.11b, and can thus be transmitted with 11Mbps.

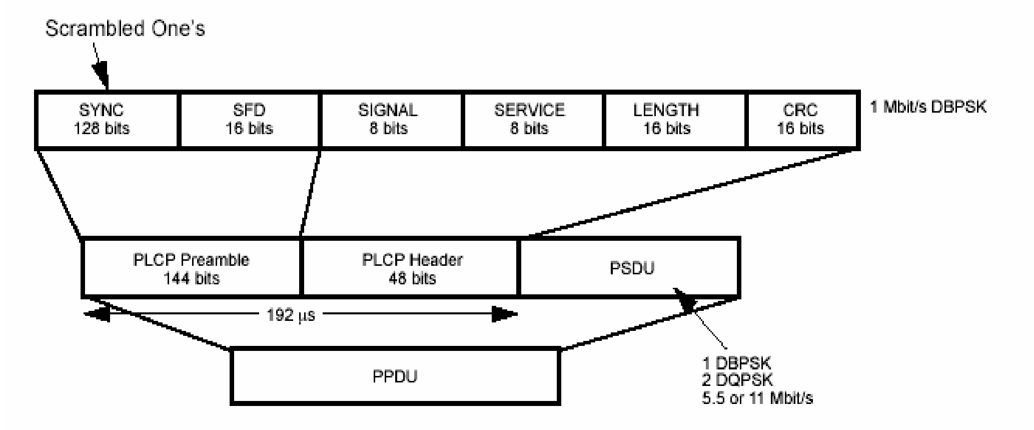


Figure 2.6 Packetization on WLAN PHY layer with Long Preamble and Header

2.4 The Transmission Control Protocol

Transport protocols have an important function in data transmission over the Internet. The protocols, which are mainly used, are the User Datagram Protocol (UDP) and the Transmission Control Protocol (TCP). Since wireless access to the Internet is the most common application of this kind of networks, it is reasonable to assume that the traffic will be carried over the well-known TCP/IP protocol suite. Hence, we only explain the TCP protocols in more details below.

The transmission control protocol [TCP] is the standard transmission layer protocol for providing a point-to-point full-duplex connection-oriented service. In contrast to UDP, a TCP necessitates a connection to be set up and kept alive between two hosts for sending a certain amount of data. A three-way handshake is done to inform each other about allocated buffers (sender and receiver) and initial values of sequence numbers one for the sender and one for the receiver. The sequence number is increased when each new segment sent and is used to keep track of the segments in the transmission. The receiver in turn, will send so-called *acknowledgement* (ACK) packets. With the built-in feedback mechanism, a TCP source can detect if a packet has been lost and has to be transmitted again, thus providing a reliable transfer of data. The feedback mechanism also makes it possible for TCP to detect the network conditions, upon which its sending rate is calculated. TCP adds more overhead to the network than UDP. Besides the additional TCP ACK packets, TCP's header is also larger than UDP. To regulate the rate at which segments are sent, TCP implements two control mechanisms, the *flow control* and the *congestion control*.

2.4.1. TCP flow control

TCP's flow control is a technique whose primary purpose is to properly match the transmission rate of the sender to that of the receiver and the network. TCP's flow control is end-to-end, which means that the sender must not inject into the network more than the receiver can hold in its buffer. It is important for the transmission to be at a high enough rate to ensure good performance, but also to protect against overwhelming the network or the receiving host.

This control is done with a window that limits the number of packets the sender can inject into the network and that slides when the sender receives an acknowledgement saying that the receiver has correctly received some data and handed them to the application. A sending station is allowed to transmit more packets during one RTT. The time interval between the moment a signal is sent and the moment a response is received is defined as the *Round-Trip Time* (RTT). The RTT contains two major components. The first one is a fixed part that is caused by the propagation delay. The second component is a variable part that is caused by queuing delays, which depend on the degree of congestion at the transmission links.

The maximum number of outstanding packets, which have been sent and are still not acknowledged, is determined by a *receiver advertised window*, maintained at the sender. The size of the *advertised window* is determined at the receiver, by subtracting buffer space used from the buffer allocated. This window is advertised by the receiver at the connection set-up period and it is updated during the connection lifetime if the buffer space at the receiver changes. Since the sender knows the number of unacknowledged segments, it can compare this with the *advertised window* to make sure that it does not overflow the receiver buffer.

2.4.2. TCP congestion control

TCP congestion control and Internet traffic management issues in general are an active area of research and experimentation. Congestion control in data networks means that the different sources must adapt their transmission rates as a function of network load. The rate control in TCP is done by changing the size of a congestion window that has been added to the protocol [Jacobson 88] and that also limits the number of packets the source can transmit before the receipt of an acknowledgement. Furthermore, this window is upper bounded by the receiver window for end-to-end flow control purposes [Stevens 97].

The congestion control mechanism of TCP consists of four different algorithms: slow start, congestion avoidance, fast retransmit and fast recovery [Stevens 97]. They act together to try to avoid overflowing intermediate router buffers in the network between the sender and receiver. Congestion control uses an additional *congestion window*, to regulate the senders' transmission rate. The amount of unacknowledged data must be less than or equal to the minimum of *congestion window* and *receiver advertised window*.[Jacobson 90]. Below we present a brief description of these features.

Slow start

Slow Start is a mechanism used by the sender to control the transmission rate, otherwise known as the sender-based flow control. This is accomplished through the return rate of acknowledgements from the receiver. In other words, the rate of acknowledgements returned by the receiver determines the rate at which the sender can transmit data. As stated before, the sender in each TCP connection maintains a window of the maximum number of outstanding packets, the so-called *congestion window* (C_{wnd}). At the same time there is also an *advertised window* by the receiver. When a new TCP connection is established in a network, the Slow Start algorithm initializes the C_{wnd} to one segment, which is the maximum segment size (MSS) initialized by the receiver during the connection establishment phase. When acknowledgements are returned by the receiver, the congestion window increases by one segment for each acknowledgement returned. Thus, the sender can transmit the minimum of the congestion window and the advertised window of the receiver, which is simply called the transmission window. This provides an exponential increase of C_{wnd} during slow start in each RTT. The maximum number of ongoing packets is increased accordingly.

Congestion avoidance

During the initial data transfer phase of a TCP connection the Slow Start algorithm is used. However, there may be a point during the Slow Start that the network is forced to drop one or more packets due to overload or congestion. If this happens, Congestion Avoidance is used to slow the transmission rate. In the Congestion Avoidance algorithm a retransmission timer expiring or the reception of duplicate ACKs can implicitly signal the sender that a network congestion situation is occurring. If the sender detects that the link is "congested", TCP's Congestion Avoidance algorithm applies. At the establishment of a TCP connection the *SlowStart Threshold* (*SSThresh*) is set to a default value, which is dependent on the TCP implementation. At the moment that the sender is aware of link congestion, i.e., time-outs or duplicated ACKs have occurred, it sets *SSThresh* to one-half of the current window size C_{wnd} (the minimum of the congestion window and the receiver's advertised window size) and sets C_{wnd} to one. It means that the TCP source has to do slow start again. At a given moment C_{wnd} will be greater than *SSThresh* (half way to the previous congestion point), and then collision avoidance will take over. From now on C_{wnd} will increase by 1/ C_{wnd} at each successful reception of each ACK. This results in an increase of one C_{wnd} in each RTT, as opposed to the exponential increase during slow start. If congestion was indicated by duplicate ACKs, the Fast Retransmit (and/or Fast Recovery) algorithms are invoked (see below).

Fast retransmit

When a duplicate ACK is received, the sender does not know if it is because a TCP segment was lost or simply that a segment was delayed and received out of order at the receiver. If the receiver can re-order segments, it should not be long before the receiver sends the latest expected acknowledgement. Typically no more than one or two duplicate ACKs should be received when simple out of order conditions exist. If however more than two duplicate ACKs are received by the sender, it is a strong indication that at least one segment has been lost. In this case, the sender does not even wait for a retransmission timer to expire before retransmitting the segment (as indicated by the position of the duplicate ACK in the byte stream). This process is called the Fast Retransmit algorithm and was first defined in [Jacobson 90].

In Figure 2.7 the slow start, collision avoidance and fast retransmit mechanism of a TCP connection are illustrated.

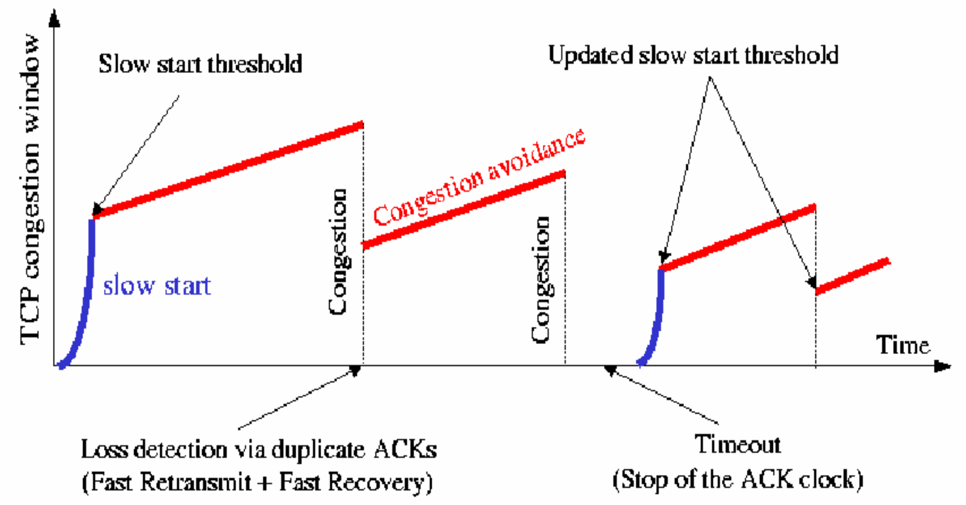


Figure 2.7 TCP congestion control mechanisms

2.5 Network simulator 2 (NS-2)

In most studies on mobile ad hoc networks, simulations are used for the evaluation of network protocols and devices under certain specific conditions. Of the different academic and commercial network simulation tools, in this project, we will focus on one such academic network simulation tool called the Network Simulator or mostly referred to as NS-2.

The Network Simulator 2 (NS-2) is commonly used and widely accepted in the networking research community as the basic simulating tool for network evaluations, mostly because it is open-source [Fall 97], [NS-2]. NS-2 is a discrete event simulator targeted at networking research. NS-2 provides substantial support for simulation of TCP, routing, and multicast protocols over wired and wireless (local and satellite) networks. Because of its good TCP implementations, it has been often used to simulate and evaluate TCP performance as well.

Berkeley released the initial code that made wireless network simulations possible in NS-2. That code provided some support to model wireless LANs, but was fairly limited. As a result of the Monarch project at Carnegie Mellon University [CMU] the simulator was extended with support for node mobility, a realistic physical layer, radio network interfaces and an implementation of the IEEE 802.11 DCF MAC protocol. This work was presented as part of a larger study of performance for different ad-hoc routing protocols. It was this contribution that made it possible to perform real wireless simulations [Broch 99] with NS.

NS-2 is an event based simulator, where a scheduler activates events sequentially. The idea of a discrete event scheduler is that actions may only be started as a result of an event. At present, there are four different schedulers which use different data structures implemented in NS. Usually the scheduler selects the next event from the event queue and finishes it. After that the next event is taken and so on. There are no parallel operations in NS. If two or more events have same starting time, the scheduler selects the event that was first put into the event queue.

NS-2 is a hybrid of C++ and OTcl (Object Tcl) programming languages. OTcl works mainly as a configuration and command interface but is also a part of an implementation of NS. This 'two language'- dilemma is also the origin of the some problems encountered in NS; while C++ has to be compiled, OTcl is interpreted. One great benefit of this is that there is no need to recompile the simulator between different simulations since you are able to set up topology, link bandwidth, traffic sources, etc. from the OTcl scripts. So, once you have implemented the basic functionality within the simulator (C++ code) you only have to change the OTcl scripts to run various simulations.

The underlying channel model in NS-2 is quite simple. Each mobile node has a position and a velocity and moves around on a topography that is specified using either a digital elevation map or a flat grid. The position of a mobile node can be calculated as a function of time, and is used by the radio propagation model to calculate the propagation delay from one node to another and to determine the power level of a received signal at each mobile node.

Each mobile node has one or more wireless network interfaces, with all interfaces of the same type (on all mobile nodes) linked together by a single physical channel. When a network interface transmits a packet, it passes the packet to the appropriate physical channel object. This object then computes the propagation delay from the sender to every other interface on the channel and schedules a “packet reception” event for each. This event notifies the receiving interface that the first bit of a new packet has arrived. At this time, the simulator calculates the receiving power *Pr* for every transmission between two nodes with the chosen propagation model. The power level is compared to two different values: the carrier sense threshold CS_{Thresh} and the receive threshold RX_{Thresh}.

The channel model distinguishes primarily between three cases. In case *Pr* is greater than the receiving threshold RX_{Thresh}, the transmission has enough power to allow proper reception at the receiver side. Other simultaneous transmissions with reasonable transmission powers may certainly interfere with this transmission and make a correct reception impossible. Then the packet is simply handed up to the MAC layer. If *Pr* is below RX_{Thresh} but greater than the carrier sense threshold CS_{Thresh}, the packet is marked as a packet in error before being passed to the MAC layer and the receiving node must drop the packet. However, the receiving power of this transmission is still strong enough to interfere with other simultaneous transmissions. Consequently, these interfered packets are also invalid and nodes must drop them as well. Transmissions with receiving powers *Pr* smaller than CS_{Thresh} do not even obstruct other simultaneous transmissions at the same node. And the packet is discarded as noise.

To allow reasonable simulations within an acceptable amount of time, propagation models must simplify calculations and reduce the required computation to a minimum. The network simulator NS-2 knows three different propagation models to simulate wireless ad hoc networks, the free space (FS) model, the two ray ground (TRG) model and the shadowing model. The Ricean and Rayleigh model is extended by the ARC Group of Carnegie Mellon University [Punnoose 00]. In our research's simulation, the TRG model is used. More details are described in the next chapter.

Once the MAC layer receives a packet, it checks to insure that its receive state is presently "idle". If the receiver is not idle, one of the following two things can happen. If the power level of the packet already being received is at least 10 dB greater than the received power level of the new packet, we assume that capture this packet, discard the new packet, and allow the receiving interface to continue with its current receive operation. Otherwise, a collision occurs and both packets are dropped. If the MAC layer is idle when an incoming packet is passed up from the network interface, it simply computes the transmission time of the packet and schedules a "packet reception complete" event for itself. When this event occurs, the AC layer verifies that the packet is error-free, performs destination address filtering, and passes the packet up the protocol stack.

The link layer implements the complete IEEE 802.11 standard [IEEE 99] Medium Access Control (MAC) protocol Distributed Coordination Function (DCF) in order to accurately model the contention of nodes for the wireless medium. DCF is designed to use both *physical carrier sense* and *virtual carrier sense* mechanisms to reduce the probability of collisions due to hidden terminals. The transmission of each unicast packet is preceded by a Request-to-Send/Clear-to-Send (RTS/CTS) exchange that reserves the wireless channel for transmission of a data packet. Each correctly received unicast packet is followed by an Acknowledgment (ACK) to the sender, which retransmits the packet a limited number of times until this ACK is received. Broadcast packets are sent only when virtual and physical carrier sense indicates that the medium is clear, but they are not preceded by RTS/CTS and are not acknowledged by their recipients.

2.6 Related work

A lot of research is done in the field of performance modeling of the IEEE 802.11 protocol. Most of the earlier work is focused on the simulation of the efficiency of the IEEE 802.11 protocol, by investigating the maximum throughput that it can achieve under various network configurations, e.g., [Weinmiller 97].

Bianchi developed and analyzed a detailed mathematical performance model of DCF [Babich 00], which has been improved later by Wu et al [Wu 02]. Both papers have studied the "saturation throughput" of IEEE 802.11's MAC layer, which represents the maximum load the system can carry in stable conditions. Next to this, both assume a simplified physical layer model. They propose to model the MAC access mechanism with a Markov Chain, neglecting only minor dependencies among the behavior of different stations. This yields an accurate approximation for the WLAN saturation throughput. However, persistent UDP flows instead of TCP are studied in their work, which means that the sources always have packets ready to send in their queue.

Based on this Markov Chain model, the situation with non-persistent traffic sources is studied in [Litjens 03_2] and [Winands 04]. The number of active stations varies dynamically in time according to the initiation and completion of file transfers at random time instants. In their research they have proposed an integrated packet/flow level modeling approach. On packet level the WLAN DCF performance can be modelled using a discrete time Markov Chain as described in [Babich 00]. From the flow level point of view, the WLAN can be viewed as a processor sharing type of queuing system.

On the current Internet, the major part of traffic is controlled by TCP. TCP also plays an important role in the data flow over WLAN and its performance has a direct effect on an end-user's perceived quality. Extensive works have also been conducted to model the behavior of TCP over wired links, e.g., [Mathis 97], [Padhye 98]. The steady-state throughput of a TCP flow under given packet loss and roundtrip time conditions are studied in these papers. But very few research results are known about its behavior over IEEE 802.11 WLAN. [Lassila 03] proposed an integrated packet/flow level model for estimation of the mean transfer time of TCP flows over a fixed capacity (full duplex) bottleneck link. For the packet level they use the results of [Kelly 01]; for the flow level a processor sharing type of queueing model is used reflecting TCP's design principle of fair resource sharing

Mobility of wireless stations may be the most important feature of a wireless LAN. A WLAN would not serve much purpose if stations were not able to move about freely from location to location either within a specific WLAN or between different WLAN 'segments'. So it is important to understand the impact of the mobility of the mobile terminal on performance of mobile internet service via WLAN networks**.** Burmeister investigated the performance of mobile terminals by simulation [Burmeister 04]. These simulation results give us some idea about the relationship of network factors and the performance metrics. However, to the authors' knowledge there are no papers providing analytical modeling approaches to the performance of TCP over WLAN experienced by the mobile terminals.

2.7 Conclusions

In this chapter, the background information of our research is introduced. Wireless networking is a hot industry segment. Several wireless technologies have been targeted primarily for data transmission: 802.11, Bluetooth, and Third-generation (3G) mobile networks. However, equipment based on the 802.11 standard has been an astounding success, because of its high performance and low price.

In our studies we focus on the 802.11 architecture ESS. In the MAC sub layer we studied the DCF, which is based on the Carrier Sense Multiple Access with Collision Avoidance protocol. We also studied two control mechanisms implemented in TCP, the flow control and the congestion control. The Network Simulator 2 (NS-2) is used for simulation validations of our models in our research. It is commonly used and widely accepted in the networking research community as the basic simulating tool for network evaluations. The DCF MAC protocol is implemented in NS-2, which can handle DATA/ACK/RTS/CTS, as well as the broadcasting type of packets.

Chapter 3
The single-class single-cell model

3.1 Introduction

Our research goal is to analyze the performance under a given scenario. To this end, we first consider cellular networks based on the 802.11 (WLAN) protocol and investigate the performance models from simplified scenario to complicated realistic scenario.

Because the factors that can affect the performance of wireless communication are highly complicated, it is better to first simplify the factors which are taken into account when we build the models. To start, in Section 3.2 we only consider one very simplified scenario. In this scenario a single mobile station drives across the coverage of one single AP cell which follows one given direction with a fixed velocity. We develop a quantitative model for the throughput for this scenario. To evaluate this model, we simulate this simple scenario via NS2.

Based on the results of the previous step, in Section 3.3 we can analyze the average throughput of any number of stations when they cross the coverage of the AP. It is important to consider the effect between the different user flows. We consider some characteristics of user behavior, for example, the user arrival rates, the user mobility pattern, and so on. In the same way, we evaluate these models by simulating the scenario in NS2.

At the end, a conclusion of this chapter will be given in Section 3.4.

3.2 The single mobile station model

3.2.1 Simulation preliminaries

We start with an analysis of the simplest case, namely a single TCP connection between the Access Point (AP) and the mobile station. We now focus on the packet level studies. As we explained in sections 2.3 and 2.4, both TCP and 802.11 MAC layers ACK packets are sent. The interaction between the TCP source and destination plays an important role. (To avoid confusion in the remainder of this thesis, we denote ACK packets on WLAN MAC layer by MAC ACK and ACK packets in a TCP flow by TCP ACK.)

To get a better insight into the interactions between 802.11 MAC and TCP, and how NS-2 interprets them, we have run a simulation with only one TCP connection involved. The TCP connection is made between the AP and one mobile wireless station (STA). The STA moves with determined velocity through the coverage of the AP, as shown in Figure 3.2. In this simulation, the traditional TCP protocol (Tahoe) is used without delayed acknowledgement, which is used in a vast amount of applications on different platforms today. Part of the trace file generated by NS-2, about the transmissions of the first few TCP data packets and other relevant packets, is shown below.

```
#=============================================================
s 19.005462874  0  AGT  --- 4 tcp 1040 [0 0 0 0] ------- [0:0 1:0 32 0]
s 19.005462874 _0_ AGT  --- 5 tcp 1040 [0 0 0 0] ------- [0:0 1:0 32 0]
r 19.005752661 _1_ MAC  --- 0 ACK 38 [0 1 0 0]
s 19.005941874 _0_ MAC  --- 0 RTS 44 [13ee 1 0 0]
r 19.006294661 _1_ MAC  --- 0 RTS 44 [13ee 1 0 0]
s 19.006304661 _1_ MAC  --- 0 CTS 38 [12b4 0 0 0]
```

```
r 19.006609448 _0_ MAC  --- 0 CTS 38 [12b4 0 0 0]
s 19.006619448 _0_ MAC  --- 4 tcp 1092 [13a 1 0 800] ------- [0:0 1:0 32 0]
r 19.011084236 _1_ MAC  --- 4 tcp 1040 [13a 1 0 800] ------- [0:0 1:0 32 0]
s 19.011094236 _1_ MAC  --- 0 ACK 38 [0 0 0 0]
r 19.011109236 _1_ AGT  --- 4 tcp 1040 [13a 1 0 800] ------- [0:0 1:0 32 0]
s 19.011109236 _1_ AGT  --- 6 ack 40 [0 0 0 0] ------- [1:0 0:0 32 0] [1 0]
r 19.011399023 _0_ MAC  --- 0 ACK 38 [0 0 0 0]
s 19.011489023 _0_ MAC  --- 0 RTS 44 [13ee 1 0 0]
r 19.011841810 _1_ MAC  --- 0 RTS 44 [13ee 1 0 0]
s 19.011851810 _1_ MAC  --- 0 CTS 38 [12b4 0 0 0]
r 19.012156597 _0_ MAC  --- 0 CTS 38 [12b4 0 0 0]
s 19.012166597 _0_ MAC  --- 5 tcp 1092 [13a 1 0 800] ------- [0:0 1:0 32 0]
r 19.016631385 _1_ MAC  --- 5 tcp 1040 [13a 1 0 800] ------- [0:0 1:0 32 0]
s 19.016641385 _1_ MAC  --- 0 ACK 38 [0 0 0 0]
r 19.016656385 _1_ AGT  --- 5 tcp 1040 [13a 1 0 800] ------- [0:0 1:0 32 0]
s 19.016656385 _1_ AGT  --- 7 ack 40 [0 0 0 0] ------- [1:0 0:0 32 0] [2 0]
r 19.016946172 _0_ MAC  --- 0 ACK 38 [0 0 0 0]
s 19.017075385 _1_ MAC  --- 0 RTS 44 [44e 0 1 0]
r 19.017428172 _0_ MAC  --- 0 RTS 44 [44e 0 1 0]
s 19.017438172 _0_ MAC  --- 0 CTS 38 [314 1 0 0]
r 19.017742959 _1_ MAC  --- 0 CTS 38 [314 1 0 0]
s 19.017752959 _1_ MAC  --- 6 ack 92 [13a 0 1 800] ------- [1:0 0:0 32 0]
r 19.018217747 _0_ MAC  --- 6 ack 40 [13a 0 1 800] ------- [1:0 0:0 32 0]
s 19.018227747 _0_ MAC  --- 0 ACK 38 [0 1 0 0]
r 19.018242747 _0_ AGT  --- 6 ack 40 [13a 0 1 800] ------- [1:0 0:0 32 0]
s 19.018242747 _0_ AGT  --- 8 tcp 1040 [0 0 0 0] ------- [0:0 1:0 32 0] [3 0]
r 19.018532534 _1_ MAC  --- 0 ACK 38 [0 1 0 0]
s 19.018742534 _1_ MAC  --- 0 RTS 44 [44e 0 1 0]
r 19.019095321 _0_ MAC  --- 0 RTS 44 [44e 0 1 0]
s 19.019105321 _0_ MAC  --- 0 CTS 38 [314 1 0 0]
r 19.019410108 _1_ MAC  --- 0 CTS 38 [314 1 0 0]
s 19.019420108 _1_ MAC  --- 7 ack 92 [13a 0 1 800] ------- [1:0 0:0 32 0]
r 19.019884896 _0_ MAC  --- 7 ack 40 [13a 0 1 800] ------- [1:0 0:0 32 0]
s 19.019894896 _0_ MAC  --- 0 ACK 38 [0 1 0 0]
r 19.019909896 _0_      ---        40 [13a 0 1 800] ------- [1:0 0:0 32 0]
#============================================================
```

At time 19.005462874 the Access Point transferred two TCP data packets (Packet 4 shown in Red and Packet 5 shown in Blue). After the RTS/CTS handshake between AP (node_0_) and STA (node _1_), the AP's MAC layer transmits packet 4. After STA's MAC layer received packet 4 at 19.011094236, it is forwarded it up to the TCP layer, and a TCP ACK is generated (*ack* packet 6 shown in Green). At the same time, a MAC ACK segment is being transmitted. After the successful transmission of the MAC ACK, the receiver waits more than a DIFS before it begins with the transmission of the TCP ACK packet. The interval is determined to be always a sum of the duration of a DIFS and a random number of timeslots. Apparently, the medium is sensed busy because the receiver itself is sending a MAC ACK packet, if the TCP ACK packet is offered to the MAC layer. In the meantime, the sender has a packet (packet 5 in blue) queued in its TX-buffer. It can be seen from the trace file that the TCP data packet "wins" this contention and is granted to transmit first. Later, at time 19.016656385, the same situation occurs again. The STA generates ACK packet 7 (shown in pink). This time the receiver "wins" and hence the TCP ACK (packet 6) is transmitted first.

3.2.2 Modeling the single mobile station case

The mobility of wireless stations may be the most important feature of a wireless LAN. Compared to a wired LAN, WLAN is easier to install and serves nicer since stations are able to move about freely from location to location either within a specific WLAN or between different WLAN 'segments'. It is thus important to understand the impact of the mobility of the mobile terminal on the performance of the mobile internet service via WLAN networks**.** In our research, we focus on performance as experienced by the end users in the Access Point domain (in terms of throughput and total amount of received data), for a given set of protocol choices and realistic user-behavior scenarios. In this section we first derive the expression of the performance for only one single station in the AP domain (cell).

Modeling the performance of wireless channels is usually a complex task, because the performance of wireless channels inherently depends on radio propagating modes, such as, line of sight (LOS) radiation, reflections from a smooth surface, diffractions around a corner, and scattering caused by an object with dimensions on the order of the wavelength. To derive the closed-form expression of the throughput of the system, we can not consider all the details. Hence, we derive the expression step by step, from a physical radio propagation model, to the MAC model and finally to the TCP performance expression.

Mobile radio propagation model

The mobile radio channel places fundamental limitations on the performance of wireless communication systems. The transmission path between the transmitter and the receiver can vary from a simple line-of-sight to one that is severely obstructed by buildings, mountains, and foliage. Unlike the wired channels that are stationary and predictable, radio channels are extremely random and do not offer easy analysis. Even the speed of motion impacts how rapidly the signal level fades as mobile terminal moves in space.

Based on the distance over which a mobile moves, there are two different types of fading effects: large-scale fading and small-scale fading [Sklar 97]. Large-scale fading represents the average signal power attenuation due to motion over large areas. Small-scale fading refers to the dramatic changes in signal amplitude and phase that can be experienced as a result of small changes (as small as a half-wavelength) in the distance between the transmitter and the receiver. If the mobile moves away from the transmitter over a large distance, the received signal will experience a large-scale signal variation. When there are a large number of reflective paths with no LOS signal components, the envelope of the received signal can be statistically described by the Rayleigh distribution [Babich 00]. If dominant non-fading components exist, such as a LOS propagation path, the small-scale fading envelope is Rician distributed [Babich 00].

One of the most important characteristics of the propagation environment is the path (propagation) loss. An accurate estimation of propagation losses provides a good basis for a proper selection of base station locations and a proper determination of the frequency plan. An accurate prediction of the field strength level is a very complex and difficult task. To date, various field strength prediction methods have been proposed in the literature, e.g. [Rappaport 02] [Neskovic 00] [Simon 98]. In our research, the STAs move in the outdoor environment and the speed of movement is relatively fast. So the effect from the small-scale fading is relatively small. Furthermore, the small-scale fading also introduces some kind of unpredictability for data transmissions. To simplify the complexity, for physical radio propagation, we take the Two-Ray Ground (TRG) model.

The two-ray ground reflection model is a useful propagation model, an improved version of the Free Space (FS) model, which is based on geometric optics, and considers both the direct path and a ground reflected propagation path between the transmitter and the receiver. This model has been found to be reasonably accurate for predicting the large-scale signal strength over distances of several kilometers for mobile radio systems that use tall towers, as well as for line-of-sight micro-cell channels in urban environments. As with the FS model, both nodes are assumed to be in LOS. The receiving power *Pr* depends on the transmitted power *Pt*, the gain of the receiver and transmitter antenna (*Gt*, *Gr*) the wavelength λ, the distance *d* between both nodes and a system loss coefficient *L*. The heights of both antennas over the ground are depicted with *ht* and *hr*. All parameters, but the distance *d*, are system wide constants. Up to the crossover distance $d_{Thresh} = 4\pi \cdot h_t \cdot h_r / \lambda$, the TRG model is equal to the FS model. Beyond this distance, the ground reflection destructively interferes with the direct ray and further reduces the field strength. The receiving signal strength is then inverse proportional to d^4. Just like the

FS model, TRG contains only the distance between sender and receiver as variable parameters. The receiving power *Pr* can be expressed as

$$P_r = \begin{cases} \dfrac{P_t \cdot G_t \cdot G_r \cdot \lambda^2}{(4\pi \cdot d)^2 \cdot L} & d < d_{Thresh} \ , \\ \dfrac{P_t \cdot G_t \cdot G_r \cdot h_t^2 \cdot h_r^2}{d^4 \cdot L} & d \geq d_{Thresh} \ . \end{cases}$$

For both cases in the TRG model, the sender-receiver distance is the only variable parameter during the single transmission. This forms a circular coverage around a sending node and a sharp range limit. Beyond this range, no further reception is possible. Since we do not consider fading or shadowing effects, there will be no random events. So whether the packets are received or not is trivial, and this only depends on the distance. If the distance between the AP and STA is within the AP's coverage, the receiving packets will be higher than the critical threshold, so the packets are always correctly received. Inversely, if the STA is out of the AP's range, no packets are received correctly. We can make the reasonable assumption that the channel is loss free during the period when the STA keeps inside the hearing range of the Access Point.

TCP throughput model

As we explained before, our research focuses on the average throughput that the station can achieve during the period inside the AP coverage. According to the definition of the throughput (TP), we are able to express the TP as the packet length (excluded the protocol overhead) divided by the time needed to transfer this packet.

$$TP = \frac{E[data\ payload\ size]}{E[time\ needed\ to\ transfer\ these\ data]} \ . \tag{3.1}$$

We have explained in Section 2.3, there is some overhead added to the data payload for the protocol usage: the physical layer header, the long PLCP preamble header, the MAC layer header plus the FCS field and the TCP/IP header. Furthermore, not all bits sent by a station can profit from the higher access rates. Both the PLCP preamble and the physical header are transmitted only with 1 Mbps data rate. The basic rate for the control segment RTC and the CTS is 1Mbps or 2 Mbps. We can now calculate the time necessary to transmit the long PLCP preamble and the physical-layer header.

$T_{PHY} = L_{PHY}$ / 1Mbps = 48 bits / 1Mbps = 48μs ,

$T_{PLCP} = L_{PLCP}$ /1Mbps = 144 bits / 144 bits = 144μs .

We denote T_P as the total time spent on the PLCP and physical layer overhead transmission, so we have

$T_P = T_{PHY} + T_{PLCP}$ = 48μs + 144μs = 192μs . (3.2)

Furthermore, we denote

- R_{Basic} = the RTC/CTS transmission rate,
- R_{Data} = the data payload transmission rate,
- DIFS = the time interval Distributed InterFrame Spacing,
- SIFS = the time interval Short InterFrame Spacing,
- L_{RTS} = the length of the RTS control packet,

- L_{CTS} = the length of the CTS control packet,
- L_{MAC} = the overhead added by the MAC layer (MAC header plus FCS field),
- L_{TCPIP} = the length of the TCP/IP header,
- L_{ACK} = the length of the MAC layer ACK,
- L_{Data} = the length of the TCP packets payload,
- T_{slot} = the length of the time slot.

In our research, we care about the Infrastructure BSS, in which usually the RTS/CTS access mechanism is adopted. So here we only conduct the throughput for this access mode. According to the procedure explained in Section 2.3.2 and the 802.11 data format explained in Section 2.3.3, we can draw the data transmission figure as shown in Figure 3.1.

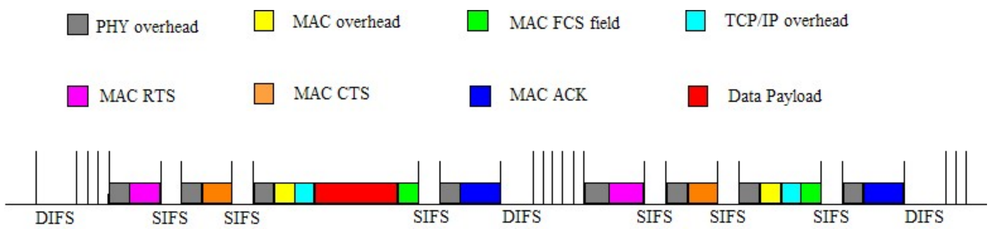


Figure 3.1 Data transmission in 802.11

First, we only consider about the raw transmission time for a TCP data packet and a TCP ACK packet, disregarding the backoff procedure. The WLAN MAC layer is not aware of the content of the data if it receives it from higher OSI layers. Hence TCP data packets and TCP ACK packets are treated ***equally***. Since there is only one physical channel available, it has been suggested that a TCP data packet contends with its own TCP ACK packet occasionally [Wu 02]. This means that even with one single TCP connection through AP, there are contentions. From Figure 3.1, we can easily derive the following expression of the time needed for the data packet,

$$T_{TCP\_Data} = DIFS + T_P + \frac{L_{RTS}}{R_{Basic}} + SIFS + T_P + \frac{L_{CTS}}{R_{Basic}} + SIFS + T_P + \frac{L_{MAC} + L_{TCPIP} + L_{Data}}{R_{data}} + SIFS + T_P + \frac{L_{ACK}}{R_{Basic}}$$

$$= DIFS + 4\,T_P + 3\,SIFS + \frac{L_{RTS} + L_{CTS} + L_{ACK}}{R_{Basic}} + \frac{L_{MAC} + L_{TCPIP} + L_{Data}}{R_{data}} . \qquad (3.3)$$

A TCP ACK packet consists just of the TCP/IP header and the MAC layer header, so the time for a TCP ACK transmission can be given by

$$T_{TCP\_Ack} = DIFS + T_P + \frac{L_{RTS}}{R_{Basic}} + SIFS + T_P + \frac{L_{CTS}}{R_{Basic}} + SIFS + T_P + \frac{L_{MAC} + L_{TCPIP}}{R_{Data}} + SIFS + T_P + \frac{L_{ACK}}{R_{Basic}}$$

$$= \text{DIFS} + 4\ T_P + 3\ \text{SIFS} + \frac{L_{RTS} + L_{CTS} + L_{ACK}}{R_{Basic}} + \frac{L_{MAC} + L_{TCPIP}}{R_{Data}}. \qquad (3.4)$$

In [Bianchi 00], it is proved that at backoff stage k, the average backoff time is half of the contention window size. So according to the exponential backoff mechanism, (more details are explained in Section 2.3.2), the contention window size is adjusted according to the following expression

$$cw(k) = \begin{cases} 2^k (cw_{\min} + 1), & 0 \le k \le m, \\ 2^m (cw_{\min} + 1), & m \le k \le \gamma. \end{cases} \qquad (3.5)$$

where γ is the maximum number of retransmissions for a given data packet and m is the maximum number of times that the contention window is doubled after a failed transfer attempt. Moreover, the average backoff time can be given by

$$\overline{T}_{backoff} = \frac{cw(k) - 1}{2}. \qquad (3.6)$$

As we explained in Section 2.3.2, each time when collisions happen, the Contention Window (CW) will be doubled, until CW achieves the maximum value. In WLAN, collisions will occur either when the backoff counters of multiple stations reach zero simultaneously, or in case a so-called hidden station fails to freeze its backoff counter when it cannot sense another station's transmission. In this single mobile station scenario, there are only two users (the Access Point and the Mobile Wireless Station) who may content the channel resource. Just as we explained in the simulation Section 3.1, when A is transmitting the packet, B can be able to always sense it as busy and transfer to the backoff model. Thus, the probability of these two nodes' backoff counters reaching zero at the same time is almost zero. Furthermore, only when they can hear each other, they can communicate. There is no probability of the existence of hidden nodes. Hence, we can state that there is no collision in this system, but it is collision-free. These two nodes will stay at backoff stage 0 all the time. We can give the average backoff time by specializing (3.5) and (3.6) to k=0

$$\overline{T}_{backoff} = \frac{cw(0) - 1}{2} = \frac{2^0 (cw_{\min} + 1) - 1}{2} = \frac{cw_{\min}}{2}. \qquad (3.7)$$

Furthermore, since there are only two nodes, there is also no traffic conflict in the TCP layer. We can make the following reasonable assumptions as in [Miorandi 04]:
1) The retransmission timer at the TCP source is large enough so that no timeouts take place 2) The transmit buffers at the nodes are well-dimensioned, in the sense that no packet drops take place due to buffer overflow.

Thus packet drops because of contention are considered negligible. Moreover, as we explained in the "*Mobile Radio Propagation Model"* section above, the network operates over a lossless channel. So in principle, the TCP congestion window would keep growing. However, since the TCP window is bounded by the receiver's advertised window W_{max}, this growth can not continue indefinitely. Thus after a transient phase, the TCP window stabilizes at the maximum TCP window.

We now use the above results to obtain *TP*, the throughput achieved by a single TCP connection in presence of two competing connections, i.e., the download link (AP to STA) and the upload link (STA to AP), where each receiver is employing a non-delayed ACK. In such a case, we get

$$TP = \frac{L_{Data}}{T_{TCP\_data} + T_{TCP\_ack} + 2 \cdot \overline{T}_{backoff} \cdot T_{slot}} = \frac{L_{Data}}{T_{TCP\_data} + T_{TCP\_ack} + cw_{\min} \cdot T_{slot}}, \tag{3.8}$$

where T_{TCP_Data} and T_{TCP_Ack} can be obtained from (3.3) and (3.4), so we can get

$$\frac{L_{Data}}{2DIFS + 8T_P + 6SIFS + \frac{2(L_{RTS} + L_{CTS} + L_{ACK})}{R_{Basic}} + \frac{2(L_{MAC} + L_{TCPIP})}{R_{Data}} + \frac{L_{Data}}{R_{Data}} + cw_{\min} \cdot T_{slot}}. \tag{3.9}$$

3.2.3 Numerical validation

To validate the model, we have run some simulations with the NS-2 2.26 simulator [NS-2]. The scenario used is shown in Figure 3.2. A Web server acts as the Internet Service Provider (ISP). One hot spot Access Point is connected via a wired link to the ISP network. The mobile station (user) connects to the Web server via a Wireless connection with the Access Point. The STA drives with constant velocity on one straight road which is across the AP's coverage.

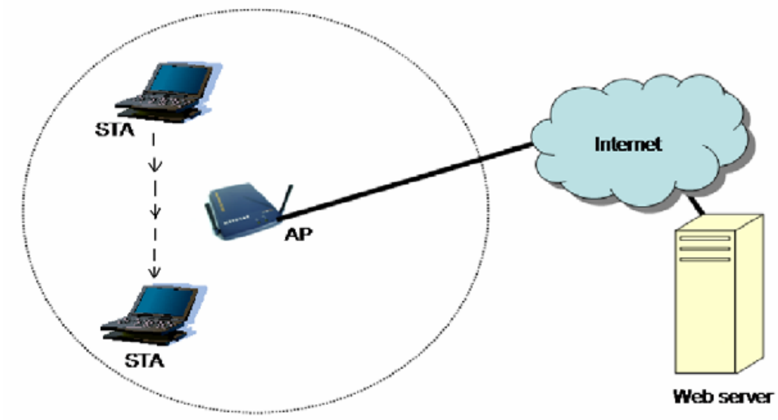


Figure 3.2 TCP over WLAN connected to a Web Server

The values of the parameters used to obtain numerical results, for both the analytical model and the simulation runs, are summarized in Table 3.1. The WLAN parameters are used as the packet level simulations are specified for the Direct Sequence Spread Spectrum (DSSS). The frame sizes are those defined by the 802.11b MAC specification [802.11b]. When we run the simulations with 11 Mbps WLAN's, we choose for WLAN the parameters data rate 11Mbps and basic rate 2Mbps (802.11b multi rates support, see details [802.11b] revised version 7.3.2.2 and 9.2). When we run the simulations with 1 Mbps, both the data rate and the basic rate are chosen as 1Mbps. Furthermore, we assume the average TCP packets are L_{Data} 8000bits long. Furthermore, to avoid the hidden node problem, we set the carrier-sensing range to be double of the receiving range. (More details about the hidden node problem are described in the next section.)

Parameter	**Value**	**Parameter**	**Value**
Radio-propagation model	Two-Ray Ground	Modulation	DSSS
Carrier Frequency	2412 MHz	Transmitted Power	24.5 dBm
Receive Threshold	-64 dBm	Carrier Sense Threshold	-78 dBm
Antenna Height	1.5 meters	Antenna Gain	1dB
Data Rate	{1, 2, 11} Mbps	Basic control rate	{1, 2} Mbps
CW min	31	CW max	1023
Retry limit (m)	7	γ	5
PHY header	48 bits	T_{PHY}	48µs
PLCP preamble	144 bits	T_{PLCP}	144µs
MAC header	272 bits	MAC ACK	112 bits
RTS	160 bits	CTS	112 bits
Time slot	20 µs	SIFS	10 µs
PIFS	30 µs	DIFS	50µs
TCP/IP header	320 bits	TCP packets	8000 bits

Table 3.1 The most important parameters for IEEE 802.11b

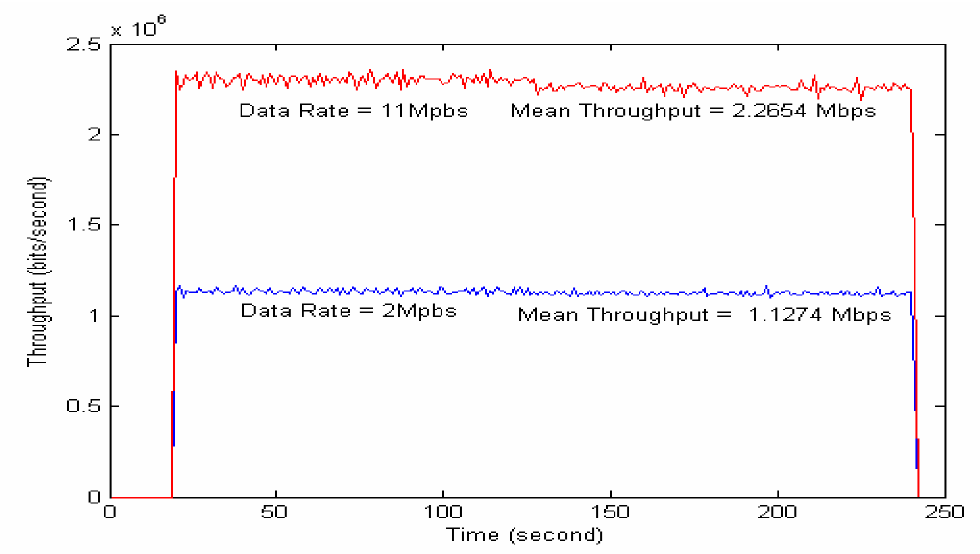


Figure 3.3 Throughput from simulation results

The TCP throughput gained by the station during his movement inside the AP's coverage is shown in Figure 3.3. The application gets a trigger from the WLAN interface, as soon as a network is available and an asscociation to an access point is established. After the authentication and authorization process, the user requires the download by sending a message to the server via the Access Point. A TCP connection is setup between the mobile station and the web server via Access Point. The server starts the transmission of the data via a TCP connection to the mobile station user. As we can see in Figure 3.3, the TCP throughput remains around the same value until the STA moves out of the hearing range. As it moves away, packets start getting dropped. The TCP server, as not receiving acknowledgements for all packets, assumes that the link is congested and decreases the sending rate. Together with the lost packets on the link, the TCP connection is finally lost.

Furthermore, these simulation results also support our loss-free channel assumption. From the trace file, we did not see any packets dropped when the STA was inside the range of the AP.

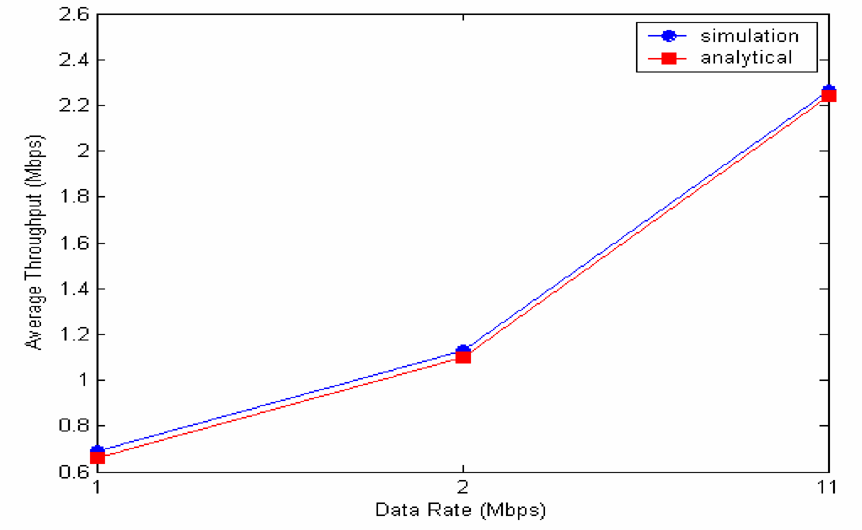


Figure 3.4 Average throughput: analysis versus simulation

Figure 3.3 shows the instant throughput the station gained, but our analytical model concerns about the average throughput. Figure 3.4 shows the average throughput results of the analytical model versus the simulation results. It shows that the analytical model is extremely accurate. The analytical results (red square) practically coincide with the simulation resutls (blue circle). The exact value is shown in Table 3.2. We can see the difference between analysis and simulation is small.

Physical Data Rate	Analysis (Mbps)	Simulation (Mbps)	difference
1Mbps	0.6621	0.69124	4.2%
2Mbps	1.0971	1.1274	2.6%
11Mbps	2.2631	2.2654	0.1%

Table 3.2 Comparison analysis vs. simulation

3.3 The Multiple stations single-class model

3.3.1 Simulation preliminaries

3.3.1.1 RTS/CTS issues

The performance of a wireless network is largely determined by the maximum data rate at the physical layer and the MAC layer protocols defined by the IEEE 802.11 standards [IEEE 99], [802.11 b]. As we explained in Section 2.3.2, CSMA/CA– Carrier Sensing Multiple Access is the most commonly used MAC protocol in WLANs. However, it has a problem when there are multiple nodes accessing a base station that cannot see or hear each other. This is also known as the hidden node problem**.** If a node is located within the transmission range of the receiver but not of the sender, it does not hear the packet exchange and may start sending packets to the same receiver. This results in costly data packet collisions. Such a node is called a hidden node [Tobagi 75], which is shown in Figure 3.5. We can see that node B is within the receiving range of both nodes A and C, but node A is hidden from node C and vice versa. In this case, node A is transmitting to node B. Node C cannot hear the transmission from A. During this transmission when C senses the channel, it falsely thinks that the channel is idle. If node C starts a transmission, it interferes with the data reception at B. Hence, hidden nodes can cause collisions on data transmission.

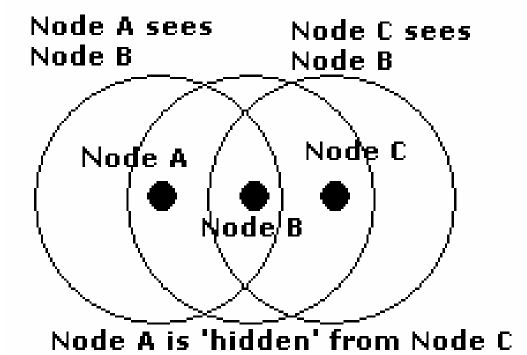


Figure 3.5 Hidden node problem in 802.11

Hidden nodes are solved by the use of a RTS/CTS protocol prior to packet transmission, which uses a variant of MACAW- *Multiple Access Collision Avoidance for Wireless* along with CSMA. In the three node network in Figure 3.5, node A sends a small RTS packet which is heard by node B which sends a small CTS packet which is heard by both nodes A and C. Node C will not transmit in this case. In order to gain more insight into the hidden node problem and RTS/CTS mechanism, and the way NS-2 interprets them, we have run some simulations.

In the simulator NS, every node has a signal receiving range and a carrier sensing range. In the Two-Ray Ground Reflection Radio Propagation Model, a node can not receive any packet from beyond its receiving range and cannot sense any neighboring transmission beyond the carrier sensing range. With the NS default parameters, the Carrier Sensing Range is calculated as 550m (CS_{Thresh} = -78dBm) which is double that of the Receiving Range of 250m (RX_{Thresh} = -64dBm). NS simulations with these default parameters did not show any signs of the hidden nodes problem. From Figure 3.6, we can see that when node A is transmitting a data packet to node B, although node C is not within the receiving range of node A, it senses this packet exchange between node A and node B as it is within the sensing range of node A. Node C would thereby refrain from sending packet to node B avoiding data packet collisions. This prevention of packet collision is only possible when the sensing range is set to double the receiving range. NS thus blocks all data packet collisions due to hidden nodes by setting these parameters. However in real networks, the hidden node problem exists and impacts network performances considerably, with equal receiving and sensing ranges. But modifying the CS_{Thresh} and RX_{Thresh} to be equal did show the hidden node problem without the RTS/CTS mechanism.

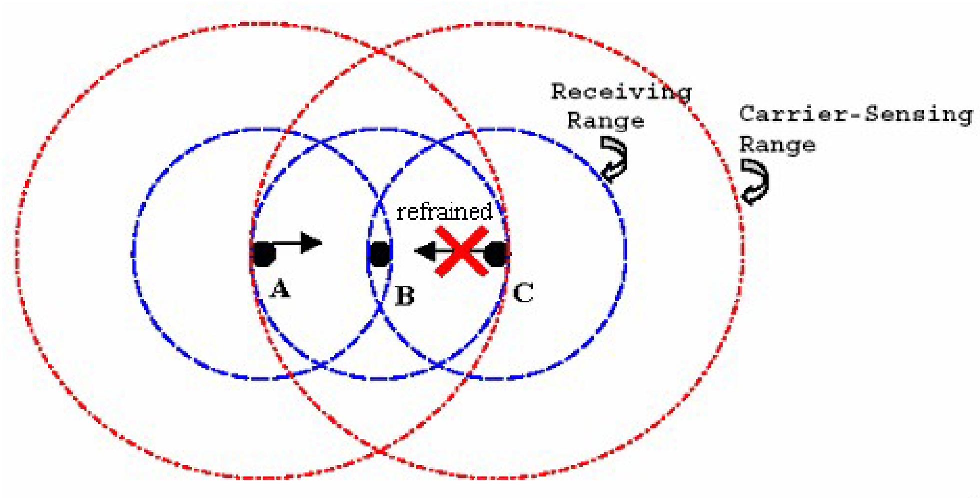
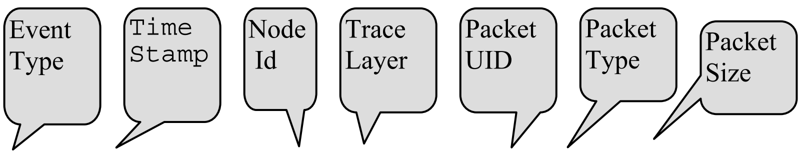


Figure 3.6 No hidden nodes problem in NS

We simulated the simple three-node scenario as shown in Figure 3.6, but with more realistic values as $RX_{Thresh} = CS_{Thresh}$ = -66dBm and no RTS/CTS mechanism. (A detailed description of the TCL file to simulate wireless networks in NS is given in the Appendix.) By default, NS has implemented the RTS/CTS mechanisms for all transmissions. However we can choose to use the basic mode. This RTS/CTS mechanism can be turned off by setting the *RTSThreshold* to a high value like 3000 bytes. This means that NO RTS will be sent for any data packet transmission of a size below 3000 bytes. If we then simulate the packet transmission with a lower packet size than 3000 bytes, no RTS will be sent.

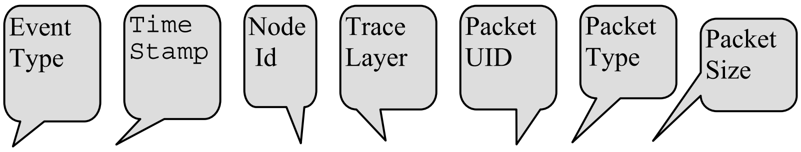
```
# Define the Mac RTS threshold to 3000
Mac/802_11 set RTSThreshold_ 3000
```

The simulation of the three nodes scenario of Figure 3.6 generated a trace file, which shows the data packets sending, receiving, and dropping events. The wireless trace file has a different format from the wired trace results. A part of the trace file for the CBR packet is as follows.




```
s  12.441355434  _2_  MAC ---  9  cbr 1050  [a2 1 2 800]
s  12.441535434  _0_  MAC ---  8  cbr 1050  [a2 1 0 800]
D  12.450059934  _1_  MAC ---  9  cbr 1050  [a2 1 2 800]
D  12.450239934  _1_  MAC ---  8  cbr 1050  [a2 1 0 800]
s  12.450907434  _2_  MAC ---  9  cbr 1050  [a2 1 2 800]
s  12.451747434  _0_  MAC ---  8  cbr 1050  [a2 1 0 800]
```

The *s, D* and ***r*** in the first column stand for send, drop and receive by the node_id in the third column. The seventh column shows the packet type where *cbr* is the data packet type in this case. From the trace results it can be seen that data packet (cbr) numbers 8 and 9 sent by node1 are dropped as a result of the hidden node. Initially node2 starts sending packet number 9 to node1. And then node0, which is hidden from node2 and is aware of this transmission, sends a packet, i.e., packet number 8 also to node1. They both collide at node1 and get dropped. Then both node0 and node1 try to retransmit the dropped packets. This can be avoided with the RTS/CTS handshake mechanism. We modified our simulation parameters by setting the RTS_Threshold back to a zero value, which means there will be always a RTS sent for any data packet transmission. The NS simulation trace file for this scenario shows that the RTS/CTS handshake mechanism solves the hidden node problem.




```
s   6.012349440  _1_  MAC  --- 0  RTS  44 [13ee 1 0 0]
r   6.012702257  _2_  MAC  --- 0  RTS  44 [13ee 1 0 0]
s   6.012712257  _2_  MAC  --- 0  CTS  38 [12b4 0 0 0]
r   6.013017074  _1_  MAC  --- 0  CTS  38 [12b4 0 0 0]
s   6.013027074  _1_  MAC  --- 4  tcp  1092 [13a 1 0 800]
r   6.017491891  _2_  MAC  --- 4  tcp  1040 [13a 1 0 800]
s   6.017501891  _2_  MAC  --- 0  ACK  38 [0 0 0 0]
r   6.017806708  _1_  MAC  --- 0  ACK  38 [0 0 0 0]
```

To summarize, the three nodes topology is simulated for three cases and the NS trace confirms the facts as predicted in table 3.3.

Simulation Conditions	Results
CSThresh_ = 2* RXThresh_ and No RTS-CTS control mechanism	No packet dropped
CSThresh_ = RXThresh_ and No RTS-CTS control mechanism	Packet dropped
CSThresh_ = RXThresh_ and With RTS-CTS control mechanism	No packet dropped (Remedy-- RTS/ CTS may be dropped)

Table 3.3 RTS/CTS solves the hidden node problem

3.3.1.2 Fairness issue

Fairness in wireless networks has been studied under various network scenarios, ranging from cellular networks and WLANs to ad hoc networks. An initial study on fairness in a packet cellular environment is presented by Lu et al. in [Lu 97], who first address the issue of fairness among wireless users, in presence of location-dependent channel capacity and errors. Although TCP and WLAN are both designed for fair capacity share, little is known about the behavior of TCP flows over WLAN. Therefore, the fairness issue is investigated.

We consider the following scenario: a WLAN where an Access Point (also referred to as Base Station or Hot Spots) provides access to web or a shared file system to n STAs (nodes or stations). At any instant of time, a STA keeps downloading (or, receiving) information via the AP. This file transfer is controlled using TCP. The TCP controlled data traffic flows in the downlink direction (from the AP to the stations) while the TCP acknowledgement traffic flows from the nodes to the AP. The AP and various stations use the IEEE 802.11 MAC protocol for transmission of their data (TCP DATA packets in case of BS and TCP ACK packets in case of nodes). In this section we assume the number of stations within the coverage of the AP is fixed.

1) A two-user situation

Firstly, we can make a simulation scenario in which there is one static user inside the AP's cell, while another mobile user moves across the cell with a fixed velocity of 2m/s. The data rate at the 802.11 MAC layer is 1M bps. We recorded the number of bits the station received per second. Just as expected, the users within the range of the base station share the throughput, that is, the total throughput in the system is the same. But for the individual user, its individual instant throughput does not just remain around some constant value. It fluctuates heavily, just as the figures below shows

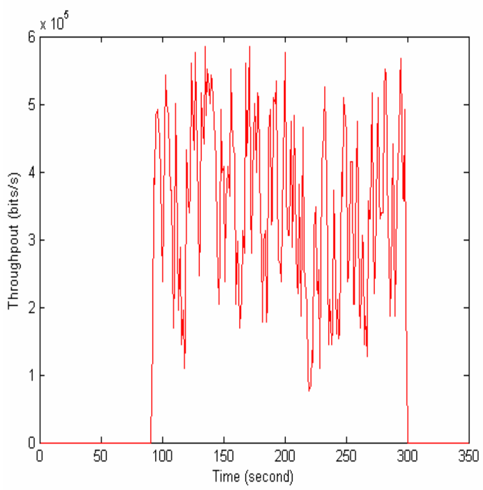


Figure 3.7 The mobile station's throughput as a function of time

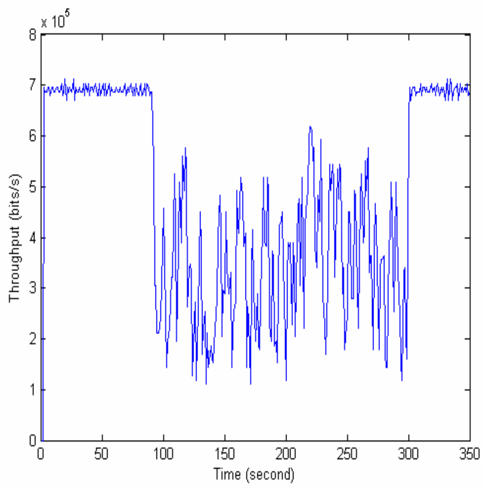


Figure 3.8 The static station's throughput as a function of time

As we can see in Figures 3.7 and 3.8, before the mobile user (node 0) moves in the hearing range of the AP, the static user (node 1) occupies the whole bandwidth of the AP. But as soon as the node 0 establishes the connection with the AP, the throughput of node1 decreases sharply. Then they just share the throughput which the AP can provide. Although the throughput of the individual users is not constant, the sum of these two users is constant as Figure 3.9 shows. But at the moment when mobile node0 moves out the range of the AP, the total throughput of the system decreases partly. We think this can be explained by the fact that when node0 is already out of the hearing range, the AP has no idea about this. The AP just keeps sending packets to node0, but all the packets are dropped by node0. Only after one or three RTT times (depending on the TCP mechanism), if the AP does not get any ACK from node0, it will terminate this connection. After that, node1 can utilize the whole throughput again.

Let us have a look at the time average behavior of two stations. We may calculate the average throughout of these two stations during the time when the mobile station and static station are both within the system, i.e., from 92 second to 299 seconds. From Figure 3.10 we can see that the average throughput the two individual users got is almost the same (43.572kbps vs. 42.915 kbps) and each them took half of the total amount of throughput.

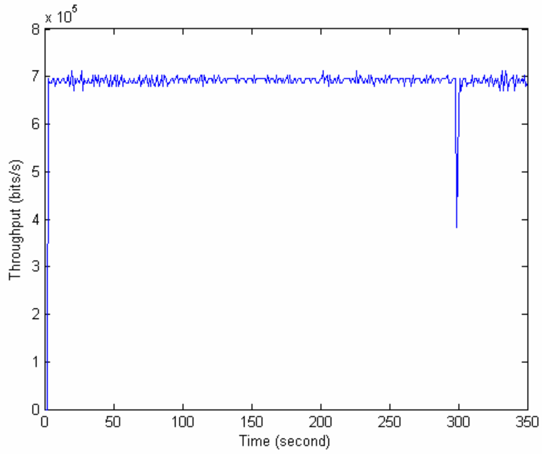


Figure 3.9 The total throughput

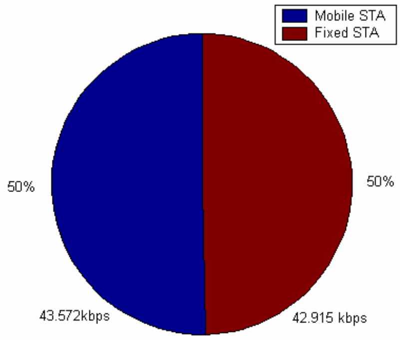


Figure 3.10 Fraction of mean throughput

2) A three-user situation

Without losing generality, we also simulated a three users scenario as follows. Two static users are settled inside the coverage of AP, and one mobile user moves across the cell with a velocity of 2m/s. Figure 3.11 shows the individual throughput of these three stations. As we have explained before, the users share the total throughput of the base station. From this figure we can see that, before mobile user (node0) moves in the hearing range of the AP, the other two static users (node1 and node2) share the whole bandwidth of the AP. After the mobile node comes inside the range of the AP, it occupies some fraction of the total throughput, and the throughput of the other two nodes decreases some degree. And after mobile node 0 moves out of the system, the throughput of node 1 and node2 increase again. The total throughput of this three-node system is almost constant, and looks similar to

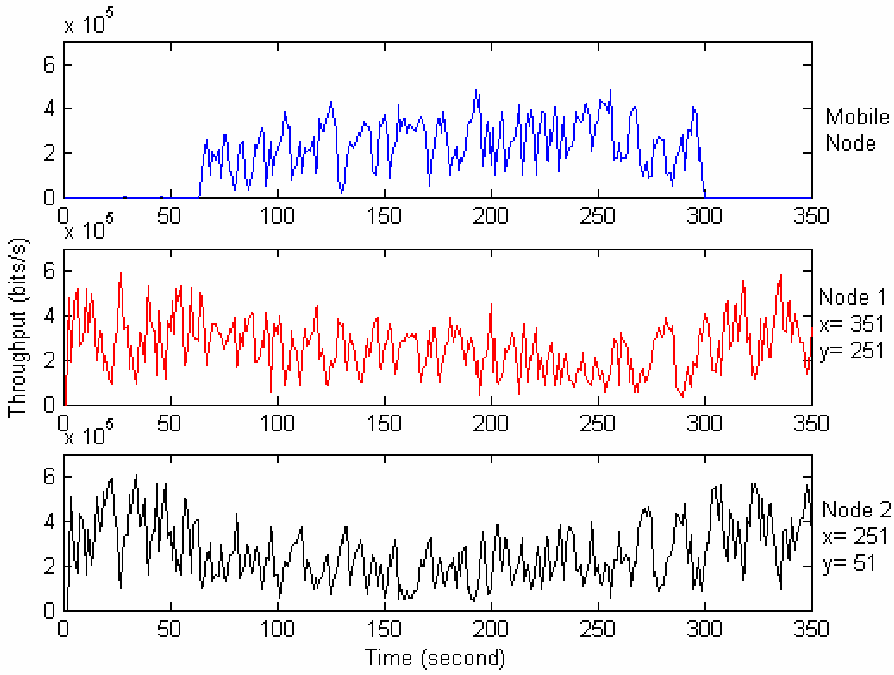


Figure 3.11 The three STAs' individual throughput as a function of time

Figure 3.9. But at the moment when the mobile node moves outside the hearing range, the whole system loses some amount of throughput due to the departure of this node.

Here we consider the average behavior of the nodes. To be more precise, we draw the following figure to show the sharing statures between the nodes. This is shown in Figure 3.12. Before the mobile node comes inside the hearing range of the AP, the other two static stations share the throughput. In theory the throughput will be equally divided by these two nodes, that is, 50% and 50%. But in practice, the percentage is 47% against 53%. After the mobile node moves inside the range, the three stations start to share the capacity. Then, when it moves out, only the other two stations share the capacity again. Note that these results are not perfect, this is only one run. We made three simulation runs and present the average throughput sharing in Figure 3.13. The average behavior shows the fair sharing between the users.

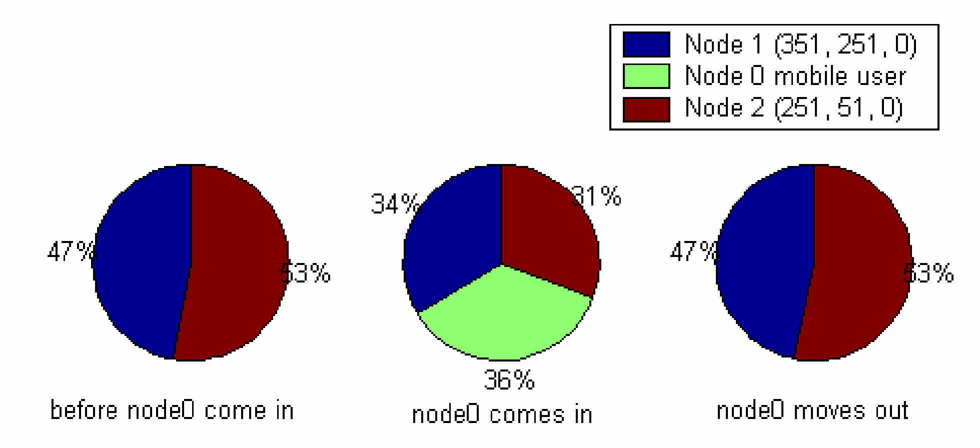


Figure 3.12 The throughput sharing fraction between the stations

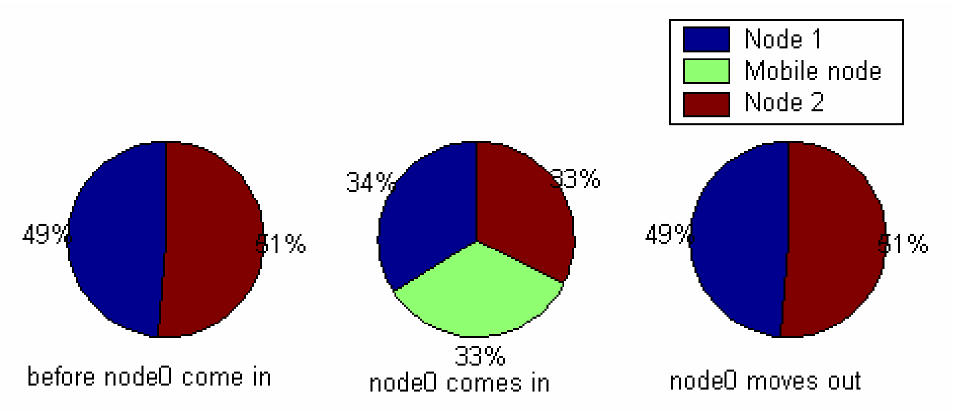


Figure 3.13 The average throughput sharing fraction

3) The random users position situation

To investigate the effect of the station's location, i.e., whether the station's throughput will change if the location of the station changes, we do the following simulations: one mobile station moves across the AP's coverage area, and one fixed node stays within the range. We change the location of the fixed station, in three different positions: (251, 51, 0), (351, 26, 0) and (351, 251, 0).

Figure 3.14 shows the instantaneous throughput of the mobile station. The shapes of the data from these three simulations look similar. As we have explained before, the aggregated throughput of all the two stations remains around a constant, as shown in Figure 3.15. When there is one user moving out of the base station's hearing range, the total throughput decreases by some amount which is allocated to that user (around half of the total amount).

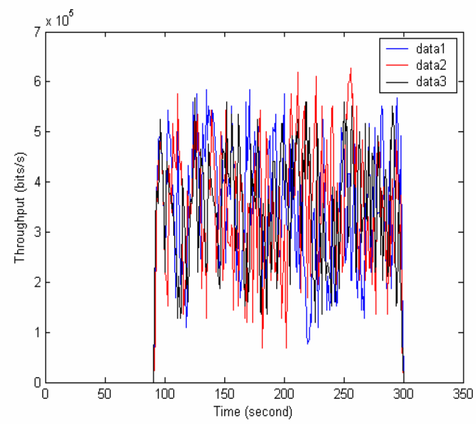


Figure 3.14 The mobile station's throughput

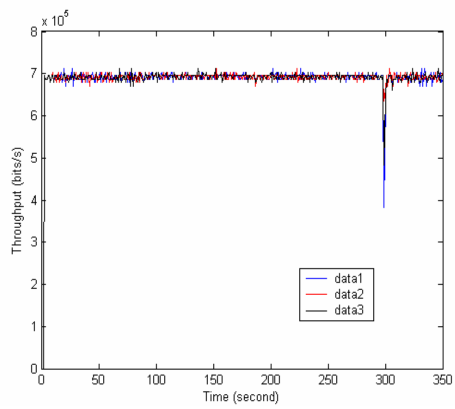


Figure 3.15 The total throughput of the AP

Now, let us have a look at the average behavior of the users. From Figure 3.16, we can see that the average throughput the two individual users got are almost the same and are the half of the total amount. In other words, the location of the stations within the system does not affect the throughput the stations get. The average throughput of the individual user shows the fair sharing property. We can say it is divided between the users fairly. That is, 802.11 DCF provides only long-term fairness, but suffers from short-term unfairness.

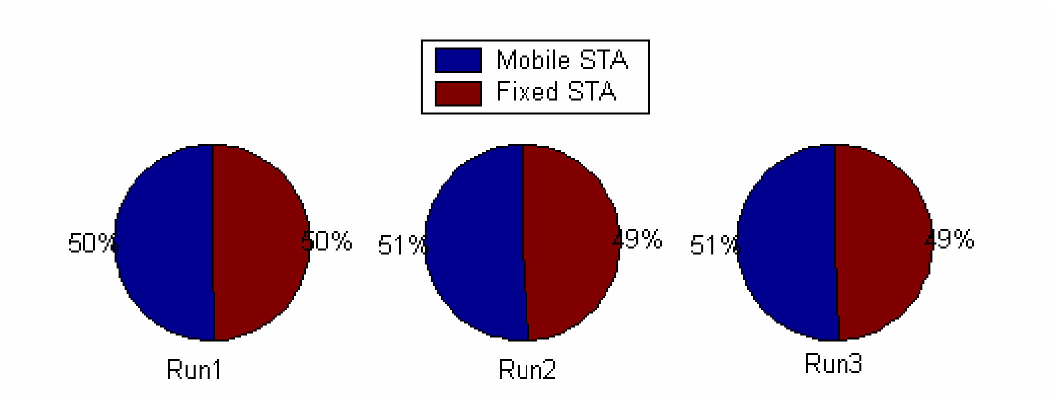


Figure 3.16 The average throughput sharing fraction

3.3.2 Modeling the multiple stations case

In this section, our goal is to investigate the average throughput (bits/s) of any number of stations when they cross the coverage of the AP. The analysis scenario in our research for the single cell case is shown in Figure 3.17. There is one straight highway, and one Access

Point is located *d* meters away from the main road. When the users drive on this road, they can access Internet services via this AP. We denote the coverage of the Access Point as *R* meters, and the stations' average velocity as *V* m/s.

A mobile station can be associated with only one AP at any given time. As soon as the station moves inside the AP's coverage range, it associates itself to the nearby Access Point. After an authentication and authorization process, the user can access the wireless medium to download information from the Web Server until he drives out of the range or the AP.

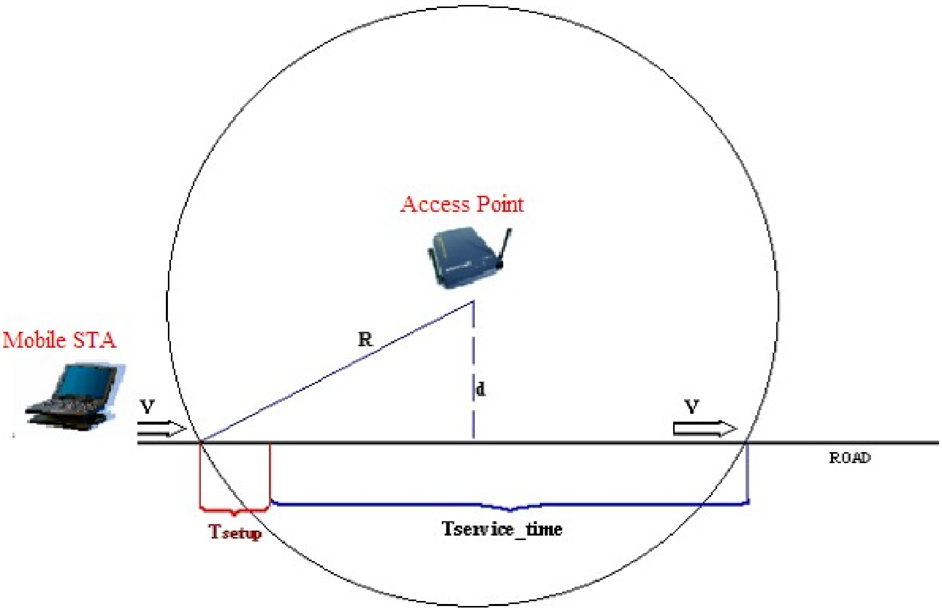


Figure 3.17 Analysis system model

1) Set up procedure modeling

In order to deliver a message within a Distributed System (DS), the distribution service (i.e., the web server) needs to know which AP to access for the given IEEE 802.11 STA. This information is provided to the DS by the concept of association. Before a STA is allowed to send a data message via an AP, it first becomes associated with the AP. The act of becoming associated invokes the association service, which provides the STA to AP mapping to the DS. At any given instant, a STA may be associated with no more than one AP. This ensures that the DS may determine a unique answer to the question, "Which AP is serving STA X?" Once an association is completed, a STA may make full use of a DS (via the AP) to communicate. Association is always initiated by the mobile STA, not by the AP. An AP may be associated with many STAs at one time. A STA learns what APs are present and then requests to establish an association by invoking the association service. The STA transmits an association request to an AP with which that STA is authenticated. Whenever an *Association Request* is received from a STA and the STA is authenticated, the AP shall transmit an A*ssociation Response* with a status code. If the status value is "successful," the Association ID assigned to the STA is included in the response. If the STA is not authenticated, the AP transmits a *Deauthentication* frame to the STA. If an *Association Response* is received with a status value of "successful," the STA is now associated with the AP and it sends a confirmation packet indicating the successful completion of the operation. When the AP receives the association response with a status value of "successful" which is acknowledged by the STA, the STA is considered to be associated with this AP [IEEE 99].

As the association procedure described above, we can model the Access Point server as a queuing system with infinitely many servers. Because, when the mobile user comes inside the range of the AP, the AP server is immediately available for each arriving user to start the association service, i.e., each user always finds a free server. Moreover, the distribution of the arrival of a user will not affect the service the user can obtain from the AP server and the service time of each user is independent. So we build the Setup service server as a $\cdot / G / \infty$ queuing server [Tijms 94]. Furthermore, we make the assumption that the average service time depends on the average velocity (V m/s) of the user. If the V is less than 1, the setup service time (T_{setup}) will be around 1 second. If V is around 2, T_{setup} is around 4s. If V is larger than 2 and less than 4, T_{setup} is around 5s. If V is larger than 5, T_{setup} is around 6s. The rules are listed in Table 3.4.

Velocity (m/s)	v ≤1	1< v ≤3	3< v ≤5	v >5
Setup Time (s)	1	4	5	6

Table 3.4 Setup Time selecting rules

2) The aggregated throughput (C)

The aggregated throughput, i.e., the available capacity for the TCP flows, is an important parameter in our model. In this section, we derive the aggregated throughput which the AP system can provide to all the stations within its hearing range.

Suppose there are N stations inside the system, i.e., there are N concurrent TCP connections at any given time. As we investigated in Section 3.3.1, the capacity of the system is shared among the users. In order to get the approximation of the aggregated throughput, we make the following assumption: a kind of round robin token passing between different connections takes place, i.e., the users obtain the service one by one. It is thus a collision free scenario. In such a case, the time needed for one packet transmission is

$$T_{packet} = N \cdot T_{single\_connection}, \qquad (3.10)$$

where $T_{single\_connection}$ can be obtained from Section 3.2.2 as

$$\begin{aligned} T_{single\_connection} &= T_{TCP\_data} + T_{TCP\_ack} + 2 \cdot \overline{T}_{backoff} \cdot T_{slot} \\ &= T_{TCP\_data} + T_{TCP\_ack} + cw_{\min} \cdot T_{slot} \quad . \end{aligned} \qquad (3.11)$$

Given there are N stations inside this system, for each station, the throughput is

$$TP(N) = \frac{L_{data}}{T_{packet}} = \frac{L_{data}}{N \cdot T_{single\_connection}} = \frac{L_{data}}{N \cdot (T_{TCP\_dtat} + T_{TCP\_ack} + cw_{\min} \cdot T_{slot})} . \qquad (3.12)$$

So, the aggregated throughput of the AP system is

$$\begin{aligned} \mathrm{C} = TP_{Aggregated} &= N \bullet TP(N) = N \bullet \frac{L_{data}}{N \cdot (T_{TCP\_data} + T_{TCP\_ack} + cw_{\min} \cdot T_{slot})} \\ &= \frac{L_{data}}{T_{TCP\_data} + T_{TCP\_ack} + cw_{\min} \cdot T_{slot}} . \end{aligned} \qquad (3.13)$$

As we can see formula 3.13 is the same as formula 3.8, so the aggregated throughput C can be calculated using the same formula as the one obtained from the single station scenario.

3) The Average Throughput (TP)

As explained before, we focus on exploring the impact of the mobility of the mobile station on the performance the station can gain, in terms of the average throughput (bits/s) of any number of stations when they cross by the coverage of the AP.

Usually, the arrival process of users in the Internet service systems can be modeled as Poisson process. In our work, we also adopt this method. Here we assume that the stations come inside the AP's coverage one be one; the inter-arrival times are independent and have a common exponential distribution with mean $1/\lambda$, i.e., the Poisson arrival rate of the user is λ.

After the mobile user has successfully been associated to the Access Point, he can access the Internet service to download information via the AP. Since the Access Point server is immediately available for each arriving user, we can say there are infinitely many servers available in the system, but the service rate (throughput) depends on the number of users inside the system. Here we also model the internet service system as an $M/G/\infty$ queuing system.

In theory, the time spent to cross the AP's coverage area can be calculated as the distance divided by the velocity of the station

$$T_{total} = \frac{2 \cdot \sqrt{R^2 - d^2}}{V}, \tag{3.14}$$

where R is the coverage of the Access Point; d is the distance between the AP and the main road, and V is the stations' average velocity.

So the average time when the station can get internet service via the Access Point can be given as

$$E(S) = T_{sojourn\_time} = T_{total} - T_{setup} = \frac{2 \cdot \sqrt{R^2 - d^2}}{V} - T_{setup}. \tag{3.15}$$

Since we only focus on the first moment of the sojourn time (average value), the variance of the speed does not matter at all. Although the speed of the user during the period he resides inside the AP cell may vary, we only care about the average equivalent velocity (V). That is an important insensitive property for this formula. Furthermore, based on the given coverage range R, the distance d, and the station velocity V, the total time T_{total} is determined. The setup time T_{setup} is also determined (as explained in the previous section), so in such a case, the sojourn time is constant for each station, denoted as $E(S)$.

As we investigated in Section 3.3.1.2, the average throughput of the individual user shows the fair sharing property, we can use the Processor Sharing (PS) model to analyze our system. Since the pioneering work by [Heyman 97], Processor Sharing based models have been proposed and studied in order to characterize the bandwidth sharing performed by the TCP protocol. The PS model was formally introduced and analyzed by Kleinrock [Kleinrock 76] as the limiting discipline obtained by letting the quota go to zero in a Round-Robin policy. In the PS model, all jobs present in the system obtain a fair share of the capacity, i.e., assuming a server with unit capacity, if at arbitrary time $t > 0$ there are N jobs in the system each job will be served at rate 1/N. Thus, suppose at any time t, a mobile station A observes that there are k (k= 0, 1, 2…) other users inside the system. Then according to the Process Sharing theory, the throughput that each user can get at time t is *C*/(*k*+1), where *C* is the capacity (aggregated throughput) of the system. So the average throughput one mobile station can obtain during the time when he cross the coverage of the AP can be calculated by

$$TP = \sum_{k=0}^{\infty} p_k \cdot \frac{C}{k+1}, \qquad (3.16)$$

where p_k is the probability of k users inside the system.

In order to get the expression of p_k, we can describe this M/G/∞ queuing system by selecting the birth-death coefficients as follows

$$\begin{cases} \lambda_k = \lambda & k = 0,1,2,\ldots, \\ \mu_k = k\mu & k = 1,2,3,\ldots, \end{cases}$$

where the service rate can be denoted as $\mu = \frac{1}{E(S)}$. Here the state-transition-rate diagram is shown in Figure 3.18.

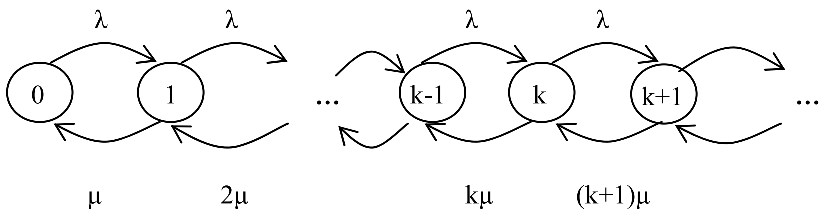


Figure 3.18: State- transition-rate diagram for the infinite-server case M/G/∞

We can easily establish the equilibrium difference equations for our system in the way explained in [Kleinrock 76]

$$\lambda p_{k-1} + (k+1)\mu p_{k+1} = (\lambda + k\mu) p_k .$$

After some simple calculus, we can obtain the steady state probability of k users in the system by

$$p_k = e^{-\lambda \mathrm{E}(s)} \frac{[\lambda \mathrm{E}(S)]^k}{k!} \qquad k = 0, 1, 2, \ldots \quad . \qquad (3.17)$$

So the expression of the average throughput TP can be written as

$$TP = \sum_{k=0}^{\infty} p_k \cdot \frac{C}{k+1} = \sum_{k=0}^{\infty} e^{-\lambda E(S)} \cdot \frac{[\lambda E(S)]^k}{k!} \cdot \frac{C}{k+1} = e^{-\lambda E(S)} \cdot C \cdot \sum_{k=0}^{\infty} \frac{[\lambda E(S)]^k}{(k+1)!} .$$

Denote $\lambda E(S)$ as γ, we have:

$$TP = \frac{e^{-\gamma} \cdot C}{\gamma} \cdot \sum_{k=0}^{\infty} \frac{\gamma^{k+1}}{(k+1)!} \ .$$

Then we denote k+1 as m, since k =0, 1, 2, …, m = 1, 2, … thus we have

$$\begin{aligned} TP &= \frac{e^{-\gamma} \cdot C}{\gamma} \cdot \sum_{m=1}^{\infty} \frac{\gamma^m}{m!} = \frac{e^{-\gamma} \cdot C}{\gamma} \cdot (\sum_{m=0}^{\infty} \frac{\gamma^m}{m!} - 1) \\ &= \frac{e^{-\gamma} \cdot C}{\gamma} \cdot (e^{\gamma} - 1) = \frac{C}{\gamma} \cdot (1 - e^{-\gamma}) \\ &= \frac{C}{\lambda E(S)} \cdot (1 - e^{-\lambda E(S)}) \ . \end{aligned} \tag{3.18}$$

where $E(S)$ can be obtained from Equation 3.15 and C can be obtained from Equation 3.13.

Another important performance metric in our research is the total amount of data, denoted as $E(B)$ in bits, which the mobile user can receive during the period he across the coverage of the AP. We can easily get the expression of $E(B)$ as the average throughput (TP bits/s) multiplied by the total amount of time the mobile user stays inside the system ($E(S)$ seconds)

$$\begin{aligned} E(B) &= E(S) \cdot TP = E(S) \cdot \frac{C}{\lambda E(S)} \cdot (1 - e^{-\lambda E(S)}) \\ &= \frac{C}{\lambda} \cdot (1 - e^{-\lambda E(S)}) \ . \end{aligned} \tag{3.19}$$

Notably, the expected performance metrics (the average throughput and the total amount of received data) only depend on the first moment of the service time $E(S)$, which shows the so-called *insensitivity* property.

3.3.4 Simulation validation

To validate the model, we have run some simulations with the NS-2 2.26 simulator [NS-2]. The scenario used is the same as shown in Figure 3.2, but here we generated in total 100 stations, and the inter-arrival time of these stations is exponentially distributed with mean value 10 seconds. The values of the parameters used to obtain numerical results, for both the analytical model and the simulation runs, are summarized in Table 3.5. Other parameters are the same as in Table 3.1. The AP is located 50 m away from the road, and the hearing range of the AP is still set to 250 m. The stations drive across the road with constant velocity. In each simulation run, the mobile stations keep the same velocity. For different simulation runs, the users may have different velocities. The setup time adopts the value which is shown in the Table 3.4.

Parameter	Value	Parameter	Value
Access Point Coverage R	250 m	AP distance (d)	50 m
Data Rate	2Mbps	Poisson Arrival Rate	0.1 user/second
Station velocity	2m/s	Setup Time	4 seconds

Table 3.5 The parameters for simulations

1) The aggregated throughput C validation

First we investigate the aggregated throughput of the system from a single run and from multiple simulation runs. The simulation results from one single run (v = 2m/s) are shown in Figure 3.19. From this figure, we can see that at the beginning of the curve, the aggregated throughput remains almost constant. But after around 245 seconds, the curve becomes fluctuant, which is caused by the departure of the user as we explained before. When a STA is already out of the hearing range, the Access Point has no idea about this, and it just keeps sending packets to the STA, but all the packets are dropped by STA. Only after one or three RTT times, if the AP does not get any ACK from the STA, it will terminate this connection. After that, the other stations can share the whole throughput

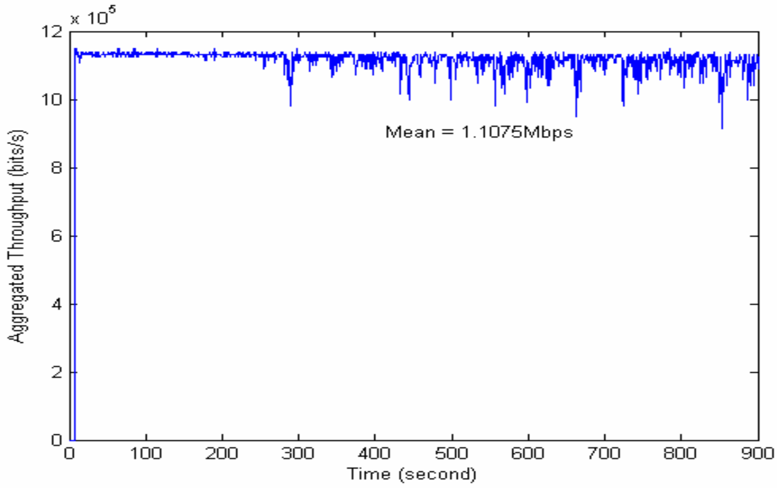


Figure 3.19 The aggregated throughput from one run

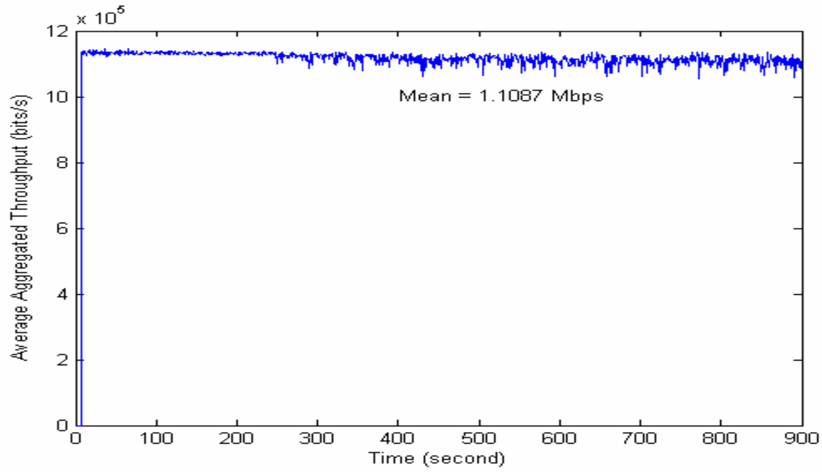


Figure 3.20 The average aggregated throughput of multiple runs

again. Figure 3.20 shows the average aggregated throughput from the four data sets of different simulation runs. We can see clearly that the curve becomes smoother at the tail. So we can see the average aggregated throughput of the whole AP system remains constant at any time. Furthermore, the mean simulation value of the aggregated throughput is 1.1085Mbps, while the analysis results we got from Section 3.2 is 1.1 Mbps. The difference between them is 0.77%, so we think our analysis model for the aggregated throughput is sufficiently accurate.

Secondly, we investigate the average aggregated throughput of the mobile station with different average velocities. The results of v = 2, 5, 8,10,15,20 are shown in Table 3.4. From the table we can see that the aggregated throughput decreases as the average velocity of the mobile user increases. To analyze this behavior, we plot the instant aggregated throughput (observed at every second) for three cases v = 2, 5, 10 in Figure 3.21. The length of the road which is inside the range of the AP is fixed. If the velocity of the mobile user becomes larger, the time he needs to cross that road will be smaller. During the time he stays inside the AP cell, there will be fewer users on average. At the moment when the mobile station moves out of the range of the AP, his part of the throughput will be wasted. So when there are fewer users inside the system, the throughput each user can get is larger. Thus the part of wasted system capacity will be larger. That is why from Figure 3.12, we can see that the curve fluctuates heavier as the velocity of the user becomes larger. In the same way, since the wasted throughput is larger as the velocity becomes larger, the average aggregated throughput will be smaller.

The average value of these six results in Table 3.6 is 1.0854 Mbps and the 95% confidence intervals are [1.0681, 1.1027]. Our expected mean is 1.1 Mbps, which lies within the 95% confidence intervals. We therefore conclude that our model's prediction is accurate enough.

Velocity m/s	2	5	8	10	15	20
Mean (Mbps)	1.1083	1.0979	1.0881	1.0822	1.0732	1.0628

Table 3.6 The average aggregated throughput of different velocities

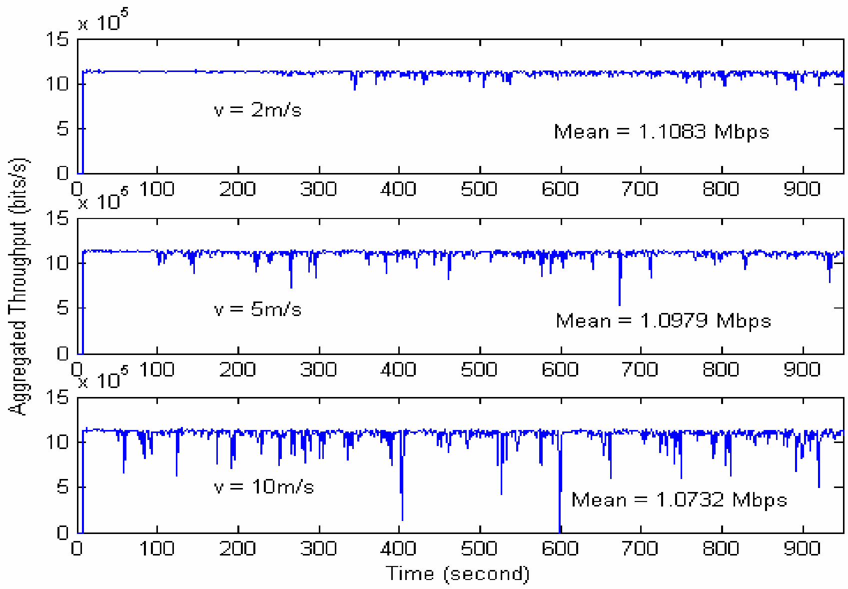


Figure 3.21 The average aggregated throughput of different velocities

2) The setup time & the service time

In our model, we assume that the setup time needed for the association and authentication of the stations is constant, which depends on the average velocity of the mobile station. With a given coverage range R, the AP distance d, and the station velocity V, the total time T_{total} is determined, and the setup time T_{setup} is also constant. The sojourn time $T_{sojourn}$ is thus also determined. We also investigate the setup time and service time spent in the simulation. First we investigate the setup time and sojourn time for different individual STAs (around 60 stations) with an average velocity of 2m/s. The simulation results are shown in Figure 3.22 and Figure 3.23. In total there are four simulation runs. For each simulation run, we have calculated the setup time for each station, which is shown in four figures separately.

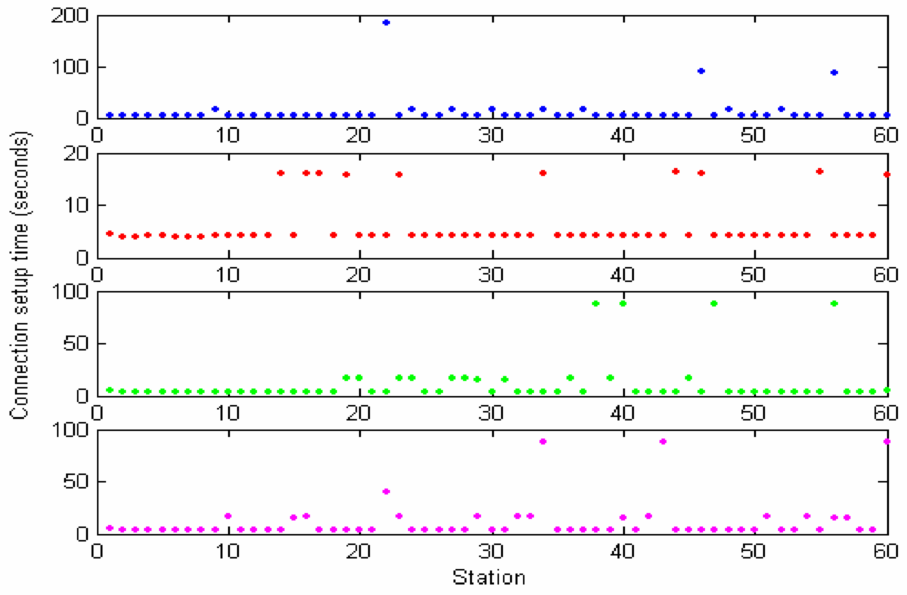


Figure 3.22 Setup time of each individual station

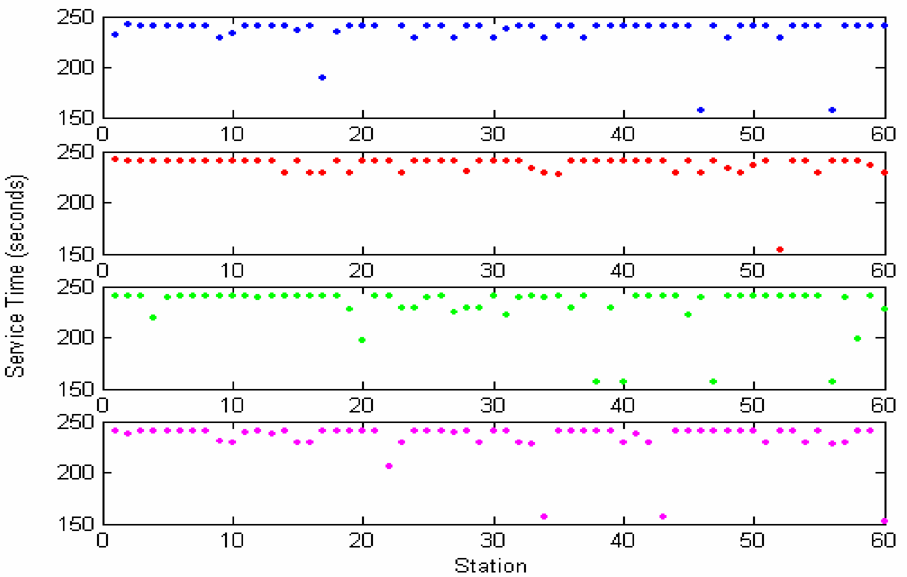


Figure 3.23 Sojourn time of each individual station

We can clearly see from the figures that most of the setup time remains at a constant level. We did some data statistics: 80% of the data falls in interval [3.5, 4.5], 15% in [15.8, 16.3], and 5% in [40, 184]. If we ignore the minority of outliers, we can say the setup time is constant for each station around 4 seconds. Furthermore, the service times of the stations is also the same situation as the setup time: around 80% of the data falls in interval [230, 241], 15% in [221,230] and 5% in [150, 219].

Moreover, we also investigate the setup times of different velocities, which are shown in Figure 3.24. We can see that for each velocity, the setup time almost remains constant at some level. So we think our assumption about the constant value of setup time and service time is reasonable.

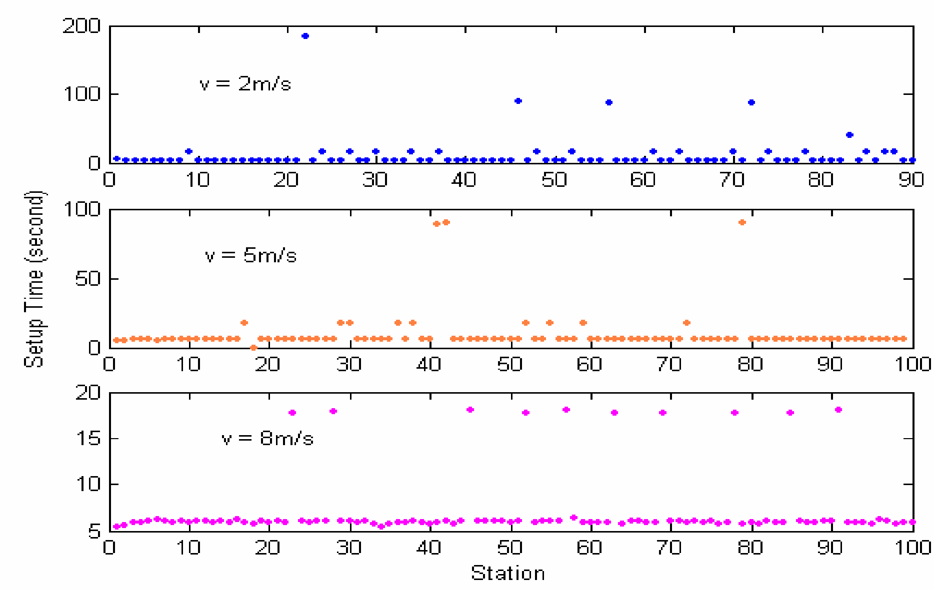


Figure 3.24 Setup times of each individual station with different velocities

Furthermore, we also made the assumption that the average setup time depends on the average velocity of the mobile stations, and the value of the setup time for different velocities is shown in Table 3.6. To validate this assumption, we run a few simulations for different velocities from 2 to 80 and calculate the average setup time of the 100 stations. The numerical results are shown in Table 3.7. And the curve of the setup time as a function of the average velocity is shown in Figure 3.25. Compare the numerical results and the model assumptions, when v is equal to 2m/s, the setup time of the simulation is 4.19 seconds, while the model value is 4 seconds. And after v is larger than 5, all of the setup times are almost around 6 seconds. Especially, the tail of the curve in Figure 3.25 is almost constant around 6 seconds. So we think our assumption about the setup times is reasonable.

	2	**4**	**5**	**6**	**8**	**10**	**15**
time	4.1933	5.3883	5.6214	5.7117	5.927	6.0249	6.0418
	20	**30**	**40**	**50**	**60**	**70**	**80**
time	5.9587	5.9659	5.9634	6.0495	5.942	5.9517	5.9443

Table 3.7 The simulation results of average setup time for different velocities

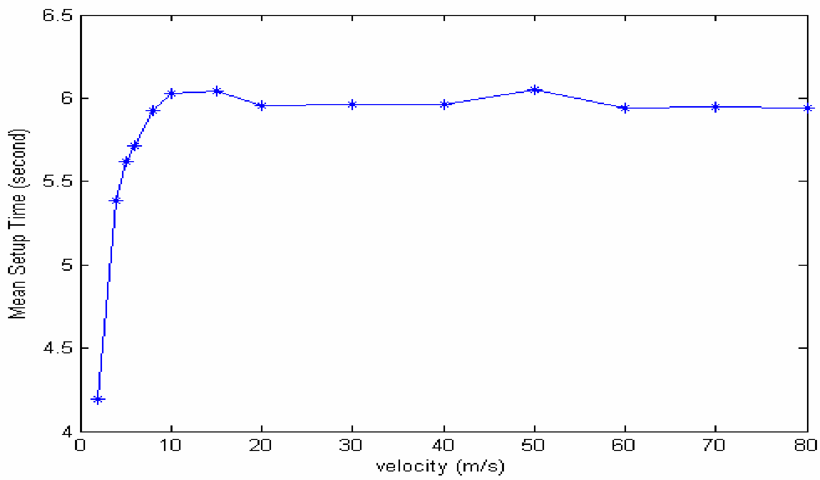


Figure 3.25 The setup time as a function of the average velocity

3) The average individual throughput

To evaluate our expressions for the average throughput and the total amount of received data, we did 13 simulations runs with different velocities. In each simulation run, all the stations had the same velocity and arrived in the system according to a Poisson process. Figure 3.26 show the average throughput of each station with v = 2, 5, and 10 m/s. Since the system starts empty, at the beginning there will be a system warm-up period. When the first station comes inside the system, he occupies all the capacity, and then after another station arrives, they will share the capacity, which causes the high average throughput of the first few stations. But as more stations cross the system, the system will tend to saturation. That is why at the tail of the curves the data fluctuates around some constant value.

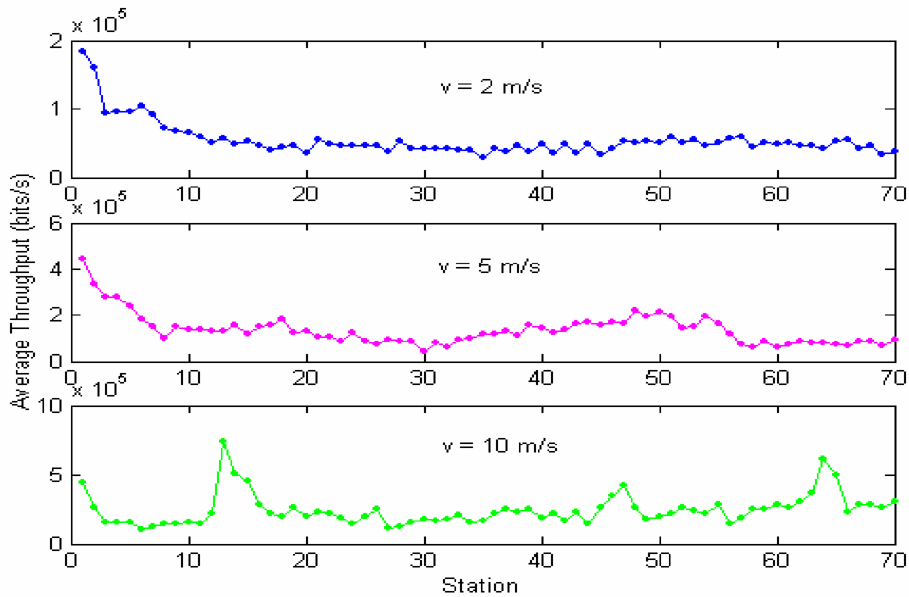


Figure3.26 Average throughput for each individual station with different velocities

We count the total number of received bits and then calculate the average individual throughput as the total number of bits the station received divided by the total time when it receives a packet (also known as sojourn time). But we discard the data we got from the stations that are in the warm-up period. The analysis results are calculated from formulas 3.18 and 3.19 and the values of the parameters used are summarized in Table 3.1, 3.4 and 3.5. The results of the13 simulation runs and the model prediction curve are drawn in Figure 3.27. The comparison of the numerical results from the model and simulation results are shown in Table 3.8.

Velocity (m/s)	Model (bps)	Simulation (bps)	Difference	95% confidence Intervals of simulation results
80 m/s	1.0932e+006	1.1181e+006	2.2270%	[1.0769, 1.1506] e+006
70 m/s	1.0469e+006	1.0631e+006	1.5238%	[1.0073 , 1.1190] e+006
60 m/s	0.9891e+006	1.0040e+006	1.4841%	[0.9516, 1.0664] e+006
50 m/s	9.1523e+005	9.1379e+005	0.1576%	[8.6186, 9.6572] e+005
40 m/s	8.1803e+005	8.0542e+005	1.5656%	[7.5181, 8.5904] e+005
30 m/s	6.8584e+005	6.5352e+005	4.9455%	[6.0514, 7.0191] e+005
20m/s	5.0119e+005	5.0747e+005	1.2375%	[4.3889, 5.7605] e+005
15m/s	3.8392e+005	3.7754e+005	1.6899%	[3.3418, 4.2090] e+005
10m/s	2.5240e+005	2.4930e+005	1.2435%	[2.2596, 2.7264] e+005
8m/s	1.9835e+005	1.9545e+005	1.4838%	[1.7356 , 2.0747] e+005
5 m/s	1.1829e+005	1.1461e+005	3.2109%	[1.0584, 1.2337] e+005
4m/s	8.7974e+004	8.9017e+004	1.1717%	[8.5219, 9.2815] e+004
2 m/s	4.5653e+004	4.550e+004	0.3363%	[4.325 , 4.808] e+004

Table 3.8 The simulation results vs. the model prediction of the average throughput

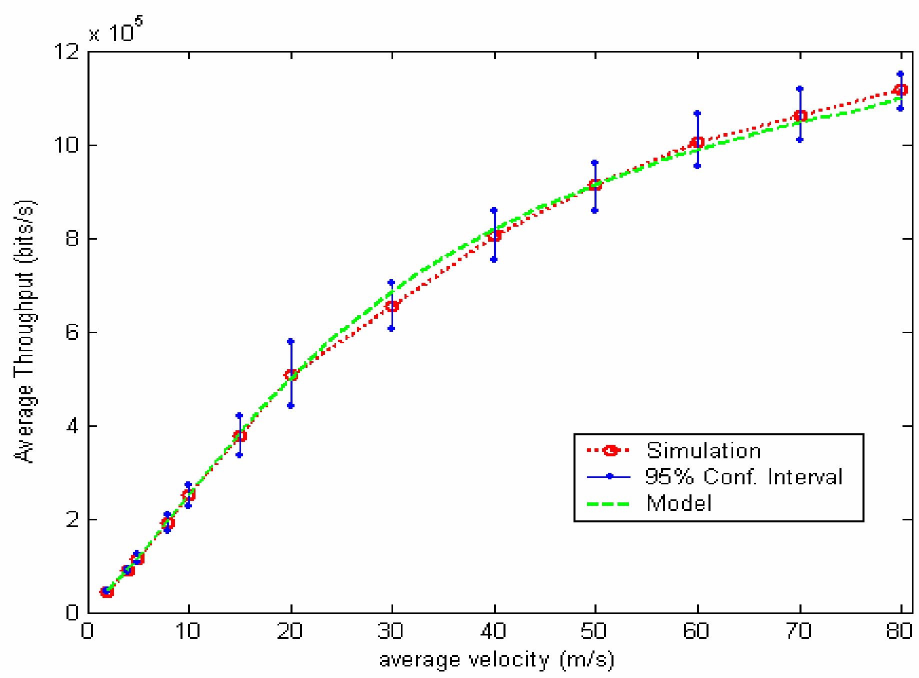


Figure 3.27 Model and simulation's average throughput as a function of the velocity

From Table 3.8, we can see that the differences between the simulation results and the model prediction values are all less than 5%. From Figure 3.27, we can see that the model's curve almost covers the simulation sample data and all lie inside the 95% confidence intervals. So we can say that our model can predict the average throughput for different velocities accurately.

4) Total amount of received data

Another important performance metric in our research is the total amount of received data in bits. We recorded all the packets the stations received during their movement inside the coverage of AP. The analytical results are calculated from formula 3.19. The simulation data compared with the model's analytical value are shown in Figure 3.28. It can be seen clearly from Figure 3.28 that in most cases, our proposed model can predict the performance metric accurately. The curve from our approximation model is quite close to the simulation data; furthermore, they all lie within the 95% confidence intervals of the simulation results.

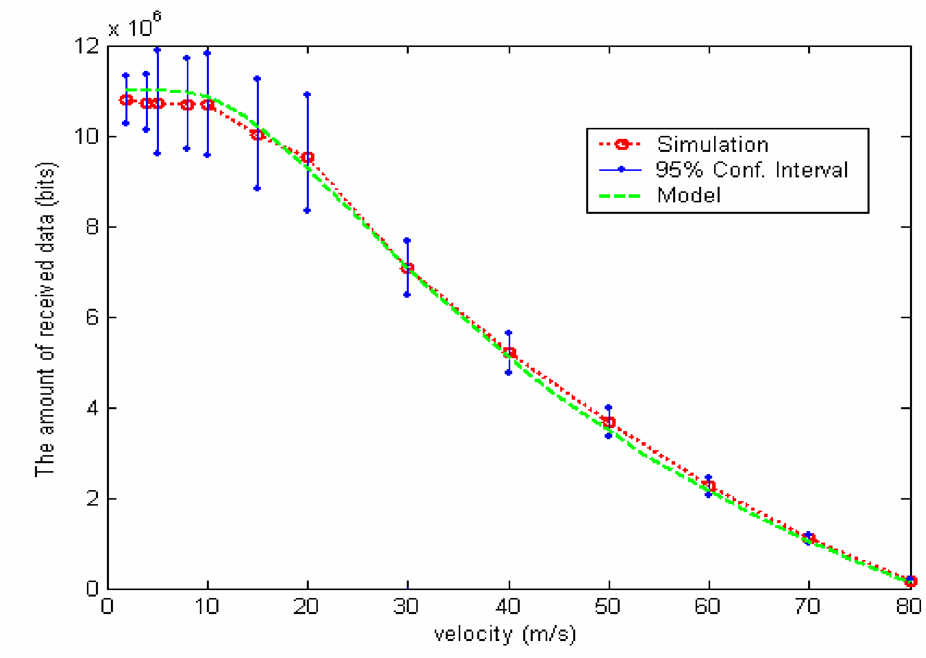


Figure 3.28 The amount of received data as a function of the velocity

3.4 Conclusions

In this chapter, we have investigated the performance of a single class user across a single AP cell.

To start, we only considered one very simplified scenario: a single mobile station drives across the coverage of one single AP cell. We did some simulations to understand the interaction between the TCP and MAC layer. Then we analyzed the network performance of an 802.11-based WLAN in presence of a single TCP connection. To simplify the complexity, for the physical radio propagation we took the Two-Ray Ground model. A reasonable assumption is made that the channel is loss free during the period when the STA remains inside the hearing range of the Access Point. According to the data transmission procedure of the RTS/CTS access mechanism and the 802.11 data format, we derived the throughput for a single TCP connection in RTS/CTS mode. To validate the model, some simulations run with the NS-2 2.26 simulator showed that the analytical results practically coincide with the simulation resutls.

To understand the effect between the different user flows, we first did some simulations. The NS simulation trace file showed that the RTS/CTS handshake mechanism solves the hidden node problem and that no packets are dropped. The simulation also shows that 802.11 DCF provides only long-term fairness, but suffers from short-term unfairness. Furthermore, the location of the stations within the system does not affect the throughput the stations get. We modeled the single AP cell as two $M / G / \infty$ queuing servers in tandem. One is the setup server, in which the service time is fixed according to the average velocity of the users. Another one is the Internet service server, in which the user sojourn time is fixed depending on the range of the AP, the average velocity. Based on findings of these simulations and using the results of the previous step, we derived the average throughput (and the total amount of received data) of any number of stations with the same average velocity when they cross the coverage of the AP. The expressions of the performance are thoroughly validated by the simulations in NS2. The results of the simulations proved our assumptions about the setup time and aggregated throughput reasonable. Moreover, the results show that our model can predict the average throughput and the total amount of received data accurately.

Chapter 4
The Multi-class single cell model

4.1 Introduction

In Chapter 3, we have developed and validated an analytical model of the performance metrics experienced by one single class of mobile users inside the system. All the users have the same average velocity when they move across the coverage of the AP. However, in real life situations there is always more than one class of users. People can access the internet when they are walking with their PDA, using their laptops while driving a car, etc. In this chapter, we consider multi-class users in the single cell scenario, that is, the users drive through one AP cell with different average velocities. The expressions of the performance are derived in Section 4.2. Simulations are run for validating the models in Section 4.3. After that, the optimization of our model is analyzed in Section 4.4.

4.2 Model description and analysis

As we explained in Chapter 3, the AP first provides the association and authentication service to the mobile users. After the successful association, the AP can provide the internet service to the users. Thus, we model the Access Point as two servers in tandem. When mobile users come inside the range of the AP, it will be immediately available for them by starting the association service. In other words, each user always finds a free server and after the association the Access Point server is immediately available for internet services. We define an infinite number of servers available in the system. In the same way as in Chapter 3, we build the AP server as two $M/G/\infty$ queuing servers in tandem, as shown in Figure 4.1.

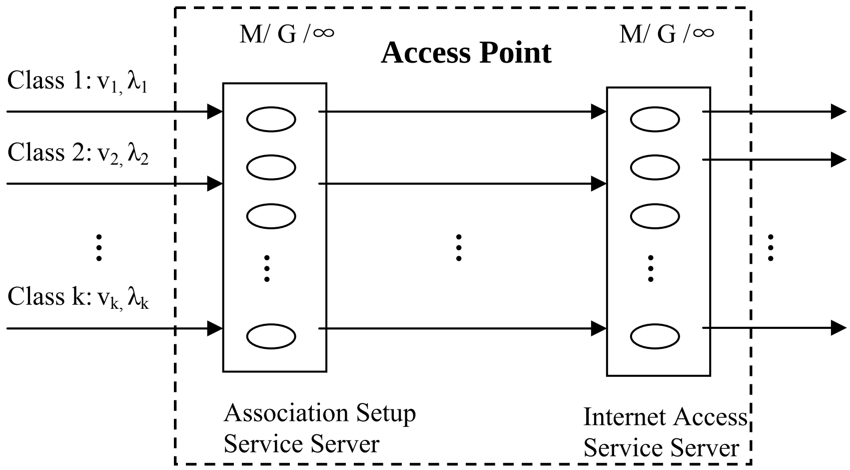


Figure 4.1 Single access point cell model

The difference between the scenario we analyze here and the one of the previous chapter, is that in the latter there are different classes of mobile users which drive with different average velocities, while in the former there is only one class of mobile users which drive with the same average velocity. Suppose there are k classes of mobile users. Users of different classes arrive according to mutually independent Poisson processes. The arrival rate for class i is λ_i, $i= 1, \cdots, k$. The average velocity of the users for class i is v_i, $i = 1, \cdots, k$. So in theory, the mean time the mobile user of class i will be inside the AP cell will be

$$T_{total\_i} = \frac{2 \cdot \sqrt{R^2 - d^2}}{v_i} \text{ , } i = 1, \cdots, k \text{ ,} \tag{4.1}$$

where R is the coverage of the Access Point, d the distance between the AP and the main road, and v the stations' average velocity.

As we analyzed in Section 3.3.2, the service time of the association server will depend on the velocity which follows the rule listed in Table 3.4. We denote this association service time for class i as $E(U_i)$, $i= 1, \cdots, k$. Thus the average sojourn time the mobile will stay inside the Internet Access Service server will be,

$$E(S_i) = \frac{2 \cdot \sqrt{R^2 - d^2}}{v_i} - E(U_i) \text{ , } i = 1, \cdots, k. \tag{4.2}$$

As we explained before, based on the given coverage range R, the distance d, and the station velocity v_i, the total time T_{total_i} is determined. The setup time T_{setup} is also determined, so in such a case, $E(S_i)$ is determined for each station.

Since each class of users is a Poisson process, the number of users of this class will follow a Poisson distribution, which has been analyzed in Section 3.3.2. Here we rewrite formula 3.17 for multiple classes. The probability of n_i users for class i ($i= 1, \cdots, k$) in the system is

$$\Pr(N_i = n_i) = e^{-\lambda_i \mathrm{E}(S_i)} \frac{[\lambda_i \mathrm{E}(S_i)]^{n_i}}{n_i!} \qquad n_i = 0, 1, 2, \ldots, \quad i = 1, 2, \ldots, k \text{ .} \tag{4.3}$$

From the whole system's point of view, the probability of users is a joint probability from these k classes of users. Since we assumed before that all users of different classes arrive according to mutually independent Poisson processes, the probability can be calculated by

$$\Pr(N_1 = n_1, N_2 = n_2, \cdots, N_k = n_k) = \Pr(n_1) \cdot \Pr(n_2) \cdots \Pr(n_k) \text{ .} \tag{4.4}$$

Suppose at an arbitrary time t, mobile station A observes the system to see how many other users are inside the system. Suppose there are n_i users for class i ($i= 1, \cdots, k$) inside the system. Then according to the Process Sharing theory, the throughput that each user can get at time t is $\frac{C}{\sum_{i=1}^{k} n_i + 1}$, where C is the capacity (aggregated throughput) of the system, which can be calculated from Equation 3.13. So the average throughput a mobile station can obtain during the time when he crosses the coverage of the AP is given by

$$TP = \sum_{n_1=0}^{\infty} \sum_{n_2=0}^{\infty} \cdots \sum_{n_k=0}^{\infty} \mathrm{P}(N_1 = n_1, \cdots, N_k = n_k) \cdot \frac{C}{\sum_{i=1}^{k} n_i + 1}$$

$$= \sum_{n_1=0}^{\infty} \sum_{n_2=0}^{\infty} \cdots \sum_{n_k=0}^{\infty} \mathrm{P}(N_1 = n_1) \cdot \cdots \cdot \mathrm{P}(N_k = n_k) \cdot \frac{C}{\sum_{i=1}^{k} n_i + 1}. \tag{4.5}$$

For the Processor Sharing model, only the total number of users of all the classes determines the throughput obtained by each user. Here we should focus on the sum of the number of users of different classes inside the system, i.e., $\sum_{i=1}^{k} n_i$. The expression of TP can then be rewritten as

$$TP = \sum_{m=0}^{\infty} \mathrm{P}(\sum_{i=1}^{k} n_i = m) \cdot \frac{C}{m+1}. \tag{4.6}$$

To get the closed-form of the throughput expression, we hereby use the superposition property of Poisson Processes: the superposition of two independent Poisson processes with rate λ_1 and λ_2 is a Poisson process with rate $\lambda_1 + \lambda_2$. That is: $Ps(\lambda_1) + Ps(\lambda_2) = Ps(\lambda_1 + \lambda_2)$. So the probability that there are in total m users inside the system, regardless of how many users for each class there are , is

$$\mathrm{P}(m) = e^{-\sum_{i=1}^{k} \lambda_i \mathrm{E}(S_i)} \cdot \frac{\left(\sum_{i=1}^{k} \lambda_i \mathrm{E}(S_i)\right)^m}{m!} \qquad m = 0, 1, 2, \ldots. \tag{4.7}$$

Defining $\gamma_i = \lambda_i E(S_i)$, we get

$$\mathrm{P}(m) = e^{-\sum_{i=1}^{k} \gamma_i} \cdot \frac{\left(\sum_{i=1}^{k} \gamma_i\right)^m}{m!} \qquad m = 0, 1, 2, \ldots . \tag{4.8}$$

The expression of the TP is then

$$TP = \sum_{m=0}^{\infty} \mathrm{P}(\sum_{i=1}^{k} n_i = m) \cdot \frac{C}{m+1} = \sum_{m=0}^{\infty} e^{-\sum_{i=1}^{k} \gamma_i} \cdot \frac{\left(\sum_{i=1}^{k} \gamma_i\right)^m}{m!} \cdot \frac{C}{m+1}. \tag{4.9}$$

Define $\gamma = \sum_{i=1}^{k} \gamma_i$. After the same derivation process as Formula 3.18, we can get the simple closed-form expression of the average throughput for the situation of the multiple classes as

$$TP = \frac{C}{\gamma} \cdot (1 - e^{-\gamma}) = \frac{C}{\sum_{i=1}^{k} \lambda_i \mathrm{E}(S_i)} \cdot \left(1 - e^{-\sum_{i=1}^{k} \lambda_i \mathrm{E}(S_i)} \right). \quad (4.10)$$

The Access Point Server is a non-priority server, i.e., it treats each class of users the same, so the average throughput the user can get within each class will be the same, which is present in our expression of TP.

Another important performance metric, the total amount of received data for class *i* denoted by $E(B_i)$ (i= 1, ⋯, k) in bits. We can easily get the expression of $E(B_i)$ as the average throughput (TP bits/s) multiplied by the total amount of time when the mobile user can stay inside the system ($E(S_i)$ seconds)

$$\begin{aligned} E(B) &= E(S_i) \cdot TP = E(S_i) \cdot \frac{C}{\gamma} \cdot (1 - e^{-\gamma}) \\ &= E(S_i) \cdot \frac{C}{\sum_{i=1}^{k} \lambda_i \mathrm{E}(S_i)} \cdot \left(1 - e^{-\sum_{i=1}^{k} \lambda_i \mathrm{E}(S_i)} \right). \end{aligned} \quad (4.11)$$

Notably, the expected performance metrics (the average throughput and the total amount of received data) only depend on the first moment of the service time $E(S_i)$ of each class.

4.3 Simulation validation

To validate our multiple-class model, simulations have been run for different velocity combinations of two classes. The simulation scenario is similar to Figure 3.2. Instead, there are two classes of users that drive across the coverage of the AP independently. The parameters used for both the analytical model and the simulation runs are shown in Table 4.1. Others physical and MAC layer parameters are identical to those shown in Table 3.1.

Parameter	Value	Parameter	Value
Access Point Coverage	250 m	AP distance (d)	50 m
Data Rate	2Mbps	Station velocity (m/s)	{2, 8,10, 20}
Arrival Rate of class 1	0.1 user/second	Arrival Rate of class 2	0.05 user/second

Table 4.1 The parameters for simulations and numerical results of the multiple-class case

1) The aggregated throughput C validation
In our two classes' scenario, although there are two independent Poisson process, from the point of view of the system there is one superposed Poisson process. First we investigate the aggregated throughput of the system to see whether our model's assumption is

reasonable or not. The results are shown in Table 4.2. From the four group results, we can see that the aggregated throughput decreases a bit as the velocity of the one class increases. To analyze this behavior, we plotted the aggregated throughput of three groups. As shown in Figure 4.2, the curves become more fluctuant, as the velocity increases, which is caused by the departure of the users. As we explained before, the larger the velocity is, the smaller the number of users is in the system, thus the larger part of the system capacity will be wasted when a user leaves the system. In our model, we adopt the value 1.1 Mbps as the system capacity. Compared to the simulation results in Table 4.2, the largest difference between the model prediction and the simulation results is only 2.87%. We thus believe our assumption is reasonable for this model.

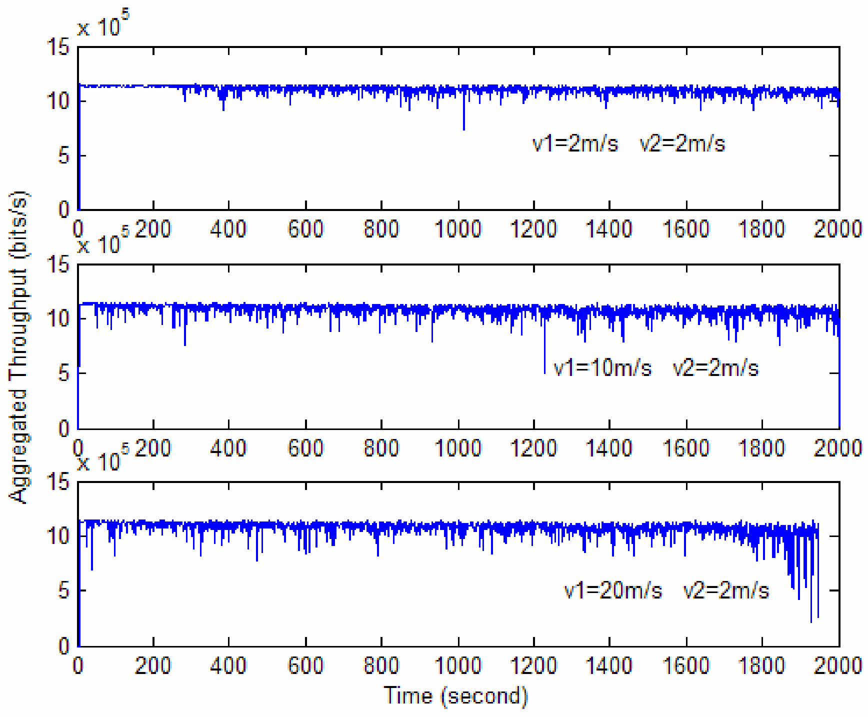


Figure 4.2 The instant aggregated throughput of two classes

Group ID	A	B	C	D
V1 m/s	2	8	10	20
V2 m/s	2	2	2	2
C (Mbits/s)	1.0985	1.0834	1.0782	1.0693

Table 4.2 The aggregated throughput of the two classes' case

2) The setup time

For this multiple classes' model, we still assume that the setup time needed for the association and authentication of the stations is constant, which depends on the average velocity of the mobile station of that class, which is shown in Table 3.4. To validate this assumption, we ran a few simulations for different combinations of velocities from 2 to 20 m/s, to calculate the average setup time of the 300 stations.

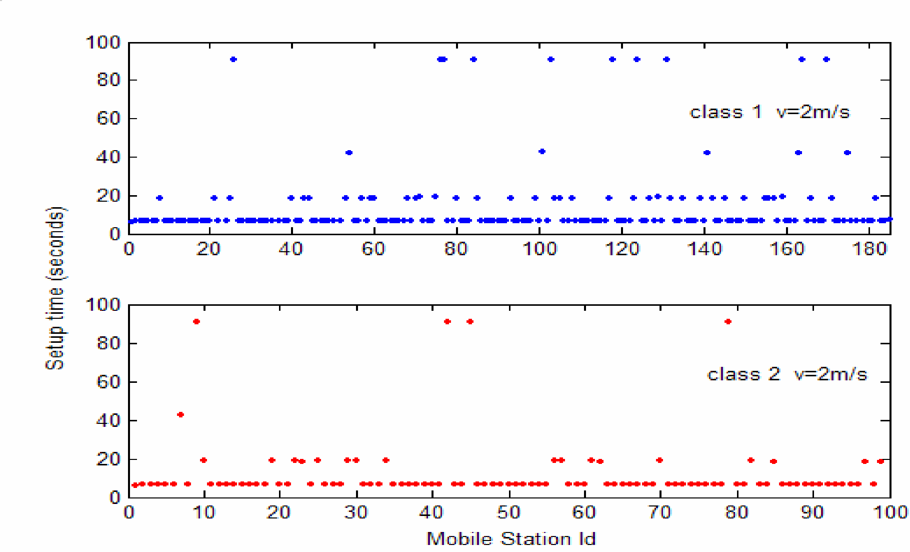


Figure 4.3 Setup times of each station of two classes with the same velocity

Figure 4.3 shows the simulation results of the two classes with the same velocity (2 m/s). From the figure, we think that most of the data concentrate around a constant value. We did some data statistics: for class 1, 78% of the data falls in the interval [3.5, 4.5], and for class 2, 81% of the data falls in interval [3.5, 4.5]. If we ignore the minority of outliers, we can say the setup time is constant for each station with a mean around 4 seconds.

Figure 4.4 shows the simulation results of the two classes with the different velocities (2m/s and 10 m/s). The figure illustrates that most of the data concentrates around a constant value. We did some data statistics: for class 1 (v = 10m/s), 85% of the data falls in interval [5.5, 6.5], and for class 2 (v = 2m/s), 87% of the data falls in the interval [3.5, 4.5]. If we ignore the minority of outliers, we can say the setup time obtained from the simulation matches the model's assumption.

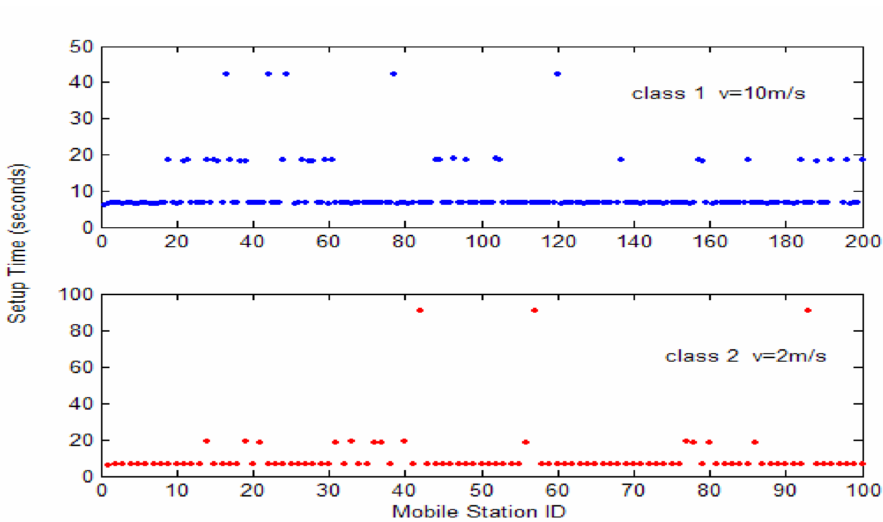


Figure 4.4 Setup times of each station of two classes with the different velocities

3) The average throughput and the total amount of received data

Figure 4.5 and Figure 4.6 show the average throughput of each station of two classes with v = 2 or 10 m/s and 2 or 20 m/s. The shapes of the two curves in both figures look similar. Since the system starts empty, in the beginning there is a system warm-up period. Because there are two classes in the system at the same time, the warm-up period for both classes will be shorter compared to those of the single class case. When the first station comes inside the system, it uses all the capacity. As soon as another station arrives, the capacity is shared, which causes the high average throughput of the first few stations. As more stations cross the system, the system will tend to saturation. That is why at the tail of the curves the data fluctuates around some constant value.

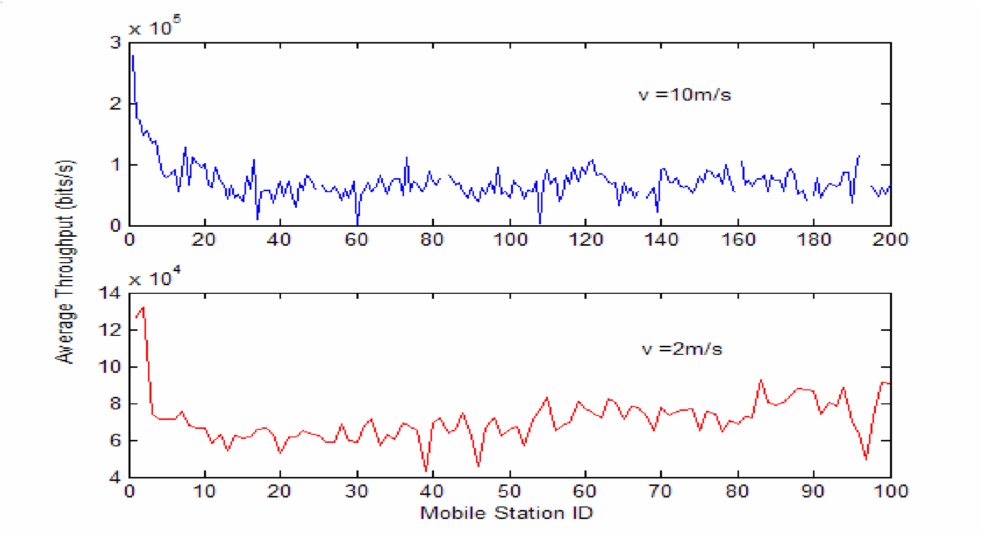


Figure 4.5 Average throughput for each station with velocity of 2 or 10

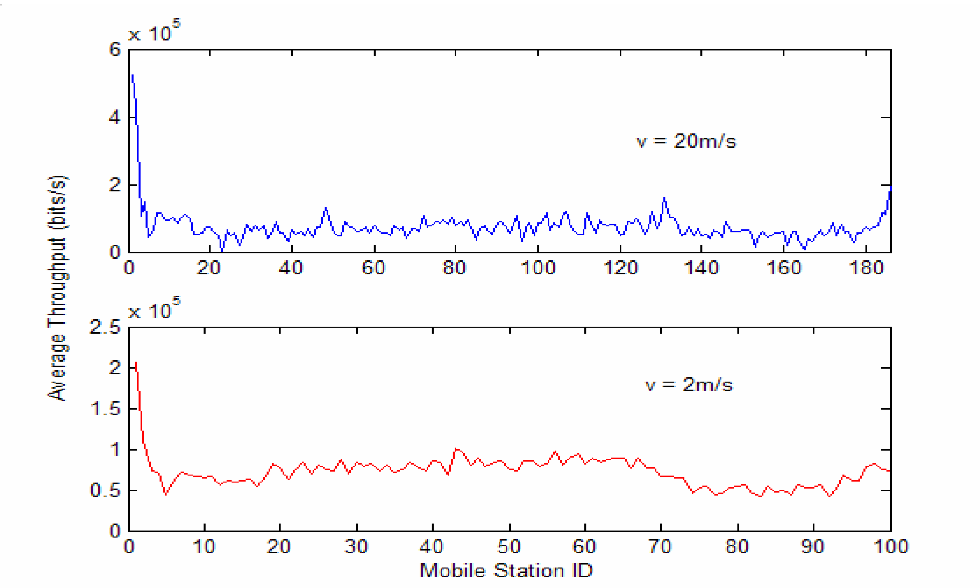


Figure 4.6 Average throughput for each station with velocity of 2 or 20

According to Formula 4.10, in the section of two classes, the analytic expression for the average throughput can be written as

$$TP = \frac{C}{\lambda_1 E(S_1) + \lambda_2 E(S_2)} \cdot \left(1 - e^{-\lambda_1 E(S_1) - \lambda_2 E(S_2)}\right). \tag{4.12}$$

According to formula 4.11, the expressions for the total amount of received data for two classes are

$$E(B_1) = \frac{C \cdot E(S_1)}{\lambda_1 E(S_1) + \lambda_2 E(S_2)} \cdot \left(1 - e^{-\lambda_1 E(S_1) - \lambda_2 E(S_2)}\right), \tag{4.13}$$

and

$$E(B_2) = \frac{C \cdot E(S_2)}{\lambda_1 E(S_1) + \lambda_2 E(S_2)} \cdot \left(1 - e^{-\lambda_1 E(S_1) - \lambda_2 E(S_2)}\right). \tag{4.14}$$

We count the total number of received bits and then calculate the average individual throughput as the total number of bits the station obtained, divided by the total time when it received a packet (also known as the sojourn time). We discard the data got from the stations that are in the warm-up period. The analysis results are calculated from formulas 4.12, 4.13, 4.14 and the values of the parameters used are summarized in Tables 3.1, 3.4 and 4.1. The comparison of the numerical results from the model and simulation results are shown in Table 4.2, and Table 4.3. From these two tables we can see that the differences between the simulation results and the model prediction values are all less than 7%. We can say that our model can predict the average throughput for different velocities accurately.

Group ID	Speed (m/s)	Throughput Simulation (bits/s)	Throughput Model (bits/s)	Difference
A	2	3.2467e+004	3.0435e+004	6.26%
	2	3.2733e+004		7.02%
B	2	6.6151e+004	6.2603e+004	5.36%
	8	6.6617e+004		6.03%
C	2	6.9021e+004	6.7293e+004	2.57%
	10	6.6822e+004		0.70%
D	2	7.8942e+004	7.9154e+004	0.27%
	20	7.8091e+004		1.34%

Table 4.3 The simulation results vs. the model prediction of the average throughput

Group ID	Speed (m/s)	Total-received Simulation (bits)	Total-received Model (bits)	Difference
A	2	7.4834e+006	7.3333e+006	2.01%
	2	7.5479e+006	7.3333e+006	2.84%
B	2	1.4854e+007	1.5084e+007	1.54%
	8	3.5421e+006	3.4580e+006	2.37%
C	2	1.6247e+007	1.6214e+007	0.20%
	10	2.7994e+006	2.8929e+006	3.34%
D	2	1.8432e+007	1.9072e+007	3.47%
	20	1.5135e+006	1.4639e+006	3.28%

Table 4.4 The simulation results vs. the model prediction
of total amount of received data

4.4 Optimization of the model

Now that we developed and validated models for performance, we can use them to optimize the system design and to determine quantities such as the optimal value of the distance of the AP, the optimal value of the coverage of the AP, etc. In the following part, we will give examples to show how our models can be used.

4.4.1 Find optimal location of Access Points

We can use the expressions we derived in Section 4.2 to optimize the system parameters to satisfy the given performance requirement. Consider the following situation: a mobile user wants to finish a download of a fixed size of information (e.g., one city map, one email with an attachment etc.) when he crosses the AP area. That means the total amount of received data E(B) should be larger than that required file size. Suppose the information sums up to a size of 1.45Mbits and the coverage of the AP cell is 250m. There are two classes of users inside the AP cell. The Poisson arrival rates of these two classes are 0.05 user/second and 0.1user/second separately. Class1 represents the users who are walking with their PDAs with an average velocity of 2 m/s and class2 represents the users who are working with their laptop in a car with an average velocity of 20 m/s. As we analyzed before, if the distance between the AP and the road is smaller, the sojourn time of the users will be larger. On the other hand, if the sojourn time is larger, the average number of users who stay inside the system becomes larger, and thus the average throughput of the user is smaller. Since the E(B) is the product of the average throughput and the sojourn time, there is a trade off between a large sojourn time and a large throughput. By using Expressions 4.13 and 4.14, we can determine where to place the AP on the road, that is, the distance *d* parameter in Equation 4.1. The numerical results for the scenario we supposed here are shown in Table 4.5. Since the sojourn time of the users in class1 is much larger than the one of the class2, the total amount of data of class1 is large than class2. To find the optimal d value, we only need to consider about the requirement for the class2. We can easily know that when the distance of the AP is 80m, the users of class2 can receive 1.4506Mbits data in average, which can satisfy the requirement. So the optimal value of d is 80.

Distance (m)	150	100	80	50
Class1 received data (bits)	1.9250e+007	1.9126e+007	1.9099e+007	1.9072e+007
Class2 received data (bits)	1.3750e+006	1.4369e+006	1.4506e+006	1.4639e+006

Table 4.5 The total received data with different given AP location

4.4.2 Connection Admission control (CAC)

Performance is an important quality factor for users. As we analyzed before, in a Processor Sharing model, the server will accept all incoming users who will share the available throughput. Suppose there are thousands of users, each of them only gets very little throughput. The network service performance will then not be satisfactory. To improve the system performance, we can try to consider the Quality of Service (QoS) aspect. We can add an Admission Control module to the protocols to guarantee that each accepted user will at least get some minimum throughput, denoted as $TP_{threshold}$. Since the capacity of the system (C) is fixed, to guarantee this QoS the system can only accept a limited number of users. According to the Processor Sharing property, the maximum number of users the system can sustain can be given by

$$N = \frac{C}{TP_{threshold}} \quad . \tag{4.15}$$

If there are already N users inside the AP cell, the users who arrive after that will be rejected by the AP. They will not get associated to the AP, and thus can not get any internet service. We can say that this part arrival of users is lost. Thus, we can refine our model to the Erlang loss model. Using the same Markov chain analysis way as in Chapter 3, we can use calculate the probability of the number of users. Here we only illustrate a simple single class scenario. Suppose the total user arrival rate is λ, and the service rate can be denoted as $\mu = \frac{1}{E(S)}$. The state transition-rate diagram of this Markov system is shown in Figure 4.7.

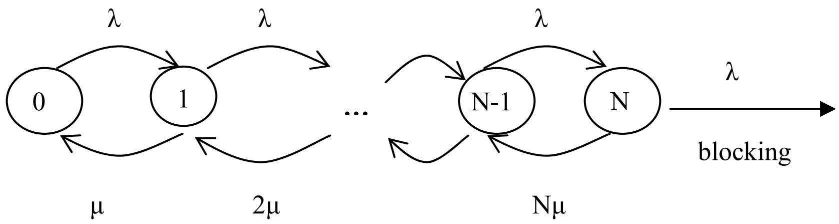


Figure 4.7 State- transition-rate diagram for the Erlang Loss Model

It is easy to derive the stationary distribution of the number of users inside the system as

$$\pi_k = \frac{(\lambda E(S))^k / k!}{\sum_{i=0}^{N} (\lambda E(S))^i / i!}, \quad k = 0,1,\ldots,N. \tag{4.16}$$

Although Expression 4.16 is derived from a single class scenario, due to the insensitive property of this formula and the superposition property of Poisson process, it can be easily extended to the multiple class scenario as

$$\pi_j = \frac{\gamma^j / j!}{\sum_{i=0}^{N} \gamma^i / i!} \quad j = 0,1,\ldots,N, \tag{4.17}$$

where $\gamma = \sum_{i=1}^{k} \lambda_i E(S_i)$ is the aggregated parameter for the k-class superposed Poisson processes.

So the probability when the arrival user is rejected by the AP is the probability of the system in the N-user state, which can be calculated by

$$P\{user\ rejected\} = \pi_N = \frac{\gamma^N / N!}{\sum_{i=0}^{N} \gamma^i / i!} \ . \qquad (4.18)$$

Usually, to guarantee the QoS, the system designer requires a maximum user rejection probability, denoted as $\alpha \in (0,1)$. That is, the probability of the user who finds the system is full when he arrives should be smaller than α. We can use Formula 4.18 to determine the optimal value of N such that

$$P\{user\ rejected\} \le \alpha \ .$$

After we determine the value of N, we can use Equation 4.15 to determine the required system capacity C. As we analyzed in Section 3.2, C mainly depends on the 802.11 protocol data rate. Using our formula, it can be easily decided which 802.11 protocols (802.11b, 802.11a, or 802.11g, etc.) are to be used to satisfy the system requirement. Here we give a simple example to illustrate how to use our model. We assume that the user rejection probability should smaller than 1%. Suppose there are two classes' users, the arrival rates are λ_1= 0.05, λ_2=0.1, the average velocities are v_1= 2m/s, v_2= 10m/s. According to Equation 3.15, we can derive the sojourn times $E(S_1)$ = 241 second, $E(S_2)$ = 43 second. Using Equation 4.18, we calculated that when N is equal to 26, the probability is 0.0071 and when N is equal to 25, the probability is 0.0113. To satisfy the blocking probability requirements, the system has to at least afford 26 users at the same time. Suppose it is required that each accepted user will at least get minimum throughput 80Kbps. Using Equation 4.15, we can obtain that it is required that the system can supply at least 2.06 Mbps capacity. According to the analysis in Section 3.2, we know that the 802.11b protocol with 11Mbps physical data rate can provide system capacity around 2.2631 Mbps. So the system designer can choose 802.11b protocols to satisfy the system QoS requirements.

4.5 Conclusions

In this chapter, we have investigated the multi-class users in the single cell scenario, that is, the users drive through one AP cell with different average velocities.

Basically, we have extended the single cell model to multiple-class model. Similar to Chapter 3 we modeled the AP server as two $M/G/\infty$ queuing servers in tandem. The difference is that there are k different classes of mobile users which drive with different average velocities in this chapter, while there is only one class of mobile users which drive with the same average velocity in Chapter 3. By using the superposition property of the Poisson Processes, we derived the closed-form expressions of the performance, in terms of throughput and total received data. To validate our multiple-class model, simulations were run for different velocity combinations of two classes. The results of simulations show that our model can predict the performance metric for different velocities accurately.

Subsequently, we gave two examples to show how we can use our model to optimize the system designs. The first one illustrated that if a mobile user requires finishing a download of a fixed size of information, where the AP should be deployed to make the total amount of received data of this user is larger than that size. The second one showed another aspect of system design: the choice for the protocol used. Suppose an Admission Control module is added to the protocols, if the system can not guarantee the arrival user can get the minimum performance service, the user will be rejected. We showed how to use our model to choose the protocol which can supply high enough system capacity to guarantee user rejection probability lower than a given requirement.

Chapter 5
The Multiple-cell model

5.1 Introduction

In the previous chapters, we have studied the performance expressions for multiple classes of users in a single AP cell. However, usually there are many APs deployed on the highway and the users will cross them one by one. Therefore, it is important to study the performance of the users in a multiple-cell scenario. In the next section we derive the performance expressions of users who cross several AP cells. After that the optimization of our model is analyzed in Section 5.3.

5.2 The model of the 802.11 multiple-cell and analysis

Suppose there are a few roads that cross each other, many APs are deployed on the road and different classes of mobile users drive along these roads. A simple example of this scenario is shown in Figure 5.1. If one mobile station drives along the road, such as STA1 in Figure 5.1, then what is the performance measured?

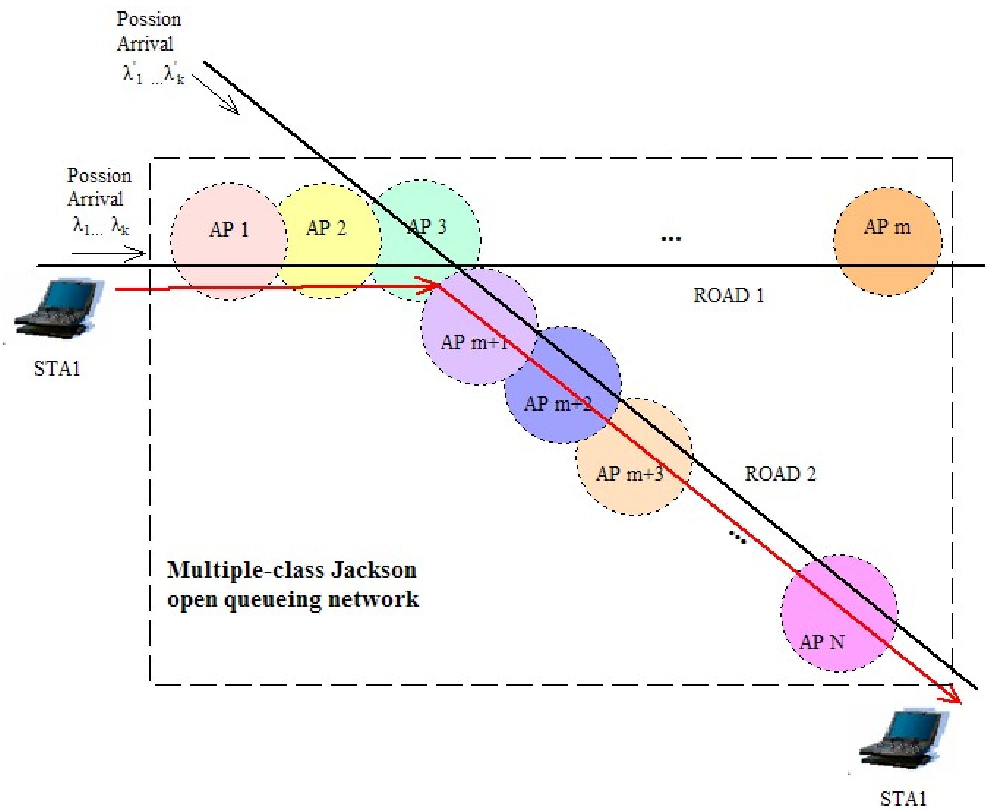


Figure 5.1 Example of the system topology of a multiple cell scenario

We can treat the area where the APs are accessible as an integrated system. Just as shown in Figure 5.1, the APs which are inside the square with dashed lines can be treated as a whole system. We model this system by a Markovian open queuing network (i.e., a Multiple-class Jackson open queuing network [Jackson 57]) where users of different classes are routed from one server to another according to a transfer probability matrix. Consider this system consisting of N cells where the ith node consists of two $M/G/\infty$ queuing servers in tandem (as we analyzed in Chapter 4).

The coverage of the Access Point in the ith node is R_i and the distance between this AP and the main road is d_i, i= 1, ⋯, N. There are R classes of mobile users, and the stations' average velocity for class k is $v^{(k)}$, k= 1, ⋯, R. As we analyzed in Section 3.3.2, the service time of the association server will depend on the velocity which follows the rules listed in Table 3.4. We denote this association service time for class k as $E(U^{(k)})$, k= 1, ⋯, R. We denote the average sojourn time the mobile user of class k will stay inside the ith node as $E(S_i^{(k)})$, which can be calculated by

$$E(S_i^{(k)}) = \frac{2 \cdot \sqrt{R_i^2 - d_i^2}}{v^{(k)}} - E(U^{(k)}), \quad i = 1, \cdots, N. \tag{5.1}$$

Since there is no handoff control implemented in the 802.11 protocol, when the mobile user moves from one AP cell to another, he just repeats the association and authentication procedure again and after successful association to the next AP he starts to get the internet service. Thus, the mobile users in each cell are served separately. Furthermore, we suppose the ith node receives arrivals of class k of users from outside the system in the form of a Poisson process at rate $\alpha_i^{(k)}$. Parts of class k users who completed their service at cell i are automatically routed to cell j with probability $p_{ij}^{(k)}$, ($i \neq j$), i,j = 1, ⋯, N; k=1, ⋯, R. A part routed from cell i to cell j is automatically accepted by cell j (i.e., cell j has no control capability to reject or block the arrival of a part into it). We must calculate the total average arrival rate of users for class k to a given node. To do this, we must sum the (Poisson) arrivals from outside the system plus arrivals (not necessarily Poisson) from all internal nodes. By denoting the total average arrival rate to node i by $\lambda_i^{(k)}$, we easily find that this set of parameters must satisfy the following equation

$$\lambda_i^{(k)} = \alpha_i^{(k)} + \sum_{j=1,\, i \neq j}^{N} \lambda_j^{(k)} \cdot p_{ji}^{(k)} \quad i = 1, 2, \ldots, N, \quad k = 1, 2, \ldots, R. \tag{5.2}$$

What is amazing is that Jackson was able to show that each node (say the ith) in the network behaves as if it were an independent $\cdot/G/\infty$ queuing system with a Poisson input rate $\lambda_i^{(k)}$ for each class k. In general, the total input will not be a Poisson process. The state variable for this N-node system consists of the vector $(n_1, n_2, \ldots, n_N)$, where n_i is the total number of users in the ith node. Let the equilibrium probability associated with this state be denoted by $p(n_1, n_2, \ldots, n_N)$. Similarly we denote the marginal distribution of finding n_i users in the ith node by $p_i(n_i)$ [Kleinrock 76]. Jackson was able to show that the joint distribution for all nodes factored into the product of each of the marginal distributions, that is,

$$p(n_1, n_2, \ldots, n_N) = p_1(n_1) \cdot p_2(n_2) \cdots p_N(n_N), \tag{5.3}$$

and $p_i(n_i)$ is given as the solution to the single cell system, see, Equation 4.4 with the obvious change in notation. In the same way, we define $n_i^{(k)}$ the number of users of class *k* in the *i*th node. The equilibrium probability associated with the state in the *i*th node is denoted by $p\left(n_i^{(1)}, n_i^{(2)}, \ldots, n_i^{(R)}\right)$. Similarly we denote the marginal distribution of finding $n_i^{(k)}$ users of class *k* in the *i*th node by $p_i\left(n_i^{(k)}\right)$. In Chapter 4 it is shown that the joint distribution for all classes can be factored into the product of each of the marginal distributions, that is,

$$p\left(n_i^{(1)}, n_i^{(2)}, \ldots, n_i^{(R)}\right) = p_i\left(n_i^{(1)}\right) \cdot p_i\left(n_i^{(2)}\right) \cdots p_i\left(n_i^{(R)}\right), \tag{5.4}$$

and $p_i\left(n_i^{(k)}\right)$ is given as the solution to the single cell system, (see Equation 4.3). Then from Equations 5.3 and 5.4 one sees that

$$P\left\{N_i^{(k)} = n_i^{(k)}, i = 1, \ldots, N; k = 1, \cdots, R\right\} = \prod_{i=1}^{N} \prod_{k=1}^{R} \frac{e^{-\rho_i^{(k)}} \cdot \left(\rho_i^{(k)}\right)^{n_i^{(k)}}}{n_i^{(k)}!}, \tag{5.5}$$

where $\rho_i^{(k)} = \lambda_i^{(k)} \cdot E(S_i^{(k)}), i = 1, \ldots, N \quad k = 1, \cdots, R$ is the Poisson distribution parameter for each class. Then it is easily verified that

$$P\left\{\sum_{k=1}^{R} N_i^{(k)} = n_i, i = 1, \ldots, N\right\} = \prod_{i=1}^{N} \frac{e^{-\rho_i} \cdot (\rho_i)^{n_i}}{n_i!}, \tag{5.6}$$

where $\rho_i = \sum_{k=1}^{R} \rho_i^{(k)} = \sum_{k=1}^{R} \lambda_i^{(k)} \cdot E(S_i^{(k)}), i = 1, \ldots, N.$ (5.7)

Therefore, if we are interested only in the total number of users of different classes in the set R of server cells, we may concentrate on one infinite-machine server.

Maybe in different cells, different 802.11 protocols will be used, for example 802.11b, 802.11a, or 802.11g. Suppose the aggregated throughput of the users in the *i*th node is C_i. By using the same method as in Chapter 4, we can derive the average throughput the mobile users can get in the *i*th node TP_i by

$$TP_i = \sum_{n_i=0}^{\infty} \mathrm{P}\left(\sum_{k=1}^{R} n_i^{(k)} = n_i\right) \cdot \frac{C_i}{n_i + 1} = \sum_{n_i=0}^{\infty} \frac{e^{-\rho_i} \cdot \rho_i^{n_i}}{n_i!} \cdot \frac{C_i}{n_i + 1}$$

$$= \frac{C_i}{\rho_i} \cdot (1 - e^{-\rho_i}), \tag{5.8}$$

where ρ_i can be calculated form Equation 5.7, $\lambda_i^{(k)}$ from Equation 5.2, and $E(S_i^{(k)})$ from Equation 5.1. Then it is also easy to derive the total amount of received data for class *k* in the *i*th node $E(B_i^{(k)})$ by

$$E(B_i^{(k)}) = E(S_i^{(k)}) \cdot TP_i = E(S_i^{(k)}) \cdot \frac{C_i}{\rho_i} \cdot (1 - e^{-\rho_i}) . \tag{5.9}$$

5.3 Applications of the multiple-cell model

Our model is very powerful. We derived very simple closed-form expressions for the performance metrics, which can be easily used for the performance analysis and system optimization.

5.3.1 Performance evaluation

By using our models, service providers are able to predict the performance of their networks under hypothetical scenarios. For example, if you get a map of the location of the APs the users' traffic patterns and the user's travel route, you can easily calculate the performance metrics provided by this system with very little effort. In general, for a given user traffic pattern (arrival rate and average velocity), we can easily calculate the total amount of received data. Suppose one mobile user has crossed the AP covering area, and in total he passed by J cells. Route vector (A_j , $j = 1, \ldots, J$) includes the cell node IDs which the users cross during their journey. Thus the total amount of received data for this user when he crosses the whole system can be calculated by

$$E(B_{total}) = \sum_{j=1}^{J} E(B_{A_j}^{(k)}) = \sum_{j=1}^{J} E(S_{A_j}^{(k)}) \cdot \frac{C_{A_j}}{\rho_{A_j}} \cdot (1 - e^{-\rho_{A_j}}) . \tag{5.10}$$

For example, in the case of STA1 as shown in Figure 5.1, according to Formula 5.2, the total average arrival rate to node *i* of class *k* can be rewritten as

$$\begin{cases} \lambda_1^{(k)} = \lambda_2^{(k)} = \lambda_k , \\ \lambda_3^{(k)} = \lambda_k + \lambda_k^{'} , \\ \lambda_{m+1}^{(k)} = \lambda_{m+2}^{(k)} = \cdots = \lambda_N^{(k)} = \lambda_k^{'} . \end{cases} \tag{5.11}$$

where λ_k is the arrival rate of class *k* of users from outside the system through ROAD1 in the form of a Poisson process and $\lambda_k^{'}$ is the arrival rate of class k through ROAD2.

The route vector A of STA1 is

A = {1, 2, 3, m+1, m+2, …, N}. (5.12)

We can then use Formulas (5.1), (5.7), (5.10), (5.11) and (5.12) to calculate the total amount of data received by STA1 during the period he moves on ROAD1 and ROAD2.

5.3.2 Optimal placement of Access Points

Secondly, if we know the user traffic pattern, we can decide how many AP spots we need and where to put them. Let us have a look at the simple scenario as shown in Figure 5.2. There is one highway road whose length is *D*m, and only one class of users drives across this road, for example the cars. Their Poisson arrival rate is λ m/s, and the average velocity of the users is v m/s. If we would like to make the mobile users receive as much data as they can during the period when they drive across this road, how many AP spots do we need? Suppose the coverage of each AP cell is R m , and the length where the road is covered inside the AP is L m. As we analyzed before, the users will spend a fixed amount of time, denoted as T s, to associate with the AP. The setup time is determined by the average velocity of users, which follows the rule shown in Table 3.4. During this association period, the users can not receive any data. If we set L too small, then all the time will be consumed by the connection setup. So we would like to set L larger. On the other hand, if the L is too large, at the same time, there will be too many users inside the system. According to the Processor Sharing theory, the average throughput each user gets will be very small, and the total amount of received data will be small. Thus, we would prefer a small value of L. To find the optimal value of L, we perform the following analysis.

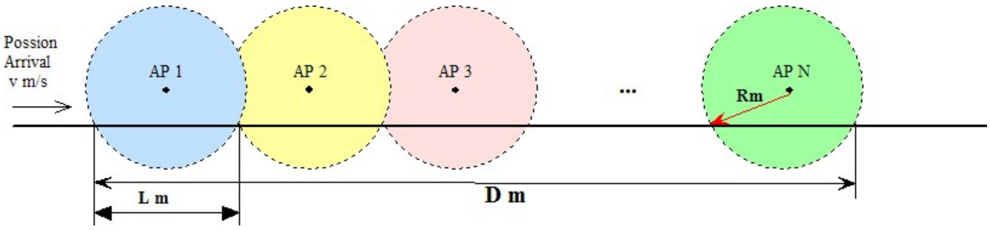


Figure 5.2 Cell dimensions

Suppose there are N AP cells deployed on the road. The coverage length in each cell can be written as

$$L = \frac{D}{N}, \tag{5.13}$$

and the average sojourn time can be calculated as

$$E(S) = \frac{D}{N \cdot v} - T. \tag{5.14}$$

Thus we can express N by E(S) as

$$N = \frac{D}{v \cdot (E(S) + T)}. \tag{5.15}$$

According to Formula 3.19, in total, the users that cross the road will receive

$$E(B_{total}) = N \cdot E(B) = N \cdot \frac{C}{\lambda} \cdot (1 - e^{-\lambda E(S)}). \tag{5.16}$$

Applying Equation 5.15 to Equation 5.16, we have

$$E(B_{total}) = \frac{D}{v \cdot (E(S)+T)} \cdot \frac{C}{\lambda} \cdot (1-e^{-\lambda E(S)})$$

$$= \frac{D \cdot C}{v} \cdot \frac{1}{\lambda E(S) + \lambda T} \cdot (1-e^{-\lambda E(S)}) . \quad (5.17)$$

Denote $\lambda E(S)$ as x, and $E(B_{total})$ as $f(x)$, then Equation 5.17 can be written as a function of x.

$$f(x) = \frac{D \cdot C}{v} \cdot \frac{1-e^{-x}}{x + \lambda T} . \quad (5.18)$$

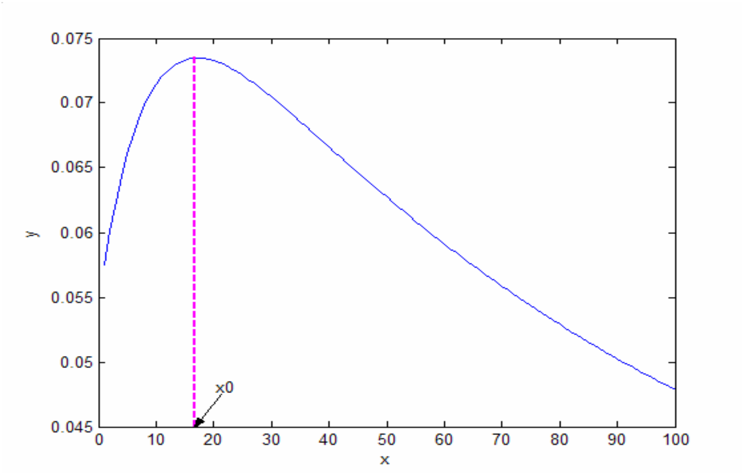


Figure 5. 3 The shape of function 5.18

The shape of y as a function of x is shown as in Figure 5.3. Although for different values of D, C, v, λ, the shape of the function will be a bit different, only one maximum value of y exists. By using standard calculus, it is easily verified that $f(x)$ has a unique maximum value at $x = x_0$. To optimize the system, we would like to find the value of x_0 which leads to the greatest value of $E(B_{total})$. To do this, we take the derivative of $f(x)$ with respect to x, which is given by

$$f'(x) = \frac{D \cdot C}{v} \cdot \frac{(x+\lambda T) \cdot e^{-x} - (1-e^{-x})}{(x+\lambda T)^2} . \quad (5.19)$$

Let $f'(x) = 0$, then we have the equation of x as

$$x - e^{x} + \lambda T + 1 = 0 \quad (5.20)$$

Equation 5.20 shows that the optimal value of x only depends on the Poisson arrival rate λ and the setup time T, which is determined by the average velocity v. In other words, given the user traffic pattern we can always find the optimal cell coverage size (L) which can maximize the performance. Although there is no closed-form solution for Equation 5.20, it can be easily solved numerically by reducing this equation to a fixed point equation, i.e., $f(x) = x$, where $f(x) = e^x - \lambda T - 1$. It can be easily proved that there is a unique solution for this fixed point equation as shown below.

Suppose we have two functions $f(x) = e^x - \lambda T - 1$ and $g(x) = x$. We define the difference between these two functions as

$$h(x) = f(x) - g(x) = e^x - \lambda T - 1 - x.$$

We take the derivative of $h(x)$ with respect to x, which is given by

$$h'(x) = e^x - 1.$$

For each $x \geq 0$, $h'(x) \geq 0$, thus the differential function $h(x)$ keeps increasing. For this reason, we know that $h(x)$ has maximum 1 zero point. That is, there is only one solution x which makes $f(x) = g(x)$.

Since there are many standard methods to solve these types of expressions (e.g., Picard iteration), we can easily get the optimal value of x_0, and then calculate the system parameter L or the number of cells (N) needed for the Dm long road.

To illustrate this, consider the following example. Suppose a part of the highway with length D is 10 Km and the Poisson arrival rate is 0.1users/second. 802.11b protocols with 2Mbps physical data rate are used. As we analyzed in Chapter 3, the system capacity C is then around 1.1 Mbps. We would like to find the optimal value of L to maximize the performance of the users with different velocities. Figure 5.4 shows the total amount of received data *f(x)* as a function of x for different velocities, where $x = \lambda E(S)$. Since the setup time when the average velocity is 10 m/s is equal to the one of velocity 20 m/s, the

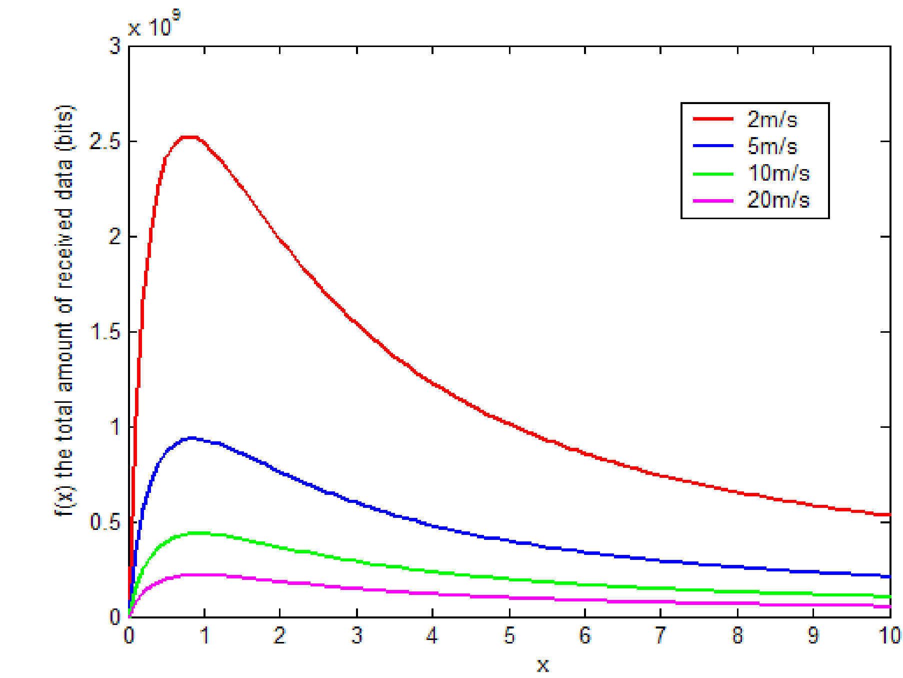


Figure 5.4 $E(B_{total})$ (*f(x)*) as function of $\lambda E(S)$ (x) for different velocities

optimal value of x should be the same for these two cases. Using the method we explained in the previous paragraph, we can solve the fixed-point equation to numerically obtain the optimal value of x to maximize $f(x)$. After we obtain the value of x, we can easily get the numerically optimal results of the cell size L and the number of cells N, by using the Equations (5.13) and (5.14). The numerical results are shown in Table 5.1. Furthermore, we also illustrated the total received data as function of the number of cells (N) for different velocities in Figure 5.5.

Velocity (m/s)	2	5	10	20
Received data (Mbits)	2524.4	933.12	435.29	217.64
x ($\lambda E(S)$)	0.78	0.86	0.93	0.93
Cell Size L(m)	23.6	68	153	306
Number of cells N	424	147	145	33

Table 5.1 Numerical results of the optimization

Table 5.1 shows that the less the average velocity is, the smaller the cell size is. As we analyzed before, if the cell size L is smaller, the sojourn time $E(S)$ of the users will be larger. On the other hand, if $E(S)$ is larger, the average number of users who stay inside the system becomes larger, and thus the average throughput of the user is smaller. Since the E(B) is the product of the average throughput and the sojourn time, there is a trade off between a large sojourn time and a large throughput. That is, if the cell size is too large when the velocity of the users is small, there will be too many users inside the system which can severely decrease the system performance. So if the velocity of the majority users in the cell is slow, we would prefer to adopt a small cell size for the AP.

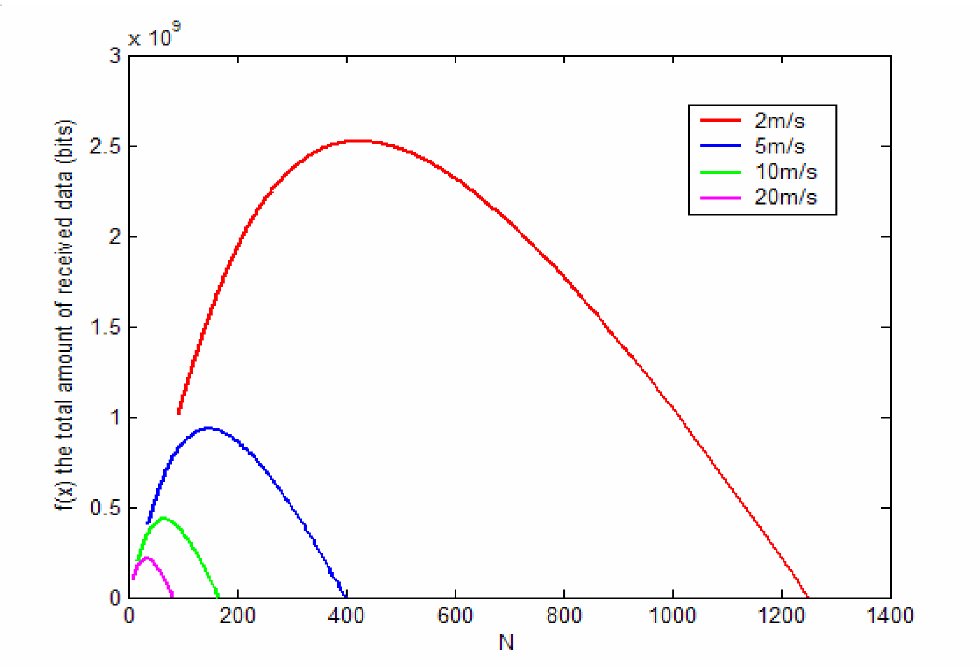


Figure 5.5 The total amount of received data as function of the number of cells (N) for different velocities

5.4 Conclusions

In this chapter we have investigated the performance of multiple-class users across multiple AP cells. We extended the model made in the previous chapter to a Multiple-class Markov open queuing network model. Using the Jackson open queuing network theory, we derived simple closed-form expressions for the average throughput the mobile users can get in the ith node (TP_i) and the total amount of received data for class k in the ith node $E(B_i^{(k)})$. There is no hand-off procedure in 802.11 protocols, thus each AP cell is independently and stochastically identical. For this reason, we have only validated our model for the single cell scenario in Chapter 3 and 4.

We provided two examples on how our model can be used to analyze practical problems. In the first example we showed that it is really easy to calculate the performance metrics provided by this system with very little effort. In the second example we illustrated that for a given users' traffic pattern, we can determine how many AP spots the system designers need and where to put them.

Chapter 6
Conclusions

6.1 Introduction

This chapter presents the conclusions of this research. In the next section a summary of the results of our research is given. In Section 6.3 we indicate the contributions of our research, and in Section 6.4 directions for future work are given.

6.2 Summary

The aim of this thesis is to understand the impact of the mobility of the mobile terminal on performance of internet service via WLAN networks**.** Models are developed that measure performance as experienced by the end users in this domain (in terms of throughput and total amount of received data), for a given set of design choices and realistic user-behavior scenarios.

To achieve this goal, in Chapter 2 we first investigated the details of the 802.11 WLAN protocols and TCP protocols. In our studies, we focus on the 802.11 architecture ESS, in which the AP provides a local relay function to a distribution system. In the MAC sub layer, we studied the DCF, which is based on the Carrier Sense Multiple Access with the Collision Avoidance protocol. There are two access methods specified in DCF: the basic access method and the optional four-way access method. In the latter the station will first send a RTS packet to the destination. After the CTS packet is received, the station can start the packet transmission in a similar way as the basic access method does.

The wireless internet traffic is commonly carried over the well-known TCP/IP protocol suite. TCP implements two control mechanisms: the flow control and the congestion control. Flow control is implemented with a *receiver advertised window* that limits the number of packets the sender can inject into the network. Congestion control uses an additional *congestion window*, to regulate the senders' transmission rate. We studied the Network Simulator 2 (NS-2), which is commonly used and widely accepted in the networking research community as the basic simulating tool for network evaluations. We used NS-2 to validate our models.

After these preliminary explorations we investigated the performance models ranging from a simplified scenario to more complicated realistic scenario's. Because the factors that can affect the performance of Wireless communication are very complicated, it is better to first simplify these factors and take them into account when we build the models. To start, in Section 3.2 we only analyzed the network performance of an 802.11-based WLAN in the presence of a single TCP connection. By taking the Two-Ray Ground model for the physical radio propagation, a reasonable assumption is made that the channel is loss-free during the period when the STA keeps inside the hearing range of the Access Point. According to the data transmission procedure of the RTS/CTS access mechanism and the 802.11 data format; we derived the throughput for a single TCP connection in RTS/CTS mode. Simulations ran with the NS-2 2.26 simulator showed that the analytical results coincide with the simulation results. Based on those validated results, in Section 3.3 we analyzed a more complicated scenario: any number of stations cross the coverage of the AP with the same average velocity. We modeled the single AP cell as two $M/G/\infty$ queuing servers in tandem: the setup server and the internet service server. Based on findings of the simulation preliminaries and using the results of the previous step, we derived the average throughput (and the total amount of received data) of any number of stations with the same average velocity when they cross the coverage of the AP. The expressions of the performance are thoroughly validated by the simulations in NS2.

In real life people can access the internet when they are walking with their PDA, use their laptops while driving a car, etc. This implies that there is always more than one class of users. In Chapter 4, we considered a more complicated case: multi-class users in the single cell scenario. We continued to use the single cell model made in Chapter 3, which modeled the AP server as two $M/G/\infty$ queuing servers in tandem. The difference is that there are *k* different classes of mobile users which drive with different average velocities. By using the superposition property of Poisson Processes, we derived closed-form expressions of the performance, in terms of throughput and total received data. The results of the simulations showed that our model can predict the performance metric for different velocities accurately. Subsequently we gave two examples to show how we can use our model to optimize the system designs.

After thoroughly studying the performance expressions for multiple classes of users in a single AP cell, in Chapter 5 we investigated the performance of multiple-class users across multiple AP cells. We extended the model made in Chapter 4 to a multiple-class Markov open queuing network model. Using the Jackson open queuing network theory, we derived simple closed-form expressions for the average throughput and the total amount of received data of different classes of users. After that we provided two examples on how our model can be used to analyze practical problems. In the first example we showed that it is really easy to calculate the performance metrics provided by this system with very little effort. In the second example we illustrated that for a given users' traffic pattern, we can determine how many AP spots the system designers need and where to put them.

6.3 Contributions

The mobility of wireless stations may be the most important feature of a wireless LAN. A WLAN would not serve much purpose if stations were not able to move about freely from location to location either within a specific WLAN or between different WLAN 'segments'. So it is important to understand the impact of the mobility of the mobile terminal on performance of mobile internet service via WLAN networks**.** A lot of research is done in the field of performance modeling of the IEEE 802.11 protocol. However, to the authors' knowledge there are no papers providing analytical modeling approaches to the performance of TCP over WLAN experienced by the mobile terminals. In this thesis, we present an analytical approach to the performance evaluation of TCP over IEEE 802.11 WLAN, thus filling the gap in existing literature. In particular, we developed a tractable analytical model for the average throughput and the total amount of received data over multiple AP cells with any number of mobile users.

1. We first focused on the packet level analysis to derive the expression of the performance for only one single station in the AP domain (cell). The well known analytical model TRG is used for the mobile radio propagation. Based on the insight in the packet level behavior of both TCP and the DCF of the 802.11 MAC layer, we derived an approximation closed-form expressions of the aggregated throughput.

2. We modeled the single AP cell as two $M/G/\infty$ queuing servers in tandem. One is the setup server, in which the service time is fixed according to the average velocity of the users. Another one is the Internet service server, in which the user sojourn time is fixed depending on the range of the AP, the average velocity. The WLAN is considered as a

Processor Sharing (PS) queuing system with service capacity equal to the aggregated throughput derived in the previous step. By using the known analytical results of the Poisson process and the PS property, we derived the approximate of the average throughput and the total amount of received data. NS-2 simulations are used to validate the results of these approximations.

3. By using the known superposition property of Poisson Processes, we extended the results of a single class user case to a multiple-class user case. NS-2 simulations are also used to validate the results of these approximations. Examples are given to show the practical usage of our approximations of the performance.

4. The analytical results of Jackson's open queuing network are used to extend our previous results to the multiple-class multiple-cell case. At the end we also illustrated the approach to do performance analysis and system optimization using our analytical results.

6.4 Future work

1) Physical and TCP layer model enhancement

The mobile radio channel places fundamental limitations on the performance of wireless communication systems. The transmission path between the transmitter and the receiver can vary from a simple line-of-sight to one that is severely obstructed by buildings, mountains, and foliage. Unlike the wired channels that are stationary and predictable, radio channels are extremely random and do not offer easy analysis. Even the speed of motion impacts how rapidly the signal level fades as the mobile terminal moves in space.

The investigations are currently based on a simple two-ray ground reflection model, which does not consider the small-scale fading effect. This is going to be enhanced in one of the next steps. There we will introduce the mobile radio propagation model of small-scale fading superimposed on large scale fading, e.g., the Shadowing model, the Ricean model and the Rayleigh model. These models introduce some kind of unpredictability for data transmissions. Correct receptions are guaranteed for close proximities and impossible over long distances, whereas correct receptions are unpredictable for medium distances. The signal strength variations are not direction-dependent and possible errors can occur during every transmission. It varies significantly between consecutive transmissions and even differs for the reception of the same transmission at different receivers. Thus, the assumption on the loss-free channel in our current research is not reasonable. We should calculate the packet error rate. Moreover, due to the packet loss, the situations in TCP become more complicated: the TCP's congestion control window is decreased whenever a lost packet is detected, with the amount of the decrease depending on whether packet loss is detected by duplicated ACKs or by a timeout. We therefore have to integrate the effect of the TCP's congestion control mechanisms to our current model as well.

Because of the strong influence of the mobility model on the performance, we expect the results to be heavily dependent on the chosen environment and other mobility parameters.

2) Coexistence of immobile users and mobile users

Up to now, we analyzed the performance experienced by the mobile users, but we did not consider other immobile users inside the AP cells. In our next step we would like to extend our model to include somehow Wi-Fi users that are not mobile, stay within a single cell during some arbitrary time interval (during which they download some files). This will have an influence on the performance other mobile users experience.

In this step defining the immobile user traffic pattern is an important factor for the model. Examples of the traffic definition are how long time immobile users will use the internet service, whether the immobile users will leave the system after they finish downloading the file, and whether they will reinitiate another file transfer. For example, we can make the assumption similar to [Litjens 03_2]: a number of fixed users generate file downloads according to a Poisson process with rate λ. The file sizes have a general distribution with mean $1/\mu$ (in bytes). Thus the service time of the immobile users is determined. But in our current model, the sojourn time of the mobile users is determined. Under this kind of assumption the analysis approach of the two kinds of users should be different. Secondly we can make another assumption, stating that the behavior of immobile users (entering a given cell and leaving after a certain period) follows a general distribution with mean D. Now there is no fixed number N of permanent users. Instead N is a random variable with Poisson distribution with mean λD, where λ is the rate (not necessarily Poisson) at which users enter (and hence leave) the cell. Under this assumption the immobile users can be simply included by defining an additional user class, as long as we assume that duration of these sessions are independent random variables.

All of these tentative plans will be shown analytically and by means of simulations in our future work.

3) Integration models of WLAN and 3G cellular networks

The hardware and software to accomplish the integration of WLAN and 3G cellular networks already exists. Many cellular service providers such as T-Mobile and AT&T Wireless have deployed WLAN hot-spots that are integrated with their existing GSM/GPRS cellular network. Other companies such as Bell Canada have already deployed WLAN hot-spots that are integrated with their existing CDMA 1X network. Mobile terminal companies such as DoCoMo [NTT 04] can now provide a user with a dual mode device the ability to have high-speed wireless data access in hot-spots, and to fallback to the cellular data network when outside hot-spot areas.

In an integrated scenario, cellular networks are deployed in order to provide wide area coverage, while the WLANs access technology is used to cost-effectively boost capacity in local hot spots such as airports or conference centers. This is a very interesting topic. Our future research will focus on investigating the analytical models for such integrated networks.

A Glossary

ACK	Acknowledgement (packet)
AP	Access Point
BSS	Basic Service Set
CBR	Constant Bit Rate
CFP	Contention Free Period
CP	Contention Period
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance
CTS	Clear To Send
CW	Contention Window
DCF	Distributed Coordination Function
DIFS	Distributed IFS
DS	Distribution System
DSSS	Direct Sequence Spread Spectrum
EDCF	Enhanced Distribution Coordination Function
EIFS	Extended IFS
ESS	Extended Service Set
FHSS	Frequency Hopping Spread Spectrum
FS	Free Space
FTP	File Transfer Protocol
GPS	Generalized PS
GPRS	General Packet Radio Service
GSM	Global System for Mobile Communication
IEEE	Institute of Electrical and Electronics Engineers
IEK	Industrial Economics and Knowledge Center
IFS	Interframe Spacing
IP	Internet Protocol
ISM	Industry, Science and Medical
ITRI	Industrial Technology Research Institute
ITU-T	International Telecommunication Union, Telecom standardization
LAN	Local Area Network
LLC	Link Logic Control

MAC	Medium Access Control
NAV	Network Allocation Vector
OSI	Open Systems Interconnection
PC	Point Coordinator
PCF	Point Coordination Function
PS	Processor Sharing
PPDU	PHY protocol data units
QoS	Quality of Service
RTS	Request To Send
RTT	Round Trip Time
SIFS	Short Inter Frame Spacing
STA	(Wireless) Station
TC	Traffic Class
TCP	Transmission Control Protocol
TRG	Two-Ray Ground
TP	Throughput
UDP	User Datagram Protocol
WiMAX	Worldwide Interoperability for Microwave Access
WLAN	Wireless Local Area Networks

Appendix

Simulation in NS-2

To simulate a wireless scenario, usually a *tcl* script is written for simulation and there are several factors that are to be considered. The topology of the network is given as nodes like: one base station BS(0), some mobile nodes node_(0), node(1), … ,node(n). The mobile node moves within an area whose boundary is defined in this example as 1000mX1000m. A TCP connection is setup between the base station and mobile node. Packets are exchanged between the nodes as they come within hearing range of one another. As they move away, packets start getting dropped.

A mobile node consists of network components the like Link Layer (LL), the Interface Queue (IFQ), the MAC layer, the wireless channel nodes transmit and receive signals from etc, which is explained in the beginning of this chapter. At the beginning of a wireless simulation, we need to define the type for each of these network components. Additionally, we need to define other parameters like the type of antenna, the radio-propagation model, the type of ad-hoc routing protocol used by mobile nodes etc. We begin the Tcl script with a list of these different parameters described above, as follows:

```
# ==================================================================
# Define options
# ==================================================================
set val(chan)           Channel/WirelessChannel    ;# channel type
set val(prop)           Propagation/TwoRayGround
                                         ;#radiopropagation model
set val(netif)          Phy/WirelessPhy
                                         ;# network interface type
set val(mac)            Mac/802_11                 ;# MAC type
set val(ifq)             Queue/DropTail/PriQueue
                                            ;# interface queue type
set val(ll)             LL                        ;# link layer type
set val(ant)            Antenna/OmniAntenna       ;# antenna model
set val(ifqlen)         50                     ;# max packet in ifq
set val(adhocRouting)   DumbAgent               ;# routing protocol
set val(nn)             3                  ;# number of mobilenodes
set num_bs_nodes        1        ;# number of base stations
# ==================================================================
```

In this example we have the number of mobile nodes as 3, but for large networks we can set it to higher numbers like:

```
    set val(nn)             300   ;# number of mobilenodes
```

And then, we should configure the lower physical layer parameters, for example the height of the antenna, the radio frequency, the transfer energy etc. as follows:

```
# ==================================================================
# Unity gain, omni-directional antennas
# Set up the antennas to be centered in the node and 1.5 meters
above it
# ==================================================================
Antenna/OmniAntenna set X_ 0       ;# X dimension of the Antenna
Antenna/OmniAntenna set Y_ 0       ;# Y dimension of the Antenna
Antenna/OmniAntenna set Z_ 1.5    ;# Z dimension of the Antenna
Antenna/OmniAntenna set Gt_ 1.0   ;# Transmit antenna Gain
Antenna/OmniAntenna set Gr_ 1.0   ;# Receive antenna Gain
# ==================================================================
# Initialize the SharedMedia interface with parameters to make
```

```
# it work like the 2412 MHz Lucent WaveLAN DSSS radio interface
#===================================================================
Phy/WirelessPhy set CPThresh_ 10.0       ;# Capture threshold
Phy/WirelessPhy set CSThresh_ 1.559e-11  ;# Carrier sense threshold
Phy/WirelessPhy set RXThresh_ 3.652e-10  ;# Receiving threshold
Phy/WirelessPhy set bandwidth_ 2e6       ;# Bandwidth
Phy/WirelessPhy set Pt_ 0.28183815       ;# transmission power
Phy/WirelessPhy set freq_ 2412e+6        ;# Radio frequency
Phy/WirelessPhy set L_ 1.0           ;# System loss
```

After declaring these parameters, the instance of the Simulator, trace files and the topology are defined.

```
#===================================================================
# Initialize Global Variables
# create simulator instance
set ns_    [new Simulator]

# create trace object for ns and nam
set tracefd     [open $opt(tr) w]
set namtrace    [open $opt(nam) w]

$ns_ trace-all $tracefd
$ns_ namtrace-all-wireless $namtrace $val(x) $val(y)

# set up topography object
set topo       [new Topography]

# define topology
$topo load_flatgrid $val(x) $val(y)

# Create God
create-god [expr $val(nn) + $num_bs_nodes]
#==================================================================
```

While running wireless simulations in NS-2, we can set four types of traces: the Agent Trace, the Router Trace, the Mac Trace, and the Movement Trace. We can experiment with traces by turning them on. For example: in our simulation, we care about the performance of the TCP layer and also the interaction between TCP and MAC, so we only turned the agent trace and the Mac trace on as follows:

```
# =================================================================
# configure API for mobile nodes
$ns_ node-config        -agentTrace ON \
                        -routerTrace OFF \
                        -macTrace ON \
                         -movementTrace OFF
```

There are two ways to define the mobility of the nodes. In the case of simple node movement, we can just define it inside the Tcl scripts. If there are a large number of nodes and complex movements, we better define them in a separated file and load them into the Tcl scripts. For example, in our simulation, the nodes exhibit no mobility so that we turn off any random motion or any kind of other motion. The nodes are given an initial position as follows:

```
#===================================================================
for {set i 0} {$i < $val(nn) } {incr i} {
    set node_($i) [$ns_ node]
    $node_($i) random-motion 0    ;# disable random motion
    # Provide initial co-ordinates for mobilenodes
    $node_($i) set X_ $val(st_x)
    $node_($i) set Y_ $val(st_y)
    $node_($i) set Z_ 0.0
}
```

And then, we can use such a command to produce some node movements:

```
$ns_ at 50.0 "$node_(0) setdest 25.0 20.0 15.0"
```

It means that at time 50.0s, node0 starts to move towards the destination (x=25,y=20) at a speed of 15m/s. This API is used to change direction and speed of movement of the mobile nodes. In our simulations, we need to generate the Poisson user arrival process. As the Poisson process definition, the interval times between events are independent, identically exp(λ) distributed. We use such codes as follow to define the nodes' mobility:

```
#====================================================================
# create exponential distributed inter arrival time
set interArrival [new RandomVariable/Exponential]
$interArrival set avg_ [expr 1.0/$val(lambda)]

set next_(0) $val(move-start)
# create the following poisson user arrival process
for {set i 1} {$i < $val(nn) } {incr i} {
    # produce node movements
    set inter [$interArrival value]
        set next_($i) [expr $next_([expr $i-1])+$inter]
    puts $tracefd "the node_($i) will move at  $next_($i)"
    $ns_ at $next_($i) "$node_($i) setdest $val(dst_x) $val(dst_y)
$val(v)"
}
```

Now the traffic flows between the Access point and the mobile nodes are set with sources as the FTP Application Agent with TCP Agents. The receivers are set as the TCPSink Agents. Such traffic flows between Base station (Access Point) and node_(n) look like this.

```
#====================================================================
for {set i 1} {$i < $val(nn) } {incr i} {
    # Setup traffic flow between nodes
    # TCP connections between node_($i) and BS(0)
        set tcp_($i) [new Agent/TCP]
        $tcp_($i) set class_ 2
        set sink_($i) [new Agent/TCPSink]
        $ns_ attach-agent $BS(0) $tcp_($i)
        $ns_ attach-agent $node_($i) $sink_($i)
        $ns_ connect $tcp_($i) $sink_($i)
        set ftp_($i) [new Application/FTP]
        $ftp_($i) attach-agent $tcp_($i)
        $ns_ at $next_($i) "$ftp_($i) start"
}
```

At the end, the stop time is defined when the nodes reset themselves. The procedure stop{} is called to flush out traces and close the trace file. And finally the simulation is run.

```
#====================================================================
# Tell all nodes when the simulation ends
for {set i } {$i < $val(nn) } {incr i} {
    $ns_ at $val(stop).0 "$node_($i) reset";
}
$ns_ at $val(stop).0 "$BS(0) reset";

$ns_ at $val(stop).001 "stop"
$ns_ at $val(stop).002 "puts \"NS EXITING...\" ; $ns_ halt"

proc stop {} {
    global ns_ tracefd namtrace
#   $ns_ flush-trace
    close $tracefd
    close $namtrace
}
puts "Starting Simulation..."
$ns_ run
```

Bibliography

[802.11 b] Supplement to 802.11-1999, Wireless LAN MAC and PHY specifications: Higher Speed Physical Layer (PHY) extension in the 2.4 GHz band, IEEE Std., Sep 1999.

[Alonso 02] P. L. C. Alonso, "Technical and Business comparison of Cellular Wireless LAN and Bluetooth", 2002, Available from http://internetng.dit.upm.es/ponencias-jing/2002/chas/v1.PDF.

[Babich 00] F. Babich and G. Lombardi, "Statistical Analysis and Characterization of the Indoor Propagation Channel", IEEE Trans. Commun., vol. 48, no. 3, Mar. 2000, pp. 455–64.

[Bianchi 00] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function", IEEE Journal on Sel. Areas in Comm., vol. 18, pp. 535-547, 2000.

[Burmeister 04] C. Burmeister, U. Killat, "Location-Based Services via IEEE 802.11b Hot Spots", Proceedings of the 3rd Polish-German Teletraffic Symposium PGTS 2004, Dresden, Germany, pp. 93—98, Sep. 2004.

[Cesana 03] Matteo Cesana, Lecture notes "An overview of WLAN Standards", 2003. Available from http://www.keskus.hut.fi/opetus/s38118/s00/tyot/25/page1.shtml .

[CityLan] CityLan by Agile communications, Available from http://www.citylan.com.au/ .

[ETSI 01] ETSI, "Digital Cellular Telecommunications System (Phase 2+); General Packet Radio Service; Service Description; Stage 2", GSM 03.60., 2001.

[Fall 97] Kevin Fall and Kannan Varadhan, editors. "*ns* notes and documentation", The VINT Project, UC Berkeley, LBL, USC/ISI, and Xerox PARC, November 1997. Available from http://www.isi.edu/nsnam/ns/

[Heyman 97] D. P. Heyman, T. V. Lakhsman, and A. L. Neidhardt, "A new method for analysing feedback–based protocols with applications to engineering web traffic over the Internet", Proc. of Sigmetrics, Seattle, 1997.

[HipGF] Hiperlan/2 Global Forum. Available from http://www.hiperlan2.com .

[IEEE 99] IEEE standard for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, IEEE Std., Aug 1999, http://standards.ieee.org/getieee802/download/802.11-1999.pdf .

[Jackson 57] J.R.Jackson, "Networks of waiting lines", Operations Research,5, 518-521, 1957.

[Jacobson 88] V. Jacobson, "Congestion avoidance and control", In. Proceedings of SIGCOMM '88 (Stanford, CA, Aug.1988), ACM.

[Jacobson 90] V. Jacobson. "Modified TCP Congestion Control Avoidance Algorithm", End-2-end-interest mailing list, April 30, 1990.

[Kelly 01] F.P. Kelly, "Mathematical modeling of the Internet", Mathematics Unlimited—2001 and Beyond, B. Engquist and W. Schmid, Eds, Springer—Verlag, Berlin, 2001.

[Kleinrock 67] L. Kleinrock, "Time-shared systems: A theoretical treatment," Journal of the Association for Computing Machinery, vol. 14, no. 2, pp. 242– 261, 1967.

[Kleinrock 76] L. Kleinrock, "Queueing Systems", vol. 1. Theory, John Wiley and Sons, 1976.

[Kleinrock 76] L. Kleinrock, "Queueing Systems", vol. 2. Computer applications, John Wiley and Sons, 1976.

[Lassila 03] P. Lassila, J.L. van den Berg, M. Mandjes and R.E. Kooij, "An integrated packet/flow level model for TCP performance analysis", Proceedings of ITC 18, Berlin, Germany, 2003.

[Litjens 00] R. Litjens and R.J. Boucherie, "Radio resource sharing an a GSM/GPRS network", Proceedings ITC Specialists Seminar on Mobile Systems and Mobility, March 22-24, Lillehammer, Norway, P.J. Emstad (Editor) pp. 261--274, 2000.

[Litjens 03_1] R. Litjens, "Capacity Allocation in Wireless Communication Networks: Models and Analysis", ISBN 90-9017132-20, Febodruk, Enschede, The Netherlands, 2003.

[Litjens 03_2] R. Litjens, F. Roijers, J. L. van den Berg, R. J. Boucherie and M. Fleuren, "Performance analysis of Wireless LANs: An integrated packet/flow level approach", Proceeding of ITC 18, Berlin, Germany, 2003.

[Lu 97] S. Lu, V. Bharghavan, and R. Srikant, "Fair scheduling in wireless packet networks", *ACM Sigcomm'97,* Cannes, France, Sep. 1997.

[Mathis 97] M. Mathis, J. Semke, J. Mahdavi and T. Ott, "The macroscopic behavior of the TCP congestion avoidance algorithm", Computer Communication Review, vol. 27, no. 3, pp. 67-82, 1997.

[McCullagh 05] D. McCullagh, "Intel ships wireless broadband chips", April 18, 2005. Available from http://news.com.com/Intel+to+release+WiMax+chip/2100-1006_3-5674136.html?tag=nefd.top .

[Miorandi 04] D. Miorandi, A. A. Kherani and E. Altman, "A Queueing Model for HTTP Traffic over IEEE 802.11 WLANs", ITC Specialist Seminar on Performance Evaluation of Wireless and Mobile Systems, Antwerp, Sep. 2004.

[Neskovic 00] A. Neskovic, N. Neskovic, and G. Paunovic, "Modern Approaches in Modeling of Mobile Radio Systems Propagation Environment", *IEEE Commun. Surveys*, vol. 3rd Quarter, Sept. 2000, pp. 2–12.

[NTT 04] Paul Kallender. "DoCoMo to release 3G/WLAN mobile phone". Available from http://www.nwfusion.com/news/2004/1115docomtore.html .

[NS-2] Network Simulator 2, Available from http://www.isi.edu/nsnam/ns .

[Padhye 98] J. Padhye, V. Firoiu, D. Towsley , J.Kurose, "Modeling TCP throughput: A simple model and its empirical validation", University of Massachusetts Technical Report, CMPSCI TR 98-008.

[Pareek 05] D. Pareek, "Who will win the wireless broadband battle – WiFi, WiMax or Cellular fixed wireless", 2005. Available from http://www.wimaxtrends.com/articles/feature/f032805b.htm .

[Peters 05] J.W. Peters, "United Airlines Approved for In-Flight Internet Service", Jun. 2005. Available from http://www.nytimes.com/2005/06/06/technology/06united.html .

[Punnoose 00] Ratish J. Punnoose, Pavel V. Nikitin, and Daniel D. Stancil. "Efficient Simulation of Ricean Fading within a Packet Simulator", Vehicular Technology Conference, Sept 2000. Available from http://www.ece.cmu.edu/wireless/.

[Rappaport 02] T. S. Rappaport, "Wireless Communications: Principles and Practice", Prentice Hall, 2002.

[Reardon 05] M. Reardon, "The citywide Wi-Fi reality check", May 27, 2005. Available from http://news.com.com/The+citywide+Wi-Fi+reality+check/2100-7351-722150.html?part=dht&tag=ntop&tag=nl.e703 .

[Roberts 00] J. Roberts and L. Massoulie, "Bandwidth sharing and admission control for elastic traffic", Telecommunication Systems, vol. 15, pp. 185–201, 2000.

[Simon 98] M. Simon and M.-S. Alouini, "A unified approach to the performance analysis of digital communication over generalized fading channels", Proc. IEEE, vol. 86, pp. 1860--1877, Sept. 1998.

[Sklar 97] Bernard Sklar, "Rayleigh Fading Channels in Mobile Digital Communication Systems, Part I: Characterization", IEEE Communications Magazine, vol. 35, pp. 90 - 100, Jul. 1997.

[Sklar 97] Bernard Sklar, "Rayleigh Fading Channels in Mobile Digital Communication Systems, Part II: Mitigation", IEEE Communications Magazine, vol. 35, pp. 102 - 109, Jul. 1997.

[Stevens 97] W. Richard Stevens. "TCP slow start, congestion avoidance, fast retransmit, and fast recovery algorithms", Rfc, IETF, January 1997.

[Swisscom 04] "Swisscom Mobile complements WLAN and UMTS with EDGE", 2004, http://www.3g.co.uk/PR/April2004/6866.htm.

[Tech 05] Tech Trends,2005.Available from http://www.telecom.globalsources.com/gsol/I/Mobile-wireless/a/9000000060453.htm .

[TCP] IETF, *RFC793: Transmission Control Protocol*. Available from http://www.faqs.org/rfcs/rfc793.html.

[Tijms 94] H.C. Tijms. "Stochastic Models: An Algorithmic Approach", John Wiley & Sons, 1994.

[Tobagi 75] F.A. Tobagi and L. Kleinrock, "Packet Switching in radio channels: Part 2 - the hidden node problem in carrier sense multiple access modes and the busy tone solution", IEEE Transactions on Communications, vol.23, no. 12, pp.1417-1433, 1975.

[Weinmiller 97] J. Weinmiller, M. Schlager, A. Festag, and A. Wolisz, "Performance study of access control in wireless LANs IEEE 802.11 DFWMAC and ETSI RES 10 HIPERLAN", *Mobile Networks and Applicant.*, vol. 2, pp. 55–67, 1997.

[Winands 04] E.M.M. Winands, T.J.J. Denteneer, J.A.C. Resing and R. Rietman, "A finite-source feedback queueing network as a model of the IEEE 802.11 distributed coordination function", Proceedings of European Wireless'04, Barcelona, Spain , 2004.

[Wu 02] H. Wu, Y. Peng, K. Long, S. Cheng and J. Ma, "Performance of reliable transport protocol over IEEE 802.11 wireless LAN: analysis and enhancement", Proceedings of IEEE INFOCOM '02, New York, USA, 2002.