Chapter 5

WIRELESS INTERNET: IEEE 802.11B

Abstract

This tutorial article describes the IEEE 802.11b Wireless Local Area Network (WLAN) standard, which is commonly referred to as "WiFi". This standard offers up to 11 Mbps of transmission capacity at the physical layer of the protocol stack, and is one of the key enabling technologies for wireless Internet, mobile computing, and ad hoc networking applications. After introducing the standard and its features, the latter part of the article discusses protocol interactions that occur when popular Internet applications, such as multimedia streaming and the World Wide Web, operate over IEEE 802.11b WLANs. These interactions can lead to performance problems in the TCP/IP Internet protocol stack.

1. Introduction

Two of the most exciting and fastest-growing Internet technologies in recent years are the World Wide Web and wireless networks. The Web has made the Internet available to the masses, through its TCP/IP protocol stack and the principle of layering: Web users do not need to know the details of the underlying communication protocols in order to use network applications. Wireless technologies have revolutionalized the way people think about networks, by offering users freedom from the constraints of physical wires. These technologies are available today, in laptop or handheld form, at relatively modest cost. Mobile users are interested in exploiting the full functionality of the technology at their fingertips, as wireless networks bring closer the "anything, anytime, anywhere" promise of mobile networking.

One of the primary challenges in this new networking context is "performance transparency": providing an end-user Internet experience that is hopefully no worse than that in the traditional wired-Internet desktop environment. Significant advances are taking place in both wired and wireless networking environments that substantially increase the raw bit rate available at the physical layer. However, these advances are of little value if the extra bandwidth cannot

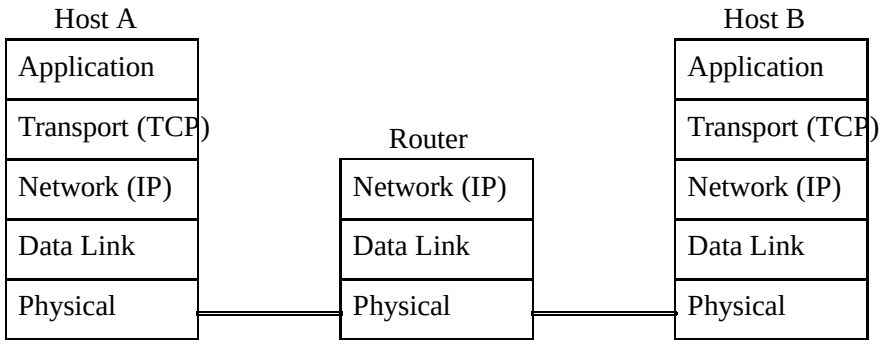

Figure 5.1. Illustration of the Internet TCP/IP Protocol Stack

be delivered all the way up to the application layer. In some cases, performance problems occur at intermediate layers of the protocol stack.

This tutorial focuses on one particular wireless networking technology, namely the IEEE 802.11b Wireless Local Area Network (WLAN) standard [ANSI 1999], and the protocol performance issues that arise in that environment. The first part of this tutorial provides an overview of IEEE 802.11b WLAN protocols, as well as TCP/IP protocols, and some popular Internet applications used on wireless LANs. The last part of the tutorial focuses on protocol performance issues for wireless Internet applications. To illustrate the issues, practical examples are used. These include wireless TCP performance, multimedia streaming, TCP performance in multi-hop ad hoc networks, and Web performance in wireless ad hoc networks.

2. Background

Figure 14.1 provides an illustration of the Internet protocol stack [tanenbaum]. A protocol stack provides a modular architecture and a conceptual framework for discussing communication protocols and their functionality. Note that this diagram shows only a 5-layer protocol stack, compared to the 7-layer protocol stack in the classic OSI network reference model [tanenbaum].

The lowest layer of the protocol stack is the *Physical Layer*. The physical layer deals with the raw transmission of bits between two communicating devices. Many different transmission media are possible at the physical layer, including wired (guided) media such as twisted pair (copper), coaxial cable, or optical fiber, and wireless (unguided) media such as microwave, satellite, IR (Infra-Red), or RF (Radio Frequency) transmission. The physical layer

performs the signalling and modulation required to encode information (e.g., binary 0's and 1's) on the channel, by varying physical characteristics of the signal (e.g., amplitude, frequency, phase). The coding techniques used are highly dependent upon the properties of the transmission medium chosen at the physical layer.

The next layer up the protocol stack is called the *Data Link Layer*, or the Link Layer for short. This layer deals with a larger logical unit called a *frame*. A frame typically carries several hundred or several thousand bits. Frames may be fixed-size or variable-size, depending on the specific networking technology being used. For example, Asynchronous Transfer Mode (ATM) networks use fixed-size frames called *ATM cells*, while Ethernet and IEEE 802.11b LANs allow variable-size frames, with upper and lower limits on the legal frame sizes permitted.

The Link Layer provides two main services. First, it regulates access to the channel amongst the contending stations. In a broadcast network, this *Medium Access Control* (MAC) mechanism is important, since at most one station can successfully transmit on the shared channel at a time. In a point-to-point network, the MAC protocol has a very minor role, since each link has only two endpoints. Second, the Link Layer provides framing, flow control, and error control services, to provide reliable hop-by-hop communication. Commonly-used mechanisms at this *Logical Link Control* (LLC) sublayer are checksums, sequence numbers, acknowledgements (ACKs), timeouts, and retransmissions.

The *Network Layer* of the protocol stack builds upon the Link Layer services, by adding addressing, routing, and internetworking functionality at the *packet* level. Addressing uniquely identifies any endpoint host on the network. Routing determines a path for reaching a destination. Internetworking support allows communication across different networks by defining how to translate packet formats and how to accommodate diverse packet sizes across heterogeneous networking technologies. In the Internet, the Network Layer protocol is called the Internet Protocol (IP). It provides a "best effort" datagram delivery model. Most of the IP packets that are sent will correctly arrive at the intended destination, but there is no guarantee that they will do so. Packets are sometimes delayed, lost, duplicated, or corrupted in transit.

The *Transport Layer* provides end-to-end services between two communicating entities on the Internet. While IP routing gets a packet to the correct host, an additional layer of transport-level addressing (e.g., port numbers) is needed to deliver data to the correct recipient (of many possible recipients) on that host. The Transmission Control Protocol (TCP) on the Internet is one example of a Transport Layer protocol. It provides end-to-end reliable data delivery. More details on TCP are provided in Section 4. Another example is the User Datagram Protocol (UDP), which is a minimal mechanism transport-layer protocol. It provides a connection-less service model similar to IP.

The topmost layer of the Internet protocol stack is the *Application Layer*. Many user-level network applications reside here: electronic mail, file transfer, network news, media streaming, peer-to-peer, and the World Wide Web. Each of these applications has a well-defined application-layer protocol, such as SMTP (Simple Mail Transfer Protocol), FTP (File Transfer Protocol), or HTTP (Hyper-Text Transfer Protocol). These protocols offer services to end users of the Internet.

This layered Internet protocol stack model provides a reference point for the discussion of IEEE 802.11b protocols in the next section, as well as the discussion of TCP/IP protocol performance issues later in the article.

3. The IEEE 802.11b WLAN Standard

One of the most popular solutions for wireless Internet access today is the IEEE 802.11b wireless local area network standard [ANSI 1999], commonly referred to as "WiFi" (Wireless Fidelity). This section provides information on the basic operation of IEEE 802.11b, as well as some of its features.

Overview

Wireless networking refers to the use of infrared (IR) or radio frequency (RF) signals to transmit information between devices, without requiring physical cabling between them. Commonplace examples are using remote control devices to change the channel on your television, open your garage door, or advance the slides on your laptop during a conference presentation. Simple wireless devices convey only control information (e.g., open or close your garage door), while more sophisticated ones allow the transmission of arbitrary data (e.g., using a wireless keyboard to interact with the TV/computer in your hotel room, or using your personal digital assistant (PDA) to 'beam' your business card and contact information to your colleague's PDA).

Wireless networking solutions typically require line-of-sight transmission, or at least close proximity to the devices being controlled. For example, your TV remote control does not work very well from the bathroom, and your garage door opener does not work when you are several blocks away from your house. One reason is the limited transmit power used by the source, which curtails the physical distance that an intelligible signal can propagate (e.g., the signal strength typically diminishes proportionally with the square of the distance travelled). The limit on transmit power is often regulated by the federal government to reduce the interference between devices operating in different jurisdictions, and to minimize health and safety concerns. A second reason is the operating frequency (in Hertz (Hz)) used in the electromagnetic spectrum. Some signals (e.g., broadcast radio, RF, X-rays) are able to pass through solid objects (e.g., walls, humans), while other signals (e.g., IR, visible light) are not. This is why

a household baby monitor can be used to listen to an infant sleeping in the nursery upstairs, while your TV remote control cannot turn down the volume on the TV in the apartment next door to your own.

The IEEE 802.11b WLAN standard is an attractive and popular wireless networking solution based on these principles. IEEE 802.11b offers physical layer data rates of up to 11 Mbps, which makes it a cost-effective LAN solution similar to the classic 10 Mbps Ethernet LAN. This "WiFi" LAN solution is attractive in business, education, and research environments because it enables tetherless access to the Internet. With the wireless network card commonplace in laptops today, users can roam from office to office or lab to lab in their organization while still maintaining network connectivity for email, Web browsing, or other Internet-related activities. WiFi "hotspots" are widely deployed in many cities (e.g., at airports, hotels, coffee shops, and bookstores) for general Internet access.

The following sections provide greater detail on IEEE 802.11b.

Physical Layer

The IEEE 802.11 Working Group has developed an entire family of IEEE 802.11 protocols, which continues to grow and evolve today. The standards define both the Physical Layer and the Data Link Layer operation for IEEE 802.11 protocols.

The first version of the IEEE 802.11 standard supported a data rate of 1 Mbps, using a physical layer transmission technique called Frequency Hopping Spread Spectrum (FHSS). Later improvements doubled the data rate to 2 Mbps, while still maintaining backward compatibility with the 1 Mbps version. These standards were developed for both FHSS and DSSS (Direct-Sequence Spread Spectrum) transmission at the physical layer.

Within the DSSS portion of the IEEE 802.11 family, a high data rate extension called IEEE 802.11b was defined. This extension supports higher data rates of 5.5 Mbps and 11 Mbps, using more sophisticated modulation schemes for physical layer transmission. This standard also maintains backward compatibility with earlier devices, by supporting the 1 Mbps and 2 Mbps data rates in the original IEEE 802.11 standard.

The operating frequencies for IEEE 802.11b are in the Industrial, Scientific, and Medical (ISM) band of the electromagnetic spectrum, near 2.4 GHz. Specifically, the frequency band ranges from 2400 MHz to 2483.5 MHz. Many vendors have produced products that operate in this frequency range, including IEEE 802.11b, IEEE 802.11g, Bluetooth, baby monitors, microwave ovens, Home RF, and some cordless phones. Interference from other devices is a prevalent concern for IEEE 802.11b WLANs, since it can lead to unpredictable WLAN performance.

Within the assigned portion of the electromagnetic spectrum, IEEE 802.11b devices can choose from 14 “channels”, each approximately 22 MHz wide. However, many of these channels overlap. Only channels 1, 6, and 11 are non-overlapping. The choice of channels can be manually configured, so that multiple WLANs in close proximity do not interfere with each other. An IEEE 802.11b device can automatically detect when it is in the range of more than one wireless Access Point (AP), and dynamically select the AP that provides the strongest signal. (See Section 3.0.0 for more details on the Infrastructure Mode of operation.)

The commonly used transmit power for IEEE 802.11b is 100 milliWatts, which typically provides about 100 meters of omni-directional coverage. Some users have been able to achieve much greater distance coverage (e.g., several kilometers) using higher transmit power and directional antennas.

A recent modification to IEEE 802.11b is the IEEE 802.11g standard. IEEE 802.11g offers data rates up to 54 Mbps, using the same basic technology as IEEE 802.11b, and the same busy portion of the electromagnetic spectrum (2.4 GHz). Many chip sets produced today support both IEEE 802.11b and 802.11g.

The follow-on to IEEE 802.11b is IEEE 802.11a, which offers data rates up to 54 Mbps. Two important differences exist between IEEE 802.11b and 802.11a. First, IEEE 802.11a operates in the 5 GHz band of the electromagnetic spectrum, which is distinct from that of IEEE 802.11b. This new frequency band is attractive because there is (so far) less interference in this portion of the spectrum. However, not all vendors have commodity chip sets designed for operation in this regime yet, so the product prices are significantly higher than IEEE 802.11b. Second, IEEE 802.11a uses a different physical layer modulation technique called Orthogonal Frequency Division Multiplexing (OFDM). This technique provides greater error resilience than DSSS.

Channel Access Protocols

The IEEE 802.11b standard defines three channel access protocols that can be used at the Medium Access Control (MAC) sublayer. These protocols are called Distributed Coordination Function (DCF), Request-To-Send/Clear-To-Send (RTS/CTS), and Point Coordination Function (PCF).

Distributed Coordination Function (DCF).. The most commonly used MAC protocol option, and the default in most IEEE 802.11b WLAN deployments, is DCF. This protocol defines a distributed algorithm that allows multiple stations to compete for the use of a single broadcast channel in the wireless coverage area, which is called a *cell*. All wireless devices in the cell must share use of the same channel.

The fundamental rule that must be obeyed in the WLAN cell is that at most one successful frame transmission can be in progress at a time on the network.

If no stations transmit, the channel is idle. If exactly one station transmits, there is a very high probability that the receiver will receive the frame successfully. If two or more stations transmit, the result is a *collision* on the channel, which results in unintelligible data for the receivers. Such colliding frames waste network resources, since they require retransmission at a later time for successful delivery.

The purpose of the MAC protocol is to determine which station is allowed to transmit, particularly when multiple stations have frames ready for transmission. Desired properties for the MAC protocol include: a low channel access delay for acquiring the channel; a low collision rate on the network, so that few retransmissions are required; high efficiency under high load, so that the maximal network throughput can be achieved; and fairness, so that each station is equally likely to acquire the channel when it is available.

The DCF protocol used is called Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA). The "Multiple Access" part of this name refers to the coordination problem defined previously: multiple stations competing for use of a single shared transmission channel. The "Carrier Sense" part of this name identifies an important aspect of the protocol: the stations can listen to the channel to see if it is available or not prior to transmitting a frame. In particular, a ready station with a frame to transmit must first listen to the channel. If the channel is idle, the station can then begin transmitting its frame. If the channel is busy, the station must defer (i.e., refrain from sending), because transmitting would surely cause a collision on the network (hence the term "Collision Avoidance" in the name of the protocol).

Randomization is also an important part of the channel access protocol. A variable called the Contention Window (CW) is used for this purpose. A station wishing to transmit a frame chooses a random BackOff (BO) time between 0 and CW, and this extra BO delay must elapse before the actual frame transmission can begin. If the channel is busy and collisions are observed, stations dynamically increase (e.g., double) CW. If frame transmissions are routinely successful, CW can be reset to its default value CW_{min}.

The CSMA/CA protocol reduces the number of collisions on the channel, but does not eliminate collisions entirely. For example, if two stations become ready at exactly the same time, and both sense the channel idle, then both could start transmission at the same time, and collide with each other.

Handling this type of collision problem is tricky. In an Ethernet LAN environment, a transmitting station can use its transceiver (transmitter/receiver) to listen to the channel during its own outgoing frame transmissions. This property allows a station to detect discrepancies between what it was trying to send and what was actually observed on the wire. Discrepancies between the two indicate a collision on the network (i.e., more than one station transmitting at the same time). A station detecting such a collision aborts the transmission of

its frame, and generates a noise burst on the wire for all stations to hear. This protocol is called CSMA/CD, with the CD standing for Collision Detection. It is used in Ethernet wired LANs.

Unfortunately, the Collison Detection (CD) mechanism is not applicable for IEEE 802.11b WLANs. Physically, stations can either transmit or receive using their antenna, but they cannot do both at the same time. Even if they could, the transmit power would be so dominant that it would be almost impossible to detect a received signal. Further complicating matters are the noisy characteristics of the wireless propagation environment: not all stations may hear the collision if there was one, and some stations may hear a collision even if there wasn't one.

The IEEE 802.11b DCF protocol solves this problem with a combination of mechanisms: acknowledgements, timeouts, and retransmissions. Upon the successful arrival of a frame at the intended recipient, the receiver sends back a control frame with a positive acknowledgement to the sender. This acknowledgement tells the sender that its frame transmission was successful. In the absence of the acknowledgement, the sender will retransmit another copy of the same frame after a randomly chosen short timeout interval (e.g., up to 1 millisecond). This mechanism handles collision-related losses just the same as corruption-related losses due to wireless channel errors. It also recovers from the loss of either the data frame or its acknowledgement. In both cases, a frame retransmission is required. If repeated retransmissions are required for the same data frame, the timeout interval is repeatedly doubled, up to a maximum limit CW_{max}. If the maximum number of retries (e.g., default 8 in most implementations) is reached, then the frame transmission is aborted. Further error recovery is left to higher-layers of the protocol stack.

To ensure that the recipient of a successful frame can acquire the channel to send an acknowledgement, the IEEE 802.11b standard defines two separate time intervals. The Short Inter-Frame Space (SIFS) is the amount of time that the recipient waits before sending its acknowledgement. This time is 28 microseconds (μsec). The Distributed Inter-Frame Space (DIFS) is the amount of time that a station must observe a quiescent channel before concluding that it is idle, and proceeding with its random BackOff (BO) and frame transmission. This DIFS time is 128 μsec. With these settings, the recipient of a successful frame is the first station entitled to use the shared channel to send back an acknowledgement. This Positive Acknowledgement with Retransmission (PAR) protocol is used only for unicast (one-to-one) frames on the WLAN. It is not used for multicast or broadcast frames that are addressed to many recipients.

Figure 5.2 summarizes the CSMA/CA DCF protocol. The diagram illustrates an example frame transmission, say from station A to station B. After sensing the channel idle for the DIFS period, and waiting for its random BO period, station A grabs the channel and transmits its data frame. After receiving the

Figure 5.2. Illustration of CSMA/CA DCF in IEEE 802.11b WLAN

frame successfully, station B waits for the SIFS period and then sends its positive ACK to A.

Request-To-Send/Clear-To-Send (RTS/CTS).. The RTS/CTS protocol is designed to handle the *hidden node* problem that can arise in wireless networks. Consider three stations called A, B, and C, who happen to be geographically in alphabetical order in the network. Depending on the relative distances between the nodes, it is possible that A and B can hear each other, that B and C can hear each other, but A and C cannot hear each other. In such a scenario, A might transmit a frame to B at the same time that C transmits a frame to B. The result is a collision (and unintelligible garbage) at B.

Neither A nor C is aware of this collision problem, since they are not within range of each other. Their only clue is the lack of a positive ACK from B, which will soon trigger retransmissions of the colliding frames. The larger the frame size, the greater the proportion of time wasted on collisions, and the greater the probability that the retransmitted frames will also collide. From A's viewpoint, C is a hidden node, and from C's viewpoint, A is a hidden node. The hidden node problem can dramatically degrade wireless channel performance.

The RTS/CTS protocol resolves this problem by having the intended receiver node (B) manage the shared channel amongst its neighbours (A and C, in this example). In particular, the neighbours must make advance reservations of the channel for their transmissions. For example, node A sends to node B a short control frame called the Request-To-Send (RTS) frame. The RTS indicates the size of the frame that A wants to send, and the intended recipient B. If this is the only RTS request that node B receives, then node B can send a short Clear-To-Send (CTS) control frame to A to grant its request. The broadcast nature of this CTS transmission tells node A to go ahead with its frame transmission, while also telling node C (since C can hear B) to refrain from sending. The information contained in the CTS frame conveys a Network Allocation Vector (NAV) that expresses the amount of channel time required for A and B to complete the exchange of their data frame and acknowledgement. Station C simply defers from accessing the channel for this NAV interval, and thus avoids colliding with A's frame.

Figure 5.3. Illustration of RTS/CTS in IEEE 802.11b WLAN

Similarly, node C can initiate a frame transmission to B using the RTS/CTS exchange. If node B receives multiple RTS requests, it can choose which one to grant, as long as it has at most one CTS outstanding at a time. Because RTS control frames are very short, it is unlikely for them to collide. Nevertheless, if they do collide, the random timeout and retransmission mechanisms described previously resolve this. A station that does not receive a CTS response to its RTS request after a maximum number of RTS retries will abort its attempted transmission of the current frame.

Figure 5.3 provides an illustration of the RTS/CTS protocol. Again, assume that the frame transmission is from A to B. Once A receives the CTS response to its RTS request, it can initiate its frame transmission. The receiving station B sends an ACK to A upon receipt of a successful frame.

Some implementations of IEEE 802.11b trigger the use of RTS/CTS automatically when the average number of collisions in the DCF protocol exceeds a threshold. Other implementations require manual selection of either the DCF or RTS/CTS protocol.

Point Coordination Function (PCF).. The two foregoing MAC protocols for IEEE 802.11b WLANs are not suitable for real-time applications, since there is no bound on the maximum latency for channel access and frame transmission. That is, there is no guarantee of when (or even if) a data frame will be successfully transmitted across the network. These protocols should not be used for hard real-time control systems applications (e.g., blast furnace operation, nuclear power plant, braking system on your car), but they may be satisfactory for some soft real-time applications (e.g., wireless video streaming, home security monitoring, network intrusion detection).

The IEEE 802.11b standard defines an additional MAC protocol that is better suited to real-time applications. This protocol is called Point Coordination Function (PCF). It is intented for WLANs that are operating in infrastructure mode (see Section 3.0.0), with an Access Point (AP) to coordinate usage of the shared wireless channel amongst multiple wireless devices. This PCF mode of operation is similar to the Master/Slave mode of operation in Bluetooth scatternets and piconets. Few vendors of IEEE 802.11b products support the PCF mode of operation.

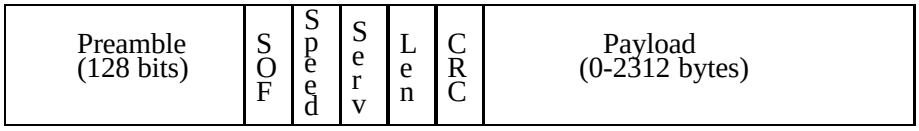

Figure 5.4. Illustration of Link-Layer Frame Format in IEEE 802.11b

PCF is a polling protocol. The AP has an explicit list of all wireless devices operating on the network, and polls (asks) each device in turn if it has any information to send. By bounding the number of devices in the network, and the maximum frame size used, the AP establishes a schedule of service with an upper bound on the channel access latency for each device. Typically, all devices in the network are equivalent in priority, so one slot of service is provided to each device in turn in the service cycle. In general, however, multiple slots can be assigned to some devices within each service cycle, to accommodate devices with different priorities or bandwidth requirements.

While the PCF protocol conceptually bounds the channel access delay for each device (by eliminating collisions from the MAC protocol), there is still the possibility of wireless channel errors (e.g., due to noise or external interference on the wireless LAN). The timeout and retransmission events in the PAR protocol can still occur, which in turn implies that there is no deterministic guarantee on when (or if) a given data frame is successfully transmitted on the WLAN. In other words, IEEE 802.11b WLANs are not a perfect solution for hard real-time applications.

Link-Layer Frame Formats

The purpose of the MAC sublayer protocol in the previous section is to ensure that at most one station is transmitting a frame on the WLAN at a time. The purpose of the Logical Link Control (LLC) sublayer above the MAC is to ensure that the frame is sent reliably.

The IEEE 802.11b link layer protocol uses variable length data frames, with the frame format illustrated in Figure 5.4. Two different versions of this frame format are supported, called the Long Preamble and the Short Preamble. The choice between these formats is usually software-settable in the configuration parameters of an IEEE 802.11b product.

The default frame format in IEEE 802.11b is the Long Preamble format. The transmission of a frame begins with 128 bits of preamble: an alternating bit pattern of 1's and 0's that is used to establish signal clocking and synchronization between sender and receiver. The preamble is followed by a 16-bit Start Of

Frame (SOF) delimiter. This bit pattern is "10101010 10101011". The last two bits of this field tell the receiver that the important control header of the frame is about to begin.

The Physical Layer Convergence Protocol (PLCP) header at the LLC layer is 48 bits long, and has four fields. The first 8-bit field indicates the data rate (Signal Speed) that will be used for the payload portion of the frame transmission. The valid choices are 1 Mbps, 2 Mbps, 5.5 Mbps, and 11 Mbps. The second 8-bit field specifies an optional Service Type, which is currently unused in IEEE 802.11b. The third field (16 bits) indicates the size in bytes of the payload portion of the frame. The fourth field is a 16-bit Cyclic Redundancy Check (CRC) for the frame.

The preamble and the PLCP header of the LLC frame are transmitted at 1 Mbps. This feature makes the IEEE 802.11b protocol backwards-compatible with older IEEE 802.11 devices that might be part of the network. While these devices would be unable to receive frames at any of the higher data rates, they can at least listen to the network and correctly determine when frames begin and end. The payload portion of an LLC frame is transmitted using the data rate indicated in the PLCP header of the frame.

The Short Preamble version differs from the foregoing format in two ways. First, the length of the preamble is reduced to 56 bits instead of 128 bits. This change makes the entire preamble (including the SOF delimiter) 72 bits instead of 144 bits, reducing the amount of channel time consumed by the preamble. Second, the 48-bit PLCP header is transmitted at 2 Mbps instead of at 1 Mbps. Again, this slightly reduces the time consumed on the network by each frame. The payload portion of an LLC frame is transmitted using the data rate (1, 2, 5.5, or 11 Mbps) that was indicated in the PLCP header of the frame.

Regardless of the frame format used, the payload portion of the frame contains additional control information (e.g., 48-bit MAC address of the sender, 48-bit MAC address of the intended receiver) plus the useful data, if any (e.g., a TCP/IP packet carrying user-level data). The maximum payload size is 2312 bytes. Many implementations use a Maximum Transmission Unit (MTU) size of 1500 bytes at the network layer, to be compatible with Ethernet LANs.

The combined overheads of the link-layer frame format and the channel access protocol limit the effective throughput that can be achieved over IEEE 802.11b WLANs. A general rule of thumb is that about 60% of the stated physical layer data rate can be achieved as user-level throughput at the application layer. For IEEE 802.11b, this means that about 6.5 Mbps of throughput is possible for TCP/IP. For IEEE 802.11a, the corresponding value is about 32 Mbps. A paper by Jun *et al.* [JPS 2003] provides a careful analysis of the maximum throughput that is theoretically possible for IEEE 802.11 networks.

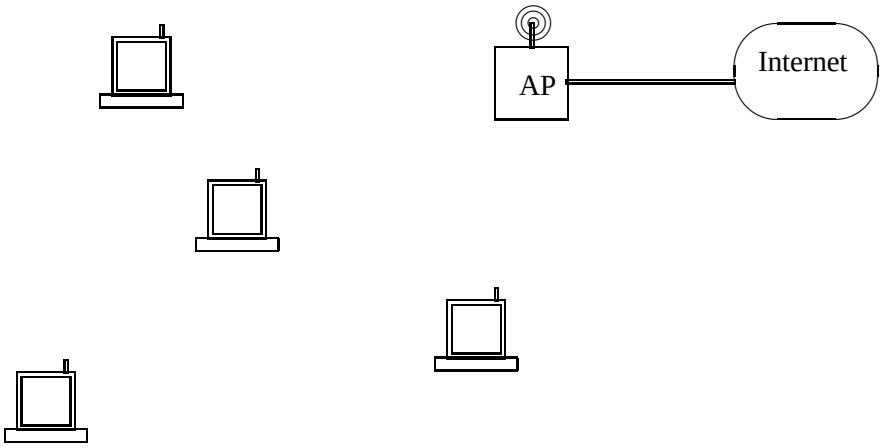

Figure 5.5. Illustration of an IEEE 802.11b WLAN in Infrastructure Mode

Other Features

There are two different ways to use IEEE 802.11b WLANs: *infrastructure mode*, and *ad hoc mode*.

Infrastructure Mode.. Infrastructure mode requires an Access Point (AP) that functions as a gateway or bridge between the wireless access network and the general wired Internet. Figure 5.5 shows a WLAN operating in infrastructure mode. The AP has two network interfaces: one for transmitting and receiving information on the WLAN, and one for transmitting and receiving information on the wired network, such as an Ethernet LAN.

In infrastructure mode, all communication to and from a mobile wireless device must traverse the AP. For example, if the mobile device wishes to access content from the general Internet, then the request is first transmitted to the AP, which forwards the request to the Internet on behalf of the mobile device. When the response returns to the AP from the wired Internet, the AP transmits the response on the WLAN as wireless data frames addressed to the client that initiated the request. Similarly, if mobile device A in the WLAN wants to communicate with mobile device B also in the WLAN, the request must be relayed via the AP. That is, A sends a frame to the AP, which acknowledges its successful delivery. Then the AP transmits the frame to B, which acknowledges its successful delivery. In this communication scenario, the frames between A and B are transmitted twice on the WLAN: once to the AP, and once by the AP.

This can compromise the efficiency of the WLAN. (In ad hoc mode described below, nodes A and B can communicate directly with each other.)

The AP plays a central role in an infrastructure-based WLAN. An AP advertises its presence on the WLAN by broadcasting *beacon frames*, typically every 100 milliseconds (i.e., 10 times per second). These management (control) frames identify the AP, its MAC address, its Service Set Identifier (SSID), as well as whether it is using WEP (Wired Equivalent Privacy) encryption or not. These frames are broadcast omni-directionally by the AP, so that all wireless devices within the coverage area (cell) of the AP can detect its presence. Wireless devices in the cell, such as client laptops with IEEE 802.11b WLAN cards, can detect this signal and associate with the AP if desired. The nominal coverage area for an AP has a radius of about 100 meters, but the quality of coverage may vary depending on building construction materials (e.g., reinforced concrete, tinted windows, metallic coatings on glass). This range can be significantly extended with directional antennas, if desired.

Typically, the deployment of IEEE 802.11b WLANs requires multiple APs (e.g., for a university residence network, a campus-wide wireless network, or a WLAN in a business organization) [Bennington et al. 1997, Kotz et al. 2002, Schwab et al. 2004, TB 1999, Tang et al. 2000]. To reduce interference, adjacent APs in the logical network topology can be configured to use different channel numbers in the portion of the electromagnetic spectrum used by IEEE 802.11b. In North America, there are eleven such channels to choose from, with each occupying about 22 MHz of spectral bandwidth. However, many of these channels partially overlap. Only channels 1, 6, and 11 are sufficiently well-spaced to be non-overlapping. These three channel numbers are commonly used in WLAN deployments. A mobile user can roam through a composite of multiple WLAN cells. If a laptop detects beacon signals from multiple APs, then the laptop can select the AP with the highest signal strength. Conceptually, the handoff from one AP to the next AP should be seamless and invisible to the user. Smooth handoffs are not always the case in practice, particularly if APs from multiple vendors or if IP subnetting are being used [Balachandran et al. 2002, Kotz et al. 2002, Schwab et al. 2004].

Ad Hoc Mode.. The second mode of operation supported by IEEE 802.11b is *ad hoc mode*. In this mode, a collection of wireless devices can operate together as a standalone wireless network, completely independent of the Internet. That is, there is no AP required for access to the general Internet. This standalone mode of operation is often used for special purposes, such as military network applications, sensor networks, peer-to-peer networks, or multi-player wireless gaming. An example of an ad hoc network is shown in Figure 5.6.

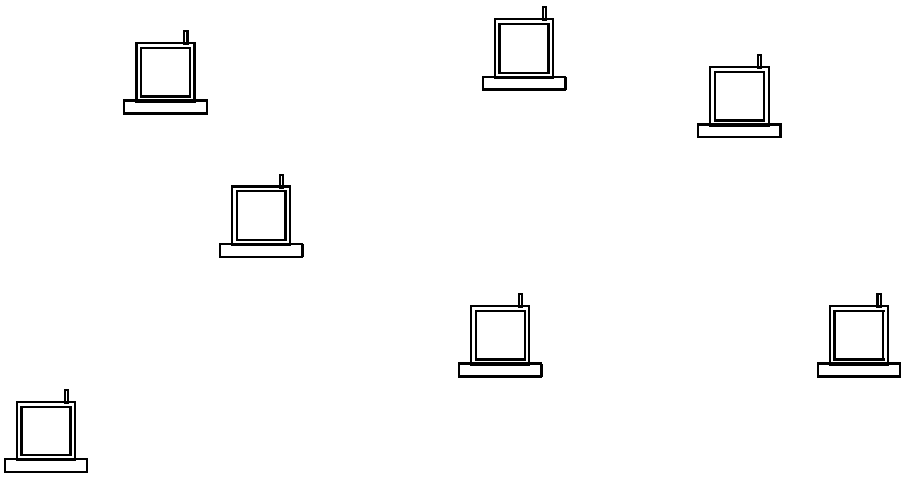

Figure 5.6. Illustration of an IEEE 802.11b WLAN in Ad Hoc Mode

In ad hoc mode, every wireless device is an equal peer with other wireless devices. Channel access is regulated using the DCF protocol most of the time, with the use of RTS/CTS to resolve hidden node problems, if they occur.

Ad hoc networks are especially interesting when they are *multi-hop* wireless ad hoc networks [gerla, Singh et al. 2002, WTLS 2002]. Suppose that there are hundreds of wireless devices in the WLAN, and that the geographic coverage area of the entire WLAN is far greater than the transmission range of any single wireless device (perhaps because of transmit power limitations to conserve energy and extend battery lifetime). In such a scenario, a device A at one location of the network (e.g., the left edge) may not be able to communicate directly with another device Z elsewhere in the network (e.g., the right edge).

Communication between A and Z can only be realized by having other nodes (e.g., B, C, D. . .) function as intermediate routers for forwarding frames on behalf of other nodes. This is the principle upon which multi-hop ad hoc networks are based. The routing protocols used in such networks are complicated, since they must dynamically determine valid routing paths to the intended destination. Nodes can move at any time, changing the topology of the network, and thus changing the quality of the routes used. Some routes may break, and other routes (better or worse) may become available at any time.

In addition to the routing problem, multi-hop ad hoc networks induce other performance problems for wireless Internet protocols. For example, the end-to-end performance of TCP degrades significantly over multi-hop ad hoc networks,

because of contention and collision problems. Problems such as these are discussed in more detail in Section 5, following the TCP review in the next section.

4. The Web and TCP/IP

This section provides background on the Web and TCP/IP. An understanding of these protocols is required prior to the discussion of the wireless Internet protocol performance problems in Section 5.

The Web

The Web relies primarily on three communication protocols: IP, TCP, and HTTP. The Internet Protocol (IP) is a connection-less network-layer protocol that provides global addressing and routing on the Internet. The Transmission Control Protocol (TCP) is a connection-oriented transport-layer protocol that provides end-to-end data delivery across the Internet. Among its many functions, TCP has flow control, congestion control, and error recovery mechanisms to provide reliable data transmission between sources and destinations. The robustness of TCP allows it to operate in many network environments. Finally, the Hyper-Text Transfer Protocol (HTTP) is a request-response application-layer protocol layered on top of TCP. HTTP is used to transfer Web documents between Web servers and Web clients (browsers). Currently, HTTP/1.0 and HTTP/1.1 [RFC1945, RFC2616] are widely used on the Internet.

TCP Overview

TCP is a connection-oriented, end-to-end reliable-byte-stream transport-layer protocol [stevens, tanenbaum]. The basic mechanisms of this general-purpose protocol are quite robust: the TCP protocol has undergone relatively minor changes in its 30-year existence, over a time period that has witnessed dramatic changes in computing and networking technologies.

The basic unit of data transfer in TCP is a *byte* (i.e., for sequence numbering, flow control, and error control purposes). However, TCP implementations generally work with a larger logical unit size called a *segment* when transmitting packets across an IP internetwork. The Maximum Segment Size (MSS) is a settable parameter for TCP. The choice of the MSS depends on the Maximum Transmission Unit (MTU) size supported by the underlying network layer. In most cases, each TCP segment is carried in one IP packet; hence the terms segment and packet are often used interchangeably. The task of TCP is to divide the application-layer data into one or more segments, transmit them across the network, and deliver them reliably (and in order) to the receiving TCP. Each segment carries an explicit sequence number, for the purposes of ordering and reliability.

There are several mechanisms in TCP to ensure reliable packet delivery. For example, when a sender transmits a segment, it sets a timer. If this segment is received successfully, then the receiver sends back an acknowledgement (ACK). TCP ACKs are cumulative, and always indicate the next expected TCP sequence number. The sender uses the ACK for flow control and error control purposes, as well as to estimate the round-trip time (RTT) to the destination. If the timer expires before an ACK is received, then the sender retransmits the outstanding segment. Another commonly used strategy is *Fast Retransmit* [floyd], which uses duplicate ACKs to trigger the retransmission of a missing segment, often well before the retransmission timer expires. This approach works well in recovering from single packet losses [fall].

TCP uses sliding window flow control to limit the maximum number of bytes outstanding (i.e., not yet acknowledged) between a sender and a receiver at any time. A sender is allowed to transmit the segments in a window as quickly as it wishes, providing that data is available to transmit. As ACKs are received, the flow control window advances, and new segments are transmitted.

A congestion control mechanism was added to TCP in 1988, based on algorithms proposed by Jacobson [jacobson]. These algorithms use adaptive window-based flow control to achieve congestion control, since the IP network layer in the Internet does not provide congestion control.

In TCP congestion control, the flow control window size is adjusted dynamically based on two TCP state variables: the congestion window (*cwnd*), and the slow-start threshold (*ssthresh*). The initial value of *cwnd* is one segment, and *cwnd* is increased as successful ACKs are received. The increase is exponential in the slow-start phase (i.e., doubling *cwnd* every RTT, until *ssthresh* is reached), and linear in the congestion avoidance phase (i.e., increasing *cwnd* by one segment for every complete window's worth of data exchanged) [jacobson].

TCP uses packet loss (due to buffer overflow at a router) as an implicit signal of network congestion. Each time a packet loss is detected, TCP updates its estimate of the slow-start threshold (e.g., $ssthresh = cwnd/2$), reduces its congestion window size (e.g., $cwnd = MSS$), and re-enters the slow-start phase. The *Fast Recovery* [floyd] mechanism reduces the congestion window size by half (e.g., $cwnd = cwnd/2$) following a Fast Retransmit, rather than reducing it to one segment.

The foregoing algorithms are part of most TCP implementations, including Reno TCP and New Reno TCP that are widely used on the Internet today [floyd].

5. Protocol Performance Issues

Many interesting protocol interactions occur when TCP/IP Internet applications are carried over wireless access networks such as IEEE 802.11b WLANs.

This section discusses four examples of these Wireless Internet protocol performance problems.

Wireless TCP Performance

A well-documented TCP performance problem occurs if packets are lost in wireless networks [itcp, wirelesstcp2, wirelesstcp1, fahmy, victor, survey]. Most TCP implementations use packet loss as an implicit signal of network congestion, and use backoff mechanisms to reduce the packet load offered to the network. This design assumption is clearly articulated in the TCP slow start congestion control mechanism [jacobson]. This approach works well for the wired Internet, since losses of packets due to congestion dominate packet losses due to transmission errors. In wireless networks, however, the situation is reversed: losses due to transmission errors dominate congestion losses. Upon losing a packet due to a transmission error, the desired behaviour in a WLAN is to retransmit right away, but a conventional TCP implementation instead experiences timeout and backoff. This behaviour can lead to very low TCP throughput on wireless networks [wirelesstcp2].

One solution to this wireless TCP performance problem is to implement a "wireless-aware" version of TCP at the boundary between the wired backbone network and the wireless access network. This approach (referred to as Proxy-TCP, Indirect-TCP, or Snoop-TCP in the literature) logically splits the end-to-end TCP control loop into two smaller control loops, with one covering the wired segment of the network, and the other the wireless segment of the network. The parameters for these two control loops can be set and tuned separately, with the intermediate TCP "proxy" handling the buffering and forwarding of packets between the communicating TCP endpoints. While this approach technically violates the end-to-end semantics of TCP acknowledgements, it is effective in improving TCP performance over wireless network environments [wirelesstcp1].

Our own network protocol performance research at the University of Calgary has identified three additional TCP-related performance problems in IEEE 802.11b wireless LANs [KXW 2003]. These problems are:

- low throughput from improperly configured wireless network cards;
- network thrashing from dynamic MAC-layer rate adaptation; and
- high collision rates between TCP data and acknowledgement packets.

In the following paragraphs, we briefly describe each of these protocol performance problems. Further details are provided in [KXW 2003].

The first anomaly that we observed in our IEEE 802.11b WLAN was low TCP throughput for bulk data transfers. The throughput was approximately 1.2 Mbps, compared to the expected value of approximately 6.0 Mbps. The primary

cause for this problem was an improperly configured network card. In particular, the Linux device driver for the IEEE 802.11b card was statically setting the card's transmission rate to 2 Mbps rather than 11 Mbps. We diagnosed this problem with the help of a wireless network analyzer: the PLCP header of each transmitted frame showed a signal speed of 2 Mbps. Since the frame header is transmitted at 1 Mbps, and the payload at 2 Mbps, it is no surprise that our throughput barely exceeded 1 Mbps. Fixing the device driver setting increased throughput by about a factor of 4. A related, but much more subtle performance problem has to do with the Universal Serial Bus (USB) protocol used to transfer TCP/IP packets from the kernel to the network card or vice versa. Because the USB transfer size is much smaller than the TCP/IP packet size, the USB overhead can dominate. For the 8 different USB configurations considered in [KXW 2003], the TCP throughput varied by more than an order of magnitude, from 0.4 Mbps to 5.2 Mbps. For many of these configurations, the USB was the bottleneck. Only in a few of the configurations did the throughput approach the theoretical capacity of an IEEE 802.11b WLAN. These observations highlight the importance of proper parameter configuration at all layers of the protocol stack.

The second performance anomaly that we observed was related to the dynamic rate adaptation feature in IEEE 802.11b. That is, if the link-layer protocol detects an excessive number of retransmissions when using high data rate (11 Mbps) transmissions, it can revert to lower-rates (e.g., 5.5 Mbps, 2 Mbps, or 1 Mbps) on subsequent retransmissions, which may be more resilient to wireless channel errors. We tested this rate adaptation behaviour in a WLAN environment with poor signal quality and user mobility. We found that some implementations of dynamic rate adaptation produce a cyclic pattern of rate oscillation, which in turn results in frequent periodic TCP packet losses. Greater hysteresis is required in the channel quality estimation if dynamic rate adaptation is to perform well in a noisy WLAN environment.

The third and final performance anomaly that we noticed was an excessive number of collisions on the WLAN when the TCP protocol is used. For example, with UDP for wireless data transfers, collision rates are below 0.5% on the WLAN. With TCP, the collision rates are 4-7%. What is particularly surprising about these collision rates is that they occur with only a *single* wireless client (communicating with a server on a wired network), competing with the AP for use of the wireless channel. We have shown experimentally that this high collision rate is related to TCP: namely, the contention between TCP data packets sent by the client and TCP ACK packets being forwarded by the AP. The design of the IEEE 802.11b MAC protocol assumes that stations generate frames at random times. With TCP, this assumption is not true. A sending station often has a backlog of frames to send, as soon as the channel access protocol permits. Similarly, TCP ACK packets are traversing the reverse path in the network,

contending for the channel at the wireless access link. The TCP-level ACKs induce correlated behaviours on the network, leading to the higher collision rates observed. Fortunately, the random backoff in the MAC-layer retransmission resolves most of these collisions without causing TCP packet loss, but the efficiency of the IEEE 802.11b WLAN still suffers.

Multi-hop Ad Hoc Networks

TCP performance can suffer greatly in multi-hop wireless ad hoc networks, where intermediate wireless nodes are used as routers for forwarding packets from a source to a destination. Performance problems arise from the physical characteristics of the wireless channel, the Medium Access Control (MAC) protocols, node mobility, and the dynamics of TCP.

Ignoring node mobility for the moment, even the topology of an ad hoc network and the wireless propagation characteristics can adversely affect TCP performance. For example, Fu *et al.* [gerla] use simulation and analysis to show that in an N-hop "chain" network topology, the end-to-end TCP throughput is much lower than the nominal throughput of the wireless channel. The problems occur because of the burstiness of TCP transmissions: multiple TCP data packets within the congestion window are competing for channel access on the forward path, multiple TCP ACK packets are competing for channel access on the reverse path, and interference effects at the wireless layer (e.g., hidden node, exposed node) preclude adjacent nodes on the routing path from forwarding packets at the same time. Fu *et al.* recommend a carefully chosen TCP window size that depends on the routing path length (e.g., a window size of N/4 packets for an N-hop chain topology gives the maximal TCP throughput [gerla]). Other authors propose rate-based flow control or multi-channel approaches to expedite TCP data transfer [multichannel, WTLS 2002].

Node mobility in ad hoc networks can make the TCP performance problem even worse. In addition to network congestion and wireless channel errors, node mobility can produce a *third* type of packet loss: losses due to transient routing failures when IP routing paths are disrupted. The TCP protocol has no way to differentiate these types of packet losses. Some authors have proposed Explicit Loss Notification (ELN) [wirelesstcp2], so that TCP packet transmissions are suspended while the IP route discovery process is re-initiated.

Wireless Media Streaming

The popularity of multimedia streaming on the Internet, combined with the growing deployment of wireless access networks, augurs the converging usage of these two technologies in the not-too-distant future. Experience with wireless multimedia streaming on today's networks can provide valuable insights into the design of future wireless multimedia networks and applications.

In a recent paper [KW 2002], we presented a measurement study of RealMedia streaming traffic on an indoor IEEE 802.11b wireless LAN. The traffic is analyzed hierarchically, from the application layer to the network layer to the data link layer. We focus on the traffic structure at each layer, and on interaction effects across layers.

Our main observation is that multimedia streaming quality is quite robust in all but the poorest channel conditions, despite the inherent burstiness of both the RealMedia application workload and wireless channel errors. Several factors contribute to these good results. First, although RealVideo is typically Variable-Bit-Rate (VBR) at the application layer, it is often streamed as Constant-Bit-Rate (CBR) at the network layer, reducing burstiness and thus the chances of packet losses due to buffer overflow in the network path. Second, while the wireless channel has bursty error characteristics, MAC-layer retransmission in 802.11b hides most errors from higher-layer protocols. Finally, the application layer's NACK-based error control is effective in recovering missing packets when needed. Our results demonstrate the viability of multimedia streaming on current and future wireless LANs.

Wireless Web Performance

Two of the most popular Internet technologies from the past ten years are the World Wide Web and wireless networks. A natural step in the wireless Internet evolution is the convergence of these technologies to form the "wireless Web": the wireless classroom, the wireless campus, the wireless office, and the wireless home. In fact, the same technology that allows Web clients to be mobile (i.e., wireless network cards) also enables the deployment of wireless Web servers.

Mobile Web servers play a useful role in *short-lived networks*. A short-lived (or *portable*) network is created spontaneously, in an *ad hoc* fashion, at a particular location in response to some event (scheduled or unscheduled). The network operates for some short time period (minutes to hours), before being disassembled, moved, and reconstituted elsewhere. Examples of deployment scenarios for short-lived networks are sporting events, disaster recovery sites, press conferences, conventions and trade shows, and classroom area networks. The potential for entertainment applications (e.g., media streaming, home networking, multi-player gaming) is also high. In many of these contexts, an ad hoc wireless network (with a wireless Web server as an information repository) provides a suitable solution.

In recent work [BOW 2003, BW 2003, Ola 2003], we have explored the feasibility of wireless Web servers. In [BW 2003], we present simulation results that are validated with empirical measurements from wireless Web server usage in a classroom environment. These measurements are then augmented with laboratory tests to determine experimentally the upper bounds on achievable

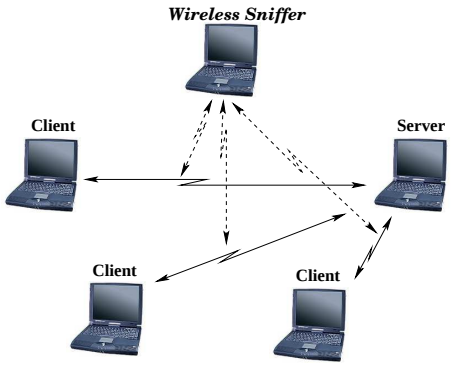

Figure 5.7. Experimental Setup for Wireless Web Server Benchmarking

performance [BOW 2003, Ola 2003]. In particular, we focus on the performance capabilities of an Apache Web server running on a laptop computer with an IEEE 802.11b wireless LAN interface.

The experimental setup for our measurements is illustrated in Figure 5.7. We study in-building Web performance for wireless Web clients. All mobile computers are configured in ad hoc mode, since no existing network infrastructure is assumed. The clients download content from the wireless Web server. A wireless network analyzer is used to collect and analyze traces from the experiments, with traffic analysis spanning from the Medium Access Control (MAC) layer to HTTP at the application layer.

Our experiments focus on the HTTP transaction rate and end-to-end throughput achievable in an ad hoc wireless network environment, and the impacts of factors such as number of clients, Web object size, and persistent HTTP connections. The results show the impacts of the wireless network bottleneck, either at the client or the server, depending on the Web workload. Persistent HTTP connections offer significant improvements both in throughput and in fairness for mobile clients accessing content from a wireless Web server.

There are three main observations from our wireless Web server experiments:

- *TCP is an extremely "chatty" protocol for wireless Web access.* An example of its behaviour is shown in Figure 5.8(a). Downloading a single 1 KB Web object using HTTP/1.0 requires 10 TCP packets on the network, with 6 sent by the client and 4 by the server. Only 2 of these packets carry actual user-level data: the client's GET request that specifies the desired URL, and the server's HTTP response with the Web object data. The other packets are TCP control packets to establish, maintain, update, and close TCP connection state information. This 80% protocol overhead has dire performance impacts when WLAN channel access is the bottleneck. In our experiments, the Web server can achieve

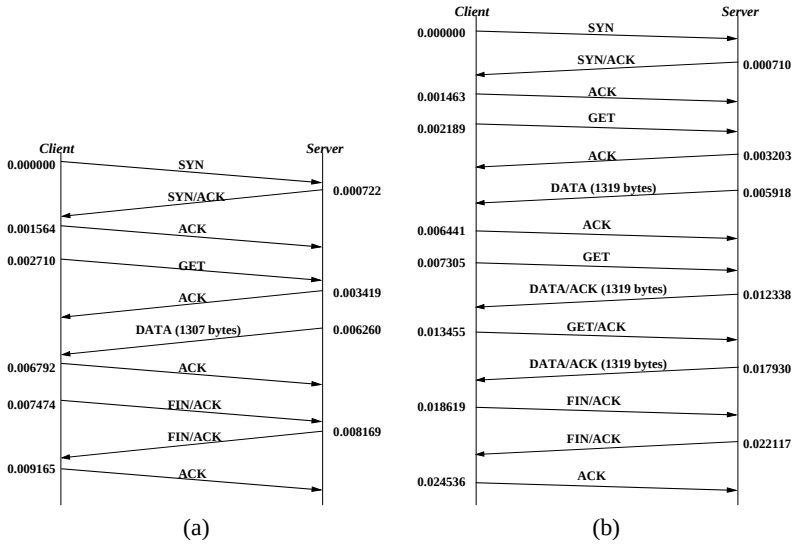

Figure 5.8. Example of a 1 KB HTTP Transaction on IEEE 802.11b WLAN: (a) HTTP/1.0; (b) HTTP/1.1

only about 100 HTTP/1.0 transactions per second for 1 KB objects. The user-level throughput is below 1 Mbps.

Persistent-HTTP connections [PM 1995] improve performance by a factor of 3 to 5 in the WLAN environment. The performance advantages arise because the TCP connection handshaking packets are amortized over multiple HTTP transactions. For example, the first HTTP transaction within the TCP connection in Figure 5.8(b) requires 4 TCP packets, rather than 10. The second HTTP transaction requires only 2 TCP packets, since it takes advantage of TCP ACK piggybacking on outbound data packets in each direction. The same observation applies for the third HTTP transaction to retrieve another embedded object in the Web page. This TCP efficiency dramatically reduces the demand on the wireless channel access protocol, leading to much faster HTTP response time and better network throughput.

- *The wireless network bottleneck manifests itself differently, depending on the Web workload*. For small Web objects, the bottleneck is at the client: there is a finite limit on the HTTP request rate that can be achieved before packets are lost from the link-layer transmit queue at the client. For large Web objects, the bottleneck is at the server, since it sends more packets (and larger packets) than the client.

- *TCP behaviour is erratic under overload.* We have observed two anomalous TCP phenomena. For multiple clients requesting small Web objects from the server, the bottleneck at the server's outgoing transmit queue can lead to TCP packet loss. The loss of TCP connection handshake packets can severely affect clients, even leading to unfairness. For large Web objects, multiple clients can cause *network thrashing*, wherein the WLAN is so busy sending TCP data packets and retransmissions that clients eventually timeout and abort their HTTP transfers, drastically reducing the goodput of the network.

All three of these problems manifest themselves acutely in an IEEE 802.11b WLAN environment [BOW 2003].

6. Summary and Outlook

Wireless Internet technologies have progressed tremendously in the past five years, and will continue to reshape the networking landscape in the years ahead. The IEEE 802.11b wireless LAN standard has been a major part of the success story. This WiFi standard provides a flexible and cost-effective solution for wireless network access, while supporting mobile users in either the infrastructure mode or the ad hoc mode of operation. The user-level achievable throughput on IEEE 802.11b WLANs is approximately 6 Mbps, about an order of magnitude faster than on previous generation wireless technologies.

The future of wireless Internet is even brighter. The IEEE 802.11a standard promises up to 54 Mbps of physical-layer transmission capacity, in a much less crowded portion of the electromagnetic spectrum. This technology will offer about 32 Mbps of user-level throughput for TCP/IP networking applications. Next-generation wireless networking technologies promise greater software support, easier network configuration, better network security, and lower prices.

Wireless network access technologies will soon become an invisible part of our ubiquitous computing infrastructure. Nevertheless, protocol performance issues will continue to be a problem. Understanding these protocol performance problems, and solving them, will be important to maximize the benefits of Wireless Internet technologies.

Acknowledgements

Financial support for this research was provided by iCORE (Informatics Circle of Research Excellence) in the Province of Alberta, and by the Natural Sciences and Engineering Research Council of Canada, through NSERC Research Grant OGP0121969. The author is grateful to iCORE, NSERC, the Canada Foundation for Innovation (CFI), and the University of Calgary for help in establishing the Wireless Internet Performance Laboratory (WIPL) and the Experimental Laboratory for Internet Systems and Applications (ELISA).

The author thanks his iCORE research team and graduate students for their many contributions to Wireless Internet protocol performance research. Special thanks go to Guangwei Bai and Kenny Oladosu for their work on wireless Web servers, and for contributing several of the diagrams in this tutorial article. Tianbo Kuang and Nayden Markatchev initiated most of the work on wireless media streaming, and Fang (Shelly) Xiao has spent many hours studying wireless TCP behaviours. For all of these efforts, I am grateful.

References

ANSI/IEEE Standard 802.11b, "Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specificiations: Higher-Speed Physical Layer Extension in the 2.4 GHz band", 1999.

G. Bai, K. Oladosu, and C. Williamson, "Portable Networks: Prototype and Performance", submitted for publication, 2003.

G. Bai and C. Williamson, "Simulation Evaluation of Wireless Web Performance in an IEEE 802.11b Classroom Area Network", *Proceedings of the Third International Workshop on Wireless Local Networks* (WLN 2003), Bonn, Germany, pp. 663-672, October 2003.

A. Bakre and B. Badrinath, "I-TCP: Indirect TCP for Mobile Hosts", *Proceedings of the 15th International Conference on Distributed Computing Systems* (ICDCS), Vancouver, BC, pp. 136-143, May 1995.

A. Balachandran, G. Voelker, P. Bahl, and P. Rangan, "Characterizing User Behavior and Network Performance in a Public Wireless LAN", *Proceedings of ACM SIGMETRICS*, Marina Del Rey, CA, pp. 195-205, June 2002.

H. Balakrishnan and R. Katz, "Explicit Loss Notification and Wireless Web Performance", *Proceedings of IEEE GLOBECOM*, November 1998.

H. Balakrishnan, V. Padmanabhan, S. Seshan, and R. Katz, "A Comparison of Mechanisms for Improving TCP Performance over Wireless Links", *IEEE/ACM Transactions on Networking*, Vol. 5, No. 6, pp. 756-769, December 1997.

B. Bennington and C. Bartel, "Wireless Andrew: Experience Building a High Speed, Campus-Wide Wireless Data Network", *Proceedings of ACM MOBICOM*, Budapest, Hungary, pp. 55-65, September 1997.

S. Fahmy, V. Prabhakar, S. Avasarala, and O. Younis, "TCP over Wireless Links: Mechanisms and Implications", Technical Report CSD-TR-03-004, Purdue University, 2003.

K. Fall and S. Floyd, "Simulation-based Comparisons of Tahoe, Reno, and SACK TCP", *ACM Computer Communication Review*, Vol. 26, No. 3, pp. 5-21, July 1996.

S. Floyd, "A Report on Recent Developments in TCP Congestion Control", *IEEE Communications*, Vol. 39, No. 4, pp. 84-90, April 2001.

Z. Fu, P. Zerfos, H. Luo, S. Lu, L. Zhang, and M. Gerla, "The Impact of Multihop Wireless Channel on TCP Throughput and Loss", *Proceedings of IEEE INFOCOM*, San Francisco, CA, April 2003.

V. Jacobson, "Congestion Avoidance and Control", *Proceedings of ACM SIGCOMM*, Stanford, CA, pp. 314-329, August 1988.

J. Jun, P. Peddabachagari, and M. Sichitiu, "Theoretical Maximum Throughput of IEEE 802.11 and its Applications", *Proceedings of the 2nd IEEE International Symposium on Network Computing and Applications* (NCA'03), Cambridge, MA, pp. 249-256, April 2003.

D. Kotz and K. Essein, "Analysis of a Campus-Wide Wireless Network", *Proceedings of ACM MOBICOM*, Atlanta, GA, September 2002.

T. Kuang and C. Williamson, "RealMedia Streaming Performance on an IEEE 802.11b Wireless LAN", *Proceedings of IASTED Wireless and Optical Communications Conference* (WOC 2002), Banff, AB, pp. 306-311, July 2002.

T. Kuang and C. Williamson, "A Bidirectional Multichannel MAC Protocol for Improving TCP Throughput in Multihop Wireless Ad Hoc Networks", submitted for publication, 2003.

T. Kuang, F. Xiao, and C. Williamson, "Diagnosing Wireless TCP Performance Problems: A Case Study", *Proceedings of SCS SPECTS Conference*, Montreal, PQ, July 2003.

V. Li, Z. Liu, and S. Low, "Enhancing TCP Performance over Wireless Networks", *Proceedings of the IST Mobile and Wireless Telecommunications Summit*, pp. 85-89, Thessaloniki, June 2002.

K. Oladosu, *Performance and Robustness Testing of Wireless Web Servers*, M.Sc. Thesis, University of Calgary, August 2003.

V. Padmanabhan and J. Mogul, "Improving HTTP Latency", *Computer Networks and ISDN Systems*, Vol. 28, pp. 25-35, December 1995.

K. Pentikousis, "TCP in Wired-Cum-Wireless Environments", *IEEE Communications Surveys and Tutorials*, Vol. 3, No. 4, Fourth Quarter 2000.

RFC 1945: "Hypertext Transfer Protocol – HTTP/1.0", www.ietf.org/rfc/rfc1945.txt

RFC 2616: "Hypertext Transfer Protocol – HTTP/1.1", www.ietf.org/rfc/rfc2616.txt

D. Schwab and R. Bunt, "Characterizing the Use of a Campus Wireless Network", to appear, *Proceedings of IEEE INFOCOM*, Hong Kong, April 2004.

H. Singh and P. Singh, "Energy Consumption of TCP Reno, TCP NewReno, and SACK in Multihop Wireless Networks", *Proceedings of ACM SIGMETRICS*, Marina Del Rey, CA, pp. 206-216, June 2002.

W. Stevens, *TCP/IP Illustrated, Volume 1: The Protocols*, Addison-Wesley, 1994.

A. Tanenbaum, *Computer Networks*, Fourth Edition, Prentice-Hall, 2003.

D. Tang and M. Baker, "Analysis of a Metropolitan-Area Wireless Network", *Proceedings of ACM MOBICOM*, Seattle, WA, pp. 13-23, August 1999.

D. Tang and M. Baker, "Analysis of a Local-Area Wireless Network", *Proceedings of ACM MOBICOM*, Boston, MA, pp. 1-10, August 2000.

S. Wu, Y. Tseng, C. Lin, and J. Sheu, "A Multi-Channel MAC Protocol with Power Control for Multi-Hop Mobile Ad Hoc Networks", *Computer Journal*, Vol. 45, No. 1, 2002.